

Managing Large Distributed Data Sets using the Storage Resource Broker

Reagan W. Moore
San Diego Supercomputer Center
9500 Gilman Drive, MC-0505
La Jolla, CA 92093-0505
moore@sdsc.edu
Telephone: 858 534 5073

Abstract:

High performance computing can now be applied to collections of distributed simulation and observational data. A typical collection may have over a million files stored on multiple types of storage systems and at multiple sites. The observational data may be in a repository at a different site from the simulation output, and may be controlled by a separate administrative domain. The Storage Resource Broker data grid enables the formation of shared collections that span administrative domains and multiple storage resources. Uniform access mechanisms, uniform global names, and uniform access controls are applied to the data, no matter where they are located. This makes it possible to automate data analyses across the distributed data, removing the need to aggregate data into a single location. The Storage Resource Broker is generic infrastructure that supports the requirements of multiple data management applications, ranging from data grids for shared collections, to digital libraries for data publication, to persistent archives for data preservation, to real-time data collection federation.

Introduction:

The creation of a shared collection makes it possible to apply management control on shared data. The academic community is implementing data sharing environments that are used to promote collaborative research. The data sharing environments are built on data grid technology, software middleware installed at each site participating in the collaboration. The shared collections are assembled by collaborators residing at multiple institutions. The data grid manages all interactions with the shared data, including access controls, persistent state information, and transport over wide-area-networks.

The Storage Resource Broker (SRB) data grid is used to build shared collections out of digital entities that are located at multiple sites across multiple administrative domains [1,2]. The digital entities may be files, directories, database tables, SQL commands, or URLs. The shared collection organizes the digital entities into collection (or directory) hierarchies. Each digital entity may independently be assigned descriptive metadata attributes. The metadata can be queried to discover relevant data. The data can then be retrieved from the remote site and manipulated within a preferred client interface or web browser.

The SRB provides persistent global name spaces for the names of the digital entities, the curators of the shared collection, the storage resources, and even the metadata attributes associated with each file [3]. The result is an environment in which the files may be moved from site to site without having to worry about the file name changing,

and access controls that are set on the files and user-defined metadata remain unchanged as well.

The data grid makes it possible to discover a digital entity without having to know its name or its storage location. The user can retrieve the digital entity without having local technology that can execute the protocol that the remote storage system understands, and can manipulate the digital entity from a preferred access client. It is possible to construct an environment in which all the properties of the shared collection are managed independently of the choice of storage system or database. This capability is called infrastructure independence, and is the essential design feature needed to manage technology evolution [4]. Data grids use infrastructure independence to ensure that data can be uniformly managed with strong access controls, even when the files are distributed across multiple types of storage systems.

Storage Resource Broker Data Grid

The SRB data grid has been under development for ten years at the San Diego Supercomputer Center. The SRB technology has been widely deployed in support of academic projects and US federal agencies. More than 200 academic institutions and federal agencies have downloaded the SRB technology, with about half of the sites supporting internationally shared collections. Examples of federal agencies using the technology are listed in Table 1.

Table 1. Federal agencies using/testing the SRB data grid

Institution	Project
Environmental Protection Agency	EPA Data Grid initiative
Environmental Protection Agency	P2Tools Design & Development Team Leader
Library of Congress	Data storage project
LLNL	Climate change digital Library
NARA	Transcontinental Persistent Archive
NASA Ames	Information Power Grid
NASA Goddard	EOSDIS Distributed Active
NASA Goddard Space Flight	Storage integration
NAVY	SPAWAR
NIH	National Cancer Institute data grid
NOAO	NOAO astronomy data grid
ORNL	Photon light source data grid
Pacific Northwest National Laboratory	BioPilot digital library
SAIC/NASA	Atmospheric Sciences Data grid
US Army Research Laboratory	Rapid Unified Generation of Urban Databases (RUGUD)
USGS	Bedford Oceanography,Canada

The applications range from construction of a data grid to enable sharing of data across multiple sites; to managing remote storage systems; to creation of digital libraries for publishing data; to creation of preservation environments that manage technology evolution; to management of data on an intercontinental scale.

Academic institutions within the US also are using or testing the SRB in support of multiple scientific disciplines.

