Data Access and Storage Technology – Any Chance to Meet Future Requirements ?

Michael Ernst

Deutsches Elektronen - Synchrotron DESY, Notkestrasse 85,
22607 Hamburg, Germany

The data storage and management requirements of future HEP - experiments will greatly exceed those of current experiments. Multi petabyte of new data per year and per experiment is expected. This paper describes current trends in data storage technology, system architecture and data management.

1 Introduction

In HEP we realize the fact that current and especially future experiments are growing in terms of collaborators and complexity, while the number of concurrent experiments is going down. The driving force for Collaboration is the enormous cost (accelerator, detector, running cost, etc.) of the experimental equipment and by the wealth of physics topics that may be studied in the data acquired by one detector. So, former continent wide collaboration will be extended to worldwide collaborations by the end of this millenium (already exists to some extent for the HERA collaborations). The HERA collaborations H1 and ZEUS are currently formed by 400 physicists each. Faculty and students are often driven to spend long periods, perhaps years, at the accelerator laboratory simply to work on data analysis. It is crucial for current and especially future experiments that we use all possible means to allow university based collaborators to participate without having to relocate or spend 90% of the physic department's budget for travelling.

Though it is not the declared topic of this paper, I can't resist pointing at those who have an extremely negative influence on the health of worldwide collaborations. True especially for Europe, the PTT tariff structure for international lines doesn't allow us to open appropriate communication channels. A physicist working actively in the analysis process needs approximately 1% of his experiment's computing power, data and data flow.

Current practise for physicists in universities is they are having X-Stations and/or 100 MIPS workstations. But they can forget about bringing 1% of the interesting data to the desktop. Common sense appears to dictate a highly centralized model where more than 90% of both data and compute power for one experiment sit at the accelerator lab to which the worldwide collaboration has access via X-Windows. This model has only a chance, because a major step in preparing the data for the final analysis is data reduction to a very large extent. Fortunately, the result is pretty compact; hence, the amount of data to be transferred via expensive links relatively small.

Everybody goes distributed today. Because of the reasons mentioned above, decentralized data repositories are unrealistic. The positive side effect is, no distributed data management is needed. However, the schizophrenia gets into the business, because the funding agencies reluctantly accept that they have to pay for costly detector components lying deep underground in a foreign country. They rarely accept that a significant part of their budget spent on computing should be invested in a foreign country also.

Though physics related questions have moved to different topics over time the overall data charateristics stays the same. Events are analyzed in any order, in many cases in parallel. Historically, HEP data sets have been orders of magnitude too large for affordable random access storage, so the HEP analysis principle is still oriented towards large, sequential files stored on magnetic tapes.

However, at the event level, access to HEP data is intrinsically random; even within events, particular studies may make very selective use of data. The ideal data storage system for a year 2000+ experiment would offer random access to multi 100 terabytes of structured data selected from a multi petabyte tape store, with the granularity going down to to the level of a few bytes. The bandwidth required by analysis processes will be in the order of 100 MBytes/s at minimal latency.

2 Current Technology Trends

Getting now to the issues around the Mass Storage arena dealing with Storage Technology, Storage Management and Network Attached Storage Devices. Important trends we currently observe include:

- Declining drive and media cost
- Very high performance interfaces
- Increased functionality (optical, mixed media libraries)
- Advances in drive and media technology (increased recording densities, also resulting in increased bandwidth)
- Advances in Storage Management Software (providing access to all data stored in the system (incl. offline), automatic hot and cold migration is optimizing price / performance trade-offs.
- Standardization

The importance of rapid and straight development for Mass Storage products has been acknowledged by the US-Government and the industry. A program (5 years, 61 MUS$) has been launched in order to do investigations on Ultra-High Density Recording. Sixteen companies and sixteen universities collaborate. Key topics include the goal to achieve 10 Gbits/in.^{2} on magnetic and optical disk, 1 TByte/in.^{3} on magnetic tape and 2 TByte/in.^{3} on optical tape, and extremely fast random access times in the order of <1 ms on optical disks, respectively. Already widely agreed, 10 Gbits/in.^{2} can be achieved within the next couple of years. Experts start talking about the next step towards 100 Gbits/in.^{2} Because of the necessary reduction concerning the bit size, problems with the media grain size, affecting the Signal-to-Noise-Ratio (SNR) have to be tackled. Also media quality degradation over time needs to be solved. Another solution may be to use structured media in which the bit cells are defined by lithographic cells in the recording media. If they are defined, then each bit can be stored on a single particle (instead requiring 1000 grains/bit, 1 grain/bit would be adequate). In this case recording densities as high as 10 Tbits/in.^{2} would theoretically be thermally stable with today's materials.

