Samba as a gateway to OpenAFS

Fabrizio Manfredi Furuholmen

Agenda

- Goal
- Solution
- Gateway Architecture
- Gateway Configuration
- Integration Tools
- Tuning
- Performance
- Result

ZEROPIU
Difference as value

Project Goal

Primary goal of the project was to design and build an inexpensive storage system

Requirements:

- Handle terabytes of data
- Transparent to final user
- Working in WAN environment
- Good level of scalability

Solution

ZEROPIU
Difference as value

Considerations

- Centralize Storage (hardware solution)
  - SAN
    - Blockdevice interface
    - Performance
  - NAS
    - Filesystem interface
    - Shared filesystem
- Distributed Filesystem (software solution)
  - Filesystem interface
  - Single file system across multiple computer nodes

Solution

Considerations

- Big Server vs Small Server (Google Techs)
- Small number of inexpensive fileservers provides similar performance to client side
- Increase in capacity are inexpensive
- Better manageability and redundancy.

Solution

Storage Price

- Terabyte Cost (SAS/FB)
  - 14k euro NAS/SAN
  - 4k euro DFS
- Disks Size
  - 143 vs 300 SAS/FB reduce 30%
- Disks Type
  - 250/500 SATA Disk reduce >50%
- Installation
- Software
- Discount
- Administration

Components	NAS	SAN	DFS
Storage1.5 Tb with 10 disks (110/150)	52.000	52.000	
Storage 14TB 100 disks (110/150)	200.000	200.000	
3 Server Storage 500Gb (SAS)			9.000
14 Server Storage 1Tb (SAS)			56.000
4 FB interface		1.600	
2 Switch FB		6.000	
2 Server Gw		2.000	2.000
2 Switch Gb	1.200	1.200	1.200
TOTAL for 1.5 Tb	**53.200**	**62.800**	**12.200**
TOTAL for 14 Tb	**201.200**	**210.800**	**59.200**

ZEROPIU
Difference as value

Solution

- Distributed Filesystem
  - AFS
    - Free available and stable
    - Support of large installations (>200TB with 40 milion Files)
    - More then 20 platforms are supported
    - Aggressive Roadmap ($350,000 per year from CSG)
  - Samba (Gateway)
    - AFS windows client uses internal file server emulation (slow)
    - Clientless
    - Fast and stable
- User Identity
  - Heimdal Kerberos Autentichation (SSO)
    - KA emulation
    - LDAP backend
    - 2b protocol (large kerberos ticket)
  - Openldap
    - Centralize storage
  - User administration scripts (custom provisioning)

