MarmottaProposal

Apache Marmotta incubation proposal

Status

Proposal has been accepted, further details at http://incubator.apache.org/projects/marmotta.html

Abstract

Marmotta is a Linked Data platform for industry-strength installations.

Proposal

The goal of Apache Marmotta is to provide an open implementation of a Linked Data Platform that can be used, extended, and deployed easily by organizations who want to publish Linked Data or build custom applications on Linked Data.

The phrase "Linked Data" is used here idiosyncratically to refer to a data integration paradigm across the Web. The term was coined by Tim Berners-Lee in 2006, and it is based on four very simple principles which basically describe recommended best practices for exposing, sharing, and connecting pieces of data, information, and knowledge on the Semantic Web using URIs and the RDF technology stack. Therefore Linked Data is about using the Web to connect related data that wasn't previously linked, or using the Web to lower the barriers to linking data currently linked using other methods.

Marmotta will follow the core recommendations of the W3C on RDF, SPARQL and Linked Data publishing, particularly the emerging Linked Data Platform (LDP) recommendation. It will also offer extensions for frequently needed additional functionalities like Linked Data Querying, WebID, WebACL, Reasoning, and Versioning. Marmotta aims to cover both, Linked Open Data, as well as Enterprise Linked Data scenarios, providing data governance facilities to deal with different data sources and requirements (small data/big data, open access/restricted access, etc).

Background

The Semantic Web isn't just about putting data on the web. It is about making links, so that a person or machine can explore the web of data. Moreover, the Web has quickly evolved to a Read-Write paradigm, and Linked Data technologies too. And Marmotta will address this challenge and offer a common infrastructure for organizations working in this area.

Marmotta comes as a continuation of the work in the Linked Media Framework (aka LMF) project. LMF is an easy-to-setup server application that bundles central Semantic Web technologies to offer some advanced services. The Linked Media Framework consists of LMF Core which provides a Read-Write Linked Data server, plus some modules that complement the server with other added added capabilities, such as, SPARQL 1.1, LDPath, LDCache, Reasoning, Versioning, etc. Besides, LMF also provides a Client Library, currently available in Java, PHP, and Javascript, as a convenient API abstraction around the LMF web services. Currently LMF integrates with other relevant tools (Apache Stanbol, Google Refine or Drupal) to cover a wider range of use cases and needs.

Rationale

Linked Data technologies are now at a turning point from mostly research projects to industrial applications, and a lot of standardisation is currently in progress. Industrial applications require a reliable and scalable infrastructure that follows and helps defining a standard way of publishing and consuming Linked Data on the Web. The proposers have a strong background in building such applications and have invested considerable effort in the last years to building up an initial version of such a platform (the “Linked Media Framework” or “LMF”). Starting from this solid base, we strongly believe that Apache is the right environment to open the development of this project to a wider scope.

Marmotta has the potential of being a reference implementation and Apache provides a better environment for a collaborative development effort. With its well-established governance model based on meritocracy and handling IP/legal issues, people from different organizations can more easily contribute to the project. This will help unify the efforts of people implementing the Linked Data Platform specification and other Semantic Web standards. In addition, it would considerably help organizations in adopting Linked Data technologies and would provide a solid base for further research activities in the community.

Initial Goals

- Foster the use of Semantic Web Technologies in industry
- Provide an open source and community-driven implementation of a Linked Data Platform and related Semantic Web standards, LDP 1.0 Draft and SPARQL 1.1 mainly
- Move the existing LMF source from the current Google Code page to the Apache infrastructure
- Remove LMF extensions that are not relevant for a core Linked Data platform (e.g. semantic search and content enhancement)
- Define a plugable architecture for providing a data governance framework for enterprise legacy sources
- Revise the architecture, moving to a non-proprietary RDF API (Sesame or Jena) and deciding whether to move to OSGi/Felix or stay with CDI /JavaEE as SOA framework
- Identify and replace dependencies with a non-compatible license (e.g. replace XOM with JDOM)

Current Status

The source for the current LMF is a stable software artifact that, having emerged from research circles, has already a relevant number of real world installations i.e. Red Bull Media House, Salzburger Nachrichten, derStandard.at, etc.

Meritocracy

LMF is the outcome of a number of research projects coordinated/participated by Salzburg Research during the last five years. The original developers are still part of the core development team, while at the same time many new committers have joined the team. Taking this step we have made it clear to our community that going forward, the community, rather than a single organization, will determine the future of Marmotta.

Meritocracy is inherent in the research community we come from, and since Apache Marmotta aims to be a unifying project for this community it is only natural to continue this approach.