Whatever form of recording media is utilized, it is likely that some form of near-field probe head (magnetic or optical) will be required to record and playback the data. It's likely that head-to-media spacing will go down to approx. 10 nm, meaning that the head has to run in contact or in close proximity with the media. Experts believe that 10 Gbits/in.^{2} are achievable with longitudinal recording; however, densities in the order of 100 Gbits/in.^{2} require some changes in approach.

In conclusion, the potential derived from scientific studies look pretty optimistic. However, our burning question concerning the physical and logical organization of HEP data repositories is untouched, so far.

3 Organization of HEP Data Repositories

In the HEP community, the diversity of requirements has continually reinforced the need for an economical storage hierarchy. Continuing advances in technology have enabled the development of new devices with enhanced capabilities. The acceptance of new devices is tempered, however, by the investment most users have in existing tape storage volumes. For removable tape storage systems, this therefore presents a dilemma. A 10X rather than incremental improvement in storage cost-performance assessment must be offered to make the conversion to a new system attractive (unless there is no other technical alternative which in turn implies a lot of compatibility issues. Compromises and intermediate solutions are very likely to happen).

3.1 Storage Hierarchy

Though a lot more complicated, the necessity of having a storage hierarchy will persist over a long period of time.

- Magnetic tape storage will remain significantly less expensive than hard disk storage. While lower cost per storage will be a continuing trend for all technologies, it is expected that the cost ratio will remain intact.

- The volumetric density of tape storage relative to other storage technologies will, for fundamental reasons, always be a large ratio.

- Although significant advances in electronic data transfer communication networks can be expected in the next decade, because of bandwidth limitations and telecom costs, data interchange via physical transport of removable media will remain the most economical procedure.

But there is more hardware-oriented issues besides the pure tape drive technology. With the rapid growth in processing power, significant expansion at (local area -) high performance networking, and the increased complexity of application data sets, the requirement for high performance, large capacity, reliable and secure, and most of all affordable robotic tape storage libraries has greatly increased. With today's open system compute servers the need for reliable secure MSS has taken an ever increasing importance to our processing center's ability to satisfy operational requirements. Depending on various parameters, library support for mixed media, especially in a Hierarchical Storage Environment, is getting to a high importance level.

The End User Parameters - Capacity, Data Rate and Reliability - need no elaboration. Access time to data and drive utilization, however, have importance beyond the obvious. These attributes serve as key elements in optimizing the price-performance of the complete storage subsystem, i.e. automation plus storage devices plus media. Though different weighting factors need to be applied, in the final analysis, a device with higher throughput (data rate, search speed, load/unload time (cartridge vs. cassette)) requires fewer devices to achieve a given function. Applications such as 'digital libraries' and network-attached HSM servers will be the major benificiaries of such characteristics. There is, however, one important point which needs to be emphasized. Going along with much higher transfer rates of today's most advanced drives, there is a clear tendency towards increased media capacity, probably keeping the same mechanical form factor at the same time (like 3490 and NTP). This turns out to be not always benifitial at all concerning data access times. Mapping those new parameters to the 'Needle in the Haystack' - problem implies a huge contention potential due to many different processes requesting the same medium in a highly asynchronous fashion, forcing the system to mount a 50 GByte-Cartridge in order to extract a few MBytes, and then, due to the small number of drives, dismount the cartridge in order to serve the next request.

3.2 Topology

After having described recent and future storage technology its time to start discussing Topology and Data Access issues.

