GIO: A Semantic Web Application Using the Information Grid Framework

Omar Alonso Sandeepan Banerjee Mark Drake

Oracle Corp.
500 Oracle Parkway
Redwood Shores, CA 94065 USA

ABSTRACT

It is well understood that the key for successful Semantic Web applications depends on the availability of machine understandable meta-data. We describe the Information Grid, a practical approach to the Semantic Web, and show a prototype implementation. Information grid resources span all the data in the organization and all the metadata required to make it meaningful. The final goal is to let organizations view their assets in a smooth continuum from the Internet to the Intranet, with uniform semantically rich access.

Categories and Subject Descriptors

H.3.3 [Information Search and Retrieval] Search process; clustering; H.2 [Database management] General

General Terms: Design, Management.

Keywords: Semantic Web, RDF, search, browsing, clustering, meta-data, information visualization, databases, tools, user interface.

1. INTRODUCTION

Two somewhat contrary-sounding drivers fuel the current trend in enterprise data management: *virtualization* and *convergence*. Virtualization is a framework for dividing up the resources of an organization into multiple execution environments, by applying one or more technologies such as hardware clustering, software partitioning, application modularization, emulation, and so on. The driver behind virtualization is the lowering of cost. Convergence, on the other hand, seeks to bring together the management of all your data assets. Today, a small percentage of the world's information is managed, and most of what is found to be valuable to manage (e.g. capture, store, index, search, analyze) falls into the category of structured data. Being able to manage the remaining data is what convergence is all about. In XML we finally have a data model that is capable of addressing highly structured data, textual unstructured-data, and anything semi-structured in between. The real driver behind convergence is better business intelligence across all your assets. When unstructured information becomes a managed resource, it can be integrated into more day-to-day organizational processes, such as search and compliance. Moving towards a new data management architecture based on XML-backed information repositories will be a key future step for organizations. We call this architecture, which combines virtualization and convergence, the *information grid*.

WWW 2006, May 23–26, 2006, Edinburgh, Scotland.
ACM 1-59593-323-9/06/0005.

2. THE INFORMATION GRID

The resources in the information grid span all the data in the organization, as well as all the metadata required to make that data meaningful. This data may be structured, semi-structured, or unstructured, stored in any location, such as databases, local file systems, or email servers, and created by any application. The vision for the information grid builds on the vision of the semantic web; the goal is to enable organizations to view all their assets in a smooth continuum, from the Internet to the Intranet, with uniform semantically rich access.

Within an application grid, individual modules run on different parts of the infrastructure, with sharing of application state and control enabled via web services. Each module, however, is still tightly coupled to its data – say database, file-system, mail server – and intelligence about the data have to be compiled into the application module. An information grid, in contrast, is self-describing; the application modules can discover what sources exist, what data they possess, what the life cycle of that data is, and how that data should be interpreted.

The main components of the information grid are: 1) Repository, Metadata and Service Management, 2) Semantic Crawlers and Search, 3) Information Presentation and Visualization, and 4) Inference.

3. OVERVIEW OF ORACLE TECHNOLOGY

In this section we briefly summarize the main features of the Oracle database in the context of an information grid [3].

- XDB Repository. Oracle XDB provides a storage independent, content independent, and programming language independent infrastructure to store and manage XML data.
- Search API. Oracle provides a rich full-text search API that can be used to build information retrieval applications.
- RDF support. The RDF data model supports three types of database objects: model (RDF graph consisting of a set of triples), rulebase (set of rules), and rule index (RDF graph) [4].

4. PROTOTYPE

GIO was written in Java and leverages all the Oracle10*g*R2 features that were described in the previous section. We have used the DBLP digital library collection that provides a copy of the data set in RDF as well as a simple ontology in OWL.

GIO's architecture consists on several back end processes that capture data, extract metadata, and creates indexes; along with other computations that manipulates the RDF data set. The other aspect of GIO is a servlet that provides search, browsing, clustering, and visualization features using tree maps and social networks (figure 1).

