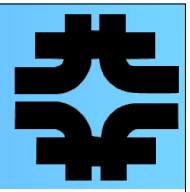


d-Cache and Grid Enabled Analysis

Ian Fisk

June 23, 2003

Introduction

What dCache is?

- D-cache is a disk caching system developed at DESY as a front end for Mass Storage Systems
  - It now has significant developer support from FNAL and is used in several running experiments
- We are using it as a way to utilize disk space on the worker nodes and efficiently supply data in intense applications like simulation with pile-up.
  - Applications access the data in d-cache space over a POSIX compliant interface. The d-cache directory (/pnfs) from the user perspective looks like any other cross mounted file system
    - Since this was designed as a front-end to MSS, once closed, files cannot be appended
- Very promising set of features for load balancing and error recovery
  - d-cache can replicate data between servers if the load is too high
  - if a server fails, d-cache can create a new pool and the application can wait until data is available.

Use in CMS

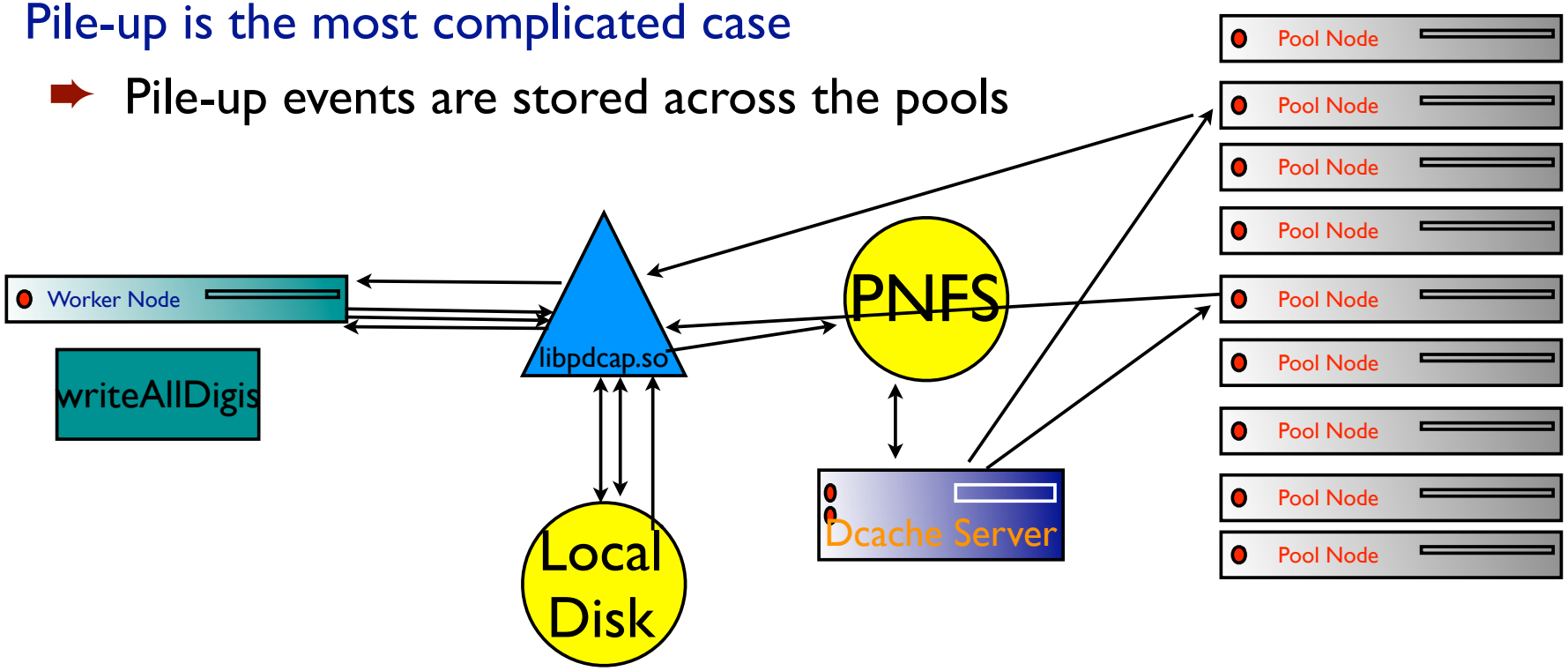
Michael Ernst and I demonstrated it could be used for CMS very high data rate applications as a replacement for some of the Objectivity AMS functionality

- CMS event building with pile-up is the most intense application
  - 70MB of pile-up accessed at random per event simulated
  - Pile-up sample is large and needs to be spread across many servers
- dCache can handle the loss of a data server without killing the application
- dCache is capable of making multiple copies of data to balance the load across data servers

