Final report on work for Center for Gyrokinetic Particle Simulation of Turbulent Transport in Burning Plasmas — Tools for Improved Data Logistics

Micah Beck
Department of Electrical Engineering and Computer Science
University of Tennessee
September 14, 2008

Project goals:

This project focused on the use of Logistical Networking technology to address the challenges involved in rapid sharing of data from the the Center's gyrokinetic particle simulations, which can be on the order of terabytes per time step, among researchers at a number of geographically distributed locations. There is a great need to manage data on this scale in a flexible manner, with simulation code, file system, database and visualization functions requiring access. The project used distributed data management infrastructure based on Logistical Networking technology to address these issues in a way that maximized interoperability and achieved the levels of performance the required by the Center's application community. The work focused on the development and deployment of software tools and infrastructure for the storage and distribution of terascale datasets generated by simulations running at the National Center for Computational Science at Oak Ridge National Laboratory.

Project Activities and Results:

The work proceeded in two phases. In the first phase, we developed of a version of the Standard I/O library based on the Logistical Runtime System (LoRS) and collection of tools which allowed the porting of the NetCDF I/O package and later the HDF5 I/O package to make use of logistical storage resources, which are called "depots". The results of this effort consisted of software delivered for use by project scientists. This software made it possible for simulation results generated on ORNL's Jaguar Cray XT3, and moved for post processing to the Ewok cluster, to be written to local depots and then distributed at high performance to depots in the Center's community, as in Figure 1.

The primary goal of the second stage was the development and adaptation of software known as the Logistical Distribution Network (LoDN) for cataloging and managing these datasets. This work was based on research tools developed under previous NSF funding. According to specifications developed in collaboration with the Center's users, the LoDN service was adapted to meet the specific needs of the Gyro Center and to run in National Center for Computational Science (NCCS) environment at the Oak Ridge National Laboratory (ORNL). This tool was far more stable than the earlier version and could be used to place and access data far more accurately than the initial version that was developed within the Gyro Center. To achieve this end, a variety of specific requirements had to be met:

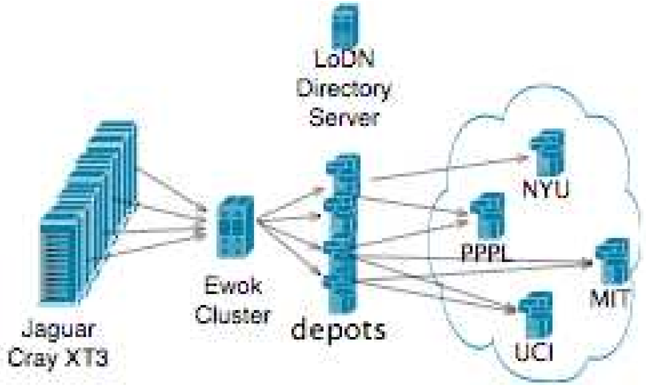

Figure 1: Distributing Gyro Center simulation results to user community using LN tools and infrastructure

- *Controlled data mirroring*: LoDN was modified to enable user control of automated data mirroring to collaborative sites on per file or (recursive) per folder basis, but firewall constraints imposed by the NCCS security environment required mirroring of the metadata generated by this process to an outside server. As delivered, LoDN significantly improved control over automated data distribution. Replica sites and striping of data over multiple depots is now user specified. Distribution of multiple copies of a single data set is now tree-structured, to take advantage of data parallelism.
- *Security measures*: The need for our LoDN Directory Service to pass the NCCS security screening meant that all Web scripts had to be vetted for security holes. In addition, SSL connections were implemented for all transactions between LoDN and its client.
- *Integration with I/0 tools and libraries*: The integration of the LoDN directory service with the POSIX & HDF5 layers involves having open files stored on LoDN by URL. Data is then automatically distributed to collaboration sites according to user-specified control on a per-file or per-directory basis.

Performance testing of the system showed little in the way of limiting speed. A single client writing to a single IBP depot on Ewok, then storing over Gigabit Ethernet to a file system implemented by Lustre achieved 25% of the bandwidth to disk seen by direct write to Lustre over Infiniband. In later work multiple depots & clients were used and the client/depot throughput was optimized.

Tests showed that application transparency was achieved in both when writing and reading the data to the wide area system. When writing, data is staged to initial depots, then distributed (Figure 1). Data verification is continuous and asynchronous. Distribution of data sets is directed by users through the use of a policy mechanism, controllable by the user, that requires no ongoing, manual intervention from either the user or the application. When reading, replicas are used for Bit Torrent-like parallel transfers. Using this technique it is possible to read data that is stored in a file that is only partially distributed or has been corrupted (Figure 2). This functionality was even further improved by the use of the *Resource and Education Data Depot Network (REDDnet)*, an NSF funded wide area storage infrastructure based on Logistical Networking depot technology. Data is read and written using an API-compatible libStdio. This includes the POSIX API reimplemented over Logistical Networking (LN), so that complex applications can be ported through simple relinking. Both local and remote files can be accessed by using URLs in place of filenames.

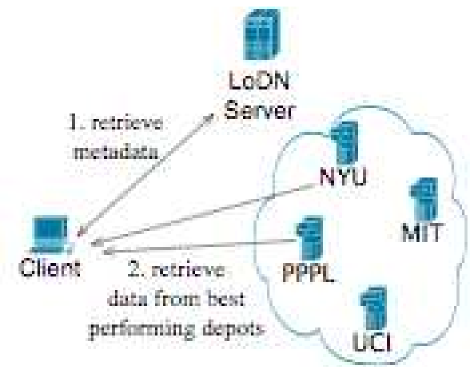

Figure 2: Achieving application transparent, high peformance reads using LN tools and infrastructure

In response to feedback from Gyro Center community, we also created two other tools to maximize ease of use:

- Filesystem in Userspace (FUSE) for LN: The FUSE package for LN allows the Linux read/write API to access LN with no relinking (LoDNFS). Extra parameters placed in user config file to maintain API interoperability. This provides a high degree of Linux compatibility.
- *Logistical Networking Copy (LNCP)*: LNCP is a version of cp, the Unix file copying utility, that has been linked with libStdio, enabling copying of data to and from files stored in Logistical Storage resources and cataloged in the Logistical Distribution Network (LoDN) directory service. It represents a single synthesis of logistical file upload and download utilities in a package that is familiar to Linux users. An important element of developing LNCP was resolving firewall issues that kept us from being able to run the LNCP client on Ewok.