

# Iteration Aware Prefetching for Scientific Data

Dan R. Lipsa  
Department of Information  
Technology Armstrong Atlantic  
State University Savannah, GA  
31419  
dlipsa@drake.armstrong.edu

R. Daniel Bergeron  
Department of Computer Science  
University of New Hampshire  
Durham, NH 03824  
rdb@cs.unh.edu

Philip J. Rhodes  
Department of Computer and  
Information Technology University  
of Mississippi  
rhodes@cs.olemiss.edu

## 1 Problem and Motivation

We define scientific data as multidimensional data obtained either from the real world or from a simulation. Two characteristics which are important for our research are that this data is large in size and the retrieval is often “volumetric”, meaning that data retrieval is done by querying a sub-volume of data. [4].

Today’s scientific data are measured in terabytes. The sensors used to acquire this data are becoming more and more sophisticated; therefore, the size of the data that we need to process is likely to keep increasing.

In the past fifteen years there was a thousand-fold increase in processor speed, and even a greater increase in memory and hard-disk size [1, 2]. However, the average seek time of hard disk drives have improved only marginally in the same time period. If this trend continues in the future, hard disk seek time will lag even further behind. This fact negatively affects the speed of scientific data retrieval.

A file system uses a general method (file system cache) to reduce the effect of slow hard disk seek time, which does not take into consideration the multidimensional nature of scientific data, and the volumetric retrieval performed on the data.

The file system cache usually prefetches and stores data that is nearby *in the file*, to the data retrieved by a “read” command. However, that data might not be nearby *in the volume* represented by the file, so the file system caching method will not prefetch or store data immediately required by the retrieval method (retrieval is volumetric). Due to this fact, this caching method is not useful for scientific data retrieval.

There are other methods for improving access to scientific data [5]. Their major drawback is that they require reorganization of the data according to the expected pattern of access. This process can be inconvenient because of the huge size of the data, and because the reorganization might not match all possible ways data will be re-

trieved.

## 2 Background and Related Work

In [3] the authors designed a toolkit for supporting visualizations on complex scientific data. To improve performance, they implemented a caching system using a paged array. They also implemented intelligent iterators that use the knowledge of the N-dimensional structure of the data to guide prefetching. The difference from our research is that they use prefetching to overlap computation and I/O, we use it to minimize the numbers of read commands issued, and so to minimize the impact of hard-drive latency time on the total read time.

In [6, 7] the authors present the model for iteration aware prefetching, but only for directions of retrieval along the principal axes. We plan to extend this work to cover iteration along an arbitrary axis and evaluate the speed-up that may result from this new approach. In [5] the authors present a comprehensive model and implementation for a scientific database and they also cover iteration aware prefetching only along the principal axes.

## 3 Approach and Uniqueness

Our solution uses the multidimensional nature of the data and the volumetric nature of the access, and does not require changing the original data. To speed up the access to scientific data, we are going to use a multidimensional cache block, that is built based on the access pattern through the data. In our solution, an iterator (an object that is used for retrieving data in a certain order) is used, both to specify in advance the access patterns through the data, and to retrieve the data.

To implement our solution, we are going to enhance the Granite Library [5] built at the University of New Hampshire. We are going to develop, test and evaluate an iterator and the associated cache method for retrieval in an arbitrary direction of the volume of data.

## 4 Results and Contributions

The outcome of this project is the development, testing and evaluation of an iterator together with the associated cache. The retrieval (iteration) will be done in an arbitrary direction. For evaluation, we are going to use a test application that displays progressive two-dimensional slices of a three-dimensional volume. The data displayed will be the 39 GB Visible Woman dataset from the National Institutes of Health. The evaluation will consist of timing the retrieval of the entire volume of test data. One parameter will be the

angle at which the retrieval will be done. Another parameter will be the cache memory size. In all these cases we are going to measure the speed-up when compared with retrieval without any cache. These results will show if our caching method provides adequate speed-up not only for an iteration along a principal axis, but for an iteration along an arbitrary axis as well.

## 5 Additional Authors

Ted M. Sparr, Department of Computer Science, University of New Hampshire, Durham, NH 03824 email:tms@cs.unh.edu

## 6 References

- [1] F.W. Chang. *Using Speculative Execution to Automatically Hide I/O Latency*. PhD thesis, School of Computer Science, Carnegie Mellon University, 2001.
- [2] Thomas Coughlin. High density hard disk drive trends in the usa. Technical report, <http://www.tomcoughlin.com>, 2000.
- [3] David R. Nadeau. An architecture for large multi-dimensional data management. Scalable Visualisation Tools White Paper, <http://vistools.npaci.edu>.
- [4] John L. Pfaltz, Russell F. Haddleton, and James C. French. Scalable, parallel, scientific database. In *Proc. 10th International Conference on Scientific and Statistical Database Management*, Los Alamos, CA, 1998. IEEE.
- [5] Philip J. Rhodes. *Granite: A scientific Database Model and Implementation*. PhD thesis, University of New Hampshire, 2004.
- [6] Philip J. Rhodes, Xuan Tang, R. Daniel Bergeron, and Ted M. Sparr. Iteration aware prefetching for large multidimensional scientific datasets. In *SSDBM'2005: Proc. of the 17th international conference on Scientific and statistical database management*, pages 45–54, Berkeley, CA, US, 2005. Lawrence Berkeley Laboratory.
- [7] Philip J. Rhodes, Xuan Tang, R. Daniel Bergeron, and Ted M. Sparr. Out of core visualization using iterator aware multidimensional prefetching. In *SPIE VDA*, 2005.