A Comparison of the Performance of XML and Relational Databases for the Grid-SAFE project

Josh Green, Stephen Booth, Adrian Jackson,
EPCC, The University of Edinburgh

XML databases have become more common in the Grid and web services arena recently, mainly due to the popularity of XML for data transport/transfer. Many Grid applications or web services perform operations that take data and transmit it to another application or service in XML. If a traditional relational database is used the application has to retrieve data from a database, convert it into XML and transmit it to another application or service, or receive data in XML format and convert it into a format that can be stored in a traditional relational database. Converting between relational format and XML can be time consuming, especially if the applications or services are involved in high volumes of message traffic. Therefore, using a database that stores the data in the same format that it is transmitted in could improve the performance of any service that undertakes these types of functions.

More than 20 XML database implementations have been developed over the last few years, each varying how they stored data and what types of queries they support. They also range from open source to commercial, and portable to platform dependent. Two methods for querying XML databases are XPath and XQuery (both query languages defined by the W3C). XPath is generally supported by all XML databases but only provides minimal querying potential (XPath provides a simple path system to enable users to identify XML nodes that match a particular set of requirements). XQuery, which extends XPath with a subset of SQL-like queries, is less widely supported but more powerful.

For the Grid-SAFE project, which is funded by JiSC, we needed to evaluate whether an XML database could be used as the data repository for a resource usage and accounting service, as we have development of a number of different web service interfaces for Grid-SAFE that allow XML data to be uploaded or queried where previously querying and uploading has been performed using purely web pages or java servlets and the data has been stored in a MySQL relational database.

Whilst XML databases can technically support the functionality required for the service being developed, it was unclear whether they could provide the performance necessary to query and process large amounts of usage data. Specifically, Grid-SAFE has a requirement to allow users to query large amounts of data, possibly perform some processing on that data, and produce a report within a reasonable time period (i.e. users will not wait more than a few seconds for the report to generate). A typical reporting query for Grid-SAFE would be:

```
SELECT a, COUNT(*), SUM(b)
FROM job_usage_record
WHERE ‘StartedTimestamp‘> 1167609600
AND ‘CompletedTimestamp‘> 1170316800
AND ‘CompletedTimestamp‘< 1170435600
GROUP BY a;
```

Where `a` and `b` vary but `start` and `completed time` are almost always used to filter entries based on values supplied by the user. The underlying data can be anywhere from hundreds of thousands of entries up to tens of millions of entries, and as we need these queries to complete in a matter of seconds, the performance of the underlying database technology is important to us.

We chose two different XML databases, eXist and Sedna, and compared their performance on a variety of queries against MySQL (the relational database currently used by Grid-SAFE) to try and evaluate whether using an XML database for Grid-SAFE would give the required performance. Both these databases support XQuery, something which is necessary to perform the querying currently available in Grid-SAFE.

All the database implementations were analysed using a range of test queries, ranging from queries that worked on a few thousand records up to ~300,000 usage records. Some of the results from the performance tests are shown in Figure 1. It must be noted that the MySQL times on the graph in Figure 1 have been multiplied by 100 to enable all the results to be displayed on the same graph.

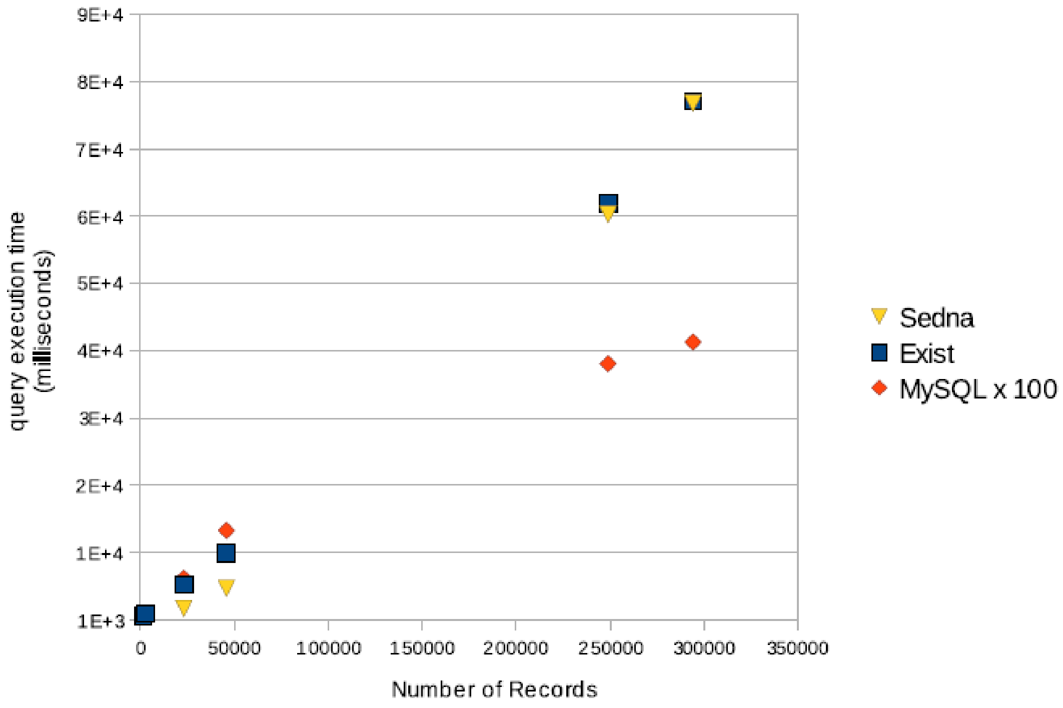

Figure 1: Time to complete query

We can see from Figure 1 that MySQL significantly outperforms both the XML databases tested, meaning that it is not feasible to move from a relational database to an XML database for the task we are undertaking at this time, especially as a number of our use cases for Grid-SAFE involve users generating reports or querying data from a web portal rather than from a web service. However, given we are providing web service interfaces to Grid-SAFE which will take data from the relational database and transform it into XML, we will still experience some performance degradation due to this transformation step. Our next task is to evaluate the computational cost of the transformation step and compare it to the impact of using an XML database.