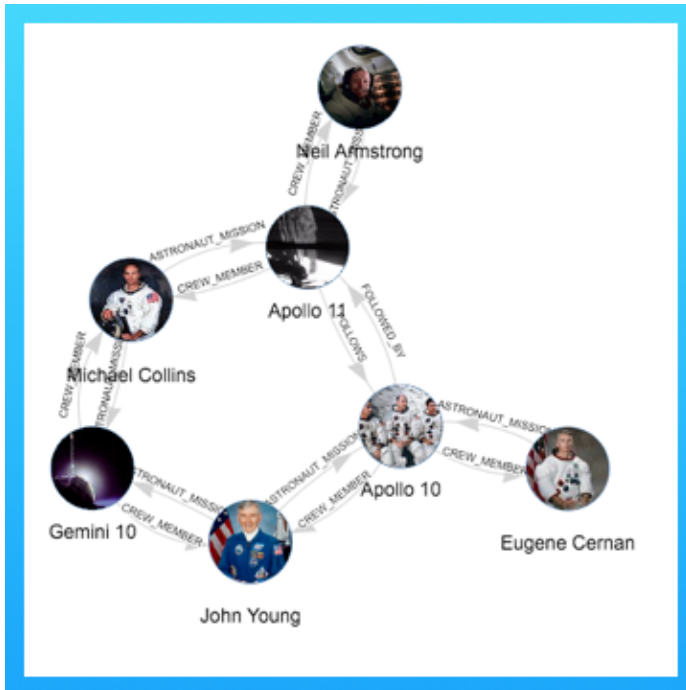


Navigate *All* the Knowledge

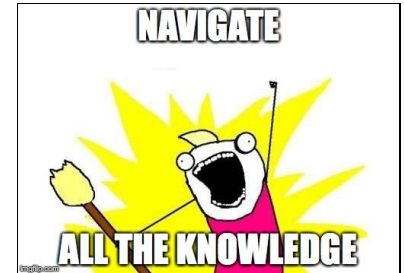
Getting started with ConceptMap.io

<http://ConceptMap.io>



James L. Weaver
Developer Advocate

Pivotal™



Twitter: @JavaFXpert

Email: jweaver@pivotal.io

<http://JavaFXpert.com>

<http://CulturedEar.com>

@JavaFXpert

About the Presenter

Author of several Java/JavaFX/RaspPi books



Java Champion, JavaOne Rockstar, plays well with others, etc :-)

@JavaFXpert **Pivotal**™

App is live at **ConceptMap.io**

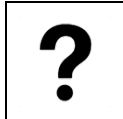
ConceptMap.io source code for services and UI:

<https://github.com/JavaFXpert/wikibrowser-service>

Open source, licensed under the Apache License, Version 2.0

Note: These **Technical Presentation** slides are available at

<http://slides.com/javafxpert/conceptmap-technical>

as well as by clicking the Help  button in **ConceptMap.io**

Wikimedia has many projects



Graphic from presentation by Lynda Pintscher

@JavaFXpert **Pivotal**[™]

... the most famous is Wikipedia



articles: 23 000 000+

edits: 1 400 000 000+

languages: 286

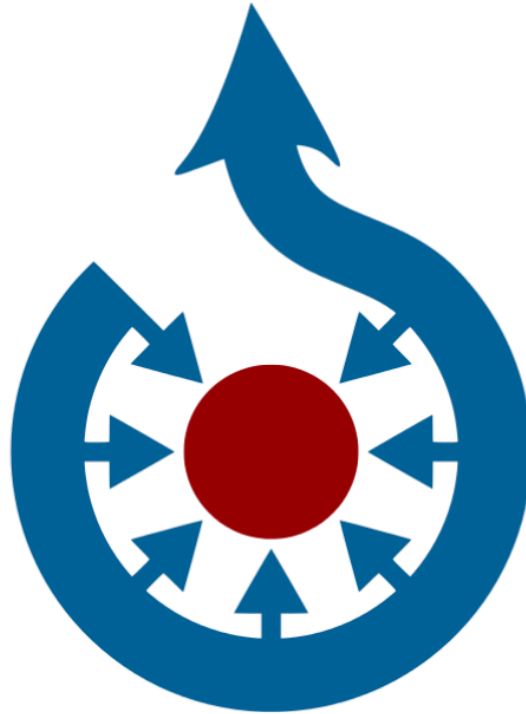
Graphic from 2013 presentation by Lynda Pintscher

@JavaFXpert

Pivotal[™]

Wikimedia Commons

Media files leveraged by Wikimedia projects



Graphic from 2013 presentation by Lynda Pintscher

@JavaFXpert

Pivotal™

Wikidata

Central storage for the structured data of Wikimedia projects



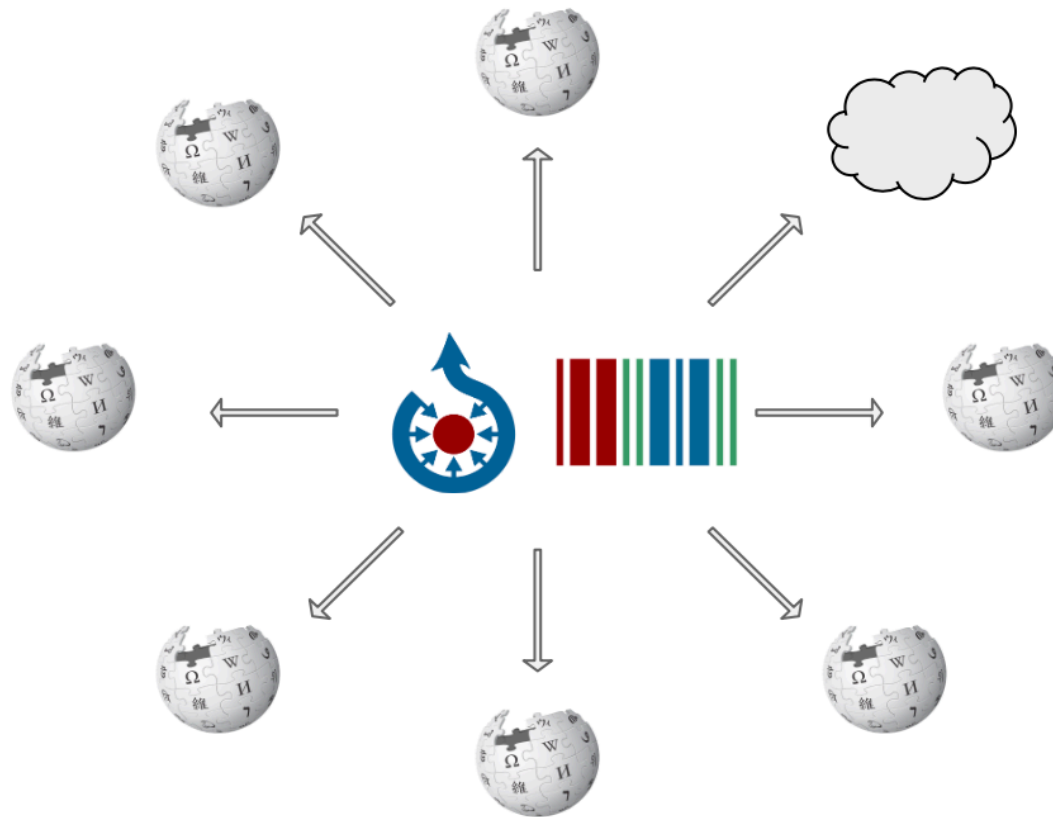
Graphic from 2013 presentation by Lynda Pintscher

