Disclaimer

This note has not been internally reviewed by the DØ Collaboration. Results or plots contained in this note were only intended for internal documentation by the authors of the note and they are not approved as scientific results by either the authors or the DØ Collaboration. All approved scientific results of the DØ Collaboration have been published as internally reviewed Conference Notes or in peer reviewed journals.

DØ Note 2578

Managing and Serving a Multi-Terabyte Data Set at the Fermilab DØ Experiment

by Lee Lueking
for the DØ Collaboration
Fermilab
Batavia, Illinois, USA

Abstract

The DØ experiment at Fermilab is accumulating data from the electronic detection of collisions between protons and anti-protons. This presentation will describe the data structure, data cataloging and serving of the Multi-Terabyte data set to a user community. The current data consists of over 85 TBytes stored in a hierarchy of data sets with various latencies and frequencies of use. The primary data storage is on some 40,000 8mm tapes while the most frequently used data is on nearly 300 GBytes of SCSI disks. Data is served to VMS and UNIX analysis clusters over an FDDI network from a centralized File Server. Also described will be our plans for handling a future data set anticipated to be an order of magnitude larger. Some of the ideas being considered are alternative data structures, parallel disk access, automated tape libraries, and centralized analysis servers.

1. Introduction

The DØ experiment at Fermilab is accumulating data from the electronic detection of particles generated in collisions between energetic protons and anti-protons. The experiment consists of a large detector with over 100,000 channels of electronic readout triggered at a rate of 200KHz. Each trigger or "event" represents the collision of a proton and anti-proton and only a small fraction of them produce a pattern in the detector showing "signatures" that are useful for further study. An on-line filtering system selects these events and reduces the amount of recorded data to 2-3 events per second. These events are written to 8mm tape with a data rate of 1.5 to 2 MBps.

This data is subsequently processed to reconstruct the information in each event into particle tracks and energy deposits that are interpreted as particle "objects". These objects are measured and combined in subsequent analysis steps to study and search for interesting phenomena. The amount of trigger data that has been accumulated during the current data taking period is over 23 TB and will increase to over 40 TB by the end of this year. The amount of data that has been produced in the reconstruction stage is an additional 20 TB and will be over 30 TB by the end of the year.

In addition to the data produced by the on-line system, some data is produced by Monte Carlo simulations. This data is also processed through the same channels as the detector data. Although this represents a small fraction of the total data needed for the experiment, it is still a vital part of the information needed to produce experimental results.

A central file server system consisting of SCSI disk and tape drives mounted on a VMS cluster was used to manage and serve this data effectively. This arrangement, networked to a set of analysis clusters through an FDDI ring, has provided high performance at low cost. There are however several limitations in the system and, as we contemplate a future data set which is much larger, we need to explore other options.

2. User community and data access requirements

In order to understand the access patterns to the data, it is important to understand the user community and the data model. The user community consists of 400 physicists located locally and at various institutions around the world. These researchers are divided into groups, each with interests in particular portions or aspects of the data. Much of the data processing and serving is performed on workstation clusters located at Fermilab connected via FDDI rings, though some analysis is performed remotely at various universities and laboratories with network connections to Fermilab.

2.1 The event structure

The basic unit of data is the event. Each event has a tree-like structure provided by using a data management system called ZEBRA[1]. This package provides convenient input and output tools that utilize a file structure called FZ giving sequential access to the data. This structure enables linking and cross-linking various categories of information within the event to each other. For example, an event might have a set of muons with a set of variables that characterize them. Each of these muons is linked to a set of hits from which they were formed. The event will also have electrons with the variables that describe them, linked to clusters of hits in calorimeter channels. Furthermore, many of the objects in the event, e.g., muons, electrons, jets, etc., may be linked to a common point of origin in the detector, where the collision occurred.

In order to provide the most efficient access to the data, it is important to understand how events are used. There are several types of data access that require varied approaches. Some analyses involve searching for a small set of events with particularly interesting signatures, for example looking for a Top quark. A second type of data usage requires examining a few characteristics of each event in a large portion of the entire set. A third category may require detailed study of certain detector qualities for small portions of the data, to calibrate or better understand how the detecting apparatus is working. Each of these types of analyses has specific needs in terms of the number of events involved and the amount of information required.

A good example of the first type of access is a new particle search that might proceed with the following steps:

1. Examine a small amount of information from each event in the entire data set to select interesting events.
2. Select additional information for this selected set of interesting events and make tighter requirements to reduce the sample further.
3. Collect all information possible for this select set of very interesting events.
4. Reprocess the selected events to understand every aspect of each of these events.

This type of analysis can proceed very quickly, in the initial stage, if it has access to the minimum amount of information required from each event to choose the interesting events. Therefore, storing events with only the core of the information is very advantageous. However, as the analysis progresses, more detailed information is required and finally, in the final stages, the entire event information is needed.

