SRB/iRODS

Leesa Brieger

Renaissance Computing Institute
**Data Grids, as defined by Wikipedia**

A *data grid is a grid computing system that deals with data — the controlled sharing and management of large amounts of distributed data.*

*These are often, but not always, combined with computational grid computing systems.*
Growing Data Demands

• Community-based sharing of resources:
  • biology, astronomy, medical, emergency response...

• Scientific data collection projects
  • Large Hadron Collider, Large Synoptic Survey Telescope, ...

• Growth of digital data assets:
  • scientific & computational growth, digitization of assets, replication and preservation of assets
As pointed out in Wikipedia...

Many applications use the SDSC Storage Resource Broker (SRB) for underlying data grid technology:

– allows widely distributed access to data by many people in many places

– supports collaborative environments for distributed but coordinated scientific research

– see http://www.sdsc.edu/srb/index.php/Main
Storage Resource Broker: Design Goal

Single interface and authorization mechanism to access data across:

• multiple hosts
• multiple Operating Systems
• multiple resource type (UNIX FS, HPSS, UniTree,...)
SRB: Global View

• **Global Logical Name space**
  – UNIX-like directories (collections) and files (data)
  – UNIX-like API and utilities
  – Mapping of logical name to physical attributes - host address, physical path (transparent to users)

• **Single Global User Name Space**
  – Single sign-on
  – No need for UNIX account on every system
  – Robust access control
SRB Architecture

• Federated middleware system

• Client/server model
  – Federation of resource servers with uniform interfaces
  – All servers use same software
  – Robust access control

• MCAT: Metadata CATalog for handling the mapping between logical name and physical attributes
SRB Federation of Servers

- Data Grid has arbitrary number of servers
- Complexity is hidden from users
SRB Performance Enhancement

- **Parallel I/O**
  - For transferring large files
  - Uses multi-threads for data transfer and disk I/O
  - Interface with HPSS’s mover protocol for parallel I/O
  - Parallel third party transfer for copy and replicate
  - One-hop data transfer between client and data resource

- **Bulk Operation**
  - Uploading and downloading large number of small files
  - Multi-threads
SRB Unix-like Commands

- Sls, Spwd, Scp – ls, pwd, cp
- Sput, Sget, Srsync, Schksum – put files into SRB “vaults”, get from SRB, rsynch collections, checksum tracking for error checking
- Sinit, Sexit – initialize and end a session
- Replication (can replicate onto different resources):
  - Sreplicate – replicate a file to a specified resource
  - Sbackupsrb – backup a file to a specified resource
  - SsyncD – Synchronize the replica of a file
Other Client Interfaces

- inQ – Window GUI browser
- Jargon – Java SRB client classes
  - Pure Java implementation
- mySRB – Web based GUI
  - run using web browser
- Java Admin Tool
  - GUI for User and Resource management
- Matrix – Web service for SRB work
For More Info

- See
  http://www.sdsc.edu/srb/index.php/Main_Page
SRB is going over to iRODS (Rule-Oriented Data Structure)

- rule-based data management
- truly Open Source
- preserves SRB functionality and adds more
- first release is out, with partial functionality
- full functionality is still a year or so away
iRODS

• Integrate a rule engine with a data grid

• Map management policies to rules

• Express operations within the data grid as micro-services

• Support rule sets for each collection and user role
Preservation Technology

• Current challenge is the technological management of preservation policies (i.e. implementation)
  – Characterize policies as rules
  – Apply rules on each operation performed by the data grid
  – Manage state information describing the results of rule application
  – Validate that the preservation policies are being followed
Preservation Technology

NARA ERA's list of 854 required capabilities inspired the SRB/iRODS group to define 174 generic rules which can be composed to offer the required functionalities.

iRODS is targeting this, with ~75 rules implemented so far in the current release.

See
http://irods.sdsc.edu/index.php/Main_Page

(leesa@renci.org)