

基于LOD技术的 知识优化和知识表示

富士通研究所 西野文人

nishino@jp.fujitsu.com

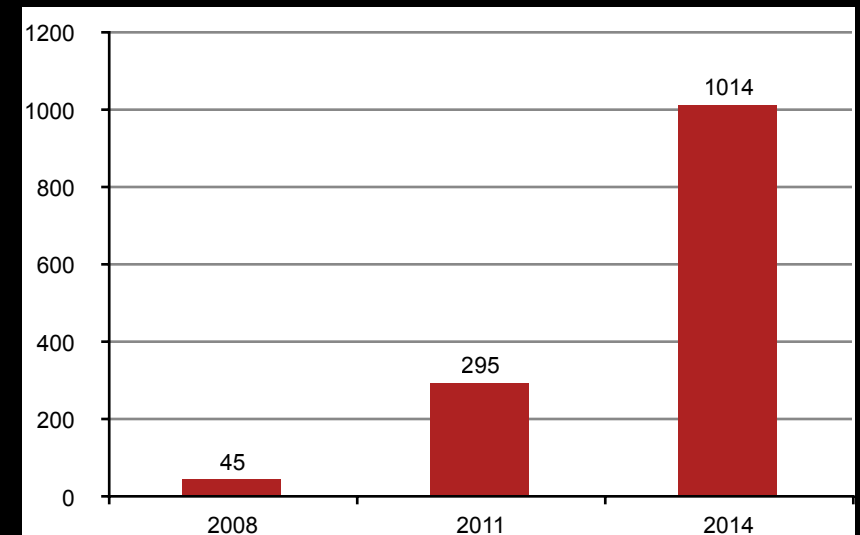
ORCID: 0000-0001-7368-4923

Outline

- 知识集成：LOD4ALL简介
- 知识表示：LOD4ALL Frontend
- 知识优化：推定DBPeida type

LOD的现状

- LOD的数量越来越多了



LOD公开网站数

- 分别公开

- DBpedia : <http://dbpedia.org/resource/xxx>
- GeoNames : <http://sws.geonames.org/nnn>
- NYTimes : <http://data.nytimes.com/nnn>
- ...

- SPARQL endpoint利用率很低 (46%)



LOD的问题 1

- 我要的数据在哪儿？

应用程序
开发者



如不下载，不知道数据的内容

因特网

?

LOD的问题 2

- 想利用数据, 可是做应用程序很辛苦

理解元数据的构造与词汇

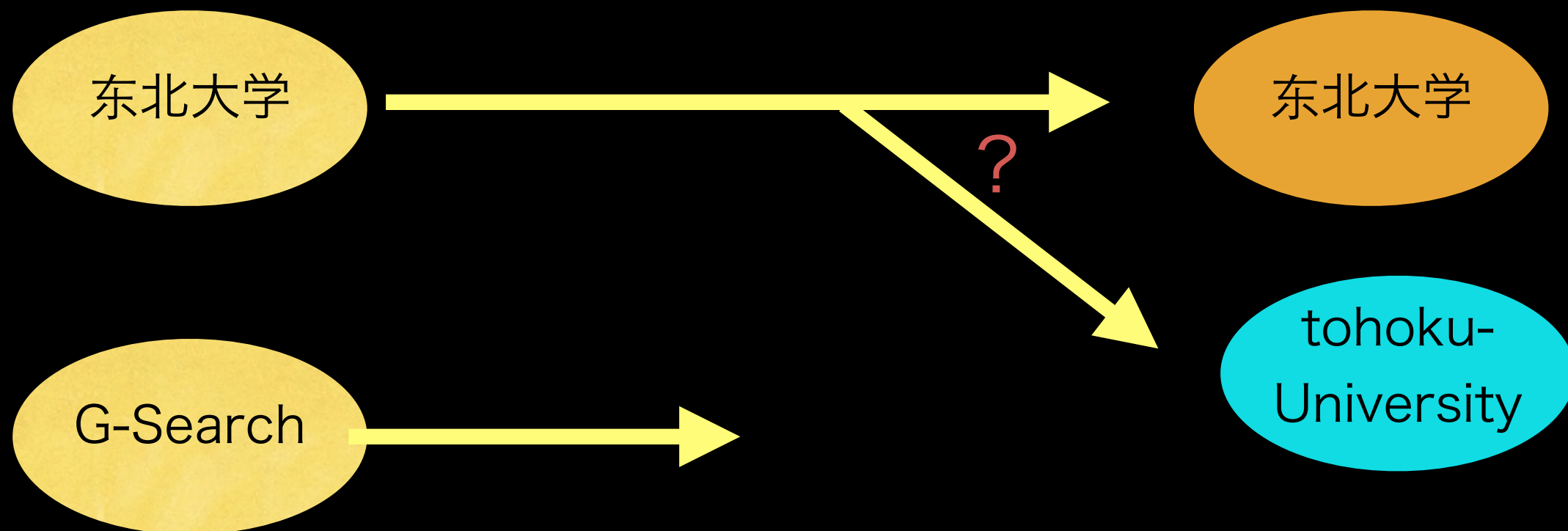
SPARQL搜索

数据变换

视觉化

LOD的问题 3

- 相同的个体之间还没描述同一性的属性
- 需要的信息在LOD里还没有



谋求解决

- 问题 1 :
 - 提供一个如本站所示的可用数据集的目录, 而是促成和推进开放式数据的使用 → LOD4ALL
- 问题 2 :
 - 提供应用程序开发环境 → LOD4ALL frontend
- 问题 3 :
 - 补充在现在的LOD范围内缺信息 →
 - a) 基于判定关联性 链接 (今天不说)
 - b) 基于推论 知识优化
 - c) 基于Web信息抽取 知识获取 (今天不说)

LOD4ALL
(公开服务)

LOD4ALL简介

- <http://lod4all.net> 公开中(日文, 英文, 中文)
- 大約68亿triples
- 为LOD的使用 提供一站式入门站点
- 提供一个使用LOD的应用程序开发平台
- 功能
 - 数据集搜索
 - 实例搜索
 - 用SPARQL搜索
 - 编程接口 (API)



数据集搜索

- 488datasets
- 搜索dataset 名称, 概要等等
- dataset信息也是 RDF, 所以可以用 SPARQL搜索



LOD4ALL里的LOD

1. 从DataHub信息获得：Dbpedia（各国語版），GeoNames, NYTimes等等
 - 条件：可以下载而且明示许可证CC-BY-SA,CC-BY,CC-0 等等
 - CC-BY-NC, 不明示 → 不能用
2. 接受个别申请（例子：J-GLOBAL Knowledge：化学辞典）
 - 跟[LinkData.org](http://linkdata.org)协作
3. 独自数据（自己做，RDF化）

LinkData.org

LinkData.org 是 LinkData开发的平台

RDF化, 发布
应用程序开发



有个地方政府用LinkData.org 开放数据

通过活动(datathon,
hackerthon)普通人做
Open Data, 发布在
LinkData.org

设备

AED, 消火栓, ...

小说里的地点

时刻表

食谱



设备 9238件, 防災4694件, ...

