NGS National Grid Service

http://www.ngs.ac.uk

NW-GRID

Data services on the NW-GRID

Richard Hopkins

PPARC

http://www.pparc.ac.uk/

Edinburgh e-Science TOE Training Outreach and Education

http://www.nesc.ac.uk/training

eGee

http://www.eu-egee.org/

Policy for re-use

- This presentation can be re-used for academic purposes.

- However if you do so then please let training-support@nesc.ac.uk know. We need to gather statistics of re-use: no. of events, number of people trained. Thank you!!

2 main types of data services on Grids

- **Simple data files on grid-specific storage**
- **Middleware supporting**
  - **Replica files**
    - make data close computation
    - For resilience
  - **Logical filenames**
  - **Catalogue**: maps logical name to physical storage device/file
  - **Virtual filesystems**, POSIX-like I/O
  - Services provided: storage, transfer, catalogue
  - **Solutions include**
  - **gLite data service**
  - **Globus: Data Replication Service**
  - **Storage Resource Broker**

- **Other data!**

  e.g. ….
  - Structured data: RDBMS, XML databases,…
  - Files on project's filesystems
  - Data that may already have other user communities not using a Grid

- **Require extendable middleware tools to support**
  - Computation near to data
  - Controlled exposure of data *without replication*

- **Basis for integration and federation**

- OGSA –DAI
  - In Globus 4
  - Not (yet...) in gLite

Contents

- OGSA-DAI ←
- SRB
- Grid FTP

What is OGSA-DAI?

- The Open Grid Services Architecture Data Access and Integration project

- Concerned with constructing middleware to assist with
  - access and integration of data
  - from separate data sources
  - via the grid.

- Project conceived by the UK Database Task Force
- Working closely with
  - the Open Grid Forum DAIS-WG
  - the Globus team.

Motivation

- OGSA-DAI is motivated by the need for:
  - Extensible framework
    - for easily **integrating** data resources on to Grids.
  - Data discovery from previously unknown locations.
  - Different types of data models
    - from distributed data resources
    - to be easily **integrated** to Grid applications.
  - Uniform interfaces for data access
  - **integration** of data from various sources
    - to obtain the required information.
  - …

OGSA-DAI Provides

- Access to and updating of data resources
- Exposure of Data Resources to the Grid
- Additional data manipulation functionality at the service level
- Uniform access to disparate, heterogeneous data resources
  - Does not hide underlying data model
- Data resources exposed through services
  - Clients interact with these services

Design Principles – I

- Efficient client-server communication
  - Minimise where possible
  - One request specifies multiple operations
- No unnecessary data movement
  - Move computation to the data
  - Utilise third-party delivery
  - Apply transforms (e.g., compression)
- Build on existing standards
  - Fill-in gaps where necessary

Design Principles – II

- Do not hide underlying data model
  - Users must know where to target queries
  - Data virtualisation is hard
- Extensible architecture
  - Modular and customisable
  - e.g., to accommodate stronger security
- Extensible activity framework
  - Cannot anticipate all desired functionality
  - Activity = unit of functionality
  - Allow users to plug-in their own

Why OGSA-DAI?

- Can embed additional functionality at the service end
  - Transformations, compressions, third party delivery
  - Extensible activity framework
- Avoiding unnecessary data movement
- Common interface to heterogeneous data resources
  - Relational, XML databases, and files
- Language independence at the client end
  - Do not need to use Java
- Platform independence
  - Do not have to worry about connection technology, drivers, etc
- Standards Based –
  - *DAIS - Database Access and Integration Services* - Working Group of the *Global Grid Forum* (GGF)
  - formulating standards for database access and integration services
  - Working closely with OGSA-DAI
  - OGSA-DAI expected to be a reference implementation of the final standard

Contents

- OGSA-DAI
- SRB ←
- Grid FTP

What is SRB?

Storage Resource Broker

- a software product developed by the San Diego Supercomputing Centre (SDSC).
- Allows users to access files and database objects across a distributed environment.
- Actual physical location and way the data is stored is abstracted from the user
- Allows the user to add user defined metadata describing the scientific content of the information
- Scales Well (many millions of files, terabytes)
- Supports Multiple Administrative Domains / MCATs (srbZones)
- Includes SDSC Matrix: SRB-based data grid workflow management system to create, access and manage workflow process pipelines.

