The Physics Analysis Server Project

M. Bowen[1], G. Landsberg[1], and R. Partridge[1]

[1] Brown University, Department of Physics, Providence, RI 02912 USA

Abstract

We propose a new concept in handling multi-Terabyte data sets that are increasingly common in HEP. The PHysics Analysis SERver (PHASER) unifies a relational database with an inexpensive direct-access storage solution for quick access to a desired event subset. The Physics Object Database stores meta-data describing the physics objects in the events, allowing a database query to quickly identify the desired data subset. A DVD library that contains DST-level information is then used to extract complete features of the desired events. Access to the PHASER system will be through a Web interface that allows easy manipulation, handling, and querying of the data by local and remote users. We present first tests of this new concept using data acquired by the DØ experiment at Fermilab during the 1992-1996 running period.

Keywords Physics Analysis, Database, DVD, Storage, Large Data Sets

1. Introduction

The PHysics Analysis SERver (PHASER) project seeks to substantially increase the productivity of physicists analyzing the multi-Terabyte data sets that are increasingly common in HEP. Our focus is on the event selection stage, where there is a transition in data handling from monolithic reconstruction processing to the much more chaotic processing of summary data sets by large numbers of physicists creating "ntuples" for further analysis. The event selection stage is both IO and CPU intensive, requiring access to multi-Terabyte data sets and application of final calibration, particle ID, and event selection algorithms to several hundred million events. Finite computing resources often lead to an event selection bottleneck, delaying traditional physics analyses and stifling innovative approaches.

The PHASER architecture seeks to eliminate the event selection bottleneck by minimizing the IO and CPU resources required. The PHASER architecture (Fig. 1) includes:

- The Physics Object Database (POD), which stores the meta-data used by most physics analyses for their initial event selection,
- Physics Object and Particle ID tables in POD that store calibrated 4-vectors, object quality variables, and the results of particle ID algorithms for each physics object in an event, and
- Storage of the full summary (μDST) data set, along with particularly useful subsets of the larger DST and STA data sets, in DVD libraries.

The PHASER architecture has a number of features that significantly reduce the time required to make an event selection:

- New calibrations and particle ID algorithms can be quickly incorporated into POD since only the changes need to be imported, whereas regenerating the much larger μDST is a much slower and resource intensive process that is done infrequently,
- Having up-to-date versions of calibrations and particle ID algorithms stored in POD avoids the need to re-apply these algorithms for each event selection pass,
- The Particle ID tables are very small, making it possible to quickly eliminate events not having the desired set of physics objects, and

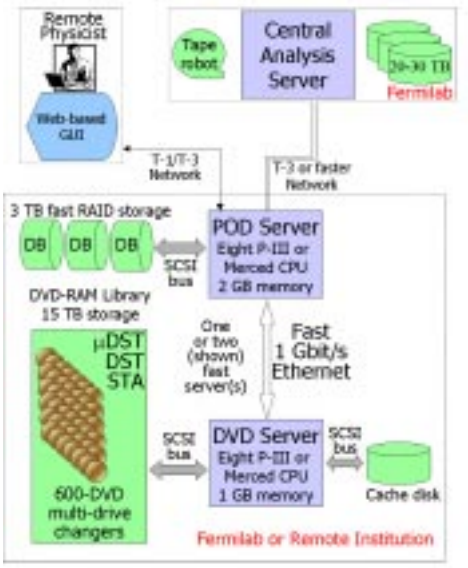

Figure 1: Schematic drawing showing the PHASER architecture. Also shown are the interactions between PHASER, DØ physicists, and the DØ Central Analysis Server.

- Direct access to the full µDST sample on DVD allows a µDST subset containing only the selected events to be quickly generated. This is particularly useful for advanced analyses that are developing innovative ID algorithms not yet reflected in POD Particle ID tables.

In the sections below, we describe the POD database and prototype studies performed using the DØ Run 1 data set acquired during the 1992-1996 Fermilab Tevatron collider run, the DVD storage used to hold the full µDST data set, and the design of a web-based user interface to PHASER.

2. The Physics Object Database

The Physics Object Database (POD) is used to store the fully calibrated meta-data associated with the various physics objects (*e.g.* leptons, photons, jets, missing E_T, secondary vertices, triggers, etc.) in an event. There is typically a fixed set of attributes associated with each physics object, so a relational database is a good choice. For example, an electron candidate would have its energy, direction, and various quantities used in the electron ID algorithms stored in the database. Each different type of physics object is associated with a table in a relational database, with each object attribute associated with a database column and each instance of the physics object stored in a database row. Table 1 lists the physics objects used to store the DØ Run 1 data in our prototype studies described below.

A primary key is needed for each table to uniquely identify each physics object. For physics objects that only have a single instance in an event (such as missing E_T), we use the run number, event number, and an index to the data source to uniquely identify the object. For objects that may have more than one instance (such as electrons), the primary key includes the instance number. While the primary key described above is not the most compact one possible, it has the ability to link together the physics objects from a given event and can be constructed from the source data without reference to the database.

Table 1 : Properties of the Physics Object tables used to store 62 million events from the DØ Run 1 data sample. Shown are the number of data columns (not including the primary key), the number of rows in the table, and the storage required for the table (including the primary key and database overhead).