实例(instance)搜索

- 可按关键词搜索triple的Object Literal
- 搜索Subject IRI

LOD4ALL

利用条件 Please select 日本語 English

ホーム データセット一覧 インスタンス検索

インスタンス検索

LOD4ALLに登録されているインスタンスを検索できます。検索方法は以下の2つがあります。

- Keyword: 検索ボックスに入力された文字列を目的語に含むトリプルを検索します。
- URI: 入力したURIを主語とするトリプルを検索します。

サリチル酸

☒ Keyword ☐ URI

Filter Graph URI

Search

検索結果:2450件

ヘッダをクリックすると、インスタンスの詳細を表示します。

Category:サリチル酸	サリチル酸	Category:サリチル酸
Matched Triple: g: http://ja.dbpedia.org/ s: http://ja.dbpedia.org/resource/Category:サリチル酸 p: http://www.w3.org/2004/02/skos/core#prefLabel サリチル酸	Matched Triple: g: http://mk.dbpedia.org/ s: http://mk.dbpedia.org/resource/サリチル酸 p: http://www.w3.org/2000/01/rdf-schema#label α:サリチル酸	Matched Triple: g: http://ja.dbpedia.org/ s: http://ja.dbpedia.org/resource/Category:サリチル酸 p: http://www.w3.org/2004/02/skos/core#prefLabel α:サリチル酸
200907063992207037 Matched Triple: s: http://stirdf.jst.go.jp/rdf/chemical s: http://stirdf.jst.go.jp/id/200907063992207037 p: http://www.w3.org/2000/01/rdf-schema#label サリチル酸	200907055170183958 Matched Triple: g: http://stirdf.jst.go.jp/rdf/chemical s: http://stirdf.jst.go.jp/id/200907055170183958 p: http://vocab.jst.go.jp/terms/sti#systematic-name α:サリチル酸/鉛,(1:1)	Category:サリチル酸 Matched Triple: g: http://ja.dbpedia.org/ s: http://ja.dbpedia.org/resource/Category:サリチル酸 p: http://www.w3.org/2000/01/rdf-schema#label α:サリチル酸

ja.dbpedia

日化辞

確認实体(entity)

■ 利用条件

Please select

日本語

English

LOD4ALL

ホーム | データセット一覧 | インスタンス検索 | AP

インスタンス詳細

Salicylic Acid

- URI: <http://stir.df.jst.go.jp/id/200907063992207037> ☞
- Class: <http://vocab.jst.go.jp/terms/sti#Chemical> ☞

インスタンス情報

虫眼鏡をクリックするとLOD4ALLへの検索を行います。URIのリンクはそのリソースへのリンクです。

Predicate	Object
identifier http://purl.org/dc/elements/1.1/identifier ☞	J2.370A
chem-formula http://vocab.jst.go.jp/terms/sti#chem-formula ☞	C7-H6-O3
chem-formula http://vocab.jst.go.jp/terms/sti#chem-formula ☞	C7H6O3
common-name http://vocab.jst.go.jp/terms/sti#common-name ☞	2-Hydroxybenzenecarboxylic acid
common-name http://vocab.jst.go.jp/terms/sti#common-name ☞	2-Hydroxybenzoic acid
common-name http://vocab.jst.go.jp/terms/sti#common-name ☞	NSC-180
common-name http://vocab.jst.go.jp/terms/sti#common-name ☞	Retarder W
common-name http://vocab.jst.go.jp/terms/sti#common-name ☞	SA
common-name http://vocab.jst.go.jp/terms/sti#common-name ☞	SAX
common-name http://vocab.jst.go.jp/terms/sti#common-name ☞	Salicylic acid

公开编程接口 (API)

- 提供Sparql Endpoint
- 提供应用程序组建环境 (Javascript+SPARQL)

用SPARQL搜索

您可以对LOD4ALL网站执行SPARQL

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>
PREFIX prop-ja: <http://ja.dbpedia.org/property/>

SELECT * WHERE{
  <http://dbpedia.org/resource/Tokyo> ?p ?o .
} limit 100
```

显示语言信息和数据类型

运行查询

状态 Ready

LOD4ALL

ホーム | データセット一覧 | インスタンス検索

API

API利用方法 | アプリ一覧

検索API (LFASparql)

LFASparqlについて

LFASparqlはLOD4ALL上のリソースを検索するためのJavaScriptライブラリです。LFASparqlを利用することにより、LOD4ALLに登録されているデータセットのリソースを横断的に検索することが出来ます。

- 利用方法
- サンプルコード

ページトップへ

1. 利用方法

以下の手順で利用することができます。

1. jQuery, LFASparqlの読み込み

LFASparqlは内部でjQueryを利用しています。利用のためにはまずjQuery, LFASparqlを読み込んでください。jQueryのバージョンは1.5以降をご利用ください。またLFASparqlについては、"http://lod4all.net/api/lfasparql-1.1.1.js"を参照してもよいですし、ダウンロードして利用してもかまいません。

```
1 | <script type="text/javascript" src="http://code.jquery.com/jquery-1.11.1.min.js"></script>
2 | <script type="text/javascript" src="http://lod4all.net/api/lfasparql-1.1.1.js"></script>
```

2. LFASparqlクラスのインスタンスを作成

```
1 | var lfasparyl = new LFASparql();
```

3. 検索リクエストの送信

executeSparqlメソッドを利用することで、LOD4ALLに検索リクエストを送信することができます。executeSparqlの引数にはappId(アプリケーションID)、検索するためのSPARQL文、検索成功時・失敗時のコールバック関数を含む連想配列を入れます。

