Tertiary Storage

Keith Fitzgerald - Harvard Holmes

NERSC File Storage Group

Storage Charging: Questions and Answers

Why charge for storage?

The answer is for the same reason we charge for CPU time on our compute servers. Our resources have limitations and we have the responsibility to provide a deterministic, automated mechanism which insures that the resources are utilized as DOE desires. Our experience has led us to believe that the most critical resources in the storage environment are:

1. Bandwidth. This limitation can show up in many areas .... network, disk cache, archive, etc. but the bottom line is that when you approach a limit, users suffer.
2. The name server. When this is overloaded, service degrades.
3. The archive. We've already experienced and fixed this problem. l> A good management scheme should provide a deterministic mechanism which will encourage users to optimize their utilization of the storage environment and reward those users who invest the thought and time required to optimize their usage. The only management mechanism which provides the desired capabilities is charging for storage with the ability to trade storage for computation. The other alternatives considered were quotas and separate storage allocations.

   The quota mechanism is deficient in almost every way. It protects only the archive. There is no way to influence user behavior in terms of bandwidth or name server usage, and there is no incentive to use less than the entire storage quota. Charging a separate storage allocation provides a better management mechanism because there would be a limit on the total resource consumed and a tradeoff between bandwidth, archival storage and number of files. However, there would be no incentive to use less than the entire allocation--therefore no incentive to cleanup. Costs for resources and their management vary, especially in how they scale. Charging can assist in the process of making sure that resources are used and managed in a cost effective way. Charging and the corresponding statistics can also assist in justifying future expansion and changes in configurations.

   Assuming we adopt a storage charging scheme based upon CRUs, what stops users from increasing their computational allocation by requesting storage which they don't plan to use?

   The best answer I can give is nothing stops someone from doing this once. However, the ERCAP process may not be very understanding when they submit their next allocation request - it would be pretty obvious what had happened at the end of the year.

   How would we convert to a charging scheme?

   If a charging scheme is approved, we would like to put the mechanism into place as soon as possible, but NO CRUs would be deducted from any repository until the beginning of the next fiscal year. Hopefully, you would have several months of experience with the charging scheme which could be used as the basis for your next year's ERCAP allocation request. At the start of the next fiscal year ('98) we would begin actually charging the repositories.

   Where does the superhome fit in?

   Current thinking is that a reasonable amount (more than 500MB) of global storage will be provided for each user. This storage will be free and will be managed under a quota mechanism. A user with small to moderate storage needs may never have to access the storage system directly. Users with large storage needs may want to use the superhome to satisfy their small file requirement. At some time in the future, the superhome may evolve into a DFS based capability which will also be accessible directly from your workstation.

Review of MSS systems for NERSC

1. Requirements
2. Vendor Survey
3. User Survey
4. Preliminary Conclusions
5. More Information

Server Graphics:

6. "Classical" Server
7. "Third Party" Server
8. "Third Party" Server with Client Movers
9. "Third Party" Server with Pooled Movers

Requirements>

User Level Requirements

- Reliability -- users do not want any files lost ever!
- Performance -- users want:
  - high capacity
  - high transfer rates
  - quick file operations l>
  - Convenience -- users want uniform access to all their files, including those in Mass Storage l>

    Technical Requirements

    Performance:

    - Scalability: (within a single system image/name space)
    - Disk Speed: utilize full capabilities of our hardware (500MB/s or more)

- Tape Speed: utilize full capabilities of our hardware (200MB/s or more) l> Support for Existing Hardware:
    - This is not hard and fast, but a question of economics l> Reliability and Stability:
        - Multiple copies of data: at least 4 copies allowed
        - Metadata integrity and security: backup, logging, and mirroring
        - Broad user base: we'd like to see 10 sites with configurations similar to ou rs
        - Operation in degraded mode: don't give up if you don't have to l> Functionality and Extensibility:
            - APIs and libraries for a wide variety of clients: to support application lev el access to MSS and to support MSS versions of utilities like cp, ls, chmod, et c.
            - Servers for transparent file access: NFS and/or DFS l> Management Provisions:
                - Configuration ease: both initially and as the system changes and grows
                - Monitoring ease: clear and specific status, operational and error messages l>

