Data clustering research in CMS

Koen Holtman

CERN, EP division, CH-1211 Geneva 23, Switzerland

Abstract

The clustering of objects in an object database is the mapping of objects to locations on physical storage media like disk farms and tapes. The performance of the database, and the physics application on top of it, depends crucially on having a good match between the object clustering and the database access patterns of the physics application. We discuss the results and conclusions of a 3-year research project on clustering and reclustering, that has been performed by CMS as part of its contribution to RD45. We focus on the implications of the project results for the long term LHC computing strategy and risk analysis. We give an overview of the risks related to the I/O capacity needs for LHC physics analysis, and discuss how the use of automatic reclustering systems can mitigate some of these risks. Based on our project experience, we also speculate on which risks can be successfully handled, for example through large scale simulation studies.

Keywords: data clustering, object clustering, reclustering, CMS, physics analysis, high performance I/O, risk analysis

1 Introduction

This paper reports on the results of a 3-year research project on clustering and reclustering, which has been performed in CMS as part of its contribution to the RD45 collaboration [1]. The project ran from 1997 to 1999, and was structured as a research project in the CERN technical and doctoral student programme, with additional funding by the Stan Ackermans Institute at Eindhoven University of technology. Outside of this project, work on clustering has also been performed elsewhere in HEP, see for example [2] [3] [4] [5].

The *clustering* of objects (pieces of data) in an object database is the mapping of objects to locations on physical storage media like disk farms and tapes. The performance of the database, and the physics application on top of it, depends crucially on having a good match between the object clustering and the database access operations performed by the physics application. *Reclustering* is the changing of a clustering arrangement to re-optimise performance in the face of changing access patterns. Clustering and reclustering are not issues specific to object databases. At the core, the clustering problem is one of reducing disk seeks, tape seeks and tape mounts, and this problem exists equally well in physics analysis systems that directly use the filesystem for data storage, without a database on top of it.

The main results of the project are the following.

- A rigorous study has been made of the I/O performance issues surrounding physics analysis software, with a particular emphasis on the later stages of physics analysis [6] [8]. As a result of this study, clustering and reclustering were chosen as the main focus of the project.
- A set of basic storage management strategies has been developed [8] [13].

- A running prototype of a disk based storage management system, which includes reclustering optimisations, was created [7] [14].
- A design of a full-scale storage management system to optimise disk and tape access has been made. The performance of this system has been analysed under a range of physics workloads [11] [12].
- The scalability of I/O intensive physics applications, based on Objectivity/DB, was studied up to 240 concurrent worker processes and up to 170 MB/s throughput [9] [10]. These studies also confirmed the scalability of the basic storage management and (re)clustering strategies developed in the project.

In the remainder of this paper, some specific results and conclusions from the project are discussed. We focus on results related to the risk analysis and long term computing strategy for LHC offline physics analysis.

2 I/O related risks in LHC computing

At the core of our I/O related risk analysis for LHC computing is the risk that the physics analysis system, in production from 2005 on, will deliver insufficient I/O speed to support the physics analysis goals of the experiment. There are two sides to this: how much speed is needed, and how much can be feasibly offered. Unfortunately, there are huge error bars on the estimates of the I/O speed needs for physics analysis. Some of this uncertainty is inevitable. For example, the offline physics analysis I/O speed needed to find the Higgs boson depends for a large part on the mass of the Higgs boson. Another part of the uncertainty will be reduced in future, as performance of the (sub)detectors under construction becomes better known. The error bars will become smaller as we near turn-on in 2005, but we do not expect that they will become much smaller. Thus, before turn-on, the best strategy to meet the I/O needs for LHC physics analysis is to maximise the I/O speed that can be offered in 2005 as much as possible. The speed that can be offered depends for some part on the hardware performance in 2005. Extrapolating current trends for hard disks, and the likely computing budget, it seems likely that some 50-200 GB/s of sequential I/O disk speed can be available to CMS in 2005. The error bar on this hardware number is about a factor 2 to either side: much lower than the error bars on the estimates of the I/O needs. Predictions for other devices, except WAN links, have similar error bars. However, the above GB/s number for sequential I/O which is the best case in I/O performance, and there is a large gap between best-case and worst-case I/O performance. For example, a current hard disk can read sequentially at some 5 MB/s, and randomly (the worst-case) at some 100 *objects* (contiguous pieces of data) per second. When reading, say, 1 KB objects, this amounts to a speed of 0.1 MB/s, a gap of a factor 50. The best-case worst-case gap for disk devices will only widen in future. For tape, of course, the the gap is still larger: for truly random tape access one can expect a speed of 1 object per minute. The actual I/O performance delivered to the physics application thus depends crucially on the access pattern of the application, which determines the actual I/O performance.

Unfortunately, there are large error bars on the expected access patterns for physics analysis, so one cannot just assume that the above best-case sequential reading hardware speed of 50-200 GB/s will be delivered to the physics applications. Due to the application of cut predicates, physics applications will often display *selective reading* patterns, in which the application moves over a set of objects sequentially, the order in which the objects have been clustered, but actually reads only a fraction of these objects. The potential I/O performance loss due to selective reading has been studied extensively in this project, and it was concluded that a reclustering optimisation is needed to prevent the occurrence of highly selective reading, with often worst-case I/O performance, when many cut predicates are used. The availability of reclustering makes the large error bars on the

expected access patterns unimportant: no matter what the pattern, reclustering will ensure that the I/O performance remains near the best-case performance of sequential I/O. Reclustering can be done automatically or by hand. Automatic systems were developed in this project.