```
1 | // アプリケーションIDには"newsaykncb"を指定してください!!
2 |
3 | var sparql_query = "select distinct * where { <http://ja.dbpedia.org/resource/東京都> ?p ?o . }";
```


样本应用程序

- 样本应用程序公开中（日文版，英文版，中文版）

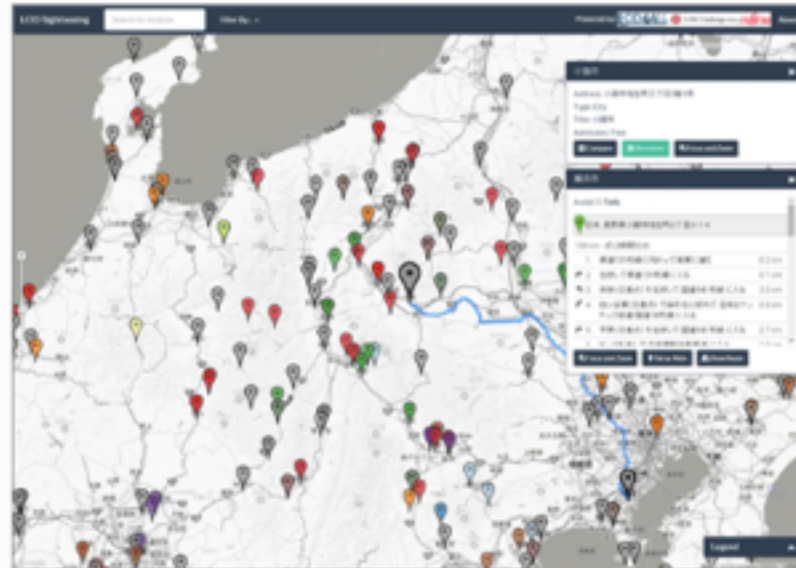
アプリ一覧

LOD4ALLのAPIを利用して開発されたアプリの一覧です。本環境を利用して開発したアプリがあり、もしこちらに記載してもよいと
格ください。

観光アプリ

LinkData.orgを通じてデータをLOD4ALLにアップロードし、API経由でデータを利用しているサンプルアプリケーションです。

- 作成者: LOD4ALL
- URL: <http://lod4all.net/apps/sightseeing/jp/>
- 利用データセット: [日本の観光地](#)



自治体比較アプリ

日本の各自治体の人口や病院数など様々な指標を組み合わせて評価指標を作り、ランキングを表示することができるサイトです。

- 作成者: LOD4ALL
- URL: <http://lod4all.net/apps/city-comp/>

比较地域特性

你住的地方什么是第一位？

日本語 ▾ あなたの住んでる街の一番を探そう！ 神奈川県 ▾ 川崎市 ▾

項目の選択

項目1 出生数 ▾ 項目2 死亡数 ▾ 演算子 項目1 - 項目2 ▾ 比較

比較結果

順位	都道府県名	自治体名	項目1 (人)	項目2 (人)	計算値
1	神奈川県	川崎市	14292	9272	5020
2	神奈川県	横浜市	32053	27304	4749
3	福岡県	福岡市	14483	10131	4352
4	埼玉県	さいたま市	11131	8645	2486
5	広島県	広島市	11478	9102	2376
6	宮城県	仙台市	9295	7371	1924
7	愛知県	豊田市	4286	2608	1678
8	千葉県	船橋市	5830	4223	1607
9	千葉県	市川市	4625	3140	1485

地図で表示

统计数据（人口, 图书馆数, 交通事故数, 老人院数, 托儿所数, 土地面积, 林野面积, 财政收入, 地方税, ...）

出生数 - 死亡数
第一位 是 川崎市

EvaCva

- 分析城市特性(EvaCva)的数据

オープンデータによる地域特性の発見
EvaCva [エヴァシーヴァ]

HOME EvaCvaとは ご意見・ご要望



地域の特性を見る

オープンデータを使って、日本全国の地域の特性を、市区町村単位で比較することができます。ブラウザは、Internet Explorer 9, 10, もしくはGoogle Chromeをご利用ください。使い方は、下記をご参照ください。

[動画を見る](#)  [PDFを見る\(約5M\)](#) 



データを取得する

EvaCvaで利用しているオープンデータは、他のデータとの連携を容易にするためLinked Open Data(LOD)として保存しています。本システムでは、そのデータを取得することができます。

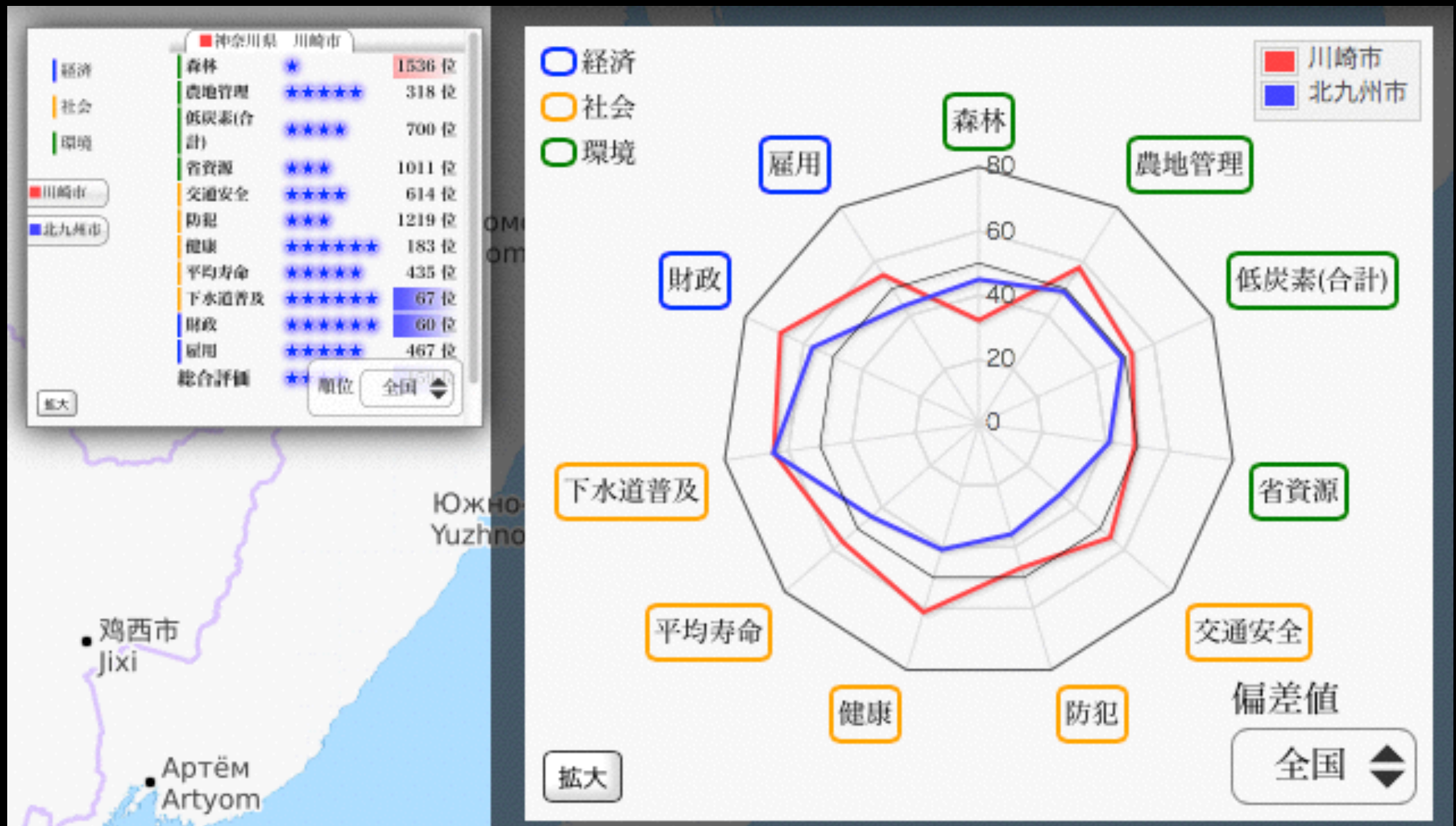
EvaCvaは、こんな時に使うと便利です

- 住民の方: 自分の住んでいる市が、どのような特性なのか知りたい
- 自治体職員の方: 来年度計画のために、自治体の課題を客観的に把握したい
- 店舗経営者の方: 現在の店舗のある地域と、新規に出店する地域の特性の違いを知りたい
- 都市開発企業の方: 開発地区の特徴を活かすため、その地域の特性を知りたい