@JavaFXpert

Pivotal™

Wikidata

Provides semantic structure for Wikipedia articles in any language



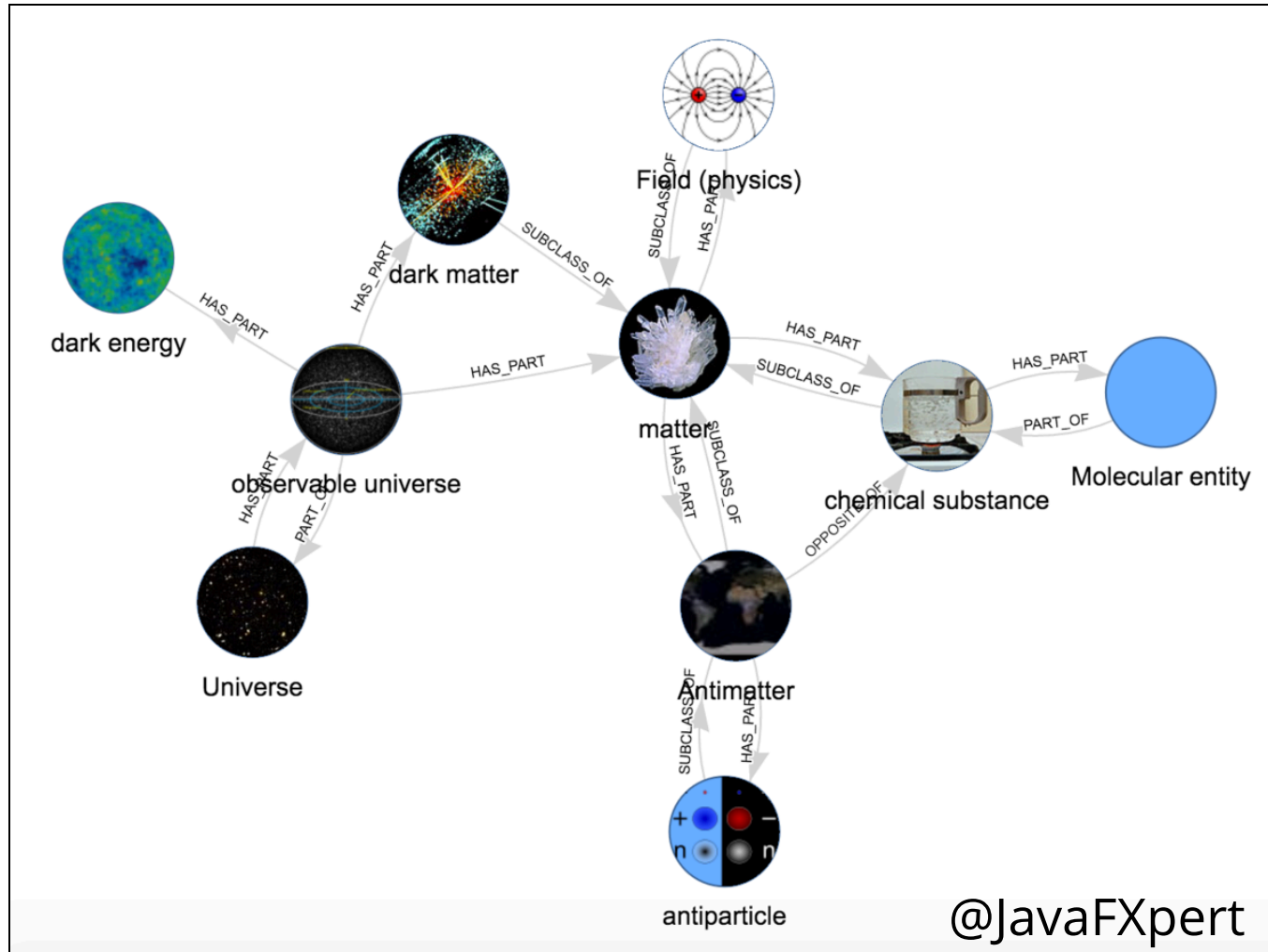
Graphic from 2013 presentation by Lynda Pintscher

@JavaFXpert

Pivotal™

Wikipedia + Wikidata

Wikipedia articles and their relationships to each other








The Big Idea: ConceptMap.io

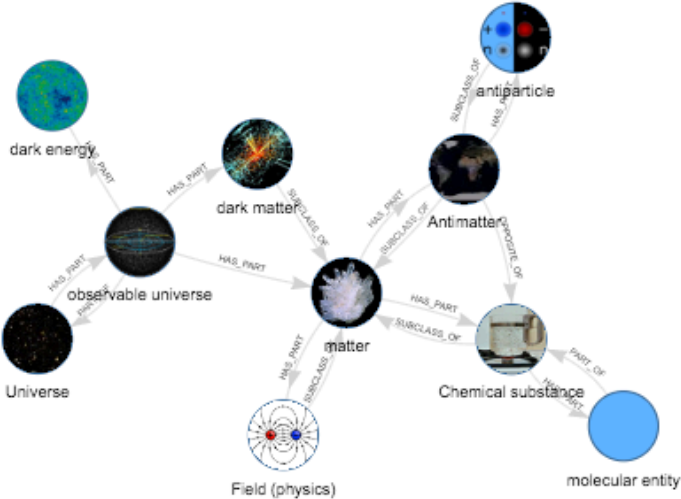
Navigate/pin Wikipedia articles via links and Wikidata relationships

ConceptMap
powered by Spring Cloud Foundry Neo4j GrapheneDB Wikidata

Universe :Q1 [en]

Search

☒ Pin item



ic rp Wikidata

Relationships:

x1 x* pu pf

+ has part :P527

x1 x* pu pf

+ has parts of the class :P2670


x1 x* pu pf

+ history of topic :P2184

Universe

For other uses, see [Universe \(disambiguation\)](#).

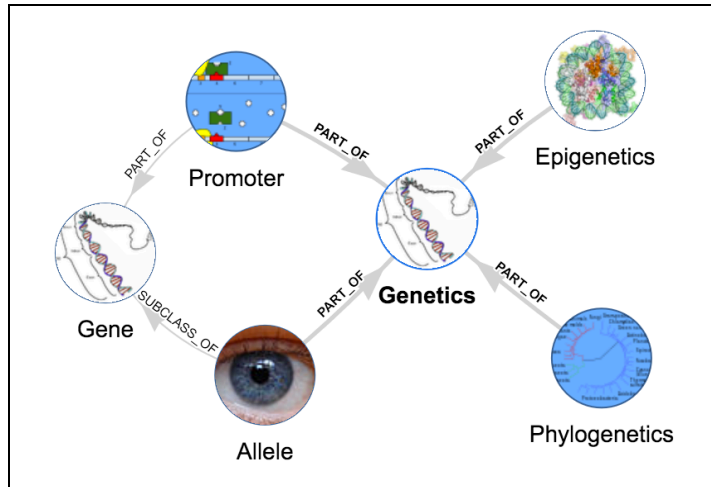
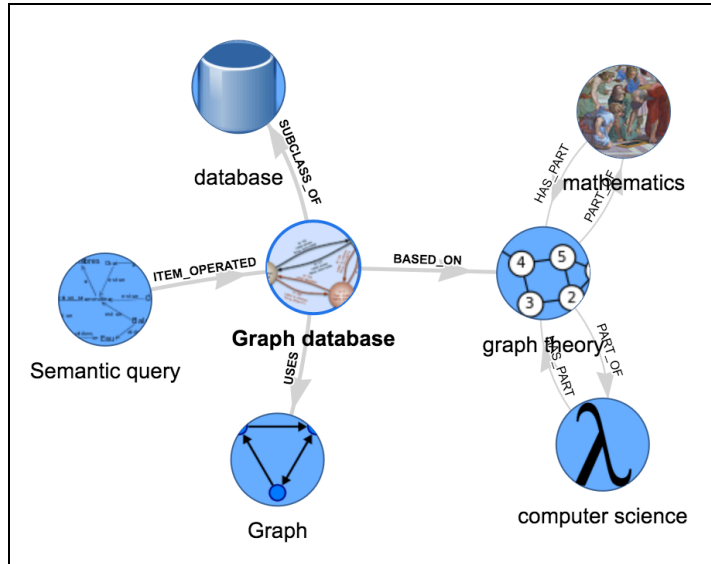
Universe



@JavaFXpert

Designed to facilitate:

- Learning
- Teaching
- Research



Learning/Teaching Scenario

Search for an item and pin it to the concept map

ConceptMap

powered by Spring

Cloud Foundry

Neo4j


GrapheneDB

Wikidata

Graph database :Q595971 [en]

Graph database

☒ Pin item



Graph database

ic rp Wikidata

Relationships:

x1 x* pu pf

+ based on :P144

x1 x* pu pf

+ subclass of :P279

x1 x* pu pf

+ topic's main category :P910

x1 x* pu pf

+ uses :P2283

From Related Items:

x1 x* pu pr

Graph database

In [computing](#), a **graph database** is a [database](#) that uses [graph structures](#) for [semantic queries](#) with nodes, edges and properties to represent and store data.

Most graph databases are [NoSQL](#) in nature and store their data in a [key-value store](#) or [document-oriented database](#). In general terms, they can be considered to be key-value databases with the additional *relationship* concept added. Relationships allow the values in the store to be related to each other in a free

Learning/Teaching Scenario

Select a link in the Wikipedia article

ConceptMap

powered by SpringCloud FoundryNeo4jGrapheneDBWikidata

Graph database :Q595971 [en]

Graph database

▼


🔍

⊖

☒ Pin item

🔗

?


Graph database

icrpWikidata

Relationships:

x1x*pupf

+ based on :P144

x1x*pupf

+ subclass of :P279

x1x*pupf

+ topic's main category :P910

x1x*pupf

+ uses :P2283

From Related Items:

x1x*pupr

Graph database

In [computing](#), a **graph database** is a [database](#) that uses [graph structures](#) for [semantic queries](#) with nodes, edges and properties to represent and store data.

Most graph databases are [NoSQL](#) in nature and store their data in a [key-value store](#) or [document-oriented database](#). In general terms, they can be considered to be key-value databases with the additional *relationship* concept added. Relationships allow the values in the store to be related to each other in a free

Learning/Teaching Scenario

Pin the item to the concept map

ConceptMap

powered by SpringCloud FoundryNeo4jGrapheneDBWikidata

Database :Q8513 [en]

Search

▼

🔍

⊖

☒ Pin item

🔗

?

Graph database

Database

SUBCLASS_OF

icrpWikidata

Relationships:

x1x*pupf

+ has part :P527

x1x*pupf

+ subclass of :P279

x1x*pupf

+ topic's main category :P910

From Related Items:

x1x*pupr

+ category's main topic :P301

Database

"Database Software" redirects here. For the computer program, see [Europress](#).

A **database** is an organized collection of [data](#).
[1] It is the collection of schemas, [tables](#), [queries](#), reports, [views](#) and other objects. The data are typically organized to model aspects of reality in a way that supports [processes](#) requiring information, such as modelling the availability of rooms in hotels in a way that supports finding a hotel with vacancies.

A **database management system (DBMS)** is a [computer software](#) application that interacts

Learning/Teaching Scenario

Select an item in the map, expand a relationship, and select an item

ConceptMap

powered by SpringCloud FoundryNeo4jGrapheneDBWikidata

Graph database :Q595971 [en]

Search

▼

🔍

⊖

☒ Pin item

🔗

?