At the other extreme is an analysis requiring somewhat more information for each event to the very final stages. An example of this kind of operation is the study of jets of energy in the data that occur in many of the events. Such an analysis proceeds with the following steps:

1. Select a large sample of events, possibly as much as 30 - 50% of the entire sample.
2. Select key elements from each event and make adjustments, based on detector performance, to carefully measure the event characteristics.
3. Study and compile measurement results based on the adjusted parameters.

These steps can be performed efficiently when the correct information is made available to the researcher in the initial stage. Since the procedure relies on a large portion of the events, if too much information for each event is provided, the process is inefficient and slow.

Unfortunately, the kinds of information needed for each kind of analysis are slightly different. Therefore, there is always a tradeoff that needs to be made; speed and efficiency of access verses the amount of information available for each event.

Table 3.1. Characteristics of the data types.

Data Type	Event size (KB)	Total Size (TB)	Storage	Access Pattern	Description
RAW	600	23	8mm Tape	Rare	Unprocessed data
STA	250-500	20	8mm Tape	Occasional	Reconstructed data
DST	50	2.2	8mm Tape	Frequent	Important information
Micro-DST	5	0.2	Online Disk	Very frequent	Most important information

3. Current data model and file serving

The goal in data serving is to provide all of the information to the users as efficiently and quickly as possible. The approach which has been pursued during the last 2 years has been to adopt the event as the basic entity and make it available in three quantized levels. These levels or data types are defined by the amount of information included with each event and are named Standard (STA), Data Summary (DST), and Micro-DST (MDST). A typical DØ event is about 600 KBytes, in its unprocessed or RAW form, which is reduced to about 250-500 KByte when it is reconstructed to STA. The STA data contains all of the information for the event. The DST is a subset of the most useful parts of the STA and the Micro-DST contains most of the information in the DST, in a compressed format. Characteristics of the various data types are summarized in Table 3.1.

The data is broken into streams to provide quick access to key categories of events. This process entails reading each event and directing it to a particular output file, based on a set of algorithms written to detect certain signatures. This kind of physical streaming is performed for both STA and DST data types and there are occasionally large overlaps between the events that are passed to various streams. For the DST data, there are 17 streams each with a size ranging from 1% to 40 % of the full DST data set. For the STA streaming operation there are 10 streams, each containing from 1 to 15 % of the total STA data set. These streams provide users very quick access to data that would otherwise be unavailable due to its size. Table 3.2 summarizes these streams and their sizes.

Efficient access to the Micro-DST data events is provided through a technique called virtual streaming which is described later.

Storage characteristics of all of the data which has been produced by the experiment are summarized in Table 3.3. The largest category of data for which analysis is performed is broken into categories such as RAW, STA, and so on. The remainder of the data is specified as "Other". This category includes data which is not described in the processing model presented in this paper, or data which has been re-processed and is not used extensively.

For most of the RAW, STA and DST unstreamed data, files contain around 500 events each. This is a convenient quantity with a size of around 300 Mbyte for the RAW data, and works well within the tape and disk constraints of the system. When data is streamed, files are merged to increase the number of streamed events stored in each file and to reduce the total number of files that need to be cataloged in the data bases.

Table 3.2 Stream descriptions and their sizes.

Name	Stream Description	% of total	STA (TB)	DST (TB)
RGE	Generally interesting events	40	-	0.80
ELF	Electron-like	27	-	0.54
QCJ	QCD Jets	23	-	0.46
HLP	High Pt Lepton	15	3.0	0.30
B1M	Single Muon B	11	-	0.22
TBG	TOP Background	10	2.0	0.20
QCR	QCD Rapidity Gap	7	1.4	0.14
MU1	Single Muon	6	1.2	0.12
QGA	QCD Gamma	4	-	0.08
TPJ	Top to Jets	5	1.0	0.10
QEM	QCD EM	3	0.6	0.06
B2M	Two Muon B	4	0.8	0.08
PI0	Pi - zero	3	-	0.06
NPA	New Phenomena	3	0.6	0.06
MIN	Minimum Bias	1	0.2	0.02
BSM	Small Angle Muons - B	2	0.4	0.04
MET	Missing Et	6	1.2	-
TAU	Tau	3	0.6	-

3.2 Overall data flow

Most of the processing and analysis work is centered at Fermilab on VMS and UNIX clusters that are connected via an FDDI ring. The central data server or, DØ File Server (D0FS), and is equipped with over half a GB of on-line disk, 32 8mm tape drives and 4 DLT tape drives. All data is cataloged both in a hierarchical and in a relational database that are used for data accesses. A summary of the entire data flow is provided in Figure 3.2.

The data produced by the on-line system is written to 8mm tapes that are then processed on a 4000 MIPS UNIX farm. The farm is composed of 74 SGI Indigo's and 24 IBM 320 nodes, which process roughly 100 GB of RAW data each day into 60 GB of STA and 12GB of DST information. The STA data is spooled directly to 8mm tape that are subsequently streamed on two SGI Crimsons each equipped with eight 8mm tape drives and 16GB of spooling disk.