Assuming that we accept the centralized computing and data repository model described in the introduction, we are facing a situation where we are *not* using a monolithic machine delivering all required CPU-cycles and having connected all the mass storage devices forming the central data repository. Reality today is a highly distributed system combined out of various functional blocks:

- CPU Servers
- Data Servers (Tape and Disk)
- Backplane (Network)

3.3 CPU Servers

The CPU Server's main purpose is to deliver CPU-cycles to the applications, however, today we very often see a profile where a significant part of cycles is spent on Data Management, Data Access to Mass Storage Devices (Disk, probably Tape) and -not to forget- Networking. The reason for that is the fact that a central service provider in a lab always looks for the best buy in respect to current requirements. When, for example, H1 decided for an SGI Challenge XL machine, the decision was mainly driven by the CPU requirements and a data capacity demand for random access of roughly 500 GByte. Mixing old low with high capacity disk drive technology led to a utilization of 30 SCSI channels. From the hardware's point of view the Challenge XL can perfectly handle that, however, we hit the limit. Also, depending on the access method used, e.g. whether using the full file system including the buffer cache, or using the file system but bypassing the buffer cache, or using raw devices, is making an enormous difference concerning CPU utilization for Data Access. According to our experience, usage of the full file system including buffer cache is causing a 25% load on the machine, just leaving less than 3/4 of the CPU for the application. The overhead goes down to 5 - 10% when just bypassing the buffer cache, an option offered by SGI through 'Direct I/O'.

3.4 The Backplane

For a number of reasons, including efficiency and scalability, one should consider dedicated servers, reserved to run batch jobs or deliver data to them. This is not new at all, since it was proposed by CERN in the SHIFT project long time ago. However, the critical point in the past was always the Backplane, interconnecting CPU- and Data Servers. We discovered very early that standard network technology, like FDDI, wasn't sufficient (small packets = large protocol processing overhead, limited bandwidth of $<$ 10 MBytes/s) and that only special purpose hardware, like Ultranet, came close to what we needed. Though we would have been in bad shape without Ultranet, we were facing a lot of compatibility and reliability issues, unavoidable for a small company dealing with 5 to 10 computer vendor's underwear (I/O subsystems). Also, the internal limits of Ultranet ($<$ 32 concurrent streams) forced us to look for alternatives, preferably standard network technology supported by the vendors envolved in the computer and data storage business. Just to mention their names, there is HIPPI (parallel 32/64 bit channel architecture, 100/200 MByte/s, a *very* pragmatic approach, believed to have a limited lifetime), ATM and FCS providing for a variety of speeds (ranging from 155 to gigabits/s). Common to them is they were or are going to be approved by standard committees, like ANSI. Not all of them are available on any of our favourite computing platforms today, however, especially true for ATM, the number of vendor supported interfaces including drivers, high level APIs and IP is growing daily.

3.5 Disk and Tape Servers

When thinking of remote Data Access to Disk based Filesystems, the Network File System (NFS) comes to many people's mind. No question, this is a very convenient solution; on the other hand, current practise proves that raw data rates of 5 to 7 MBytes/s per Disk Drive decreases to 1 MByte/s by the time the data has traversed the protocol stack labyrinth (both disk access and network related).

Hence, network based transport protocol alternatives, based upon TCP/IP sockets

were chosen in order to avoid performance penalties. Still the remote file systems are NFS-mounted, however, this path is just used for administrative purposes, strictly not for data transfer. A package offering this kind of functionality is called RFIO, also part of the SHIFT software.
There is a very important detail concerning the backplane. As the backplane in not necessarily homogeneous, meaning it's *not* made out of a single network technology, like FDDI, HIPPI, ATM or FCS, a method had to be selected allowing to transparently forward data through different technology approaches. The answer was, as mentioned already, the Internet Protocol Stack IP. The described architecture was implemented for the ZEUS experiment, including a Data Server, based on a SGI Challenge DM with a HIPPI interface. The NetStar GigaRouter [1], a Mixed Media Router, transparently forwarding IP packets between HIPPI, FDDI and ATM, forms the backplane. Sustained data rates of more than 30 MBytes/s (multiple streams) into the ZEUS batch machines are deliverd at a cost of 30% of the available CPU on the Challenge DM. Again, the DM is nothing but an 'I/O Crossbar' between a large disk farm (> 200 Disk Drives) and the network. Similar approaches were taken in order to achieve Network Attached Tape Devices. We choose FDDI attachment for our STK 3490s by means of a RS/6000 based 'Controller', because the limited data transfer rate of 3 MBytes/s makes FDDI perfectly suitable. On the other hand, connecting a number of high performance D2 or D3 helical tape drives to an 'I/O Crossbar', HIPPI attachment gives a good balance on the network side.

