

Linked Data Fragments

by iMinds, presented by:

Richard Nagelmaeker - *FloatingData*

Contents

- Denklijn
- Voorbeelden
- Werking van Triple Pattern Fragments
- Aan de slag
- Evaluatie

Introduction

Linked Data Fragments, is bedacht door:
Ruben Verborgh en collega's van iMinds

Ik ben Richard Nagelmaeker, van *FloatingData*
en geef met deze presentatie tekst en uitleg
over Linked Data Fragments.

Denklijn

Dingen hebben eigenschappen



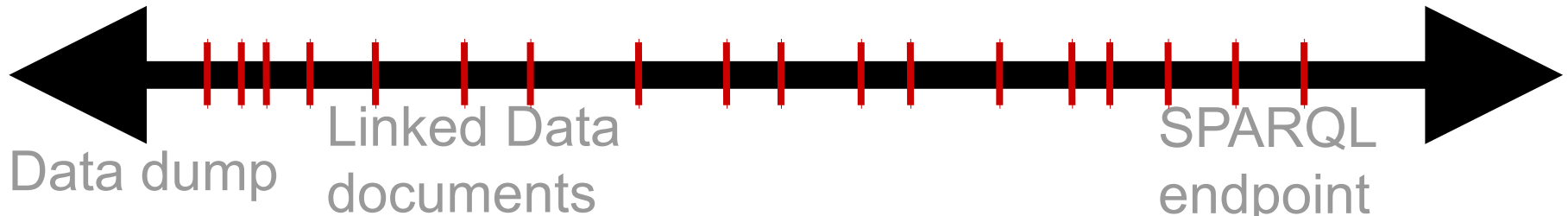
Alleen welke eigenschappen wil jij?



Linked Data Fragments

verouderde data
hoog bandbreedte
gebruik
lage beschikbaarheid
hoge kosten voor client
lage kosten voor server

live data
laag bandbreedte gebruik
lage beschikbaarheid
lage kosten voor client
hoge kosten voor server



En bij Linked Data Fragments?

Met welke eigenschappen bouw ik een Linked Data interface die werkt voor mijn uitdaging?
Dit is de vraag achter Linked Data Fragments.

Vandaag gaan we in op een specifieke samenstelling van een aantal eigenschappen:
Triple Pattern Fragments.

Voorbeelden van Linked Data Fragments

Linked Data Fragments

A **Linked Data Fragment** of a *Linked Data dataset* is a set of RDF triples that consists of three parts:

data

all triples of this dataset that match a specific *selector*;

metadata

triples that describe the dataset and/or the Linked Data Fragment;

controls

hypermedia links and/or forms that lead to other Linked Data Fragments.

[Specifications](#)

Data dump

Data

alle triples in de dataset

Metadata

het aantal triples, de omvang van de dump

Controls

(geen)

Linked Data Document

Data

triples over een onderwerp

Metadata

creator, maintainer, etc.

Controls

links naar andere Linked Data Documents

Sparql query result

Data

triples over een onderwerp

Metadata

(geen)

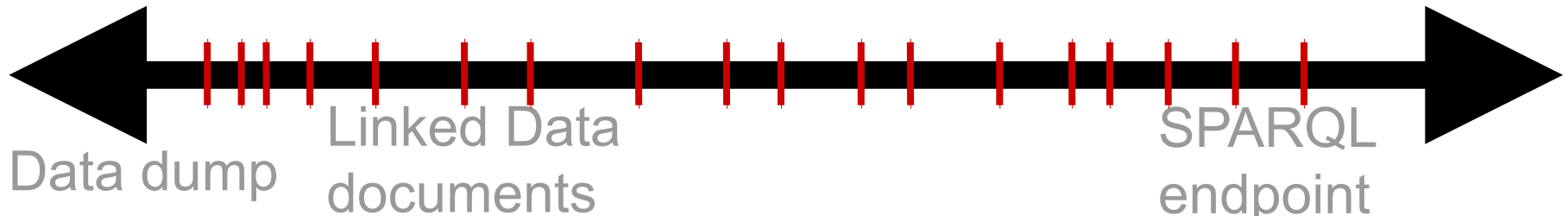
Controls

(geen)

Linked Data Fragments

verouderde data
hoog bandbreedte
gebruik
lage beschikbaarheid
hoge kosten voor client
lage kosten voor server

live data
laag bandbreedte gebruik
lage beschikbaarheid
lage kosten voor client
hoge kosten voor server