Table 2: Academic institutions in the US that have downloaded SRB technology

Caltech	National Virtual Observatory
City University of New York	NSDL (SRMA)
Colorado University	Cires/Cism
Cornell University	Fedora project
Drexel University	Digital library project
Howard University	CAHPC, Deputy Director
Indiana University	Digital Library Program
Kentucky Dept Libraries & Archives	PAT project
MIT	Integration of Dspace , SRB
NYU Libraries	Web-at-Risk NDIIPP (CDL)
Ohio State University	Ohio State Univ.
Oregon State University	Computer Science
Penn State University	WUN (ITS Group)
Pittsburgh Supercomputing Center	ETF project
Purdue University	TeraGrid project
SDSU	SDSU portal GA project
SLAC/Stanford	BaBar high energy physics
Space Telescope Science Inst.	Baltimore, MD
Stony Brook, Univ of NY	Stony Brook, Univ of NY
Texas A & M	Multiview Storage System
UC Merced	CUAHSI/ DLS
UCAR	NCAR Visualization
UCD	DBIS Lab
UCLA	Chemistry/Biochemistry
UCSC	UC at Santa Cruz
UCSD/NCMIR	TeleScience
UCSF	VA Medical Center, Workflow Project
University of Buffalo	NEES project
University of Florida	UF Research Grid(HPS)
University of Hawaii	Institute for Astronomy
University of Kansas	Bioinformatics
University of Maryland	Department of Computer Science, DataCutter
University of Michigan	Uof Michigan,CAC department
University of Minnesota	NEES project
University of New Mexico	LTER
University of Pittsburgh	Library archive
University of Texas, TACC	Microscopic Imaging
University of Washington	Streaming Technologies(WUN)
University of Wisconsin	Condor Project
USC	Southern California Earthquake Center
Washington University	Department of Anatomy and Neurobiology
Woods Hole Oceanographic Institution	Woods Hole Oceanographic Inst
Yale University Library	Yale University Library

Table 3. Overseas Sites Using the Storage Resource Broker Data Grid

British Antarctic Survey, UK	Data management project
Cambridge e-Science Center, UK	eMinerals
Cardiff University, UK	Welsh e-Science Centre
Chinese Academy of Science	Visualization in scientific computing
CINECA, Bologna, Italy	HPC-EUROPA project
Cranfield University	UK (Silsoe)
CRS4, Italy	Bio-medicine
CSIRO, Australia	Bureau of Meteorology
Data Storage Institute, Singapore	Quality of Storage service
French National Center	Enabling Grids for E-science
GeoForschungsZentrum, Germany	Telegrafenberg, Germany
Griffith University, Australia	Research Computing Services, Nathan campus
ISREC, Switzerland	Swiss Institute for Exp. Cancer Research
KEK, Japan	High Energy Accelerator
Konkuk University, Korea	Korean national grid
Leiden University, The Netherlands	LIACS(Leiden Inst. Of Comp. Sci)
Library of Chinese Academy	Beijing, China
Macquarie University	Sydney, Australia
Max Planck Institute, Netherlands	Netherlands
Melbourne, Australia	APAC Grid Project
Monash University, Australia	Microsoft SQL Server(MCAT port)
Nara Institute of Science & Tech, Japan	Preservation environment
National University of Mexico	UNAM Grid
National University, Singapore	Bio data grid
Osaka University, Japan	Virtual Tissue Bank
Queen's University, UK	Belfast e-Science Centre
SARA	Computing and Network Services, Netherlands
Sejong University, South Korea	IT Department
Sickkids Hospital, Toronto	Toronto, Canada
South Australian Advanced Computing	APAC project
Swiss Federal Institute (EPFL)	Switzerland
Taiwan university, Taipei Taiwan	Protein structure prediction
Tokyo Institute of Technology	NEES project
Trinity College, Ireland	HPC-Europa NA3
University of Amsterdam	Virtual Laboratory for eScience
University of Bergen	Parallab(HPC-EUROPA project)
University of Bologna	Grid for Logistic Optimization, CS Dept.
University of Bristol, UK	Physics Labs
University of Calgary	Research Repository with DSpace
University of Cambridge	UK e-Science
University of Edinburgh	University of Edinburgh
University of Genoa, Italy	Laboratory for Bioimages and Bioengineering
University of Hong Kong	Computer Centre (Data Grid)
University of Leeds, UK	School Computing

