HPTFS: High Performance Tape File System

Xianbo Zhang, David Du
University of Minnesota (DISC)
Jim Hughes, Ravi Kavuri
Sun StorageTek
5/18/2006

Tape Background

- Huge capacity: tape capacity is doubling every two years or 18 months
  - In 2006, one DLT-S4 tape cartridge reaches the capacity of 800 GB native data capacity
- Relatively high streaming rate: tape drive speed is increasing
  - In 2005, Sun StorageTek T10000 drive provides 120MB/s native data transfer rate
- Tape storage has the advantage of low cost per GB, off-site portability and less power consumption compared to other storage solutions

Technologies for Fast Data Location

- Tape cartridge embedded memory chip
  - AIT and LTO tape cartridge embedded memory chip may help obtain data location information without involving tape movement
- Dual mode tape wrap for fast search
  - W/o tape wrapping around drum, tape drive performs fast forward and rewind
  - W/ tape wrapping around drum, tape drive performs write, read and low-speed search

Tape Capacity/Speed Migration Path

Tape technology	2000	2001	2002	2003	2004	2005	2006
VXA		33GB, 3MB/s	80GB, 6MB/s			160GB, 12MB/s	
DLT Value line		40GB, 3MB/s		80GB, 8MB/s		160GB, 10MB/s	
AIT		100GB, 12MB/s				200GB, 24MB/s	
SAIT				500GB, 30MB/s			800GB, 45MB/s
Super DLT			160GB, 16MB/s		300GB, 36MB/s		800GB, 60MB/s
LTO	100GB, 15MB/s		200GB, 35MB/s		400GB, 80MB/s		
STK9940	60GB, 10MB/s		200GB, 30MB/s			500GB, 120MB/s	
IBMTS1120				300GB, 50MB/s		500GB, 100MB/s	

