ulm university universität ulm

Semantic Web Grundlagen

Linked Data

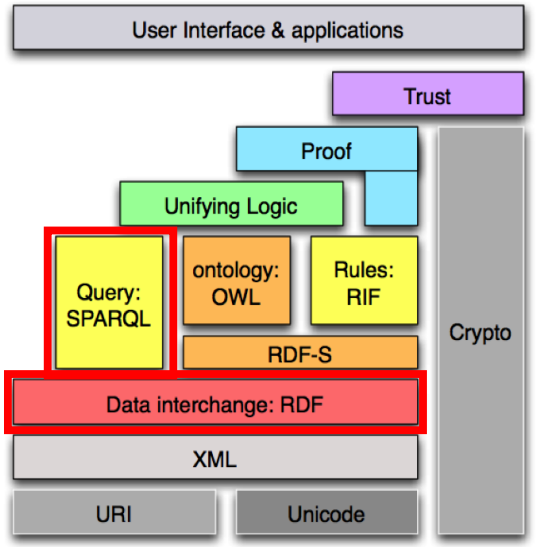
Birte Glimm
Institut für Künstliche Intelligenz | 02. Feb 2012

Organisatorisches: Inhalt

Einleitung und XML	17. Okt	Hypertableau II	12. Dez
Einführung in RDF	20. Okt	Übung 4	15. Dez
RDF Schema	24. Okt	SPARQL Syntax & Intuition	19. Dez
fällt aus	27. Okt	SPARQL Semantik	22. Dez
Logik – Grundlagen	31. Okt	SPARQL 1.1	9. Jan
Übung 1	3. Nov	Übung 5	12. Jan
Semantik von RDF(S)	7. Nov	SPARQL Entailment	16. Jan
RDF(S) & Datalog Regeln	10. Nov	SPARQL Implemetierung	19. Jan
OWL Syntax & Intuition	14. Nov	Ontology Editing	23. Jan
Übung 2	17. Nov	Übung 6	26. Jan
OWL & BLs	21. Nov	Ontology Engineering	30. Jan
OWL 2	24. Nov	Linked Data	2. Feb
Tableau	28. Nov	SemWeb Anwendungen	6. Feb
Übung 3	1. Dez	Übung 7	9. Feb
Blocking & Unravelling	5. Dez	Wiederholung	13. Feb
Hypertableau	8. Dez	Übung 8	16. Feb

Abfragen und RIF wurde gestrichen

Linked Data

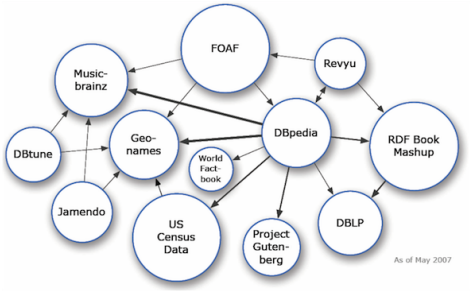


Daten im Web

- Immer mehr Websites stellen einen programmatischen Zugriff auf ihre Daten zur Verfügung
- Dabei werden Semantic Web Standards verwendet, z.B. die Linking Open Data (LOD) Initiative
  `http://www.w3.org/wiki/SweoIG/TaskForces/CommunityProjects/LinkingOpenData`
- Verwendet werden APIs, z.B. via JSON/REST

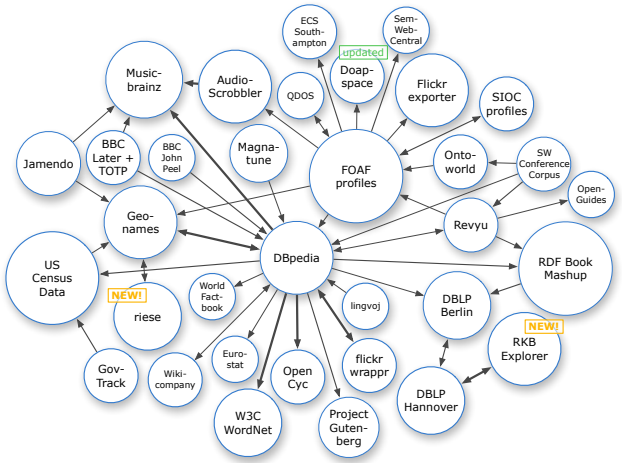
- Semantic Web Technologien vereinfachen die Integration von Daten aus verschiedenen Quellen
- Die Kombination von Daten erlaubt auf tiefere Einblicke