Vendor Survey: Who are the players?>

First Cut using Infotech "The Mass Storage Report 1996"
Some adjustments from other site experiences/plans

	High Performance	**Many Files**	**Large Files**	**Existing Hardware**	**Pass/Fail**
HPSS	**Yes**	**Yes**	**Yes**	**Yes**	**Pass**
Convex UniTree	**Yes**	**Yes**	**Yes**	**No**	**Maybe-hw**
DMF	**Yes**	**Yes**	**Yes**	**Yes**	**Pass**
FileServ	**Yes**	**Yes??**	**Yes**	**No**	**Maybe-hw**
AMASS	**No**	**No**	**??**	**No**	**Fail**
Epoch	**No**	**No**	**No**	**No**	**Fail**
ADSM	**??**	**Yes**	**??**	**Yes**	**????**
SAM-FS	**Yes**	**Yes**	**Yes**	**No(Sun/SGI only)**	**Maybe-hw**
CA-Unicenter	**Yes??**	**Yes??**	**??**	**No(no Cray client)**	**Fail**
Metior	**Yes**	**Yes**	**Yes**	**No**	**Maybe-hw**
Data Migrator	**--**	**--**	**--**	**No**	**Fail**
OSM	**No**	**No**	**No**	**Yes**	**Fail**
MastarMind	**No**	**No??**	**Yes**	**No**	**Fail**

User Survey: Summary of Operations at Other Centers>

Site	**Supercomputer**	**Mass Storage**	**Comments**
Argonne/ECT	**--**	**none**	**plans based on DFS**
Brookhaven/RHIC Planning	**UNIX farm**	**none**	**Metior, HPSS planned**
Cornell Theory Center	**IBM SP2(512)**	**HPSS**	**--**
ECMWF	**Fujitsu VPP**	**--**	**--**
Fermilab	**--**	**HPSS**	**--**
Jefferson Lab (CEBAF)	**--**	**OSM**	**--**
LLNL	**T3D, SP2, DEC**	**UniTree** **NSL UniTree** **HPSS**	**--**
LANL, ACL	**many**	**CFS, HPSS**	**--**

Maui HPCC	**IBM SP(563)**	**HPSS**	**--**
NASA Goddard	**--**	**Convex UniTree**	**--**
NCSA	**Convex, SGI, CM-5**	**Convex UniTree**	**--**
Oakridge NL (CCS)	**--**	**HPSS, UniTree**	**HPSS testbed**
PNNL (EMSL)	**--**	**FileServ HSM**	**--**
Pittsburgh SC	**Cray**	**DMF**	**recently upgraded**
SDSC	**Cray, Intel**	**HPSS, UniTree**	**HPSS on IBM SP2**

Preliminary Conclusions>
Leading contenders are HPSS, DMF, and FileServ

	HPSS	DMF	FileServ
$$$	**$2.5M (recent upgrade)**	**??**	**$2.4M(@PNNL)**
What You Get	**2 libraries 25 tape drives 750GB disk control CPU**	**--**	**1 library 8 tape drives 400BG disk 2 big SGI systems**
Performance	**Outstanding**	**Excellent**	**Excellent**
Advantages	**scalability single system image existing user base desktop support< br>supports a variety of computing environments**	**stable well supported single-system image existing user base**	**turnkey system supports a variety of computing environments**
Staffing	**some**	**some**	**little**
Conversion Effort	**easy - UniTree moderate - CFS**	**hard**	**unknown**
Futures	**bright**	**limited to Cray environments**	**unknown at best**

RECOMMENDATION: HPSS

Time Table:

- Now --- HPSS test system running
- 3 Mos. -- HPSS in production
- 6 Mos. -- NSL UniTree converted
- 9 Mos. -- CFS converted l>

More Information>

Some Questions and Answers

What is "third party" data transfer?

It is the separation of control flows and data flows in the mass storage system. It is used to facilitate parallelism in data transfers.

How does third party transfer work without third party support in the clients?

