# Hosting Queryable and Highly Available Linked Data for Free

Luca Matteis @lmatteis
Ruben Verborgh @RubenVerborgh

# Motivation 1

SPARQL endpoints require the need to
buy and configure complex servers.

You need to worry about:
- having the funds to keep the server running
- making sure the system is up-to-date
- many other sys-admin tasks

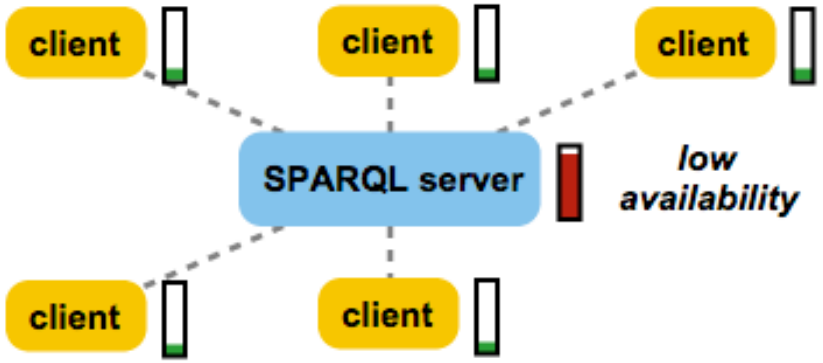
***TL;DR***

Hosting an RDF file is a lot easier than hosting a SPARQL endpoint.

# Motivation 2

SPARQL endpoints suffer from low-availability.

They offer consumers the ability to run any query they want. Consumers will run any query they want and will quickly overload the server with too many complex requests.



# Problem

Hosting SPARQL endpoints requires too much effort, both in terms of cost and server maintenance.

Fortunately SPARQL endpoints are not the only way of publishing queryable Linked Data.

*Triple Pattern Fragments* is a way of publishing queryable Linked Data on low cost servers.

# LDstatic & LDF-GAE

We have developed two tools, LDstatic & LDF-GAE, that implement the *TPF* protocol on low cost servers.

***TL;DR***

Users can run SPARQL queries against Linked Data published on online file repositories and cloud hosting services such as **GitHub**, **Google Code**, **Google App Engine** or **Dropbox**.

# LDstatic


**GitHub Pages**, and other online file repositories, can serve static HTML files.

It can also serve other type of content, such as N-Triples (`.nt`).

This means that GitHub Pages can serve triple pattern fragments.

# LDstatic

We want to match

```
<foo> <bar> "literal" .
```

using:

```
/?subject=<foo>
/?predicate=<bar>
/?object="literal"
/?subject=<foo>&predicate=<bar>
```

...

We need at least 8 static files for each triple if we want to match all combinations.

# LDstatic

Having all combinations is important for triple pattern fragments enabled clients so they can run complex queries, even SPARQL queries!

So we can host queryable linked data on GitHub Pages:

`http://lmatteis.github.io/ldstatic/`

# LDF-GAE

Now we talk about our other TPF implementation, which lets us run SPARQL queries on Google App Engine.

`http://ldf-gae.appspot.com/`

Instead of static files we use their native APIs to store and retrieve triples.

# LDF-GAE

Storing triples on App Engine's high-replication datastore is quite simple, and we can even match combinations using their native APIs.

## Stored as:

```
{
  "subject"  : triple.getSubject().toString(),
  "predicate": triple.getPredicate().toString(),
  "object"   : triple.getObject().toString()
}
```

# LDF-GAE

To retrieve each triple pattern combination we use their native API calls:

```
var filter = {};
if(subject)
    filter['subject'] = subject;
if(predicate)
    filter['predicate'] = predicate;
if(object)
    filter['object'] = object;

// query datastore
var query = new Query('triple');
for(var x in filter)
    query.addFilter(x, Query.FilterOperator.EQUAL, filter[x]);
```