- 教育関係の方: 授業や研究に、全国のまちの特性を簡単に調査したい

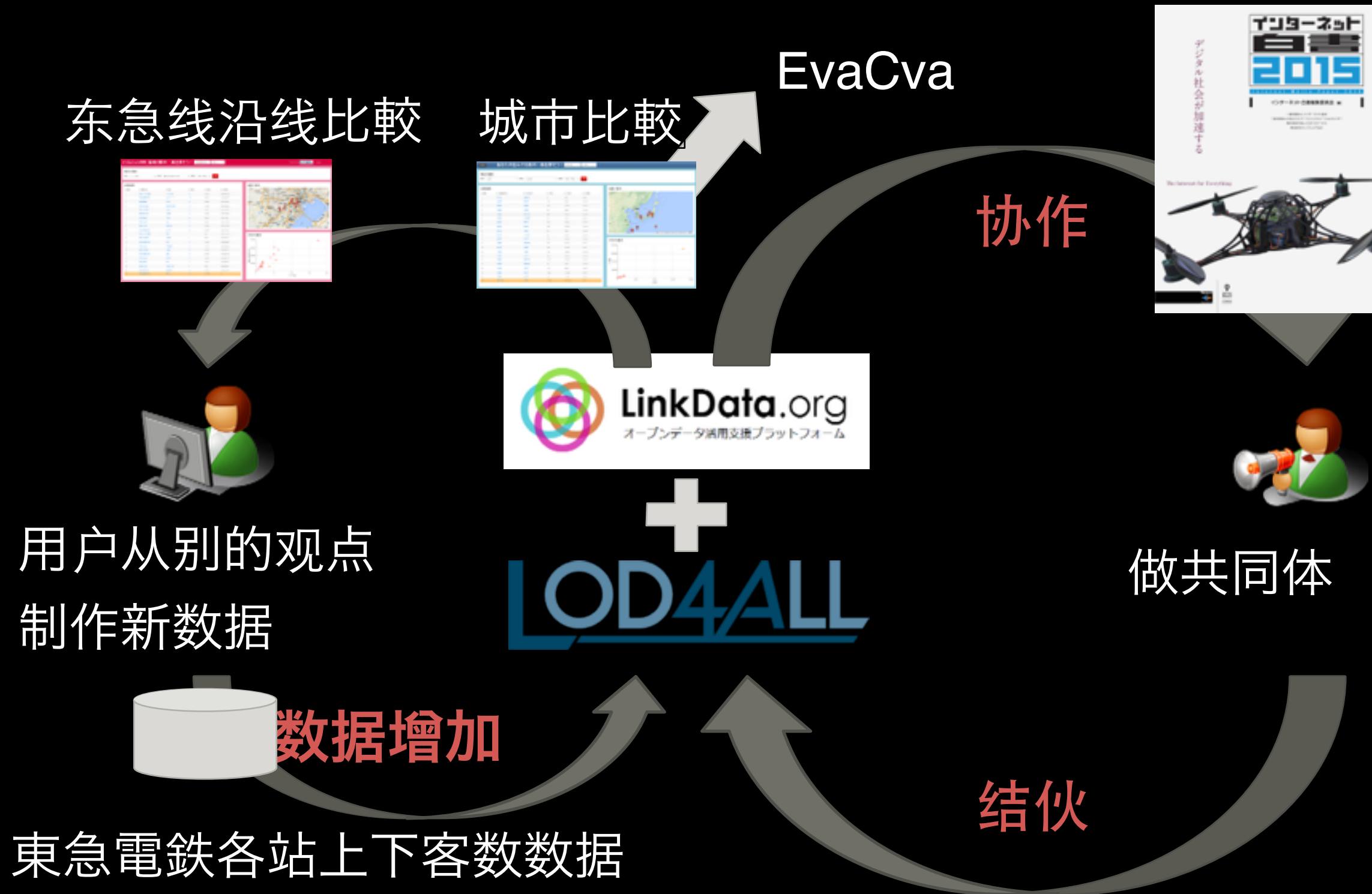
<http://evacva.net>

EvaCva

- <http://evacva.net>



循环系统



日英中样本

世界遺産一覧Timeliner

European Footballer of the Year winners

できるアプリケーション

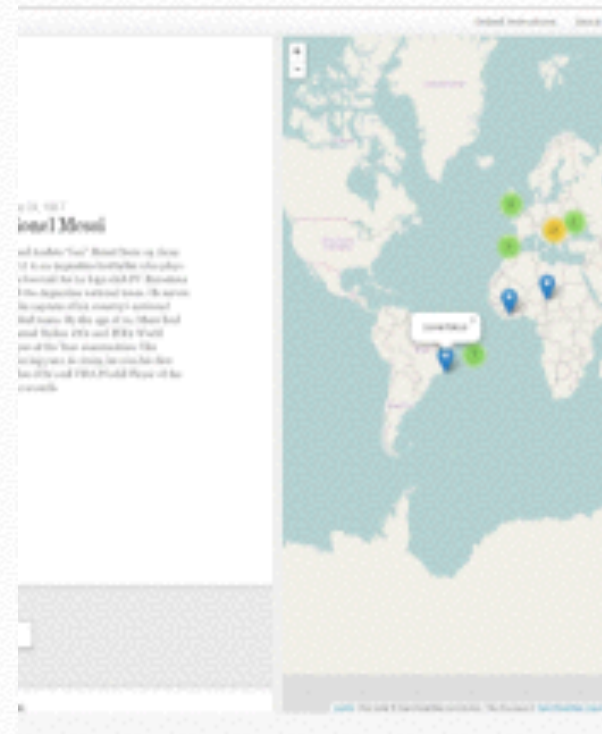
中国君主列表

- 创建者: LOD4ALL
- URL: http://lod4all.net/apps/rulers_of_china/
- 数据集: [DBPedia](#)

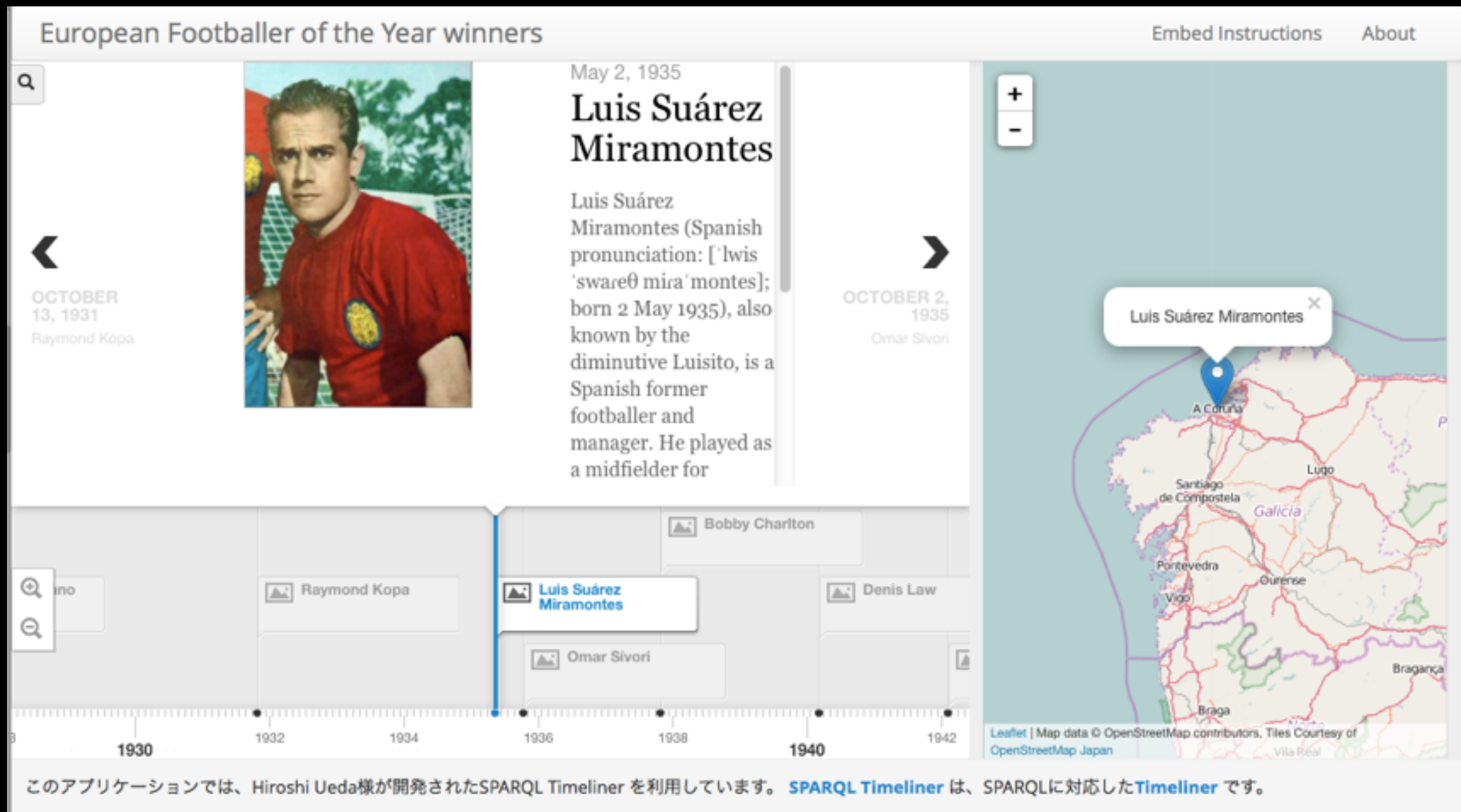
ALL

4all.net/apps/ballon_dor/
[dia](#)

apps/world_heritage/
[dia in Japanese](#)



European Football of the Year Winner



中国主君列表

中国君主列表

211
司马昭



259
秦始皇

秦始皇（前259年正月—前210年七月丙寅），嬴姓，名政，生于邯郸，秦庄襄王之子，出生于赵国首都邯郸。唐代司马贞在《史记索隐》引述《世本》称其为赵政，但仍有争议。後世俗称嬴政或秦王政，亦被某些文学作品称为「祖龙」。自谥「始皇帝」。是由

JULY 21, 541
隋文帝

汉宣帝

孙策

司马昭

曹操

汉献帝

刘备

曹丕

秦始皇

天镇县

阳原县

浑源县

灵丘县

阜平县

平山县

平定县

乐平镇

11县

涉县

临漳县

大名县

肥乡县

馆陶县

临清市

魏城市

济宁市

德州市

南宮市

邢台市

沙河市

邢台市

元氏县

新乐市

安国市

任丘市

河间市

沧州市

深州市

晋州市

石家庄市

高碑店市

保定市

怀来县

延庆

密云县

北京市

4

秦始皇

Date: 0259

Leaflet | Map data © OpenStreetMap contributors, Tiles Courtesy of OpenStreetMap Japan

このアプリケーションでは、Hiroshi Ueda様が開発されたSPARQL Timeliner を利用しています。SPARQL Timeliner は、SPARQLに対応したTimeliner です。

LOD4ALL Frontend
(近日公开预定)

LOD Browser

- 把RDF的内容出示



Pubby



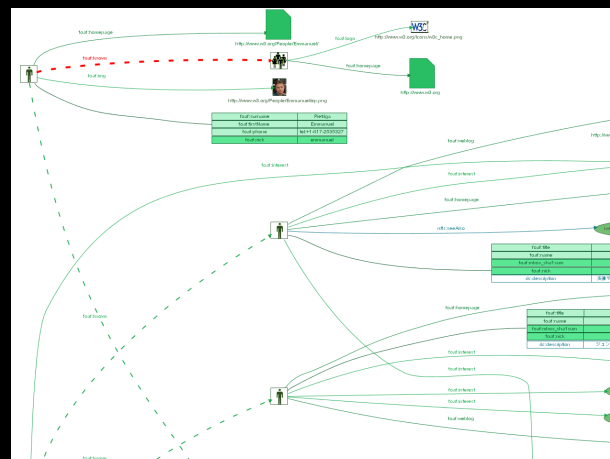
Graphite



OpenLink Data Explorer(ODE)



VisualRDF



IsaViz

LOD4ALL frontend的目的

- 用某种意图发布RDF
- 表示丰富的信息
- 容易使用LOD4ALL frontend

LOD4ALL frontend

- 开发Linked Data应用程序的平台
- 选择Gadget, 搭配
 - Gadget = SPARQL + 可视化

The screenshot displays the LOD4ALL frontend interface for the Sarcolic acid (サリチル酸) page. The URL is <http://xtirdf.jst.go.jp/id/200907063992207037>.

The interface is divided into several sections:

