

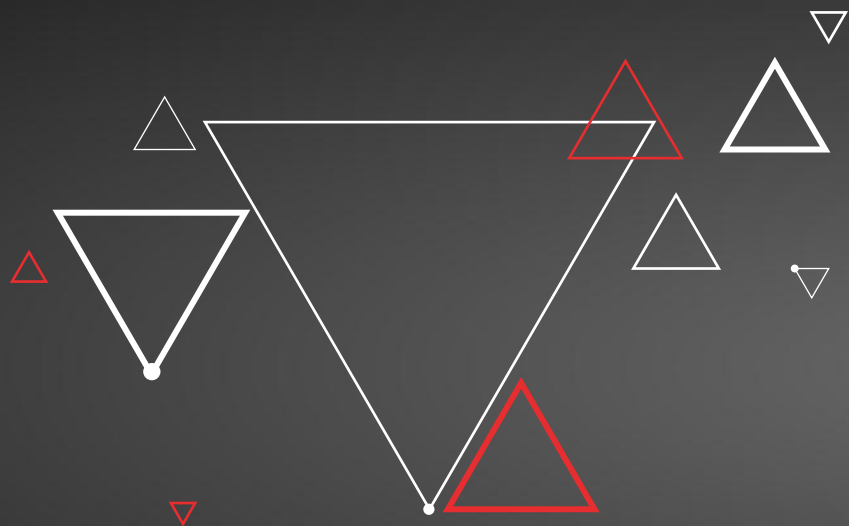
2019上图竞赛培训系列

细谈SPARQL之美

陈 涛

2019-05-23





01 什么是SPARQL?

02 为什么用SPARQL?

03 如何写SPARQL?

04 案例分享 (LOD)



本网站作为上海图书馆数字人文项目的开放数据平台，将陆续以关联数据（[Linked Data](#)）的方式向互联网公开发布上图数字人文项目所用的基础知识库（人、地、时、事、物），文献知识库（家谱、手稿档案、古籍等），本体词表，和数字人文项目建设过程中所用到的各种数据清洗和转换工具、以及项目组发表的相关论文、课件等研究资料。并以[REST API](#)，[Sparql Endpoint](#)，内容协商（[Content Negotiation](#)）等方式提供各种数据消费接口供开发人员调用，以促进数据的开发获取、共享和重用。调用接口前，需[注册API Key](#)，并阅读[开发者指南](#)。

人名规范库

More...

华人姓氏列表

More...

中国历史纪年表

More...

地理名词表

More...

机构名录

More...

上海年华

More...

印章知识库

More...

避讳字知识库

More...

1

Sparql endpoint <http://data.library.sh.cn:8890/sparql>

Virtuoso SPARQL Query Editor [About](#) | [Namespace Prefixes](#) | [Inference rules](#)

Default Data Set Name (Graph IRI)

Query Text

```
select distinct ?Concept where {[] a ?Concept} LIMIT 500
```

Sponging:

Results Format:

Execution timeout: milliseconds (values less than 1000 are ignored)

Options: Strict checking of void variables Log debug info at the end of output (has no effect on some queries and output formats)

(The result can only be sent back to browser, not saved on the server, see [details](#))

Copyright © 2019 [OpenLink Software](#)
Virtuoso version 07.20.3217 on Linux (x86_64-unknown-linux-gnu), Single Server Edition

什么是SPARQL ?

There have been many proposals for **RDF** and **RDFS** query languages:

- RDQL (<http://www.w3.org/Submission/2004/SUBM-RDQL-20040109/>)
- ICS-FORTH RQL (<http://139.91.183.30:9090/RDF/RQL/>)
- SeRQL(<http://www.openrdf.org/doc/sesame/users/ch06.html>)
- SPARQL (<http://www.w3.org/TR/rdf-sparql-query/>)

...

In this course we will only cover **SPARQL** which is the current W3C recommendation for querying RDF data.

