



Handling RDF data with Apache Jena

Paolo Castagna

Thursday, 10th November 2011

Agenda

- Introduction to RDF and SPARQL
- Apache Jena
 - Architecture
 - Examples: input/output, storage, query, ...
- Contacts and Q&A

Resource Description Framework (RDF)

subject		object	



- Schema-less, data model is a graph:
 - subject can be a URI or a blank node
 - predicate is a URI
 - object can be a URI, a literal or a blank node

Turtle

```
@prefix      :    <http://example.org/> .
@prefix foaf:  <http://xmlns.com/foaf/0.1/> .

:alice
  a          foaf:Person ;
  foaf:name  "Alice" ;
  foaf:mbox  <mailto:alice@example.org> ;
  foaf:knows :bob ;
  foaf:knows :charlie ;
  foaf:knows :snoopy .

:bob
  foaf:name  "Bob" ;
  foaf:knows :charlie .

:charlie
  foaf:name  "Charlie" ;
  foaf:knows :alice .
```

Want more on Turtle?

<http://www.w3.org/TeamSubmission/turtle/>

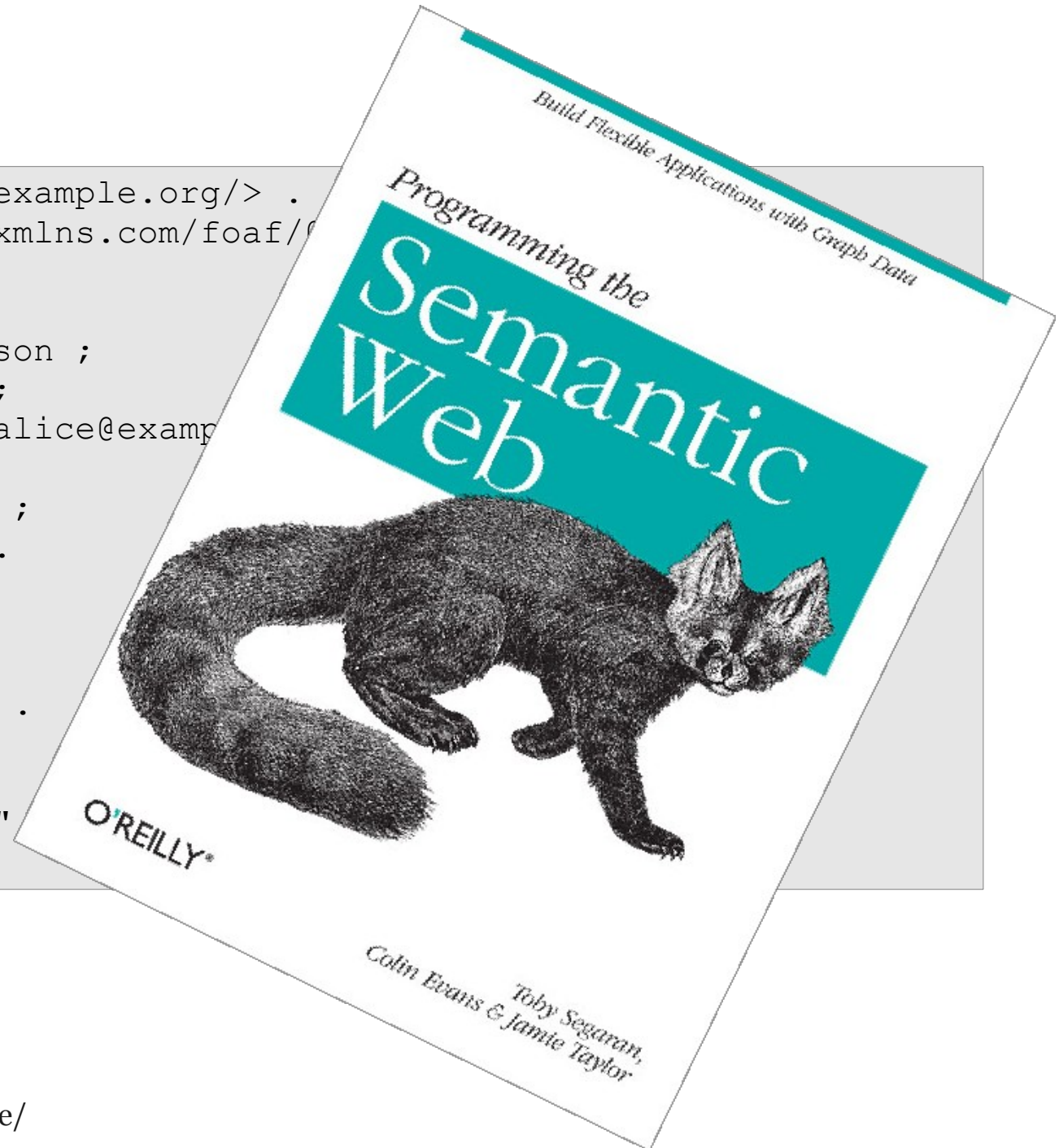
Turtle

```
@prefix      : <http://example.org/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

:alice
  a          foaf:Person ;
  foaf:name  "Alice" ;
  foaf:mbox  <mailto:alice@example.org> ;
  foaf:knows :bob ;
  foaf:knows :charlie ;
  foaf:knows :snoopy .

:bob
  foaf:name  "Bob" ;
  foaf:knows :charlie .

:charlie
  foaf:name  "Charlie" ;
  foaf:knows :alice .
```