Object	Columns	Rows	Size (MB)
Electron	28	52,540,491	6,979
Muon	37	79,688,956	13,576
Photon	22	69,278,259	7,647
Jets (3 cone sizes)	3 x 14	472,626,080	36,672
Jets with e/γ removed (3 cone sizes)	3 x 6	67,003,537	3,162
Missing E_T	14	62,353,601	4,951
Vertex	6	90,004,529	4,248
Trigger	19	62,353,601	3,566
Event Parameters	5	62,353,601	1,817
Totals	191	1,018,202,655	82,618

The DØ experiment expects that >99.5% of the events acquired during the next run will originate from QCD processes. To identify the small fraction of events with a non-QCD signature (*e.g.* a high p_T lepton), a variety of post-reconstruction particle ID algorithms are applied to the data representing different compromises between efficiency and background rejection. It is quite useful to store the results of the particle ID algorithms in Particle ID tables to speed up the event selection. For example, the LooseElectron table has 748,811 entries stored in 25 MB. These compact object ID tables are a key feature of the Physics Object Database.

We have studied the performance of a POD prototype by loading the DØ Run 1 data into a relational database. We first generated column-wise ntuples from >62 million events that make up the standard μDST data set. The ntuples were then loaded into the database, requiring ~80 GB of disk storage; ~70% of the disk space is devoted to physics object data storage, with the remaining 30% equally split between database overhead and primary key storage.

The database server we used is a 450 MHz dual-processor Pentium II with 256 MB of RAM. IO throughput appeared to be limited to <18 MB/s by an older generation RAID controller. One of the advantages in choosing a relational database to store the POD data is that there are a number of vendors with quite capable products. The results presented here are based on Microsoft SQL Server 7 running on Windows NT 4.0.

To benchmark the performance of the POD prototype, we generated queries that selected events consistent with the processes $Z \rightarrow ee$ and $W \rightarrow e\nu$. We found that the Z selection takes 7 seconds (~6k events) and the W selection takes 18 seconds (~86k events) when we make use of the Particle ID tables. These are extremely impressive results, especially when compared to the ~1000 CPU hours on an SGI Challenge required to generate the ntuples used in this study.

3. DVD-RAM Storage of Summary Data Sets

Many physics analyses will want to use features of the events not contained in POD. Therefore, it is useful to complement the database with mass storage containing more complete data sets that can quickly deliver the required data for events selected by the database query, as well as be used to update the database with new object ID information as better tuned algorithms become available. Such a combination results in a powerful and versatile Physics Analysis Server.

The requirements for the mass storage system include the ability to directly access a given event, unattended processing, and expandability. We have selected DVD-RAM technology for the mass storage system. This decision is based on the fact that multi-Terabyte DVD libraries are less expensive than disk storage, do not require tape backup, and, unlike magnetic tape storage, provide fast direct access to individual events. DVD technology is well supported by the computer industry, and is quickly becoming the *de facto* standard of large capacity storage. Currently, the DVD-RAM format has a storage capacity of approximately 2.6 GB/side, with new, 4.7 GB/side disks to appear early this year. Recordable DVD media offer an extremely long shelf life (300 years) and have redundant information for the recovery of simple data errors, *e.g.* those coming from small scratches on the DVD surface.

Commercially available DVD-RAM changers can hold up to 600 DVD-RAM disks and up to six DVD-RAM drives. The average disk load time is 4.5 s, so that all disks can be cycled through in less than an hour. Current data transfer speeds are 1.4 MB/s per drive, with increases in transfer speed expected in the future. These properties meet our requirements, providing fast and inexpensive access to the full μDST data needed for advanced analyses.

The database will not only be used as the front-end event selector, but also will track the content of the DVD library, making direct access to any given event a straightforward task. In fact, all library operations will be done through the database, which ensures that the database always has up-to-date information on the files in the DVD storage system.

4. Web-based User Interface

We plan to develop a web-based user interface for the PHASER system. The user will enter the event selection requirements using a Java applet. For example, a user selecting $W \rightarrow e\nu$ events might ask for events with an electron object satisfying the "Tight Electron" particle ID algorithm with a transverse momentum of at least 25 GeV and at least 25 GeV of missing E_T. The Java applet will then communicate the request using CORBA to the "Physics Intelligence" middleware running on the PHASER system. The middleware will generate the appropriate SQL syntax needed to identify the events, ensure that the requested event selection can be reasonably accommodated, issue the query to POD, and produce the requested output files.

The user will have the choice of several output options, including:

- A list of run and event numbers that satisfy the request,
- An ntuple created from the information stored in POD, and
- A μDST stream containing the requested events obtained from the DVD library.

The output files will generally be small enough to transfer over the network. Alternatively, a DVD disk can be written and physically sent to the physicist for further analysis.

5. Conclusions

The PHASER architecture appears to be well suited for efficiently analyzing multi-Terabyte HEP data samples. Prototype studies of the Physics Object Database are quite encouraging, showing that typical event selections using the DØ Run 1 data set can be performed in a matter of seconds. Integrating the POD event selection with DVD storage of the full μDST sample enables a wide range of physics analyses to be performed locally at a reasonable cost, which may be particularly attractive to remote institutions without fast access to the primary data sets. Further information about POD can be found at http://www.hep.brown.edu/phaser.