University of Liverpool	Dept. of Computer Science
University of Manchester, UK	WUN data grid
University of Oslo	archiving scientific data (WUN)
University of Oxford	LHC Computing Grid
University of Queensland, Australia	The Earth Systems Science Ctr
University of Sao Paulo, Brazil	Instituto do Coracao
University of Sheffield, UK	White Rose Grid
University of Southampton, UK	GRIA and SRB
University of Technology Sydney	APAC Data grids
University of the West Indies	Kingston Jamaica (Jgrass)
University of Zürich	Computational Chemistry
York Univ, UK	WUN data grid
ZIB, Germany	German DGrid

Disciplines that are managing shared collections include Astronomy, Biology, Chemistry, Cognitive Science, Cosmology, Ecology, Education, Engineering, Environmental Science, Global Climate Change, High Energy Physics, Medicine, Neuroscience, Oceanography, and Seismology.

Table 4. Shared collections managed at the San Diego Supercomputer Center

Date	5/17/02		6/30/04			9/18/06		
	GBs of data stored	1000's of files	GBs of data stored	1000's of files	Users with ACLs	GBs of data stored	1000's of files	Users with ACLs
Data Grid								
NSF / NVO	17,800	5,139	51,380	8,690	80	107,903	15,166	100
NSF / NPACI	1,972	1,083	17,578	4,694	380	35,602	7,240	380
Hayden	6,800	41	7,201	113	178	8,013	161	227
Pzone	438	31	812	47	49	23,592	13,790	68
NSF / LDAS-SALK	239	1	4,562	16	66	153,062	175	67
NSF / SLAC-JCSG	514	77	4,317	563	47	18,052	1,876	55
NSF / TeraGrid			80,354	685	2,962	276,413	7,694	3,267
NIH / BIRN			5,416	3,366	148	18,921	18,500	385
Digital Library								
NSF / LTER	158	3	233	6	35	257	41	36
NSF / Portal	33	5	1,745	48	384	2,620	53	460
NIH / AfCS	27	4	462	49	21	733	94	21
NSF / SIO Explorer	19	1	1,734	601	27	2,681	1,201	27
NSF / SCEC			15,246	1,737	52	168,767	3,545	73
Persistent Archive								
NARA	7	2	63	81	58	3,793	4,983	58
NSF / NSDL			2,785	20,054	119	5,699	50,600	136
UCSD Libraries			127	202	29	190	208	29
NHPRC / PAT						1,888	521	28
TOTAL	28 TB	6 mil	194 TB	40 mil	4,635	830 TB	126 mil	5,447

At the San Diego Supercomputer Center, the SRB technology is used to support shared collections in collaborations with other academic institutions. Table 4 shows how these shared collections have grown over time. The shared collections are characterized as data grids, digital libraries, and persistent archives. It is now common to see shared collections that are being assembled by teams of 20 to 100 persons (Users with Access Controls), that have more than 10 million files, and that aggregate over 100 Terabytes of data. In total, SDSC manages over 830 terabytes of data under shared collection control, comprising more than 126 million files.

The SRB is implemented as peer-to-peer software servers, installed at each site where data will be stored or accessed. A user accesses the shared collection from their preferred client, whether web browser, GridSphere portal, workflow system (Kepler), load library (Perl, Python, Windows), Unix shell command, “C” library call, digital library (DSpace, Fedora), OpenDAP data access protocol, Open Archives Initiative Protocol for Metadata Harvesting, MPI-IO library, etc. The client communicates with a local SRB server, which resolves the requested operation by extracting information from a SRB metadata catalog (MCAT). The metadata catalog holds all information needed to map from the persistent logical name spaces to physical users, files, and resources in the shared collection. Once the location of the desired file is resolved, the request is forwarded to a SRB server at the remote location where the data resides. The remote SRB server retrieves the data, and sends the data directly back to the requesting client.

For the peer-to-peer servers to manage data securely, the SRB implements trust virtualization. Data is contributed to the shared collection, which then stores the shared data under account identifiers that correspond to the shared collection. At each site where data is stored, an account is established under which the SRB data grid can run as application level middleware. When a user requests data, he/she is authenticated by the SRB data grid. Access controls are checked by the SRB data grid to determine whether the user has the required permission for the desired operation. Access controls are maintained separately by the SRB on files, metadata, and storage resources. The SRB server then authenticates itself to the remote SRB server, which in turn authenticates requests to the remote storage system. Thus the SRB accesses shared data on behalf of the user. The types of authentication systems that can be used include challenge-response in which the password does not go over the network, public key infrastructure based on the Globus Security Infrastructure, and ticket mechanisms that control the number of times data may be accessed and the time period the ticket is valid.

