DELIS
Dynamically Evolving, Large-scale Information Systems

Integrated Project
Member of the FET Proactive Initiative Complex Systems

Deliverable D2.3.4

Final Design of the Management Platform
<table>
<thead>
<tr>
<th><strong>Start date of the project:</strong></th>
<th>January 2004</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Duration:</strong></td>
<td>50 months</td>
</tr>
</tbody>
</table>
| **Project Coordinator:**      | Prof. Dr. math. Friedhelm Meyer auf der Heide  
                               | Heinz Nixdorf Institute, University of Paderborn, Germany |
| **Due date of deliverable:**  | December 2007|
| **Actual submission date:**   | December 2007|
| **Dissemination level:**      | PU – public  |
| **Work Package 2.3:**         | Global management of competing transport services over large net domains |
| **Participants:**             | RadioLabs (RAL), Universita di Roma “Tor Vergata”, Italy  
                               | Technical University Berlin (TUB), Germany |
| **Authors of deliverable:**   | Giovanni Cortese (g.cortese@computer.org)  
                               | Fabrizio Davide (fabrizio.davide@telecomitalia.it)  
                               | Felix Heine (fh@upb.de) |
1 Introduction

This report describes the final design of the data management platform which was the main subject of the work in workpackage 2.3. In workpackage 2.3, we research topics in distributed data management and dissemination, which can be applied to realize a management infrastructure for large collections of computing and communication resources (including transport networks, GRIDs, content distribution networks, etc). During the project, several research topics have been studied related to peer-to-peer networks and data management, using both structured and unstructured P2P networks as overlay architecture.

This deliverable summarizes activities related to the data management platform carried out in the 4th year of the Delis project. Full technical reports are provided in the DELIS TR series and referenced from this deliverable.

2 Design of Management Platform

In this section we present a summary of research activities on the management platform. Full descriptions of the algorithms and designs are provided in the DELIS TR series. The platform software is available as opensource package.

2.1 Babelpeers

The platform component BabelPeers developed by TUB, supported so far the RDF Schema semantics, which is an important first step, but not enough for many applications. TUB researched ways to support more complicated reasoning tasks without sacrificing scalability. The de-facto standard today for reasoning in semantics which are more expressive than RDFS are tableau algorithms. However, these algorithms assume that all knowledge is available in a central place. This approach is clearly not compatible with large-scale P2P networks. However, there are newer ideas based upon query rewriting that are more scalable but that restrict the expressivity of the underlying logic.

We have grounded on this work and extended it in two ways. In a companion deliverable, D2.3.5, research on algorithm is documented. As part of this activity, ways to extend these approaches to include important modeling primitives like inverse roles, transitive roles, and complex concept descriptions have been found. These techniques have been incorporated into the BabelPeers system implementation so that they are combined with scalable P2P based query evaluation algorithms. The component now provides the functionality of a P2P based RDF datastore, that supports complex queries including the above mentioned reasoning features. This is a truly novel result, as previous results in these areas always focussed either on more semantics or better scalability, but not on both goals together. The overall approach is documented in detail in [DELIS-TR-0601]. The main rewriting algorithm is detailed in [DELIS-TR-0600], which is a paper currently under review for the ESWC 2008 conference. The final design of Babelpeers is described in the two technical reports [DELIS-TR-0599] and [DELIS-TR-0622].

2.2 XGR

XGR design has been completed, and refined to better support applicability of the middleware in practical scenarios. Virtual data management has been added to the platform list of features. Virtual data refers to a feature of the platform, to provide added flexibility in dealing with data values managed in Local Storage mode. Virtual data need not to be encoded and stored as XML tuples. Data could be maintained in the format which is used by the application, for example as rows in a RDMBS table, files on the file system, lines in a local log file, etc. and can be ‘virtually’ inserted in the XGR storage: only a reference to the data item is stored in the XGR storage and the
data value is accessed by the platform using a custom class implemented and registered by the data provider in the local XGR node. With respect to replication, mechanisms for better handling phases with high peer turnover (specifically, system startup and shutdown) have been added. This was felt especially critical in development environment.

The final design of XGR is described in the technical report [DELIS-TR-0622].

3 Contributions to Deliverable

The main contribution to section 2.1 is described in [DELIS-TR-0599].

The main contribution to section 2.2 is described in the technical report [DELIS-TR-0622].

References