Want more on Turtle?

<http://www.w3.org/TeamSubmission/turtle/>

SPARQL

RDF data in Turtle format:

```
@prefix      :      <http://example.org/> .
@prefix foaf:  <http://xmlns.com/foaf/0.1/> .

:alice foaf:name      "Alice" .
:alice foaf:mbox      <mailto:alice@example.org> .
:bob   foaf:name      "Bob" .
```

SPARQL query:

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>

SELECT ?name
WHERE {
    ?person foaf:mbox <mailto:alice@example.org> .
    ?person foaf:name ?name .
}
```

Results:

```
-----
| name      |
=====
| "Alice"   |
-----
```

Want more on SPARQL?

<http://www.w3.org/TR/sparql11-query/>

SPARQL

RDF data in Turtle format:

```
@prefix      :      <http://example.org/> .
@prefix foaf:  <http://xmlns.com/foaf/0.1/>

:alice foaf:name      "Alice" .
:alice foaf:mbox      <mailto:alice@...> .
:bob   foaf:name      "Bob" .
```

SPARQL query:

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>

SELECT ?name
WHERE {
  ?person foaf:mbox <mailto:alice@...> .
  ?person foaf:name ?name .
}
```

Results:

```
-----
| name      |
=====
| "Alice"   |
-----
```



Want more on SPARQL?

<http://www.w3.org/TR/sparql11-query/>

Jena History

From: McBride, Brian <bwm@hplb.hpl.hp.com>
Date: Mon, 28 **Aug 2000** 13:40:03 +0100
To: "RDF Interest (E-mail)" <www-rdf-interest@w3.org>

A few weeks ago I posted some suggestions for an improved **java RDF API**.
[...]

I think better tools will help encourage the adoption of RDF. I'm looking on this as an experiment to see whether this API brings any benefits. So I'd like some feedback.

Brian McBride
HPLabs

From: Dan Brickley <Daniel.Brickley@bristol.ac.uk>
Date: Mon, 28 Aug 2000 17:09:20 +0100 (BST)
To: "McBride, Brian" <bwm@hplb.hpl.hp.com>
cc: "RDF Interest (E-mail)" <www-rdf-interest@w3.org>

Nice work Brian :-)

