New Features of Xrootd SE

Wei Yang

US ATLAS Tier 2/Tier 3 meeting, University of Texas, Arlington, 2009-11-12

New features of Xrootd:

◆ Support space tokens up to 63 characters

- Previously 15 characters
- In production at SLAC

◆ Better support of composite name space (CNS)

- Easy recovery of missing entries on CNS
- Space usage auditing
- File inventory records:
  location (host) and other attributes of real data files
- Inventory feature conflicts with the current Xrootd SE.
  Several choice to address this conflict.

◆ Monitoring

- Monitoring data is XML based
- Sent to and collected by a 3rd host to feed to Ganglia, Nagios, etc.

New features of Xrootd, cont'd

- **Client lib no longer repeat identical large reads**

  Existing clients in ATLAS releases can overload Xrootd data servers in this way when large read ahead is turn on

- **Multiple read ahead/streaming algorithms**

  When will it be integrated into ROOT and ATLAS releases?

New features of XrootdFS

◆ Improved performance under small I/O blocks

- Read ahead managed by Linux kernel (configuration changes)
- Implement a write cache to capture small sequential writes

◆ Directory browsing & usage query w/o CNS

- CNS has been stable at SLAC
- XrootdFS without CNS can't tell if a data server is down
- Tier 3 users seems to be confused by CNS. Easy to make mistakes

- New XrootdFS in light usage at SLAC, should we push it to OSG?
- No CNS, no conflict with the inventory feature

New features of Bestman-gateways

◆ Support Adler32

- Built-in Adler32 calculation
- Allow site specific external tool for calculate Adler32
- Turn directory browsing on and off

◆ Support non-GridFTP protocols

Interested by ATLAS to access conditions files via site specific protocols

Xrootd SE Components

- **Bestman Gateway** ← T2/T3g
- **XrootdFS** ← For users and minimum T3g
  - Usage is like NFS
  - Based on Xrootd Posix library and FUSE
  - BeStMan, dq2 clients, and Unix tools need it
- **GridFTP for Xrootd** ← WT2 for a while
  - Globus GridFTP + Data Storage Interface (DSI) module for Xrootd/Posix
- **Xrootd Core** ← All Babar needed is this layer

  Redirector, data servers, xrdcp

User interface to Xrootd

◆ TXNetFile class (C++ and ROOT CINT)

Fault tolerance
High performance thought intelligent logics in TXNetFile and server

◆ Command line tools

- **xrdcp**

simple, native, light weight, high performance

- **Xrootd Posix preload library**

```
export LD_PRELOAD=/…/libXrdPosixPreload.so
ls/cat/cp/file root://redirector:port//path/file
```

◆ XrootdFS

Mount the Xrootd cluster on client host's local file system tree

Accessing Xrootd data from ATLAS jobs

◆ Copy input data from Xrootd to local disk on WN

A wrapper script using xrdcp, or cp + xrootd posix preload library
Panda production jobs at SLACXRD work this way.

◆ Read ROOT files directly from Xrootd storage

Identify ROOT file using Unix 'file' command (w/ posix preload library)
Copy non-ROOT files to local disk on WN
Put ROOT file's xroot URL (root://…) in PoolFileCatalog.xml
Athena uses TXNetFile class to read ROOT file
ANALY_SLAC and ANALY_SWT2_CPB use this mixed accessing mode.

Both need a set of tools for copying, deleting, file id and checksum

◆ Mount XrootdFS on all batch nodes

All files appear under local file system tree.
None of the above is needed
Untested: XrootdFS came out after SLAC sites were established.

Potential usage of Xrootd SE technology

High performance xrdcp over WAN and BitTorrent style data transfer allows

- **Super Xrootd Clusters (federation) across geographic sites**
- **Treat Xrootd SE at other sites as virtual MSS**

  Using Xrootd MSS interface to fetch data from non-Xrootd SEs ?

See Andy Hanushevsky's talk at ANL Tier 3 meeting

Virtual Mass Storage System