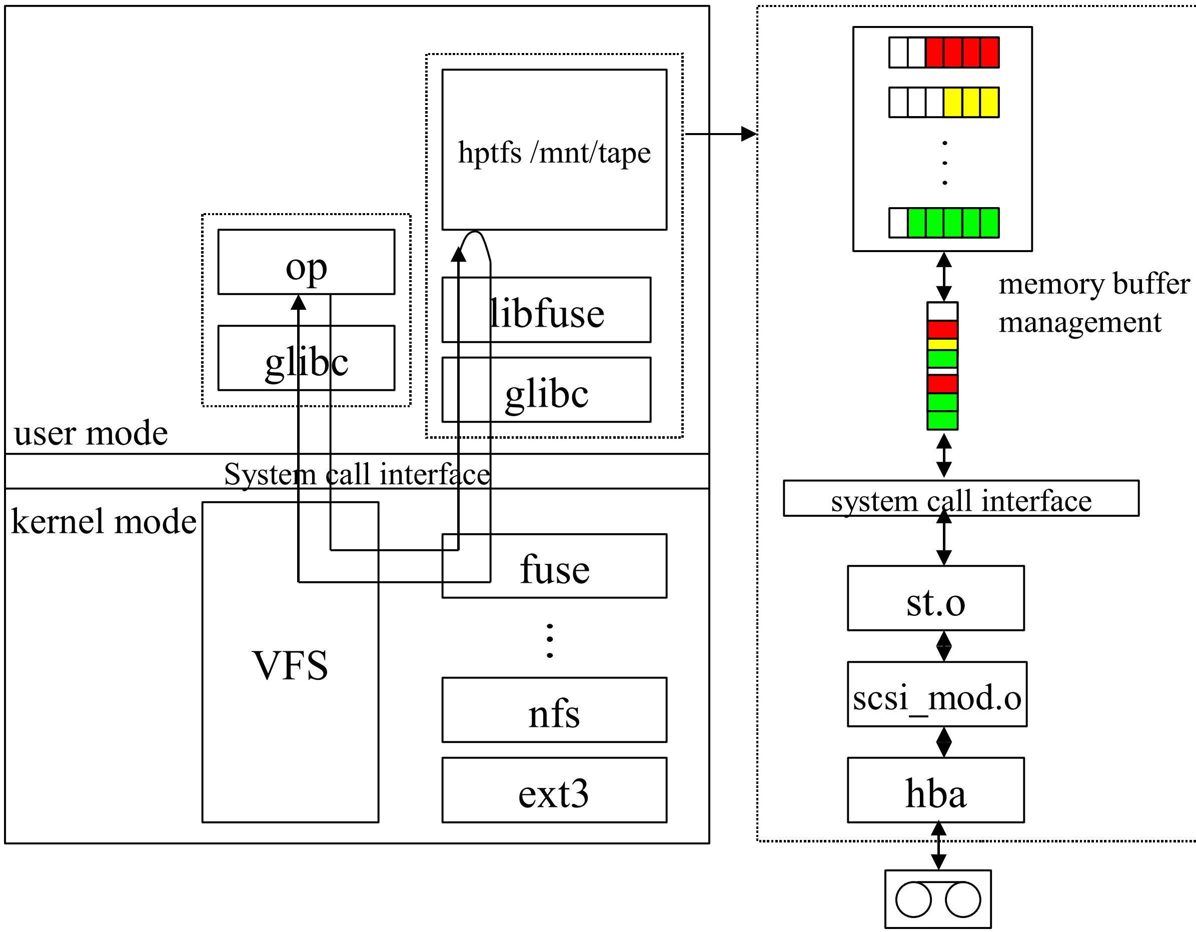
Project Motivation

- Tape is needed more than ever due to the explosive data growth rate from content-rich applications and compliance requirements
- To avoid disasters and human errors, critical data are usually backed up to tape and kept off-site
  - Network transmission is not so fast as people expect considering data size of 100’s GB
- Reducing the time to move massive data from disk to tape is critical for the data safety
- Easy to use I/O interface is one of the keys to the further success and broader use of tape storage

System Designed Features

- Providing tape storage access with generic file system interfaces
- Containing user data and corresponding metadata (including directory data) on the same tape.
- Moving data to the final destination – tapes – with streaming speed and does not involve any disk staging
  - Data can be read from tape by application directly without involving disks along the data path.
- Supporting tape drive write sharing with transparent data interleaving

