Linux Open Source Distributed Filesystem
Ceph at SURFsara

Remco van Vugt

July 2, 2013
Agenda

- Ceph internal workings
  - Ceph components
  - CephFS
  - Ceph OSD
- Research project results
  - Stability
  - Performance
  - Scalability
  - Maintenance
  - Conclusion
- Questions
Ceph components

- Monitor nodes
  - (Meta Data Server nodes)
- Object Storage Device nodes
- Object store (RADOSGW)
- Distributed filesystem (CephFS)
- Block storage (RBD)
- RADOS
  - (Reliable Autonomic Distributed Object Store)
- LIBRADOS (library)
- OSD daemons (12 per node)
CephFS

- Fairly new, under heavy development
- POSIX compliant
- Can be mounted through FUSE in userspace, or by kernel driver
CephFS (2)

Figure: Ceph state of development
CephFS (3)

Figure: Dynamic subtree partitioning
Ceph OSD

- Stores object data in flat files in underlying filesystem (XFS, BTRFS)
- Multiple OSDs on a single node (usually: one per disk)
- 'Intelligent daemon', handles replication, redundancy and consistency
CRUSH

- Cluster map
- Object placement is calculated, instead of indexed
- Objects grouped into Placement Groups (PGs)
- Clients interact direct with OSDs
Placement group

Figure: Placement groups
Failure domains

Figure: Crush algorithm
Replication

Figure: Replication
Monitoring

- OSD use peering, and report about each other
- OSD either up or down
- OSD either in or out the cluster
- MON keeps overview, and distributes cluster map changes
OSD fault recovery

- OSD down, I/O continues to secondary (or tertiary) OSD assigned to PG (active+degraded)
- OSD down longer than configured timeout, OSD is down and out (kicked out of the cluster)
- PG data is remapped to other OSD and re-replicated in the background
- PGs can be down if all copies are down
Rebalancing

**Before**

<table>
<thead>
<tr>
<th>OSD 1</th>
<th>OSD 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>PG #1</td>
<td>PG #6</td>
</tr>
<tr>
<td>PG #2</td>
<td>PG #7</td>
</tr>
<tr>
<td>PG #3</td>
<td>PG #8</td>
</tr>
<tr>
<td>PG #4</td>
<td>PG #9</td>
</tr>
<tr>
<td>PG #5</td>
<td>PG #10</td>
</tr>
</tbody>
</table>

**After**

<table>
<thead>
<tr>
<th>OSD 1</th>
<th>OSD 2</th>
<th>OSD 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>PG #1</td>
<td>PG #7</td>
<td>PG #3</td>
</tr>
<tr>
<td>PG #2</td>
<td>PG #8</td>
<td>PG #6</td>
</tr>
<tr>
<td>PG #4</td>
<td>PG #10</td>
<td>PG #9</td>
</tr>
</tbody>
</table>
Research
Research questions

- Research question
  - Is the current version of CephFS (0.61.3) production-ready for use as a distributed filesystem in a multi-petabyte environment, in terms of stability, scalability, performance and manageability?

- Sub questions
  - Is Ceph, and in particular the CephFS component, stable enough for production use at SURFsara?
  - What are the scaling limits in CephFS, in terms of capacity and performance?
  - Does Ceph(FS) meet the maintenance requirements for the environment at SURFsara?
Stability

- Various tests performed, including:
  - Cut power from OSD, MON and MDS nodes
  - Pull disks from OSD nodes (within failure domain)
  - Corrupt underlying storage files on OSD
  - Killed daemon processes

- No serious problems encountered, except for multi-mds
- Never encountered data loss
Performance

- Benchmarked RADOS and CephFS
  - Bonnie++
  - RADOS bench
- Tested under various conditions:
  - Normal
  - Degraded
  - Rebuilding
  - Rebalancing
RADOS Performance

RADOS throughput

Read
Write

megabytes / sec

Normal  Degraded  Rebalancing  Rebuilding
CephFS Performance

Bonnie throughput and IOPS

- Read megabytes / sec
- Write megabytes /sec
- IOPS

Normal
Degraded
Rebalancing
CephFS MDS Scalability

- Tested metadata performance using mdtest
- Various POSIX operations, using 1000, 2000, 4000, 8000 and 16000 files per directory
- Tested 1 and 3 MDS setup
- Tested single and multiple directories
Results:

- Did not multi-thread properly
- Scaled over multiple MDS
- Scaled over multiple directories
- However...
CephFS MDS Scalability (3)
Ceph OSD Scalability

- Two options for scaling:
  - Horizontal: adding more OSD nodes
  - Vertical: adding more disks to OSD nodes
- But how far can we scale..?
## Scaling horizontal

<table>
<thead>
<tr>
<th>Number of OSDs</th>
<th>PGs</th>
<th>MB /sec</th>
<th>max (MB /sec)</th>
<th>Overhead %</th>
</tr>
</thead>
<tbody>
<tr>
<td>24</td>
<td>1200</td>
<td>586</td>
<td>768</td>
<td>24</td>
</tr>
<tr>
<td>36</td>
<td>1800</td>
<td>908</td>
<td>1152</td>
<td>22</td>
</tr>
<tr>
<td>48</td>
<td>2400</td>
<td>1267</td>
<td>1500</td>
<td>16</td>
</tr>
</tbody>
</table>
Scaling vertical

- OSD scaling
  - Add more disks, possibly using external SAS enclosures
  - But, each disk adds overhead (CPU, I/O subsystem)
Scaling vertical (2)
Scaling vertical (3)
Scaling OSDs

- Scaling horizontal seems no problem
- Scaling vertical has it’s limits
  - Possibly tunable
  - Jumbo frames?
Maintenance

- Built in tools sufficient
- Deployment
- Crowbar
- Chef
- Ceph deploy
- Configuration
- Puppet
Research (2)

Research question

- Is the current version of CephFS (0.61.3) production-ready for use as a distributed filesystem in a multi-petabyte environment, in terms of stability, scalability, performance and manageability?

Sub questions

- Is Ceph, and an in particular the CephFS component, stable enough for production use at SURFsara?
- What are the scaling limits in CephFS, in terms of capacity and performance?
- Does Ceph(FS) meet the maintenance requirements for the environment at SURFsara?
Conclusion

- Ceph is stable and scalable
  - RADOS storage backend
  - Possibly: RBD and object storage, but outside scope
- However: CephFS is not yet production ready
  - Scaling is a problem
  - MDS failover was not smooth
  - Multi-MDS not yet stable
  - Let alone directory sharding
- However: developer attention back on CephFS
Conclusion (2)

- Maintenance
  - Extensive tooling available
  - Integration into existing toolset possible
  - Self-healing, low maintenance possible
Questions?