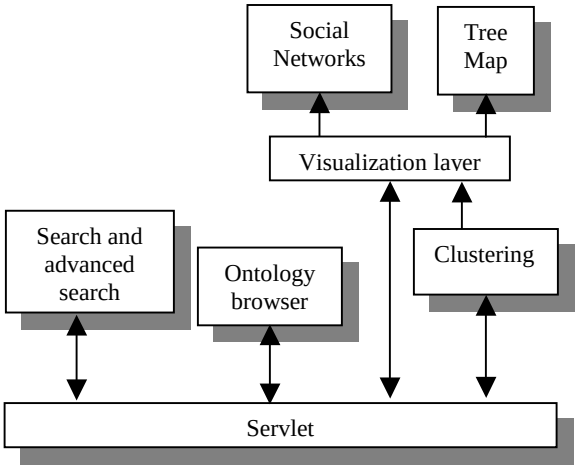

Figure 1. GIO as a web application

4.1 RDF and OWL in the Database

Typically, RDF data is presented as a single file with a root node of RDF. This node contains a large collection of description elements. In this case, it was decided to dispose of the RDF node, and treat the content as a set of description nodes, so as to enable document level access to the contents of each description element. SAX processing techniques allowed the large RDF content to be processed efficiently. In order to allow a file/folder metaphor to be used to access the description elements, a suitable folder hierarchy was generated from the ontology (OWL) associated with RDF data.

4.2 Database and Queries

Now that the content is inside the database, we can use a wide range of query mechanism, available through SQL, to retrieve data. For example a query for authors:

```
select value(auth).getClobVal()
from rdf_document_table, table(xmlsequence(
extract(
object_value,'/rdf:Description/author','xmlns:rdf="
http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns="http://example.org/"'))) auth
```

The query language is flexible enough to allow users to issue traditional SQL queries, XPath, full-text search, etc.

4.3 Information Access and User Interface

The GIO user interface consists on three views to access content. The first one is to provide browsing of articles using classes from the ontology. The second view is to use labels (or tags) based on clusters as an alternative classification scheme. Finally, the search box provides random access to any item. The single search box has the option to switch to the advanced mode. In contrast with Internet search engines, the advanced mode provides an interface that allows users to search fields depending on the availability (or not) of different sources. At the core of the system is a powerful mechanism to discover and populate metadata with XML. Using AJAX as the main technique, we built a flexible mechanism to search source and attributes dynamically. Figure 2 shows a search for DBLP publications from a particular school (expressed as an RDF attribute). Hit list clustering is used for presenting search results in context.

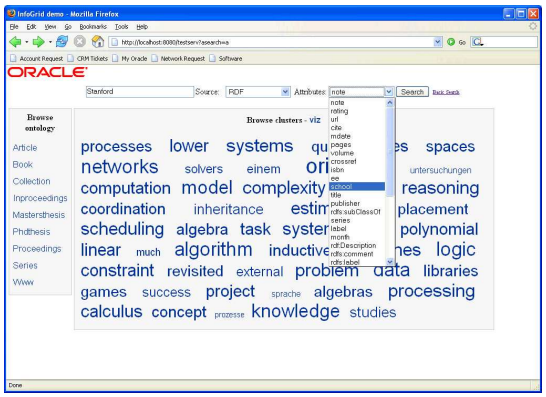

Figure 2. Discovery of RDF attributes in advanced search

4.4 Inference and Social Networks

One of most promising features of a semantic web is to see a domain of applications connected by concepts and to inference. An information grid involves more than just one data source. The ability to connect different sources and derive insight is a key part of the process. As an example of inference we want to display public personal information from a researcher and its social network (here as a co-authors). By dragging and dropping new content to the XDB repository, the system is now aware of new content. Once the XML document that represents a researcher is available, we can inference relationships (using the rules index) like co-authors who are then presented as part of a social network.

5. RELATED WORK

There has been work on applications that support a number of Semantic Web features like Haystack [1] and PiggyBank [2]. Our approach has a similar perspective on information access and offers more views like visualization. We also concentrate more on the back-end implementation with existing technologies.

6. CONCLUSIONS

We presented a prototype implementation of an architecture that can be used to build similar information grid applications. GIO is a live implementation built using existing technologies available with the Oracle 10gR2 database.

7. REFERENCES

[1] D. Karger *et al*. "Haystack: A Customizable General-Purpose Information Management Tool for End Users of Semistructured Data". CIDR 2003.

[2] D. Huynh, S. Mazzocchi, and D. Karger "PiggyBank: Experience the Semantic Web Inside Your Web Browser" 4th International Semantic Web Conference, ISWC 2005.

[3] Oracle 10gR2 Documentation. http://tahiti.oracle.com

[4] E. Chong *et al*. "An Efficient SQL-based RDF Querying Scheme", VLDB 2005.