When applied to physics data on disk, it is possible for (automatic) reclustering to correct any efficiency problems due to a bad initial clustering on disk. For physics data on tape, this is also possible in theory. However, it was found in this project that, given the current estimates of the parameters for LHC physics analysis, reclustering on tape is unlikely to be cost-effective. The object sets staged from tape will generally be cached on disk, so the access to object sets on tape is not as repetitive as it is to object sets on disk. The less repetitive the access, the more reclustering operations are needed to make the clustering arrangement reflect changing access patterns. In simulations with likely parameters, the cost of these many reclustering operations often approaches, or even equals, the savings, because of the better clustering, that these operations produce. Because of the often marginal benefit of tape reclustering, the effectiveness of the initial clustering on tape will remain very important in determining the overall tape performance. Unfortunately, because of the error bars on the access patterns, this effectiveness cannot be estimated well. Research on improving the efficiency of clusterings on tape is ongoing [3], but we will probably not know until some time after turn-on whether the clustering that was on tape is efficient enough.

The above result on tape reclustering has been obtained through a large scale simulation study over a wide range of access patterns [11] [12] [13], this wide range allowed us to take some of the error bars on the access patterns into account. Simulation studies cannot prove that the simulated system is fast enough to satisfy the I/O needs for LHC physics analysis, because the error bars on the needs are too wide. Simulation over a large parameter space is a very valuable tool however in determining whether a particular proposed optimisation has a worthwhile payoff. As such, simulation can be used to drive the architectural effort, separating the good optimisation ideas from those that look good in theory, but whose effects are, in practice, lost in the noise. It is crucial to find the effective optimisations among the huge number of optimisations that can be thought up. The need to maximise the I/O speed as much as possible, in order to minimise the risk that this speed is insufficient for the physics goals, should not be used as an excuse to implement any optimisation that can be thought up. If too many optimisations are implemented, the software system will collapse under its own complexity. To focus our work in this project, we used the guideline that an optimisation should only be pursued if it yields a potential performance improvement of at least a factor 2.

Simulation is a particularly important design tool in the area of tape based and WAN-based physics analysis, where it is important to quantify the effects of caching under 'chaotic' multi-user workloads. To better support such simulations, more knowledge is needed about access patterns in the group-level and user-level stages of physics analysis for the LHC. Such knowledge could be gained by examining access pattern or query logs in current experiments in current Monte Carlo studies for the LHC.

Performance measurements on running prototypes have traditionally played a large role in HEP, to validate architectural decisions, and we expect that this will remain the case. One of the risks involved in this strategy is that scalability to the size of the 2005 production configuration cannot be shown on today's smaller configurations. This risk was handled in this project by developing I/O policies with not just good, but extremely good scaling curves on current large configurations. These policies centre around creating sequential access patterns for every worker process individually, and using read-ahead optimisation [9] which causes 'bursty sequential reading' [12] overall, even when the physics code interleaves I/O and computation with a very fine grain size.

3 Conclusions

This paper has reported on the results of a 3-year research project on clustering and reclustering, which has been performed in CMS as part of its contribution to the RD45 collaboration [1]. More information on the project and its results can be found at `http://home.cern.ch/~kholtman/`, and in the various publications produced in the project, [6], [7], [8], [9], [10], [11], [12], and [13]. Two software packages were also released, [14] and [15].

References

1 RD45, A Persistent Storage Manager for HEP. `http://wwwcn.cern.ch/asd/cernlib/rd45/`

2 M. Schaller, Reclustering of High Energy Physics Data. Proceedings of SSDBM'99, Cleveland, Ohio, July 28-30, 1999.

3 Grand Challenge Application on HENP Data. `http://www-rnc.lbl.gov/GC/`

4 HEPODBMS reference manual. `http://wwwinfo.cern.ch/asd/lhc++/HepODBMS/reference-manual/index.html`

5 J. Becla, Data clustering and placement for the BaBar database. Proceedings of CHEP'98, Chicago, USA.

6 K. Holtman, Prototyping of CMS Storage Management, CMS NOTE/1997 - 074.

7 K. Holtman, P. van der Stok, I. Willers. Automatic Reclustering of Objects in Very Large Databases for High Energy Physics, Proceedings of IDEAS '98, Cardiff, UK, p. 132-140, IEEE 1998.

8 K. Holtman, Clustering and Reclustering HEP Data in Object Databases. Proceedings of CHEP'98, Chicago, USA.

9 K. Holtman, J. Bunn. Scalability to Hundreds of Clients in HEP Object Databases. Proceedings of CHEP'98, Chicago, USA.

10 J. Bunn, K. Holtman. H. Newman. Object Database Scalability for Scientific Workloads. Technical report, awaiting formal publication in CMS. Available from `http://home.cern.ch/~kholtman/`

11 K. Holtman, P. van der Stok, I. Willers. A Cache Filtering Optimisation for Queries to Massive Datasets on Tertiary Storage. Proceedings of DOLAP'99, Kansas City, USA, November 6, 1999.

12 K. Holtman, P. van der Stok, I. Willers. Towards Mass Storage Systems with Object Granularity. Proceedings of the Eighth NASA Goddard Conference on Mass Storage Systems and Technologies, Maryland, USA, March 27-30, 2000. (To be published)

13 K. Holtman, Prototyping of CMS Storage Management. Ph.D. Thesis, Eindhoven University of Technology, Spring 2000. (to be published).

14 Reclustering Object Store Library for LHC++, V2.1. Available from `http://home.cern.ch/~kholtman/`

15 TOPS, Testbed for Objectivity Performance and Scalability, V1.1. Available from `http://home.cern.ch/~kholtman/`