What is SRB?

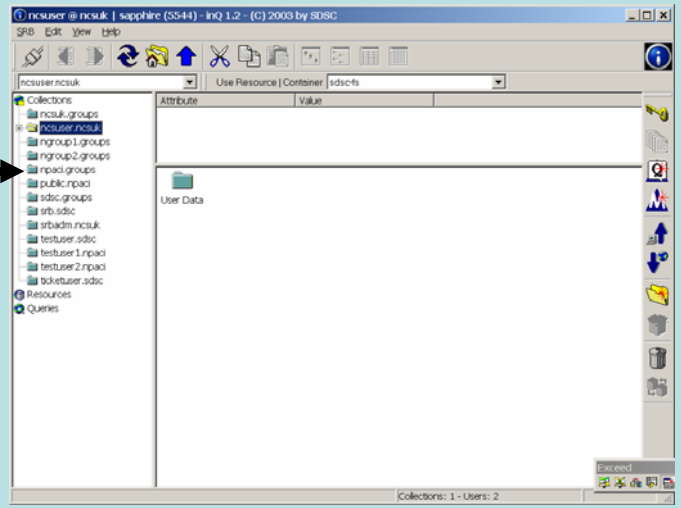
An integrated solution which includes:

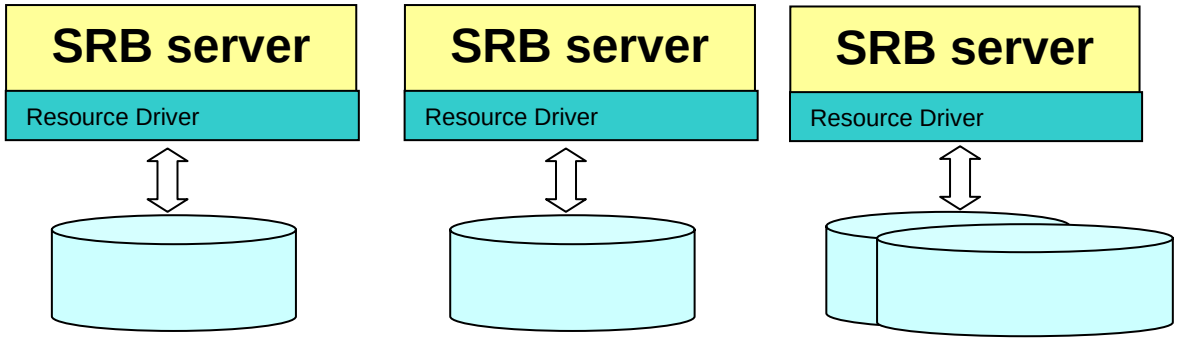
- a logical namespace,
- interfaces to a wide variety of storage systems,
- high performance data movement (including parallel I/O),
- fault-tolerance and fail-over,
- WAN-aware performance enhancements (bulk operations),
- storage-system-aware performance enhancements ('containers' to aggregate files),
- metadata ingestion and queries (a MetaData Catalog (MCAT)),
- user accounts, groups, access control, audit trails, GUI administration tool
- data management features, replication
- user tools (including a Windows GUI tool (inQ), a set of SRB Unix commands, and Web (mySRB)), and APIs (including C, C++, Java, and Python).

A Virtual Fle System

User sees a virtual filesytem:

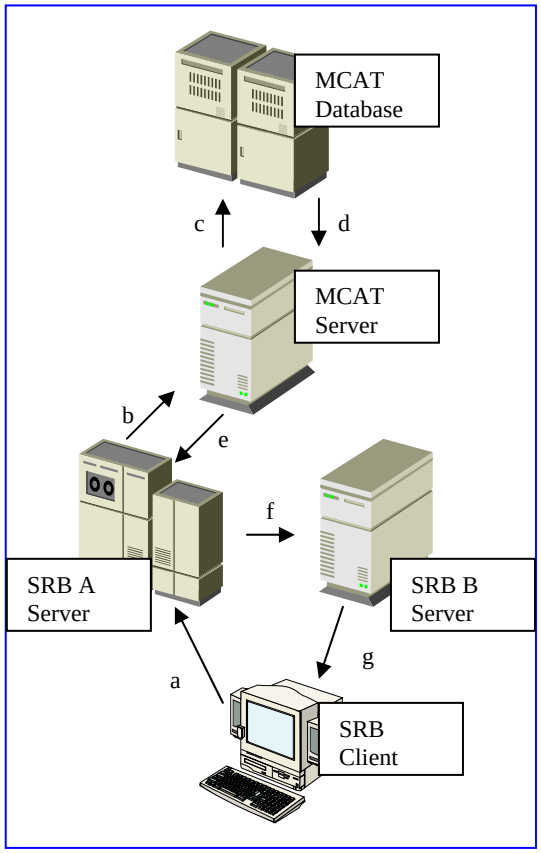
- Command line (S-Commands)
- MS Windows (InQ)
- Web based (MySRB).
- Java (JARGON)
- Web Services (MATRIX)