ZEROPIU
Difference as value

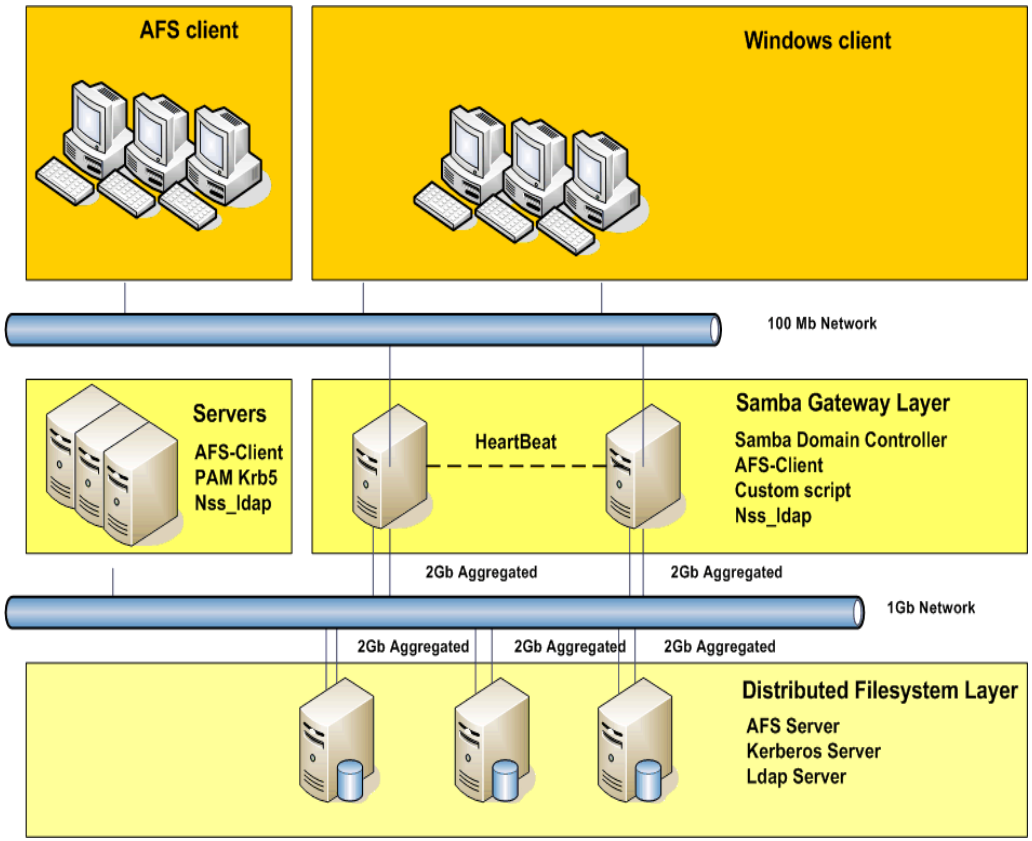
AFS Features

- Transparent Access and Uniform Namespace
  - Cell
  - Partitions and volumes
  - Mount Points
  - In-use volume moves
- Scalability
  - Client Caching
  - Replication
  - Load balance among servers while data is in use
- Security
  - Authentication and secure communication
  - Authorization and flexible access control
- System Management
  - Single system interface
  - Administration tasks without system outage
  - Delegation
  - Backup

Gateway Architeture

Architeture

- Scalability
  - Storage scalability (Filesystem layer)
  - User scalability (Samba Gateway layer)
- Performance
  - Load balancing
  - Roaming user/branch office
- Clientless
- Centralized Identity
  - Kerberos
  - Ldap



Gateway Configuration

ZEROPIU
Difference as value

Enable AFS in Samba

- Compile Options
  - Enable KA server emulation
    --with-fake-kaserver
  - Enable AFS ACL mapping
    --with-vfs-afsacl
  - Don’t use AFS clear text password switch (old not supported)
    --with-afs
- Setting Samba Trusting (undocumented)
  - Obtain KeyFile from AFS fileserver (/usr/afs/etc/)
  - Import an OpenAFS KeyFile into secrets.tdb:
    net afs key AFSKeyFile
  - Custom script for AFSKeyFile sync (Key rotation)
- Useful command (undocumented)
  - Impersonate user, create a token for user@cell:
    net afs impersonate <user> <cell>

Gateway Configuration

smb.conf

- Mapping Domain User<-> Pts
  - Single domain/unique identification:
    - afs username map = %u@zero.it
  - Multiple domain/duplicated identification
    - Store DOMAIN+user:
      afs username map = %D+%u@zero.it
    - Store the SID in pt server:
      afs username map = %s@zero.it
- Enable AFS share

  afs share = yes

Gateway Configuration

ZEROPIU
Difference as value

smb.conf locking

- Access only from samba server
  - Samba default
- Access only from samba and local gw
  - Disable oplocks , level2 oplocks ..
  - Only with Byte-range locking on AFS client (AFS>1.5.X)
- Access from all system
  - Enable strict locking option (mandatory lock)

Gateway Configuration

Samba scalability and HA

- Primary server HA (DFS Root)
  - Heartbeat
  - VIP associated to primary Samba Server
- Transparent Access (MSDFS)
  - No compile option required
  - Enable DFS on Primary Samba server
    host msdfs = yes
- Samba Scalability
  - DFS Proxy,
    - Share redirection
    - Name resolved with DNS (link is FQDN)
    (ex. msdfs proxy = \gw1.intranet.zeropiu.it\share)
  - DFS root ,
    - Directory link
    - Fault tolerance
    - (ex. ln –s msdfs:\\server1\share1,server2\share1 share1)

Gateway Configuration

ZEROPIU
Difference as value

Identity Storage

- Heimdal integration
  - Compile
    - Enable ldap backend (--with-openldap)
  - Configuration
    - Enable ldap backend
    - Enable 2b token for Kerberos V integration
    - Only if have old client: enable-kaserver / afs3-salt
- LDAP
  - Openldap 2.3 (SASL EXTERNAL)
  - Extending Schema (Samba,hdb ..)
  - nss_switch with ldap support
- PAM
  - PAM Kerberos V integration