De Uitdaging

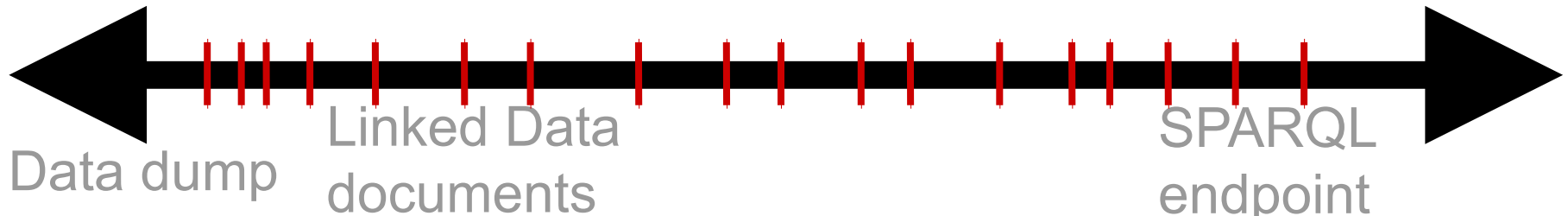
Hoe zorgen we er voor dat het live
bevragen van **Linked Data** op het **web**
betaalbaar wordt?

Door de servers eenvoudiger en de
clients slimmer te maken.

Met andere eigenschappen

verouderde data
hoog bandbreedte
gebruik
lage beschikbaarheid
hoge kosten voor client
lage kosten voor server

live data
laag bandbreedte gebruik
lage beschikbaarheid
lage kosten voor client
hoge kosten voor server



Triple Pattern Fragments



Triple pattern fragments

Data

matched aan een triple pattern

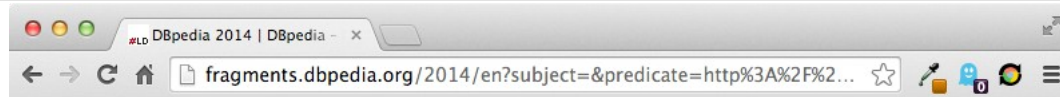
Metadata

het totaal aantal matches

Controls

toegang tot andere fragmenten

Hoe toont zich dat op de interface?



DBpedia - Linked Data Fragments



Query DBpedia 2014 by triple pattern

subject:

predicate:

object:

controls (*other fragments*)

Showing triples 1 to 101 of ±8141

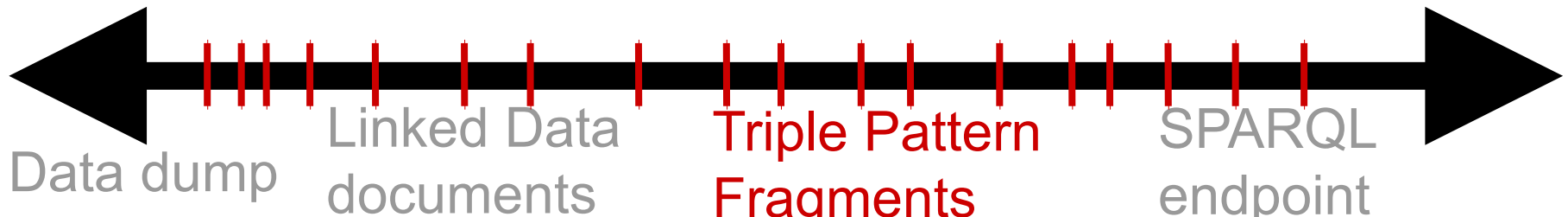
metadata (*total count*)

```
%C3%81lvaro_Crossi birthPlace Italy.  
%C3%89douard_Fachleitner birthPlace Italy.  
108_(artist) birthPlace Italy.  
A._F._K._Organski birthPlace Italy.  
Aaron_March birthPlace Italy.  
Abdon_Sgarbi birthPlace Italy.  
Abel_Gigli birthPlace Italy.  
Abelardo_Olivier birthPlace Italy.  
Abele_Blanc birthPlace Italy.  
Achille_Compagnoni birthPlace Italy.  
Achille_Coser birthPlace Italy.
```

data (*first 100*)