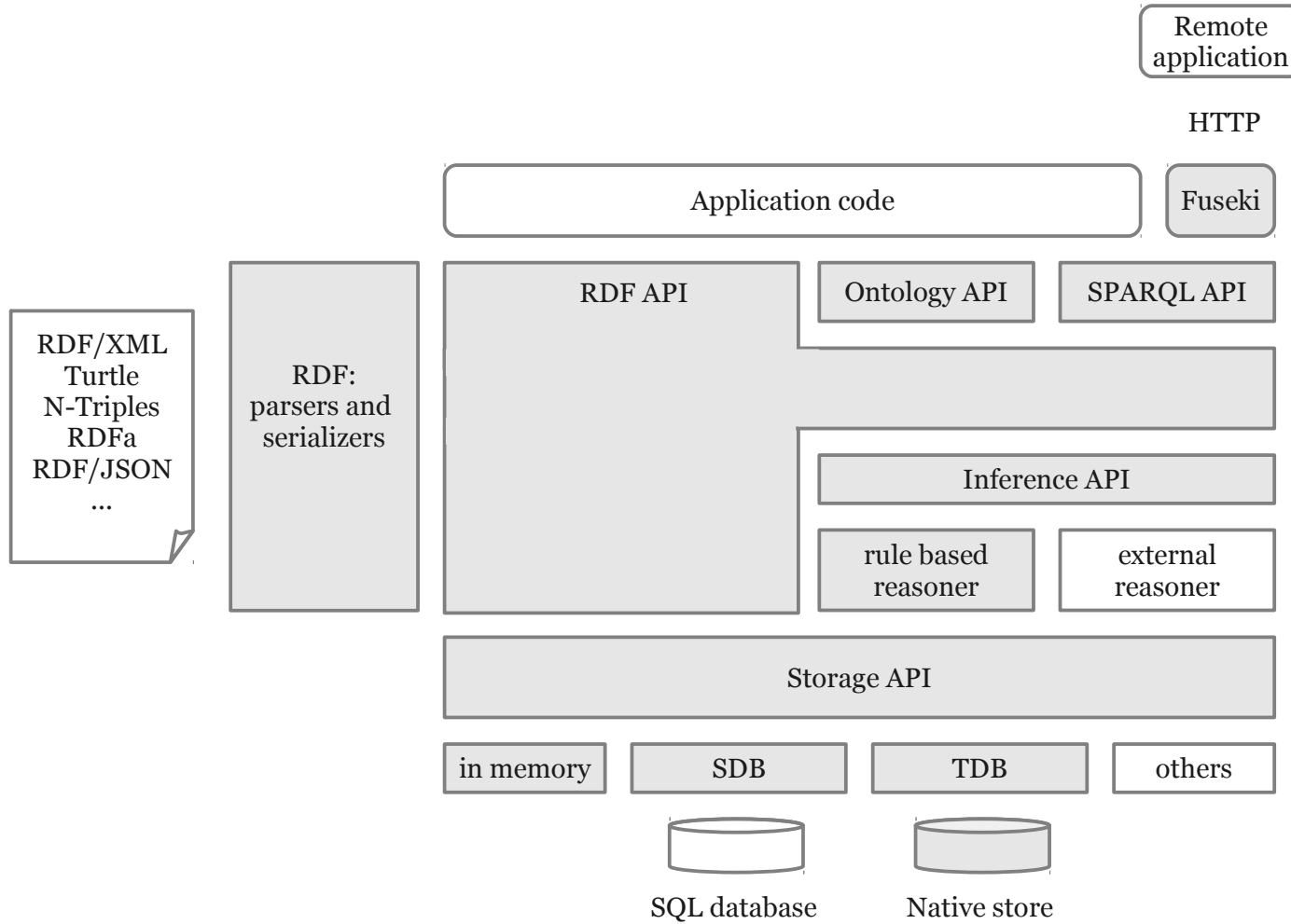
(Yet another **bristol** RDF implementation -- must be something in the Water!)

[...]