Linked Data im Web 01.05.2007



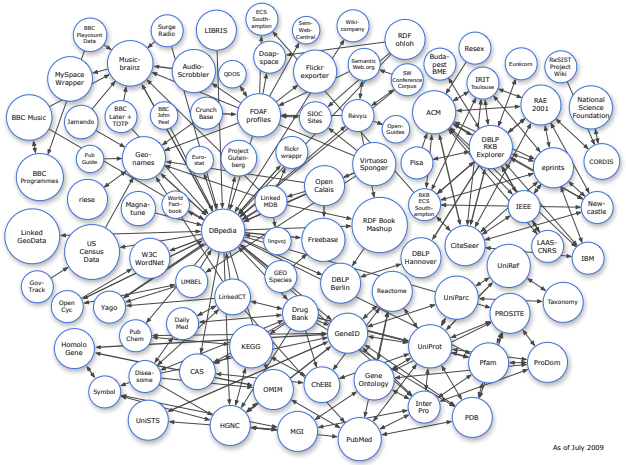
Linking Open Data cloud diagram, by Richard Cyganiak and Anja Jentzsch.
http://lod-cloud.net/

Linked Data im Web 31.03.2008



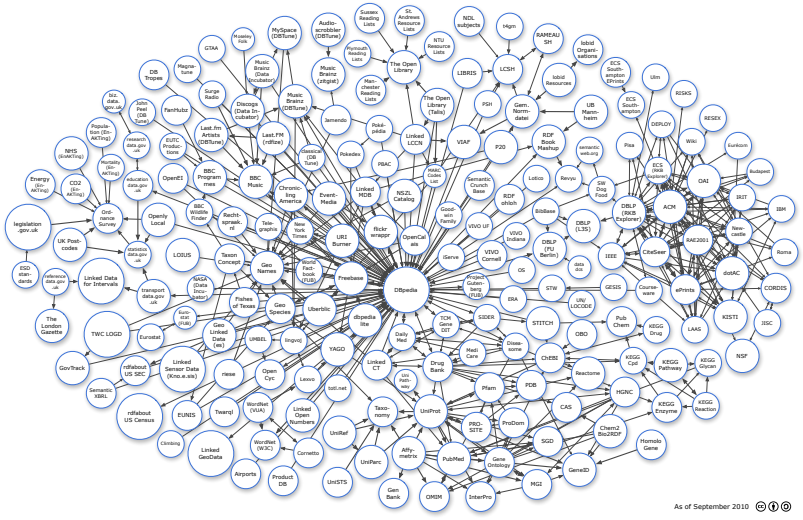
Linking Open Data cloud diagram, by Richard Cyganiak and Anja Jentzsch.
http://lod-cloud.net/

Linked Data im Web 14.07.2009



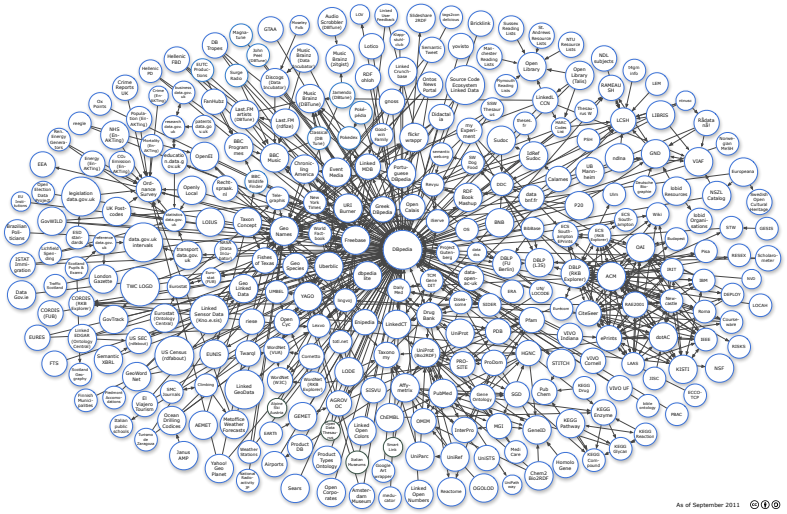
Linking Open Data cloud diagram, by Richard Cyganiak and Anja Jentzsch.
http://lod-cloud.net/

