Scalable Asynchronous Access

NEPS Workshop, CITI
4 December 2003
Craig Everhart

Agenda

- Asynchronous access
- Scaling via reducing operation traffic
- Windows features
- Read/write block maps

Asynchronous/Third-Party Data Access

- Server can *perform* or *coordinate* data access
- Perform: traditional file server, data access as proxy
- Coordinate: server gives client layout metadata, client does access directly
- Possible access methods include (striped) NFS servers, block data (SBC), object data (OSD), etc. Metadata varies with method. Client could handle multiple types
- Coordination by server provides guarantee of layout metadata validity
- NFS version 4 *delegation* serves as a model for the time validity of the guarantee

Data Security

- Block data (SBC) potentially accessible on SAN
- Useful inside a security perimeter
- Removes zoning/masking access control
  - though could zone/mask on demand under server control
- For some configurations, makes sense
- Wish to allow interoperability among implementations
- Other access metadata have less access exposure (OSD, subsidiary servers)

Reducing client-server traffic

- Open/share state: passed through to server (except under delegation). Could cache if:
  - server asked to recall state on conflict
  - client could answer Yes or No
- Similarly for byte-range locking
  - more complex recall due to byte ranges
- Need “relinquish soon” flag on recall request
  - useful if there is a waiter for a lock

More on traffic reductions

- Await a lock; perhaps even await an open
  - Server could inform client of lock availability
- Ideally under control of the server so it manages its own resources
- Perhaps opens, locks could be cached under an additional, more easily obtainable, more nuanced variety of delegation?
- Ideally could clearly regain a delegation after a recall happens, too.

Greater Windows/NTFS compatibility

- DELETE share/deny mode
  - in addition to READ and WRITE
- Directory change notification
- Reparse point data type (type plus blob of data, up to 16KB today); interpretation of reparse point content not required today

Example for SBC/block maps:
R, W extent maps for COW

- (R, W): e=empty, I=invalid, V=valid
- (e, e) initially [a single extent or block]
- (e, I) after allocate; write to “I” blocks
- (e, V) after writing and server update
- (V, e) after clone (made read-only)
- (V, I) after preparing for COW; copy/write to “I” blks
- (e, V) after writing and server update
- Other solutions possible also

Positioning

- Agnostic to back-end store
- Complementary to RDMA solutions
- Scalability, Windows friendliness applicable regardless of back-end store
- Fallback to direct data operation may or may not be available