New Data Storage Model for H1

T. Benisch[1] , *U. Berthon*[2], *R. Gerhards*[1] ,*C. Grab*[3], *T. Hadig*[4]

[1] DESY, Hamburg, Germany
[2] LPNHE, Ècole Polytechnique, IN2P3–CNRS, Palaiseau, France
[3] Institut für Teilchenphysik, ETH, Zürich, Switzerland
[4] I. Physikalisches Institut der RWTH, Aachen, Germany

Abstract

The ep collider HERA and the H1 experiment, at DESY in Hamburg, are performing substantial upgrades in the year 2000. To cope with the increased demands on data storage and data handling, the H1 computing and offline environment will undergo a major revision. After a discussion of the present system and its shortcomings, the new H1 data storage model will be presented. In the new scheme, data will be stored in a multi–level hierarchical storage system implemented in the ROOT framework. Important features of ROOT which led to this choice are discussed. Also, first preliminary benchmark results are presented.

Keywords: H1, object persistency, data model, ROOT, object–oriented framework, C++

1 Introduction

After the HERA and H1 upgrade in the year 2000, an expected increase in luminosity by a factor of 5 will put new demands on data storage and data handling. Therefore, the H1 collaboration has decided to move towards a new technology and adopt the ROOT [1] framework for data storage, physics analysis and event display. This paper describes the new H1 data storage model, the new analysis environment is presented in [2].

2 Shortcomings of the present Data Storage Model

In the present environment [3, 4], data storage is based on the memory management system BOS (Bank Organisation System) and the I/O package FPACK. An Entity–Relationship model is used as a basis for data structures. Data are stored in so–called BOS banks, which are tables described by a data definition language (DDL). Each bank has a specific four character name and contains data, which are closely related. The machine independent I/O package FPACK provides a simple and unique system for data transfer with a common user interface for all operating systems used within the experiment.

The H1 raw data (RAW) are reconstructed and classified within a few hours after data taking. The raw and reconstructed data are stored on tape as production output tapes (POT). The data summary tapes (DST), a reduced set of information, are kept on disk to allow easy and parallel access for physics analysis. The users run their analysis jobs on the DSTs in batch mode and produce Ntuples, which are analysed with PAW. Most of the reconstruction, simulation and analysis code is thereby written in Fortran.

However, the present system has some shortcomings. The current H1 analysis environment is rather inhomogenous: data storage is based on BOS/FPACK, physics analysis is done with PAW and the event display uses the graphics package LOOK. Also, in the present analysis framework, no direct access to hits from tracks is possible. In addition, event selection is based on a predefined, static 32 bit classification word. In physics analysis on DST level, the full event is always read; reading of single variables is not possible. And finally, due to the detector and machine upgrade and the higher demands on the system, the maintenance of the present system will become increasingly complicated.

3 New Data Storage Model

After the upgrade the optimisation of data access will be an issue of extreme importance. Therefore, H1 has chosen a new technology and will implement a new data storage model in the ROOT framework. Important features of the ROOT I/O model which led to this choice include support of both sequential and direct data access techniques, support of networking, and the possibility of splitting event data into several streams and writing these streams into one or several files. ROOT also supports a flexible, built–in gzip–type compression algorithm.

Tools provided by the ROOT project allow the H1 collaboration to make a smooth adiabatic transition from the present sequential event data access to the new scheme. It will be necessary to split the event data into several parts, according to the frequency of their access, and to keep these parts on different levels of a multi–level hierarchical storage system. We expect that a mode of data access in which only part of the event data is read will be essential for many physics analyses. A multi–level design of data storage is foreseen such that each higher level contains less data per event, corresponding to a higher level of abstraction. The expected levels are POT, ODS, μODS and HAT (see Figure 1).

The POT (production output tape) level contains the reconstructed and raw data. They will remain in their present BOS/FPACK format and framework. By keeping the old data format at this

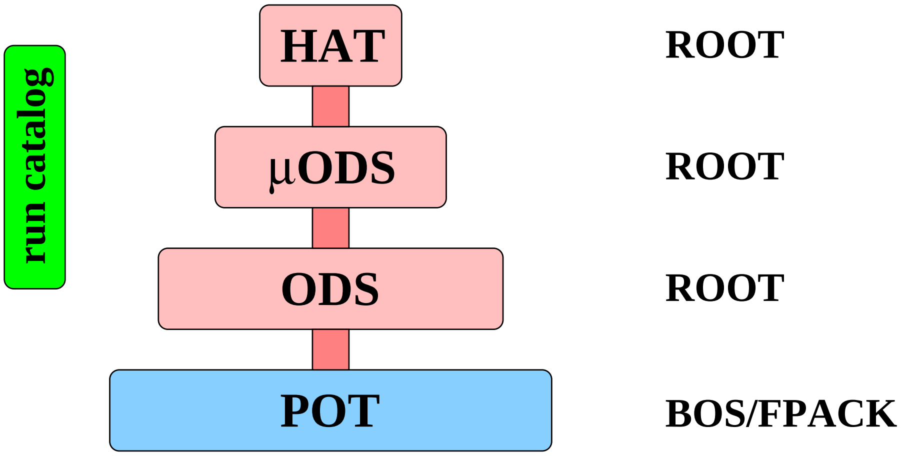

Figure 1: New H1 data storage model.