The DST data is transferred via Ethernet to temporary disk storage on the file server where it is streamed and converted into Micro-DST format. The physically streamed DST data are archived on 8mm tape while the Micro-DSTs remain in permanent disk storage for frequent user access.

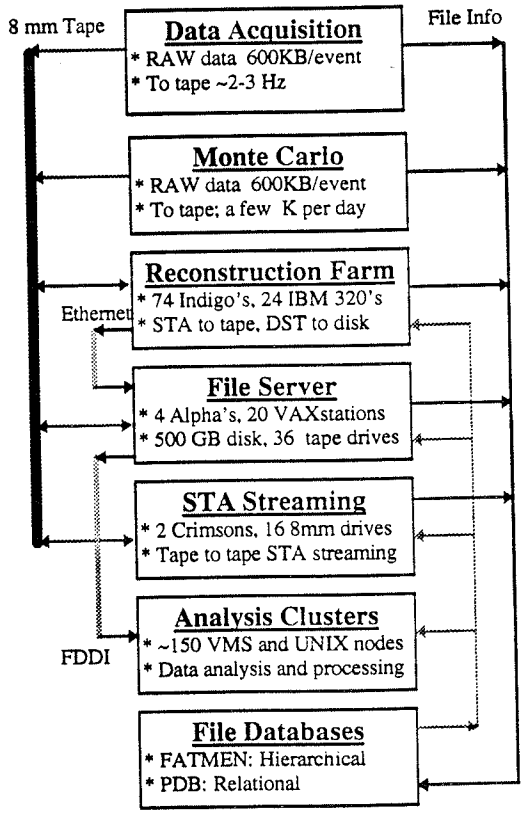


Figure 3.2. Overall data flow.

3.3 Details of data serving

3.3.1 Data cataloging. The data is completely cataloged by file and/or tape label in two data bases. The first is a hierarchical database that uses a ZEBRA structure called RZ, which provides keyed access to information in a direct access file. This is part of a complete file management system called FATMEN (File and Tape Management system) which was developed at CERN[2]. The second database is called the Production Data Base (PDB) which is a relational database using DEC RDB.

Table 3.3 Size of the total data set. The category "Other" is described in the text.

Data Type	Events (M)	Size (TB)	Files (K)	8mm Tapes (K)
RAW	46	23	119	9
STA	46	20	119	6
STA Streamed	35	6	85	4
DST	46	2	119	0.8
DST Streamed	70	3.5	92	2.1
Micro-DST	46	0.2	5.6	0.1
Other	136	31	572	21
Total	425	85	1112	43

In addition to file and tape information each of these catalogs contains entries relating to number of events, the first and last event and other processing information for each file. In addition, the PDB also contains links that enable complete tracing of all of the processing steps involved in producing any file, as well as other important information critical to the understanding of the data. The total raw and processed data files represented in these databases for the current data set exceeds 1M entries. Maintaining these large databases is an enormous task.

3.3.2 The DØ file server hardware. At the center of the data serving for the experiment is the DØ file server (D0FS). It is composed of four DEC 3000 AXP's, three VAXstation 4000-90's and 17 VAXstation 4000-60s. The

AXP's consist of three 3000-400's and one 3000-500 that are all employed as disk file servers and together have 300 GB of SCSI disks attached to them. These machines are connected together and to the analysis cluster via FDDI through a GIGA switch which supplies a peak throughput of 8 MBps. The VAXstations are employed as tape server nodes and are equipped with 130 GB disk, 31 8mm tape drives and three DLT2000 drives. The database information resides on disks connected to the 4000-90 VAXstations. There are an additional two 8mm and one DLT drives connected to the Alpha file servers for data backup and recovery. The D0FS system is summarized in Figure 3.3.

3.3.3 Project Staging. Many kinds of analysis projects involve passing over tens or hundreds of data tapes containing thousands of files. The files are staged to disk, processed and removed to allow space for additional processing. This operation has been established with some simple tools for making file lists from the FATMEN catalog, managing the input and output file staging and managing file processing while the inputs are disk resident.

The hardware for this system is a subset of the D0FS resources and consists of 8 DEC VAXstation 4000-60 machines. These nodes are equipped with 50 GB of disk and 15 8mm tape drives, of these drives 8 are located in two 116 cartridge exabyte robots and used exclusively for output. This system is capable of providing access to over 100 GB per day of data from tape. This system is used primarily for accessing DST and streamed DST and STA data. Table 3.4 shows typical access pattern over a several month period.

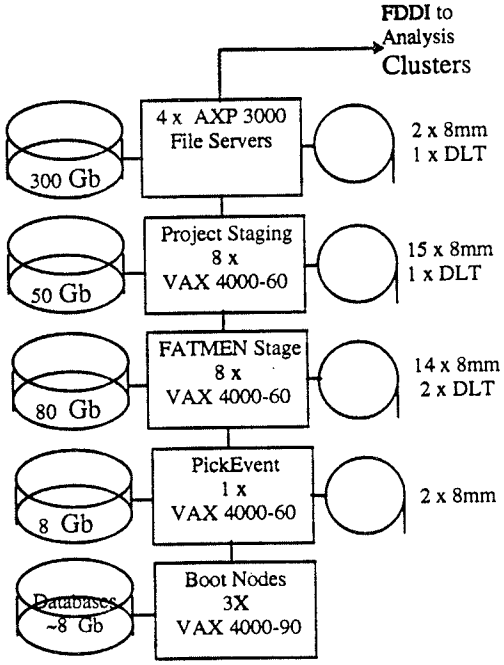


Figure 3.3 D0 File server overview.