The advantages of trust virtualization are that the SRB does not require root access for installation, the administration of the shared collection is simplified (the only account that must be installed at each site is that of the SRB data grid), and file sharing can be controlled independently of each storage location. Trust virtualization relies on the operating system security at each site, and adds a layer of security on the messages that are exchanged between sites.

The SRB architecture is shown in Figure 1. The SRB provides two indirection mechanisms to manage technology evolution; standard operations executed at the remote storage system, and standard actions that can be called by clients. The standard set of operations include the 16 standard Posix I/O commands for create, open, close, read, write, seek, stat, ... The standard operations have been extended to support latency management functions required for effective use of wide-area-networks, such as parallel

I/O, data aggregation in containers, remote procedures for minimizing the number of I/O commands sent over the network, bulk file registration, bulk metadata registration, bulk file loading, bulk file deletion, and database manipulation. The standard actions drive the execution of multiple operations at a remote site. Examples are import and export of metadata within XML files, schema extension, support for user-defined metadata, support for hierarchical metadata schema, end-to-end data encryption, replication of data, synchronization of replicas, and validation of checksums.

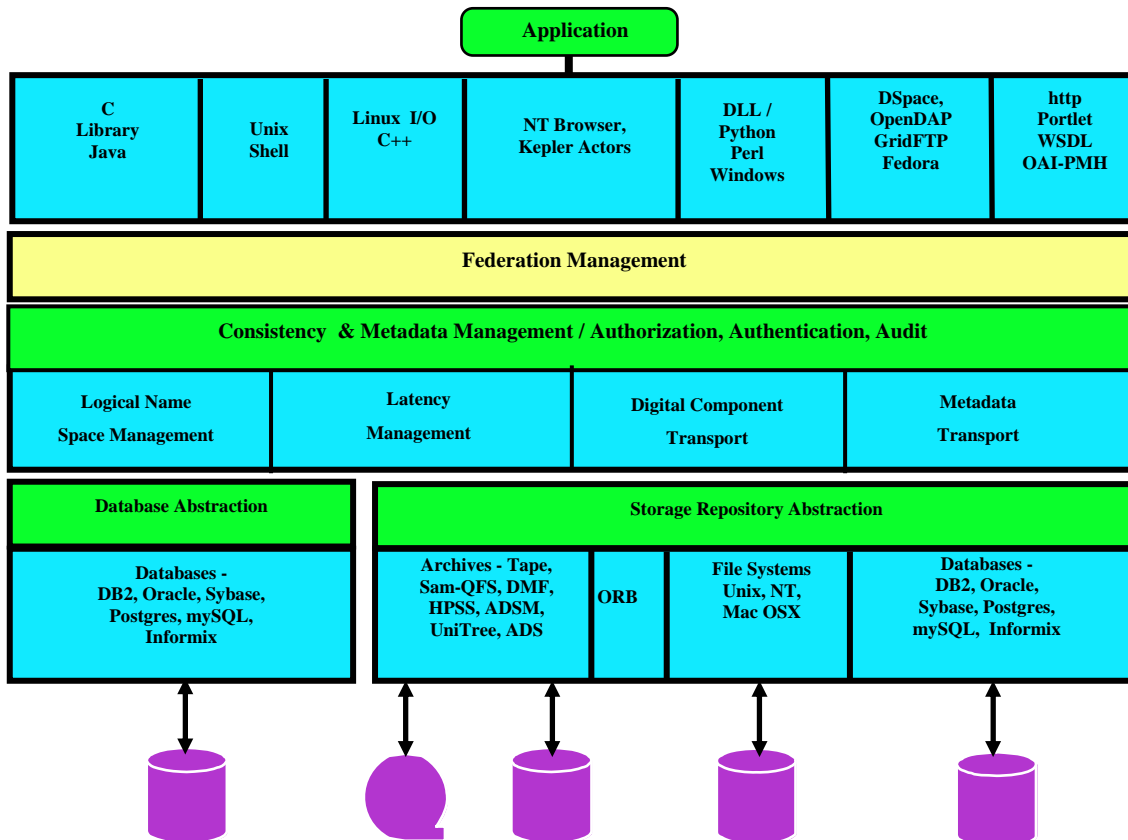


Figure 1. Storage Resource Broker Architecture

The SRB data grid supports federation. This is the controlled sharing of the logical name spaces between two independent data grids. For federation to work, trust must be established between the federated data grids. When a SRB data grid receives a request from a remote data grid, it checks that the trust mechanism has been established, and forwards an authentication request to the home data grid of the user. If the user authentication is approved, then the SRB data grid applies its local authorization controls and decides what the user will be allowed to do.

Federation is heavily used in building internationally shared collections. Each collaborating institution typically implements a local data grid. A pull-hierarchy is established between the data grids, with data and metadata replicated from one data grid to another under administrator control. This ensures that accesses are made to a local environment, that the data grid administrators control what is shared, that remote network problems do not affect local access, that data are reliably replicated in case of a natural

disaster or loss of data, and that the performance remains at the required levels. Federations of SRB data grids have been established that circle the globe. These large scale data management systems see network latencies as high as 440 milliseconds. Major examples of federated SRB data grids include the National Optical Astronomy Observatory data grid. NOAO federates 5 data grids to move images from Chile to the US. The BaBar high-energy physics data grid moves up to 3 terabytes of data per day from Palo Alto, California