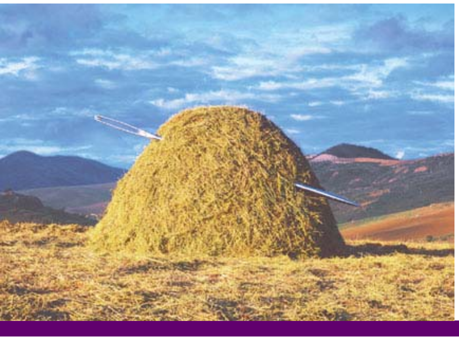


Finding the Needle in the Haystack: Metadata-Indexed Cluster Filesystems

Milo Polte and Garth Gibson
CS Department, CMU

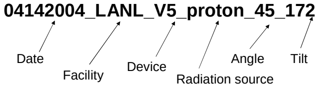
John Bent
HPC-5, LANL

Motivating Multidimensional Datasets

Effects of Radiation on Field Programmable Gate Arrays

- Studies effects of radiation on bit flip errors
- Dataset is thousands of files, millions of samples
- Groups samples into many single files whose names are a concatenation of sample attributes:

04142004_LANL_V5_proton_45_172



Work by Heather M. Quinn and Sarah Michalak at LANL

Purging a Petascale Cluster Filesystem

- Recursively walks entire file system tree to find old, large files
  - Old serial version ran for 20 hours
  - Many programmer hours later, new parallel version runs for 45 minutes

Work by Ben McClelland at LANL

Why Not a Pure Database System

- Many applications are based on the POSIX API
  - Many tools are scripts or compiled programs that might be difficult to modify to use a database
- Databases have a lot of extra things (e.g. transactions), that we don't need
- Distributed filesystems already used in large scale clusters
  - PVFS, PanFS, LUSTRE, etc.
- Our goal: Database-style search on a cluster filesystem for performance and expressiveness
- Our approach: Leverage database technology within a cluster filesystem

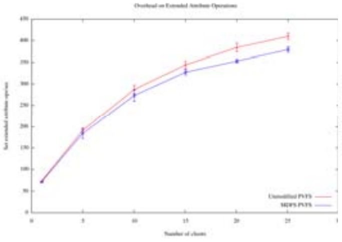
Operations: Replicating Attributes

- Client behavior remains unchanged
- Cluster filesystem asynchronously replicates attributes and extended attributes into internal database

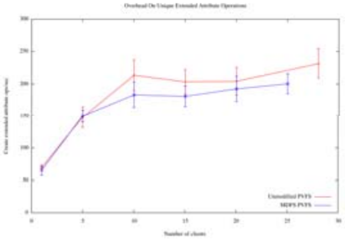
- Clients can set (and search) additional application specific tags using *setxattr*
  - Already an existing operation within some cluster filesystems

Overhead

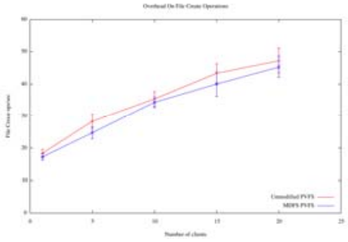
Updating Tags



Adding Application Tags



File Creation



Current File Systems and MDDS's

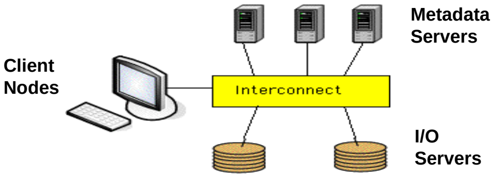
- Often users need to filter or search very large datasets
  - "Show me all proton results from LANL"
  - "Remove all large files created more than a year ago and not accessed within the last month"
  - "View all satellite images containing super-novas"
  - "Find all songs by a particular artist"
- Existing search is slow and non-parallel
- Adding a new tag is slow - may need to update (i.e. rename all files)
- Data volumes growing larger, size of data sets increasing, existing solutions becoming decreasingly tractable

"I'll be late for dinner, Dear. I'm up to my neck in paperwork."

This work was carried out under the auspices of the National Nuclear Security Administration of the U.S. Department of Energy at Los Alamos National Laboratory under Contract No. DE-AC52-06NA25396, **LA-UR-08-1959.**

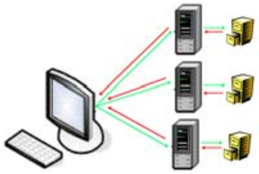
Prototype Design

- Built by extending open source Parallel Virtual File System (PVFS) distributed cluster filesystem



- Each metadata server augmented with an sqlite3 database
- Indexes both standard and extended attributes

Operations: Queries



- Querying linked to *mkdir* operation
  - On a *mkdir,* the client
    - Makes the directory normally
    - Checks if its path is of the form mkdir /mdfs/query/"<sql query>"
    - If so, issues the query to all metadata servers in parallel
      - Found files are symbolically linked into the directory
    - If not, return normally
  - Applications can use *readdir to process result of a query*

Summary

- Filename+Path is a poor way to organize large, multidimensional datasets
- Users need database style querying in cluster filesystems
- One solution is to integrate databases within filesystems
  - Retain POSIX interface as the primary application API
  - Integration results in tighter coherence, less maintenance
- Low overhead prototype demonstrates feasibility of this approach

PARALLEL DATA LABORATORY
CARNEGIE MELLON UNIVERSITY

Los Alamos
NATIONAL LABORATORY
EST. 1943