Linked Data im Web 22.09.2010



Linking Open Data cloud diagram, by Richard Cyganiak and Anja Jentzsch.
http://lod-cloud.net/

Linked Data im Web 19.09.2011



Linking Open Data cloud diagram, by Richard Cyganiak and Anja Jentzsch.
http://lod-cloud.net/

Semantic Web Technologien

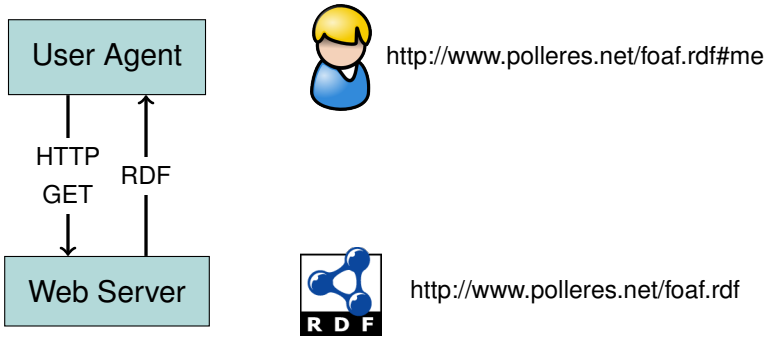
- Nützlich zum Publizieren, zum Austausch und zur Integration von Daten
- Semantic Web Technologien sind mittlerweile recht ausgereift
  - IRIs (IETF RFC 3987, 2005)
  - HTTP (IETF RFC 2616, 1999)
  - RDF (W3C Recommendation, 1999, Update in 2004)
  - RDFS (W3C Recommendation, 2004)
  - SPARQL (W3C Recommendation, 2008, Update im Moment)
  - OWL (W3C Recommendation, 2004, Update in 2009)
- Linked Data besteht aus einigen Prinzipien zum Publizieren von Datem im Web

Linked Data Principles*

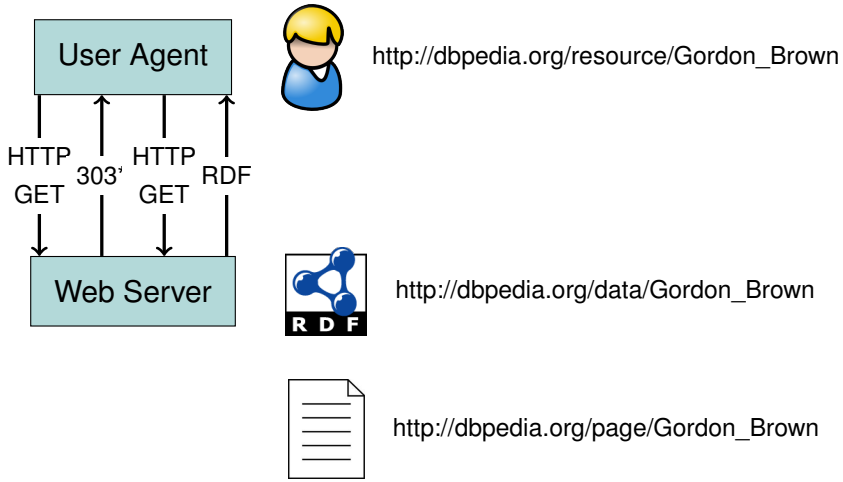
1. Use URIs to name things; not only documents, but also people, locations, concepts, etc.
2. To enable agents (human users and machine agents alike) to look up those names, use HTTP URIs
3. When someone looks up a URI we provide useful information; with 'useful' in the strict sense we usually mean structured data in RDF.
4. Include links to other URIs allowing agents (machines and humans) to discover more things

*http://www.w3.org/DesignIssues/LinkedData.html

Zusammenhang zwischen URI einer Sache und URI einer Quelle



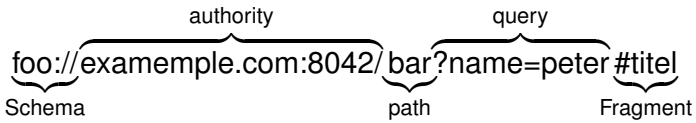
Zusammenhang zwischen URI einer Sache und URI einer Quelle