4 Data Mangement and Data Access

Briefly summarizing what we've discussed so far is that we basically know how to physically store data, to run 'Dataless' CPU Servers and let dedicated I/O Crossbars (based on suitable workstation class machines) take care to deliver the required data through the network as efficient as possible. However, the most complicated and most complex topics are still untouched. We don't know yet how to manage petabytes of data stored on the various different storage devices being produced from a variety of different storage technologies.
There is obviously a need to find a Data Model describing Storage, Access, and Analysis Methods which meet present requirements for HEP data and which is scalable to the 10^4 increases required by the time LHC experiments get online; a requirement which is not HEP-specific at all. Results from investigations made clear the simple and powerful relational dabase model is not appropriate in many scientific areas. Extensions to the model have resulted in the object-oriented data model. It should have notions, including objects, attributes, methods, inheritance, collections and versioning.

4.1 Concept for a HEP Event Store using a Database

The HEP computer user sees the computing system primarily via a workstation as a Data (object and/or event) Store. The workstation is a client on a local LAN (the principle works of course for WAN connections also, but its unrealistic to be

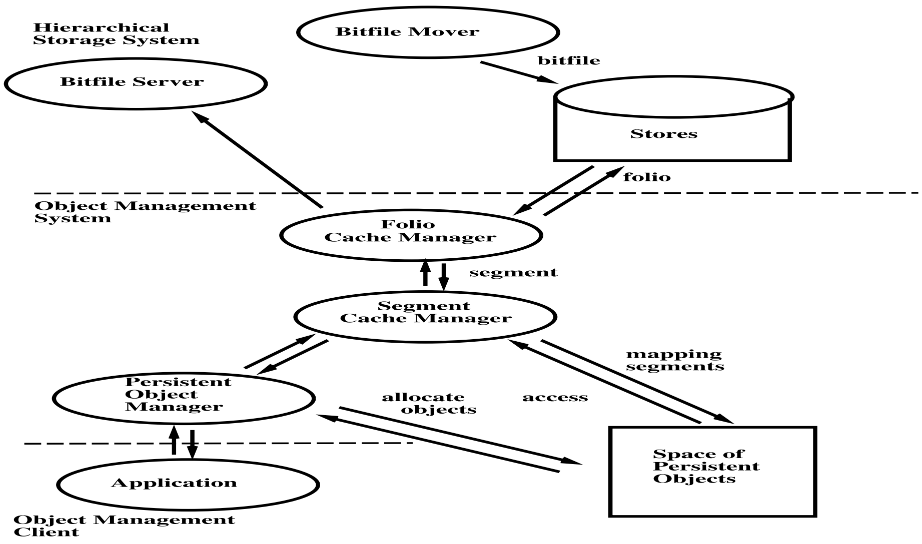

Figure 1: Proposal for a Persistent Object Store

proposed because of the current weakness of the WAN infrastructure.

Queries generated by the workstation are examined by the local DBMS to see if they can be satisfied locally. Those which can not are passed to the meta database. The results (event/object) are made available on the local file systems as cached data for the workstation to analyze. Queries which can not be satisfied by the local data based caches or the metadata caches are analyzed and optimized for transmission to the main event/object store. This would be a set of shared memory computers, equipped with multi 100 Gbyte of magnetic disk (for storing methods, objects and events), 100 TByte of automated tape libraries and interconnected via a high speed LAN to the meta database and the workstation. The incoming queries from the workstation users will trigger stored methods for the events and/or objects in the I/O clusters which are expected to reduce the returned events and/or objects, which satisfy the user queries such that the outgoing data flow is consistent with high speed network capacity.

This type of architecture demands a number of requirements:

- **A Data Base Management System** which works well in a client-server hierarchical mass storage system (e.g. IEEE Mass Storage System Reference