Graph database

database

SUBCLASS_OF

icrpWikidata

Relationships:

x1x*pupf

- based on :P144

graph theory :Q131476

x1x*pupf

+ subclass of :P279

x1x*pupf

+ topic's main category :P910

x1x*pupf

+ uses :P2283

From Related Items:

Graph database

In **computing**, a **graph database** is a **database** that uses **graph structures** for **semantic queries** with nodes, edges and properties to represent and store data.

Most graph databases are **NoSQL** in nature and store their data in a **key-value store** or **document-oriented database**. In general terms, they can be considered to be key-value databases with the additional *relationship* concept added. Relationships allow the values in the store to be related to each other in a free form way, as opposed to traditional *relational*

Learning/Teaching Scenario

Pin the item to the concept map

ConceptMap

powered by SpringCloud FoundryNeo4jGrapheneDBWikidata

Graph theory :Q131476 [en]

Search

☒ Pin item

database

Graph database

Graph theory

SUBCLASS OF

BASED ON

icrpWikidata

Relationships:

x1x*pupf

+ part of :P361

x1x*pupf

+ subclass of :P279

x1x*pupf

+ topic's main category :P910

From Related Items:

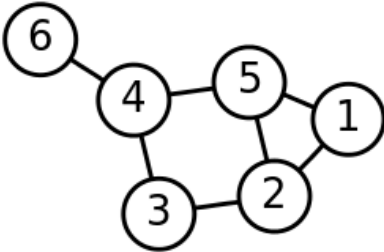
x1x*pupr

+ based on :P144

x1x*pupr

Graph theory

This article is about sets of vertices connected by edges. For graphs of mathematical functions, see [Graph of a function](#). For other uses, see [Graph \(disambiguation\)](#).



A [drawing](#) of a graph

In [mathematics](#) and [computer science](#), **graph**

Learning/Teaching Scenario

Select an item in the map, and click the [x1] button for a relationship

ConceptMap

powered by SpringCloud FoundryNeo4jGrapheneDBWikidata

Graph theory :Q131476 [en]

Search

▼

🔍

⊖

☒ Pin item

🔗

?

database

Graph database

Graph theory

SUBCLASS OF

BASED ON

icrpWikidata

Relationships:

x1x*pupf

+ part of :P361

x1x*pupf

+ subclass of :P279

x1x*pupf

+ topic's main category :P910

From Related Items:

x1x*pupr

+ based on :P144

x1x*pupr

Graph theory

This article is about sets of vertices connected by edges. For graphs of mathematical functions, see [Graph of a function](#). For other uses, see [Graph \(disambiguation\)](#).

A drawing of a graph

In mathematics and computer science, graph

Learning/Teaching Scenario

The [x1] button pinned the items found at the first level of the relationship

ConceptMap

powered by SpringCloud FoundryNeo4jGrapheneDBWikidata

Graph theory :Q131476 [en]

Search

▼


🔍

⊖


☒ Pin item

🔗


?




Database



Graph database



computer science



mathematics

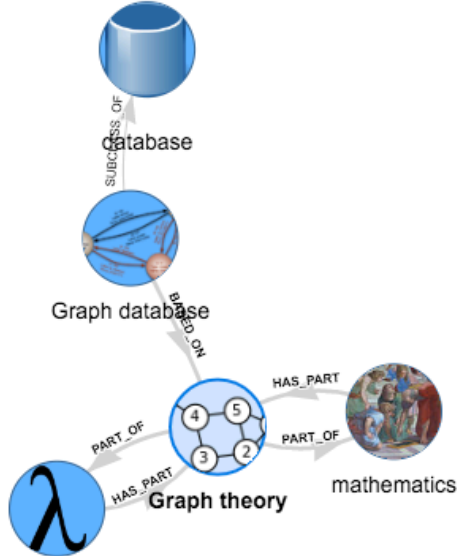
SUBCLASS OF

BUILD ON

PART OF

HAS PART

PART OF



icrpWikidata

Relationships:

x1x*pupf

+ part of :P361

x1x*pupf

+ subclass of :P279

x1x*pupf

+ topic's main category :P910

From Related Items:

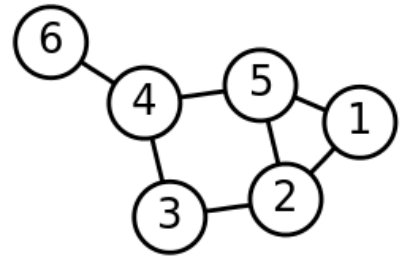
x1x*pupr

+ based on :P144

x1x*pupr

Graph theory

This article is about sets of vertices connected by edges. For graphs of mathematical functions, see [Graph of a function](#). For other uses, see [Graph \(disambiguation\)](#).

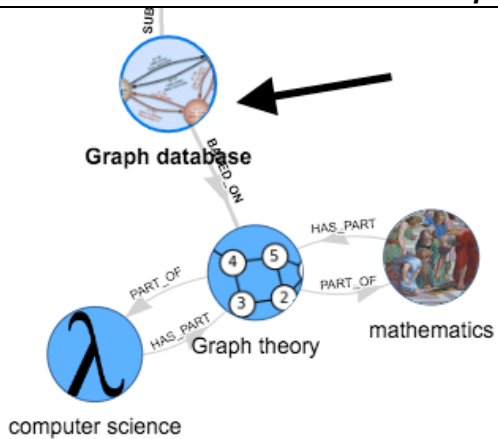


A [drawing](#) of a graph

In [mathematics](#) and [computer science](#), **graph**

Learning/Teaching Scenario

Select an item in the concept map, and select an item From Related Items



+ topic's main category :P910

+ uses :P2283

From Related Items:

x1 x* pu pr
+ category's main
topic :P301

x1 **x*** **pu** **pr**
 - instance of :P31

AllegroGraph
:Q4731408

ArangoDB
:Q20072115

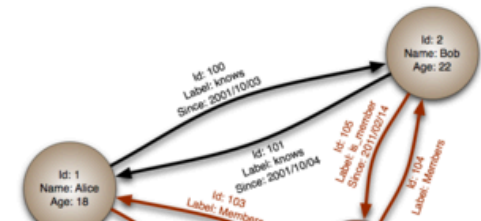
Blazegraph
:Q20127748

DEX
:Q3817722

Neo4j
:Q1628290

for **semantic queries** with nodes, edges and properties to represent and store data.

Most graph databases are **NoSQL** in nature and store their data in a **key-value store** or **document-oriented database**. In general terms, they can be considered to be key-value databases with the additional *relationship* concept added. Relationships allow the values in the store to be related to each other in a free form way, as opposed to traditional **relational databases** where the relationships are defined within the data itself. These relationships allow complex hierarchies to be quickly traversed, addressing one of the more common performance problems found in traditional key-value stores. Most graph databases also add the concept of *tags* or *properties*, which are essentially relationships lacking a pointer to another document.



Learning/Teaching Scenario

Pin the item to the concept map

ConceptMap

powered by SpringCloud FoundryNeo4jGrapheneDBWikidata

Neo4j :Q1628290 [en]

Search

▼

☒ Pin item

```
graph TD; Neo4j((Neo4j)) -- INSTANCE_OF --> GraphDatabase((Graph database)); GraphDatabase -- SUBCLASS_OF --> Database[(database)]; GraphDatabase -- BUILD-ON --> GraphTheory((graph theory)); GraphTheory -- PART_OF --> Mathematics((mathematics)); GraphTheory -- HAS_PART --> ComputerScience((computer science));
```

icrpWikidata

Relationships:

x1x*pu pf

+ instance of :P31

x1x*pu pf

+ license :P275

x1x*pu pf

+ platform :P400

x1x*pu pf

+ programming language :P277

From Related Items:

x1x*pu pr

Neo4j

Developer(s)	Neo Technology
Initial release	2007; 9 years ago ^[1]
Stable release	2.3.3 / March 24, 2016; 28 days ago ^[2]
Written in	Java
Operating system	Cross-platform
Type	Graph database

Learning/Teaching Scenario

Continue selecting items from Wikipedia articles and Wikidata Relationships

computer science

Items:

x1 x* pu pr

+ applies to part

:P518

Written in	Java
Operating system	Cross-platform
Type	Graph database
License	Dual-licensed: GPLv3 and AGPLv3 / commercial
Website	neo4j.com

Neo4j is a [graph database](#) management system developed by Neo Technology, Inc. Described by its developers as an [ACID](#)-compliant transactional database with native graph storage and processing,^[3] Neo4j is the most popular graph database according to db-engines.com.^[4]

Neo4j is available in a [GPL3](#)-licensed [open-source](#) "community edition", with [online backup](#) and [high availability](#) extensions licensed under the terms of the [Affero General Public License](#). Neo also licenses Neo4j with these extensions under closed-source commercial terms.^[5]

Neo4j is implemented in [Java](#) and accessible from software written in other languages using the [Cypher Query Language](#) through a



... and pin them to the concept map

ConceptMap powered by Spring Cloud Foundry Neo4j GrapheneDB Wikidata

Graph database :Q595971 [en]

▼
🔍
⊖
✓ Pin item
🔗
?

ic rp **Wikidata**

Relationships:

- x1 x* pu pf + based on :P144
- x1 x* pu pf + subclass of :P279
- x1 x* pu pf + topic's main category :P910
- x1 x* pu pf + uses :P2283

From Related Items:

- x1 x* pu pr

Graph database

In **computing**, a **graph database** is a **database** that uses **graph structures** for **semantic queries** with nodes, edges and properties to represent and store data.

Most graph databases are **NoSQL** in nature and store their data in a **key-value store** or **document-oriented database**. In general terms, they can be considered to be key-value databases with the additional *relationship* concept added. Relationships allow the values in the store to be related to each other in a free form.