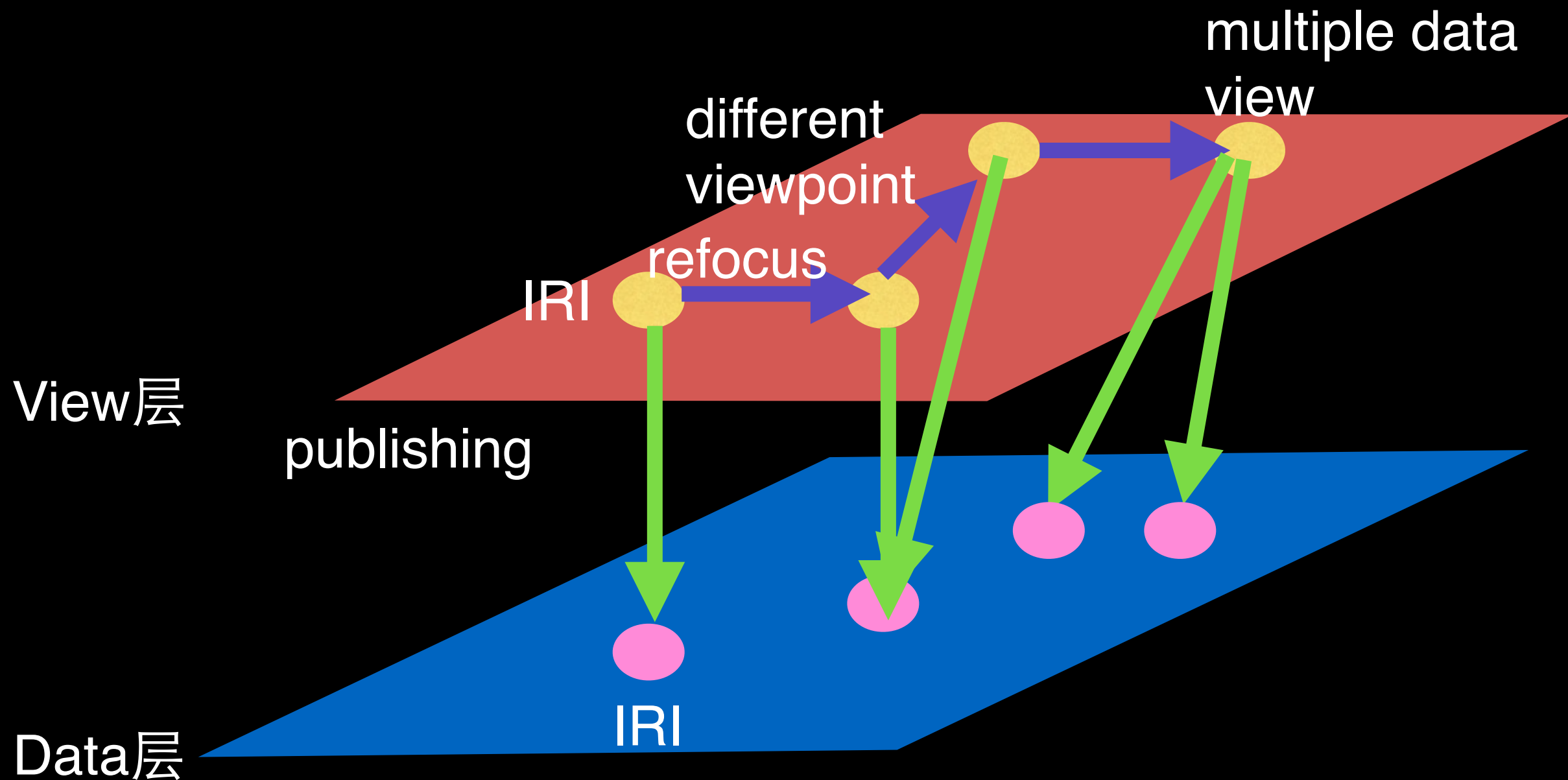
- NSDC 図**: Displays the chemical structure of Salicylic acid (2-hydroxybenzoic acid).
- DSPedia リソース情報**: Lists various resources related to Salicylic acid, including Wikidata, Wikispecies, and Wikisource.
- リソース情報**: A table listing resources and their corresponding URIs.
- DSPedia カテゴリ情報**: A section for category information, currently showing "Category:サリチル酸".
- 関連化合物**: A table listing related compounds and their corresponding URIs.
- nikkaji_related_tree**: A tree diagram showing the relationships between Salicylic acid and other compounds.

LOD4ALL frontend

- 特点:

1. 数据驱动出版 (data-driven dynamic publishing) & 领航 (navigators)
2. 按范畴图表(type-specific views) & gadget-based views
3. SPARQL & Visualization Construction

- 数据驱动出版 (data-driven dynamic publishing) & 领航 (navigators)



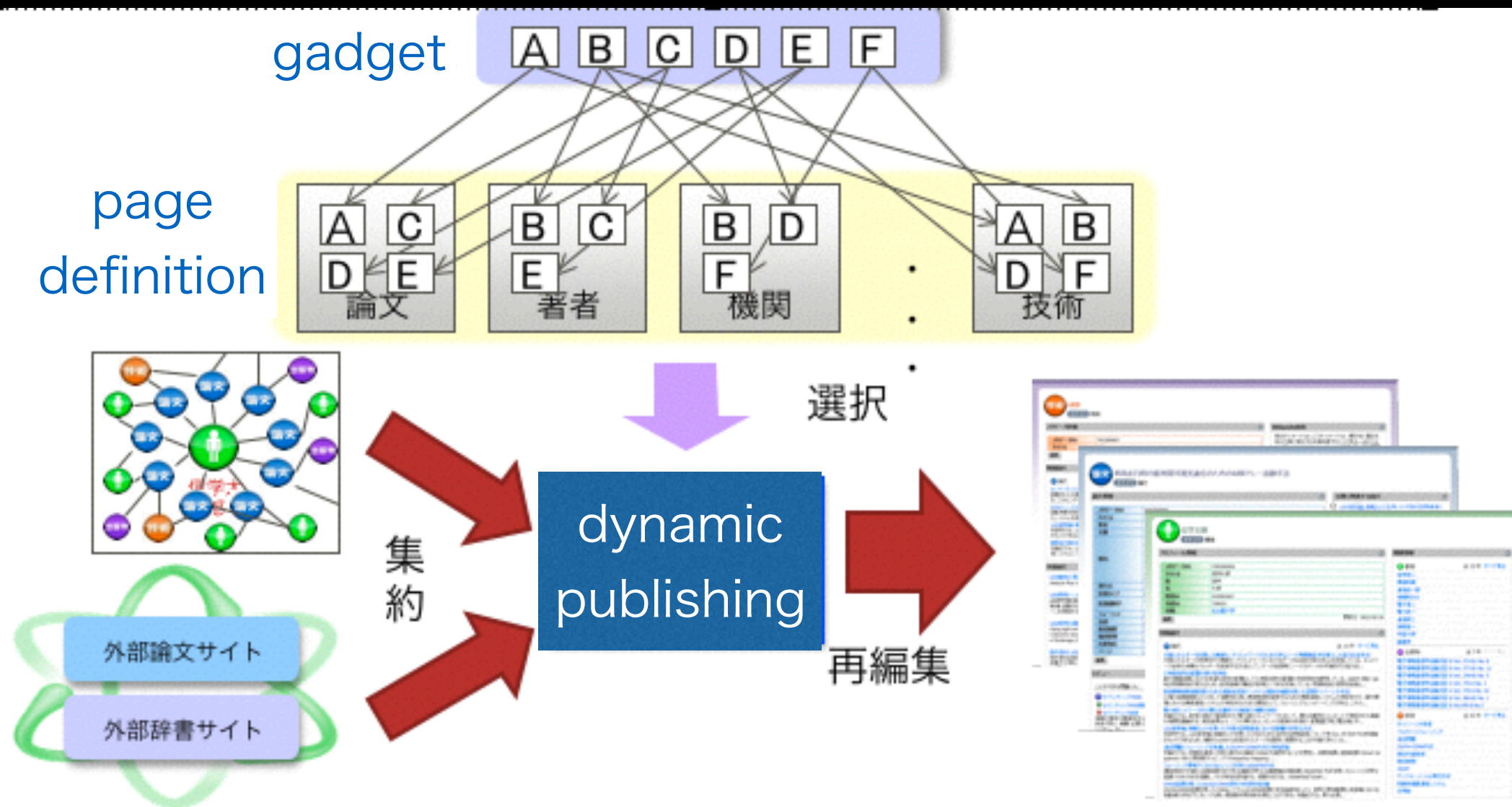
multiple data view



比較企业数据

Type specific views & Gadget-based views

- gadget = SPARQL + 可視化
- page definition(范畴, 观点) = gadget1 + gadget2 + ...



SPARQL & Visualization Construction

運用管理 ガジェット定義参照

ID	Ho_ScatterPlot01
タイトル	ScatterPlotSample
説明	散布図サンプル。農林水産植物の種類_和名の年別の登録数
公開	☒ 公開する
SPARQLクエリ	<pre>1 select ?o ?y (count(*) as ?c) where { 2 select ?o (substr(?d, 1, 4) as ?y) where { 3 ?s <http://linkdata.org/property/rdf1s2573i#農林水産植物の種類_和名> ?o . 4 ?s <http://linkdata.org/property/rdf1s2573i#登録年月日> ?d . 5 } 6 group by ?o ?y 7 order by ?o 8 # %URI%</pre> 拡大