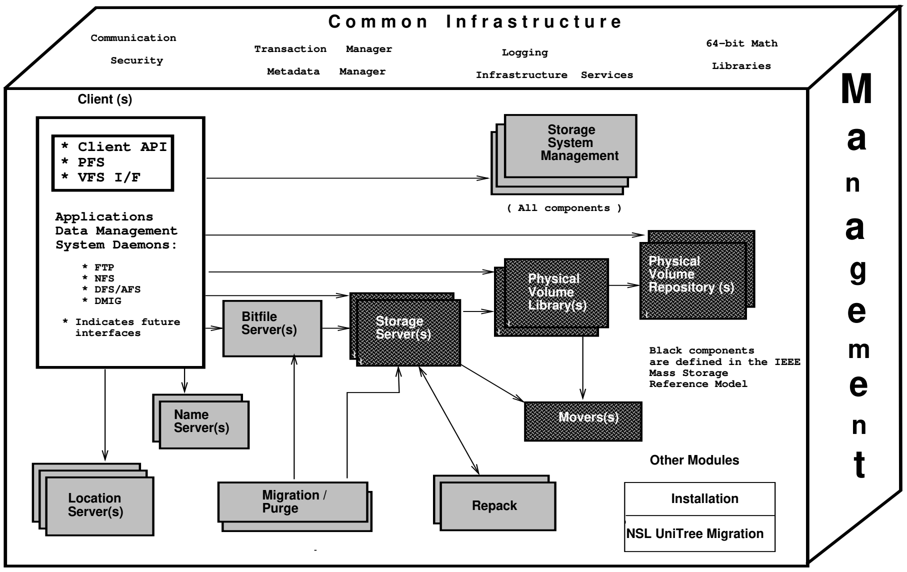

Figure 2: HPSS Software Architecture

Model)

- **I/O Subsystem** which provides high capacity, low latency and high throughput tape robots
- **A Data Model** that exploits the inherent parallelism of HEP Data which will allow a linear speed-up in both computing and data access
- **A Query Language** which refers naturally to physics objects and can express complex and numerically intensive queries

5 Hierarchical Mass Storage System

Taking an example of a real Hierarchical Mass Storage System I am going to describe an implementation following the IEEE Mass Storage System Reference Model.

5.1 High Performance Storage System (HPSS)

The HPSS [2] architecure is based on the IEEE Mass Storage Reference Model, version 5 and is network-centred, including a high speed network for data transfer and a

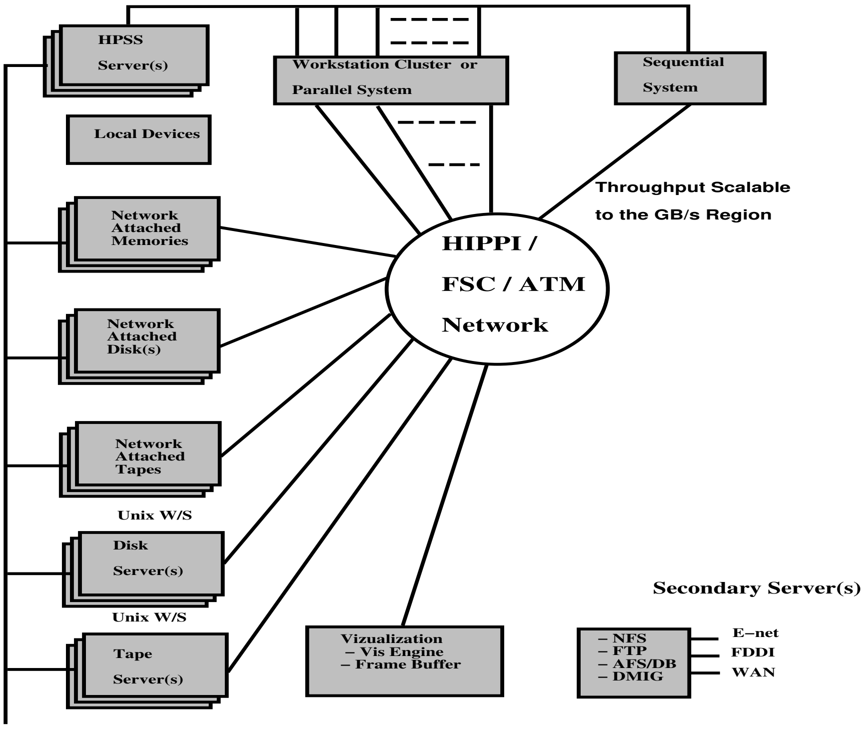

Figure 3: HPSS Configuration Example

separate network for control. The control network uses the Open Software Foundation's (OSF) Distributed Computing Environment (DCE) Remote Procedure Calls. An important feature of HPSS is its support for both parallel and sequential I/O and standard communication interfaces between processors and storage devices.