In addition to user data, this system is employed for processing incoming DST data from the UNIX Reconstruction farm. This data amounts to typically 12 GB per day that is streamed and compressed to the more compact Micro-DSTs. These files then occupy around 2 GB per day on permanent disk storage. The DST streamed data files are copied to 8mm tape using the exabyte robots to fill tapes as data from each stream accumulates. This activity often limits the throughput of this system for project staging.

Table 3.4 Typical data access patterns for Project Staging.

Data Type	File accesses per week	Tape mounts per week
RAW	0	0
STA	0	0
Streamed STA	70	10
DST	700	50
Streamed DST	500	40

3.3.4 FATMEN staging. Access to data stored on tape can be accomplished with the utilities provided by the FATMEN package. This system allows users straightforward access to any file or files by simply supplying a list of the FATMEN catalog reference names for the desired files. This is very convenient and allows transparent file staging from tape or, if the file is already staged, the system makes direct connection to the file. On the D0FS system 14 8mm drives, two DLT drives and 80 GBs of disk storage are maintained for this staging. The disk area is maintained by a demon that removes files when they exceed a certain lifetime, or the need for more staging space is required. The processing jobs run on one of the analysis clusters and the data is transferred via FDDI from the file server disks.

This system is capable of providing access to over 200 GB per day with around 200 tape mounts. Typical access patterns for this category of staging are shown in Table 3.5 averaged over a one month period.

Table 3.5 Typical data access patterns for FATMEN Staging.

Data Type	File accesses per week	Tape mounts per week
RAW	270	50
STA	300	57
Streamed STA	230	40
DST	2580	210
Streamed DST	400	30

3.3.5 Event picking. An additional vehicle for providing quick access to data is the event picking facility. This enables users to submit a list of interesting event number identifiers that they would like selected from a large set of data, either streamed or un-streamed, RAW, STA, or DST. The tape and file location of each event are determined based upon information extracted from the PDB and FATMEN catalogs. Then the events are efficiently extracted from tapes in a dedicated operation that can provide several hundred events per day with up to 200 tape mounts. The selected events are stored on disk until the requestor has retrieved them for his/her use.

Although event picking is generally used for retrieving events from tape storage, its implementation does not restrict it to this mode of operation. If an event is desired from a file that is currently stored in on-line disk, the event will be picked up from this file. This type of operation is provided by using the information maintained in the FATMEN catalog pertaining to the location of each file. This provides particularly convenient access to events that are in on-line DSTs. The usage pattern for event picking is shown in Table 3.6.

Table 3.6 Typical data access patterns for Event Picking.

Data Type	File accesses per week	Tape mounts per week
RAW	280	70
STA	910	600
Streamed STA	0	0
DST	3	3
Streamed DST	0	0

3.3.6 Direct access to Micro-DST data. The Micro-DST data is provided through a system of virtual streaming. This system is called DØ Direct Access to Data, or D0DAD. All of the MDST files are retained on-line on D0FS on 225 GB of disk. The information containing the file name, record and byte offset for each event, as well as other important characteristics, is maintained in a file called the D0DAD catalog. Access to events associated with any of the streams shown in Table 3.2 can be made quickly by scanning the catalog and creating lists of events that have met certain criteria. It is also possible to provide even finer grained streams than those used for the streaming discussed previously.

Once the D0DAD lists are created, accessing the data is done very efficiently. Only events contained in the desired list are read from disk, and transferred over the network to be processed. This significantly reduces both the network load and the time required to process a given data set. More importantly, this approach eliminates the need to physically stream the data and thus avoids event duplication and reduces the confusion inherent in that type of operation.

The D0DAD catalog itself requires 24 bytes per event and, together with various stream lists, are currently several GB in size. This information is stored on D0FS disk and heavily accessed from the analysis clusters. It is extremely useful and is being expanded to include additional trigger information characterizing each event, which will increase its size. This catalog provides the quickest access to the data available at the experiment.

3.4 Major difficulties with the approach

The hardware system described above has evolved from a somewhat more modest set of equipment with many upgrades over the last two and a half years. It has performed well and continues to serve our needs however, as the amount of data continues to grow, several problems have manifested themselves. The large number of 8mm tape drives and thousands of tapes have been a constant struggle; much of our effort is spent tracking and recovering from problems related to them. Nearly all of the tape mounting is performed manually by computer center operators and this is a task that has become immense with mount requests every 2 to 3 minutes during peak operating periods. Another limitation with the system is the speed of the network between the file server and the analysis clusters. Although the FDDI connection is extremely reliable and

provides a high data throughput, it is frequently insufficient to match the demand.