to Lyon, France. The BaBar project is now planning to move up to 5 terabytes of data per day and has moved over 320 terabytes of data so far. Finally, the ROADnet project at the Scripps Institution of Oceanography is installing independent data grids on each of the oceanographic research vessels in the US academic oceanography fleet. A stand-alone combined hardware/software system is installed on each ship (ROADNet point of presence) from which data distribution is managed from the ships to sites on land and between ships over satellite links through federated SRB data grids.

The SRB technology is implemented in about 300,000 lines of C code. Information about the SRB can be found at <http://www.sdsc.edu/srb/>. A commercial version of the SRB software is distributed by Nirvana Storage.

Operational Data Grids

Production data grids have both the challenge and the benefit that they are the only interface seen by the end user. A collaborator on a shared collection does not have to worry about the different protocols used by the multiple storage devices, or differences in administrative policies about data residency lifetime, or local site authentication. On the other hand, any problem that occurs in the environment is the responsibility of the data grid, whether it is a network outage, or a disk crash, or a corrupted tape. A production data grid has to protect itself from all possible sources of data loss and provide to the users an environment in which the data have strong guarantees on integrity and authenticity [5].

In practice, no storage system can be trusted to reliably store data for the lifetime of a collection, which may be longer than 20 years. At a minimum, the technology used within the storage system will become obsolete, and the data will need to be migrated to more cost effective technology. At the worst, the storage system will lose the data or corrupt the data. Types of events that lead to data loss include media failure, systemic vendor product failure (such as bad microcode in a tape drive), operational error, natural disaster, and malicious users.

The SRB data grid provides mechanisms to ensure data integrity, operational procedures to ensure reliable management of the collection, and consistency mechanisms to ensure that the authenticity of the data in the collection remains uncompromised. Individual users can choose whether to replicate data, set access controls for sharing of data that they own, and add descriptive metadata. The SRB data grid administrator, however, manages the properties of the shared collection. Thus the data grid administrator manages storage quotas, installs new user accounts, adds new storage systems, and manages assertions about the integrity and authenticity of the shared collections.

Data Grid Management

Before a data grid federation can be effectively integrated, each of the component data grids must be reliably managed. A data grid administrator performs operational tasks to ensure the smooth operation of the data grid, usually assisted by systems analysts who maintain the storage systems, network administrators who manage both security systems and networks, and database administrators who maintain databases in which the data grid metadata catalog is housed.

