Chapter 4.1
Accessing Grid Metadata through a Web Interface

Salvatore Scifo
Cometa Consortium, Italy

ABSTRACT

This chapter focuses on the efforts to design and develop a standard pure Java API to access the metadata service of the EGEE Grid middleware, and provide at the same time a powerful object oriented framework to allow engineers and programmers to embed metadata features inside their own application, using a standard approach based on design patterns. A specific Web interface is built on top of this framework that permits users and administrators to manage the metadata catalog, from any platform and everywhere, according to their own X.509-based credentials.

INTRODUCTION

AMGA service is the implementation of the metadata interface designed by the ARDA team for the official metadata service of the EGEE Grid middleware. It is a complete but simple interface to use. The service is flexible enough to support dynamic schemas in many application domains as well as hierarchical metadata structure definitions. The service presents a high degree of scalability to deal with large number of entries (several millions) and the security aspect is fully compliant with the Grid Security. AMGA has been designed to hide network latency (based on TPC/IP communication protocol) as well as provide local replicas for off-line access. Its database independent replication supports grid environment heterogeneity and avoids the presence of single point of failure. As a grid service, developers provide only command line client and APIs (PHP, C/C++, Python and Java). In this paper, we describe both the basic API and a web interface to achieve access to the AMGA metadata sever from any

DOI: 10.4018/978-1-4666-0879-5.ch4.1

Copyright © 2012, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.

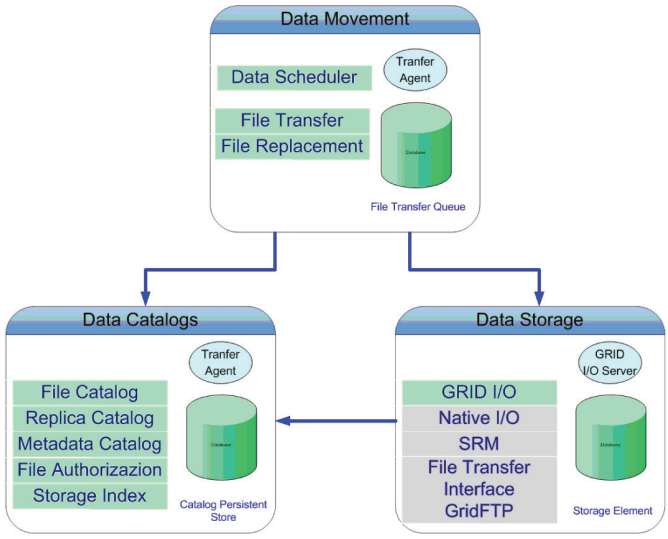
platform. The AMGA Web Interface provides a powerful interface to access and manage GRID metadata according to X.509-based credentials. To be authenticated to the AMGA server, the user needs just a web browser and a valid X509 digital certificate released by one's Grid Virtual Organization Certification Authority. After a successful login, one can browse the hierarchy of collections and inspect schema/ permissions as well as the list of entries. One can create a new collection, define a metadata schema for it, edit entries, and finally perform queries against its attribute. The AMGA web interface follows the three-layer architecture of J2EE and adopts the framework as a data access layer.

BACKGROUND

This work is closely linked to the metadata service of the gLite, the middleware of the EGEE Grid European project. Within the gLite architecture, the data management system (DMS) is an essential part that enables users and applications to handle data and metadata without referring to the complex details of the computing environment. DMS provides APIs and Client tools to store, locate, access, retrieve and move files dispersed on the distributed virtual File System. From the functional point of view, gLite DMS offers two fundamental macro features: file management and metadata management. The first one (performed by the file catalogue service and the storage resource manager), involves the storing abilities (save file, copy file, read file, list file), placing abilities (replica file, transfer file) and security stuff (ACL for files, users roles). The second one (implemented by the metadata catalogue), offers database schema virtualization (metadata handling, intelligent search), file cataloguing and file searching. Figure 1 shows data management system modules and their interrelations.

The data storage is the subsystem that manages data manipulation so that users and/or applications can access and manage their own files. The data movement subsystem enables any other Grid services or any clients (users and/or applications) to move file from/to a site, (a site is the smallest, complete and auto consistent hardware/ software resource organization within a Grid infrastructure). The data catalogue subsystem keeps track of file location in the distributed file

Figure 1. DMS architecture view



9 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:
www.igi-global.com/chapter/accessing-grid-metadata-through-web/64514

Related Content

Rule Extraction and Rule Evaluation Based on Granular Computing
Jiye Liang, Yuhua Qianand Deyu Li (2010). *Novel Developments in Granular Computing: Applications for Advanced Human Reasoning and Soft Computation (pp. 196-242).*
www.irma-international.org/chapter/rule-extraction-rule-evaluation-based/44705

A Secure and Privacy-Preserving Approach to Protect User Data across Cloud based Online Social Networks
Neelu khareand Kumaran U. (2020). *International Journal of Grid and High Performance Computing (pp. 1-24).*
www.irma-international.org/article/a-secure-and-privacy-preserving-approach-to-protect-user-data-across-cloud-based-online-social-networks/249741

Ambulance Dispatching System with Integrated Information and Communication Technologies on Cloud Environment
Jian-Wei Li, Chia-Chi Chang, Yi-Chun Changand Yung-Fa Huang (2014). *International Journal of Grid and High Performance Computing (pp. 72-87).*
www.irma-international.org/article/ambulance-dispatching-system-with-integrated-information-and-communication-technologies-on-cloud-environment/127375

Fast Paxos Made Easy: Theory and Implementation
Wenbing Zhao (2015). *International Journal of Distributed Systems and Technologies (pp. 15-33).*
www.irma-international.org/article/fast-paxos-made-easy/120458

Load Balancing in Peer-to-Peer Systems
Haiying Shen (2010). *Handbook of Research on Scalable Computing Technologies (pp. 163-190).*
www.irma-international.org/chapter/load-balancing-peer-peer-systems/36408