The strategies of providing three levels of data types and streaming each of them provide efficient access to the data, but have several drawbacks. The exact definition of the data types and the streaming specifications required a great deal of effort and deliberation. Once in place, the system is inflexible and always prone to mistakes that are difficult to diagnose. In addition, this approach involves a large amount of duplication of the data which means many files and tapes that places a large burden on the file catalogs.

4. Future needs and ideas

Our experience with serving data in the fashion described above has led us to rethink the data structure and access modes. Many important modifications are required to make the current data sets more readily accessed. These changes are critical as we contemplate handling a data set which will be an order of magnitude larger, as anticipated in the physics run scheduled to begin in 1998.

There are three primary avenues that we plan to pursue in order to improve the data access; 1. Centralized computing using Client-server approaches, 2. Increase data reading speeds using parallel disk access and 3. Improved efficiency by reading only needed data words. The first involves reducing the load on the network by moving the processing to the data. Next, the rate at which one can read in data is limited by disk reading speeds and providing access to many disks in parallel reduces this limitation. Finally, the current system has limitations because it requires the entire event be in memory even if only a single byte of information is required, this problem will be addressed. Because we anticipate much more data in our next run, another important consideration is that the solutions that we develop be scaleable to also meet those needs.

In an effort to explore these approaches, we have established two projects to study and develop methods for handling our data, which we plan to scale up for the larger future sets. The first approach employs the CERN package PAW and PIAF server with Column-wise N-tuples on an SGI Challenge L . The second approach is called The Computing for Analysis Project (CAP) which divides the data into objects that are stored in efficiently accessible pieces as needed for analysis. It provides quick access to this data by parallel disk reading provided by an IBM SP2 high speed switch. This switch connects an array of I/O nodes on which the storage is attached to an array of compute nodes for processing. This system is further enhanced by connection through a high speed link to an automated tape library. The performance and capabilities of these two systems are somewhat different and we believe they will be quite complementary in providing our needs.

4.1 PAW/PIAF

PAW (Physics Analysis Workstation) is a package that conveniently enables one to perform complex analysis and produce plots or tables with various formats from data contained in a special format called n-tuples. Work at CERN has recently provided a system called PIAF (Parallel Interactive Analysis Facility), which is a powerful way of running PAW with many features important for working with large data sets [2]. The system enables a user to employ a Client-Server approach to start a remote server on a central processor platform and issue commands for data analysis. This server then divides the task among several processes that provide parallel disk access and parallel CPU usage, if multiple processors are present. Figure 4.1 illustrates how the processes work in the PAW/PIAF configuration.

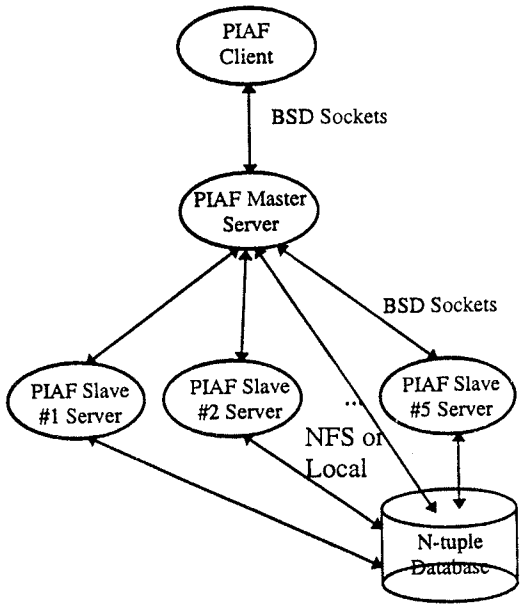


Figure 4.1. PAW/PIAF Software Configuration.

One feature that makes this system extremely efficient for certain kinds of data analysis is the use of Column-wise N-tuples. In essence, an N-tuple is a table with rows and columns in which the data is stored. In our case, the rows are events, and the columns variables pertaining to each event. In the column wise N-tuple, the data is stored in a model optimized for fast access, column by column. An illustration of this type of data storage is presented in Figure 4.2. Information arrangement of this type provides very efficient access for many of the applications that we have because only a small part of each event is usually needed.

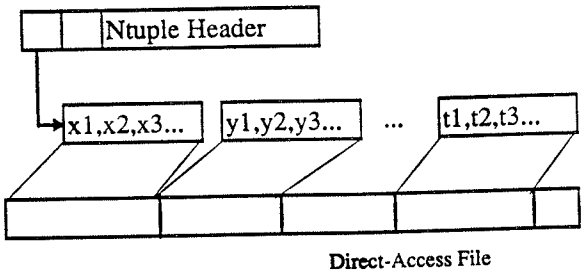


Figure 4.2 Arrangement of data in a Column-wise N-tuple.

