Berkeley Storage Manager (BeStMan)

Alex Sim

Scientific Data Management Research Group
Computational Research Division
Lawrence Berkeley National Laboratory
**Light-weight implementation of SRM v2.2**

- Works on existing disk storages with posix-like file systems
  - E.g. NFS, GPFS, GFS, NGFS, PNFS, HFS+, PVFS, Lustre, Xrootd, Hadoop, Ibrix
- Supports multiple partitions
- Adaptable to other file systems and storages
  - Supports customized plug-in for MSS to stage/archive such as HPSS
  - Easy adaptability and integration to special project environments

**Supports multiple transfer protocols**

- Supports load balancing for multiple transfer servers

**Scales well with some file systems and storages**

- Xrootd, Hadoop

**Works with grid-mapfile or GUMS server**

**Simple installation and easy maintenance**

**Packaged in VDT using Pacman**
Who is BeStMan for?

- Sites that need SRM interface to their local storage resources
- Sites with POSIX-like file systems
- Sites that need great performance
- Sites with limited resources
- Sites where minimal administrative effort is needed
- Sites that need great support
Difference between BeStMan Full mode and BeStMan Gateway mode

- Full implementation of SRM v2.2
- Support for dynamic space reservation
- Support for request queue management and space management
- Plug-in support for mass storage systems

- Support for essential subset of SRM v2.2
- Support for pre-defined static space tokens
- Faster performance without queue and space management
Interoperability with other SRMs
A few hints on what BeStMan can do

- **Dynamic installation, configuration and running**
  - If the target host does not have an SRM, BeStMan can be downloaded, installed, configured, and started with a few commands by the user.

- **BeStMan can restrict all user access to certain directory paths through configuration**

- **BeStMan can be configured to restrict user access to files by owners/creators only**

- **A site can customize the load-balancing mechanism for transfer servers through plug-in**

- **A site can customize the file system i/o mechanism for special file system or storage system through plug-in**

- **Works with other SRM v2.2 implementations**
Some Use Cases

- **CMS**
  - BeStMan Gateway as an SRM frontend for Hadoop at UNL
  - Passed all the automated CMS tests through EGEE SAM at UN

- **ATLAS**
  - BeStMan on NFS
  - BeStMan Gateway on Xrootd/FS, GPFS, Ibrix

- **STAR**
  - Data replication between BNL and LBNL
    - HPSS access at BNL and NERSC
    - SRMs in production for over 4 years
  - Part of analysis scenario to move job-generated data files from PDSF/NERSC to remote BNL storage

- **Earth System Grid**
  - Serving about 6000 users
    - Over a million files and 170TB of climate data
    - from 5 storage sites (LANL, LLNL, NCAR, NERSC, ORNL)
  - Uses an adapted BeStMan for NCAR’s own MSS
Performance test (1)

- **This stress tests done by Wei Yang at SLAC**
  - Tests on PUT and GET operations with Xrootd in the backend
- **Setup**
  - BeStMan-Gateway and XrootdFS
    - On dual dual-core AMD Opteron 275, 2.2Ghz, 4GB memory, 1Gbit NIC
  - GridFTP servers
    - osgserv01,04 on dual AMD Opteron 244 1.8Ghz (total two cores), 2GB memory, 1 Gbit NIC
    - griddev05,6,7 on dual Intel Pentium III 866Mhz, 1GB memory, 100Mbit NIC
Performance test (1)

- **Client setup**
  - Client on lxplus202.cern.ch (one node)
    - glite-url-copy was used to simulate FTS
    - On average, lxplus can run 100 glite-url-copy for PUT in 16.4 seconds and 100 GET in 14.3 seconds
    - A script contains glite-url-copy from lxplus.cern.ch to transfer 79 bytes
    - 5000 sequential jobs were run in non-block mode.
  - **PUT operation goes through several SRM interfaces**
    - srmLs for the target directory which does not exist
    - srmLs for the parent directory of the target directory which exists
    - srmMkdir for the target directory
    - srmPrepareToPut
    - Gsiftp file transfer
    - srmPutDone
  - **GET operation goes through several SRM interfaces**
    - srmPrepareToGet
    - Gsiftp file transfer
    - srmReleaseFiles
Performance test results (1)