2

为什么用SPARQL？

- 📶 在线
- ➡ 标准
- 🌐 互联
- 📅 实时

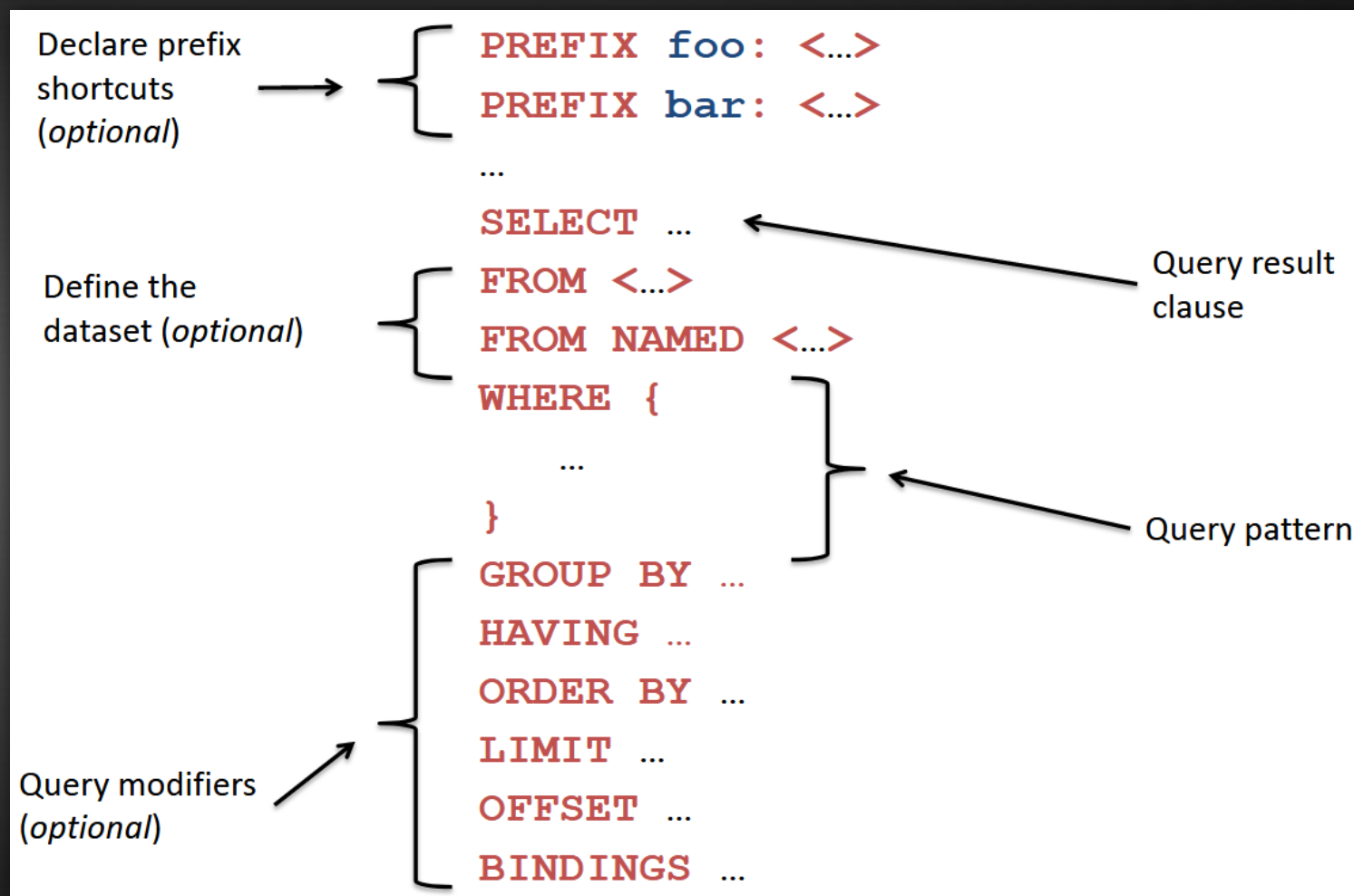


— 如何写SPARQL? —



3

SPARQL语法结构



3

SPARQL调用

<http://data.library.sh.cn:8890/sparql?<parameters>>

where *<parameters>* can include:

`query=<encoded query string>`

e.g. `SELECT+*%0DWHERE+{...`

`default-graph-uri=<encoded graph URI>`

e.g. `http%3A%2F%2Fexample.com%2Ffoo...`

n.b. zero or more occurrences of `default-graph-uri`

`named-graph-uri=<encoded graph URI>`

e.g. `http%3A%2F%2Fexample.com%2Fbar...`

n.b. zero or more occurrences of `named-graph-uri`

HTTP GET or POST. Graphs given in the protocol override graphs given in the query.

