

Java NFS

Brent Callaghan

Sun Microsystems, Inc.

brent@eng.sun.com

NFS Client

- Supports Xfile API - like Java I/O API
 - See www.sun.com/webnfs
- Tries v4 first
- Then WebNFS over TCP for v3
- Falls back to UDP if no TCP
- Falls back to v2 if no v3
- Falls back to MOUNT if no WebNFS

NFS Client Features

- read, write, create, mkdir, rmdir, remove, getattr, readdir, readdirplus, rename
- Caches file data, directories, access, attributes, symlinks
- Follows symlinks
- Read-ahead and write-behind for good performance
 - Get 1 MB/sec on 10Mb Ethernet
 - Get 4 MB/sec on a 100Mb

NFS Server

- RPC layer accepts multiple protocols
- Handles TCP or UDP connections
- Multithreaded
 - One listener per endpoint
 - One dispatcher per connection
 - One thread per request
- Very robust!

NFS Server Features

- Small, client & server (source & classes) fit on a floppy
- Supports v3 & v4 but not v2
- Limited access to attributes
 - File type, size, mod time, readonly
 - fileid is a hash of the pathname
- Filehandles are volatile - 4 byte hash
- Filehandle changes on rename

NFS Server Features (cont)

- Same server bytecode ran on Windows & Solaris
- Bakeoff tested most functionality:
 - lookup, getattr, read, write, open, close, remove, rename, commit
- “Super Lookup” gets file data in single turnaround
 - PUTFH, LOOKUP, GETFH, GETATTR, ACCESS, READ

Connectathon Testing

- Working on SETCLIENTID/OPEN/CLOSE for delegation.
- Will try for file locking
- Workarounds for limitations of Java client API, e.g. Java IO has no file locking.

Questions ?