A Case for Using CephFS

BJ Lougee
Software Engineer
Federal Reserve Bank of Kansas City
Center for the Advancement of Data and Research in Economics

The opinions expressed herein are those of the authors and do not reflect the views of the Federal Reserve Bank of Kansas City or Federal Reserve System
Our Dilemma

- PanFS is used in our HPC environment
- SAS is used for some very important workloads
  - Sporadic workloads but very IO intensive
- Carved out 70TB of PanFS for SAS temp workspace
- Needed to reclaim that space
  - Cheaper and/or faster parallel filesystem
  - It's only for temp/transient data
Choices

• Lustre
  • Hardware redundant
  • Open Source
  • Scales

• CephFS
  • **Software redundant**
  • Open Source
  • Scales
  • **Ceph ties in well with our OpenStack plans**
CephFS Testing

• #1 goal: What kind of minimum can I get away with?
• Various types of usage paradigms
• Tested performance between Giant and Hammer
• Tested a lot of different config options in ceph.conf
• Tested Different Journal Setups
SAS Regression

Time (mins)

CephFS

PanFS
SAS Bench Large

- CephFS
- PanFS

Time (mins)
cp 100,000 1MB files
CephFS in Production

Mon/MDS:
- 20 cores with 128 GB memory
- 40 Gb Ethernet

OSD Nodes:
- 16 Hot Swappable OSDs
- 8 Hot Swappable SSD Journals
- 40 Gb Ethernet
- 16 Cores with 64 Gb Memory
Ceph status

Every 0.5s: ceph -s

cluster a21445d2-
health HEALTH_OK
monmap e3: 3mons at {01= :6789/0, 02=up:active}, 2 up:standby
mdsmap e570: 1/1/1 up {0= 02=up:active}, 2 up:standby
osdmap e22042: 52 osds: 48 up, 48 in
pgmap v392579: 4608 pgs, 2 pools, 2564 GB data, 1328 kobjecls
7555 GB used, 167 TB / 174 TB avail
4608 active+clean
client io 260 MB/s rd, 1196 MB/s wr, 227 op/s

Every 0.5s: ceph osd pool stats

pool cephfs_metadata id 10
  client io 31550 B/s wr, 1 op/s

pool cephfs_data id 11
  client io 260 MB/s rd, 1196 MB/s wr, 726 op/s
### RECORD 3400221672 >>> 01 <<< (1446752344-8161) (Thu May 5 13:58:44 2015) ###

#### DECK STATISTICS (/sec=0)   

<table>
<thead>
<tr>
<th>Name</th>
<th>KBates Merged</th>
<th>I/O Size</th>
<th>KBates Merged</th>
<th>I/O Size</th>
<th>MBSize</th>
<th>Qlen</th>
<th>Wait SvcTlm Util</th>
<th>Pct</th>
</tr>
</thead>
<tbody>
<tr>
<td>d0</td>
<td>0 0 0 0 0 0</td>
<td>0 0 0 0</td>
<td>0 0 0 0 0 0</td>
<td>0 0 0 0</td>
<td>0 0 0 0 0 0</td>
<td>0 0 0 0 0 0</td>
<td>0 0 0 0 0 0</td>
<td>0 0 0 0 0 0</td>
</tr>
</tbody>
</table>

#### RECORD 3400221672 >>> 02 <<< (1446752344-8161) (Thu May 5 13:58:44 2015) ###

#### DECK STATISTICS (/sec=10)   

<table>
<thead>
<tr>
<th>Name</th>
<th>KBates Merged</th>
<th>I/O Size</th>
<th>KBates Merged</th>
<th>I/O Size</th>
<th>MBSize</th>
<th>Qlen</th>
<th>Wait SvcTlm Util</th>
<th>Pct</th>
</tr>
</thead>
<tbody>
<tr>
<td>d0</td>
<td>0 0 0 0 0 0</td>
<td>0 0 0 0</td>
<td>0 0 0 0 0 0</td>
<td>0 0 0 0</td>
<td>0 0 0 0 0 0</td>
<td>0 0 0 0 0 0</td>
<td>0 0 0 0 0 0</td>
<td>0 0 0 0 0 0</td>
</tr>
</tbody>
</table>

---

Collectl
Conclusion

• Did not lose any performance PanFS → CephFS
• Users have been happy with the performance
• There really is a minimum!
• Need MOARRR OSD nodes :)
Questions