There are other features that make this approach attractive. For sessions in which all of the needed data is stored in memory, analysis manipulations are extremely fast since the system keeps track of what has been loaded and retains it after the initial read. PAW also enables the use of complex analysis procedures written in FORTRAN or C that are dynamically linked and attached to the process. In addition, PAW is a system that is already familiar to many High Energy Physics users.

The current PAW platform is a SGI CHALLENGE L equipped with 4 R4400MC CPUs running at 150MHz with 512 MB of shared memory. This system is capable of delivering more than 320 MBps to the CPUs from 160 GB of fast and wide SCSI disks. The processors can communicate with each other with a bandwidth of 1.2 GB/s. This system was selected based on its high performance to cost ratio and can be expanded to have 4 additional processors and up to 6 GB of memory. The disk storage is expandable to over 2 GB.

Data is provided by supplying an N-tuple that contains the most important information from each event. The definition of this set is determined by the physics users and the N-tuple generated from processed data on tape. This process needs to be repeated each time it is determined that a new variable is needed. This is one of the awkward features of this approach, since it is not feasible to store all of the information for all events in Column-wise N-tuples.

Preliminary performance information from the work done so far on this system is very promising. Access rates of 35 thousand events per second are not uncommon, although actual performance varies widely depending the amount of data being extracted for each event and how the procedure is set up. With the experience gained so far the system is providing very efficient access to data and there are several improvements which should make the performance even more dramatic.

4.2 Fermilab Computing for Analysis Project

The CAP project is aimed at providing improved platforms and software for doing HEP experiment analysis in the post-event-reconstruction stages. The current focus is on "data mining" applications that must rapidly scan through vast quantities of physics data. The general goals of the project are to: 1. Provide the experiments with quick and reliable access to large amounts of data, 2. Provide parallel access to the data so as to deliver high I/O bandwidth for the data mining task, 3. Provide a structured access method for accessing the large volume of data and 4. Provide sufficient computing power for a large number of physicists to perform analysis tasks on the data simultaneously.

This approach relies on the fact that the data in each event can be broken up into classes, or categories of information. These classes are, for example, specific types of particles and all of the variables that characterize them. The data is imported into the system by splitting the information from each event in the large data set into these class "stores". The stores are then organized in a robotically mounted tape library for efficient retrieval when needed. The most frequently accessed set of stores, called the "cardinal class", are maintained on disks that are distributed on a set of I/O nodes. The other less frequently accessed tape resident stores are called "auxiliary classes".

The I/O nodes are connected to compute nodes through some high speed network that provides parallel disk access. If the information in the cardinal class is selected correctly, most of the

initial stages of data filtering can be performed without the need for additional information that requires tape access to recover. Once the sample is reduced to a small fraction of the total event sample, additional information can be retrieved from auxiliary stores for more detailed study.

The CAP testbed system currently consists of 16 IBM SP-2 and 8 SP-1 nodes. The configuration has 128 GBs of disk space on 8 designated "I/O nodes", and is attached to the CAP testbed HSM system. It is capable of delivering disk-based data across the high-performance TB2 switch, to jobs running on any of these nodes. The CAP testbed system can access a large repository of robotically accessible tape data controlled by NSL-Unitree storage management software and connected to SP nodes via Ultranet adapters and an Ultranet hub. The CAP hardware configuration is shown in Figure 4.4.

4.3 Data Storage and Cataloging

Although the cost and feasibility of vast quantities of on-line disk storage continue to improve, it is important to have fast and reliable access to off-line media also. We have been looking at alternatives to fill our needs for magnetic tape storage. One attractive media that has been explored is DEC DLT. The drive tested so far is the DLT2000 which provides storage of 10 GB uncompressed data per cartridge. With our DST data, we are able to write 14 GB compressed data to each cartridge at a sustainable rate of over 1.2 MBps. We have stored nearly 1.4 TB on about 100 cartridges to date with no serious problems and this appears to be an attractive alternative to 8mm tapes. We will continue to investigate new tape technologies and these, coupled with automated tape libraries, will become an important aspect of our processing needs.

We are considering various possibilities for file cataloging in the next run. Although the amount of data may be 10 times larger than we presently have, the number of files to catalog may not be much larger than for the current data set. Currently, file sizes range from 25 to 500 MB. It may be reasonable to increase this by a factor of 3, or so, depending on the size and speed of the tape storage and the cost of staging disk. This would reduce the load on the file cataloging.

It is also likely that one kind of catalog will be employed to track the data through reconstruction, and another once it is imported into the analysis system. For example, one might use a FATMEN-like catalog to keep track of the raw and reconstructed data, and NSL-unitree once the data is imported in to the automated tape library for analysis. There are many issues which need to be resolved with regard to this problem.

5. Conclusions

We have explored the techniques employed to extract information from a multi-TB data set and examined the problems that are inherent in the approach that was established for the DØ data analysis. Accessing data on an event-by-event basis requires reading in large amounts of unused information. Making data types with selected information and streaming each type to special samples significantly improves the access however, this approach is inflexible and requires that the data be duplicated in several ways. This causes many inefficiencies in the data analysis process and places a large burden on the data cataloging facilities.

