

# Linked Data-Fu Tutorial

## Part 1: Basics

Andreas Harth  
November 2014

# Contents

1. Intro
2. Linked Data Lookups & Queries
3. Hands-on Lookups & Queries
4. Rules & Link Following
5. Hands-on Rules & Link Following
6. Conclusion

Section 1

# INTRODUCTION

# Challenges

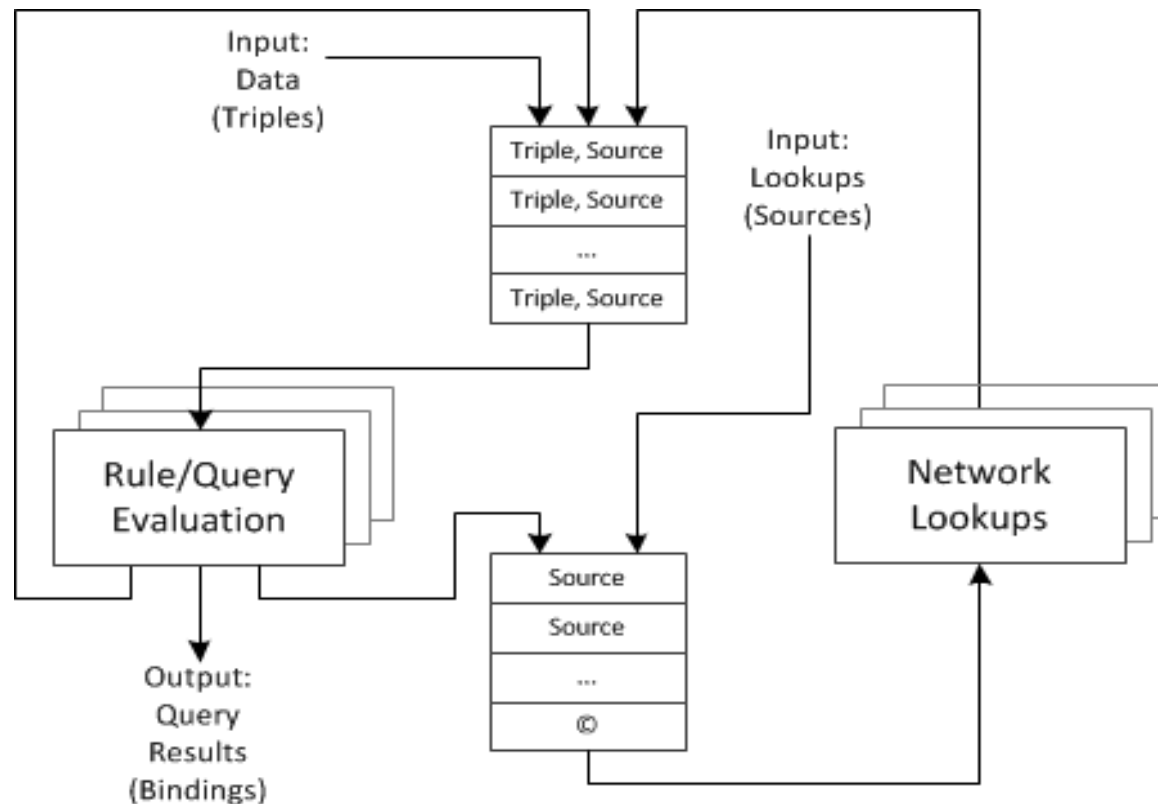
- Integration and interoperation requires combination of data in different vocabularies.
- Distributed modelling needs to take account the semantics of data items (e.g., transitivity)
- In distributed data publication, the protocol level is important (polling vs event-based processing, different performance characteristics of systems)
- Data processing and access requires parallelisation (low latency, high bandwidth)
- Termination is tricky in parallel algorithms

# Contributions

- We describe the architecture of a system which combines deduction on RDF and performing network requests via HTTP in a parallel fashion
- The presented methods are web-aware (hypermedia - link following, provenance)
- We present several algorithms for program optimisation on the logical (operator graph) and physical level (evaluator graph)
- We provide parallel rule processing methods, and means to check termination

# Architecture

- Input: Facts, Requests, Deduction Rules, Interaction Rules, Queries
- Carry out requests and process results
- Output: Query Bindings



Section 2

# **LINKED DATA LOOKUPS & QUERIES**

# Linked Data Overview

**File** <http://www.aifb.kit.edu/portal/index.php?title=Spezial:Exp...>:

@prefix aifb: <<http://www.aifb.kit.edu/id/>> .

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

aifb:Rudi\_Studer foaf:name „Rudi Studer“ .

---

**File** <http://harth.org/andreas/foaf.rdf>:

@prefix ...

<<http://harth.org/andreas/foaf#ah>>

foaf:name „Andreas Harth“ ;

foaf:knows aifb:Rudi\_Studer .



