

SRB in Action

Dr Michael Doherty

e-mail: m.doherty@rl.ac.uk

Data Management Group, CCLRC e-Science Centre, Rutherford Appleton Laboratory,
Chilton, Didcot, Oxfordshire, OX11 0QX, United Kingdom.

*Kerstin Kleese, Mark Williams, Bonny Strong, Lisa Blanshard, Rick Tyer, Tim Folkes,
David Corney, Andrew Sansum, Martin Bly, Nick White*

Abstract

The Storage Resource Broker (SRB) is a data management product developed by the San Diego Supercomputing Centre (SDSC). This software is used for distributed data management and can form the basis of a data grid. This paper will cover an overview of SRB and will then highlight how CCLRC has implemented SRB software across a number of UK e-Science projects to build data grids. Further to this, the session will describe how the Data Management Group at CCLRC are becoming involved in the development of this software and are helping shape the next generation of SRB to meet the needs of UK e-Science.

Introduction to SRB

The Storage Resource Broker (SRB) is a software product developed by the San Diego Supercomputing Centre (SDSC) [1]. It allows users to access files and database objects seamlessly across a distributed environment. In simple terms, the actual physical location and way the data is stored is abstracted from the user, who is presented with an interface similar to a regular file system. Further to this, the system allows the user to add user defined metadata describing the scientific content of the information to resources managed by the system, such that meaningful queries can be performed across data in multiple sites. Access can be via graphical or command line based tools. SRB also includes a host of management features that include data replication, access control and storage optimization. The SRB system is comprised of 4 major components:

- The Metadata Catalogue (MCAT) database

- The MCAT SRB Server
- The SRB Server
- The SRB Client

Each SRB system must consist of a single MCAT database and one or more SRB Servers and SRB clients. This is known as the SRB Domain. Note there has been some confusion in the community that SRB runs off a single world wide MCAT database. This is not true. Anyone wishing to set up an SRB is free to create their own MCAT database and hence SRB Domain. Software and instructions on how to do this can be obtained from SDSC home page [2].

SRB Components:

The key software components identified above are described in more details below. Figure 1 outlines the major components and can be used as a guide in the descriptions that follow.

MCAT Database:

The MCAT database is a metadata repository that provides a mechanism for storing information used by the SRB system. This includes both internal system data required for running the system and application (user) metadata regarding data sets being brokered by SRB. SRB makes a clear distinction between these two types of data. An example of system metadata might be the physical location of a particular file and how to access it. An example of application metadata may be the energy and luminosity of X-ray from a synchrotron.

The MCAT database is hosted within a Relational Database Management System such as Oracle, DB2 or PostgreSQL. Thus to be successful in an SRB implementation, a professionally managed database is required.

MCAT SRB Server

At least one SRB Server must be installed on the node that can access the MCAT database. This is known as the MCAT SRB Server. The MCAT SRB Server performs the following data management operations in conjunction with the MCAT database:

- Stores metadata on Data sets, Users, Resources and Access Methods.
- Maintains replica information for data and containers. Containers provide a method of storing related small data items physically close to each other to optimize storage.
- Provides “Collection” abstraction for data. A Collection is a way of logically grouping data together across multiple physical sites analogous to a directory or folder in a normal file system.
- Provides a “Global User” name space and authentication.
- Provides Authorization through Access Control Lists and tickets.
- Maintains audit trail on data and collections.
- Provides Resource Transparency to enable logical resources.