実行プログラム	<pre>1 function getResult(json) { 2 console.log(json); 3 var i=0; 4 var preValue=""; 5 json.results.bindings.forEach(function(x) { 6 if (preValue != x.o.value) { 7 i+=1; 8 preValue=x.o.value; 9 } 10 x.o.value=i; 11 }); 12 console.log(json); 13 // displayScatterPlot(json, gadgetId, width, height, labelX, labelY, varX, varY, radius, minRadius, 14 // maxRadius, marginX, marginY) 15 displayScatterPlot(json, "KIDK", "XWIDTHK", "XHEIGHTK", "Year", "Kind", "y", "o", "c", 0, 10, 5, 16 5); 17 }</pre> 拡大
呼び出し関数名	getResult
横幅 (pixel)	400
縦幅 (pixel)	200
テスト	URI: http://linkdata.org/resource/rdf1s2573i#8787 SPARQLクエリのテスト オプション: 実行プログラムのテスト

SPARQL

データ変換 & 可視化

sgvizler → d3

流用作成

戻る

LOD frontend 示例



乾健太郎

カテゴリ 人物

画像



分野

person_area_list

callret-0	callret-1
健康安全確保総合研究, 心理学, 健康安全確保総合研究分野, 情報通信技術, 情報学	出典
知能情報学, 情報学	出典

拡大

論文発表時所属

person_organization_list_rq

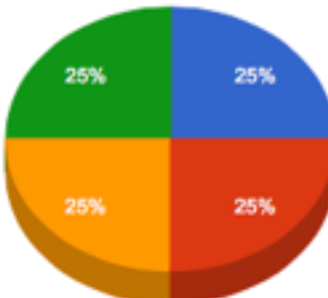
year	Organization	count
2012	東北大学	1
2010	東北大学	2
2010	奈良先端科学技術大学院大学	1
2009	情報通信研究機構	1
2007	奈良先端科学技術大学院大学	1
2001	北海道大学	1

拡大

所属タイムライン

共同研究相手組織

person_coauthor_organization_group_pie_rq



- 名古屋大学
- 北海道大学
- 岡山大学
- 京都大学

メタデータ詳細

メタデータID	乾健太郎
タイトル	乾健太郎
同一	http://i-scover.ieice.org/iscover/resource/AUTHOR-8E28B6A1-208B-AE4E-277B-8F2012016F37 , http://i-scover.ieice.org/iscover/resource/AUTHOR-B7C2B5E5-19D6-D811-B627-9FF7A36E8EC6 , http://i-scover.ieice.org/iscover/resource/AUTHOR-D0E48DB0-F74E-00D2-469F-9C66882188CB , http://i-scover.ieice.org/iscover/resource/AUTHOR-DCD8513C-4610-D5E8-4E34-189289117392
別表記	Kentaro Inui, 乾健太郎
本人画像	http://ws013.lia.flab.fujitsu.co.jp/bacoliab/img/portrait/%E4%B9%BE%E5%81%A5%E5%A4%AA%E9%83%8E.jpg

編集 RDF/XML表示 RDF/N3表示 ピックアップ

更新日 2014-11-13

論文リスト

person_paper_list_rq

year	type	Paper	keywords
2012	技	Online Large-margin Weight Learning for First-order Logic-based Abduction	abduction; Large-margin training; Latent variables; Logic-based Reasoning; online learning; Passive Aggressive algorithm; Structured learning