SPARQL查询种类

3

SELECT queries

Project out specific variables and expressions:

```
SELECT ?c ?cap (1000 * ?people AS ?pop)
```

Project out all variables:

```
SELECT *
```

Project out distinct combinations only:

```
SELECT DISTINCT ?country
```

Results in a table of values (in [XML](#) or [JSON](#)):

?c	?cap	?pop
ex:France	ex:Paris	63,500,000
ex:Canada	ex:Ottawa	32,900,000
ex:Italy	ex:Rome	58,900,000

CONSTRUCT queries

Construct RDF triples/graphs:

```
CONSTRUCT {  
  ?country a ex:HolidayDestination ;  
  ex:arrive_at ?capital ;  
  ex:population ?population .  
}
```

Results in RDF triples (in any RDF serialization):

```
ex:France a ex:HolidayDestination ;  
ex:arrive_at ex:Paris ;  
ex:population 635000000 .  
ex:Canada a ex:HolidayDestination ;  
ex:arrive_at ex:Ottawa ;  
ex:population 329000000 .
```

ASK queries

Ask whether or not there are any matches:

```
ASK
```

Result is either "true" or "false" (in [XML](#) or [JSON](#)):

```
true, false
```

DESCRIBE queries

Describe the resources matched by the given variables:

```
DESCRIBE ?country
```

Result is RDF triples (in any RDF serialization):

```
ex:France a geo:Country ;  
ex:continent geo:Europe ;  
ex:flag <http://.../flag-france.png> ;  
...
```

3

SPARQL查询Graph

```
select distinct ?graph
where {
  graph ?graph
  {?s ?p ?o}
}
```

Virtuoso SPARQL Query Editor

[About](#) | [Namespace Prefixes](#) | [Inference rules](#)

Default Data Set Name (Graph IRI)

Query Text

```
select distinct ?graph
where {
  graph ?graph
  {?s ?p ?o}
}
```

```
http://data.library.sh.cn:8890/sparql?default-graph-uri=&query=select+distinct+%3Fgraph+where+%7Bgraph+%3Fgraph+%7B%3Fs+%3Fp+%3Fo%7D%7D&should-sponge=&format=text%2Fhtml&timeout=0&debug=on
```

Result

<http://gj.library.sh.cn/graph/person>

<http://gen.library.sh.cn/graph/place>

<http://dy.library.sh.cn/graph/movie>

3

SPARQL查询古籍系统中朝代人物分布

```
select ?temporal (count(distinct ?s) as ?count)
where {
  ?s a shl:Person ;
  shl:temporalValue ?temporal .
} order by desc(?count)
```

```
http://data.library.sh.cn:
uri=http%3A%2F%2Fg
select+%3Ftemporal+%
unt%29++%0D%0Awh
+%3B+shl%3Atempora
r+by+desc%282%29&s
sponge=&format=text%
```

temporal	count
"清"	48694
"明"	11798
"清"	10550
"宋"	2862
"元"	1254
"唐"	1156
"晋"	217

```
aph-
ph%2Fperson&query=
%3Fs%29+as+%3Fco
3Fs+a+shl%3APerson
+.%0D%0A%7D+orde
debug=on
```