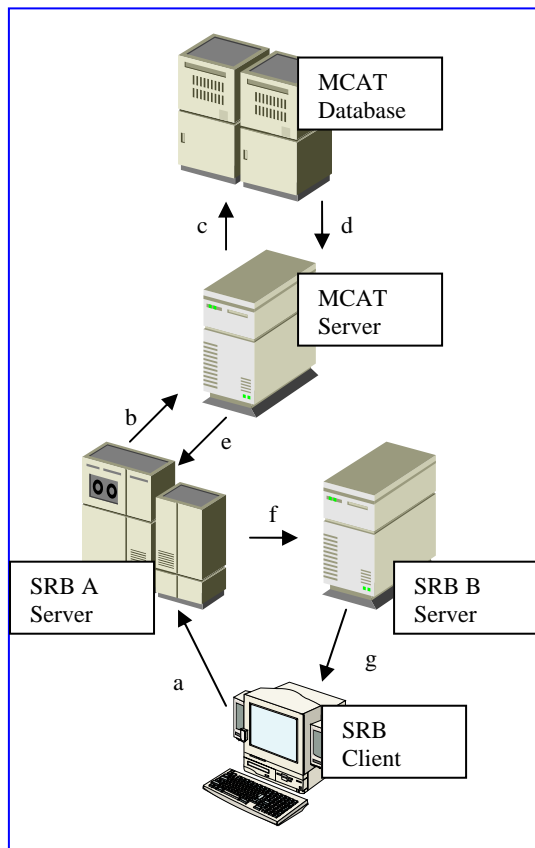


Figure 1: SRB Components

Each of these topics is a large scale subject itself and can not be discussed here fully. Further information can be found at the SRB home page [2].

SRB Server

The SRB Server is a middleware application that accepts requests from clients and obtains the necessary data sets. It queries the MCAT SRB Server to gather information on datasets and supplies this back to the SRB client. The SRB server can operate in a

“federated mode”, whereby it can request another SRB server to obtain the necessary data on its behalf. The data is then transparently passed back to the client. This is shown in Figure 1 and can be explained as follows: SRB Client contacts the local SRB Server A and requests a file. This is looked up via the MCAT SRB Server and MCAT database and then passed back to SRB Server A. As the file is located on SRB Server B, SRB Server A sends the request to SRB Server B who then services the client directly.

SRB Client

This is an end user tool that provides a user interface to send requests to the SRB server. There are 3 main implementations of this: command line, MS Windows (InQ shown in Figure 2) or web based (MySRB). A recent addition is the MySRB server component. This allows access to storage via a thin client such as Internet Explorer, Netscape or Mozilla.

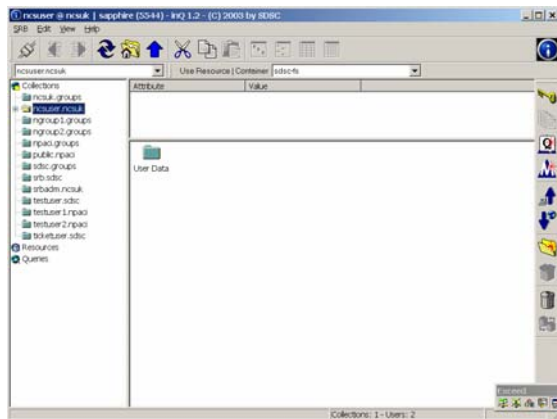


Figure 2 InQ Interface

The MySRB server is actually an application server middleware component that runs as a library in an Apache Web Server. This acts as a client to service multiple thin client sessions and is based on CGI scripts. This option is proving popular as it

allows access to the system from anywhere with an internet connection. Perhaps the most recent significant development is that SRB 2.1 includes a Web Services interface in the “Matrix” component. This effectively allows you to build an open distributed application that utilizes SRB as a storage mechanism.

An SRB programmers API is also available for programmers who wish to develop there own interfaces to SRB. C is currently supported and a new java interface was introduced in SRB 2.1

SRB Internal Architecture

SRB uses an internally layered architecture. This is shown in Figure 3. This allows various system components to be abstracted for portability, which is described here. Further to this, the whole system is based around the POSIX standard. The system can be abstracted into 2 major logical entities: SRB and MCAT.