Table I: H1 data volumes after the luminosity upgrade in the year 2000.

storage level	event size (kB)	data volume (GB/year)
POT	200	10000
ODS	15	500
μODS	3	100
HAT	0.5	15

storage level, no change to the software producing the POTs, e.g. the Fortran based reconstruction program, is necessary. All higher storage levels are implemented in the ROOT framework and written out as ROOT trees. These will be the basis for an object–oriented analysis in C++. The POT event size will be on the order of 200 kB with an expected total data volume of 10 TB per year (see Table I). The POTs are stored on tape only, all other levels will be also kept on disk.

The second storage level consists of the so–called ODS (object data store), which corresponds to the present DST and contains all relevant information for physics analysis, e.g. tracks and clusters. The ODS event size is of the order of 15 kB. At this level the existing event model is not changed, but the BOS bank structures are directly mapped into objects. Each bank is written to a seperate branch so that the ROOT concept of parallel multi–buffer I/O can be used. The package BOS2OOP will allow file conversion in both directions, from DST to ODS and vice versa. This guarantees backward compatibility during the transition phase from the old to the new data handling system.

The third storage level is the μODS, which contains information on the particle level and their 4–vectors, e.g. electron candidates, jet properties, energy flow, PID probabilities, etc. The μODS is thought to be the workhorse for physics analysis. Therefore, the full expert knowledge of all physics working groups will flow into these data. The filling code for the μODS will be part of the new H1 physics analysis package H1PHAN++ which will be written in C++ in an object–oriented way based on the algorithms of the old physics analysis package. In order to offer significant speed advantages over the ODS and to allow storage of selected μODS event samples on local machines, the μODS event size will be limited to 3 kB.

The highest storage level is the HAT (H1 Analysis Tag), a tag database for fast and dynamic data selection. The HAT will contain about 100 kinematic event variables, e.g. Q^2, x, y, etc. ($\approx$ 0.5 kB/event). A lot of experience can be used from the present tag database which was implemented by using the commercial object database Objectivity/DB [5]. With this scheme, queries executed on one or more variables or objects can be executed very efficiently since only the branch buffers containing these variables will be read into memory.

Although all the event data itself will be stored in the ROOT file format, H1 also uses databases to store a large variety of information. All the run–wise information, including calibration constants, trigger rates etc., are stored in an Oracle database. The new data handling scheme will also include a run catalog which will allow access to the different parts of an event on the above mentioned storage levels. At the moment, a prototype is implemented by using the relational database MySQL but Oracle is considered to be an alternative.

4 Benchmark Results

The tree structure of a ROOT file allows the user to read in only those branches which are necessary for the analysis and thus minimizes the required I/O bandwith. There are several benchmark results available which compare the I/O performance of ROOT to that of other I/O packages. A significant part of the benchmark testing has been done at CERN by the ROOT team which compared ROOT to ZEBRA [6] and Objectivity/DB [7]. There are also benchmark results from CDF on comparisons of ROOT to YBOS [8]. Whereas detailed information can be found in the references, it can be summarized that the ROOT I/O technology outperforms its existing competitors in the cases where large data volumes are considered.

H1 has made some preliminary performance comparisons between ROOT and FPACK. Here is a brief summary of the results. The size of a DST file converted with compression level 2 into the ROOT format is a factor 3 smaller than the size of the original FPACK file. If no compression is used, the ROOT file is a factor 1.3 bigger than the FPACK file. In a mode where the whole event is being read in, ROOT reads the data out about a factor 4.6 slower than FPACK. It is expected, that with some optimization of the ROOT I/O the data readout speed will improve. In addition, partial event reading and the use of event selection with the tag database are expected to provide significant speed improvements.

5 Conclusions

The new H1 data storage model has been presented. We have shown that, due to the unique features provided by ROOT I/O and the available benchmark results, the ROOT framework will provide an efficient solution for data handling and storage after the luminosity upgrade.

References

1 R. Brun and F. Rademakers: *ROOT: An Object Oriented Data Analysis Framework.* Proceedings AIHENP'96 Workshop, Lausanne, Switzerland, September 1996, Nucl. Instr. Meth. **A389** (1997) 81. See also `http://root.cern.ch/`

2 U. Berthon et al.: *New Data Analysis Environment in H1.* Submitted to this conference.

3 R. Gerhards et al.: *Data Storage and Data Access at H1.* Proceedings of Computing In High Energy Physics 95 (CHEP95), Rio de Janeiro, Brazil, September 18 – 22, 1995.

4 C. Grab et al.: *New Developments in H1 Computing.* International Conference on Computing in High Energy Physics 98 (CHEP98), Chicago, Illinois, August 31 – September 4, 1998.

5 `http://www.objectivity.com`

6 `http://root.cern.ch/root/Zbench.html`

7 R. Brun and F. Rademakers: *Performance Comparison between ROOT, Objectivity/DB and LHC++ histOOgrams.* See `http://root.cern.ch/root/Benchmark.html`

8 R. Brun, P. Murat, F. Rademakers: *ROOT–based data handling system for CDF in Run II.* CDF internal note 4497 (1998).