Community

Marmotta addresses two target communities: On the one hand, researchers/developers who are working with Semantic Web technologies. On the other hand, companies or organizations that require Semantic Web infrastructure. The initial committers are active participants in both communities.

Core Developers

- Sebastian Schaffert (sebastian dot schaffert at salzburgresearch dot at)
- Thomas Kurz (thomas dot kurz at salzburgresearch dot at)
- Jakob Frank (jakob dot frank at salzburgresearch dot at)
- Dietmar Glachs (dietmar dot glachs at salzburgresearch dot at)
- Sergio Fernández (sergio dot fernandez at salzburgresearch dot at)

Alignment

Marmotta complements and integrates well with the current landscape of Apache projects, especially with the emerging “semantic technologies” cluster within the ASF. Concretely, Marmotta will align with the following projects:

- Apache Commons (lang, loggging, http and so on) is extensively used in many part of the project
- Apache Tomcat is currently the primary platform for deployment; with Marmotta, Tomcat can be turned into a Linked Data server
- Apache Stanbol will very likely adopt parts of the Marmotta infrastructure, particularly for implementing the entity hub and for exposing the RDF data as Linked Data
- Apache Jena could become the RDF API used throughout Marmotta; an architectural decision is yet to be taken
- Apache Any23 could be integrated in the LMF as wrapper around non-RDF data sources to consume them as Linked Data; a similar approach has already been taken by the LMF
- Apache Tika could be use for metada extraction of content

- Apache Karaf and Apache Felix could become the OSGi container for running and configuring the Marmotta components

In addition to these more-or-less concrete proposals, there are some options that still require some strategic decisions. For example, it make make sense to build a storage backend based on Apache Hadoop for large-scale installations using HBase (e.g. jena grande, h2rdf, hdrs, hadoop rdf). Several extensions also build on existing Apache projects, most importantly the LMF Semantic Search component, which offers semantic search over Linked Data resources.

Known Risks

Probably one of the major risks will not be able to engage the community for addressing the new challenges. Knowing this, we will do our best to provide the greater facilities to attract new developers and organizations. In particular, we will try to actively engage developers from the Linked Data community through our networks.

Orphaned Products

The current project is part of the business portfolio and a strategic project of the contributor organization, and will continue in that way. So there is no risk of any of the usual warning signs of orphaned or abandoned code.

Inexperience with Open Source

The committers have large experience with open source development and communities. Several of the key committers have been actively involved in Open Source projects for more than 10-15 years. The initial code base of Marmotta has already been developed as Open Source project in the last 5 years.

Homogeneous Developers

Because we are aware about the initial list of committers is not the best for a long, it exists a strong commitment to spread the project creating a much more diverse development team. Part of the reason to enter the Apache incubation process is to open up the development to more interested participants.

Reliance on Salaried Developers

Right now most or all of that work is salaried, but the developers are identifying themselves very much with the project. When opening up the development using Apache as a platform, we expect that the future development will occur on both salaried and volunteer time, particularly by participants from the Linked Data community.

Relationships with Other Apache Projects

Although current RDF/SPARQL support in LMF is build on top of OpenRDF Sesame API, Marmotta is closely related to many Apache projects, such as Stanbol, Jena and Any23. See “Alignment” above.

An Excessive Fascination with the Apache Brand

While we expect the Apache brand may help attract more contributors, our interests in starting this project is based on the factors mentioned in the Rationale section.

Documentation

Documentation for the current project can be found at:

- http://lmf.googlecode.com
- http://doc.lmf.googlecode.com/hg/api/index.html
- http://doc.lmf.googlecode.com/hg/rest/index.html
- http://doc.lmf.googlecode.com/hg/client/index.html

Initial Source

LMF (formerly KiWi) has been developed since 2008. It is important to say that the whole LMF will not be contributed to Marmotta, actually only those parts that make up the "Linked Data Platform" functionality (Linked Data Server, RDF Store, SPARQL, LDCache, Versioning, Reasoner and LDPath) . The idea is to focus Marmotta much more in the core needs, keeping all surrounding functionality (Media-related modules and Semantic Search, basically) out of the initial scope. Although the community will be who ultimately decides what are the relevant modules. Since LMF is a very modular software artifact it will be pretty easy to make such partitioning to kick-off Marmotta.

The current source code can be found at Google Code: http://lmf.googlecode.com

Source and Intellectual Property Submission Plan

Salzburg Research Forschungsgesellschaft mbH is the sole copyright owner of the initial code to be contributed, so should not be any problem with the standard IP clearance process. Current licence is already Apache Software License 2.0.