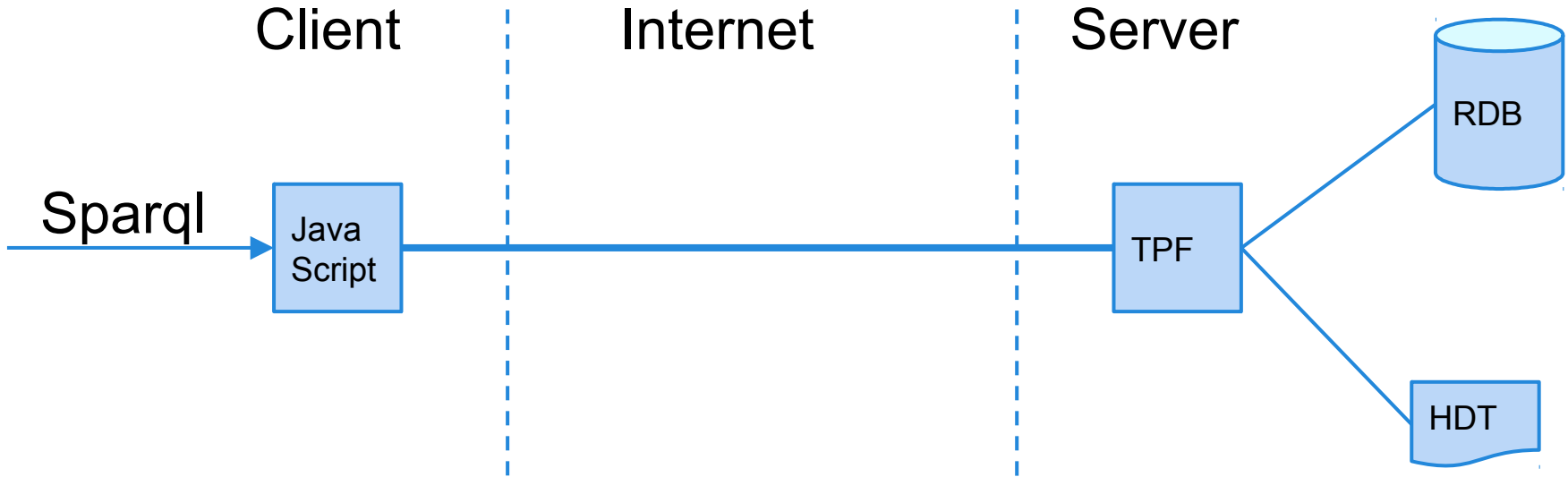
Triple pattern fragments zijn niet het antwoord op alles. **Geen enkele interface zal dat ooit zijn.**

Triple patterns laat zien hoe ver we kunnen komen met eenvoudige servers en slimme clients.

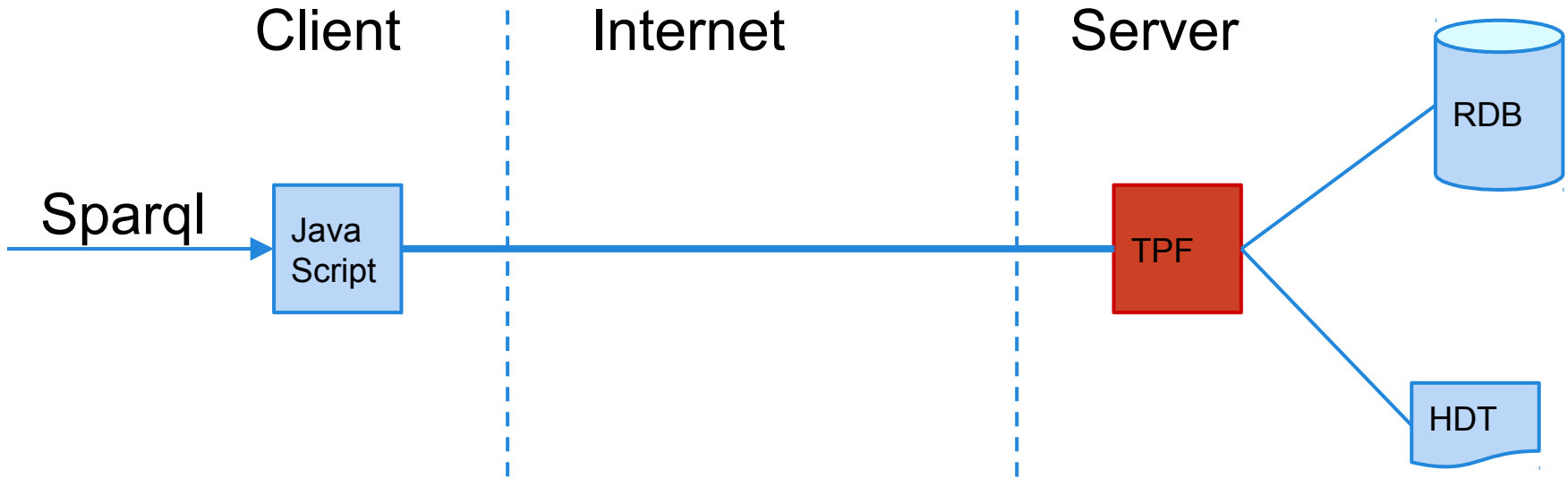


Hoe werken
Triple pattern
fragments

Overzicht



Overzicht



Het bevragen van de server (html)

De web interface van de server. Data opvragen middels een triple pattern.