Filesystems in different administrative domains

How SRB Works



- 4 major components:
  - The Metadata Catalogue (MCAT)
  - The MCAT-Enabled SRB Server
  - The SRB Storage Server
  - The SRB Client

SRB on the NW-GRID

- SRB provides NW-GRID users with
  - a virtual filesystem
  - Accessible from all core nodes and from the “UI” / desktop
  - (will provide) redundancy – mirrored catalogue server
  - Replica files
  - Support for application metadata associated with files
  - fuller metadata support from the “R-commands”
- Practical Overview
  - Use of the Scommands
    - Commands for unix based access to srb
    - Strong analogy to unix file commands
  - Accessing files from multiple (two) sites

Contents

- OGSA-DAI
- SRB
- Grid FTP ←

What is GridFTP?

- Data transfer **protocol**

  secure robust fast efficient

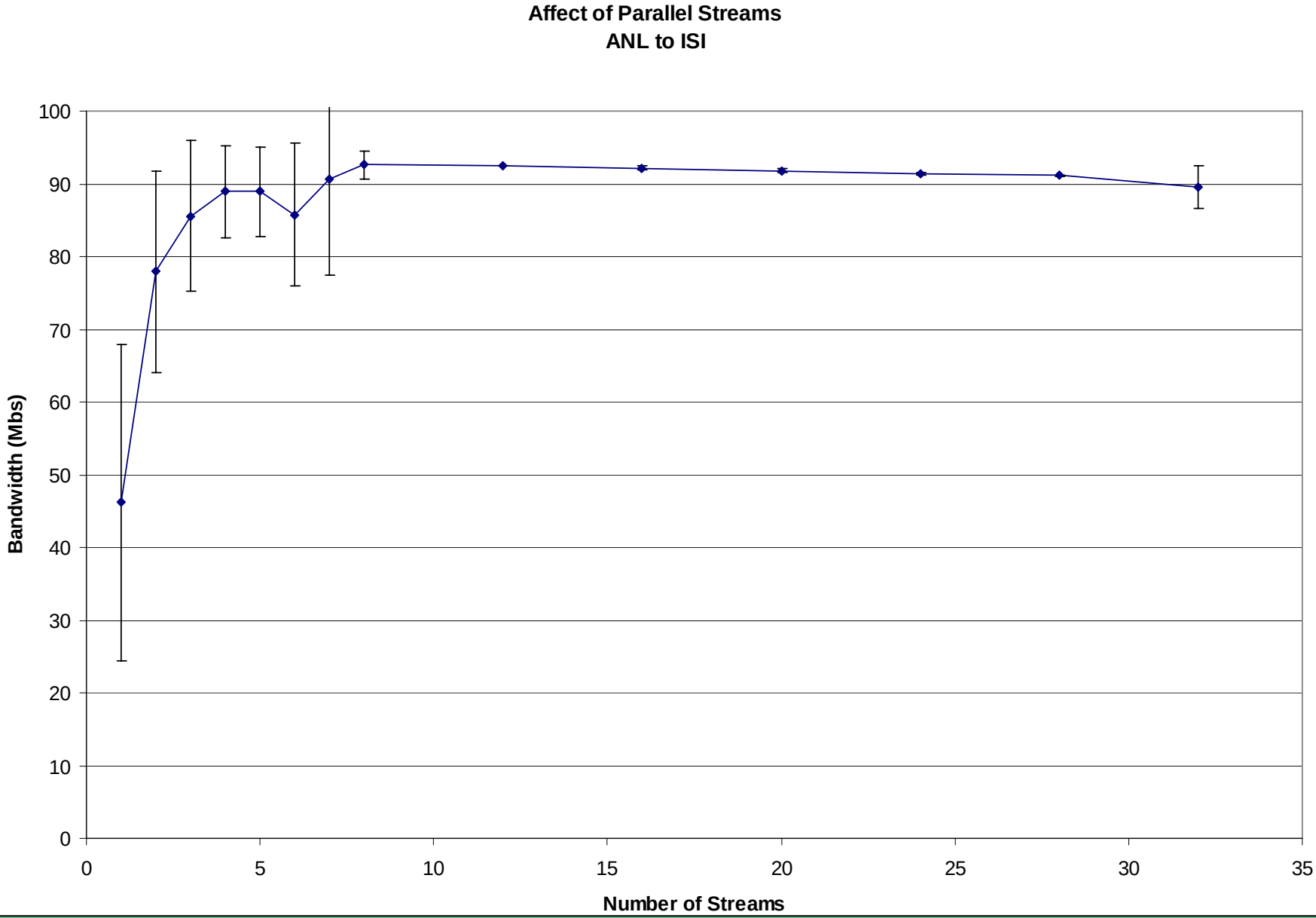
  standards based widely accepted

- Multiple independent implementations can interoperate
  - This works. Both the Condor Project at Uwis and Fermi Lab have home grown servers that work with ours.
  - Lots of people have developed clients independent of the Globus Project.
- Globus also supply a reference implementation:
  - Server
  - Client tools (globus-url-copy)
  - Development Libraries

Parallelism and Striping

- Parallelism
  - multiple TCP Streams between two network endpoints
- Striping
  - Multiple pairs of network endpoints participating in a single logical transfer (i.e. only one control channel connection)
- Multiple nodes work together and act as a single GridFTP server
- An underlying parallel file system allows all nodes to see the same file system
  - must deliver good performance (usually the limiting factor in transfer speed)
  - I.e., NFS does not cut it
- Each node then moves (reads or writes) only the pieces of the file that it is responsible for.
- This allows multiple levels of parallelism, CPU, bus, NIC, disk, etc.
  - Critical if you want to achieve better than 1 Gbs without breaking the bank

Parallel Streams



TeraGrid Striping results