The multiple levels of hardware and software systems that must work together seamlessly are:

- Data grid federation software
- Application level client software
- Data grid servers
- Data grid metadata catalog
- Security environment
- Storage systems
- Database
- Network

A failure in any one of these systems is viewed as a failure of the data grid. A data grid must ensure the end-to-end reliability and availability of the integrated system across all types of failures. Given the multiple levels of hardware and software systems that are integrated by a data grid, it appears that management of integrity and authenticity of distributed collections is very difficult. Data grids overcome these apparent difficulties through the use of checksums, replication, synchronization, and federation [7]. The intent is to provide multiple copies of each file, assert that the copies are up to date, that the copies are uncorrupted, and that a copy resides in an independently administered domain on a different type of storage system. At the same time, state information must be replicated across independent databases to ensure no single point of failure.

A list of the operations supported by data grids to maintain high availability while mitigating risk of data loss is available at the Storage Resource Broker wiki: <http://www.sdsc.edu/srb/>. The operations include:

Management of end-to-end validation of checksums. A checksum is created before a file is registered into a SRB collection, and the checksum is validated after storage of the file. Related operations include creation of checksums for previously registered data.

Management of replicas, versions, and backups of files. A replica is a true copy of a file. Changes to one of the replicas of a file can be synchronized to the other replicas. A version is a numbered copy of a file, such as a unique representation. A backup is a time-stamped copy. A request to make a replica always creates another copy of the file. A request to make a backup replaces the previous backup of the file.

Management of synchronization. This issue is more difficult than it appears, because synchronization could be done between the collection and external (non-SRB managed) storage systems, between file system buffers and disk, between disk caches and tape archives, between two SRB-managed collections, and between two SRB data grid federations. The SRB can register an external file system directory structure into a SRB collection. The registered files can then be synchronized with other SRB collections.

The ability to synchronize file system buffers is essential when dealing with small files. It is possible for a storage system to report completion after the files are written to the file system buffer and before the files are written to disk. A storage system crash will then lose the files, even though the data grid believes the data are safely stored. A similar problem occurs with storage systems that write data to a disk cache before archiving data on tape. The data grid needs to protect its integrity by forcing synchronization of the data all the way to the end storage media.

The ability to synchronize replicas is essential for managing data distributed over wide-area-networks. If an attempt to create a remote file fails because of a network problem or storage system outage, the data grid can defer creation of the replica until the system is available. Also, changes to a file can be made on a single replica, and then propagated to the other replicas after the updates are complete.

Consistency checks on the integrity of the metadata catalog. This requires two checks: that files exist for each of the entries in the metadata catalog; and that for each file in a SRB vault a metadata record exists in the metadata catalog.

Management of slave catalogs. A standard approach is to create additional read-only metadata catalogs to ensure high availability or to ensure improved response at a remote site. All writes are done to the master catalog to ensure consistency. Synchronization of the slave catalog is done at selected intervals to download changed metadata.

Management of federations. The replication of an entire collection (including name spaces, data and metadata) can be done onto a separately administered data grid. This capability ensures that an independent environment with separate operational procedures is managing a copy of the collection. An operational procedural error on one data grid can be recovered by transferring the lost data or metadata from the federated data grid. This capability is used in both preservation environments and digital libraries.

The data grid administrator executes the above operations to manage assertions about the shared collection. The assertions can be a statement that the integrity of all files has been verified within a specified time period, or that the required number of replicas exists for each file, or that the metadata catalog has been synchronized with the SRB data grid vaults. If problems are found such that a desired assertion has not been met by the data grid, the data grid administrator may need help from systems analysts about storage system interactions or from network administrators about network interactions or from database administrators about interactions between the metadata catalog and the database. Typical data grid administrator tasks include both periodic assertion testing as well as intermittent operational tasks.

Automation of Data Grid Management

While the SRB technology provides the mechanisms needed to ensure integrity and authenticity of shared collections, as the size of the shared collections increases, the management tasks become onerous. To support the automation of management policies, SDSC is developing the next generation of distributed data management technology, called iRODS – integrated Rule-Oriented Data System. The iRODS system implements management policies as rules that control the execution of remote micro-services, while managing persistent state information that can be used to validate management assertions. The system currently is about 50,000 lines of C code.