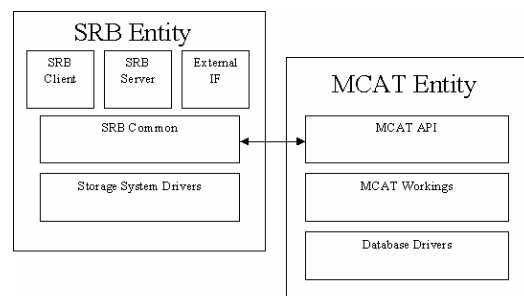


Figure 3: SRB Architecture

SRB Entity

The SRB entity comprises of an initial layer that is specific to the client, server or external interface (such as Objectivity OO Database). The specifics of each component are different at this level. These sit on top of a common SRB layer. This implements all core functionality and is shared by all components in the layer above. Below this is the driver layer. The driver layer is used to manipulate

data in all the storage systems that can be brokered by SRB. This includes file system, tape archives and RDBMS storage. Hence to bring a new storage system into SRB, all that needs to be done is for a new driver to be written. A driver template exists and the largest implementation to date is 700 lines of C code. These implement the 16 standard calls that implement the driver layer such as copy, move, delete and create.

MCAT Entity

The top layer of the MCAT entity is the Interface API. This is a function library for all MCAT operations. Below this is the MCAT workings layer. This layer abstracts the underlying logical database schema into a common set of objects that may be manipulated by the common API. Hence these are the common objects and methods that MCAT must act on. This sits on top of the database driver layer. This layer is a specific implementation necessary to manipulate each RDBMS. The physical schema for each RDBMS can be different in order that functionality of each platform may be exploited e.g. Sequences may be used in Oracle, however there is no such analogy in DB2, so an alternative must be used.

Internal Communication

In terms of overall communication, the common SRB layer of the SRB entity communicates directly with MCAT interface API.

SRB in CCLRC

The Data Management Group in CCLRC started working with SRB in November 2002 after a fact finding mission to the USA. There was an immediate requirement for a storage based product that allowed the addition of searchable metadata within CCLRC. The specific requirement was initially

for use with the CCLRC data portal [3], however other projects subsequently developed.

While other data storage projects, and specifically GIGGLE [4] show much promise (and should be monitored), it was felt that only SRB offered a production quality system that could be implemented in the time scales available. Having selected the SRB system, initial testing began in January 2003, with full scale systems becoming available in April 2003. This was tied to the introduction of the CCLRC e-Science Database Service, described elsewhere in these proceedings.

While several test systems exist, the main production systems described here are those using the CCLRC Database Service described elsewhere in these proceedings [5].

CMS and SRB

The largest project using CCLRC SRB services at present is the CERN CMS experiment. The CMS detector is a particle physics experiment based at CERN. This will be located on part of the Large Hadron Collider (LHC) particle accelerator. The CMS detector is one of the largest international scientific collaborations in history. As of February 2003 there were 2300 people working for CMS, 1940 of which were scientists and engineers. These people come from 159 institutes in 36 countries, spanning Europe, Asia, the Americas and Australasia. CCLRC as a whole has been involved in this project from an early stage for 2 major reasons: Firstly CMS make use of CCLRC computational clusters for detector simulations and secondly, CMS will need to make use of the CCLRC Atlas Data Store (ADS) for the large volumes of data they will produce. Note the actual data taking from the detector will not start until 2007.

SRB is being used by CMS now for the following specific reason: CMS are currently entering Data Challenge 2003 (DC2003) which aims to demonstrate that the collaboration is capable of dealing with the large amounts of data that the real experiment will produce. This includes both data management, data flow and data storage. To verify this, the whole experiment is simulated in software first. DC2003 commenced on 1 July 2003 and will run until September 2003. For this project so far, CCLRC Data Management Group and ADS groups contributed on several fronts:

- Developed a custom SRB storage driver for the ADS using the SRB architecture as described above.
- Set up an SRB infrastructure for transferring files from the computational clusters to the data store.
- Provided an MCAT server for the whole CMS experiment. This is hosted with the CCLRC Data Management Group Database Service.
- Provided general SRB support to the DC2003 via University of Bristol.

A crucial factor in this project was to develop an SRB driver for the ADS (see below).

CCLRC SRB Service is currently managing 73,000 files registered across 13 sites and 24 servers worldwide for the CMS experiment. This number is growing daily. In addition to the DC2003 work, these sites are managing and exchanging data via the RAL MCAT server and database.

