Mass Storage Workshop Summary

Alan Silverman

28 May 2004

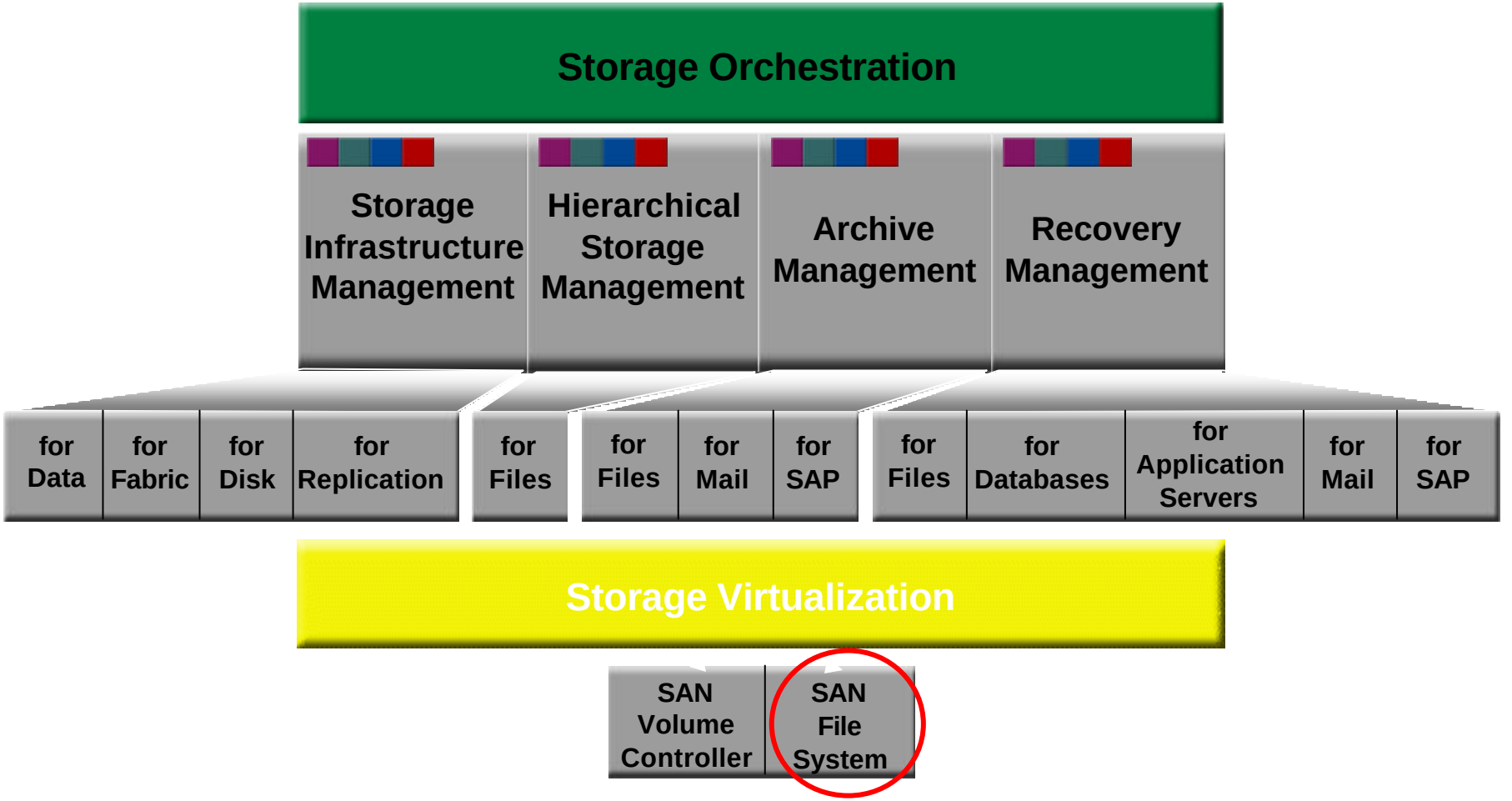
General

- First, due credit to all speakers for making these 2 days very interesting and very interactive
- Secondly, credit to Olof Barring who organised the agenda and did all the things I usually do in organising these after-HEPiX workshops
- Thanks to Dave Kelsey for overall organisation and to NESC for their generosity in hosting this meeting all week
- Apologies in advance to the speakers if I have misunderstood or mis-represented them.