Clients direct a request for data to an HPSS Server, which in turn directs network-attached storage devices or servers to transfer data directly, sequentially or in parallel to the client node(s) through the high speed data network. TCP/IP sockets and IPI-3 over HIPPI are being utilized today; Fibre Channel Standard (FCS) with IPI-3 or SCSI, or Asynchronous Transfer Mode (ATM) will also be supported in the future. Through its parallel storage support by data striping, HPSS will continue to scale upward as additional storage devices and controllers are added to a site installation. HPSS also supports high-level interfaces, currently Client API, FTP (standard and parallel) and NFS.

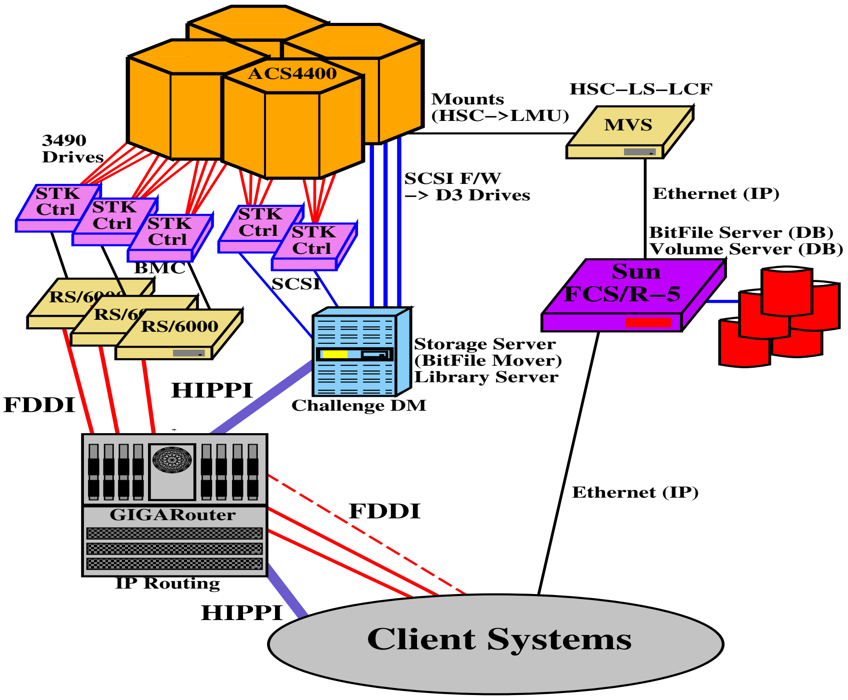

Figure 4: OSM based Storage Architecture at DESY

The design also includes a Migration Server, which provides Hierarchical Storage Management through Migration and Caching between devices. There are two types of migration and caching; disk migration and caching and tape migration and caching. Multiple Storage Hierarchies are supported. Data is cached to the highest level (fastest) device in a given hierarchy when accessed and migrated when inactive and space is required.

There are more existing implementations following the IEEE Mass Storage Reference Model: Legent's Open Storage Manager (OSM), EMASS's AMASS/DataMgr and NSL UniTree. They are very similar to HPSS in many respects.
I am sure that there are more products in the same area , both existing and on the horizon. I apologize for not having them mentioned here.

6 Summary

There are good reasons to believe the computer industry will provide cost effective I/O Subsystems for realizing the mass storage hardware requirements. The HEP community encourages the development of the IEEE Mass Storage Reference Model for the integration of these I/O Subsystems with a network based computing model. A number of vendors have picked up the model already in order to build real systems. One of the problems left is efficient Data Access. Proposals, like the one made for a 'Persistent Object Store' by Robert Grossman (University Illinois) [3] really map perfectly to HEP data access needs. This is certainly only one of many promising examples, and the Data Access topic is clearly the one which needs the community's full attention.

References

1. Michael Ernst in *Using High Performance Interconnects in a Distributed Computing and Storage Environment*, (Computing in High-Energy Physics, 1994).
2. Danny Teaff in *The Architecture of the High Performance Storage System (HPSS)*, (Fourth NASA Goddard Conference on Mass Storage Systems and Technologies, 1995).
3. R.L. Grossman in *The Design and Evaluation of a High Performance Object Store*, (University of Illinois, Chicago, 1994).