Alternative models are being explored which will provide fast and efficient access to the current data sample as well as to a much larger sample that is anticipated. These approaches involve the storage of data in a carefully arranged fashion such that only needed information is read. Parallel accessed disk storage provides a substantial improvement in performance as well. In addition, the I/O and CPU are tightly coupled avoiding the overheads involved with high-speed networks, which have become a severe bottleneck to analysis.

Two approaches are being explored and, although their performance and features are different, they appear to be complementary and we feel they will meet the needs of future analysis efforts. The PAW/PIAF approach provides very quick and efficient access to a select portion of the data. The CAP project promises to provide efficient access to the entire data set while only accessing the needed information. Both systems are scaleable and adaptable to future hardware technologies that may make data sets of even hundreds of GBs easily managed.

6. Acknowledgments

Many individuals have contributed to these efforts and to the information included in this presentation. I would like to acknowledge all of the members from the Fermilab Computing Di-

vision and from the DØ Experiment who have contributed to these projects. Special thanks go to Dorota Genser, Krzysztof Genser, Stan Krzywinski, Laura Paterno, Wayne Baisley, John Hobbs and Pushpalatha Bhat who have contributed endlessly to keeping the current system operating. I would like to thank Adam Para for all the work he has done to make the PAW/PIAF system work at DØ, and for the discussions which we have had concerning its operation. Finally, all of the people dedicated to CAP deserve special credit. I would especially like to thank Mark Galli for his help, and Mark Fischler, Stu Fuess and Seung Ahn for their documentation and discussions.

7. References

1. R.Brun, M.Goossens, J.Zoll;CERN Computer Centre Program Library Long Write-up, Q100, CERN, 1987.
2. J.Shiers; "FATMEN - Distributed File and Tape Management", CERN Program Library Q123, CERN, 1994.
3. R.Brun,O.Couet,A.Nathaniel,F.Rademakers; "Data Mining with PIAF",Proceedings of the Conference on Computing in High Energy Physics '94. pp13-22.
4. Mark Fischler; "Logical and Physical Organization of Data for CAP Data Mining",Unpublished document.