Learning/Teaching Scenario

Edit relationships for an item by selecting Wikidata icon

ConceptMap

powered by Spring Cloud Foundry Neo4j GrapheneDB Wikidata

Graph database :Q595971 [en]

Search

▼

🔍

⊖

☒ Pin item

Key-value database

Cypher Query Language

Neo4j

Semantic query

Graph database

Graph

mathematics

graph theory

computer science

database

APPLIES TO PART

INSTANCE OF

ITEM OPERATED ON

USES

PART OF

HAS PART

SUBCLASS OF

BASED ON

SUPEROBJECT OF

ic rp Wikidata

Relationships:

x1 x* pu pf

+ based on :P144

x1 x* pu pf

+ subclass of :P279

x1 x* pu pf

+ topic's main category :P910

x1 x* pu pf

+ uses :P2283

From Related Items:

x1 x* pu pr


Graph database

In **computing**, a **graph database** is a **database** that uses **graph structures** for **semantic queries** with nodes, edges and properties to represent and store data.

Most graph databases are **NoSQL** in nature and store their data in a **key-value store** or **document-oriented database**. In general terms, they can be considered to be key-value databases with the additional *relationship* concept added. Relationships allow the values in the store to be related to each other in a free

Learning/Teaching Scenario

Wikidata page appears with which anyone may edit the item's info



Main page
Community portal
Project chat
Create a new item
Item by title
Recent changes
Random item
Query Service
Nearby
Help
Donate

Print/export
Create a book
Download as PDF
Printable version

Tools
What links here
Related changes
Special pages
Permanent link
Page information
Concept URI
Cite this page
Reasonator

English JavaFXpert 0 0 Talk Preferences Beta Watchlist Contributions Log out

Item Discussion Read View history ☆ Search

Graph database (Q595971)

No description defined [edit](#)

No aliases defined

▼ In more languages [Configure](#)

Language	Label	Description	Also known as
English	Graph database	No description defined	
French	base de données orientée graphe	Type de base de donnée utilisant les relations entre entitiées pour représenter et stocker les données.	
Italian	Base di dati a grafo	No description defined	
German	Graphdatenbank	No description defined	

[More languages](#)

Statements

subclass of

database

▼ 0 references

[edit](#)
[+ add reference](#)
[+ add](#)

Learning/Teaching Scenario

Select the Link icon to create a link to the concept map

ConceptMap

powered by SpringCloud FoundryNeo4jGrapheneDBWikidata

Graph database :Q595971 [en]


Search

▼

🔍

⊖

✓ Pin item



?

Key-value database

Cypher Query Language

Neo4j

Semantic query

Graph

Graph database

database

graph theory

computer science

mathematics

INSTANCE OF

ITEM OPERATED ON

USES

SUBCLASS OF

PART OF

HAS PART

BASED ON

icrpWikidata

Relationships:

x1x*pupf

+ based on :P144

x1x*pupf

+ subclass of :P279

x1x*pupf

+ topic's main category :P910

x1x*pupf

+ uses :P2283

From Related Items:

x1x*pupr

Graph database

In **computing**, a **graph database** is a **database** that uses **graph structures** for **semantic queries** with nodes, edges and properties to represent and store data.

Most graph databases are **NoSQL** in nature and store their data in a **key-value store** or **document-oriented database**. In general terms, they can be considered to be key-value databases with the additional *relationship* concept added. Relationships allow the values in the store to be related to each other in a free form way, as opposed to traditional **relational**

Learning/Teaching Scenario

Share the link to your concept map with a colleague or student

ConceptMap

powered by Spring Cloud Foundry Neo4j GrapheneDB Wikidata

Graph database :Q595971 [en]

Search

▼

Share ConceptMap Hyperlink

×

Select, copy and share this hyperlink to your concept map

<http://bit.ly/1MQtdcu>

Key-value database

Cypher Query Language

Neo4j

Semantic query

Graph

Graph database

database

computer science

graph theory

mathematics

INSTANCE OF

ITEM OPERATED ON

USES

SUBCLASS OF

PART OF

HAS PART

BASED ON

x1 x* pu pf

+ based on :P144

x1 x* pu pf

+ subclass of :P279

x1 x* pu pf

+ topic's main category :P910

x1 x* pu pf

+ uses :P2283

From Related Items:

x1 x* pu pr

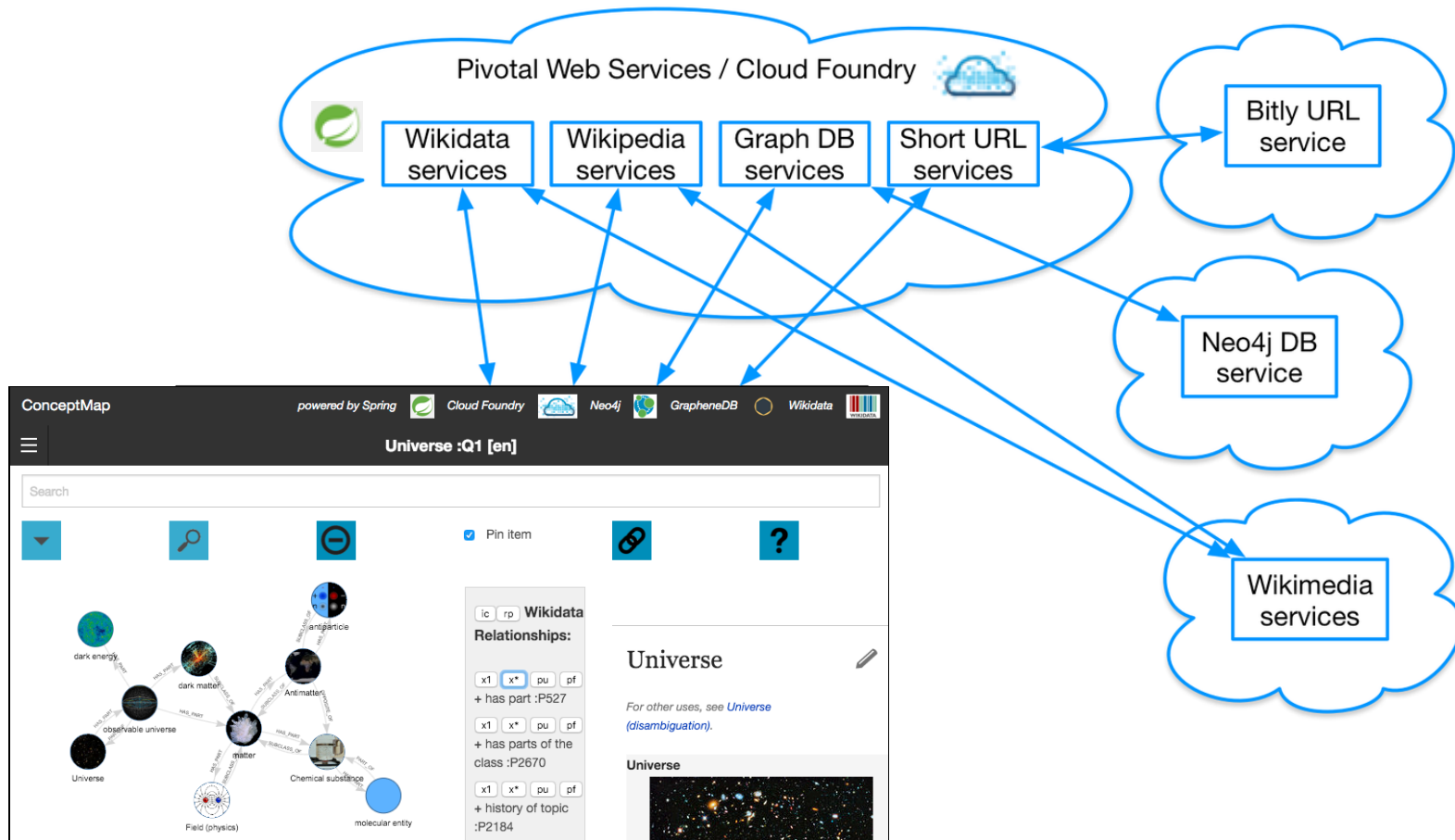
Graph database

In **computing**, a **graph database** is a **database** that uses **graph structures** for **semantic queries** with nodes, edges and properties to represent and store data.