Controls

De HTML representatie geeft aan:
“Je kunt vragen stellen middels een triple pattern”

Query DBpedia 2014 by triple pattern

subject:

predicate:

object:

`dbpedia-owl:birthPlace`

`dbpedia:Italy`

Find matching triples

Het bevragen van de server (rdf)

```
<http://fragments.dbpedia.org/2014/en#dataset> hydra:search [  
  hydra:template "http://fragments.dbpedia.org/2014/en  
    {subject, predicate, object}";  
  hydra:mapping  
    [hydra:variable "subject"; hydra:property rdf:subject ],  
    [hydra:variable "predicate"; hydra:property rdf:predicate ],  
    [hydra:variable "object"; hydra:property rdf:object ]  
].
```

Controls

De RDF representatie legt uit: “Je kunt vragen stellen middels een triple pattern”

Het bevragen van de server (html)

Showing triples 1 to 101 of ±8141

```
%C3%81lvaro_Crespi  birthPlace  Italy.  
%C3%89douard_Fachleitner  birthPlace  Italy.  
108 (artist)  birthPlace  Italy.
```

Metadata

De HTML representatie geeft aan: “dit is het aantal matches”

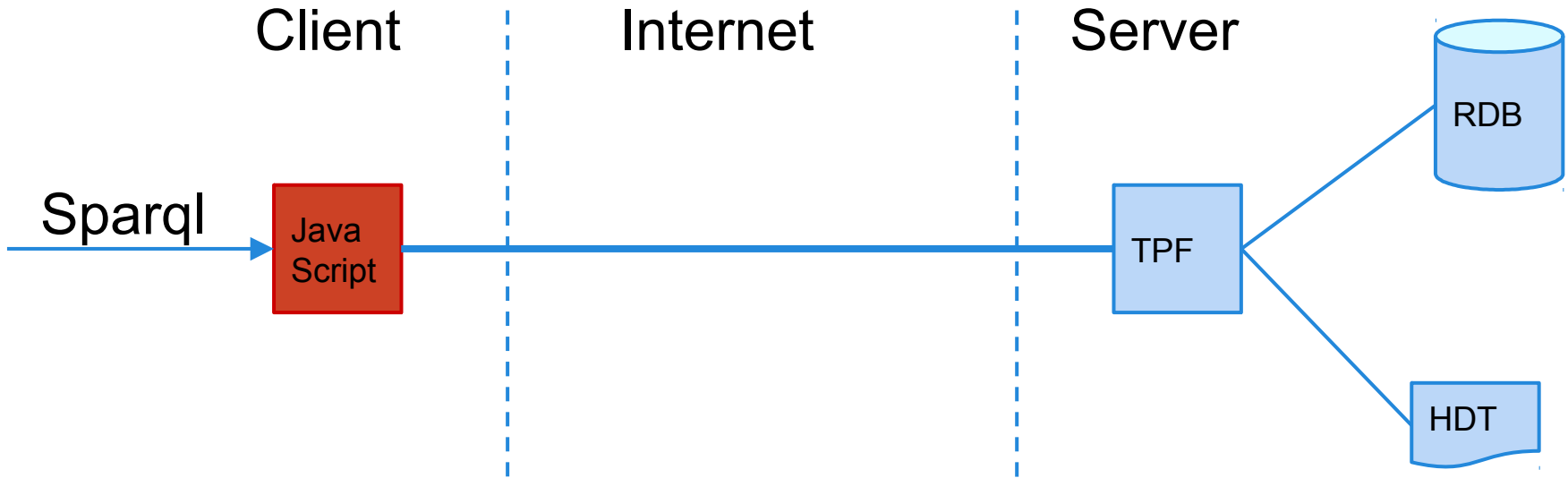
Het bevragen van de server (rdf)

`<#fragment> void:triples 8141.`

Metadata

De RDF representatie legt uit: “Dit is het aantal triples dat matched”