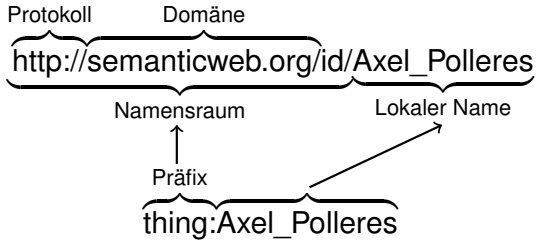
*HTTP Response Code 303: See Other

Hintergrund: Uniform Resource Identifiers

- Ein Uniform Resource Identifier ist eine kompakte Sequenz von Charakteren, die eine abstrakte oder physikalische Ressource identifizieren [RFC3986]
- Syntax
  URI = Schema ":" hier-part ["?" Abfrage] ["#" Fragment]
- Beispiel

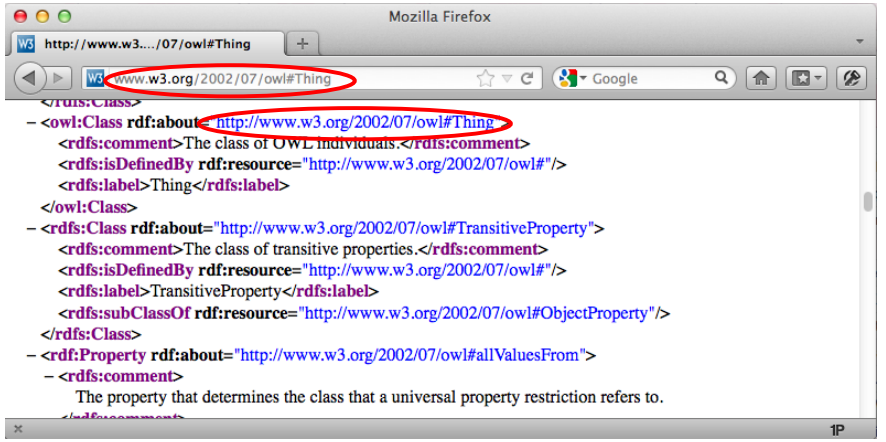


URIs/IRIs



- URIs sind “Uniform Resource Identifiers”
  - IRIs sind Unicode-basierte “Internationalized Resource Identifiers”
- Jede URI identifiziert eine Entität
- Semantic Web URIs nutzen üblicherweise HTTP
  - HyperText Transfer Protocol
  - Können idealerweise aufgelöst werden, um weitere Daten zu erhalten
  - Linked Data

Auflösung von URIs



Das HTTP Protokoll

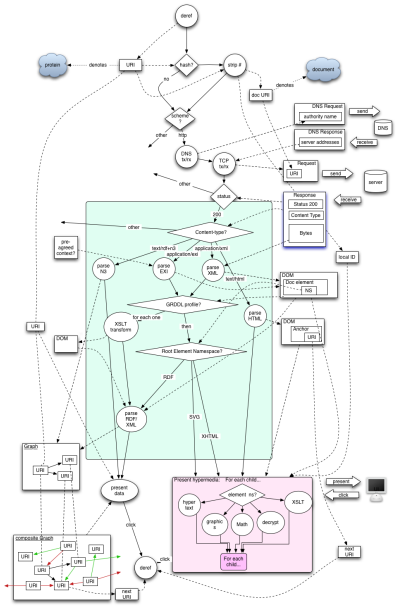
Das HTTP Protokoll ist laut [RFC2616]

- an application-level protocol for distributed, collaborative, hypermedia information systems
- a generic, stateless, protocol which can be used for many tasks beyond its use for hypertext
- a protocol which includes the typing and negotiation of data representation, allowing systems to be built independently of the data being transferred

HTTP Übersicht

- HTTP Nachrichten bestehen aus der Anfrage eines Clients an einen Server und die Antworten des Servers zum Client
- Bestimmte Methoden sind vordefiniert (z.B. GET, POST, etc.), aber weitere können definiert werden
- Eine Menge von Statuscodes ist definiert:
  - Informational 1xx, provisional response, (100 Continue)
  - Successful 2xx, request successfully received, understood, and accepted (201 Created)
  - Redirection 3xx, further action needs to be taken by user agent to fulfill the request (301 Moved Permanently)
  - Client Error 4xx, client erred (405 Method Not Allowed)
  - Server Error 5xx, server encountered an unexpected condition (501 Not Implemented)