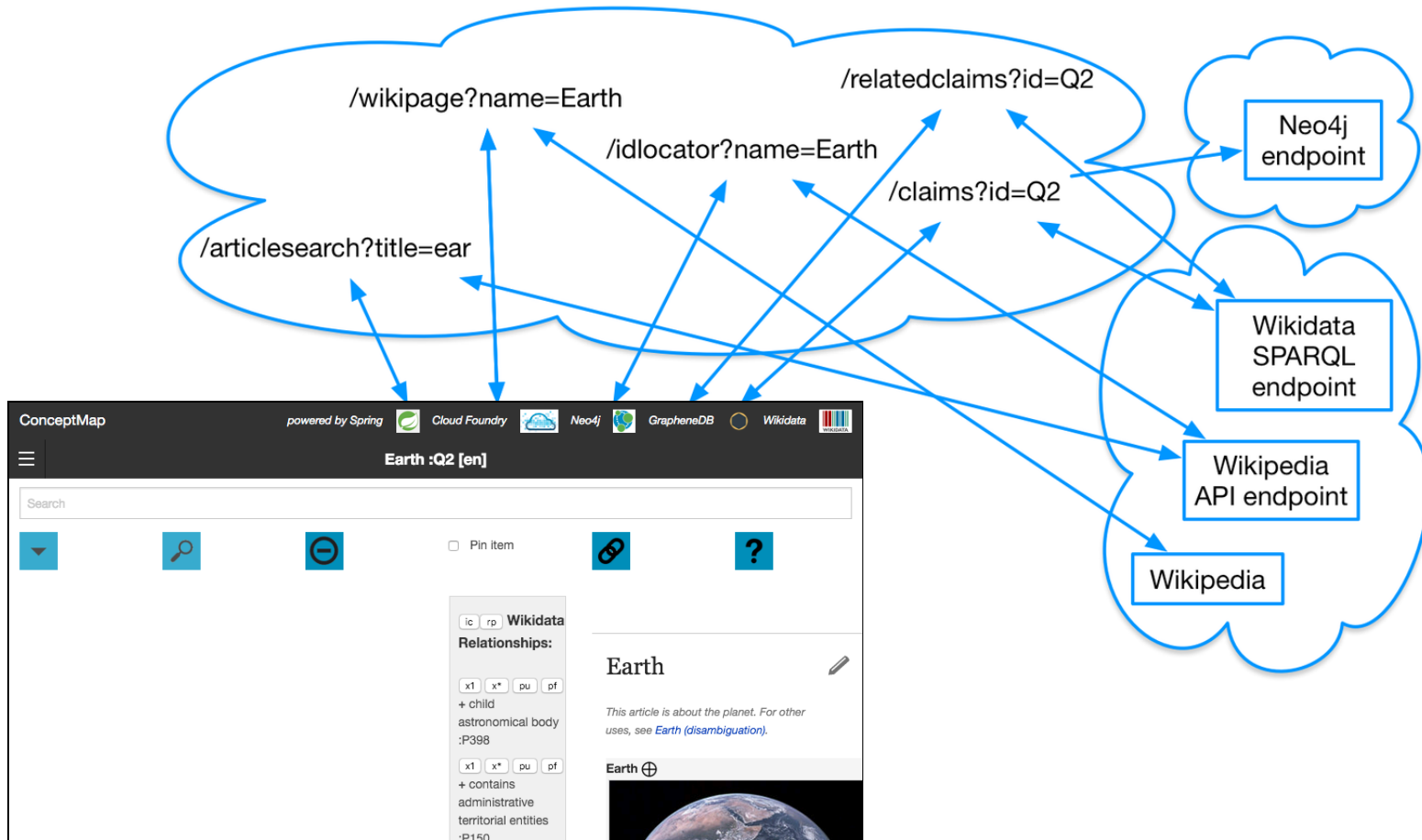
Most graph databases are **NoSQL** in nature and store their data in a **key-value store** or **document-oriented database**. In general terms, they can be considered to be key-value databases with the additional *relationship* concept added. Relationships allow the values in the store to be related to each other in a free

ConceptMap.io

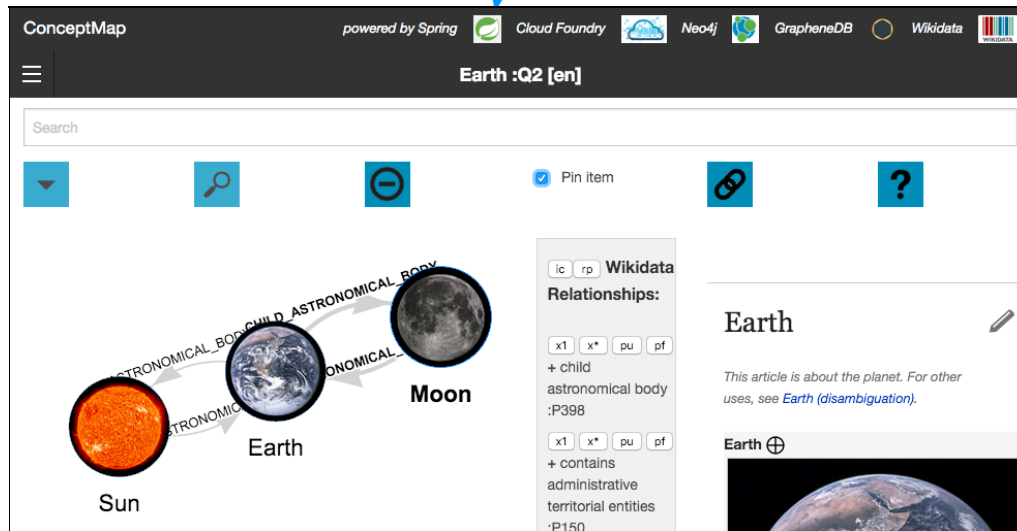
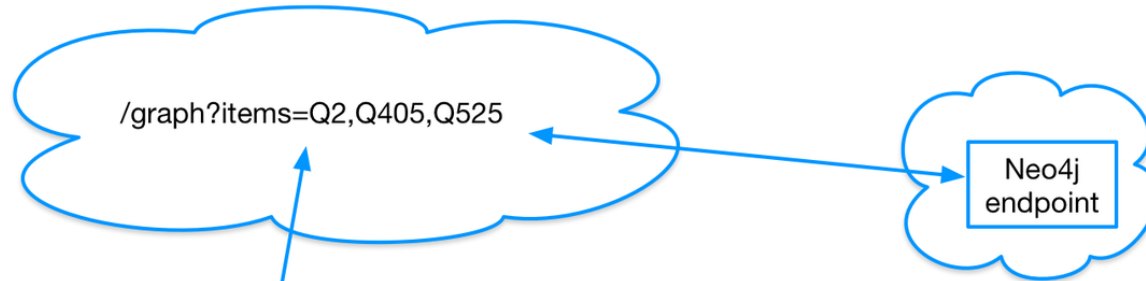
High-level view of distributed architecture



Use case: Search for Earth



Use case: Pin Earth to graph



Neo4j Cypher query

(find all relationships between pinned items)

<http://example/graph?items=Q2,Q405,Q525>



```
MATCH (a:Item), (b:Item)
WHERE a.itemId IN ['Q2', 'Q405', 'Q525']
      AND b.itemId IN ['Q2', 'Q405', 'Q525']
WITH a, b
OPTIONAL MATCH (a)-[rel]-(b)
RETURN a, b, collect(rel)
```

Resource controller

http://example/**visgraph**?**items**=Q2,Q405,Q525



```
@RestController
public class WikiGraphController {
    @RequestMapping(value = "/graph", method = RequestMethod.GET,
        produces = MediaType.APPLICATION_JSON_VALUE)
    public ResponseEntity<Object> search(@RequestParam(value = "items", defaultValue="")
        String items) {
        GraphResponseNear graphResponseNear = null;
        ...
        return Optional.ofNullable(graphResponseNear)
            .map(cr -> new ResponseEntity<>((Object)cr, HttpStatus.OK))
            .orElse(new ResponseEntity<>("Graph query unsuccessful",
                HttpStatus.INTERNAL_SERVER_ERROR));
    }
}
```

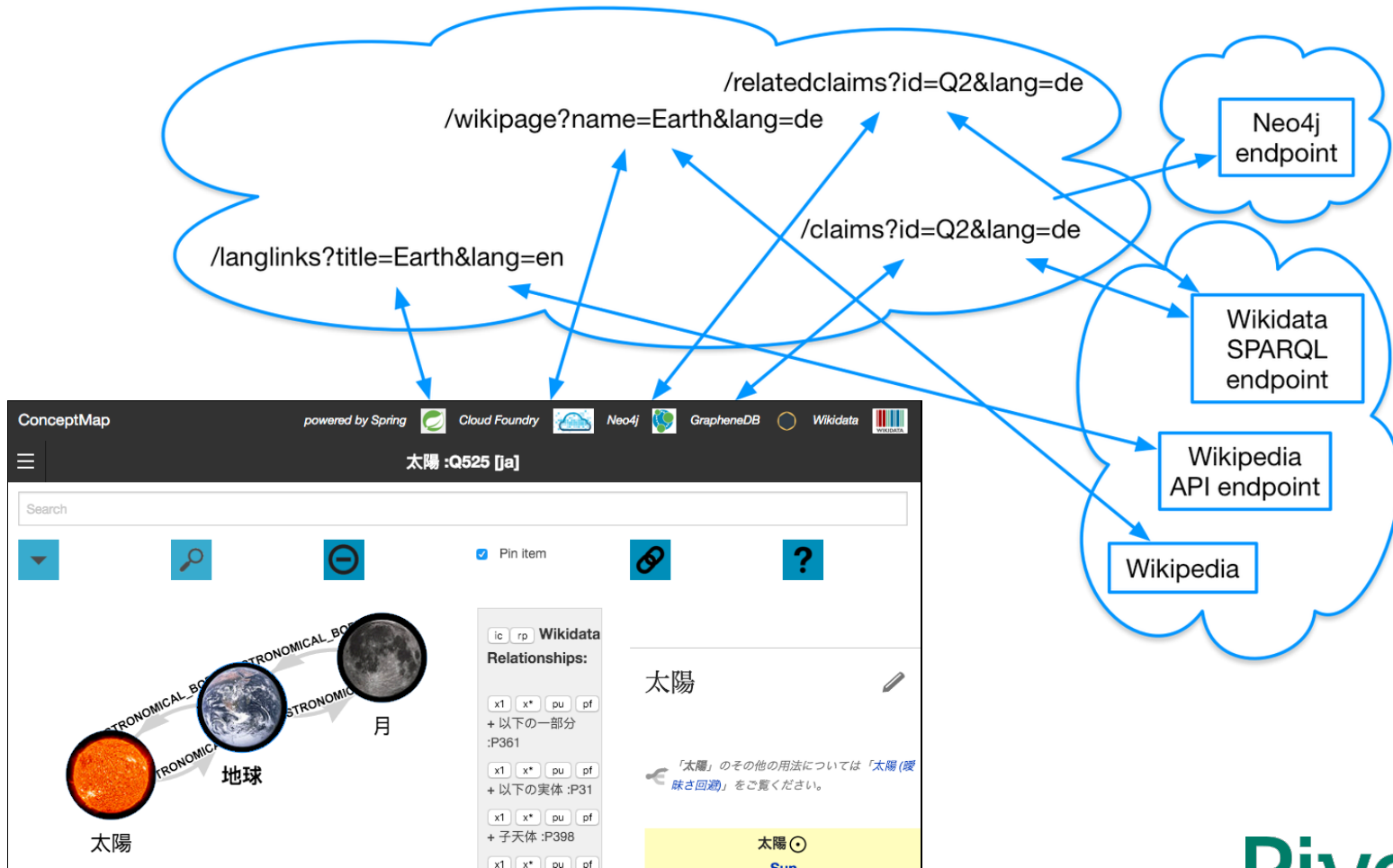
Note: Because [Jackson 2](#) is on the classpath, Spring's [MappingJackson2HttpMessageConverter](#) is automatically chosen to convert the GraphResponseNear instance to JSON