Integration Tools

Identity Administration

- Custom user administration script (iauser.pl)
  - Unix user (ldap)
  - Samba user (ldap)
  - Kerberos user (ldap)
  - Pt server user
  - Volume and mount point
- Groups administration script (iagroup.pl)
  - Create unix group (ldap)
  - Create samba group (ldap)
  - Create pt server group
- Syncronization administration script (ptsSync.pl)
  - Synchronization user from ldap to pt server

Performance

Test Enviroment

- Hardware
  - 3 FileServer Linux
    - 2 GB of RAM, 3GHz Xeon processor
    - 2x36Gb SAS RAID 1 for operating system partition
    - 4x 143GB SAS RAID5 storage
  - 2 Server Gateway Linux
    - 2 GB of RAM, 3GHz Xeon dual processor
    - 2x36Gb SCSI RAID 1 for operating system partition
- Software
  - Samba 3.22
  - OpenAFS 1.4.2
  - IOzone 3.8

Performance

Samba Client

Client:

Windows XP sp2

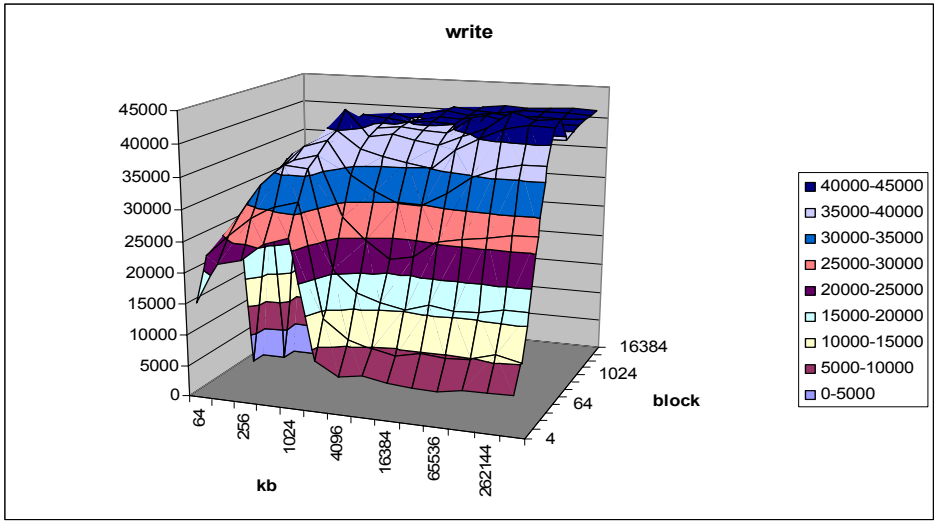
Server:

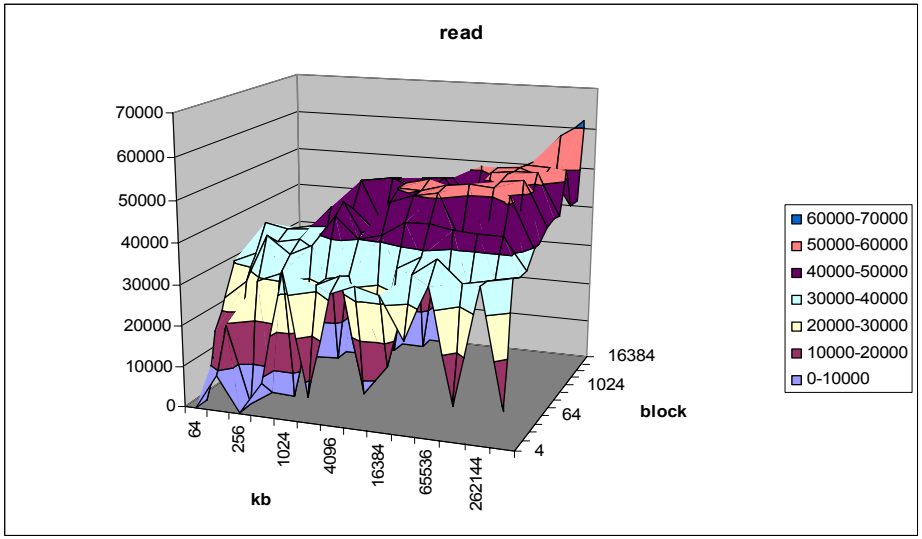
Linux 2.6.9

Samba 3.22

Write: 30-43MB/sec

Read: 40-50MB/sec





Performance

AFS Client

Client:

Linux 2.6.9

openafs 1.4.2

Server:

Linux 2.6.9

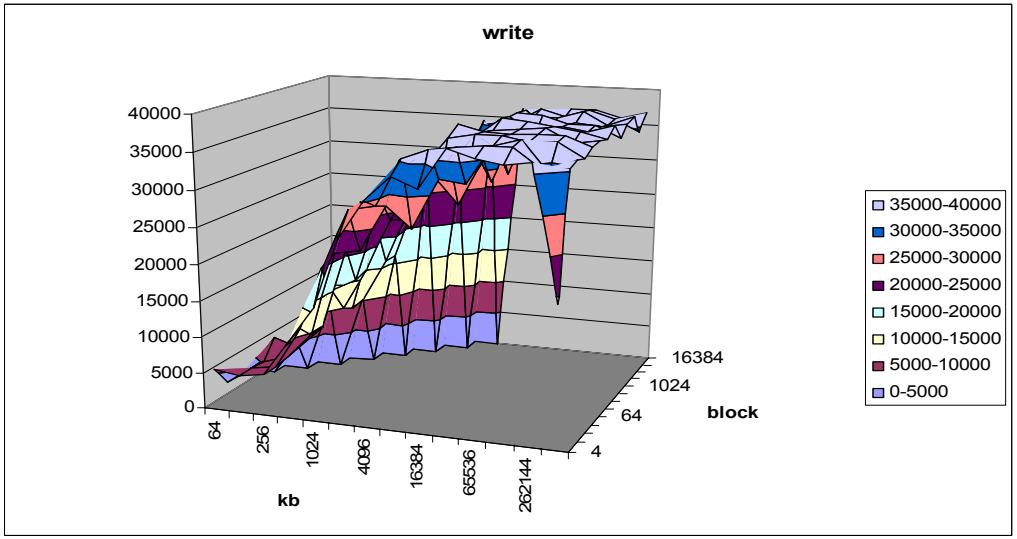
openafs 1.4.2

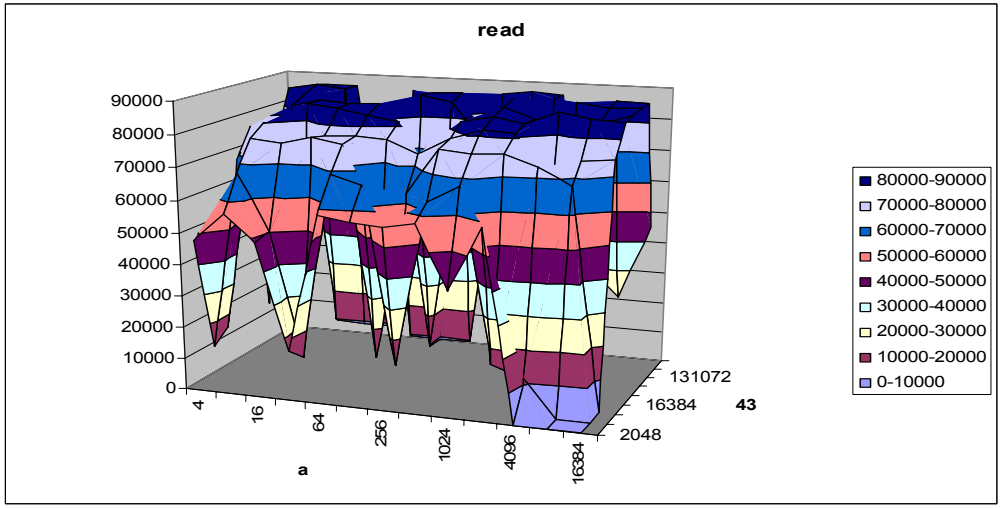
Write: 20-37MB/sec

Read

Cold Cache: 22-28MB/sec

