Towards a Data Management Infrastructure for the SNS

Sudharshan Vazhkudai

Network and Cluster Computing, CSMD
Oak Ridge National Laboratory

http://www.csm.ornl.gov/~vazhkuda
vazhkudaiss@ornl.gov

Acknowledgments: ORNL, SNS

NCSU Computer Science
May 27th, 2004
Outline

- The SNS Challenge
- Data Grids – The Solution…?
  - Middleware
  - Fabric
  - Application-layer Tools
Why is it a big deal?
- A 1.4 billion$ user facility intended to come online in 2006
- DOE effort comprising of Oak Ridge, Jefferson, Brookhaven, Argonne, Los Alamos and Lawrence Berkeley
- 18 beamlines capable of accommodating 24 instruments
- 10-100 x increased data rates and better resolution than HFIR (ORNL) and ISIS (UK)—SNS will provide brighter neutrons!
- Construction of SNS helps US take the lead in neutron science

Applications:
- Use of neutrons to probe structure, magnetism and dynamics of materials
  - Chemistry, condensed matter physics, metallurgy, bio-sciences, compact discs, credit cards, satellite weather information, etc.
A Few Challenges...

- Several orders of terabytes per year for the next decade
  - How to collect, manage, access and interpret such huge quantities of data?
- Computationally intensive analyses
  - Using disparate software tools: IDL, Matlab, DANSE, IRIS Explorer, DAVE, etc.
- Several hundreds—even thousands—of geographically dispersed users
- Collaborative visualization requirements
- Online experiment steering
  - Beam time is precious!
SNS Data Infrastructure

SNS-specific Tools
- Application Portals; User desktop tools; custom applications; Grid User portals
  ...

Middleware
- Efficient data access, transfer & discovery; online instrument control & experiment steering; co-allocation of resources; Mirroring of data; metadata; secure access
  ...

Grid Storage Fabric

Differentiated Storage Services
- High-end Storage
- Tape Library
Some Middleware Issues

- **Problem:**
  - To select datasets from among numerous alternatives
  - Predicting bulk data transfer times
  - Co-allocated downloads of bulk data
Replica Selection

Data Grid Snapshot:
- Lightly loaded server in possession of data

Dominant Factors:
- Server Load details
- Usage Policies
- Total Time-to-Download

Get me “this” data ASAP

A good heuristic = \( f(\text{Time-to-Download}, \text{Policies}) \)
Forecasting Background

- **Problem:**
  - An estimate for GridFTP end-to-end future data transfer performance

- **Options:**
  - **Option1:** Model components
  - **Option2:** Whole system evolution
Predictions on Past History – Univariate Forecasting

- Predictors feed on GridFTP performance logs

- Mean based:
  - **Pros:** simple
  - **Cons:** smoothes out fluctuations

- Median based:
  - **Pros:** simple, excludes extreme values
  - **Cons:** may not be representative of entire dataset

- Autoregressive models $Y_t = a + bY_{t-1}$
  - **Pros:** weighted averages
  - **Cons:** computational cost, larger datasets

- Context Sensitive & Insensitive Variants
  - File size categories
  - Training sets
  - Sliding windows
    - Count based
    - Temporal windows

<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>Median</th>
<th>ARIMA</th>
</tr>
</thead>
<tbody>
<tr>
<td>All data</td>
<td>AVG</td>
<td>MED</td>
<td>AR</td>
</tr>
<tr>
<td>Last 1 Value</td>
<td>LV</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Last 5 Values</td>
<td>AVG5</td>
<td>MED5</td>
<td></td>
</tr>
<tr>
<td>Last 15 Values</td>
<td>AVG15</td>
<td>MED15</td>
<td></td>
</tr>
<tr>
<td>Last 25 Values</td>
<td>AVG25</td>
<td>MED25</td>
<td></td>
</tr>
<tr>
<td>Last 5 Hours</td>
<td>AVG5hr</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Last 15 Hours</td>
<td>AVG15hr</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Last 25 Hours</td>
<td>AVG25hr</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Last 5 Days</td>
<td></td>
<td></td>
<td>AR5d</td>
</tr>
<tr>
<td>Last 10 Days</td>
<td></td>
<td></td>
<td>AR10d</td>
</tr>
</tbody>
</table>
Experiment Setup

- **Testbed**
  - Several sites connected by OC-12 / OC-48 links, Internet2
  - RAID Disks at servers and clients
  - Symmetric multi processors
  - Linux and Solaris operating systems
  - Prediction middleware in “C”

- **GridFTP Experiments**
  - Bulk transfers ranging from 10MB to 1GB
  - Transfers at random intervals
  - Tuned buffer sizes and parallel flows
  - ~100 to 200 transfers in various sizes during three distinct 2 week periods
Univariate Predictor Performance

- **Normalized Error:**

\[
\% \text{ Error} = \frac{\sum |\text{Measured}_{BW} - \text{Predicted}_{BW}|}{(\text{size} \cdot \text{Mean}_{BW})} \times 100
\]

- **Results:**
  - 15 – 25% error
  - 10% reduction due to classification
  - Moving variants perform better
Meta-Message: Simple averages and medians do not capture trends.
Multivariate Forecasting

- **Drawbacks of the Univariate Model:**
  - Grid Data Transfers can be quite sporadic
  - Log data alone not reflective of current system behavior
  - Component load variations can affect eventual throughput achieved