- Ran varying number of stripes
- Ran both memory to memory and disk to disk.
- Memory to Memory gave extremely high linear scalability (slope near 1).
- Achieved 27 Gbs on a 30 Gbs link (90% utilization) with 32 nodes.
- Disk to disk - limited by the storage system, but still achieved 17.5 Gbs

BWDP

- TCP is reliable, so it has to hold a copy of what it sends until it is acknowledged.
- Use a pipe as an analogy
- I can keep putting water in until it is full.
- Then, I can only put in one gallon for each gallon removed.
- You can calculate the volume of the tank by taking the cross sectional area times the height
- Think of the BW as the cross-sectional area and the RTT as the length of the network pipe.

globus-url-copy: 1

- Command line scriptable client
- Globus does not provide an interactive client
- Most commonly used for GridFTP, however, it supports many protocols
  - gsiftp:// (GridFTP, historical reasons)
  - ftp://
  - http://
  - https://
  - file://

globus-url-copy: 2

- globus-url-copy [options] srcURL dstURL

Important Options

- -p (parallelism or number of streams)
  - rule of thumb: 4-8, start with 4
- -tcp-bs (TCP buffer size)
  - use either ping or traceroute to determine the Round Trip Time (RTT) between hosts
  - buffer size = BandWidth (Mbs) * RTT (ms) *(1000/8) / P
  - P = the value you used for –p
- -vb if you want performance feedback
- -dbg if you have trouble

Other Clients

- Globus also provides a Reliable File Transfer (RFT) service
- Think of it as a job scheduler for data movement jobs.
- The client is very simple. You create a file with source-destination URL pairs and options you want, and pass it in with the –f option.
- You can “fire and forget” or monitor its progress.

Further Information

- OGSA-DAI
  - http://forge.gridforum.org/projects/dais-wg.
- SRB
  - http://www.sdsc.edu/srb/ and http://datacentral.sdsc.edu/user_guide.html
  - For a full SRB tutorial, at NIEES last January, see: http://www.niees.ac.uk/events/srb2006
  - http://homepages.nesc.ac.uk/~gcw/NWGrid/srb.html