Overzicht



De client stelt Sparql queries

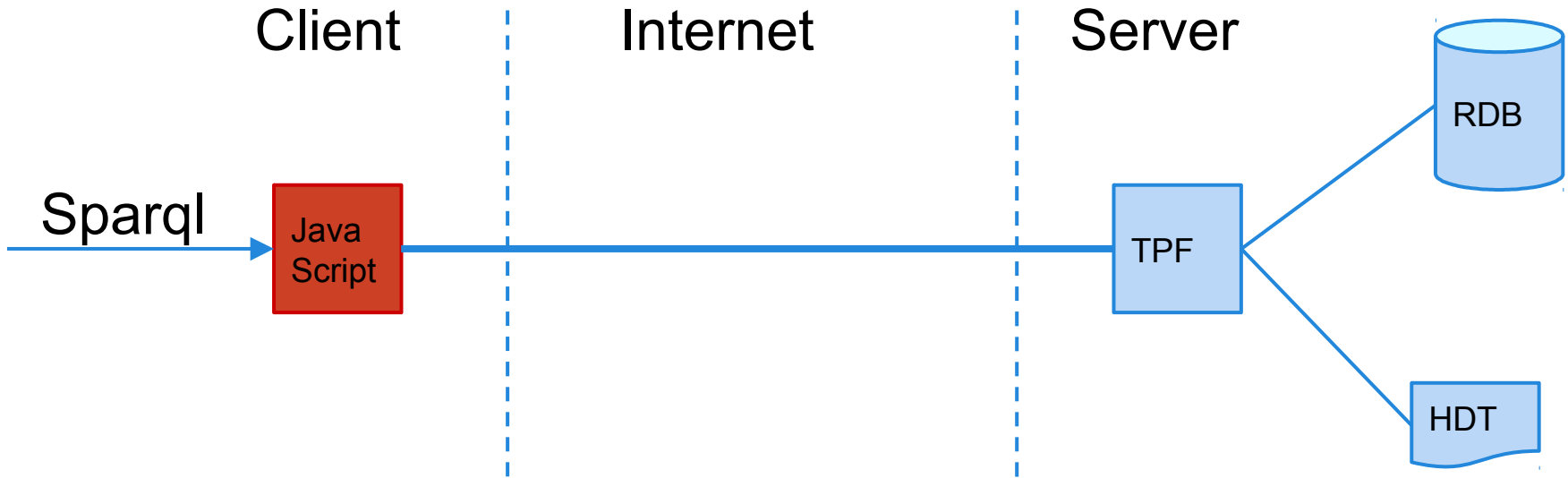
Geef aan de client een Sparql query op.
Geef ook de URL van een dataset fragment op.

De client kijkt in het fragment om te kijken hoe de dataset te benaderen.

Gebruikt de metadata om de query uit te voeren.

Aan de slag!

Overzicht



De Sparql Query

Vind de namen van de artiesten die in Siena, Italië geboren zijn. Sites: ([Client](#), [Server](#)(DBpedia 2014))

```
SELECT ?artist ?name
WHERE {
  ?artist a dbpedia-owl:Artist;
    rdfs:label ?name;
    dbpedia-owl:birthPlace dbpedia:Siena.
  FILTER LANGMATCHES(LANG(?name), 'EN')
}
```


Resultaten van de fragmenten

?artist rdf:type dbpedia-owl:Artist eerste 100 triples	96.000
?artist rdfs:label ?name eerste 100 triples	12.000.000
?artist dbpedia-owl:birthPlace dbpedia:Siena eerste 100 triples	78

Resultaten van de fragmenten

dbpedia:*Alessandro_Safina* rdf:type dbpedia-owl:Artist. 1

dbpedia:*Alessandro_Safina* rdfs:label ?name. 10

?artist dbpedia-owl:birthPlace dbpedia:Siena. 78

....

Alessandro_Safina birthPlace Siena.

Alfredo_Martinelli birthPlace Siena.

Ambrogio_Lorenzetti birthPlace Siena.

Probeer het zelf

Browser Client,
DBPedia 2014 - Server
List of TPF endpoints

Evaluatie

Berlin SPARQL benchmark

Amazon EC2 Machines

100 miljoen triples

hoge query diversiteit

BGP, UNION, FILTER,.....