To learn more, see [Building a RESTful Web Service](#) Spring Guide

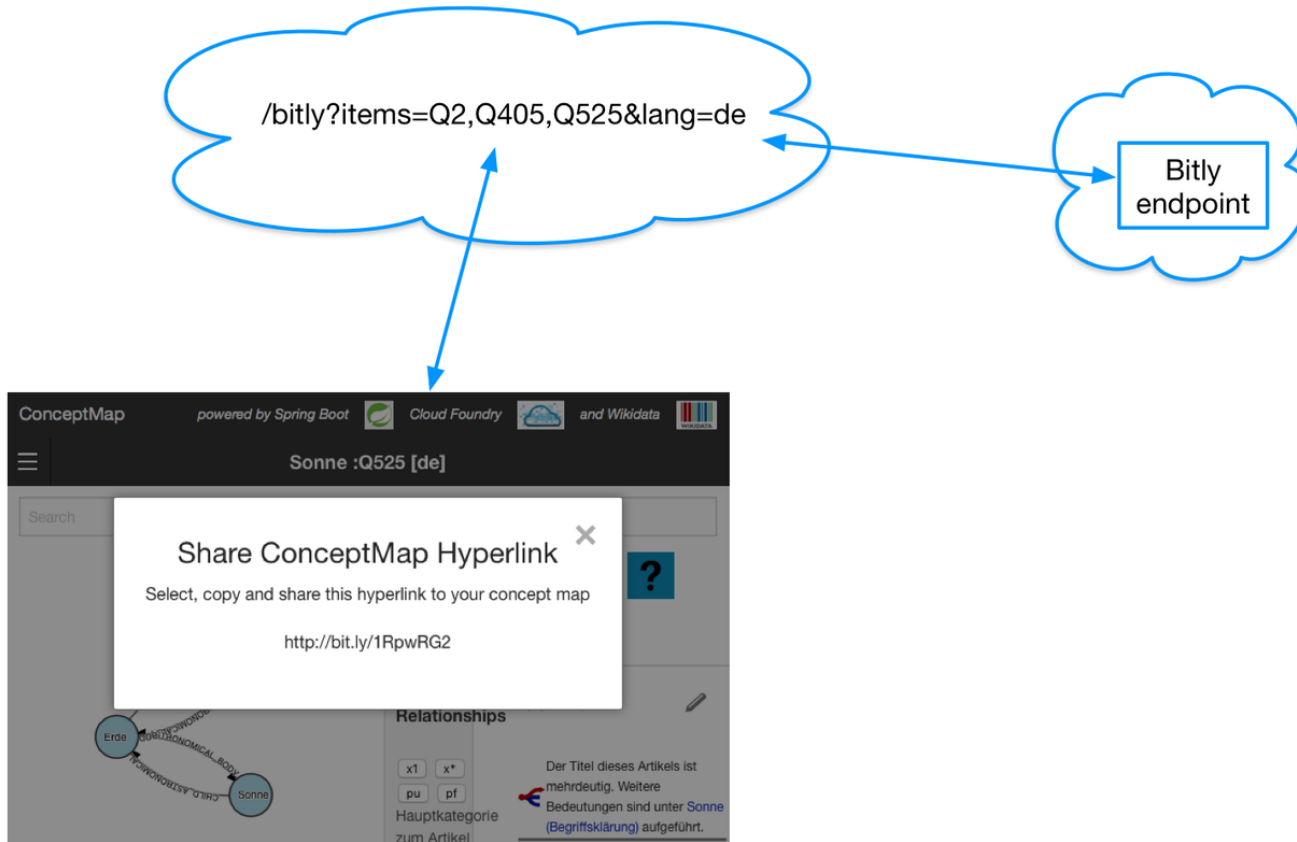
Resource representation

```
public class Item {  
    private String type;  
    private String id;  
  
    public Item() {  
    }  
  
    public String getType() {  
        return type;  
    }  
  
    public void setType(String type) {  
        this.type = type;  
    }  
  
    public String getId() {  
        return id;  
    }  
  
    public void setId(String id) {  
        this.id = id;  
    }  
}
```