External Dependencies

Most of current dependencies should have Apache compatible licenses, including BSD, CDDL, CPL, MPL and MIT licensed dependencies. We are aware of some incompatible licenses right now, but we will work to solve this issue. See Appendix A for a detailed list of dependencies.

Cryptography

Does Not Apply.

Required Resources

Mailing lists

- marmotta-dev
- marmotta-commits
- marmotta-users

Repository

- git://git.apache.org/marmotta.git

Issue Tracking

- Jira: MARMOTTA (Kanban board enabled at GreenHopper)

Other Resources

- Jenkins/Hudson for builds and test running.
- Wiki for internal documentation purposes
- Blog to improve the project dissemination

Initial Committers

- Sebastian Schaffert (sebastian dot schafftert at salzburgresearch dot at)
- Thomas Kurz (thomas dot kurz at salzburgresearch dot at)
- Jakob Frank (jakob dot frank at salzburgresearch dot at)
- Dietmar Glachs (dietmar dot glachs at salzburgresearch dot at)
- Sergio Fernández (sergio dot fernandez at salzburgresearch dot at)
- Rupert Westenthaler (rwesten at apache dot org)

Affiliations

All initial committers are currently affiliated to Salzburg Research Forschungsgesellschaft mbH.

Sponsors

Champion

- Andy Seaborne (andy at apache dot org)

Nominated Mentors

- Fabian Christ (fchrist at apache dot org)
- Nandana Mihindukulasooriya (nandana at apache dot org)
- Andy Seaborne (andy at apache dot org)