Warm Cache: >45MB/sec





Performance

Samba GW

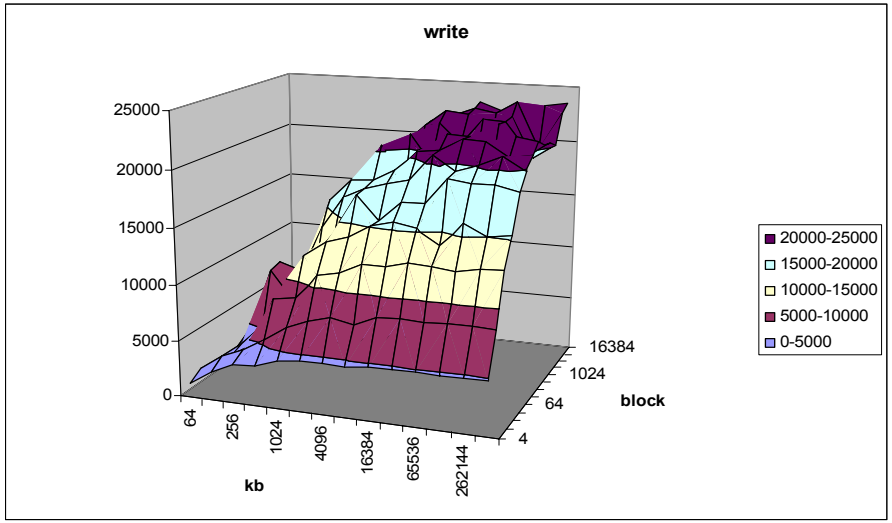
Server:
- Linux 2.6.9
- openafs 1.4.2

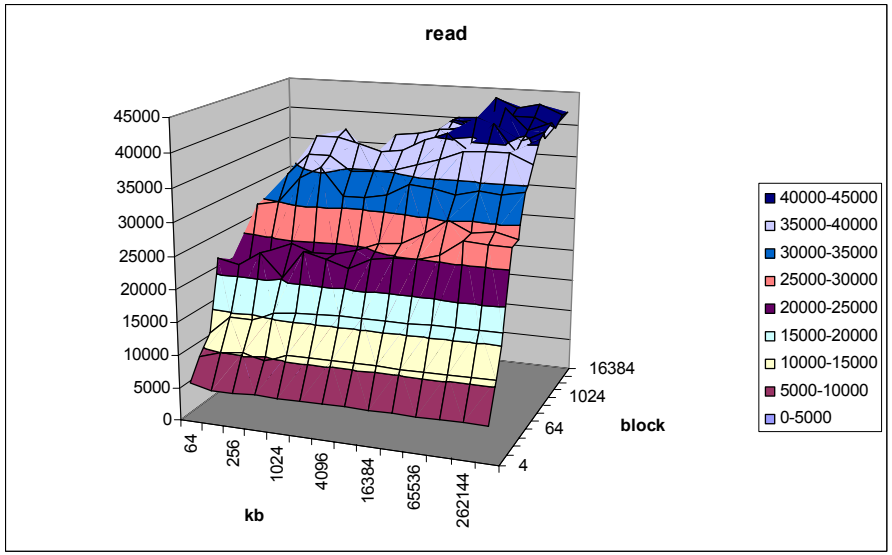
Gateway:
- Linux 2.6.9
- OpenAFS 1.4.2
- Samba 3.22

