

RRS: Replica Registration Service for Data Grids

Arie Shoshani, Alex Sim, and Kurt Stockinger

Computational Research Division,
Lawrence Berkeley National Laboratory,
University of California,
1 Cyclotron Road, Berkeley, California 94720, USA
{AShoshani, ASim, KStockinger}@lbl.gov

Abstract. Over the last few years various scientific experiments and Grid projects have developed different catalogs for keeping track of their data files. Some projects use specialized file catalogs, others use distributed replica catalogs to reference files at different locations. Due to this diversity of catalogs, it is very hard to manage files across Grid projects, or to replace one catalog with another.

In this paper we introduce a new Grid service called the Replica Registration Service (RRS). It can be thought of as an abstraction of the concepts for registering files and their replicas. In addition to traditional single file registration operations, the RRS supports collective file registration requests and keeps persistent registration queues. This approach is of particular importance for large-scale usage where thousands of files are copied and registered. Moreover, the RRS supports a set of error directives that are triggered in case of registration failures. Our goal is to provide a single uniform interface for various file catalogs to support the registration of files across multiple Grid projects, and to make Grid clients oblivious to the specific catalog used.

1 Introduction

Managing a large number of files at distributed locations is one of the challenges that many large-scale scientific experiments face. For efficiency reasons, many of the files are replicated in multiple storage systems. In order to keep track of the files and their replicas, various file and replica catalogs are used. Typical questions are: What is the name-space for files registered in the catalogs? Shall the catalog be organized as a centralized or decentralized service? How can information about files be retrieved efficiently? How can different sites interact with each other's catalogs? In order to solve some of these issues, various Grid projects have developed different catalogs for keeping track of their data files. Some experiments use specialized file catalogs, others use distributed replica catalogs to reference files at different locations. This diversity of catalogs makes it very hard to manage files across Grid projects or even within a single project. The solution to this problem is not to attempt and standardize a particular file catalog

system. Rather, the approach taken here is to provide a uniform specification of a functional interface that permits the multiplicity of catalogs to co-exist. This approach is similar to many commercial products (such as relational database systems) that have a common functional interface specification that permits multiple systems to co-exist. Furthermore, our approach isolates the Grid client programs from the specific catalog system used.

In this paper we introduce a Grid service called the Replica Registration Service (RRS), that provides a uniform functional interface to various file catalogs, replica catalogs, and meta data catalogs. It can be thought of as an abstraction of the concepts used in catalog systems to register files and their replicas. Some experiments may prefer to support their own file catalogs (which may have their own specialized structures, semantics, and implementations) rather than use a standard replica catalog. Providing a RRS that can interact with such a catalog can permit that catalog to be invoked as a service in the same way that other more general-purpose replica catalogs do. If at a later time the experiment wishes to change to another file catalog, it is only a matter of developing a RRS for that new catalog and replacing the existing catalog. Similarly, an existing replica location service (RLS, [3]) that supports the RRS interface can be plugged in instead of the current catalog. In addition, some systems use meta-data catalogs or other catalogs to manage the file name-spaces, and those could be accessed through the same RRS interface as well.

The main contributions of this paper are as follows:

- We introduce a novel Grid service - the Replica Registration Service. We discuss the design considerations and the main functional interface components.
- We report on our experience of using an early implementation of a RRS in a production environment. The results show that the RRS could greatly simplify the management of replicas and reduce registration errors.

2 Related Work

An early version of a replica management framework is presented in [4] where the terms *Replica Management* and *Replica Selection* are defined within the context of a Data Grid. In the European Data Grid Project a similar replica management framework was implemented [7]. An integrated approach for data and meta-data management is provided in the Storage Resource Broker (SRB) [1]. In general, data replication can be done at the file or object level, where multiple objects can either be stored in a single file, or a single object can be stored across multiple files. The differences between object and file replication are discussed in [13]. However, in practice only file replicas are cataloged so far, since the number of objects can be much larger than the number of files. Moreover, the implementation of object replication systems is more complex. For this reason, we focus on file replica registration only in this paper.

In the early days of Grid computing, replicas were stored in LDAP catalogs. Due to performance issues, subsequent replica catalogs or replica location ser-