HTTP Übersicht



1. Parse URI and find HTTP protocol
2. Look up DNS name to determine the associated IP address
3. Open a TCP stream to port 80 at the IP address determined above
4. Format an HTTP GET request for resource and send that to the server
5. Read response from the server
6. From status code (200) determine a successful request (representation of the resource is available)
7. Inspect the returned Content-Type (e.g., UTF-8 encoded text/html)
8. Pass the entity-body to the HTML rendering engine

HTTP GET Request

```
GET /todaysnews HTTP/1.1
Host: example.com
User-Agent: Mozilla/8.0
Accept: text/html,application/xhtml+xml;q=0.9,*/*
Accept-language: en-us
```

HTTP Response

```
HTTP/1.1 200 OK
Date: Tue, 28 Aug 2007 01:49:33 GMT
Server: Apache/2.2.11
Content-Type: text/html; charset=utf-8

<!DOCTYPE html PUBLIC
  "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
  xml:lang="en" lang="en">
<head><title>Today's news</title></head>
<body>
  <h1>Today's News: Oh boy!!</h1>
  [HTML FOR NEWS REPORT HERE]
</body>
</html>
```


Repräsentationen

- Informationsressourcen können unterschiedliche Repräsentationen haben.
- Eine Repräsentation ist ein Stream von Bytes in einem bestimmten Format wie z.B. HTML, RDF/XML oder JPEG.
- Beispiel: Eine Rechnung ist eine Informationsressource, die in HTML, als druckbares PDF oder als RDF Dokument repräsentiert werden kann.
- Eine einzelne Ressource kann viele verschiedene Repräsentationen haben z.B. in verschiedenen Formaten, Auflösungen oder Sprachen

HTTP Content Negotiation

- Content Negotiation (CN, conneg) ist der Prozess der Selektion der besten Repräsentation für eine Anfrage wenn mehrere Repräsentationen verfügbar sind
- Drei Arten: server-driven, agent-driven, transparent

```
$ curl -H "Accept: application/rdf+xml"
    http://dbpedia.org/resource/Galway

HTTP/1.1 303 See Other
Content-Type: application/rdf+xml
Location: http://dbpedia.org/data/Galway.rdf
$
```

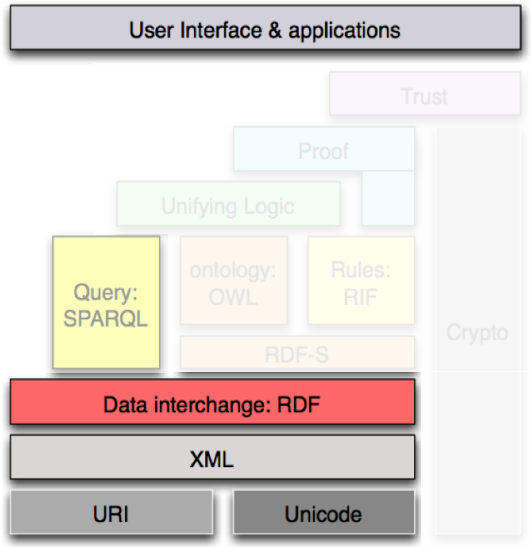
`curl` – Tool um Daten zu einerm Server zu schicken oder von einem Server zu empfangen
`-H` bedeutet nur HTTP/HTTPS

RDF als Linked Data

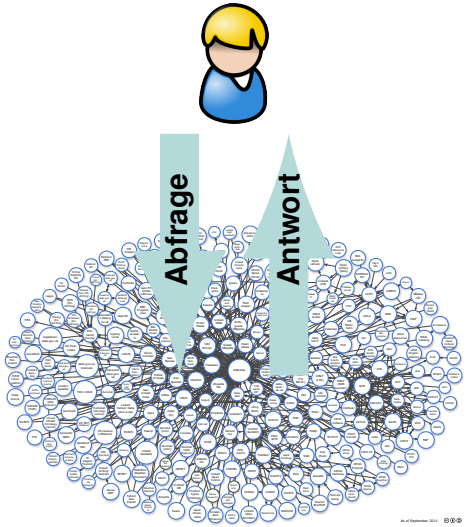
```
<?xml version="1.0"?>
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:foaf="http://xmlns.com/foaf/0.1/">
  <foaf:Person rdf:about="#ah">
    <foaf:name>Andreas Harth</foaf:name>
  </foaf:Person>
</rdf:RDF>
```

Datei veröffentlicht unter `http://harth.org/andreas/foaf.rdf`
URI bezeichnet Andreas: `http://harth.org/andreas/foaf.rdf#ah`

