

Evaluation of OGSA DAI for E2EpiPES

P.D.Mealor

June 10, 2003

1 Version history

21 May 2003 Initial version

10 June 2003 Added information from installation of OGSA-DAI implementation

2 Introduction

I have looked at the architecture of OGSA-DAI with respect to using it as the interface to the E2EpiPES performance database. Initial use of the OGSA-DAI implementation (version 2) backs up the contents of this evaluation. However, no in-depth investigation into the performance of OGSA-DAI or the use of meta-data/service data has been performed.

If it's felt to be necessary I can investigate the use of service data and I can investigate the performance of the implementation.

3 Glossary

GDS Grid Data Service

GDSF Grid Data Service Factory

GDSR Grid Data Service Registry

GSH Grid Service Handle

4 Summary

OGSA DAI can present either an SQL-based database or an XML-based database. In general, SQL databases (or Grid Data Services) will handle only SQL queries, and XML databases, XML queries.

5 Functionality

5.1 Registration and discovery

The registration process is essentially the same as the standard OGSi registration system. The only addition is an extensible `GridDataServiceMetaData` element that can be used to describe a Grid Data Service such that it can be found. A `GridDataServiceFactoryMetaData` element can be used to describe a Grid Data Service Factory.

The discovery process is extended with an additional query type: `GridDataServiceDataQueryExpression`. This query type searches a named element of the `GridServiceMetaData` for each registered GDS/GDSF for particular values specified by a Java regular expression. It returns the information about every GDS/GDSF (the client can specify one or both) which matches.

That the meta data types are extensible means that any XML data may be stored in it. Thus meta data for any particular purpose may be registered and queried.

5.2 Performing queries

Queries can be either SQL (JDBC) or XML:DB, although GDSs are likely to only support one.

Queries (and operations) are posed to the GDS in the form of Grid Data Service Perform Documents passed to the `GridDataService::perform` operation. The perform document consists of a (any number of?) request element, and an execute element; alternatively, processing can be halted with a terminate element. The perform document may contain

documentation at the root level or within a request element.

The request element consists of a query statement, either XML or SQL, and a delivery element.

The delivery element allows the response to the query to be placed in the operation response message, delivered to a Grid Data Transport port type, to a URL or GFTP server, or to a stream.

5.2.1 JDBC query statements

I expect that we would initially use the JDBC (that is, SQL) interface to the current database. Supported SQL activities are:

sqlQueryStatement activities are SQL query expressions for the GDS to evaluate.

sqlUpdateStatement activities are SQL update expressions for the GDS to evaluate. Expressions do not have to just include UPDATE, but may include insert or delete statements.

sqlStoredProcedure activities call a stored procedure of the database.

relationalResourceManagement activities are database management requests, such as the creation or removal of databases or tables.

5.2.2 XML:DB query statements

This section is included just for completeness, although an XML database might be more suited to storing information based upon NM-WG work. Supported XML activities are:

XCollectionManagement activities deal with adding or removing collections to the database.

XResourcesManagement activities deal with adding or removing resources (entries) to the database.

XPathStatement activities are queries to the database using XPath[3].

5.3 Security

The default security model in OGSA-DAI is based on the Grid Security Infrastructure[1]. GSI is based on X509 certificates, where individuals or machines can prove their identity using Certificate Authority-signed certificates. OGSA-DAI uses a role-mapping system, where an individual's Distinguished Name is mapped to a particular role for authorisation purposes.

OGSA-DAI security has been written in a modular form, to attempt to make it easier to plug in different security models, should they be necessary.

6 Implementation notes

The OGSA-DAI project provides a Globus Toolkit 3 alpha 3 implementation of OGSA-DAI in Java. The implementation can run under any Web Services container, such as Tomcat.

6.1 Registration

In the standard implementation only a GDSF may register, but with any number of GDSRs. The GDSF can be configured to register with custom registration information, allowing us to specify any service meta-data we feel would be important for E2E piPES.

A client API is provided, and this includes provision for querying registration information.

Use of the registration system could remove the need to have a virtual database behind a common front-end, replacing that with many OGSA-DAI services, found through a common registry. However, the virtual database model could be maintained by making use of the registration system behind the front-end.

6.2 Java overhead

Because the Globus Toolkit and the OGSA-DAI implementation are written in Java, there are some over-

heads associated with the design and use of applications based on them.

A worry might be that OGSA-DAI and GT3 are entirely Java-based, although GT3 has a C++ client-side API. This constrains us to either use Java for the GDS and GDSR components of the hierarchy, or to have to attempt to write an implementation in another language. (I recall that it was asserted at the E2E monitoring meeting that people were not particularly moving towards Java at the moment).

I am unable to find any studies on the performance of the OGSA-DAI implementation.

6.3 OGSA overhead

Basing the database access on OGSA requires that some infrastructure has to be put in place: a system of registries has to be created, for example. This cost can be partially shared with the cost of setting up the infrastructure for other OGSA-based components, such as a Grid Services administration interface, or analysis engine.

7 Conclusions

References

- [1] Global Grid Forum GSI-WG, http://www.ggf.org/2_SEC/GSI.htm. *Grid Security Infrastructure (GSI)*.
- [2] OGSA-DAI, <http://www.ogsadai.org.uk>. *OGSA Data Access and Integration*.
- [3] World Wide Web Consortium, <http://www.w3.org/TR/xpath>. *XML Path Language (XPath)*. Editors: James Clarke and Steve DeRose.