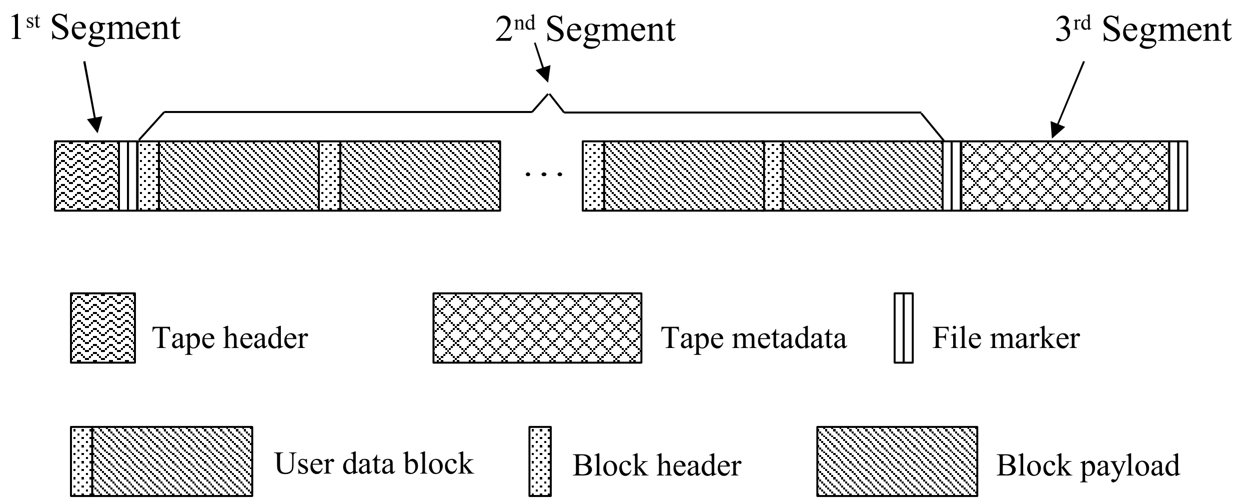
System Architecture



Tape Data Residing on Tape

- Tape data is self-contained and light-weighted
  - This is different from any tape file system in the old days
- User data and metadata
  - Each tape maintains three data segments: tape header, user data and metadata
  - Metadata contains object id, start position and end position
  - Metadata can be stored at the end of a tape or in tape cartridge embedded memory chip