Technology

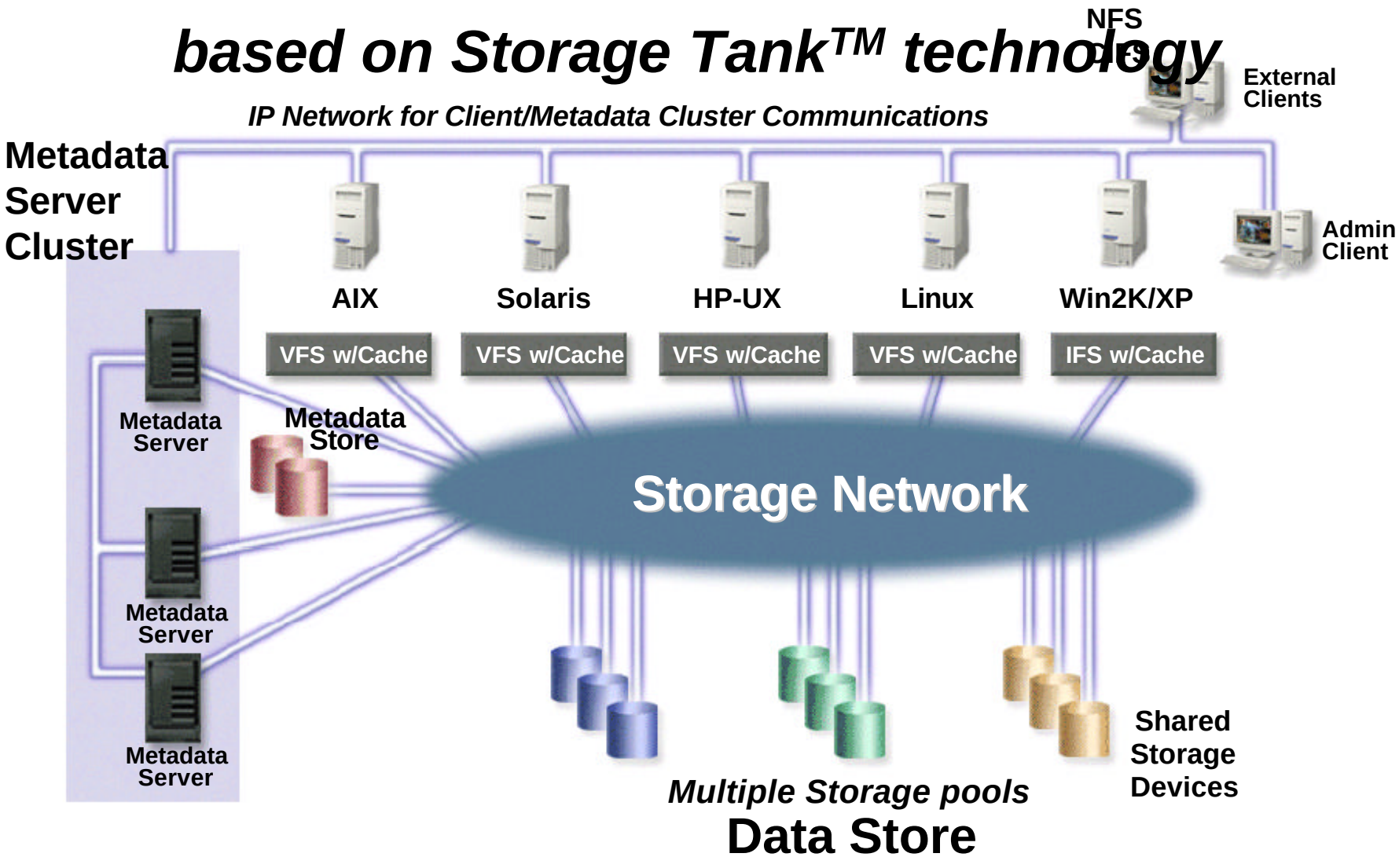
- Started with a very interesting talk from IBM on Storage Tank, otherwise known as IBM TotalStorage SAN File System. The potential interest in this product was confirmed later in the day by the CASPUR talk

IBM TotalStorage Open Software Family

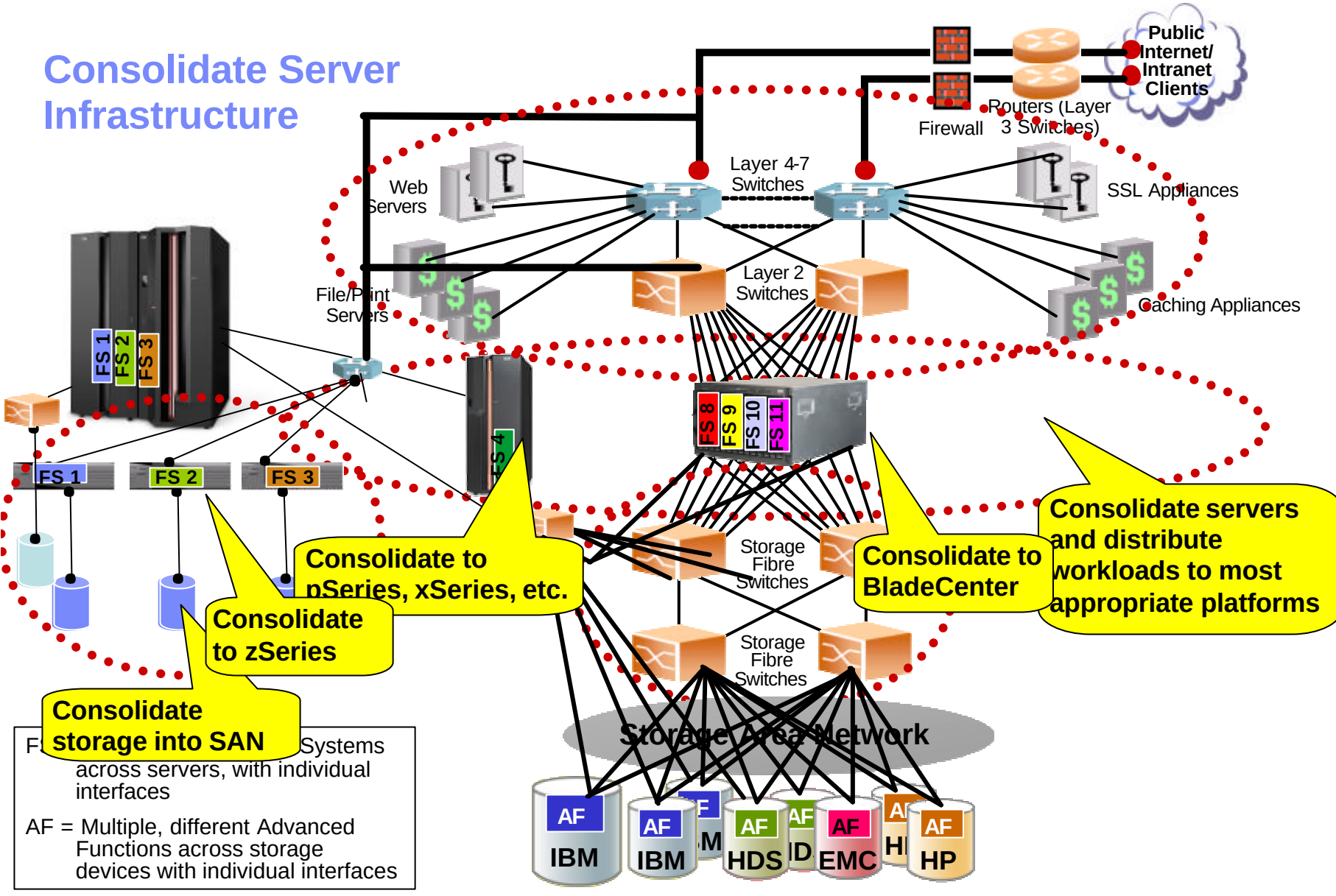


Architecture

based on Storage Tank™ technology



Consolidate Server Infrastructure



F Systems across servers, with individual interfaces

AF = Multiple, different Advanced Functions across storage devices with individual interfaces

Technology

- Started with a very interesting talk from IBM on Storage Tank, otherwise known as IBM TotalStorage SAN File System. The potential interest in this product was confirmed later in the day by the CASPUR talk
- Also a very interesting and (over-)full review of various storage-related performance tests at CASPUR.