ADS and SRB

The Datastore currently has an on-line, nominal capacity of 1 Petabyte, based

on an STK 9930 (PowderHorn) tape robot. The ADS team are currently migrating from IBM 3590 (10GB) tape drives to STK 9940B (200GB) drives. The robot is accessed by a farm of servers with associated disk cache, giving fast access to and from tape. A master server manages all of the integrated Datastore software and hardware. Though the nominal capacity of the Datastore is 1 Petabyte, tapes containing infrequently used files can be easily removed from the robot, thus making the total capacity virtually unlimited.

Currently access is via a set of home grown utilities known as SYSREQ and VTP. These present the ADS as a network mounted tape storage system to host systems via a command line interface. The system is not hierarchical and the custom software must be installed. SRB offered CCLRC a new way to present access to the ADS. Not only did this offer the opportunity to provide a more user friendly access to the system, it would also allow the system to become part of a much wider storage resource network. To this end the ADS and Data Management Group in CCLRC set about integrating SRB and ADS. It is intended that many projects will use this and the CMS usage is explained below.

The use of such a large scale and versatile tape storage facility is crucial to CMS for the following reasons. CMS simulates both physics events and detector responses to these events. This is a continual process and the data volumes involved are much larger than can be accommodated on disk systems. The only way to achieve this is to stage the current data on disk and then move processed data to a large scale storage system.

In the UK, this work is carried out at Imperial College in addition to CCLRC.

In DC2003, CMS have broken the work down into 2 distinct steps i.e. the physics events are all generated as one step and the detector responses generated as a second step. The data produced by each step is larger than the disk space available. CMS have therefore taken the following approach: Having generated a batch of simulated physics events, data is written to disk. CMS are then able to use the bulk upload facilities in SRB to move data to and from local disk into SRB space and specifically the files are loaded into the ADS. CMS then use the disk to store the results of more physics events. Having completed the event generation step for DC2003, these event files are incrementally later pulled back out of the ADS via SRB for use in simulating the detectors. These final results are then loaded back into the ADS. Any of the collaborating institutes can then pull the data out of ADS to either their local file systems or their own SRB Servers.

E-Minerals Mini-Grid

As part of the NERC Environment from the Molecular Level (e-Minerals) project, a mini-grid [6] has been set up to provide the necessary infrastructure and middleware to facilitate cooperative working throughout the distributed virtual organisation (VO). The E-Minerals Mini-Grid consists of the following components:

- An e-Minerals data portal for discovery of environmental simulation studies and associated input and output files
- A High Performance Computing (HPC) portal for access to machines in e-Minerals mini grid.
- A Metadata insertion tool for data in the Mini-Grid.
- An SRB domain and resources for data storage.

The metadata is held in the CCLRC e-Science Database Service and retrieved using the data portal. The actual data files are kept in SRB and their locations referenced in metadata

This mini-grid includes customised fully integrated instances of SRB as part of a data grid. CCLRC Data Management Group are providing both an MCAT database and the MCAT SRB server. In addition to this, CCLRC Data Management Group are providing installation and configuration expertise at member institutions. CCLRC Data Management Group plan to install SRB resources at Cambridge, Reading and Bath for storage of files that can be shared across the VO.

Projects in Development

All of the following projects are either using or investigating SRB as a the basis of data grids. CCLRC Data Management Group are providing both SRB infrastructure and expertise.

- E-Materials Project: A mini grid similar to E-minerals above.
- National Crystallographic Service (Southampton University): For both data management and storage in the ADS. Test system already set up.
- British Atmospheric Data Centre (BADC) archive. For both data management and storage in the ADS. Test system already set up.
- NERC Data Grid. Details to be confirmed.

SRB Futures

There are 3 major areas that CCLRC are commencing work with in collaboration with SDSC.