Berlin SPARQL benchmark

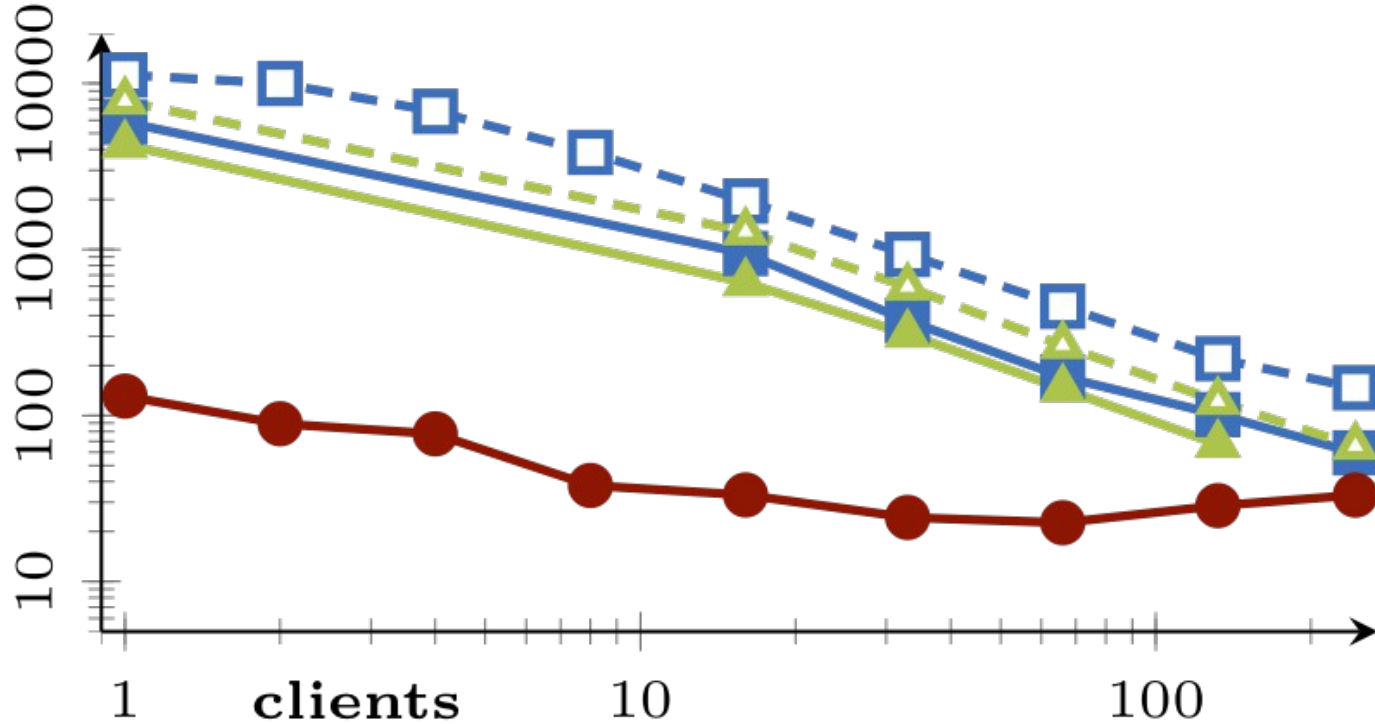
Platform