Sponsors for these test sessions:

ACAL Storage Networking	:	Loaned a 16-port Brocade switch
ADIC Soiftware	:	Provided the StorNext file system product, actively participated in tests
DataDirect Networks	:	Loaned an S2A 8000 disk system, actively participated in tests
E4 Computer Engineering	:	Loaned 10 assembled biprocessor nodes
Emulex Corporation	:	Loaned 16 fibre channel HBAs
IBM	:	Loaned a FASTt900 disk system and SANFS product complete with 2 MDS units, actively participated in tests
Infortrend-Europe	:	Sold 4 EonStor disk systems at discount price
INTEL	:	Donated 10 motherboards and 20 CPUs
SGI	:	Loaned the CXFS product
Storcase	:	Loaned an InfoStation disk system

Goals for these test series

1. **Performance of low-cost SATA/FC disk systems**
2. **Performance of SAN File Systems**
3. **AFS Speedup options**
4. **Lustre**
5. **Performance of LTO-2 tape drive**

Technology

- Started with a very interesting talk from IBM on Storage Tank, otherwise known as IBM TotalStorage SAN File System. The potential interest in this product was confirmed later in the day by the CASPUR talk
- Also a very interesting and (over-)full review of various storage-related performance tests at CASPUR.
- Information Lifecycle Mgmt talk by STK had perhaps a little too much marketing but there were some interesting glimpses of what STK has to offer and the Sanger Trust is an impressive reference site.

THE RELEVANCE OF ILM TODAY

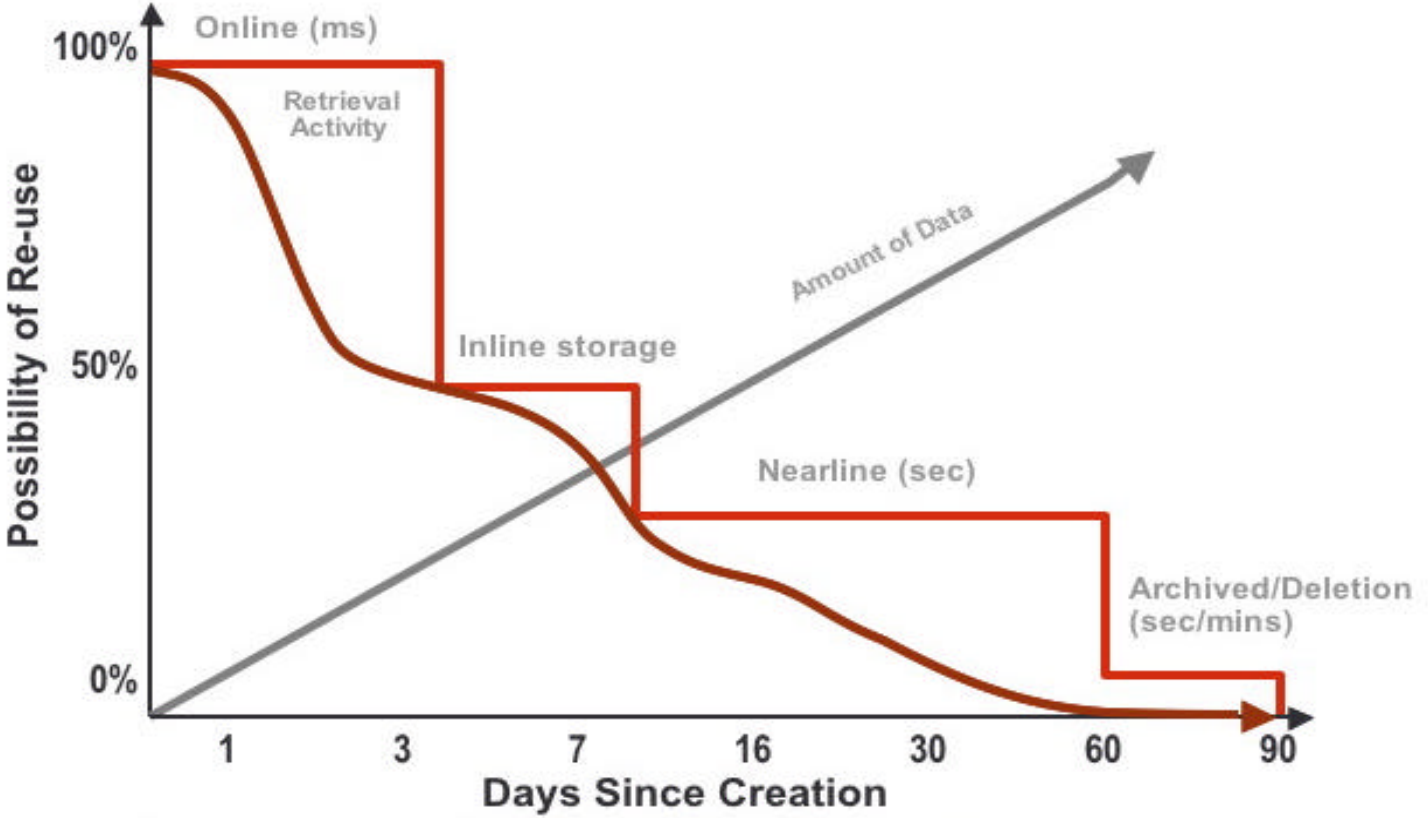
Information Lifecycle Management (ILM)

"Classifying, managing, and moving information to the most cost effective data repository based on the value of each piece of information at that exact point in time."

Implications:

- **Not all information is created equal…and neither are your storage options**
- **Information value changes over time…both upward and downward**
- **Data repositories should be *dynamically matched* with information value for security, protection and cost**

Understanding the Business Value



Source: Horison Information Strategies, Storage Technology Corp.