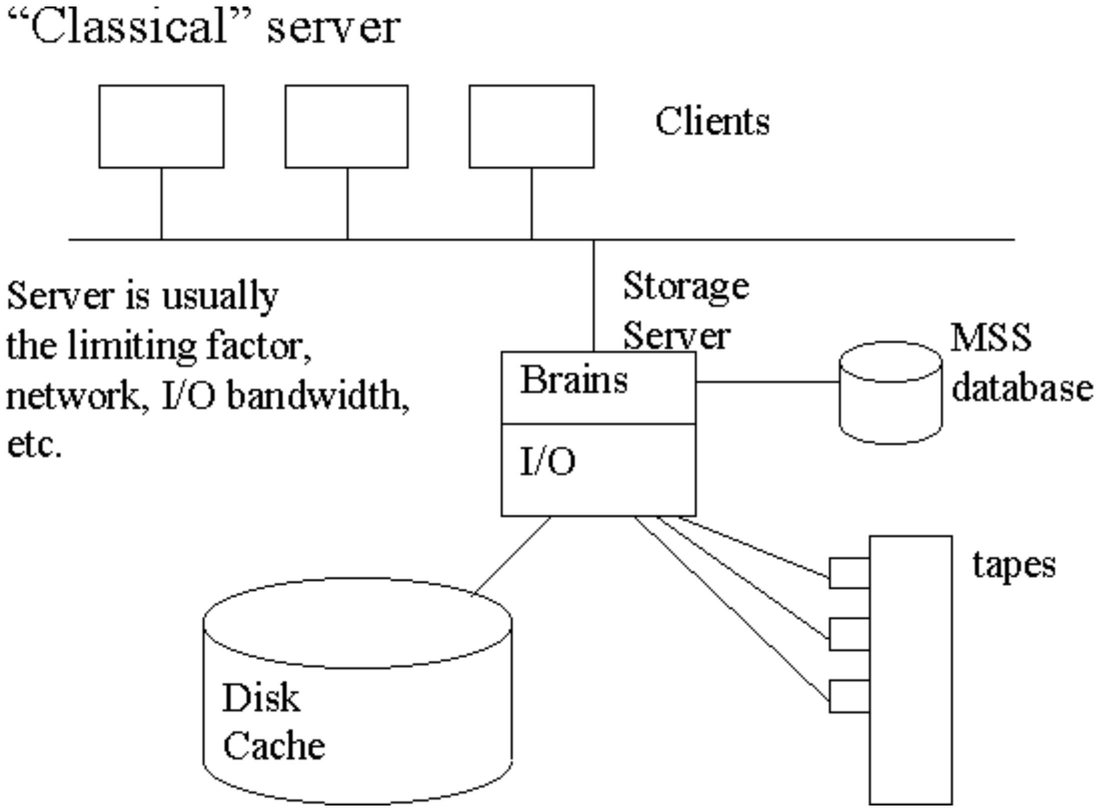
The storage system still separates control and data flows, but data traffic from clients passes through small auxiliary machines called "movers." The performance improvement of parallelism remains.

How does this relate to DMF (Cray's Data Migration Facility)?

DMF is still used to manage the Cray file systems. DMF offloads files to the mass storage system. It would be possible, with additional hardware, to connect DMF directly to tape drives.

How do we get transparent file access?