- **Why multivariate Forecasting?**
  - Mitigate the adverse effects of sporadic transfers
  - Improve prediction accuracy

- **Desired Functional Properties of additional variables:**
  - As lightweight as possible
  - Correlated to the end-to-end GridFTP throughput

- **Networks & Disks:**
  - 70% and 30% of the transfer
  - “regular” NWS (64K) and iostat probes
  - Captures ambient load during transfers
Why Regression?
- Allows us to study the effects of several independent variables on a dependent variable
- GridFTP throughput is dependent on several component load variations

Prediction Model:

Linear Regression:
- $G_l = a + bN$
- Coefficients obtained using previous network & GridFTP values (least squares)
  
  \[
  a = \text{Mean}(G) - b \times \text{Mean}(N) \\
  b = \frac{\sum NG - (\sum N \times \sum G)/\text{size}}{\sum G^2 - (\sum G)^2/\text{size}}
  \]

Polynomial Models:
- Help improve accuracy
- $G_l = a + b_1 N + b_2 N^2$

Multiple Regression:
- Account for several variables
- $G_l = a + b_1 N + b_2 D$
Multivariate Performance

- Results:
  - Within 15% error
  - Moving Avg < G+D Avg < G+N Avg < G+N+D Avg
Summary

- Prediction Architecture for bulk data transfer times
- **Univariate Tools**
  - Prediction within **25% error**
  - **10% reduction** due to file groupings
- **Multivariate Tools**
  - Capturing network and disk factors in the prediction model
  - Several statistical imputation techniques to mitigate sporadic transfers
  - Prediction within **15% error**
- **Synthesizing meaningful information:**
  - In the face of **transient conditions, failure & unavailability**
  - Scheduling is only as good as the **information in hand**
  - **Be Agnostic!!**
- Very pragmatic for a production Grid system
Co-Allocation

**Problem:**
- Typical Internet Download
- Co-Allocated Download

**Merits:**
- Obviates complex server selection
- Decentralization & fault tolerance
- Aggregate BW = $\sum$ Individual Rates
- Alleviates first mile & congestion

*Meta-Message:* CNN crashed during the September 11th “Flash Crowd”!
Download Techniques

- **Static Schemes**
  - Brute-Force
  - History-based
  - Rate differences ignored
  - No Rate Adaptation

- **Dynamic Schemes**
  - Conservative Load Balancing
  - Aggressive Load Balancing
  - Progressive increase & decrease from faster/slower servers
  - Dynamic block sizing
  - Reduces faster servers waiting on slower servers to finish

Mathematical expressions:

\[ A = \sum_{1 \leq i \leq n} B_i \]
Slower Servers

- Results:
  - Dynamic approaches perform better than static ones
  - Speedup significant with file size
  - Small files & lesser blocks; large files & more blocks

![Graph showing bandwidth (BW) vs. file size (File Size)]
Fast & Slow Servers

Results:
- Speedup regardless of server configurations
Summary

- Co-allocation Architecture for bulk data downloads

- Heuristics for distributed downloads
  - Static and dynamic schemes
  - 2x speedup
  - Improvement for all server configurations
  - Small files performed well with lesser blocks
  - Large files performed well with more blocks
  - Performance drop if bottlenecks are shared
  - Aggressive techniques result in reduced wait times for faster servers (up to 40%)
Revisiting the Grid Storage Fabric
(with Xiaosong Ma and Vincent Freeh)

- Evolving HPC Landscape
- Scavenged storage
  - Research Issues
The Evolving HPC Landscape

- Computing fabric for the Grid:  Storage Fabric...??

Meta-Message: Proprietary systems are being replaced with commodity clusters, delivering new levels of performance and availability at dramatically affordable price point.
Imagine “Condor” for Storage…

- Exploit “quantity”
- Address “quality” through aggressive replication
- Revere user autonomy through space reclaims and data evictions
- Grid Awareness
- High-throughput using parallel I/O and informed data placement
Neutron Science Gateway

- External Interface
- Internal Design
  - SNS Services as portlets
SNS Software Development Responsibilities

- Intelligent Control
- Automation
- Diagnostics
- Controls
- Sample Environment
- Instrument
- Acquisition

Real time control

Analysis

Treatment
- Analysis
- Vis
- Scientific
- Intermediate
- Raw
- Data
- Notebook
- Proposal
- Database

Sample environment
- Electronic notebook
- Raw
- Intermediate
- Database

Web Portal to Data, Software, and Analysis

Local and remote users

User Community

SNS Directed

SNS Responsible

High Performance Computing

Materials Simulation
- Simulation
- Vis
- Software repository

Instrument simulation

Local and remote users

Publications
- Documentation
- Raw
Portal aggregation tools (Jetspeed)
- Generic framework with hooks
- Specifies an API for portlets to implement
- Portlets as web services (WSRP), applets, jsp or simple html
- Polymorphic implementations of portlets

From the SNS perspective:
- Modular and maintainable
- New features can be plugged in as portlets as the facility evolves over time
Further Information

- **My Website:**
  - [http://www.csm.ornl.gov/~vazhkuda](http://www.csm.ornl.gov/~vazhkuda)

- **SNS Website:**
  - [http://www.sns.gov](http://www.sns.gov)

Research sponsored by the Laboratory Directed Research and Development Program of Oak Ridge National Laboratory (ORNL), managed by UT-Battelle, LLC for the U. S. Department of Energy under Contract No. DE-AC05-00OR22725.