Distributed file storage

By Jeroen Klaver & Roel van der Jagt
Content

- Background
- Research question
- Test approach
- SURFnet design
- Conclusion
- Future work
Background

- SURFnet is the Dutch NREN
- Storage explosion
Background

- Add more storage
Background

- And more...

- But adding includes high costs and new bottlenecks
SURFnet thinks there is a smarter way
Background

Price

Capacity

Speed/reliability
Main question

- What infrastructure and open source tools provide SURFnet or the participants a scalable and distributed any-kind-storage solution?
Product inventory

- Coda
- Lustre
- GlusterFS
- XtreemFS
- Ceph
- PVFS
- MooseFS
- Hadoop
Test approach

- First stage
  - Null test
- Second stage
  - Architectures in the lab environment
- Third stage
  - Architectures in practice
Lustre Test Approach
Lustre Performance

Iozone - Lustre file system benchmark

<table>
<thead>
<tr>
<th>Block size (KBytes)</th>
<th>Speed (KBytes/sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>256 8192</td>
<td>Sequential read</td>
</tr>
<tr>
<td>256 8192</td>
<td>Sequential write</td>
</tr>
<tr>
<td>256 8192</td>
<td>Random read</td>
</tr>
<tr>
<td>256 8192</td>
<td>Random write</td>
</tr>
</tbody>
</table>

- Lustre avg
- Null test avg
Lustre Performance – Latency impact

Iozone - Lustre with latency

<table>
<thead>
<tr>
<th>Block size (KBytes)</th>
<th>Sequential read</th>
<th>Sequential write</th>
<th>Random read</th>
<th>Random write</th>
</tr>
</thead>
<tbody>
<tr>
<td>256 8192</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>256 8192</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>256 8192</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>256 8192</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- No added latency avg
- Latency (20ms RTT) avg
Coda Test Approach

Diagram showing a network of servers and clients. The diagram includes:

- Coda1 File server
  - SCM Replica 0
  - Coda Client1 with local cache

- Replicated volume

- Coda2 File server Replica 1
  - Coda Client2 with local cache
Coda Performance

Iozone - Coda file system benchmark

<table>
<thead>
<tr>
<th>Block size (KBytes)</th>
<th>Sequential read</th>
<th>Sequential write</th>
<th>Random read</th>
<th>Random write</th>
</tr>
</thead>
<tbody>
<tr>
<td>256</td>
<td>52000</td>
<td>45000</td>
<td>40000</td>
<td>25600</td>
</tr>
<tr>
<td>8192</td>
<td>51000</td>
<td>46000</td>
<td>41000</td>
<td>25700</td>
</tr>
</tbody>
</table>

- Coda avg
- Null test avg
Coda Performance – Latency impact

Iozone - Coda with latency

<table>
<thead>
<tr>
<th>Block size (KBytes)</th>
<th>256</th>
<th>8192</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequential read</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sequential write</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Random read</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Random write</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- No added latency avg
- Latency (20ms RTT) avg
Coda vs Lustre - Performance

Iozone - File system benchmark

<table>
<thead>
<tr>
<th>Block size (KBytes)</th>
<th>Speed (KBytes/sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>256</td>
<td>Sequential read</td>
</tr>
<tr>
<td>8192</td>
<td>Sequential write</td>
</tr>
<tr>
<td></td>
<td>Random read</td>
</tr>
<tr>
<td></td>
<td>Random write</td>
</tr>
</tbody>
</table>

Lustre avg
Coda avg
Null test avg
Coda vs Lustre - Seeks

Bonnie++ - Seeks

- **Seeks (seeks/sec)**
- Bonnie++
- Lustre
- Coda
- Null test

<table>
<thead>
<tr>
<th></th>
<th>Lustre</th>
<th>Coda</th>
<th>Null test</th>
</tr>
</thead>
<tbody>
<tr>
<td>min</td>
<td>150</td>
<td>5</td>
<td>200</td>
</tr>
<tr>
<td>avg</td>
<td>250</td>
<td>5</td>
<td>600</td>
</tr>
<tr>
<td>max</td>
<td>350</td>
<td>5</td>
<td>700</td>
</tr>
</tbody>
</table>
Coda vs Lustre – Latency Performance

Iozone - Coda vs Lustre with latency

Speed (KBytes/sec)

<table>
<thead>
<tr>
<th>Block size (KBytes)</th>
<th>256</th>
<th>8192</th>
<th>256</th>
<th>8192</th>
<th>256</th>
<th>8192</th>
<th>256</th>
<th>8192</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequential read</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sequential write</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Random read</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Random write</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Lustre No added latency avg
- Lustre Latency (20ms RTT) avg
- Coda No added latency avg
- Coda Latency (20ms RTT) avg
Lustre SURFnet Design

- Requirements
  - Scalable
  - Available
  - Durable
  - Performance
  - Dynamic
  - Cost effective
  - Generic interfaces
  - Open protocols
  - Geographic dispersion
- Deployment \ Maintenance
- Security features
Coda SURFnet Design

- Requirements
  - Scalable
  - Available
  - Durable
  - Performance
  - Dynamic
  - Cost effective
  - Generic interfaces
  - Open protocols
  - Geographic dispersion
- Deployment / Maintenance
- Security features
**Conclusion**

- **Coda:**
  - Hard to configure
  - Can not handle large files
  - Promising architecture
  - Low latency impact

- **Lustre:**
  - DRBD is needed for replication / latency reduction
  - Latency has impact
  - No multi-tier
  - Promising future
Future work

- Review of GlusterFS
- Test environment without VMWare ESXi
- Further tuning and configuring of Coda
- Coda and ZFS tests
- Lustre and DRBD tests
Appendix - Heartbeat
Appendix - Lustre

Diagram:
- **Clients** (LOV)
  - File open
    - Directory Operations, file open/close, metadata, and concurrency
  - File I/O and file locking
- **MDS**
  - Recovery, file status and file creation
- **OSS**
Appendix - Lustre

![Lustre diagram]

- Metadata Server (MDS)
- Metadata Target (MDT)
- High Speed Interconnect (Ethernet, IB, etc.)
- Lustre Clients
- Object Storage Servers (OSSs)
- Object Storage Targets (OSTs)
- OSS 1
- OSS 2
Appendix – ZFS L2ARC
Appendix - Coda
Appendix - Coda
## Appendix – Seeks (seeks/sec)

### Coda

<table>
<thead>
<tr>
<th>Latency Type</th>
<th>min</th>
<th>avg</th>
<th>max</th>
</tr>
</thead>
<tbody>
<tr>
<td>No added latency</td>
<td>27,6</td>
<td>28</td>
<td>28,4</td>
</tr>
<tr>
<td>Latency (20ms RTT)</td>
<td>9,5</td>
<td>9,8</td>
<td>10</td>
</tr>
</tbody>
</table>

### Lustre

<table>
<thead>
<tr>
<th>Latency Type</th>
<th>min</th>
<th>avg</th>
<th>max</th>
</tr>
</thead>
<tbody>
<tr>
<td>No added latency</td>
<td>151</td>
<td>153</td>
<td>155</td>
</tr>
<tr>
<td>Latency (20ms RTT)</td>
<td>28,7</td>
<td>29</td>
<td>29,2</td>
</tr>
</tbody>
</table>