

Example: Jena and Fuseki



Jena and Fuseki



- **Jena** is a solid and widely used system
 - It scales reasonably well using its TDB native store
 - It has support for reasoning via a native rules engine and an API for DIG-compliant reasoners
 - There are subsets that run on Android phones
- **Fuseki** is a sparql endpoint that complements Jena
- The two are easy to install and use with Java or any other language via the API or endpoint

Download jena and fuseki



```
> curl -O http://www.apache.org/dist/jena/binaries/jena-fuseki-0.2.6-distribution.tar.gz
> curl -O http://www.apache.org/dist/jena/binaries/apache-jena-2.10.0.tar.gz
> tar -xzf jena-fuseki-0.2.6-distribution.tar.gz
> tar -xzf apache-jena-2.10.0.tar.gz
> rm *.gz
> ls
apache-jena-2.10.0 jena-fuseki-0.2.6
> export JENA=/Users/finin/.../apache-jena-2.10.0/
> export FUSEKI=/Users/finin/S.../jena-fuseki-0.2.6/
```

Start the Fuseki server



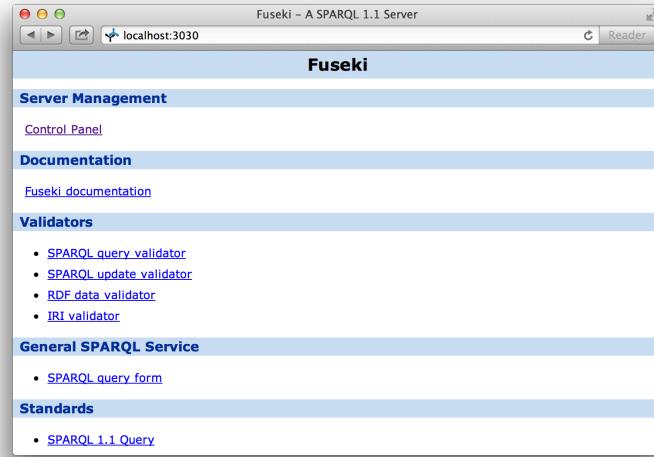
```
# create directory for the RDF data
> mkdir ~/MYTDB
# launch server allowing updates, using our data directory
# and naming the default store ds
> cd $FUSEKI
> fuseki-server --update --loc=/Users/finin/MYTDB /ds &
17:01 INFO Server :: TDB dataset: directory=/Users/finin/MYTDB
17:01 INFO Server :: Dataset path = /ds
17:01 INFO Server :: Fuseki 0.2.5 2012-10-20T17:03:29+0100
17:01 INFO Server :: Started 2013/03/31 17:01:20 EDT on port 3030
# put some data into it from a file
> s-put http://localhost:3030/ds/data default Data/books.ttl
```

Add some data 

```
> Head Data/books.ttl
@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix vcard: <http://www.w3.org/2001/vcard-rdf/3.0#> .
@prefix ns: <http://example.org/ns#> .
@prefix : <http://example.org/book/> .

> s-put http://localhost:3030/ds/data default Data/books.ttl
18:00 INFO Fuseki :: [2] PUT http://localhost:3030/ds/data?default
18:00 INFO Fuseki :: [2] 204 No Content
```

Access Fuseki via Browser 

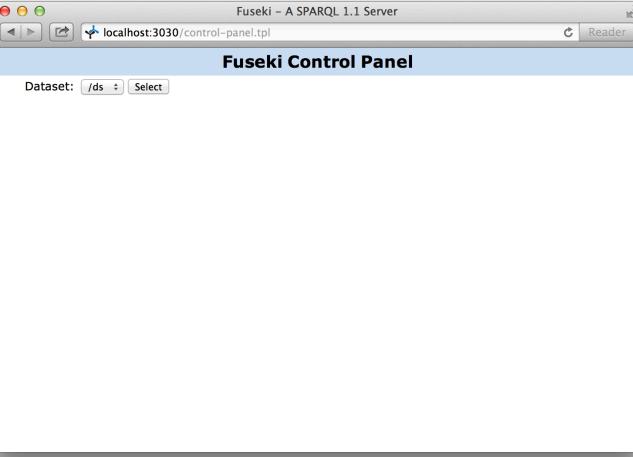


Fuseki – A SPARQL 1.1 Server

Fuseki

- Server Management**
 - [Control Panel](#)
 - [Documentation](#)
 - [Validators](#)
 - [SPARQL query validator](#)
 - [SPARQL update validator](#)
 - [RDF data validator](#)
 - [IRI validator](#)
 - General SPARQL Service**
 - [SPARQL query form](#)
 - Standards**
 - [SPARQL 1.1 Query](#)

Control panel: select the store 

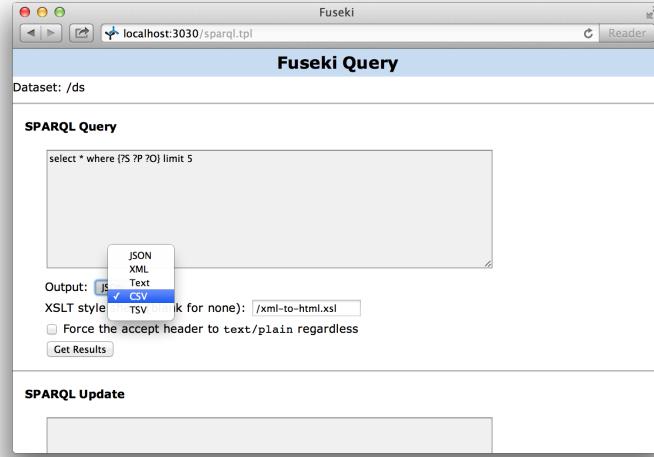


Fuseki – A SPARQL 1.1 Server

Fuseki Control Panel

Dataset: /ds

Enter a SPARQL query 



Fuseki

Fuseki Query

Dataset: /ds

SPARQL Query

```
select * where (?S ?P ?O) limit 5
```

Output: **CSV** JSON XML Text

XSLT style: (link for none): [/xml-to-html.xsl](#)

Force the accept header to text/plain regardless

SPARQL Update

Here are the results

```
S,P,O
http://example.org/book/book5,http://purl.org/dc/elements/1.1/creator,J.K. Rowling
http://example.org/book/book5,http://purl.org/dc/elements/1.1/title,Harry Potter and the Order of the Phoenix
http://example.org/book/book3,http://purl.org/dc/elements/1.1/creator,_b0
http://example.org/book/book3,http://purl.org/dc/elements/1.1/title,Harry Potter and the Prisoner Of Azkaban
http://example.org/book/book1,http://purl.org/dc/elements/1.1/creator,J.K. Rowling
```

Other interactions

- From the control panel you can also
 - Enter SPARQL update queries
 - Upload a file of RDF data into the store
- To bulk load data, use Jena's tdbloader command
 - Loads at ~50K triples/sec
 - ~ 80 minutes to ~250M triples in DBpedia's dataset