Semantic Web Application Architecture



Linked Data Anwendungen: Minimale Architektur



Beispiel: Visualisierung der Wahlergebnisse

- Daten von IT.NRW (Landesbetrieb Information und Technik Nordrhein-Westfalen) im CSV Format
- Schritt 1: Konvertierung nach RDF (mittels Google AppEngine* Wrapper oder Google Refine** mit RDF Extension***)
- Schritt 2: Linked Data Abfragen
- Schritt 3: Visualisierung der Ergebnisse

*`http://code.google.com/appengine/`
**`http://code.google.com/p/google-refine/`
***`http://lab.linkeddata.deri.ie/2010/grefine-rdf-extension/`
`http://gesis-lod.appspot.com/vis/`

Beispiel: Visualisierung der Ökonomischen Situation

- Daten von GESIS (Leibniz-Institut für Sozialwissenschaften) im CSV Format
- Schritt 1: Konvertierung nach RDF und publiziere die Daten online
- Schritt 2: Linked Data Abfragen
- Schritt 3: Visualisierung der Ergebnisse

`http://gesis-lod.appspot.com/vis/`

Beispiel: Visualisierung Eurostat Daten

- Daten von Eurostat (Statistisches Büro der EU) im CSV oder SDMX Format
- Schritt 1: Konvertierung nach RDF
- Schritt 2: Linked Data Abfragen
- Schritt 3: Visualisierung der Ergebnisse

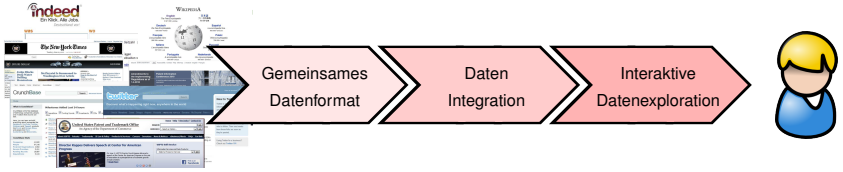
`http://estatwrap.ontologycentral.com/page/tsieb010`

Linked Data Services

- Einige Services erlauben nur eingeschränkten Zugriff auf Ihre Daten (z.B. APIs von sozialen Netzwerken)
- Manchmal wird mehr als ein Parameter benötigt (z.B. um den kürzesten Abstand zwischen zwei Punkten zu berechnen)
- Idealerweise sollte Linked Data derartige Service integrieren