Objective: Align storage cost with your information's value
Value can be relative to age, criticality (business process) &/or time

Middleware

- Good overview of Storage Resource Broker from SDSC; interesting new concept “semi open source”

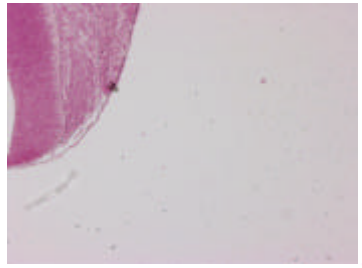
What is SRB? (1 of 3)

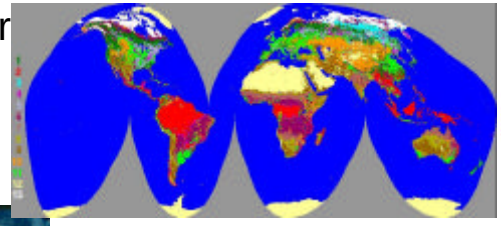
- The SDSC Storage Resource Broker (SRB) is client-server middleware that provides a uniform interface for connecting to heterogeneous data resources over a network and accessing unique or replicated data objects.

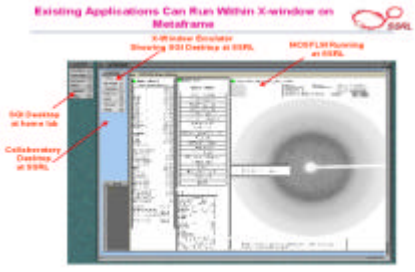
- SRB, in conjunction with the Metadata Catalog (MCAT), provides a way to access data sets and resources based on their logical names or attributes rather than their names and physical locations.

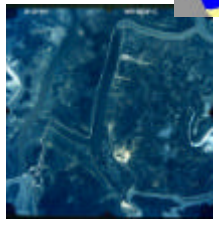
SRB Projects

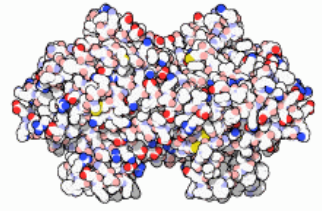
- **Digital Libraries**
  - UCB, Umich, UCSB, Stanford,CDL
  - NSF NSDL - UCAR / DLESE
- **NASA Information Power Grid**
- **Astronomy**
  - National Virtual Observatory
  - 2MASS Project (2 Micron All Sky Survey)
- **Particle Physics**
  - Particle Physics Data Grid (DOE)
  - GriPhyN
  - SLAC Synchrotron Data Repositor
- **Medicine**
  - Digital Embryo (NLM)
- **Earth Systems Sciences**
  - ESIPS
  - LTER
- **Persistent Archives**
  - NARA
  - LOC
- **Neuro Science & Molecular Scien**
  - TeleScience/NCMIR, BIRN
  - SLAC, AfCS, …

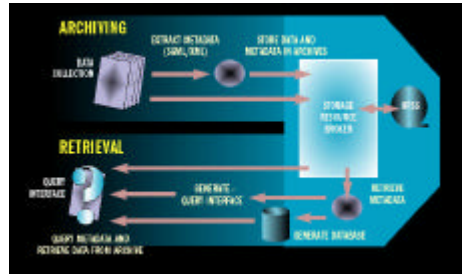












Over 90 Tera Bytes in 16 million files

Storage Resource Broker

- SRB wears many hats:
  - It is a distributed but unified file system
  - It is a database access interface
  - It is a digital library
  - It is a semantic web
  - It is a data grid system
  - It is an advanced archival system

Middleware

- Good overview of Storage Resource Broker from SDSC; interesting new concept “semi open source”
- Real-life experiences of interfacing requests to Mass Storage systems from EDG WP5 at RAL using the now widely-used SRM (Storage Resource Manager) protocol. Lessons learned include
  - look for opportunities for software reuse
  - realise that prototypes often last longer than expected

Objectives

- Implement *uniform* interfaces to mass storage
  - Independent of underlying storage system
- SRM
  - Uniform interface – much is optional
- Develop back-end support for mass storage systems
  - Provide “missing” features – directory support?
- Publish information