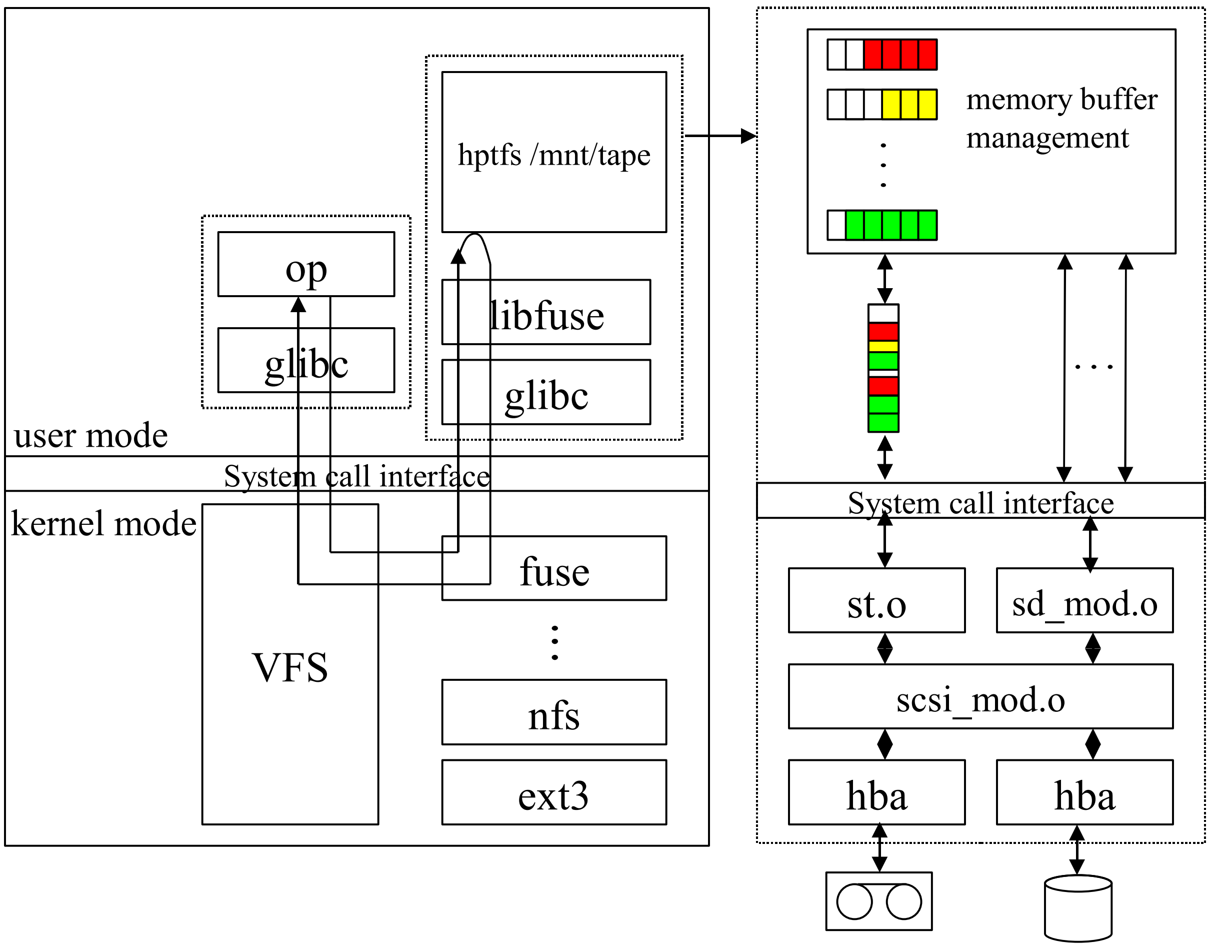
Tape Data Layout & Structure



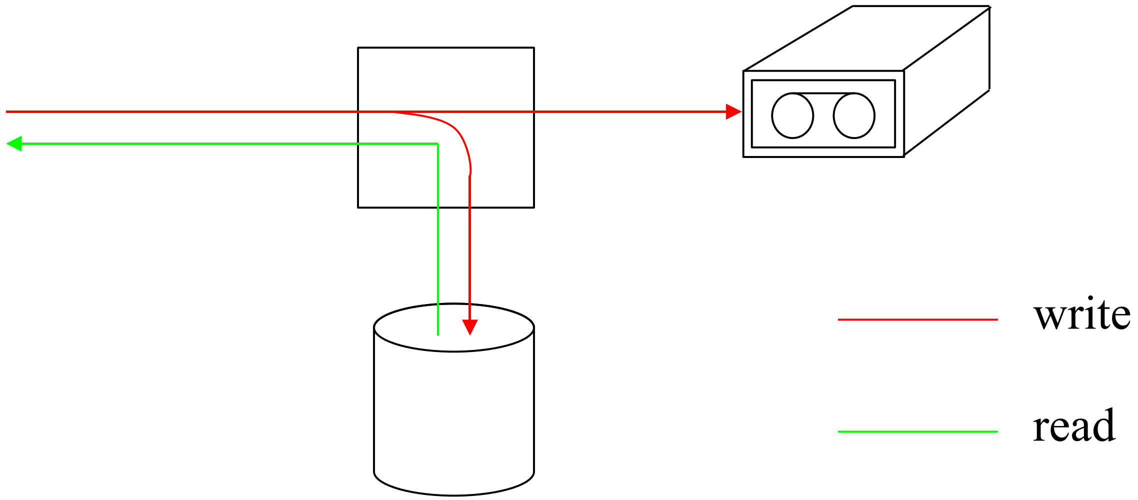

```
struct objid
{
    int vol;
    int f_no;
    int b_sp;
    int seq;
};
```

```
struct tapemeta
{
    char name[1024];
    int f_no;
    int b_sp;
    int b_ep;
    struct objid id;
    struct stat stbuf;
    struct tapemeta *next;
};
```

Write to Tape & Disk Simultaneously



Read while Write



- Disk serves read operations while tape writes in streaming mode
  - Tape read operation is expensive during tape writing process
- Disk can only hold a short period of data while tape library has "infinite" capacity
  - Requires smart purge for high performance

Example Usage of HPTFS

Commands and outputs	Notes
[root@oak lib]#./HPTFS /mnt/tape /home/xzhang/tape w	Mount tape in write mode at /mnt/tape
[root@oak lib]# ls -lt ∗.c -rw-r–r– 1 root root 61725 Jun 2 04:50 fuse.c -rw-r–r– 1 root root 12461 Jun 2 04:50 helper.c -rw-r–r– 1 root root 5064 Mar 21 05:37 fuse_mt.c -rw-r–r– 1 root root 3045 Feb 2 2005 mount.c	List all C files under current folder (on disk)
[root@oak lib]# cp ∗.c /mnt/tape	Copy all C files from disk to tape
[root@oak lib]#fusermount -u /mnt/tape	Write out metadata to tape and umount tape
[root@oak lib]#./HPTFS /mnt/tape /home/xzhang/tape r	Mount tape in read mode at /mnt/tape
[root@oak lib]#ls -lt /mnt/tape -rw-r–r– 1 root root 61725 Aug 15 23:55 fuse.c -rw-r–r– 1 root root 5064 Aug 15 23:55 fuse_mt.c -rw-r–r– 1 root root 12461 Aug 15 23:55 helper.c -rw-r–r– 1 root root 3045 Aug 15 23:55 mount.c	List all C files on tape media