Setup on Local Farm

Pile-up is the most complicated case

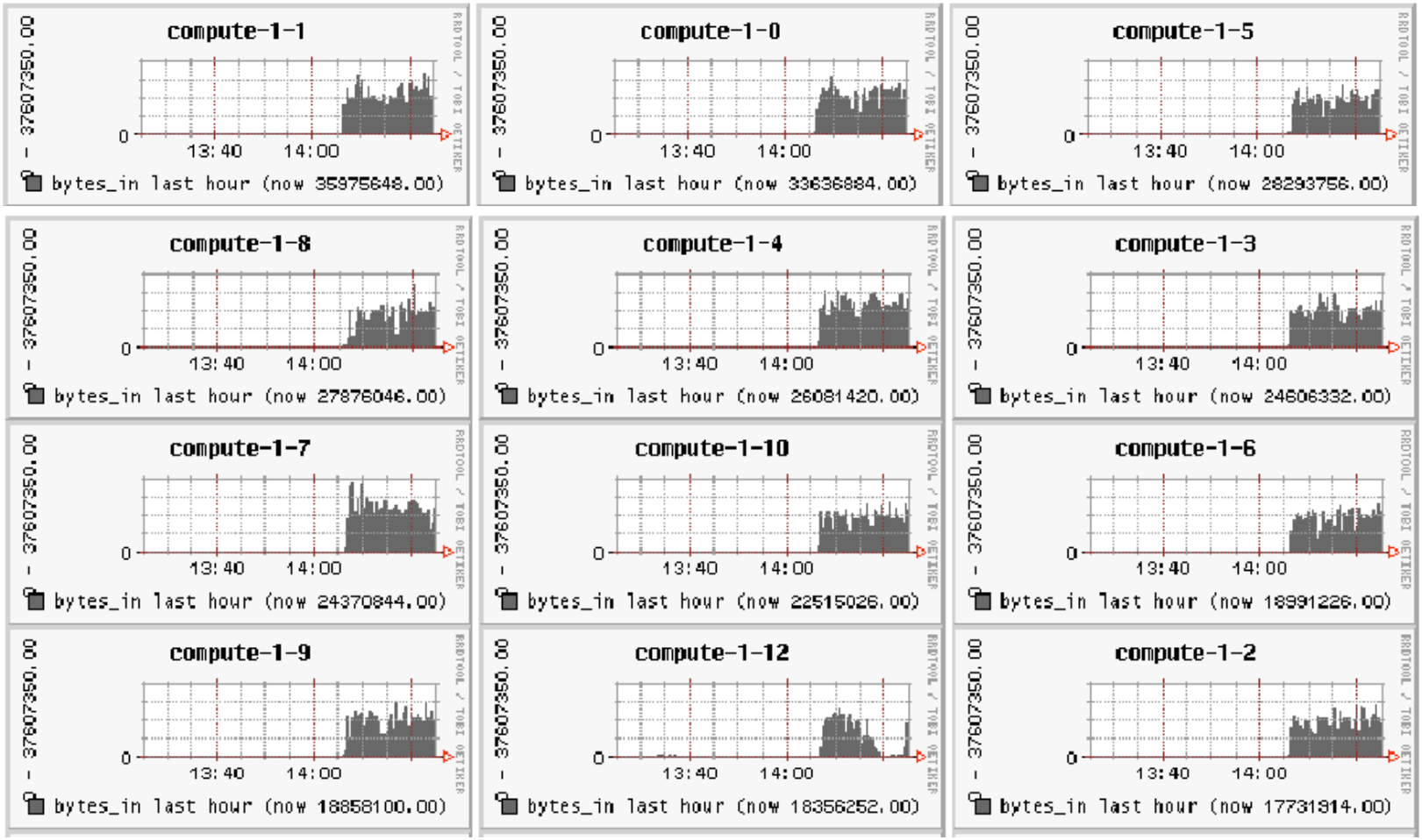
- Pile-up events are stored across the pools



- Many applications can be running in parallel each writing to their own metadata but reading the same minimum bias