1 server

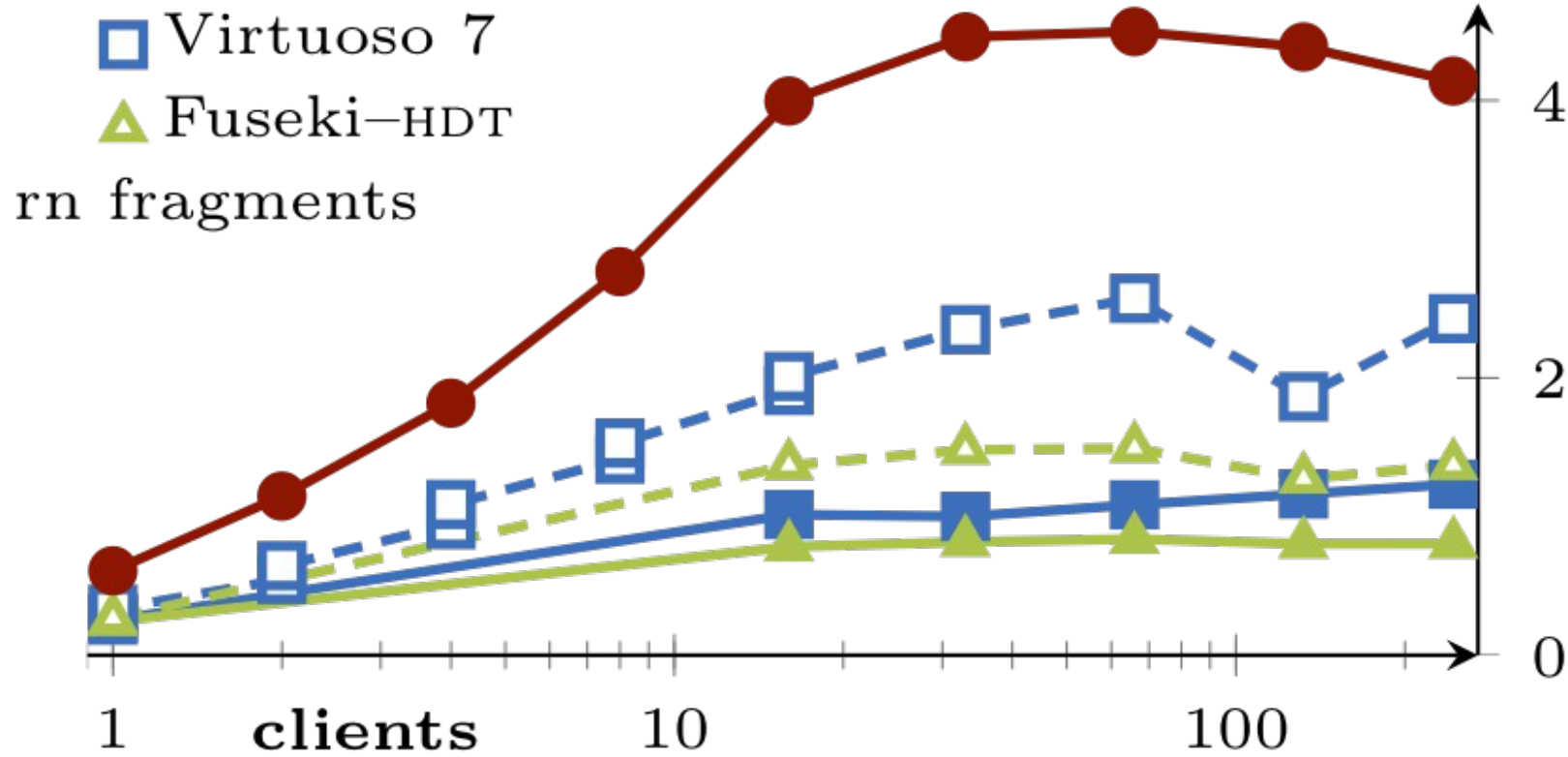
1 cache (proxy)