Use case: Change language

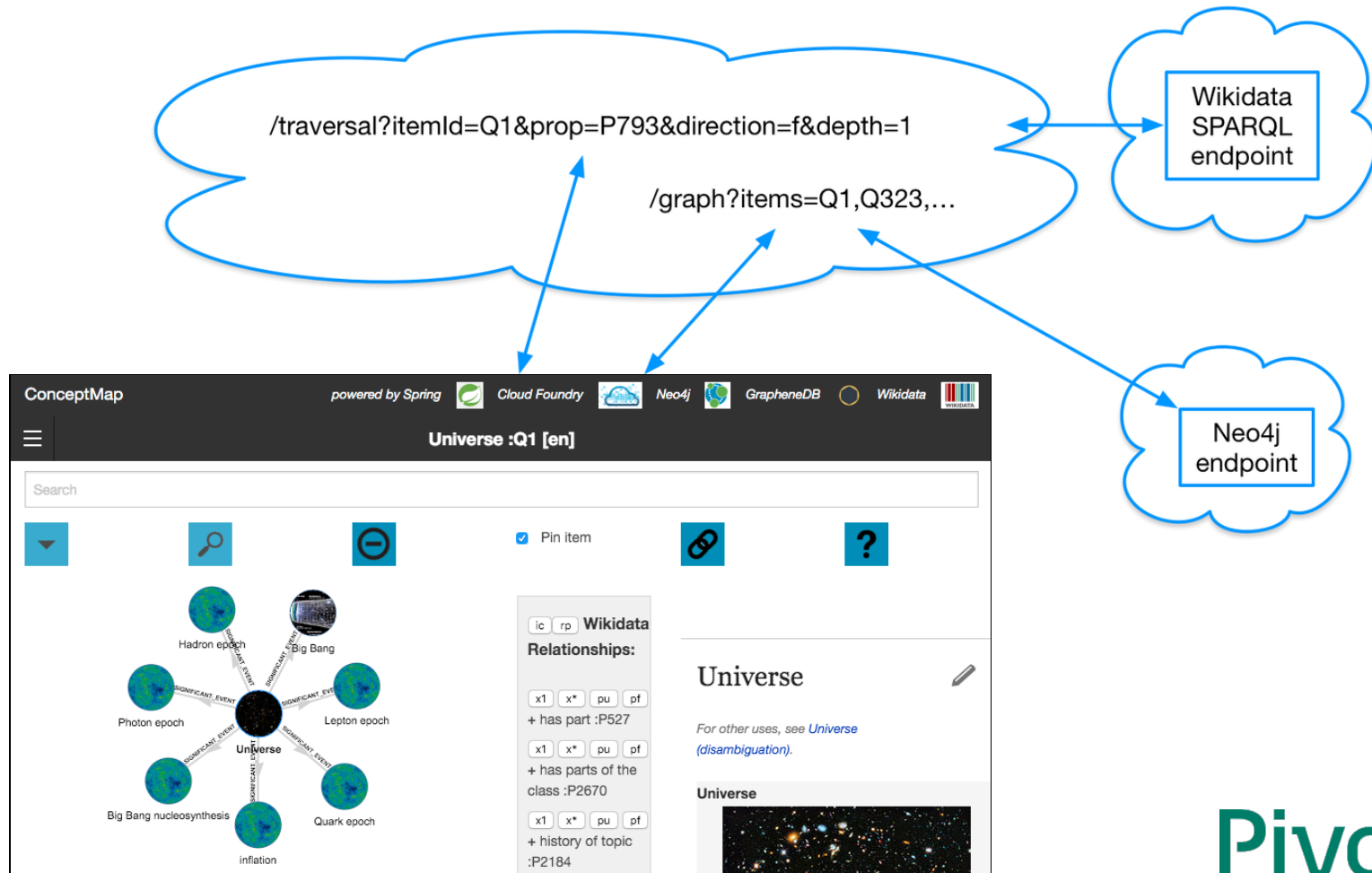


Use case: Share link



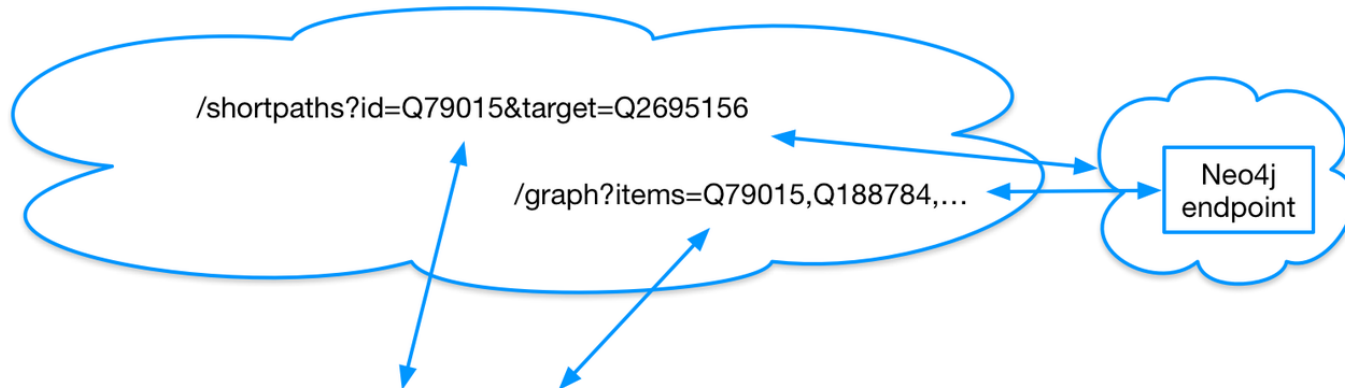
Use case: Breadth-first search

(expand items related by a given property to a given depth)



Use case: Items in common

(all shortest paths, two hops or less)



ConceptMap

powered by Spring

Cloud Foundry

Neo4j

GrapheneDB

Wikidata

Supergirl :Q8981293 [en]

Search

▼

🔍

⊖

📌 Pin item

🔗

?

```

graph TD
    Superman((Superman)) -- "INSTANCE OF" --> superhero((superhero))
    Superman -- "SUPERHERO MAIN FEATURE OR ABILITY" --> flight((flight))
    Supergirl((Supergirl)) -- "INSTANCE OF" --> superhero
    Supergirl -- "SUPERHERO MAIN FEATURE OR ABILITY" --> flight
    DCUniverse((DC Universe)) -- "FROM FICTIONAL UNIVERSE OR FROM REALITY" --> Superman
    DCUniverse -- "FROM FICTIONAL UNIVERSE OR FROM REALITY" --> Supergirl
    
```

ic rp Wikidata

Relationships:

x1 x* pu pf

+ director :P57

x1 x* pu pf

+ from fictional universe :P1080

x1 x* pu pf

+ instance of :P31

x1 x* pu pf

Supergirl

This article is about the superheroine. For other uses, see [Supergirl \(disambiguation\)](#).

Supergirl

Neo4j Cypher query

(all shortest paths, two hops or less)

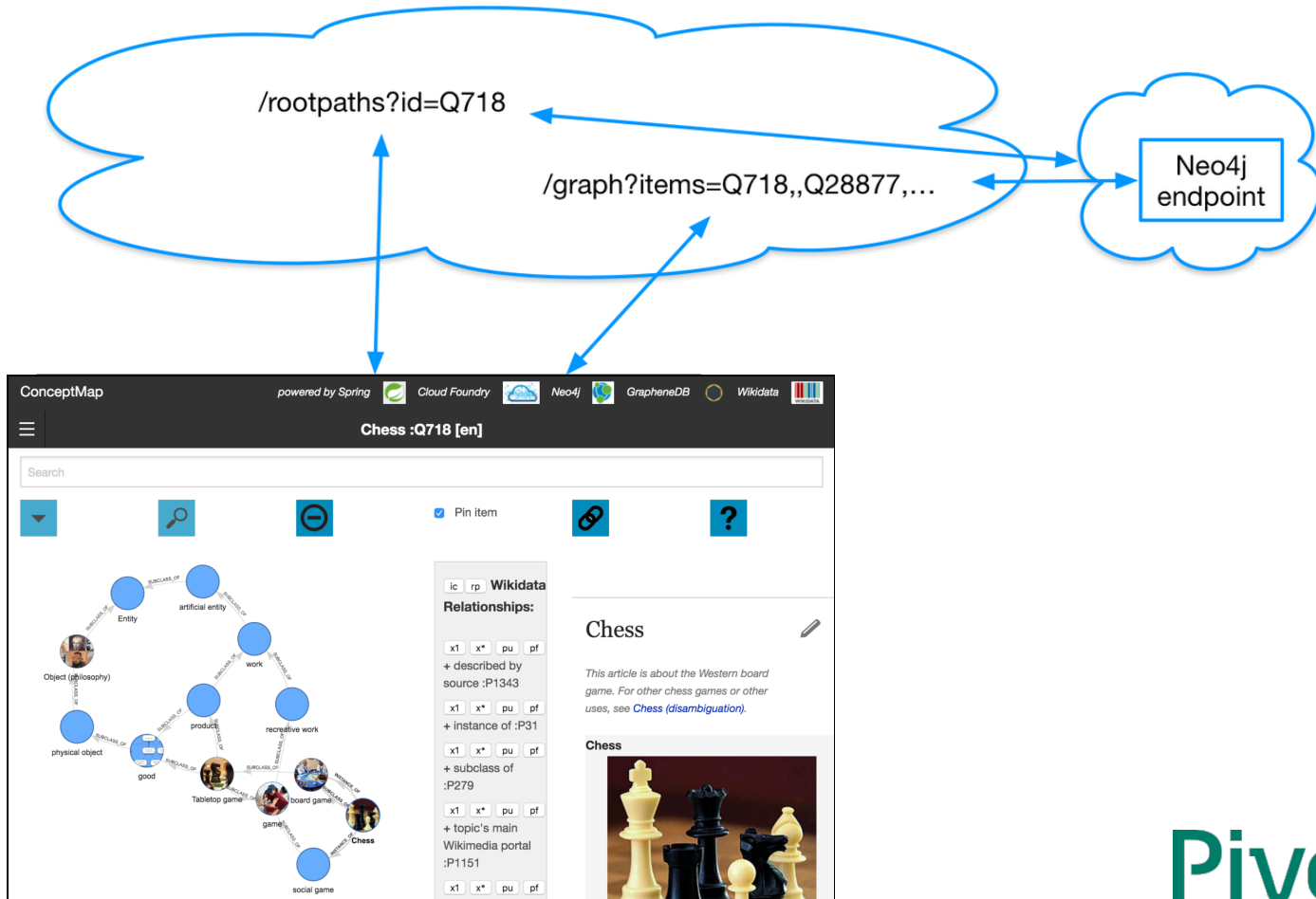
<http://example/visshortpaths?id=Q111&target=Q313>



```
MATCH p=allShortestPaths(  
  (a:Item {itemId:'Q111'})-[*..2]-(b:Item {itemId:'Q313'})  
)  
RETURN p
```

Use case: Navigate to root

(shortest path to Entity using subclass of, instance of, part of)



Neo4j Cypher query

(shortest path to Entity using subclass of, instance of, part of)

<http://example/visrootpaths?id=Q332>



```
MATCH p=allShortestPaths(  
  (a:Item {itemId: 'Q332'})-[*]->(b:Item {itemId: 'Q35120'})  
)  
WHERE NONE(x IN RELATIONSHIPS(p)  
  WHERE (x.propId <> 'P279') AND  
    (x.propId <> 'P31') AND  
    (x.propId <> 'P361'))  
)  
RETURN p
```

Make the app executable

```
package com.javafxpert.wikibrowser;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.boot.context.properties.EnableConfigurationProperties;

@EnableConfigurationProperties
@SpringBootApplication
public class WikiBrowserServiceApplication {

    public static void main(String[] args) {
        SpringApplication.run(WikiBrowserServiceApplication.class, args);
    }
}
```

To learn more, see [Building a RESTful Web Service](#) Spring Guide


App build/deploy cycle

1. `$ mvn clean install`
2. `$ cf push`



Note: One method of deployment for Spring Boot apps is a JAR file, which contains an embedded Tomcat servlet container.

Pivotal Web Services Console

**Pivotal Web Services**

culturedear > development

ORG

culturedear

SPACES

development

Marketplace

Docs

Support

Tools

Blog

Status

Welcome to Pivotal Web Services

SPACE

development

2 Running



0 Stopped

0 Down

Overview

APPLICATIONS

Learn More

STATUS	APP	INSTANCES
	ChordAnalyzerService http://chordanalyzese...	1
	CounterpointService http://counterpointser...	1

SPACE

development

2 Running

0 Stopped

0 Down

start.spring.io

SPRING INITIALIZR bootstrap your application now

Generate a Maven Project with Spring Boot 1.2.7

Project Metadata

Artifact coordinates

Group

com.example

Artifact

demo

Dependencies

Add Spring Boot Starters and dependencies to your application

Search for dependencies

Web, Security, JPA, Actuator, Devtools...

