

SWAD–Europe Deliverable 10.3: Tools for Semantic Web Scalability and Storage : Implementation report on scalable Free Software/Open Source RDF storage system

Project name:

W3C Semantic Web Advanced Development for Europe (SWAD-Europe)

Project Number:

IST-2001-34732

Workpackage name:

Workpackage description: 10: Tools for Semantic Web Scalability and Storage

Workpackage description:

<http://www.w3.org/2001/sw/Europe/plan/workpackages/live/esw-wp-10.html>

Deliverable title:

Tools for Semantic Web Scalability and Storage: Implementation report on scalable Free Software/Open Source RDF storage system.

URI:

http://www.w3.org/2001/sw/Europe/reports/rdf_scalable_storage_impl/

Outcome URI:

<http://www.w3.org/2001/sw/Europe/>

Authors:

Dave Beckett

Abstract:

A Free Software/Open Source RDF storage system along, with report and documentation on implementation issues found, for a developer audience.

STATUS:

Completed 2003-03-07.

Contents

- 1 [Introduction](#)
 - 2 [Redland and Raptor](#)
 - 3 [Semantic Web Data Scalability problems](#)
 - 4 [A solution for part of the problem - Redland Contexts](#)
 - A [References](#)
 - B [Changes](#)
-

1 Introduction

This deliverable reports on work done on implementing a scalable and efficient semantic web storage system based on the *Redland RDF Application Framework* [\[REDLAND-DI\]](#) developed at ILRT, University of Bristol over several years. The scalability requirement needed developing a major new feature - *contexts* and consequent re-design and implementation of core parts of Redland to enable it.

2 Redland and Raptor

Redland [\[REDLAND-SW\]](#) as described in *Designing and Building the Redland RDF Application Framework* [\[REDLAND\]](#) is a C library implementing the RDF triple-based graph model designed for portability, flexibility and performance. It has been developed since 2000 and the C API [\[REDLAND-API\]](#) now has several high-level language interfaces which allow rapid development of RDF systems using the API:

- Redland Java API [\[REDLAND-JAVA-API\]](#)
- Redland Perl API [\[REDLAND-PERL-API\]](#)
- Redland Python [\[REDLAND-PYTHON-API\]](#) and in Pydoc [\[REDLAND-PYTHON-PYDOC\]](#)
- Redland Ruby API [\[REDLAND-RUBY-API\]](#)
- Redland PHP API [\[REDLAND-PHP-API\]](#)
- Redland Tcl API [\[REDLAND-TCL-API\]](#)

Redland uses a related but separate library *Raptor* [\[RAPTOR-SW\]](#) also developed at ILRT which deals with the parsing of RDF syntaxes - RDF/XML [\[RDFXML\]](#) and N-Triples [\[N-TRIPLES\]](#). These two syntaxes are

managed by the *World Wide Web Consortium (W3C)* [W3C] by the *RDF Core Working Group (RDF Core)* [RDF-Core] and the two draft standards are co-/edited by this report author who has been a member of that group since May 2000. Redland and Raptor have also provided implementation experience and feedback to this W3C standardisation activity in terms of what could be designed and implemented efficiently.

Redland was designed to implement the *Web Search Environments Project* [WSE] for ILRT, as described in *Web Search Environments - Web Crawling High-Quality Metadata using RDF and Dublin Core* [WSE-PAPER]. This is still under development as a web searching and semantic web project for feedback into future digital library developments.

There are several other known *Redland and Raptor applications* [REDLAND-APPS] and others not publically announced, since they are free software requiring no registration or notification.

The most recent prominent use of Redland one is the server part of the *MEG Registry Project* [MEG-REGISTRY]. *The MEG Registry and SCART: Complementary Tools for Creation, Discovery and Re-use of Metadata Schemas* [MEG-PAPER] which created an educational metadata schema registry server and client using Redland for the server. This has also been further developed for the *CORES Registry* [CORES-REGISTRY] for the *CORES Project* funded by IST under KA3.

Edd Dumbill created a software agent *FOAFBot* [FOAFBot] using Redland as described in *Finding friends with XML and RDF* [FOAFBot-ART] to aggregate personal RDF descriptions in a vocabulary called Friend of a Friend (FOAF) collected as the agent wandered the semantic web of relationships between people and their resources (documents, interests, ...). Each of the items being read from the different web sites was tracked and checked with a digital signature and extra effort was made to enable updates of particular RDF sites to work more efficiently.

3 Semantic Web Data Scalability problems

Most of the projects described above use Redland to manipulate and especially aggregate semantic web data from various sources. This meant managing a large aggregate graph built from descriptions taken from various other places, that can be updated at different rates. This graph merging/updating problem has been variously solved by throwing away all the data and building a new graph from the aggregate - but this is not scalable or efficient. It was required that the triples were in some form tracked so that when the original source modified them, the older triples could be removed and updated, without requiring a re-indexing of the entire data set. This has been called tracking the *provenance* of the data.

