

RDF(S) Manipulation, Storage and Querying using Sesame

Demo Proposal

Jeen Broekstra
Aduna
jeen.broekstra@aduna.biz

Arjohn Kampman
Aduna
arjohn.kampman@aduna.biz

15th October 2004

1 Introduction

Keywords: repository and inference systems, reusable components and APIs for RDF

Sesame [2] is an RDF framework with support for RDF Schema inferencing. Its main features include querying in three languages (SeRQL, RDQL, RQL), parsing and writing in several serialization syntaxes, support for MySQL, PostgreSQL, Oracle and SQL server as well as in-memory. It can be deployed as an RDF database, with persistence in an RDBMS, or as a Java library for embedded use in applications. We will demonstrate the numerous features of the Sesame framework in several scenarios.

The aim of this demonstration is to make RDF tool developers aware of the possibilities of Sesame as a framework for use in combination with their own tools. Therefore, the demo will focus on the middleware capabilities of Sesame, such as expressive querying and easy API access.

2 Architecture

In figure 1, the overall architecture of the Sesame framework is shown. The foundation of the system is the *Storage And Inference Layer (SAIL)*. This SAIL is an API that abstracts from the storage device used (in-memory storage, disk-based storage, RDBMS) and takes care of inferencing. On top of that we find the *functional modules* which implement the various high-level tasks, such as querying in SeRQL [1], RQL [4], or RDQL [5], adding/remove statements, etc.

From a user perspective the *Access APIs* are the most important component. These APIs provide high-level access functionality to client applications, either locally or remote (over HTTP or RMI). The *Repository API* provides access for querying, storing files and extracting RDF in different serialization formats. The *Graph API* provides a Java object model representing RDF graphs, that allows the user to do fine-grained manipulation of RDF in their application.

As we will show in the demonstration, the Sesame architecture is very flexible: it allows deployment in a wide variety of settings, as a remote database server or an embedded component in a desktop application.

3 Demonstration

The demonstration will consist of a remote Sesame server serving aggregated RDF/FOAF data, and several client interfaces that make use of this information in a variety of ways. Flexible deployment of a Sesame repository with RDF data of the audience's choice will also be possible.

One client interface to be demonstrated is the FOAF explorer. The FOAF explorer is a web-based graphical interface that allows the user to navigate and search FOAF profiles. The backend for the explorer is a Sesame server that contains aggregated FOAF profiles crawled (or 'scattered') from the Web. Using the FOAF explorer, the user can query the remote server for FOAF profiles, and visualize these FOAF profiles and navigate to other profiles based on 'foaf:knows' or 'rdfs:seeAlso' relations.

To demonstrate the capabilities of Sesame as an inferencing and querying platform, the demo will also demonstrate the evaluation of several complex SeRQL queries directly on the repository. The use of optional path expressions and graph transformation will be demonstrated through query examples.

Finally, the use of Sesame's access APIs for fast manipulation of RDF data will be demonstrated by use of a small web-based editor of RDF data that communicates directly with the Sesame repository. The use scenario will be the updating of several foaf relations in the remote repository.

3.1 SeRQL

A significant part of the demonstration will be the use of the SeRQL query language. SeRQL is a language that combines features from several predecessor languages (such as N3, RQL and RDQL) and adds some unique features of its own. The features that will be demonstrated include:

- graph transformation using CONSTRUCT-queries.
- optional path expressions and nested optional path expressions.
- explicit RDF Schema support.
- XML Schema datatype support.

Several example queries will be explained to the audience and then executed live to see the result. Also, the client interfaces mentioned previously use SeRQL "under the hood" through Sesame remote access API. Upon request this will be shown in detail.

4 Projects using Sesame

Sesame has been successfully used in a variety of research and commercial projects. We briefly name a few of these projects:

- **On-To-Knowledge** [3] (<http://www.ontoknowledge.org/>), A European IST project about knowledge management and evolving ontologies. Sesame acts as the central hub in the project framework, that is, all the tools (editors, search engine, interfaces, etc.) communicate through data exchange with Sesame.
- **DOPE** [6] (<http://www.aduna.biz/dope/>), Drug Ontology Project for Elsevier, is a project about thesaurus based integration and search of heterogeneous data sources about scientific articles. In this project, Sesame is deployed as a distributed storage and querying system, using graph transformation queries to map heterogeneous sources to a unified model.
- **Bibster** (<http://bibster.sourceforge.net/>), a P2P application that allows sharing of citation entries in the BibTeX format, internally uses Sesame as its storage component. Queries between Bibster peers are formulated in SeRQL.

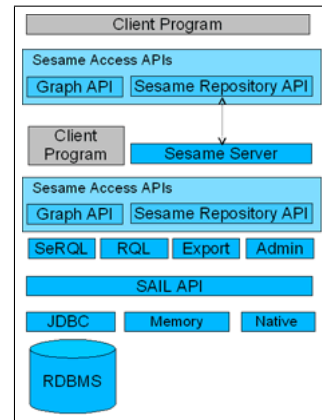


Figure 1: Sesame architecture

Other use cases and projects may be found on the website and community forum of Sesame at <http://www.openrdf.org/>.

5 Software availability and Licensing

The Sesame framework is available under the terms of the GNU Lesser General Public License (LGPL) from <http://www.openrdf.org/>. The FOAF explorer interface is currently under development but will be available from the same website.

References

- [1] Jeen Broekstra and Arjohn Kampman. The SeRQL Query Language. User manual, Aduna, 2003. See <http://www.openrdf.org/doc/SeRQLmanual.html>.
- [2] Jeen Broekstra, Arjohn Kampman, and Frank van Harmelen. Sesame: A Generic Architecture for Storing and Querying RDF and RDF Schema. In Ian Horrocks and James Hendler, editors, *Proceedings of the first International Semantic Web Conference (ISWC 2002)*, number 2342 in Lecture Notes in Computer Science, pages 54–68, Sardinia, Italy, June 9 – 12, 2002. Springer Verlag, Heidelberg Germany. See also <http://www.openrdf.org/>.
- [3] Dieter Fensel, Frank van Harmelen, Michel Klein, Hans Akkermans, Jeen Broekstra, Christiaan Fluit, Jos van der Meer, Hans-Peter Schnurr, Rudi Studer, John Hughes, Uwe Krohn, John Davies, Robert Engels, Bernt Bremdal, Fredrik Ygge, Thorsten Lau, Bernd Novotny, Ulrich Reimer, and Ian Horrocks. On-To-Knowledge: Ontology-based Tools for Knowledge Management. In *Proceedings of the eBusiness and eWork 2000 (EMMSEC 2000) Conference*, Madrid, Spain, October 18–20, 2000.
- [4] Gregory Karvounarakis, Vassilis Christophides, Dimitris Plexousakis, and Sofia Alexaki. Querying community web portals. Technical report, Institute of Computer Science, FORTH, Heraklion, Greece, 2000. See <http://www.ics.forth.gr/proj/isst/RDF/RQL/rql.pdf>.
- [5] Andy Seaborne. RDQL - A Query Language for RDF. W3c member submission, Hewlett Packard, January 2004. See <http://www.w3.org/Submission/2004/SUBM-RDQL-20040109/>.
- [6] Heiner Stuckenschmidt, Anita de Waard, Ravinder Bhogal, Christiaan Fluit, Arjohn Kampman, Jan van Buel, Erik van Mulligen, Jeen Broekstra, Ian Crowlesmith, Frank van Harmelen, and Tony Scerri. Exploring Large Document Repositories with RDF Technology - The DOPE Project. *IEEE Intelligent Systems - Special Issue on the Semantic Web Challenge*, 2004.