Performance Evaluation

Setting A: slow host with PIII 500Mhz cpu, 256 MB + STK 9840A tape drive

Setting B: faster host with four Intel(R) XEON 2.40GHz cpu's, 3GB + STK 9940A tape drive

Main observations:

- User applications directly write/read data to/from tape without the knowledge of tape storage
- Support concurrent writes nicely
- Stream tape drive if enough data are provided

Part of the Performance Results

Table 2. Tape write performance (MB/s, tape block size=256KB)

Degree of concurrency	Setting A rate		Setting B rate	
	Mean	Stdv	Mean	Stdv
2	24.148	0.433	37.709	0.004
3	24.222	0.392	37.713	0.005
4	24.169	0.373	37.719	0.005

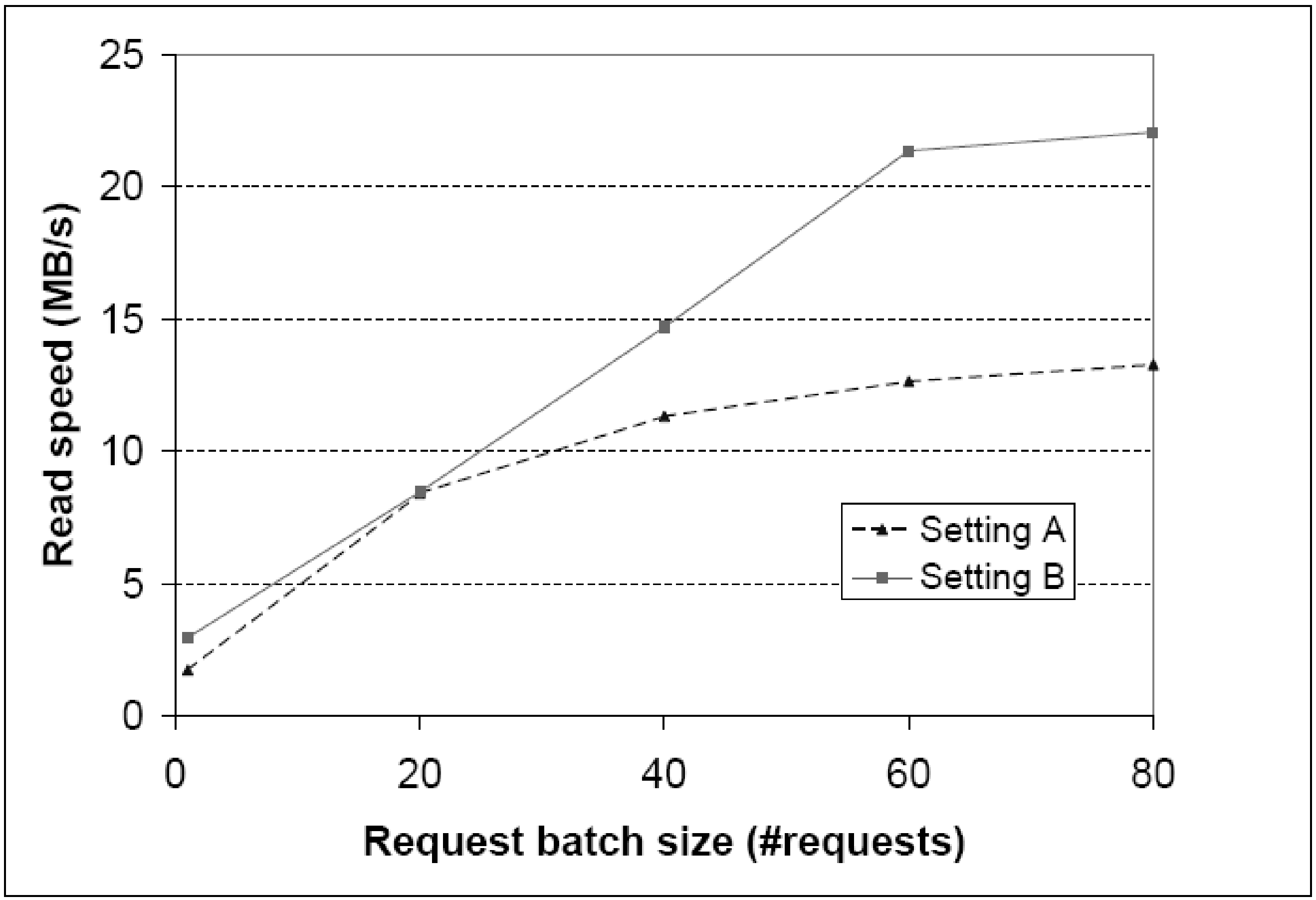
Note: write speeds of Setting A and B are rated as 29.759 MB/s and 37.604MB/s respectively