In order to mark each triple from a source with its original location, that could be done by using RDF reification on the triples, which turns each triple into 4, giving them each a unique identifier. This would make the scalability problem even worse. This does have the advantage of being a solution inside the RDF graph, where all the provenance information could be tracked. It was therefore necessary to consider outside-RDF graph solutions to solve this scalability problem.

4 A solution for part of the problem – Redland Contexts

This feature allows a Node to be given whenever a statement is added to a model which can be retrieved from any model query that returns answers as an Iterator of Nodes or a Stream of Statements. Both of these classes gained a new method `get_context` that returns the original Node that was given when the statement corresponding to the answer was added to the model.

The context node can also be used to add and remove sets of statements to/from a model, and each statement with a given context node can be listed as a stream of statements.

Adding this feature required substantial internal changes to these two classes and the internal storage apis and implementations along with moderate code changes at the application level, which are described below.

This feature can be used for the following (not an exhaustive list):

- Enable true graph merging / updating / demerging - identify the subgraphs with context nodes.
- Statement Identity - add each statement with a different context node
- Statement Provenance - use the context node as the subject of other statements about the statement that is returned.

A References

[REDLAND-SW]

[Redland RDF Application Framework](#), Dave Beckett, ILRT, University of Bristol

[REDLAND-DI]

[Design and Implementation of the Redland RDF Application Framework](#), Dave Beckett, ILRT, University of Bristol, Proceedings of 10th World Wide Web Conference ([WWW10](#)), 2001, Hong Kong, ACM Press, 449 - 456, ISBN 1-58113-348-0

[REDLAND-API]

[Redland API Reference](#), Dave Beckett, ILRT, University of Bristol

[REDLAND-JAVA-API]

[Redland Java API](#), Dave Beckett, ILRT, University of Bristol

[REDLAND-PERL-API]

- [Redland Perl API](#), Dave Beckett, ILRT, University of Bristol
[REDLAND-PYTHON-API]
- [Redland Python API](#), Dave Beckett, ILRT, University of Bristol
[REDLAND-PYTHON-PYDOC]
- [Redland Python Pydoc API](#), Dave Beckett, ILRT, University of Bristol
[REDLAND-RUBY-API]
- [Redland Ruby API](#), Dave Beckett, ILRT, University of Bristol
[REDLAND-PHP-API]
- [Redland PHP API](#), Dave Beckett, ILRT, University of Bristol
[REDLAND-TCL-API]
- [Redland Tcl API](#), Dave Beckett, ILRT, University of Bristol
[RAPTOR-SW]
- [Raptor RDF Parser Toolkit](#), Dave Beckett, ILRT, University of Bristol
[REDLAND-APPS]
- [Redland and Raptor applications](#), Dave Beckett, ILRT, University of Bristol
[MEG-REGISTRY]
- [MEG Registry Project](#), UKOLN, University of Bath, UK / ILRT, University of Bristol, UK
[CORES-REGISTRY]
- [CORES Registry Project](#) part of [CORES Project](#), IST
[MEG-WORKSHOP]
- [MEG Metadata Schemas Registry Workshop](#), 21 January 2003, UKOLN, University of Bath, UK
[MEG-PAPER]
- [The MEG Registry and SCART: Complementary Tools for Creation, Discovery and Re-use of Metadata Schemas](#), Rachel Heery, UKOLN; Pete Johnston, UKOLN; Dave Beckett, ILRT, University of Bristol; Damian Steer, ILRT, University of Bristol, proceedings of Dublin Core Conference 2002 (DC 2002), Floreny, October 13-17, 2002.
[RDFXML]
- [RDF/XML Syntax Specification \(Revised\)](#), Dave Beckett (editor), W3C Working Draft, 23 January 2003, work in progress. See also the [latest version of RDF/XML Syntax Specification](#)
- [N-TRIPLES]
- [N-Triples](#) in [RDF Test Cases](#), Dave Beckett, Jan Grant (eds.), W3C Working Draft, 23 January 2003, work in progress. See also the [latest version of RDF Test Cases](#)
- [WSE]
- [WSE](#), Dave Beckett, ILRT, University of Bristol
[WSE-PAPER]
- [Web Search Environments - Web Crawling High-Quality Metadata using RDF and Dublin Core](#) presented at [WWW2002](#), Hawaii, May 2002.
[W3C]
- [World Wide Web Consortium \(W3C\)](#)
[RDF-Core]
- [RDF Core Working Group](#), World Wide Web Consortium (W3C)
[FOAFBot]
- [FOAFBot: IRC Community Support Agent](#), Edd Dumbill
[FOAFBot-ART]
- [Finding friends with XML and RDF](#), Edd Dumbill, XML Watch, [IBM developerWorks](#), June 2002.