2010	論	テキスト情報分析のための判断情報アノテーション	コーパス; モデル解析; 情報抽出; 条件付確率場; 自然言語処理; conditional random fields; corpus construction; information extraction; modality analysis; natural language processing

左側の重み付け

拡大

I-Scover

- 日本電子情報通信学会 (IEICE)
- Linked Data & gadget based 网站

The screenshot displays the I-Scover website interface. At the top, the header includes the I-Scover logo, the text "IEICE Knowledge Discovery 電子情報通信学会 文献検索システム", and navigation links for "会員専用" and "ENGLISH". A search bar is located on the right.

The main content area is divided into two columns. The left column features a profile for author 西野文人 (Fuminori Nishino), including a "詳細情報" (Detailed Information) section with a table of his name and affiliation, and a "関連文献" (Related Literature) section listing his publications.

名前	西野文人
所属機関	富士通研究所

更新日 2014-02-28

The right column contains a "関連メタデータ" (Related Metadata) section, which includes a search filter for "キーワード" (Keywords) and a list of related documents. Below this, there is a section for "I-Scover シンポジウム" (I-Scover Symposium) and "I-Scover チャレンジ" (I-Scover Challenge).

At the bottom, the "外部サイト検索結果" (External Site Search Results) section shows search results from "Cinii", displaying a total of 48 items.

知识优化

问题

- LOD4ALL Frontend 以rdf:type范畴决定怎么表示
- 可是, 没有rdf:type范畴的个体很多
- → 推定rdf:type范畴

DBpedia的真实情况

- 怎么知道在dbpedia里的实体数？
- 除了wikipedia页相当的页还有
 - Wikipedia的 页信息 (foaf:Document)
 - Wikipedia的 范畴信息 (skos:Concept)
 - Property信息 (rdf:Property) 等等
- 有相当的Wikipedia链接 (foaf:isPrimaryTopicOf) 就作为dbpedia实体，计数