Sponsoring Entity

Apache Incubator PMC

Appendix A: list of dependencies

Here's the list of Maven artifacts of the current dependencies, omitting org.apache.* and commons-* groupIds, but including transitive dependencies:

```
antlr:antlr
asm:asm-analysis
```

```
asm:asm-commons
asm:asm
asm:asm-tree
asm:asm-util
backport-util-concurrent:backport-util-concurrent
c3p0:c3p0
ch.qos.cal10n:cal10n-api
ch.qos.logback:logback-classic
ch.qos.logback:logback-core
classworlds:classworlds
com.amazonaws:aws-java-sdk
com.ezware.oxbow:task-dialog
com.googlecode.jatl:jatl
com.google.code.tempus-fugit:tempus-fugit
com.google.guava:guava
com.h2database:h2
com.jayway.jsonpath:json-path
com.jayway.restassured:rest-assured
com.jcabi:jcabi-aether
com.jcabi:jcabi-aspects
com.jcabi:jcabi-log
com.miglayout:miglayout
com.ning:async-http-client
com.unboundid:unboundid-ldapsdk
dfki.km.json:jsonld-java
dom4j:dom4j
eu.medsea.mimeutil:mime-util
fi.tikesos:rdfa-core
info.aduna.commons:aduna-commons-concurrent
info.aduna.commons:aduna-commons-io
info.aduna.commons:aduna-commons-iteration
info.aduna.commons:aduna-commons-lang
info.aduna.commons:aduna-commons-net
info.aduna.commons:aduna-commons-text
info.aduna.commons:aduna-commons-xml
jakarta-regexp:jakarta-regexp
javassist:javassist
javax.activation:activation
javax.annotation:jsr250-api
javax.el:el-api
javax.enterprise:cdi-api
javax.inject:javax.inject
javax.servlet.jsp:jsp-api
javax.servlet:servlet-api
javax.validation:validation-api
jaxen:jaxen
junit:junit
log4j:log4j
mysql:mysql-connector-java
net.jcip:jcip-annotations
net.minidev:json-smart
net.sf.ehcache:ehcache-core
net.sf.opencsv:opencsv
org.aspectj:aspectjrt
org.ccil.cowan.tagsoup:tagsoup
org.codehaus.groovy:groovy
org.codehaus.izpack:izpack-standalone-compiler
org.codehaus.jackson:jackson-core-asl
org.codehaus.jackson:jackson-jaxrs
org.codehaus.jackson:jackson-mapper-asl
org.codehaus.jackson:jackson-xc
org.codehaus.janino:commons-compiler
org.codehaus.janino:janino
org.codehaus.plexus:plexus-classworlds
org.codehaus.plexus:plexus-component-annotations
org.codehaus.plexus:plexus-container-default
org.codehaus.plexus:plexus-interpolation
org.codehaus.plexus:plexus-utils
org.codehaus.woodstox:wstx-asl
org.freemarker:freemarker
org.fusesource.jdbm:jdbm
org.geonames:geonames
org.hamcrest:hamcrest-core
org.hamcrest:hamcrest-library
org.hibernate.common:hibernate-commons-annotations
org.hibernate:hibernate-c3p0
org.hibernate:hibernate-core
org.hibernate:hibernate-ehcache
org.hibernate:hibernate-entitymanager
org.hibernate:hibernate-validator
org.hibernate.javax.persistence:hibernate-jpa-2.0-api
org.javassist:javassist
org.jboss.interceptor:jboss-interceptor-core
org.jboss.interceptor:jboss-interceptor-spi
org.jboss.logging:jboss-logging
org.jboss.netty:netty
org.jboss.resteasy:jaxrs-api
org.jboss.resteasy:resteasy-cdi
org.jboss.resteasy:resteasy-jackson-provider
org.jboss.resteasy:resteasy-jaxrs
org.jboss.spec.javax.interceptor:jboss-interceptors-api_1.1_spec
org.jboss.spec.javax.transaction:jboss-transaction-api_1.1_spec
org.jboss.weld.servlet:weld-servlet-core
org.jboss.weld.se:weld-se-core
org.jboss.weld:weld-api
org.jboss.weld:weld-core
org.jboss.weld:weld-spi
org.jdom:jdom2
org.jooq:jooq
org.json:json
org.jsoup:jsoup
org.kuali.common:kuali-threads
org.kuali.maven.wagons:maven-s3-wagon
org.mnode.ical4j:ical4j
org.mnode.ical4j:ical4j-vcard
org.mortbay.jetty:jetty-embedded
org.mortbay.jetty:jetty
org.mortbay.jetty:jetty-sslengine
org.mortbay.jetty:jetty-util
org.mortbay.jetty:servlet-api
org.openrdf.sesame:sesame-http-client
org.openrdf.sesame:sesame-http-protocol
org.openrdf.sesame:sesame-model
org.openrdf.sesame:sesame-queryalgebra-evaluation
org.openrdf.sesame:sesame-queryalgebra-model
org.openrdf.sesame:sesame-query
org.openrdf.sesame:sesame-queryparser-api
org.openrdf.sesame:sesame-queryparser-serql
org.openrdf.sesame:sesame-queryparser-sparql
org.openrdf.sesame:sesame-queryresultio-api
org.openrdf.sesame:sesame-queryresultio-sparqljson
org.openrdf.sesame:sesame-queryresultio-sparqlxml
org.openrdf.sesame:sesame-queryresultio-text
org.openrdf.sesame:sesame-repository-api
org.openrdf.sesame:sesame-repository-event
org.openrdf.sesame:sesame-repository-sail
org.openrdf.sesame:sesame-repository-sparql
org.openrdf.sesame:sesame-rio-api
org.openrdf.sesame:sesame-rio-n3
org.openrdf.sesame:sesame-rio-ntriples
org.openrdf.sesame:sesame-rio-rdfxml
org.openrdf.sesame:sesame-rio-trig
org.openrdf.sesame:sesame-rio-trix
org.openrdf.sesame:sesame-rio-turtle
org.openrdf.sesame:sesame-sail-api
org.openrdf.sesame:sesame-sail-inferencer
org.openrdf.sesame:sesame-sail-memory
org.openrdf.sesame:sesame-sail-nativerdf
org.openrdf.sesame:sesame-util
org.quartz-scheduler:quartz
org.rometools:rome
org.rometools:rome-modules
org.scannotation:scannotation
org.slf4j:jcl-over-slf4j
org.slf4j:jul-to-slf4j
org.slf4j:log4j-over-slf4j
org.slf4j:slf4j-api
org.slf4j:slf4j-ext
org.slf4j:slf4j-log4j12
org.sonatype.aether:aether-api
org.sonatype.aether:aether-connector-asynchttpclient
org.sonatype.aether:aether-connector-file
org.sonatype.aether:aether-connector-wagon
org.sonatype.aether:aether-impl
org.sonatype.aether:aether-spi
org.sonatype.aether:aether-util
org.sonatype.plexus:plexus-cipher
org.sonatype.plexus:plexus-sec-dispatcher
org.sonatype.sisu:sisu-guava
org.sonatype.sisu:sisu-guice
org.sonatype.sisu:sisu-inject-bean
org.sonatype.sisu:sisu-inject-plexus
org.webjars:jquery
postgresql:postgresql
regexp:regexp
xerces:xercesImpl
xml-apis:xml-apis
xom:xom
```