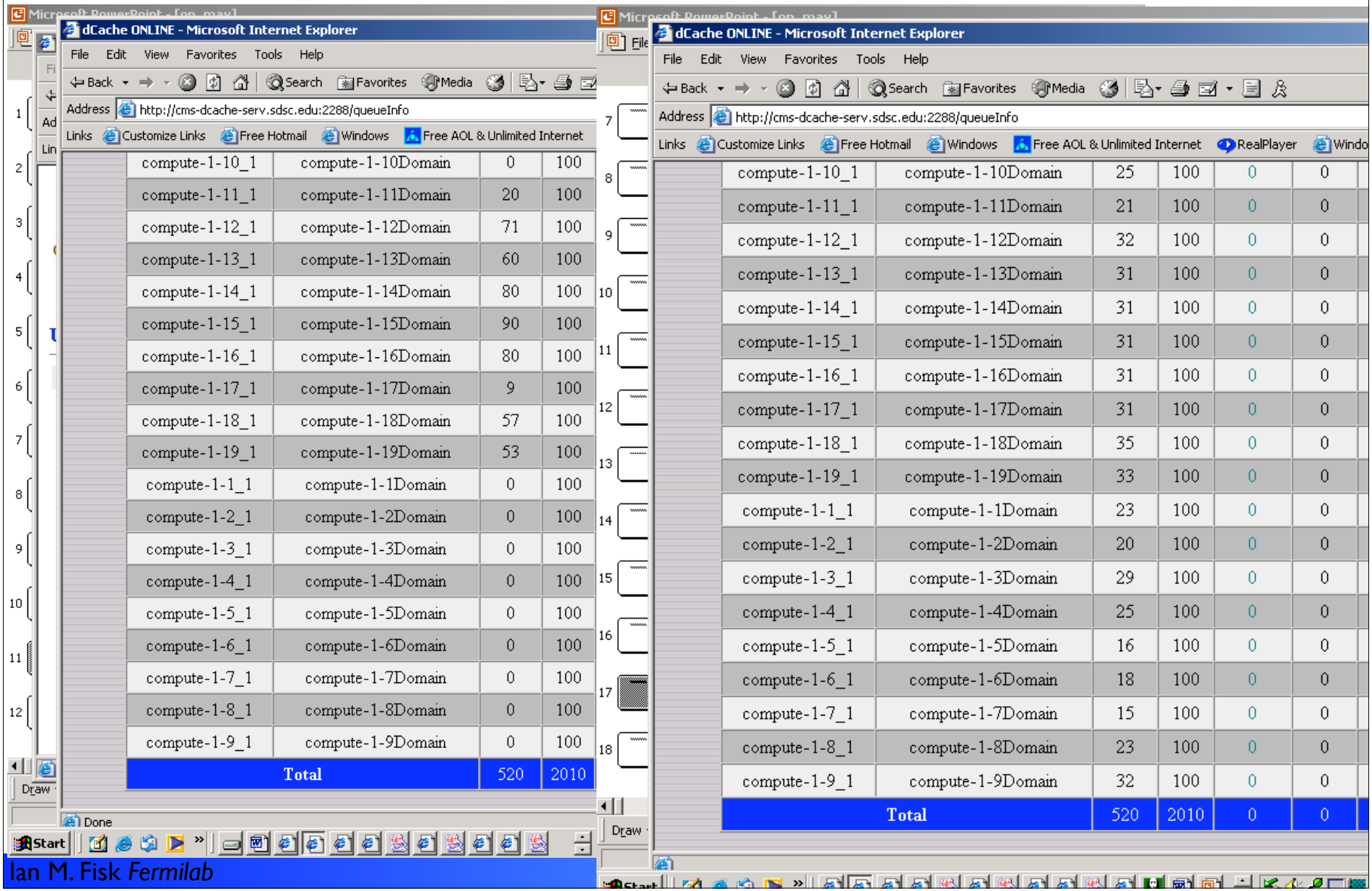
Data to Processes

Data Delivered to processes during 12 digi jobs 12 systems test

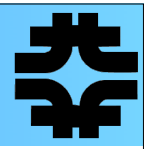


Fairly flat and stable across the test

Load Balancing



Introduction (cont.)



What dCache isn't?

- By itself dCache doesn't have any distributed data handling capabilities
  - It was designed to improve access to MSS data on a local site
  - Access to data is handled by UNIX file permissions
  - Local File Catalogue is NFS file system

- dCache has a lot of the elements that you want for a distributed analysis system
  - Fault Tolerance
  - Load Balancing

- Without addition of wide area transfer mechanism and a distributed data catalogue it doesn't have anything more to do with Grid Enabled Analysis than NFS does

What's Needed

Services are needed to build upon local data distribution tools to provide distributed and largely transparent data access

- Need implementations of RLS and RLI services to identify and distribute the data files available at sites
- Need tools for wide area replication with strong authentication
  - Needs to be linked to local data providing tools
- Need functionality to provide a data gateway for transfers in and out of centers
  - Much like the globus gateway currently used for batch requests

Current Work

There has been development on an SRM interface directly to dCache

- This provides a robust and efficient wide area transfer tool
- Provides a uniform interface to a variety of MSS

The proposal by members of the LCG and US-CMS is to create a Storage Element (SE) based on RLS and RLI data catalogue, SRM and a number of local data distribution tools

- dCache is promising because it is POSIX compliant and doesn't require changing the application
- Other local tools being investigated are RFIO, ROOTD, and simple NFS file systems

Interesting Areas of Development

A lot of the SE development has been planned with reconstruction users in mind

- Infrastructure planned is advanced, but it still expects a level of predictability in terms of the data transferred.
- At some point resource brokers will attempt to make intelligent scheduling decisions based on data location

The grid enabled analysis user doesn't currently have the infrastructure to know if a request is reasonable or possible

- The production and reconstruction user has a much more organized and predictable environment to work in.

Interesting Areas of Development

As the Storage Element Development proceeds it would be interesting work on developed the applications that help the user estimate the feasibility of an analysis requests

- User interfaces to RLS and RLI catalogues
- Estimation Tools
  - Time for transfer estimation tools based on bandwidth measurements
  - Amount of data to access