- Web Services
- High Performance Testing

- Database Advice

Web Services

The first is in the area of Web Services. As mentioned above, SRB 2.1 contains a new component called “Matrix”. This component is currently in a beta stage and the CCLRC Data Management Group have been testing this software. We now have a fully functional web services client developed and can access all standard features of SRB. The benefit of a web services interface is that it allows other applications and most notably web and grid based portal applications to utilize SRB. Thus once can effectively create a virtual application that is distributed across services. The Data Management group is especially interested in using this feature to integrate with CCLRC Data Portal. The CCLRC data portal is a piece of application middleware that allows users to find scientific studies and associated datasets across all CCLRC data holdings. This is achieved by using XML to wrap the returned metadata from each of CCLRC facilities. Currently, once data has been identified, it can be downloaded via Grid FTP. However the intention is to expand this to encompass SRB managed resources. Ultimately, it may be possible to create a data portal wrapper for SRB. This would allow the Data Portal to query SRB application metadata directly.

High Performance Testing

This work will involve testing SRB under extreme environments such as fast networking and high performance servers. CCLRC has a number of large scale computing facilities that will allow us to test future versions. Crucially, this will involve work on the new version of SRB to be released later this year (version 2.2). The key

component of this release is a distributed MCAT database. The whole system at present relies on a single MCAT database. This could be considered as a single point of failure. However, there are some limited replication features for this at present. The current features allow a series of transactions to be captured on the master MCAT database and replayed at a remote database. The remote database would be part of a separate SRB environment with its own MCAT. While vendor specific replication, such as Oracle Advanced Replication could be used to move data, this is not desirable for 2 reasons

1. SRB has to work across a number of databases and vendor specific solutions are not open.
2. It may be desirable for data to be distributed for security issues.

It could be possible that two independent domains wish to work with each other. Sensitive or private data could be specific to the domain, however there may be a requirement to allow restricted access.

The new version of SRB will address these points by allowing multiple MCAT databases to work independently and communicate with each other. The data management groups has been active in providing design input into this process.

Database Advice

The Data Management group has a background in database applications and management. As mentioned previously, it is crucial that SRB be run under a well managed database. This can be anything from basic setup to database tuning. The group has already produced an SRB Install note for Oracle databases, which is now part of the SRB documentation distribution (see “Storage Resource Broker (SRB) with

MCAT Install Note” in the documentation). The group fully intend to work with SDSC in notes and guides to help users implement SRB successfully

Conclusions

CCLRC have successfully implemented SRB across a number of projects. We believe this is the only production ready system at this moment in time. The group are now actively collaborating with SDSC and aim to help others in the UK e-Science program bring SRB into their projects.

References:

[1] The SDSC Storage Resource Broker, Chaitanya Baru, Reagan Moore, Arcot Rajasekar, Michael Wan, Proc. CASCON'98 Conference , Nov.30-Dec.3, 1998, Toronto, Canada.

[2] SRB Home Page:
<http://www.npaci.edu/DICE/SRB>

[3] The CCLRC Data portal, Glen Drinkwater, Kertsin Kleese, Shoaib Sufi, Lisa Blanshard, Ananta Manandhar, Rik Tyer, Kevin O'Neill, Michael Doherty, Mark Williams, Andrew Woolf, http://www.e-science.clrc.ac.uk/documents/projects/dataportal/DataPortal-WebServices0603_Glen.doc

[4] Giggle: A Framework for Constructing Scalable Replica Location Services. Ann Chervenak, Ewa Deelman, Ian Foster, Leanne Guy, Wolfgang Hoschek, Adriana Iamnitchi, Carl Kesselman, Peter Kunszt, Matei Ripeanu, Bob Schwartzkopf1, Heinz Stockinger, Kurt Stockinger, Brian Tierney, Proc GGF5 Conference, 21-24 July 2002, Edinburgh, Scotland.

[5] Database Clusters for e-Science, Michael Doherty, UK e-Science All Hands Meeting, 3-5 September 2003, Nottingham, England

[6] The E-Minerals Mini Grid, Kerstin Kleese, Lisa Blanshard, Richard Tyer, May 2003,
http://ws1.esc.rl.ac.uk/documents/staff/kerstin_kleese/e-min_intro.doc