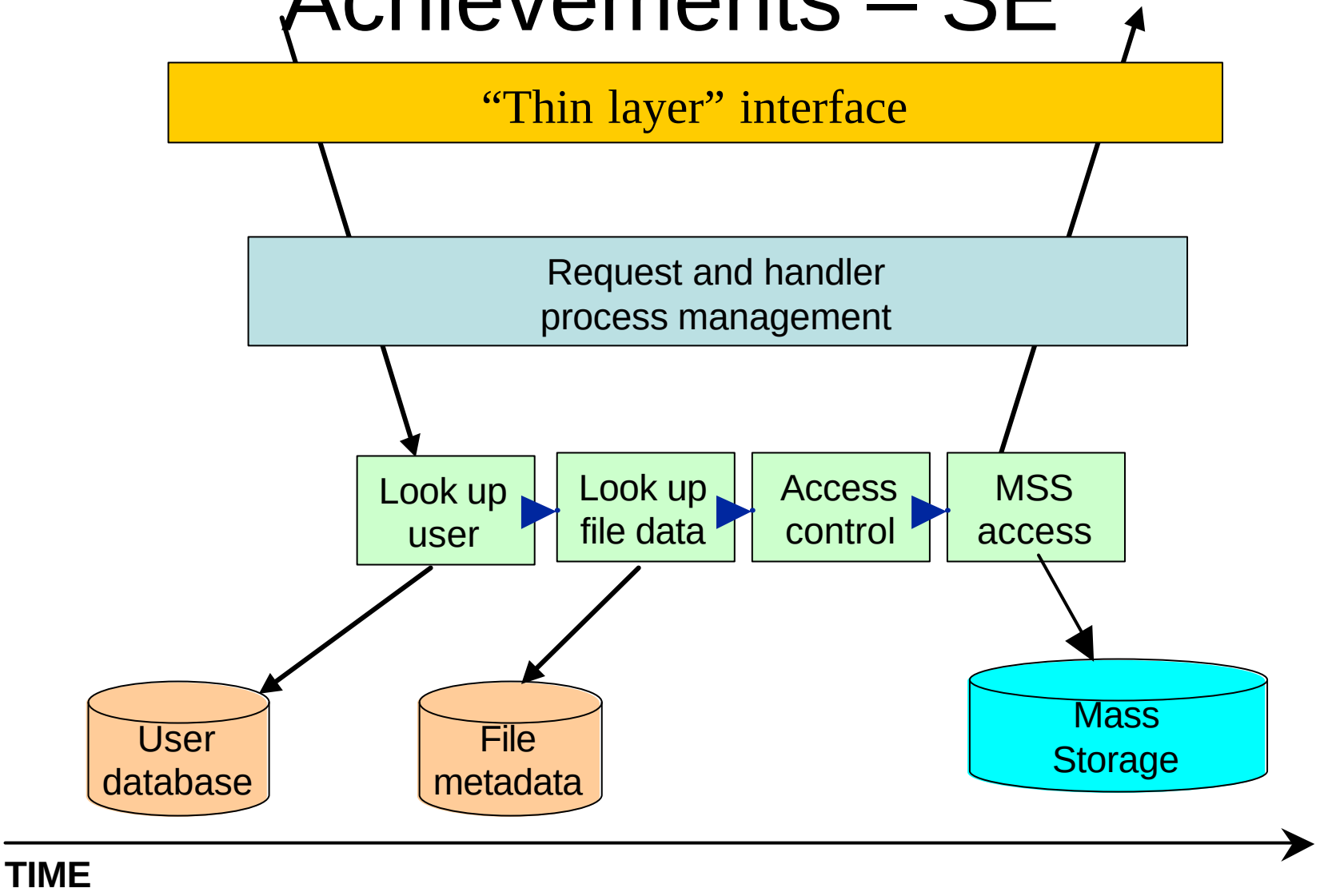
Objectives – SRM

- SRM 1 provides async **get**, **put**
  - **get** (**put**) returns request id
  - **getRequestStatus** returns status of request
  - When status = Ready, status contains *Transfer URL* – aka *TURL*
  - Client changes status to Running
  - Client downloads (uploads) file from (to) TURL
  - Client changes status to Done
- Files can be **pin**ned and **unpin**ned

Achievements

- In EDG, we developed EDG Storage Element
  - Uniform interface to mass storage and disk
  - Interfaces with EDG Replica Manager
  - Also client command line tools
  - Interface was based on SRM but simplified
    - Synchronous
    - Trade-off between “getting it done soon” and “getting it right the first time”
    - Additional functionality such as directory functions
  - Highly modular system

Achievements – SE



Middleware

- Good overview of Storage Resource Broker from SDSC; interesting new concept “semi open source”
- Real-life experiences of interfacing requests to Mass Storage systems from EDG WP5 at RAL using the now widely-used SRM (Storage Resource Mgr) protocol. Lessons learned include
  - look for opportunities for software reuse
  - realise that prototypes often last longer than expected
- Description of the work being done for GFAL; not yet well accepted by users but working to answer their concerns for the next round of data challenges, especially in performance. Both GFAL and SRM are included in the LCG-2 release

Common interfaces

- Why?
  - Different grids: LCG, Grid3, Nordugrid
  - Different Storage Elements
  - Possibly different File Catalogs
- Solutions
  - Storage Resource Manager (SRM)
  - Grid File Access Library (GFAL)
  - Replication and Registration Service (RRS)

Storage Resource Manager

- Goal: agree on single API for multiple storage systems
- Collaboration between CERN, FNAL, JLAB and LBNL and EDG
- SRM is a Web Service
  - Offering Storage resource allocation & scheduling
  - SRMs DO NOT perform file transfer
  - SRMs DO invoke file transfer service if needed (GridFTP)
- Types of storage resource managers
  - Disk Resource Manager (DRM)
  - Hierarchical Resource Manager (HRM)
- SRM is being discussed at GGF and proposed as a standard

Grid File Access Library (1)

- Goals
  - Provide a Posix I/O interface to heterogeneous Mass Storage Systems in a GRID environment
  - A job using GFAL should be able to run anywhere on the GRID without knowing about the services accessed or the Data Access protocols supported

GFAL File System

- GFALFS now based on FUSE (Filesystem in USErspace) file system developed by Miklos Szeredi
- Uses:
  - VFS interface
  - Communication with a daemon in user space (via character device)
  - The metadata operations are handled by the daemon, while the I/O (read/write/seek) is done directly in the kernel to avoid context switches and buffer copy
  - Requires installation of a kernel module fuse.o and of the daemon gfalfs
  - The file system mount can be done by the user

Current status (1)

- SRM
  - SRM 1.1 interfaced to CASTOR (CERN), dCache (DESY/FNAL), HPSS (HRM at LBNL)
  - SRM 1.1 interface to EDG-SE being developed (RAL)
  - SRM 2.1 being implemented at LBNL, FNAL, JLAB
  - SRM “basic” being discussed at GGF
  - SRM is seen by LCG as the best way currently to do the load balancing between GridFTP servers. This is used at FNAL.

Current status (2)

- EDG Replica Catalog
  - 2.2.7 (improvements for POOL) being tested
  - Server works with Oracle (being tested with MySQL)
- EDG Replica Manager
  - 1.6.2 in production (works with classical SE and SRM)
  - 1.7.2 on LCG certification testbed (support for EDG-SE)
  - Stability and error reporting being improved

Current status (3)

- Disk Pool Manager
  - CASTOR, dCache and HRM were considered for deployment at sites without MSS.
  - dCache is the product that we are going to ship with LCG2 but this does not prevent sites having another DPM or MSS to use it.
  - dCache is still being tested in the LCG certification testbed

Current status (4)

- Grid File Access Library
  - Offers Posix I/O API and generic routines to interface to the EDG RC, SRM 1.1, MDS
  - A library lcg_util built on top of gfal offers a C API and a CLI for Replica Management functions. They are callable from C++ physics programs and are faster than the current Java implementation.
  - A File System based on FUSE and GFAL is being tested (both at CERN and FNAL)

Panel

- In a panel concerned with LCG data management issues, CERN listed what is felt necessary to build up LCG towards first data taking and subsequent data distribution by the experiments. The idea is to start with the simplest form of data distribution, disc to disc file copy over a sustained period (one week, without interruption if possible) using a 10Gbit line to a single Tier 1 site.
- If successful, this would broadened to multiple sites first in series and then in parallel.
- The next stage would be to add LCG middleware components such as SRM and so on.