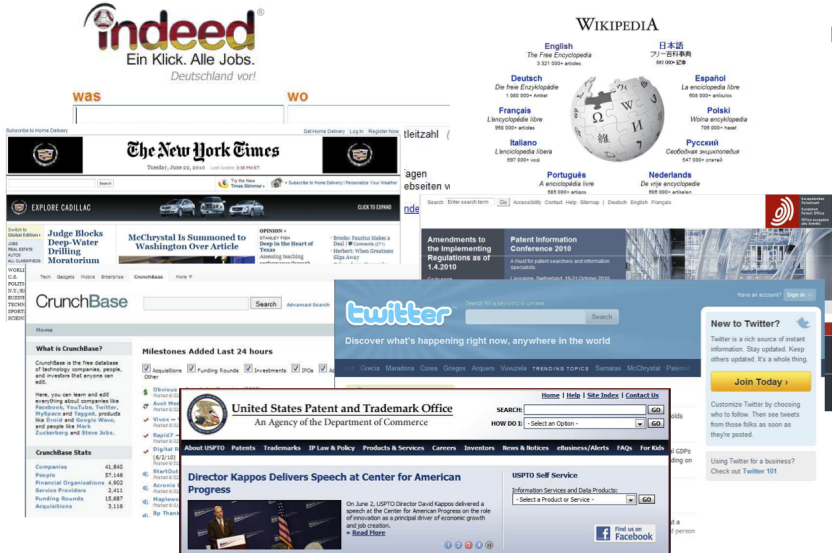
Szenario

- Typisches Datenintegrationsszenario

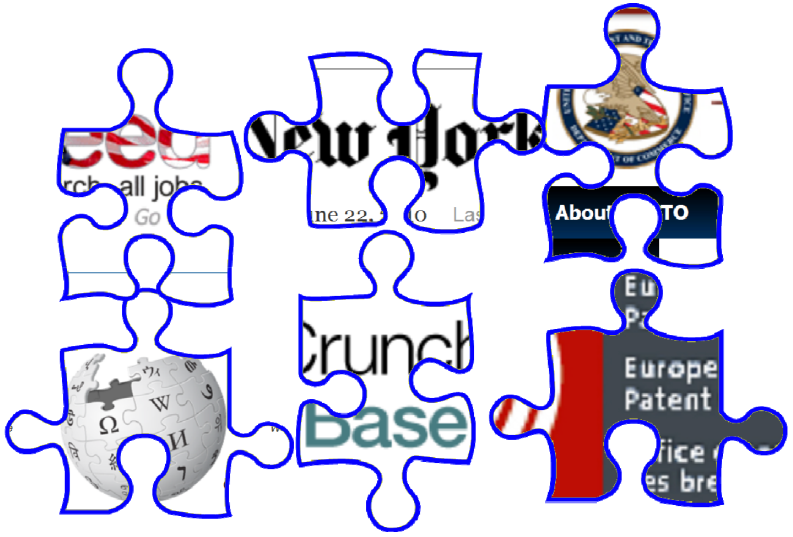


- Anfrage: Welche Jobangebote gibt es von Konkurrenten von Facebook?
- Anfrage: Nach welchem Muster vergibt Vulcan Capital Mittel?

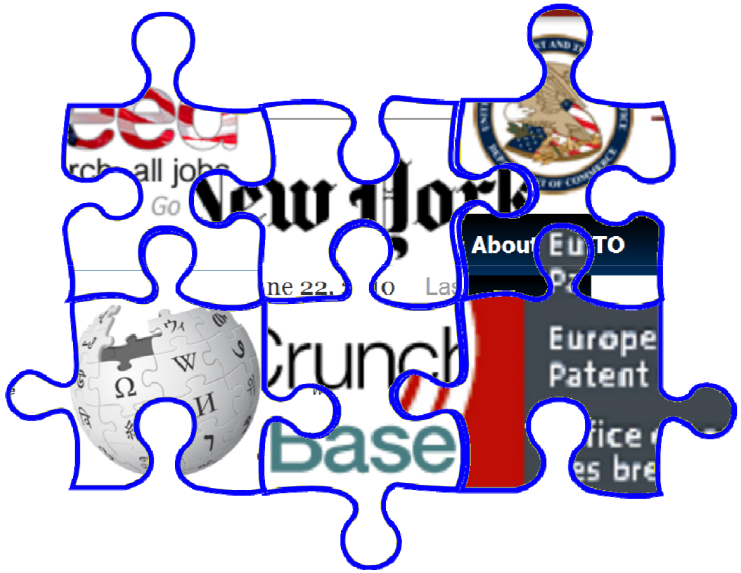
Datenqueallen



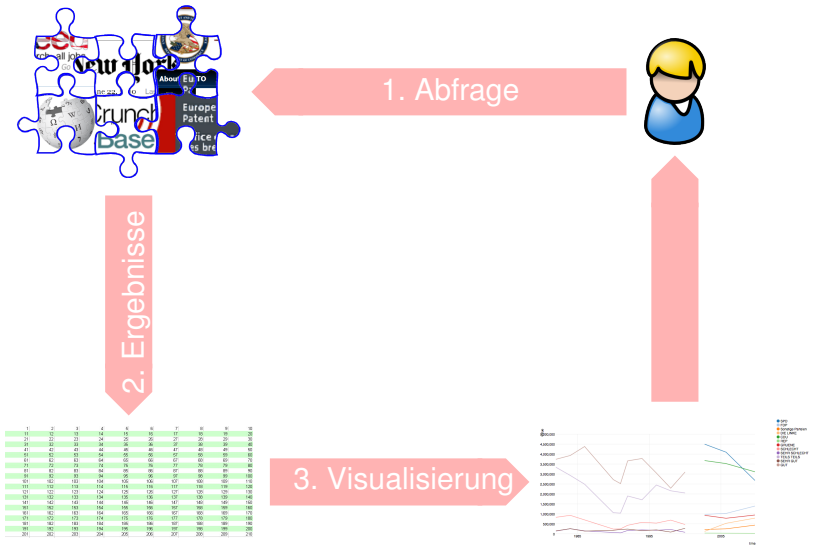
Schritt 1: Vorbereitung der Daten - Gemeinsames Datenformat



Schritt 2: Datenintegration



Schritt 3: Interaktive Datenexploration



Verlinkung von Daten mit Daten von Servicen?