The iRODS system has as its initial application the management of distributed data, and thus implements a subset of the capabilities provided by the SRB. The system is designed to support dynamic specification of management policies as rules controlling micro-services. For each micro-service, we manage persistent state information. To build a generic system, iRODS supports:

- logical name space for rules
- logical name space for micro-services
- logical name space for persistent state information

These are in addition to the logical name spaces used to manage shared collections:

- logical name space for users
- logical name space for digital entities (files)
- logical name space for resources

This means the system will be capable of supporting evolution of management policies. Multiple versions of rules can be executed independently on separate collections. Each rule controls the execution of a set of {micro-services, other rules}. For each micro-service, a recovery procedure is specified. This is required to manage consistent state information in the event of any error within the distributed environment. The types of rules that may be applied include:

- atomic constraints, such as management of the execution of each remote data manipulation micro-service through access controls
- periodic constraints, such as integrity checks
- collection assertions, such as data distribution, required descriptive metadata, authenticity, disposition, ...

The intent is to build a generic system. We are examining integration with workflow systems (such as Kepler) for long-running, scheduled processes. iRODS should be able to support the management of persistent state for any micro-service (remote procedure), whether executed under iRODS control or the control of a workflow system.

The key features differentiating iRODS from other rule environments are a combination of:

- management of rules in a distributed environment
- expression of rules as sets of micro-services
- support for nested rule sets (rules calling other rules)
- support for recovery procedures
- logical name spaces for rules / micro-services / persistent state
- management of data and metadata generated by application of the rules

These capabilities will make it possible to specify management policies, collection properties, and access policies and validate their application. An example is the set of assessment criteria for trusted digital repositories [8]. A first pass has been made at defining the rules required to validate the assertions, and enforce the micro-services needed to implement the policies [9]. An expectation is that generic infrastructure can be created that supports the application of the management policies. The set of rules that enforce the management policies varies between each community and each shared collection.

Summary:

The Storage Resource Broker data grid is used in production today to manage over a petabyte of data for internationally shared collections. The SRB is mature software that implements the capabilities required to support shared collections, digital libraries, persistent archives, and real-time data management systems. An examination of the management tasks required for distributed collections has driven the development of a new generation of software, called iRODS. Current large-scale systems require the automation of collection management tasks, of validation of assertions made about the shared collections, and of the validation of the desired properties of the collections. Rule-based systems provide the mechanisms needed to express both data management policies and assertions, and can build upon the concepts demonstrated by data grids for distributed data management.

References:

- [1] R. Moore, M. Wan, and A. Rajasekar, "Storage Resource Broker: Generic Software Infrastructure for Managing Globally Distributed Data", Proceedings of IEEE Conference on Globally Distributed Data, IEEE Computer Society, Piscataway New Jersey, June 28, 2005, pp. 65-69.
- [2] I. Foster, and C. Kesselman, "The Grid: Blueprint for a New Computing Infrastructure," Chapter 5, "Data Intensive Computing," Morgan Kaufmann, San Francisco, 1999, pp. 105-129.
- [3] R., Moore, A. Rajasekar, and M. Wan, "Storage Resource Broker Global Data Grids", Proceedings NASA / IEEE MSST2006, Fourteenth NASA Goddard / Twenty-third IEEE Conference on Mass Storage Systems and Technologies, IEEE Computer Society, Piscataway New Jersey, April 2006.
- [4] R. Moore, "Building Preservation Environments with Data Grid Technology", *American Archivist*, The Society of American Archivists, Chicago Illinois, July 2006, vol. 69, no. 1, pp. 139-158.
- [5] R. Moore, R. Marciano, "Technologies for Preservation", chapter 6 in "Managing Electronic Records", edited by Julie McLeod and Catherine Hare, Facet Publishing, UK, October 2005.
- [6] C. Baru, R. Moore, A. Rajasekar, and M. Wan, "The SDSC Storage Resource Broker," Proc. CASCON'98 Conference, Toronto, Canada, Nov.30-Dec.3, 1998, p. 5.
- [7] A. Rajasekar, M. Wan, R. Moore, and W. Schroeder, "Data Grid Federation", 2004 International Conference on Parallel and Distributed Processing Techniques and Applications - Special Session on New Trends in Distributed Data Access, Las Vegas Nevada, June 2004.
- [8] RLG/NARA Audit Checklist for Certifying Digital Repositories, http://www.rlg.org/en/page.php?Page_ID=2076

[9] R. Moore, and M. Smith, "Assessment of RLG Trusted Digital Repository Requirements," Joint Conference on Digital Libraries workshop on "Digital Curation & Trusted Repositories: Seeking Success", Chapel Hill, North Carolina, June 2006.