# Notation3

- Superset of Turtle (which is a superset of N-Triples)
- Specified by timbl to „easily scribble triples“
- Unicode as character encoding
- `subject predicate object .`
- Goes beyond RDF (Turtle)
  - Variables
  - Graph Quoting

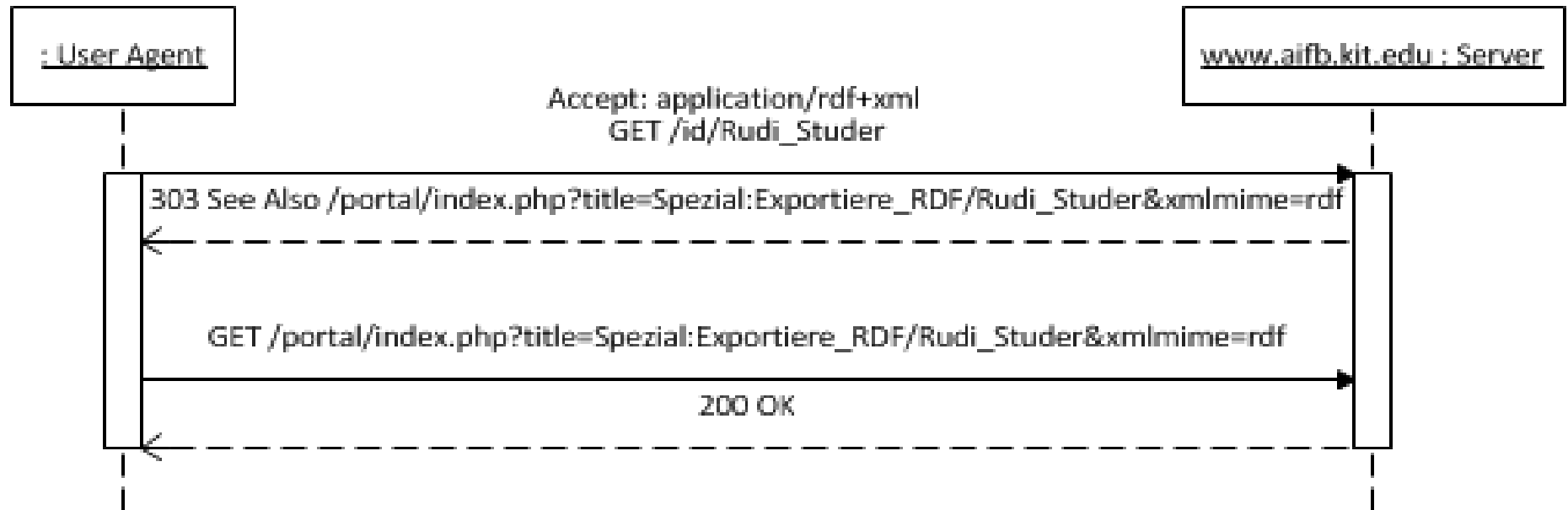
# Variables

- Variables in N3 are prefixed with ?
- E.g., ?x foaf:name ?y .

# Graph Quoting

- We can group triples with {}
- Intuitively, triples within {} are assumed to be reified triples (i.e., they are not „true“)
- However, there currently does not exist a formal definition of Notation3 graph quoting

# Things and Documents: Slash URIs



```
$ curl -v --header "Accept: application/rdf+xml"
"http://www.aifb.kit.edu/id/Rudi_Studer"
```

# HTTP Requests in Linked Data-Fu

```
@prefix http: <http://www.w3.org/2011/http#> .
```

```
@prefix httpm: <http://www.w3.org/2011/http-methods#> .
```

```
{
```

```
[ ] http:mthd httpm:GET ;
```

```
  http:requestURI <http://www.aifb.kit.edu/id/Rudi_Studer> .
```

```
}
```

# CRUD Operations

- GET, DELETE
- POST, PUT
- Send content with http:body

```
{ [] http:mthd httpm:POST ;  
  http:requestURI <http://example.org/order/> ;  
  http:body {  
    _:b rdfs:label "The Marian" ;  
    eg:isbn "0804139024" .  
  } .  
}
```

# Basic Graph Patterns

**Definition (Triple Pattern)** *Let  $\mathcal{V}$  be a set of variables; variables bind to RDF terms from  $\mathcal{I} \cup \mathcal{B} \cup \mathcal{L}$ . A triple  $p \in (\mathcal{I} \cup \mathcal{B} \cup \mathcal{V}) \times (\mathcal{I} \cup \mathcal{V}) \times (\mathcal{I} \cup \mathcal{B} \cup \mathcal{L} \cup \mathcal{V})$  is called triple pattern.*

**Definition (Basic Graph Pattern)** *A set of triple patterns is called a basic graph pattern  $P$ .*

# Queries

- We support BGP queries
- With SELECT and CONSTRUCT result clauses
- In SPARQL syntax:

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>  
PREFIX aifb: <http://www.aifb.kit.edu/id/>
```