Panel - 2

- The different Tier 1 sites represented were polled as to how ready they were, in terms of both network bandwidth, disc server capacity and local support, to participate
- The sites requested more concrete plans and a detailed plan was begun and will be completed in the near future and circulated to the Tier 1 sites
- The first tests should start already this summer

Data Management Service Challenge

Scope

- Networking, file transfer, data management
- Storage management - interoperability
- Fully functional storage element (SE)

Layered Services

  - Network
  - Robust file transfer
  - Storage interfaces and functionality
  - Replica location service
  - Data management tools

General Approach

- Evolve towards a **sustainable** service
  - Permanent service infrastructure
  - Workload generator – simulating realistic data traffic
  - Identify problems, develop solid (long-term) fixes
  - Frequent performance limits tests
    - 1-2 week periods with extra resources brought in
  - But the goal is to **integrate** this in the **standard** LCG service as soon as practicable
- Focus on
  - Service operability - minimal interventions, automated problem discovery and recovery
  - Reliable data transfer service
  - End-to-end performance

Short Term Targets

- Now (or next week) –
  - Participating sites with contact names
- End June –
  - Agreed ramp-up plan, with milestones – 2-year horizon
- Targets for end 2004 –
  1. SRM-SRM (disk) on 10 Gbps links between CERN, Triumf, FZK, FNAL, NIKHEF/SARA → **500 MB/sec (?)** sustained for **days**
  2. Reliable data transfer service
  3. Mass storage system <-> mass storage system
     1. SRM v.1 at all sites
     2. disk-disk, disk-tape, tape-tape
  4. Permanent service in operation
     - sustained load (mixed user and generated workload)
     - > 10 sites
     - **key** target is **reliability**
     - load level targets to be set

The problem (Bernd)

- One copy of the LHC raw data for each of the LHC experiments is shared among the Tier-1's
- Full copies of the ESD data (1/2 of raw data size)
- Total ~10PB/year exported from CERN
- The full machinery for doing this automatically should be in place for full-scale tests in 2006

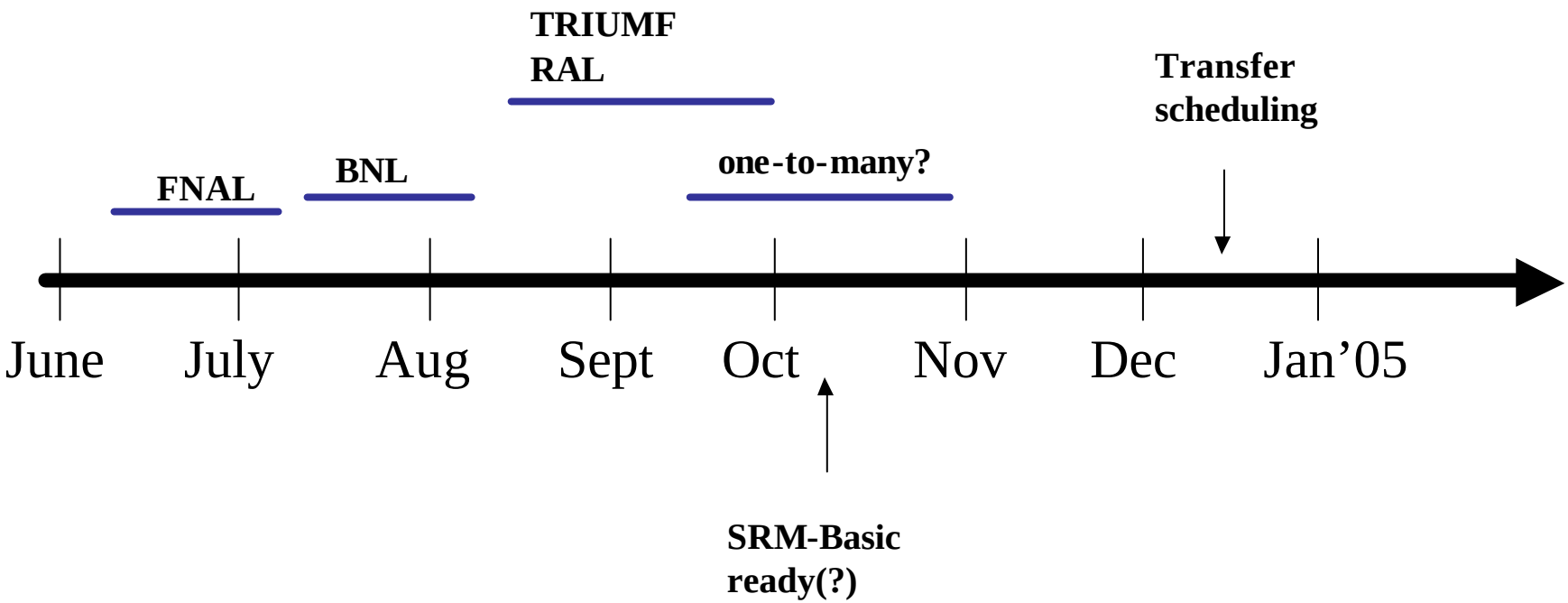
Tier-1 resources

TRIUMF	• 2 machines purchased, 1Gbit(?)
RAL	• Gbit link at present • Parallel activities from ATLAS and CMS • Not enough effort to dedicate for the moment • More hardware in September
FNAL	• Just finished CMS DC – very labor intensive • Enough resources to sustain 2TB/day
GridKA	• 1Gbit at present, expanding to 10Gbit in October/November • Storage system is ready (dCache + TSM)
BNL	• SRM service almost ready (in a month) •One gridftp node • OC12 connection, not much used
NIKHEF/SARA	• 10Gbit since more than a year •Running data challenges for experiments but mainly CPU intensive
IN2P3/Lyon	• Not yet ready with interface to MSS • 1Gbit

Agreed tests

1. Simple disk-to-disk, peer-to-peer
2. Simple disk-to-disk, one-to-many
3. MSS-to-MSS
4. In parallel?
   a) Transfer scheduling
   b) Replica catalogue & management

Timescales


TRIUMF
RAL
Transfer
scheduling
FNAL
BNL
one-to-many?
June
July
Aug
Sept
Oct
Nov
Dec
Jan'05
SRM-Basic
ready(?)

Next steps

- Statements from the other Tier-1's
  - NIKHEF, GridKA, Lyon
  - PIC, CNAF, others?
- Who is driving/coordinating? site contacts?
- Meetings ?
- Speed up SRM-basic specification process
- ...