1-240 clients tegelijkertijd

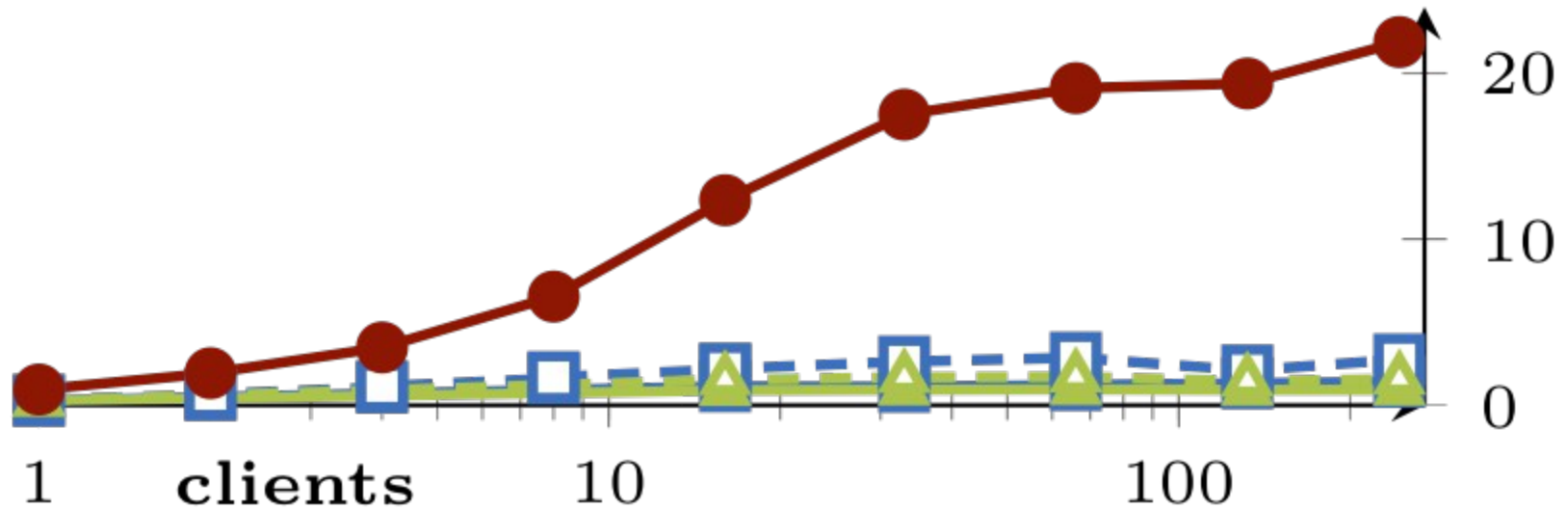
Executed Sparql queries per uur



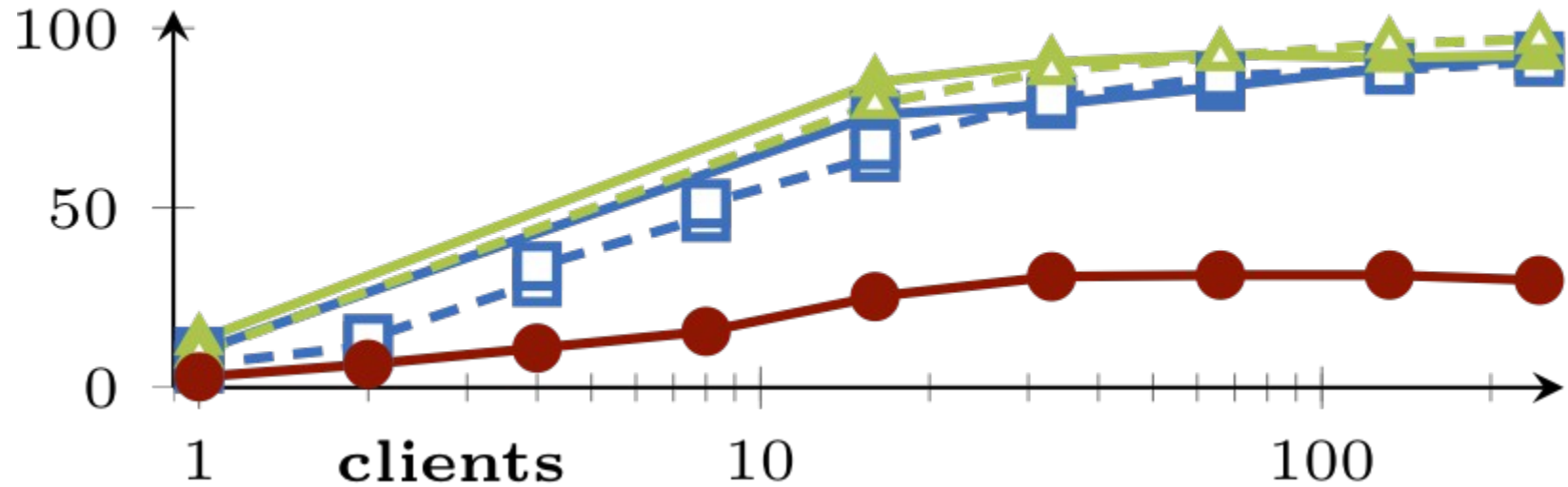
Data sent by server in MB



Data sent by cache (proxy) in MB



CPU gebruik op de server (per core)



Samenvatting

Eigenschappen

Positief:

Live data

Hoog beschikbare server tegen lage kosten

Negatief:

Meer bandbreedte gebruik

Het volledig beantwoorden van queries duurt langer
(het zijn wel streaming antwoorden)

Einde

Bedankt voor het volgen van de presentatie. Meer informatie kun je vinden op:

<http://linkeddatafragments.org/>

Download de software op:

<http://linkeddatafragments.org/software/>

Voor meer informatie, vraag het mij (info@floatingdata.nl) of mail naar ruben.verborgh@ugent.be (de maker en bedenker van LDF)

Links

Software (github)

<https://github.com/LinkedDataFragments>

Documentatie

<http://linkeddatafragments.org/>

Specificatie

<http://www.hydra-cg.com/spec/latest/linked-data-fragments/>

Triple Pattern Fragment interfaces:

<http://data.linkeddatafragments.org/>

DBPedia:

<http://fragments.dbpedia.org/>

Triple Pattern Fragment Client:

<http://client.linkeddatafragments.org/>

Software overzicht

<http://linkeddatafragments.org/software/>

Publicaties

<http://linkeddatafragments.org/publications/>

**Bedankt,
Tot ziens,
Have Fun!**

The world you see, is the world you describe - *FloatingData*, Richard Nagelmaeker