SunSoft
A Sun Microsystems Company

NFS Client Failover: “NFS server *is* responding”

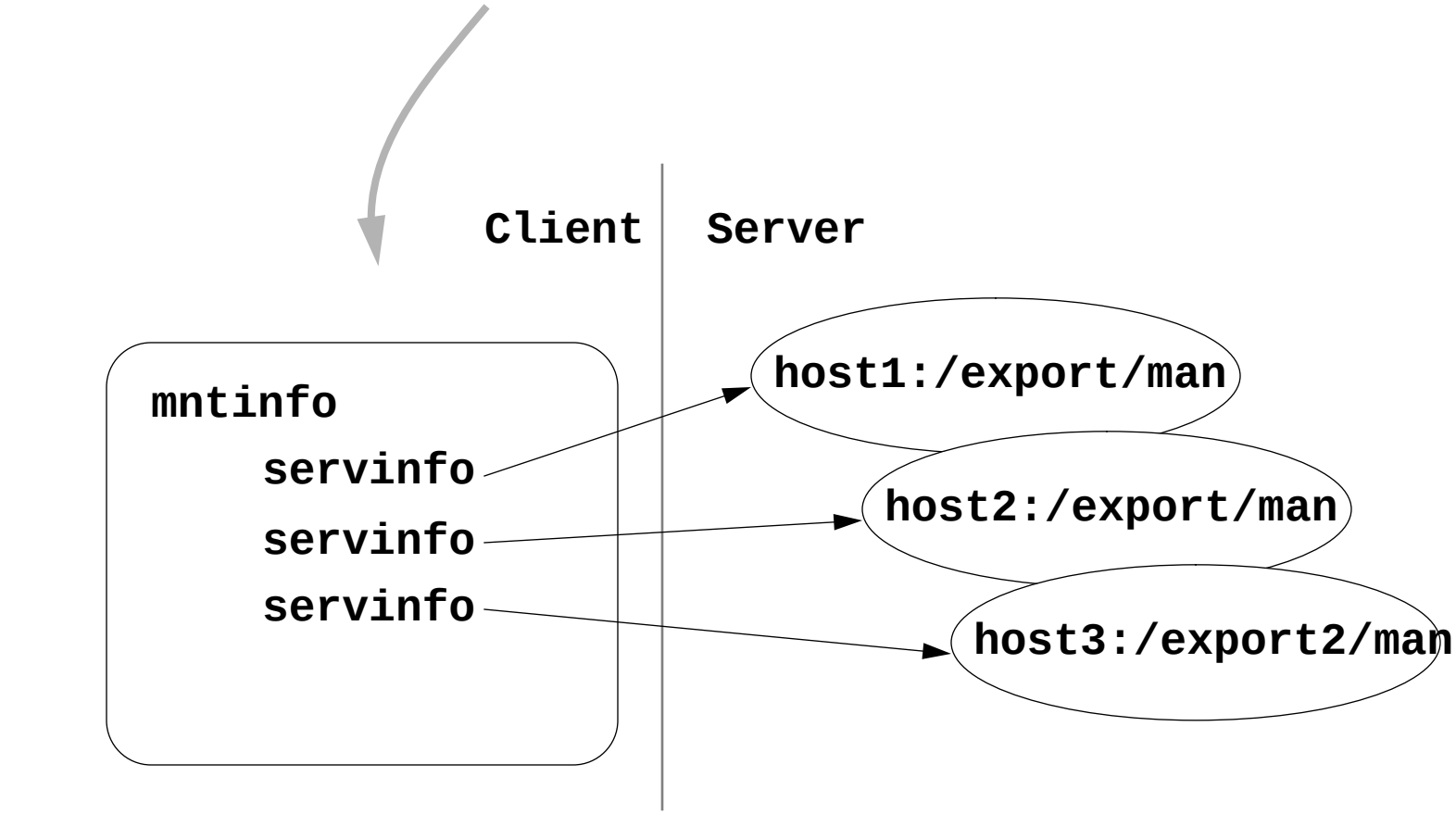
Rob Thurlow

thurlow@eng.sun.com

What is client failover?

- Want better availability of shared, replicated filesystems
  - Why hang when you know of other copies?
  - Heavily used data most likely to cause hang
- Extends automounter “multiple choice” support to kernel
  - host1,host2:/export/man,host3:/export2/man
  - automountd now sorts rather than selecting best
  - mount_nfs(1M) now also supports this syntax
  - NFS mounts pass info about N servers into kernel
  - NFS tries other servers rather than printing “NFS server not responding”
  - Failover done per-vnode, with a per-filesystem hint

Failover mounting



Client failover operation

```
peyto[88]% df /usr/dist
Filesystem             kbytes    used   avail capacity  Mounted on
udmpk17c-86,udmpk17d-86:/export/dist
                       7556525 5597760 1203115    83%    /usr/dist

peyto[89]% nfsstat -m
/usr/dist from udmpk17c-86,udmpk17d-86:/export/dist
 Flags:   vers=2,proto=udp,sec=unix,hard,intr,dynamic,
llock,rsize=8192,wsize=8192,retrans=5
 Lookups: srtt=8 (20ms), dev=5 (25ms), cur=3 (60ms)
 Reads:   srtt=25 (62ms), dev=4 (20ms), cur=5 (100ms)
 Failover:noresponse=0, failover=0, remap=0, currserver=udmpk17c-86
```

Client failover changes

- No server changes - client does it all
- Simplify, simplify, simplify
  - Read-only support, no read-write
  - Locks tracked only on client
  - Hard mounts only
  - No replication method, rdist/tar/cpio for now
- Complications cause fallback to established behaviour
  - Lack of read-only flag
  - Presence of soft flag
  - Can still see “server not responding” if switching can break things, e.g. readdir, replica differences

Failover implementation

- All server-specific fields are pulled out of “struct mntinfo”
  - root filehandle
  - hostname and netname
  - network type and address
  - authentication flavor
  - AUTH_DES/AUTH_KERB/etc. time sync address
- Stored partial pathnames from VFS root allow remapping
  - NFS lookups fill in path and remap it
  - rfscall() takes RFSCALL_SOFT option to avoid hang
- rfscall() also accepts opaque argument used in remapping

Status

- Code running on SunSoft NFS machines here
- Testing in progress now, looks good so far
- Getting close to complete, so start thinking about your shared filesystems now!