```
SELECT ?x ?name  
WHERE {  
    aifb:Rudi_Studer foaf:knows ?x .  
    ?x foaf:name ?name .  
}
```



Section 3

# **HANDS-ON LOOKUPS & QUERIES**

# Exercise

- Find your URI (you own FOAF file – foaf-a-matic; [www.aifb.kit.edu](http://www.aifb.kit.edu); [semanticweb.org](http://semanticweb.org); [data.semanticweb.org](http://data.semanticweb.org); [dblp.lis3.de](http://dblp.lis3.de)...)
- FROM clause in SPARQL can do lookups
- Perform a lookup and return the URIs of
  - people you know
  - things you've created

# Solution in SPARQL

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

```
SELECT ?x ?name
```

```
FROM <http://harth.org/andreas/foaf>
```

```
WHERE {
```

```
    <http://harth.org/andreas/foaf#ah> foaf:knows ?x .
```

```
    ?x foaf:name ?name .
```

```
}
```

---

```
$ roqet foaf-ah.rq
```

# Solution in Linked Data-Fu

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .  
@prefix qrl: <http://www.aifb.kit.edu/project/ld-retriever/qrl#> .  
{ _:a http:mthd httpm:GET ;  
  http:requestURI <http://harth.org/andreas/foaf> . }
```

---

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

```
SELECT ?x ?name WHERE {  
  <http://harth.org/andreas/foaf#ah> foaf:knows ?x .  
  ?x foaf:name ?name .  
}
```

---

```
$ bin/ldfu -p program.n3 -q query.rq outfile.xml xml
```



Section 4

# **RULES & LINK FOLLOWING**

# Informally

- A rule has the following form (body  $\Rightarrow$  head):  
 $\{ \text{body} \} \Rightarrow \{ \text{head} \} .$
- A rule is safe if the variables in the head all occur in the body
- A rule is existential if there are blank nodes in the head; a rule is non-existential if there are no blank nodes in the head

# Deduction Rules Example

```
# prp-dom
{ ?p rdfs:domain ?c . ?x ?p ?y . } =>
    { ?x a ?c . } .
```

```
# prp-rng
{ ?p rdfs:range ?c . ?x ?p ?y . } =>
    { ?y a ?c . } .
```

„=>“ is a N3 shortcut for `log:implies`, with the prefix `log` expanding to <http://www.w3.org/2000/10/swap/log#>

# Example

Input:

```
{ ?p rdfs:domain ?c . ?x ?p ?y . } =>  
                                     { ?x a ?c . } .
```

```
foaf:knows rdfs:domain foaf:Person .  
:andreas foaf:knows :rudi .
```

Output:

```
:andreas rdf:type foaf:Person .
```



# Minimal Models

- Instead of a model-theoretic definition of the semantics (where we construct arbitrary models and check whether a graph satisfies them), we want a constructive method to compute the minimal models (i.e., the triples that follow from a given graph)
- We specify the „semantics“ using N3 rules

# Computing Minimal Models

- The naive rule evaluation computes the finite minimal model
- Finite: because we have a fixed vocabulary, and there are only finite ways to combine the vocabulary into triples
- Minimal: because we only compute the inferences that follow from the data and do not add additional triples

# Link Following

- We can use rules that include requests in the consequent
- E.g., perform a GET on URIs of people that Rudi knows

```
{ aifb:Rudi_Studer foaf:knows ?x . }
```

=>

```
{ _:bn http:mthd httpm:GET ;  
  http:requestURI ?x . } .
```

Section 5

# **HANDS-ON RULES & LINK FOLLOWING**

# Exercise I

- Find your URI (you own FOAF file – foaf-a-matic; [www.aifb.kit.edu](http://www.aifb.kit.edu); [semanticweb.org](http://semanticweb.org); [data.semanticweb.org](http://data.semanticweb.org); [dblp.l3.de](http://dblp.l3.de)...)
- Follow owl:sameAs links
- Take the semantics of owl:sameAs into account (you may use owl-ld.n3)
- Return all information about yourself

# Exercise II

- Find your URI (you own FOAF file – foaf-a-matic; [www.aifb.kit.edu](http://www.aifb.kit.edu); [semanticweb.org](http://semanticweb.org); [data.semanticweb.org](http://data.semanticweb.org); [dblp.ls3.de...](http://dblp.ls3.de))
- Follow owl:sameAs links
- Take the semantics of owl:sameAs into account (you may use owl:Id.n3)
- Return things created by people you know
- Alternatively, map predicates to common vocabularies (e.g., <http://www.aifb.kit.edu/id/Attribut-3AAutor5> to the equivalent FOAF term)

Section 6

# CONCLUSION

# Conclusion and Future Work

- Linked Data-Fu is a parallel rule engine for Linked Data
- Currently in version 0.9.0
- Optimisation of memory consumption
- Publication
- Packaging for version 1.0 (fixing small bugs, improving documentation...)