Policy-based Distributed Data Management Systems

Reagan Moore
Arcot Rajasekar
University of North Carolina at Chapel Hill

Mike Wan
Wayne Schroeder
University of California, San Diego

The integrated Rule-Oriented Data System (iRODS) is a data grid that manages distributed data. The iRODS system is used to support data sharing, data publication, data preservation, and data analysis. These data management applications depend upon very similar data manipulation capabilities, but differ in terms of their management policies. The iRODS data grid organizes distributed data into a shared collection while enforcing uniform management policies across multiple administrative domains. The implementation of a consistent, extensible, scalable, and evolvable data management system required integration of concepts from a wide variety of systems: data grids, relational databases, database triggers, rule systems, workflows, and distributed operating systems. In the DICE group (Data Intensive Cyber Environments) we had previously built a widely used data grid called the Storage Resource Broker. The SRB data grid manages in production over three petabytes of data in national and internationally shared collections. Based on feedback from the SRB user community, we recognized the necessity of separating policy enforcement from procedures that are applied at remote storage locations. Our approach is based on the idea that we can build different types of data management applications by adding hooks into the framework of the data grid at those points in the execution where a policy should be enforced. The consequences of adding policy enforcement hooks are substantial, requiring a complete redesign of the approach towards distributed data management. Version 2.0.1 of the iRODS data grid was released in January 2009 as a production quality data grid that turns policies into computer actionable rules that control the execution of remote procedures expressed as computer executable micro-services.

Our approach has several significant points of departure from the design of digital libraries and from the design of the SRB data grid. These include:

1. Explicit enumeration of the locations in the data management framework at which policy needs to be enforced. The iRODS system defines a minimal set of policy enforcement locations that enable the creation of generic infrastructure that can be used to support data sharing, data publication, data preservation, data analysis, and real-time data streams, by changing the management policies.
2. Explicit enumeration of micro-services, modules of executable code from which processing workflows can be composed. The traditional file system world builds upon Posix I/O semantics for files: open, close, read, write, seek, stat, … For the iRODS environment to be feasible, the level of composition of micro-services needed to be at a higher level of granularity to simplify construction of procedures that enforce management policy.

3. Explicit enumeration of the types of structured information generated by the application of remote procedures. For efficiency, the iRODS data grid had to be able to store in memory the structures generated by a micro-service, for efficient access by a chained micro-service. The structures also had to be linearized for transmission over the network to a micro-service at a remote location or to the client. This required development of a mechanism to describe each structure, and pack and unpack the structures for transmission.
4. Explicit enumeration of the state information attributes required to implement a data management system. Each remote procedure generates state information upon successful completion. The state information must be saved to ensure consistent operation of the data grid. The state information constitutes the memory of the system, tracking the status of every record in the shared collection.
5. Explicit enumeration of the policies that are being enforced within the data grid. The policies are enforced through a distributed rule engine that is co-located with every storage system used within the data grid. Thus all operations applied by the data grid on its digital holdings are executed under the control of rules that are stored at the storage resource. This makes it possible to enforce management policies across administrative domains for retention, disposition, distribution, replication, time-dependent access control, integrity, authenticity, chain of custody, trustworthiness, Institutional Research Board access approval, HIPAA compliance, provenance.
6. Explicit support for evolution of the data grid, through use of logical name spaces for first class objects that include users, files, resources, rules, micro-services, and state information. A major need was the ability to change a management policy and the associated procedures, and migrate data from the original collection which enforced the original policies, to a new collection management by new policies. Through use of logical name spaces, versions of each first class object can be managed. Within the same data grid, the old rules controlling the old micro-services that generate the old state information for the original collection can be run in parallel with the new rules that control the new micro-services generating new state information on a new collection. A rule can control the migration of data from a collection controlled by the old policies to a collection controlled by new policies.
7. Support for deferred and periodic execution of rules to enable automation of the validation of assessment criteria. A rule can be written that checks whether the current state information matches the desired values. If a discrepancy is found, such as a corrupted file caused by a disk head crash, the rule can access a valid copy and replace the corrupted file. Such checks need to be performed periodically since there are no perfect storage systems. Data may be lost through hardware malfunction, software malfunction, operator error, natural disasters, or malicious users. Assertions about properties of the shared collection are only as good as the set of assessment criteria that are used to validate the correctness of the system. Since policies can change over time, assessment criteria must also parse audit trails to determine the impact of policy changes. Given the ability to assess the system consistency, it is possible to have the system detect and repair problems, minimizing the amount of labor needed for data grid administration.

The combination of rules, micro-services, and state information constitute representation information for the data grid. They specify the policies and procedures that comprise the purpose behind the data management application, whether it be for data sharing, data publication, data preservation, or generation of derived data products. We examine each of these design areas to understand the impact that can be achieved through the integration of policy management into a data grid.

Acknowledgement:
The research results in this paper were funded by the NARA supplement to NSF SCI 0438741, "Cyberinfrastructure; From Vision to Reality" - Transcontinental Persistent Archive Prototype (TPAP) (2005-2008) and by the NSF Office of Cyberinfrastructure OCI-0848296 grant, "NARA Transcontinental Persistent Archive Prototype", (2008-2012). The iRODS technology development has been funded by NSF ITR 0427196, "Constraint-based Knowledge Systems for Grids, Digital Libraries, and Persistent Archives" (2004-2007) and NSF SDCI 0721400, "SDCI Data Improvement: Data Grids for Community Driven Applications" (2007-2010).

References:
1. integrated Rule Oriented Data System, http://irods.diceresearch.org
2. Rajasekar, A., M. Wan, R. Moore, W. Schroeder, "A Prototype Rule-based Distributed Data Management System", HPDC workshop on "Next Generation Distributed Data Management", May 2006, Paris, France.
3. Aschenbrenner, Andres et. al., "The Future of Repositories? Patterns for Cross-Repository Architectures," D-Lib Magazine, November/December 2008, Volume 14 Number 11/1.
4. Moore, R., "Towards a Theory of Digital Preservation", IJDC Volume 3, Issue 1, August 2008.