Tape Random Read Performance from PostMark

Table 7. Tape read performance with Post-Mark(MB/s, tape block size=256KB)

Degree of interleaving	Setting A rate		Setting B rate	
	Mean	Stdv	Mean	Stdv
1	1.750	0.021	2.975	0.106
2	1.835	0.049	2.754	0.014
3	1.695	0.034	2.265	0.021
4	1.470	0.127	2.085	0.022

Tape Random Read Performance with PostMark (1,000 files and 100 read operations)

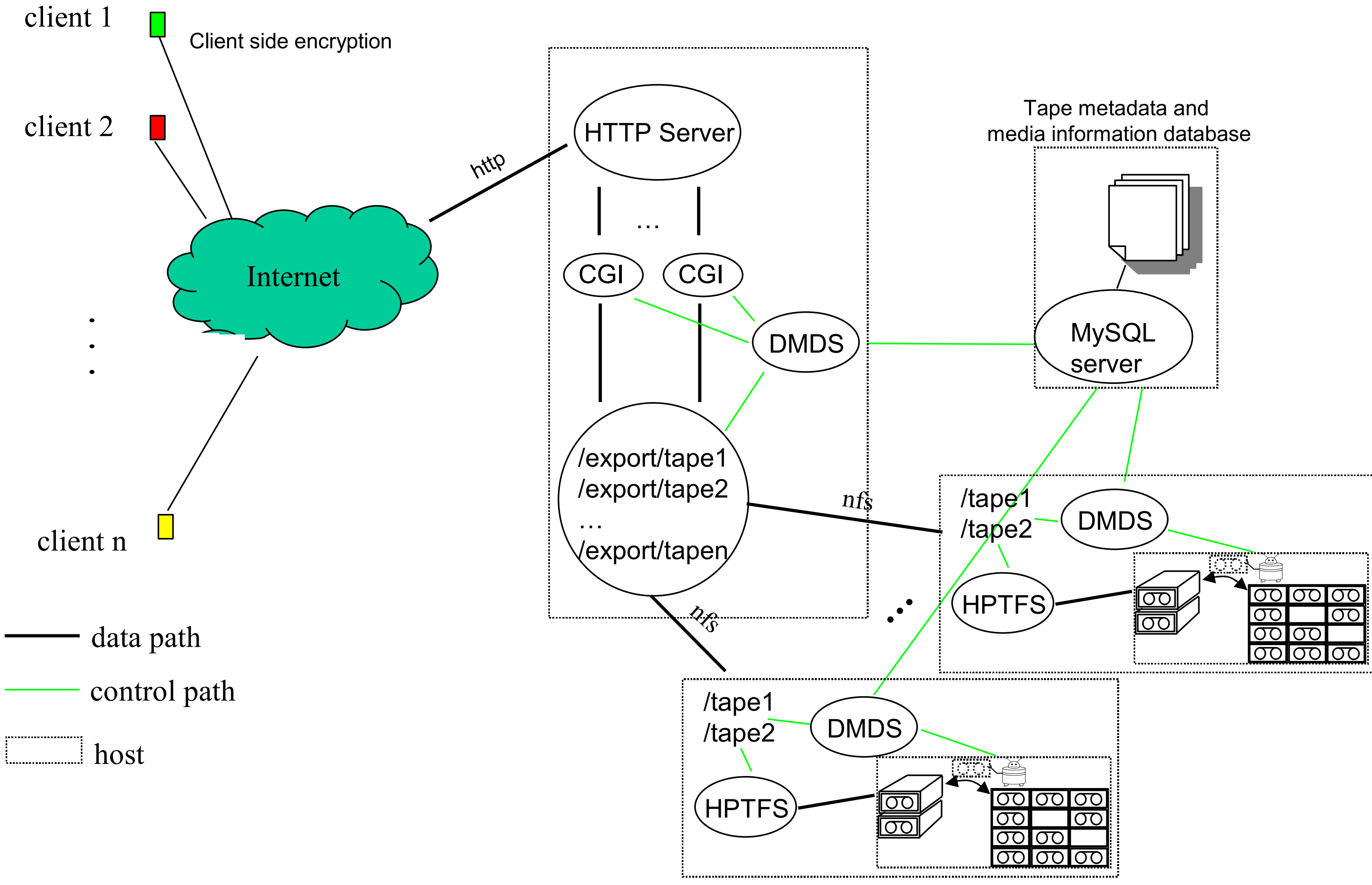


File Signature Comparison

Table 3.9: "Screenshot" and annotation

Commands and outputs	Notes
oak%./HPTFS /mnt/tape w	Mount tape in write mode at /mnt/tape
oak% ls -lt ∗.c -rw-r–r– 1 root root 61725 Jun 2 04:50 fuse.c -rw-r–r– 1 root root 12461 Jun 2 04:50 helper.c -rw-r–r– 1 root root 5064 Mar 21 05:37 fuse_mt.c -rw-r–r– 1 root root 3045 Feb 2 2005 mount.c	List all C files under current folder (on disk)
oak% cp ∗.c /mnt/tape	Copy all C files from disk to tape
oak%fusermount -u /mnt/tape	Write out metadata to tape and umount tape
oak%./HPTFS /mnt/tape r	Mount tape in read mode at /mnt/tape
oak%ls -lt /mnt/tape -rw-r–r– 1 root root 61725 Aug 15 23:55 fuse.c -rw-r–r– 1 root root 5064 Aug 15 23:55 fuse_mt.c -rw-r–r– 1 root root 12461 Aug 15 23:55 helper.c -rw-r–r– 1 root root 3045 Aug 15 23:55 mount.c	List all C files on tape media
oak% cp /mnt/tape/fuse.c ./fuse_1.c	Copy fuse.c from tape to disk as fuse_1.c
oak% openssl OpenSSL> sha1 fuse.c SHA1(fuse.c)= c8ab9be7c2edc1128db66f877b40ceeafffb74f6 OpenSSL> sha1 fuse_1.c SHA1(fuse_1.c)= c8ab9be7c2edc1128db66f877b40ceeafffb74f6	Comparing the original fuse.c on disk to fuse_1.c copied from tape with SHA1.

“Infinite” Online Backup/Archive Storage



Conclusions

- HPTFS provides generic file system interface for tape data access: writing to tape is as easy as writing to disk
- Provides tape drive sharing with high performance
- Built over HPTFS, software for backup and HSM can be made simpler
- Potential to embed HPTFS functionality into tape drive totally changing tape access paradigm
- OSD interface can be easily provided over HPTFS