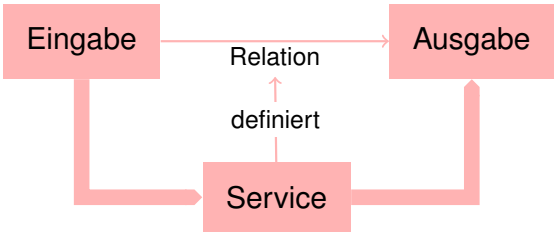
```
:facebook foaf:name "Facebook" .
:facebook cb:has_office :facebook-hp .
:facebook-hq geo:lat "37.416" .
:facebook-hq geo:long "122.152" .
:facebook-hq vc:locality "Palo Alto, CA" .
```

Gegeben der Firmenname und Ort, finde die Jobangebote

Gegeben die Koordinaten, finde nahegelegene Orte (via GeoNames)

Daten Service?

- Gegeben eine Eingabe, erzeuge die Ausgabe
- Eingabe und Ausgabe hängen in einer Service-spezifischen Art zusammen
- Weltzustand bleibt unverändert



- Beispiel: GeoNames `findNearbyWikipedia` Service
  - Eingabe: Latitude/Longitude Koordinaten
  - Ausgabe: Orte
  - Beziehung: Ausgabe Orte sind in der Nähe der Eingabekoordinaten

LIDS: Linked Data Services

- Wünschenswert wäre eine Integration von Daten Servicen mit Linked Data
  1. LIDS müssen sich an die Linked Data Prinzipien halten
- Wünschenswert wäre die Nutzung von Daten Servicen in Software Programmen
  1. LIDS brauchen maschinenlesbare Beschreibungen von Eingabe und Ausgabe

Data Services als Linked Data

- Eingabe ist gegeben als URI

Service Endpoint | Parameter | Eingabe Identifier

`http://geowrap.openlids.org/findNearbyWikipedia?lat=37.416&lng=-122.152#point`

- Auflösung der URI ergibt RDF:

```
@prefix dbp: <http://dbpedia.org/resource/> .
@prefix : <http://geo..Wiki?lat=37.416&lng=-122.152#> .
:point foaf:based_near dbp:Palo_Alto%2C_California ,
                       dbp:Packard%27s_garage .
```

Eingabe (`?lat=37.416&lng=-122.152`), Relation (`foaf:based_near`), Ausgabe (`dbp:Palo_Alto%2C_California`, `dbp:Packard%27s_garage`)

LIDS Beschreibungen mittels SPARQL

- Gegeben eine bestimmte Eingabe, kann die entsprechende Ausgabe von einer impliziten Datenquelle abgefragt werden
- Entspricht einer SPARQL `CONSTRUCT` Abfrage

```
CONSTRUCT { [output] } FROM [endpoint] WHERE { [input] }
```

- Eingabe beschreibt die benötigten Daten als Abfragemuster
- Der Endpunkt ist die base URI um eine Service Eingabe zu konstruieren
- Ausgabe beschreibt die Daten, die der Service geliefert hat

```
CONSTRUCT { ?point foaf:based_near ?feature. }
  FROM <http:/geowrap.openlids.org/findNearbyWikipedia>
  WHERE { ?point a Point ; geo:lat ?lat ; geo:long ?lng }
```

LIDS Zusammenfassung

- Dynamische Quellen (GeoNames Wrapper, Twitter Wrapper, Feeds Wrapper) können in Linked Data integriert werden
- LIDS nützlich für
  - Einfügen von LIDS in statische RDF Datensätze
  - Linked Data Endpunkte, die dynamisch Links von ihren Daten zu LIDS hinzufügen
  - Linked Data Browser, die abgefragte Daten um abgefragte Daten von LIDS erweitern
  - Integration von LIDS in die SPARQL Abfrageverarbeitung
- LIDS erlauben das Publizieren und Wiederbenutzen von Datenservices im Web

Zusammenfassung

- Die Menge an verfügbaren Daten wächst ständig
- Semantik wird gebraucht, um Daten aus verschiedenen Quellen zu integrieren
- Abfrage und Visualisierung von Daten in Kombination möglich

- Die Verarbeitung und Abfrage von Daten aus verschiedenen Quellen erhöht die Transparenz und erleichtert die Forschung (Testen von Hypothesen wird einfacher)