Final Sessions

- Investigations at FNAL to match storage systems to the characteristics of wide area networking

Wide Area Characteristics

- Most prominent characteristic, compared to LAN, is the very large bandwidth*delay product.
- Underlying structure – it's a packet world!
- Possible to use pipes between specific sites
    - These circuits can be both static and dynamic
    - Both IP and non-IP (for example, Fibre-channel over sonet)
  - FNAL has proposed investigations and has just begun studies with its storage systems to optimize WAN file transfers using pipes.

Strategies

- Smaller, lower bandwidth TCP streams in parallel
  - Examples of these are GridFTP and BBftp
- Tweak AIMD algorithm
  - Logic is in the sender's kernel stack only (congestion window)
  - FAST, and others – USCMS used an FNAL kernel mod in DC04
    - May not be "fair" to others using shared network resources
- Break the stream model, use UDP and 'cleverness', especially for file transfers. But:
  - You have to be careful and avoid congestion collapse.
  - You need to be fair to other traffic, and be very certain of it
  - Isolate strategy by confining transfer to a "pipe"

Storage System and Bandwidth

- Storage Element does not know the bandwidth of individual stream very well at all
  - For example, a disk may have many simultaneous assessors or the file may be in memory cache and transferred immediately
  - Bandwidth depends on fileserver disk and your disk.
- Requested bandwidth too small?
  - If QoS tosses a packet, AIMD will drastically affect transfer rate
- Requested bandwidth too high?
  - Bandwidth at QoS level wasted, overall experimental rate suffers
- Storage Element may know the aggregate bandwidth better than individual stream bandwidth.
  - Storage Element, therefore needs to aggregate flows onto a pipe between sites, not deal with QoS on a single flow.
  - This means the local network will be involved in aggregation.

FNAL investigations

Investigate support of static and dynamic pipes by storage systems in WAN transfers.

- Fiber to Starlight optical exchange at Northwestern University.
- Local improvements to forward traffic flows onto the pipe from our LAN
- Local improvements to admit traffic flows onto our LAN from the pipe
- Need changes to Storage System to exploit the WAN changes.

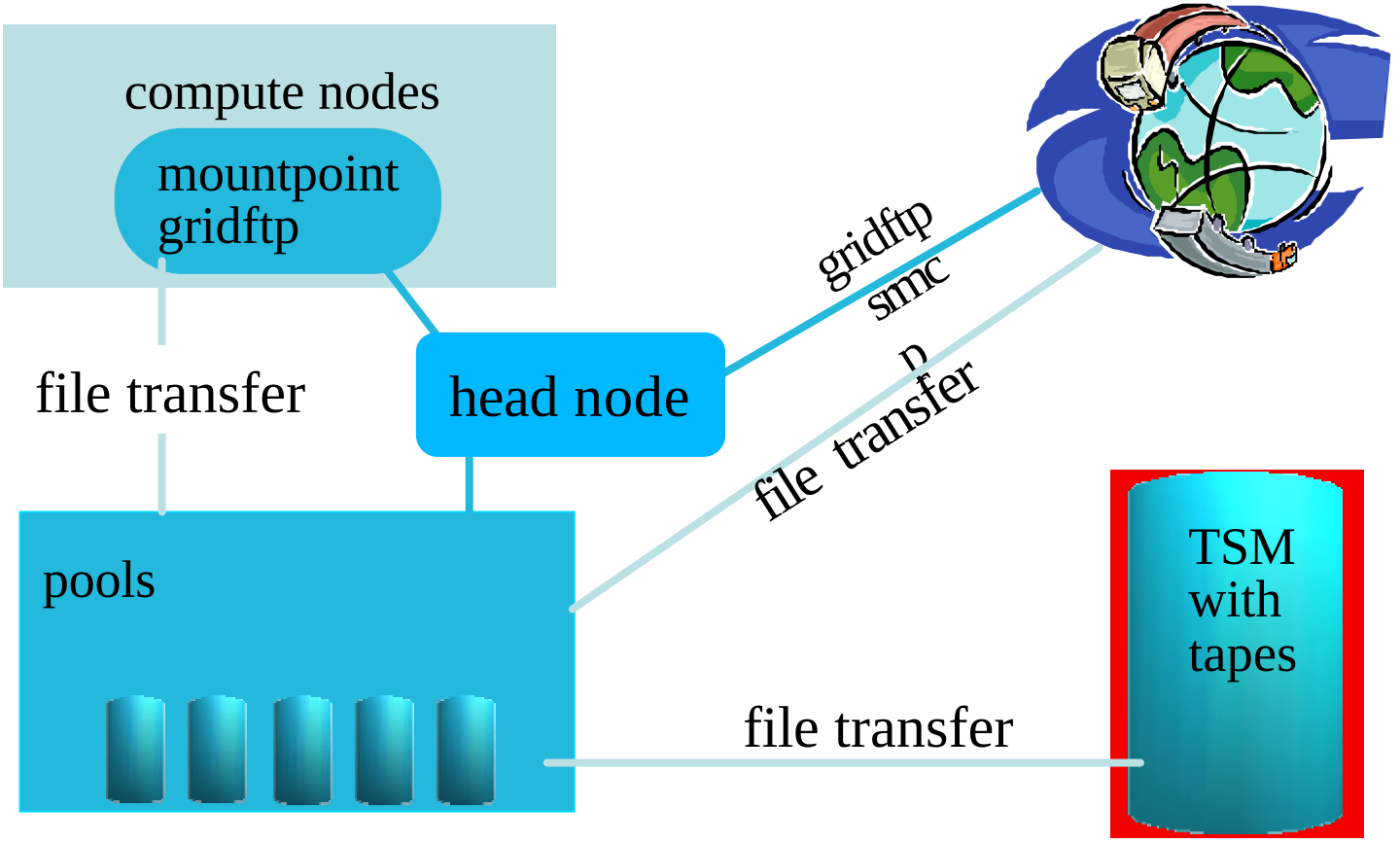
Final Sessions

- Investigations at FNAL to match storage systems to the characteristics of wide area networking
- Description of how dCache, the joint DESY/FNAL project now adopted by LCG, was integrated at GridKA

Tivoli Storage Manager (TSM)

- TSM library management
- TSM is not developed for archive
  - Interruption of TSM archive
    - No control what has been archived
- dCache (DESY, FNAL)
  - creates a separate session for every file
  - Transparent access
  - Allows transparent maintenance at TSM

dCache main components



Final Sessions

- Investigations at FNAL to match storage systems to the characteristics of wide area networking
- Description of how dCache, the joint DESY/FNAL project now adopted by LCG, was integrated at GridKA
- Experiences using CASTOR SRM 1.1 and in particular the problems met and how they were resolved