Apache Jena

- Since the beginning: open source, BSD
- Entered Apache incubator 23rd November 2010
- Work in progress:
 - first Apache release
 - community, new committers and PMC members
 - documentation and new website
 - bug fixing and small improvements (as usual)
 - upcoming W3C related recommendations (as usual)
 - ...
- Source code here:
 - <https://svn.apache.org/repos/asf/incubator/jena/Jena2/>
- Maven artifacts here:
 - <http://repo1.maven.org/maven2/com/hp/hpl/jena/>
 - <https://repository.apache.org/content/repositories/snapshots/org/apache/jena/>

Architecture



RDF API

```
// create a new Jena in-memory Model
Model model = ModelFactory.createDefaultModel();

// parse InputStream as RDF in Turtle format
InputStream in = ...
model.read(in, null, "TURTLE");

// print RDF in different serialization formats
model.write(System.out, "TURTLE");
model.write(System.out, "N-TRIPLES");
model.write(System.out, "RDF/XML");
model.write(System.out, "RDF/JSON");
...

// retrieve and parse RDF from the Web
model.read("http://www.w3.org/TR/rdf-schema/rdfs-namespace", null, "RDF/XML");

...

// parse RDF retrieving it via FileManager and ClassLoader
FileManager fm = FileManager.get();
fm.addLocatorClassLoader(Example.class.getClassLoader());
Model model = fm.loadModel("data.ttl", null, "TURTLE");
```

Want more on Jena RDF APIs?

<http://incubator.apache.org/jena/tutorials/>

<http://incubator.apache.org/jena/documentation/javadoc/jena/>

SPARQL API

```
String queryString = "PREFIX foaf: <http://xmlns.com/foaf/0.1/> " +
    "SELECT ?name WHERE { " +
    "    ?person foaf:mbox <mailto:alice@example.org> . " +
    "    ?person foaf:name ?name . " +
    "}";
Query query = QueryFactory.create(queryString);
QueryExecution qexec = QueryExecutionFactory.create(query, model);
try {
    ResultSet results = qexec.execSelect();
    while ( results.hasNext() ) {
        QuerySolution soln = results.nextSolution();
        Literal name = soln.getLiteral("name");
        System.out.println(name);
    }
} finally {
    qexec.close();
}
```

SPARQL API

```
String queryString = ...
Query query = QueryFactory.create(queryString);
QueryExecution qexec = QueryExecutionFactory.create(query, model);
try {
    ResultSetRewindable results =
        ResultSetFactory.makeRewindable(qexec.execSelect());

    System.out.println("---- XML ----");
    ResultSetFormatter.outputAsXML(System.out, results);
    results.reset();

    System.out.println("\n---- CSV ----");
    ResultSetFormatter.outputAsCSV(System.out, results);
    results.reset();

    System.out.println("\n---- TSV ----");
    ResultSetFormatter.outputAsTSV(System.out, results);
    results.reset();

    System.out.println("\n---- JSON ----");
    ResultSetFormatter.outputAsJSON(System.out, results);
} finally {
    qexec.close();
}
```

Want more on Jena SPARQL (i.e. ARQ) APIs?

http://www.openjena.org/ARQ/app_api.html

<http://incubator.apache.org/jena/documentation/javadoc/arq/index.html>

TDB

```
Location location = new Location ("/tmp/TDB");
DatasetGraphTDB dsg = TDBFactory.createDatasetGraph(location);

InputStream in = ...
TDBLoader.load(dsg, in, false);

Lock lock = dsg.getLock();
lock.enterCriticalSection(Lock.READ); // acquire a read lock
try {
    Iterator<Quad> iter = dsg.find();
    while ( iter.hasNext() ) {
        Quad quad = iter.next();
        ...
    }
} finally {
    lock.leaveCriticalSection(); // release the lock
}

...

dsg.close();
```

TDB

```
DatasetGraphTDB dsg = ...
String queryString = ...

Lock lock = dsg.getLock();
lock.enterCriticalSection(Lock.READ); // acquire a read lock
try {
    Query query = QueryFactory.create(queryString);
    QueryExecution qexec = QueryExecutionFactory.create(query, dsg.toDataset());
    try {
        ResultSet results = qexec.execSelect();
        while ( results.hasNext() ) {
            QuerySolution soln = results.nextSolution();
            Literal name = soln.getLiteral("name");
            System.out.println(name);
        }
    } finally {
        qexec.close();
    }
} finally {
    lock.leaveCriticalSection(); // release the lock
}
```

Want more on TDB?