```
SELECT (COUNT(DISTINCT ?s) AS ?no)  
       {?s foaf:isPrimaryTopicOf []}
```

DBpedia的真实情况

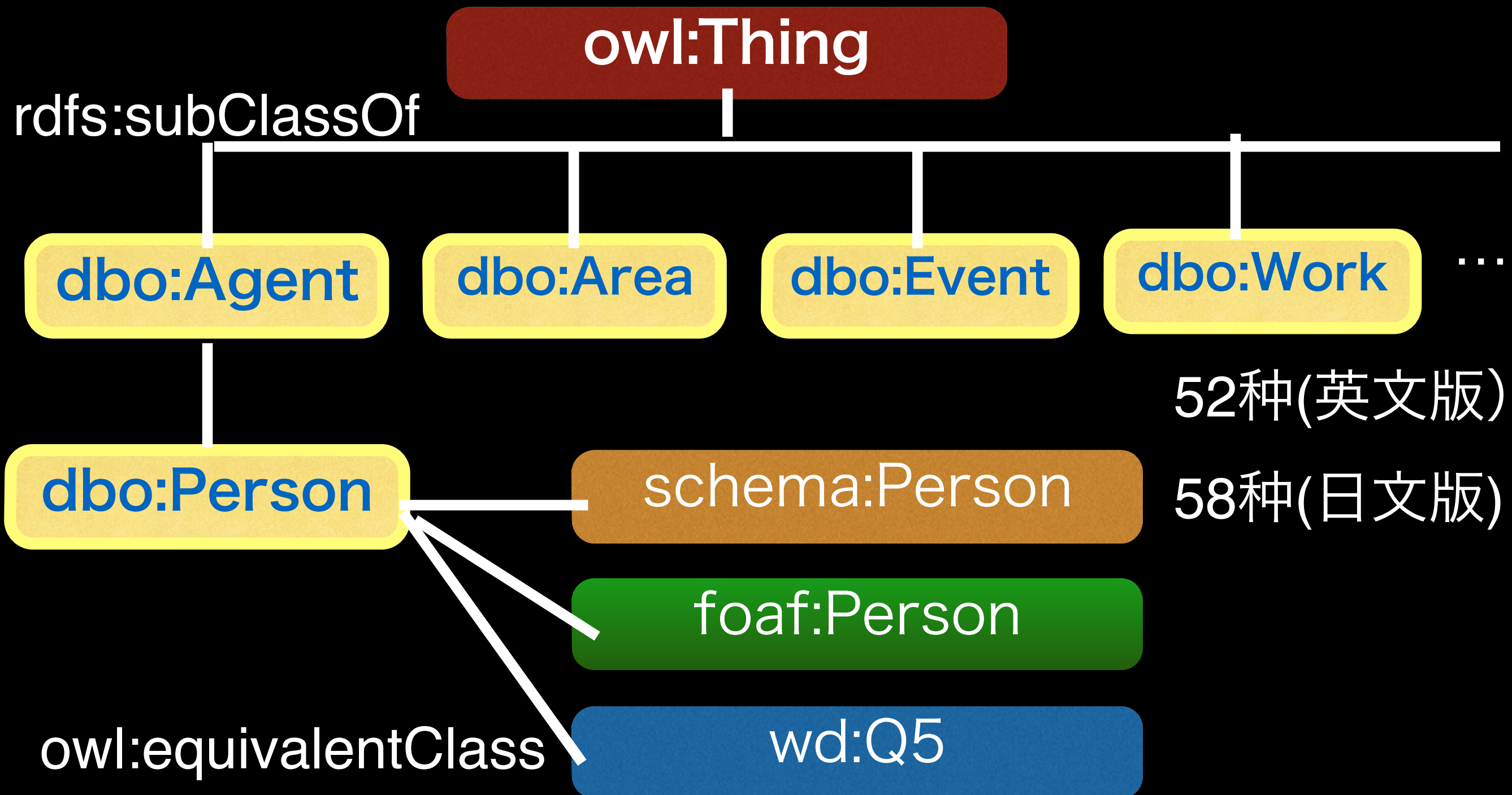
- 对个体有没有rdf:type

英語版	个体数	割合
rdf:type有	4,055,960	35.2%
rdf:type無	7,463,183	64.8%

日本語版	个体数	割合
rdf:type有	332,752	22.0%
rdf:type無	1,180,840	78.0%

没有rdf:type的个体很多

rdf:type范畴



着手解决问题

- 在实体上附加dbo的Class (rdf:type范畴)
- 尽可能附加细粒度的Class (rdf:type范畴)

附加的例子

- Original

- [<http://ja.dbpedia.org/resource/蔡依林>](http://ja.dbpedia.org/resource/蔡依林)

[rdfs:comment](#)

"蔡依林（拼音:Cài Yīlín、ツァイ・イーリン、英語名：Jolin Tsai、ジョリン・ツァイ）は、台湾を中心として香港などの中華圏芸能界で活動する女性歌手。輔仁大学英語文学科卒業。”@ja;

[dcterms:subject](#)

<http://ja.dbpedia.org/resource/Category:台湾の歌手>,

<http://ja.dbpedia.org/resource/Category:新北市出身の人物>,

<http://ja.dbpedia.org/resource/Category:台湾の芸能人>,

附加的例子

- 方法：
 - 演绎推论
 - $?s \text{ owl:sameAs/rdf:type } ?t \rightarrow ?s \text{ rdf:type } ?t$
 - 归纳推论
- 附加 (rdf:type范畴)

<http://dbpedia.org/ontology/Person>,
<http://dbpedia.org/ontology/Actor>,
<http://dbpedia.org/ontology/Singer>,
<http://dbpedia.org/ontology/Artist>

附加数

- 全体

	日本語 DBpedia	日本語 推定 rdf:type	合計	英語 DBpedia	英語 推定 rdf:type	合計
人物	10.7万	12.4万	23.1万 (216%)	213万	4.4万	217.4万 (102%)
組織	4.5万	2.3万	6.8万 (151%)	22万	10万	32万 (145%)
Work	9万	3.2万	12.2万 (136%)	39.6万	12.8万	52.4万 (132%)
Species	1.3万	0.1万	1.4万 (108%)	28.3万	0.07万	28.4万 (100.4%)
Place	4.9万	5.4万	10.3万 (210%)	72.5万	19.9万	92.4万 (127%)

附加数

- ja.dbpedia subClassOf dbo:Person 一部

	日本語DBpedia 原版	日本語 推定rdf:type値	合計
Athlete	280,976	20,234	301,210 (107%)
Artist	1,000,934	19,208	1,020,142 (102%)
Politician	36,221	10,076	46,297 (128%)
Scientist	20,695	8,934	29,629 (143%)

效果

- 附加rdf:type, 有什么好处？
 - 按范畴表示 (default → 合适的表示)
 - Entity Linkage的精确度提高
 - 使应用程序的开发变得简单

今后的课题

- rdf:type附加后
 - 英文DBpedia (35% → 39%)
 - 日文DBpedia (22% → 37%)
 - 还要增加type附加
- 查出dbpedia的不一致
 - Perl, Prolog, C → dbo:ProgrammingLanguage
 - Abstract_syntax, Metacharacter →
dbo:ProgrammingLanguage

总结

LOD4ALL展望

- LOD4ALL（数据集／实例搜索， SPARQL， API）：已经公开了
- LOD4ALL Frontend
 - 浏览→尽快公开
 - 制作Web页：暂时公开范围只限定特别用户
- DBpedia补充数据
 - 今后探讨

LOD4ALL展望

- LOD4ALL是用LOD的开发环境
- 不仅提供API, 而且提供数据
 - 追加不足的信息
 - 使Class名与Property名标准化
- 有数据不是优势, 用数据干什么 (解决什么) 是很重要
- 方针: 尽可能公开 (数据, 工具)
- 需要大家的协助



shaping tomorrow with you

谢谢