Brief overview of SRM v1.1

- SRM = Storage Resource Manager
- First (v1.0) interface definition
  - http://sdm.lbl.gov/srm-wg/doc/srm.v1.0.pdf
  - October 22, 2001
  - JLAB, FNAL and LBNL
  - Some key features:
    - Transfer protocol negotiation
    - Multi-file requests
    - Asynchronous operations
    - SRM is a management interface
      - Make files “available” for access (e.g. recall to disk)
      - Prepare resources for receiving files (e.g. allocate disk space)
      - Query status of requests or files managed by the SRM
      - ***Not*** a WAN file transfer protocol

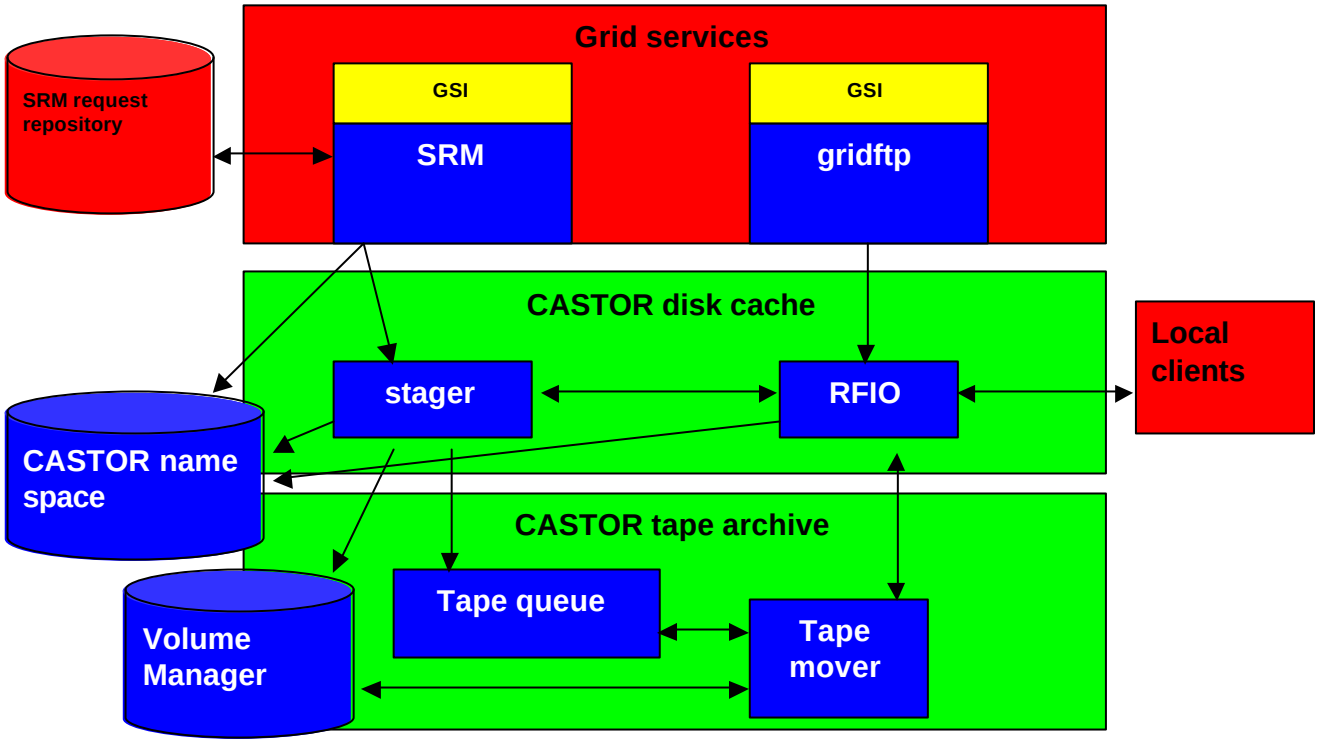
The ‘copy’ operation

- SRM v1.1 == SRM v1.0 + ‘copy’
- ‘copy’ quite different from other SRM operations:
  - Copy file(s) from/to local SRM to/from another (optionally remote) SRM
  - The target SRM performs the necessary ‘put’ and ‘get’ operations and executes the file transfers using the negotiated protocol (e.g. gsiftp)
- The ‘copy’ operation allows a batch job running on a worker node without in&out-bound WAN access to copy files to a remote storage element
- The ‘copy’ operation was documented only 4 days ago(!)
- The ‘copy’ operation could potentially provide the framework for planning transfers of a large data volumes (e.g. LHC T0 → T1 data broadcasting)??

CASTOR SRM v1.1

- Implements the vital operations
  - get, put, getRequestStatus, setFileStatus, getProtocols
- No-ops:
  - pin, unPin, getEstGetTime, getEstPutTime
- Implemented but optionally disabled (requested by LCG)
  - advisoryDelete
- CASTOR GSI (CGSI) plug-in for gSOAP
  - Also used in GFAL
- Evolution @ CERN:
  - First prototype in summer 2003
  - First production version deployed in December 2003
- Other sites having deployed the CASTOR SRM
  - CNAF (INFN/Bologna)
  - PIC (Barcelona)

CASTOR SRM v1.1



Problems found

- The interoperability problems can be classified as:
  - Due to problems with the SRM specification
  - Due to assumptions in SRM or SOAP implementations
  - Due to GSI incompatibilities
- The debugging of GSI incompatibilities is by far the most difficult and time consuming

Final Thoughts

- I personally found it very interesting – so that's what a Storage Tank is. And I now know what's the difference between SRB and SRM.
- I suspect that LCG team will be satisfied that they will move forward with their data challenges this year with more certainty than before and the Tier 1 sites now understand better what role they can and must play
- Encouraging to see the various sites, LCG and non-LCG, participating and interacting positively and agreeing how to move forward
- Proposed theme for the Large System SIG day at the next HEPiX is Technology
  - Is there a role for MacOS?
  - Is Itanium suitable for HEP?
  - Xeon or Opteron?
  - 32 or 64 bit?
- Don't forget to register for CHEP (www.chep2004.org), early registration deadline is 25th June