3

0起点学SPARQL系列（23讲）



SPARQL的联邦查询

0起点学 SPARQL

详情



SPARQL和Virtuoso数据库（1）

0起点学 SPARQL系列

详情



SPARQL的ASK、DESCRIBE和CONSTRUCT

0起点学 SPARQL系列

详情

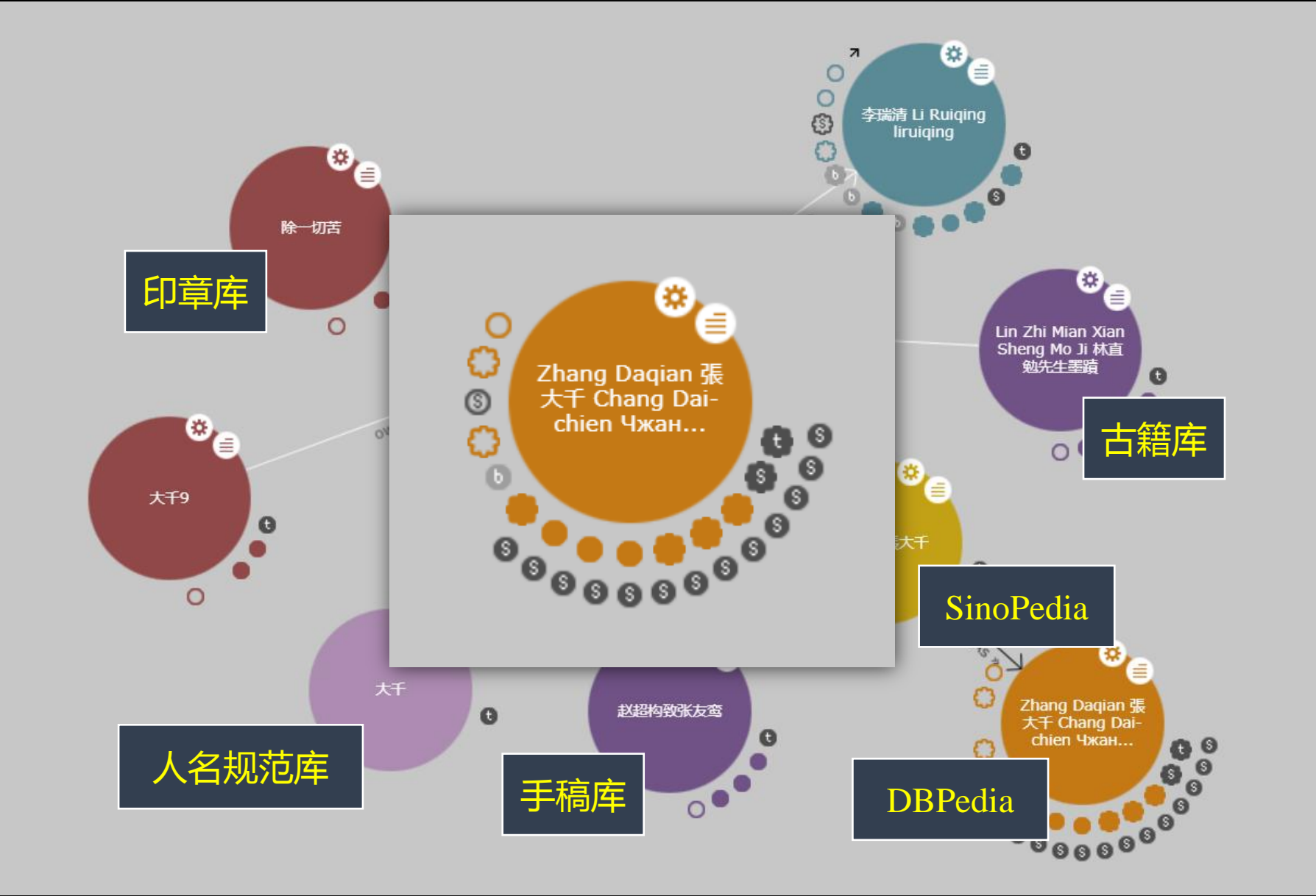


4

案例分享：LOD – Linked Open Data



张大千（Chang Dai-Chien，1899年5月10日—1983年4月2日），原名正权，后改名爰，字季爰，号大千，别号大千居士、下里港人，斋名大风堂。四川内江人，祖籍广东省番禺，1899年5月10日出生于四川省内江市中区城郊安良里的一个书香门第的家庭，中国泼墨画家，书法家。



张大千



打开一扇窗，去感受不一样的风景