Selected Starters

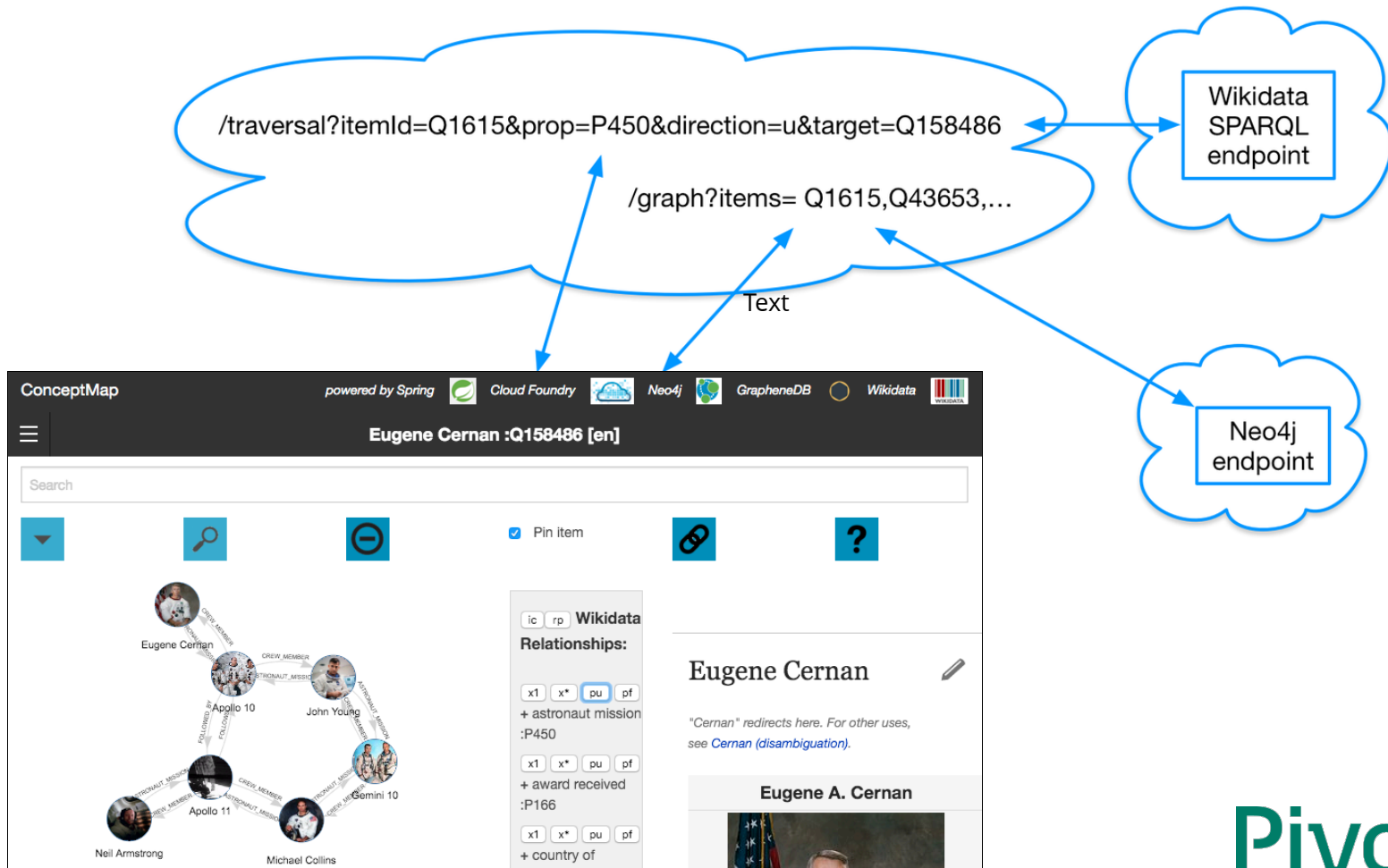
Rest Repositories ✕

Generate Project ⌘ + ↵

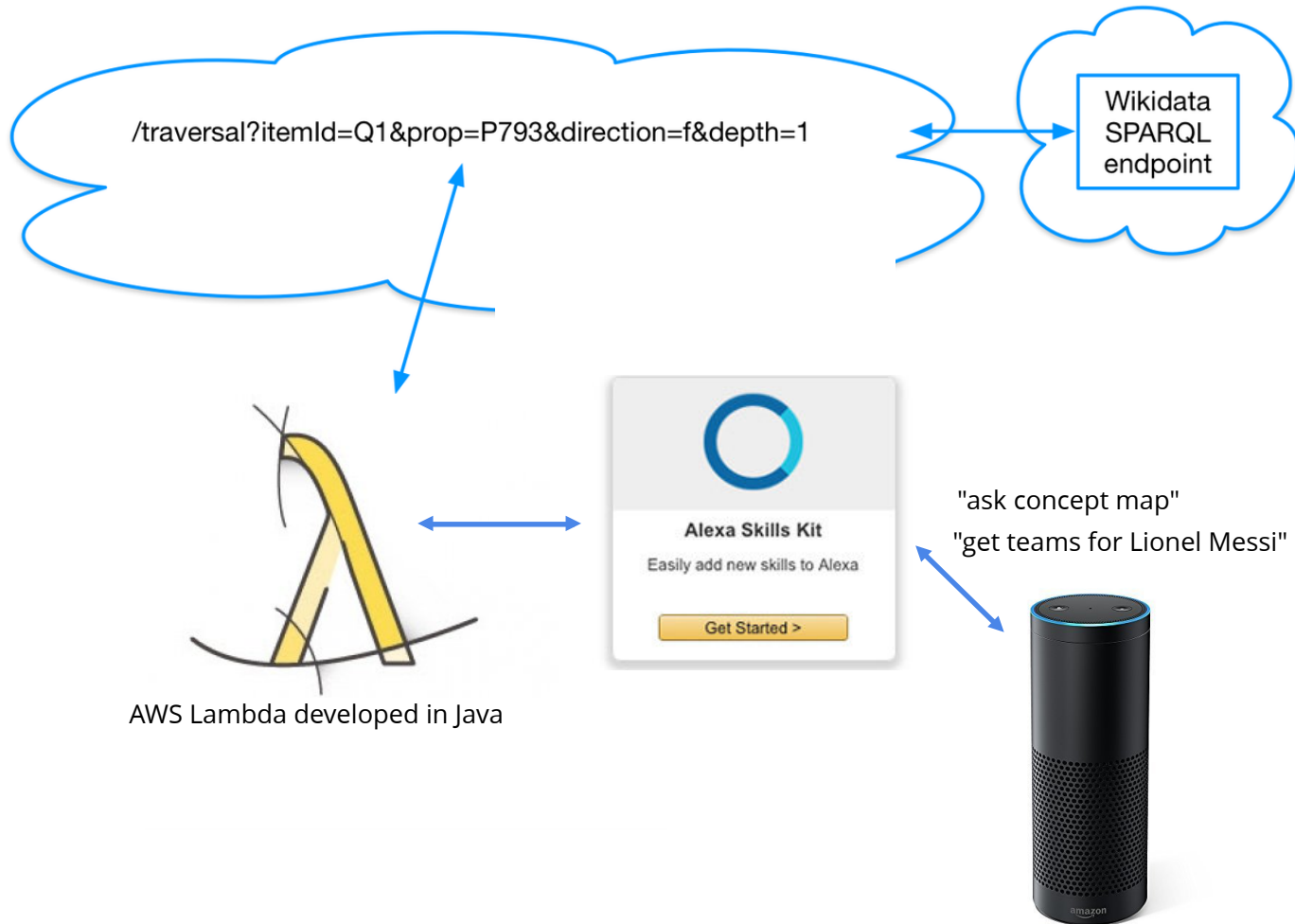
Don't know what to look for? Want more options? [Switch to the full version.](#)

Use case: Degrees of separation

(shortest path, like Kevin Bacon game)



Use case: Amazon Alexa

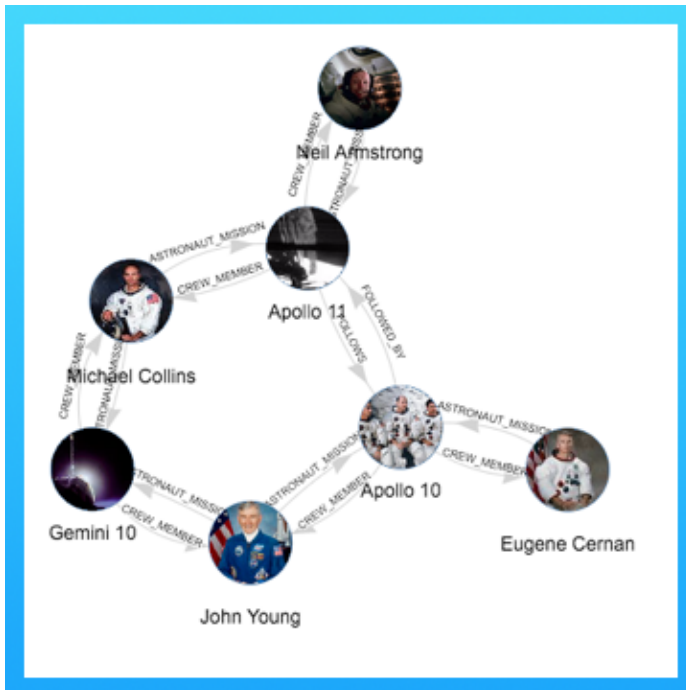


Hope you enjoyed

Navigate **All** the Knowledge

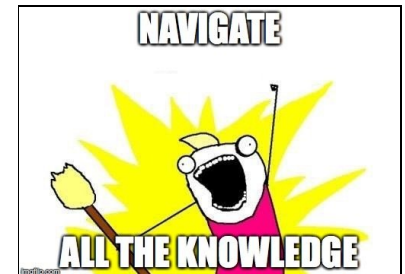
ConceptMap.io from a technical perspective

<http://ConceptMap.io>



James L. Weaver
Developer Advocate

PivotalTM



Twitter: @JavaFXpert

Email: jweaver@pivotal.io

<http://JavaFXpert.com>

<http://CulturedEar.com>