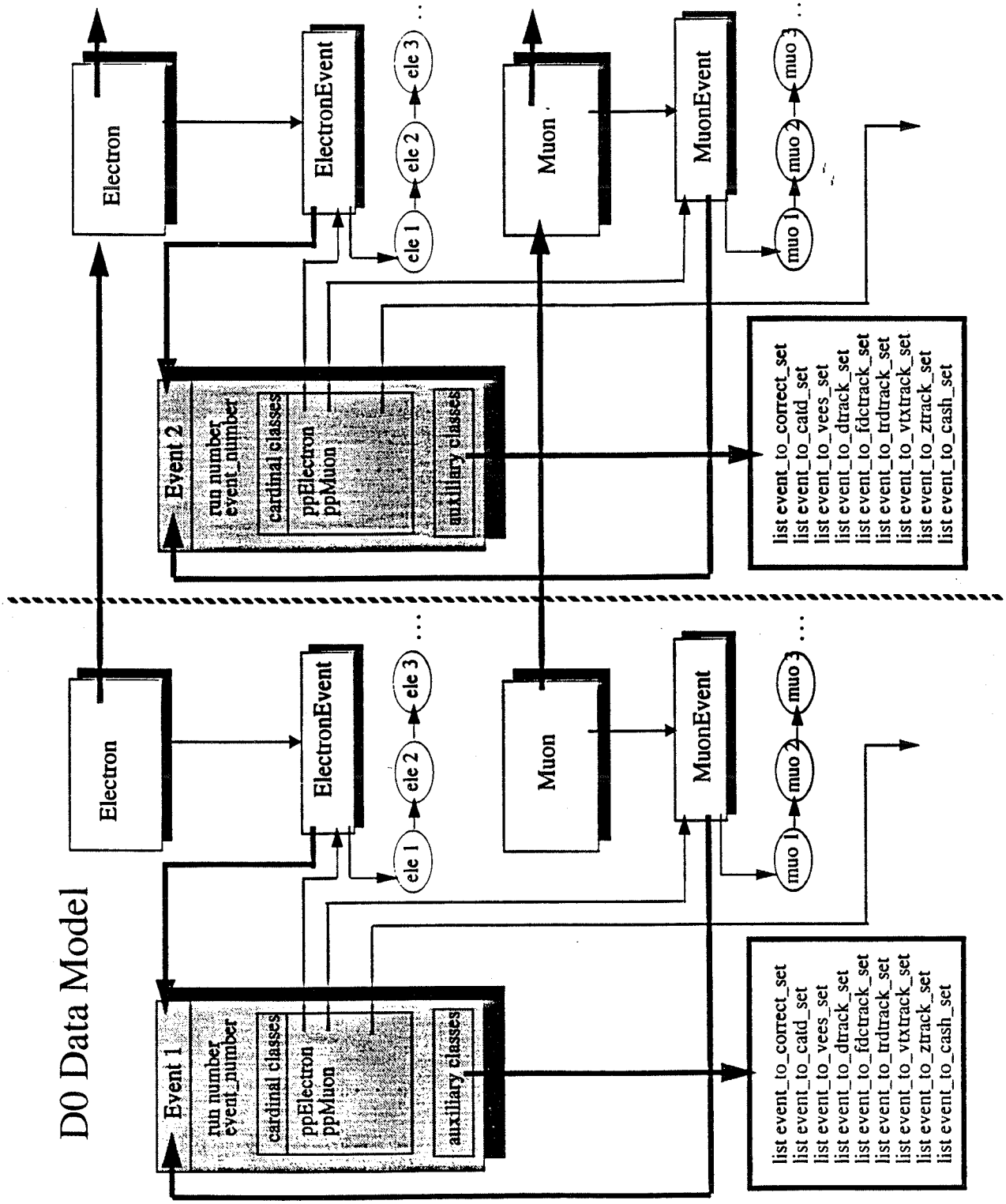
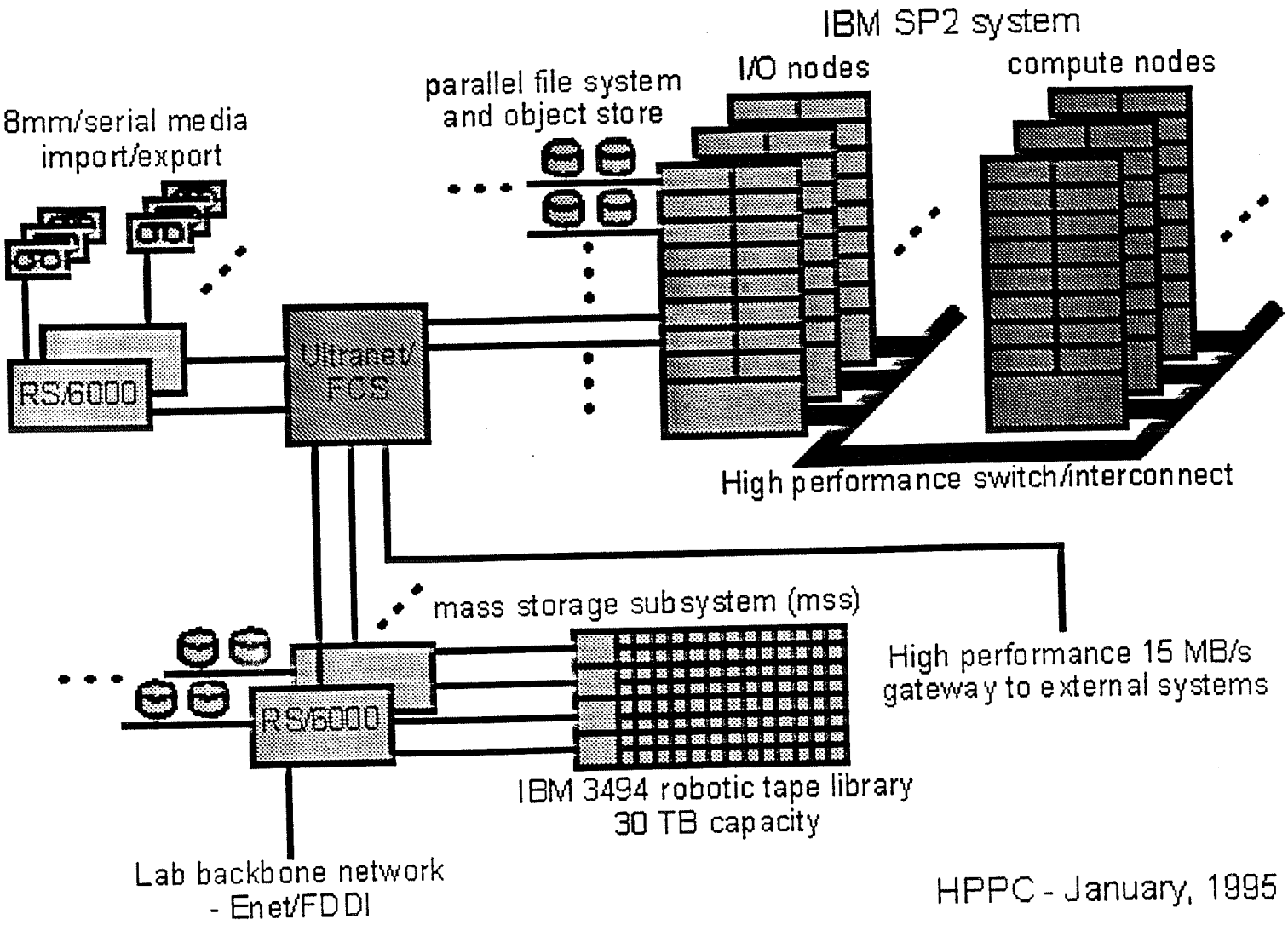


Figure 4.3 Organization of data in CAP.



4.4. Figure CAP IO/SP2/COMPUTE diagram.