GridFTP Ceph plugin

Ian Johnson
Data Services Group
Scientific Computing Department
STFC RAL

Low GridFTP Ceph Write Rates

- The problem – FTS writes from CASTOR into Ceph get a low transfer rate – only ~ 6-7 MB/s
  - NB Reading from Ceph into CASTOR ~ 147 MB/s
- The reason
  - FTS uses ‘MODE E’ (Extended Block Mode) - transfer buffer size is 256 KiB.
  - Hence, Ceph writes are 256 KiB as well. The client controls the buffer size, and a low write rate results.

Transfer Performance in Mode E vs. Stream Mode

- Comparison with GridFTP STREAM mode, e.g. globus-url-copy with no use of '-p' option:
  - For increasing read/write performance in STREAM mode, I implemented Sebastien's suggestion to use large Ceph API buffers
  - The server controls buffer size and can obtain ~ 120 MB/s write performance using a buffer of 256 MiB

Proposed Solution

- As the FTS client in Mode E sets the transfer buffer block size, the server ignores the setting of the Ceph write buffer.
- Hence, need to find another way to send a large buffer to Ceph to get a good write rate

Buffer Assembly

- Allocate a single, large ‘Ceph buffer’ (around 256 MiB, tunable in gridftp.conf)
- Store blocks in the Ceph buffer according to their offset
- When the buffer is full, write it to Ceph
- Any blocks with offset past end of Ceph buffer can be stored in an overflow array, and copied into the Ceph buffer at start of next iteration

Authorization

- RAL looking at AuthDB file approach for XRootD
  - Hope to use similar for GridFTP
  - Will try Brian Bockelman's suggestion to use XrootD libxrdserver.so to create Authorization object and XrdSecEntitity from GridFTP code