- **Test result for PUT operation**
  - 5000 PUT, all completed
  - Total 885 seconds
  - ~5.65 PUT operations per second

- **Test result for GET operation**
  - 5000 GET, all completed
  - Total 709 seconds
  - ~7.05 GET operations per second

- **Notes**
  - Xrootd storage imposes a 5 seconds delay when creating new files
  - There was only one client node in this test
Performance test (2)

- **This stress tests done by Brian Bockelman at UNL**
  - Tests on LS operation with Hadoop in the backend
- **UNL**
  - UNL is the first instance of a BeStMan gateway endpoint to pass all the automated CMS tests which is done through EGEE SAM product.
  - For US CMS and US ATLAS, it manages data transfers of 1-2 TB an hour
    - peaks up to 10Gbps, sustains 2 Gbps
- **Setup**
  - BeStMan-Gateway
    - On 2x dual core Xeons @ 2.66GHz, 2GB memory, single 160GB, 1Gbit NIC
    - Running CentOS 5.2 x86_64 with absolutely no tweaks
  - Hadoop 0.18.1 with custom site patches for approx. 110TB raw disk
    - Mounts as a normal file system through FUSE
  - GridFTP servers
    - Globus version with a custom Hadoop DSI module
    - 10 GridFTP servers with BeStMan load-balancing mechanism
Performance test (2)

- **Client setup**
  - Clients on 200 hosts
    - Each host started one script almost at the same time (+/- 2 seconds)
    - Each script did 5 sequential srmLs operations
## Performance test results (2)

<table>
<thead>
<tr>
<th>GSI proxy delegation</th>
<th>GUMS authentication</th>
</tr>
</thead>
<tbody>
<tr>
<td>ON</td>
<td>ON</td>
</tr>
<tr>
<td></td>
<td>1-3 Hz, 100% CPU utilization</td>
</tr>
<tr>
<td>OFF</td>
<td>OFF</td>
</tr>
<tr>
<td></td>
<td>10 Hz, 80 % CPU utilization</td>
</tr>
</tbody>
</table>

- Entire test was completed in 30 seconds
Memory Usage Test

- This memory test done by Junmin Gu at LBNL
  - Tests with SRM operation with NFS disks in the backend
- Setup
  - BeStMan-Gateway with a GridFTP server on a single node
- Test
  - 12 srmGetSpaceTokens
  - 54 GET operations
  - 665 srmRm
  - 2714 srmPing
  - 5343 srmLs
  - 4609 PUT operations
  - 81 srmMkdir
- Results
  - Peak memory usage was ~60MB.
Summary

• **BeStMan** is an implementation of SRM v2.2.
  - Great for disk-based storage and file systems

• **BeStMan Gateway mode on some file systems and storage gives scalable performance**

• **Install/maintain through VDT**

• **Works with other SRM v2.2 implementations**
  - Servers: CASTOR, dCache, DPM, StoRM, SRM/SRB, …
  - Clients: PhEDEx, FTS, glite-url-copy, lcg-cp, srm-copy, srmcp, …
  - In OSG, WLCG/EGEE, ESG, …
OSG Storage documentation
  - https://twiki.grid.iu.edu/twiki/bin/view/Documentation/WebHome

BeStMan
  - http://datagrid.lbl.gov/bestman
  - http://hep-t3.physics.umd.edu/HowToForAdmins.html#osgBestman
  - https://www.usatlas.bnl.gov/twiki/bin/view/Admins/BestMan
  - https://twiki.grid.iu.edu/bin/view/Documentation/BestmanGateway
  - https://twiki.grid.iu.edu/bin/view/Documentation/BestmanGateway-Xrootd

SRM Collaboration and SRM Specifications
  - http://sdm.lbl.gov/srm-wg

Contact and support : srm@lbl.gov