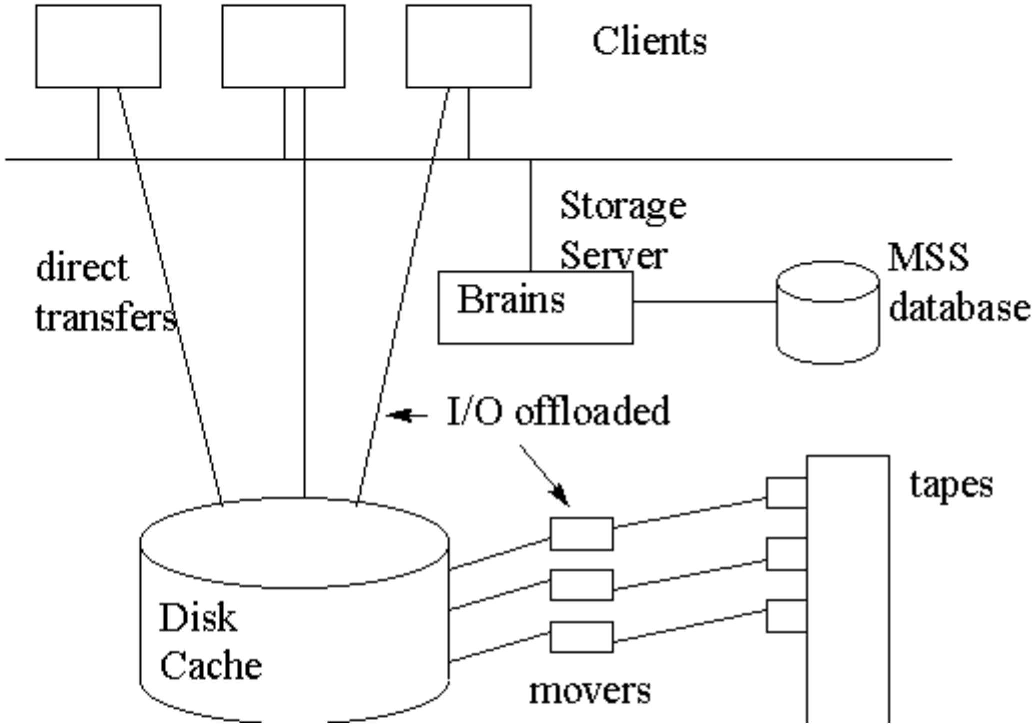
HPSS, DMF, and FileServ files can all be exported via NFS. Currently NFS has some performance and security issues. We expect much better performance using our custom utilities. We will revisit the installation of NFS after it's thoroughly tested. Our long term goal is to use DFS for our distributed file systems -- performance, security and scalability are better.



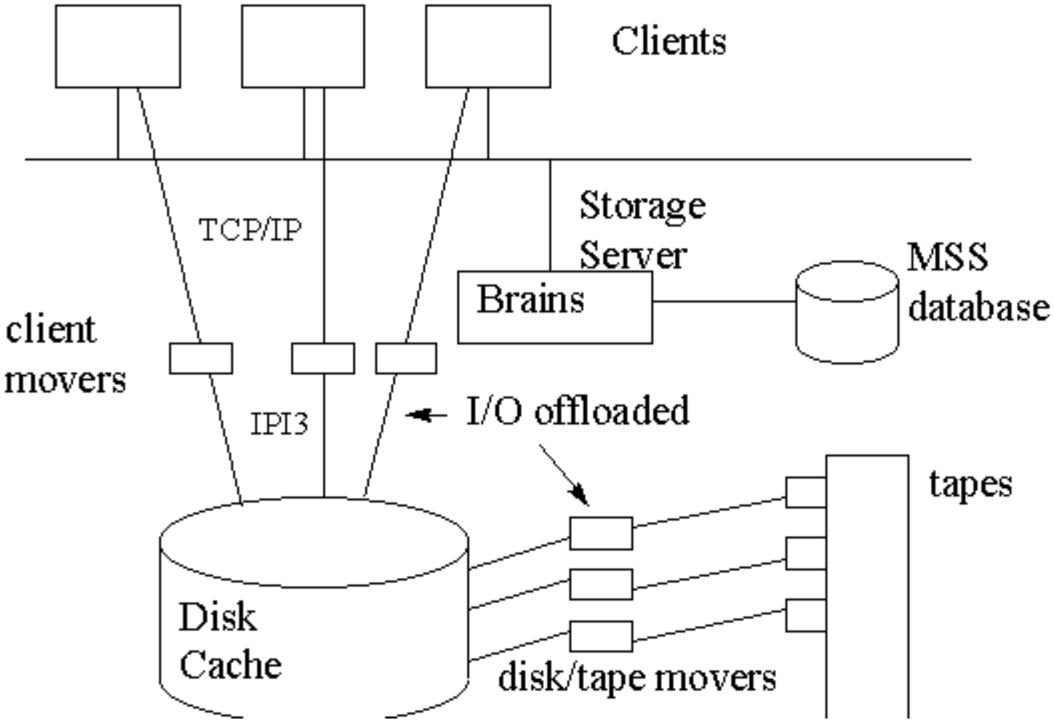
“Classical” server

Server is usually the limiting factor, network, I/O bandwidth, etc.

“Third party” server



“Third party” server with client movers



"Third Party" server with pooled movers