<http://openjena.org/wiki/TDB>

<http://incubator.apache.org/jena/documentation/javadoc/tdb/index.html>

Fuseki

- Fuseki is a SPARQL server built using Jena, ARQ and TDB
- Fuseki tracks these W3C working drafts:
 - SPARQL 1.1 Query
<http://www.w3.org/TR/sparql11-query/>
 - SPARQL 1.1 Update
<http://www.w3.org/TR/sparql11-update/>
 - SPARQL 1.1 Protocol
<http://www.w3.org/TR/sparql11-protocol/>
 - SPARQL 1.1 Graph Store HTTP Protocol
<http://www.w3.org/TR/sparql11-http-rdf-update/>

```
cd /tmp
svn co https://svn.apache.org/repos/asf/incubator/jena/Jena2/Fuseki/trunk/ fuseki
cd fuseki
mvn package

java -cp target/fuseki-0.2.1-SNAPSHOT-sys.jar tdb.tdbloader --loc=/tmp/tdb books.ttl

fuseki-server --update --loc=/tmp/tdb /apachecon

s-query --service=http://127.0.0.1:3030/apachecon/sparql "SELECT * { ?s ?p ?o }"

curl -d "query=SELECT * { ?s ?p ?o }" -H "Accept: application/sparql-results+json"
http://127.0.0.1:3030/apachecon/sparql
```


Fuseki

Fuseki Query

Dataset: /apachecon

SPARQL Query

Output: XML ▼

XSLT style sheet (blank for none):

Force the accept header to text/plain regardless

Get Results

SPARQL Update

Perform update

File upload

File: Browse...

Graph:

Upload

Extension Points

- ARQ: functions, property functions, query engines, new optimization techniques, ...
- Jena readers/writers and RIOT
- External inference engines
- Alternative storage systems
- Custom/additional indexes
- ...

LARQ

```
Dataset dataset = ...
// build the Lucene index when pointed to a non existing directory
AssemblerLARQ.make(dataset, "/tmp/lucene");

String queryString =
    "PREFIX pf: <http://jena.hpl.hp.com/ARQ/property#>" +
    "PREFIX foaf: <http://xmlns.com/foaf/0.1/> " +
    "SELECT ?name ?email WHERE { " +
    "    ?person foaf:name ?name . " +
    "    ?name pf:textMatch 'A*' . " + // use the Lucene syntax here
    "    OPTIONAL { ?person foaf:mbox ?email . }" +
    "}";

Query query = QueryFactory.create(queryString);
QueryExecution qexec = QueryExecutionFactory.create(query, dataset);
try {
    ResultSet results = qexec.execSelect();
    while ( results.hasNext() ) {
        QuerySolution soln = results.nextSolution();
        ...
    }
} finally {
    qexec.close();
}
```

Want more on LARQ?

<http://incubator.apache.org/jena/documentation/larq/>

<http://incubator.apache.org/jena/documentation/javadoc/larq/index.html>

Jena, RDF and other Apache projects

- Clerezza, Stanbol, Any23, ...
 - Clerezza (incubating) is a service platform based on OSGi for managing semantically linked data accessible through RESTful web services.
 - Stanbol (incubating) is a stack of components for semantic content management.
 - Any23 (incubating) extracts and validate structured data in RDF format from a variety of Web documents and markup formats (i.e. an RDF distiller).
- LARQ: Apache Lucene + ARQ
- RDF processing with Hadoop/MapReduce
- Spatial Information System (SIS) and GeoSPARQL (?)
- How to best store RDF in HBase? Cassandra?
- ...

Contacts

- Me:
 - Paolo Castagna <paolo.castagna at talis.com>
 - Software Engineer at Talis Systems Ltd., UK
 - Sorcerer's apprentice (and committer) in the Jena team
 - Playground on GitHub: <https://github.com/castagna>
 - Interested in *crazy* ideas you might have in relation to RDF and Jena
- Other contacts:
 - <http://incubator.apache.org/jena/>
 - jena-users@incubator.apache.org
 - <https://issues.apache.org/jira/browse/JENA>
 - <irc://irc.freenode.net/#jena>
 - Twitter: #jena, @AndySeaborne, @der42, @ephemerian, @ehedgehog, ...

Questions?



shared innovation™