Client:
- Windows XP sp2

Write: 18-25MB/sec

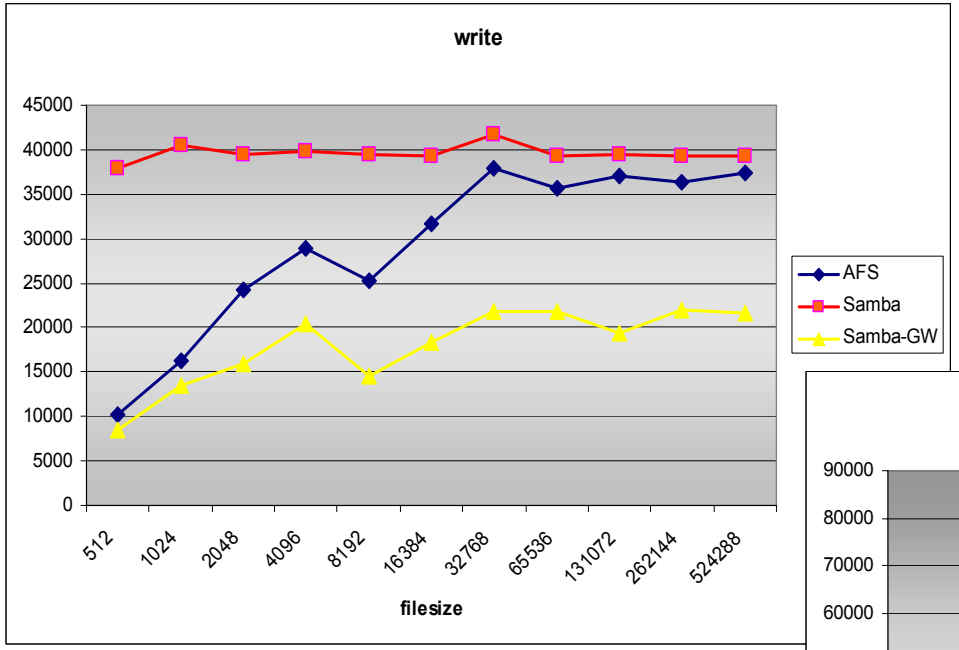
Read
Warm Cache: 30-40MB/sec

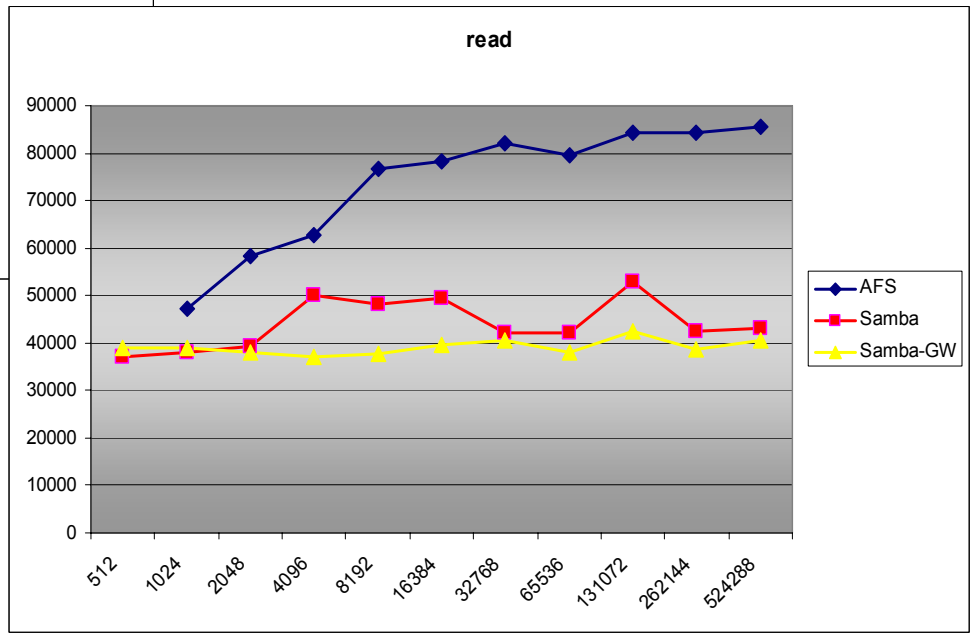




Performance

Throughoutput Coparison





Performance

ZEROPIU
Difference as value

Tuning

- Samba Configuration (increase 30%)
  - Enable socket options = TCP_NODELAY (Default)
  - Increase SO_RCVBUF (16384)
  - Increase SO_SNDBUF (32768)

- AFS Cache Manager (increase 20%)
  - Increase block size (chunksize 19)
  - Increase cache elements (dcache 5000)
  - Increase server daemon (daemons 6)
  - Increase rx packet (rxpck 2000)
  - Increase data cache file (files 50000)
  - Increase Cache size (cache size 4gb)
  - Use separated disk for cache

Result

Benefit

- Reduced cost
  - Reduced storage cost 40.000 Euro (1.5TB Storage)
  - Reduced down time
- Increase performance
  - Client side
- Simplify System Administration task
  - Data accessible from everywhere
  - High security level (kerberos base)
  - Single sign-on
  - Disaster recovery (Volume replication)

Next

Under Testing

- OpenAFS
  - Lock subsystem, support AFS 1.5.X (Byte range)
  - Windows client, support AFS 1.5.X
  - Inode interface
  - Socket communication vlserver/fileserver
  - Memory cache
  - Disable fsync on write (AFS 1.5.X + patch)
  - WebDav
- Samba
  - Cluster
- External project (www.beolink.org)
  - Ptserver with ldap backend (ptsldap)
  - Web Administration interface (AFS Manager)

ZEROPIU
Difference as value

The End

Manfred at zeropiu.it
www.beolink.org