Knowledge Engineering with Semantic Web Technologies

Lecture 6: Knowledge Engineering

01: Ontological Engineering

Dr. Harald Sack
Hasso-Plattner-Institut for IT Systems Engineering
University of Potsdam
Spring 2014

Lecture 6: Knowledge Engineering

Open HPI Course: Knowledge Engineering with Semantic Web Technologies

01 Ontological Engineering

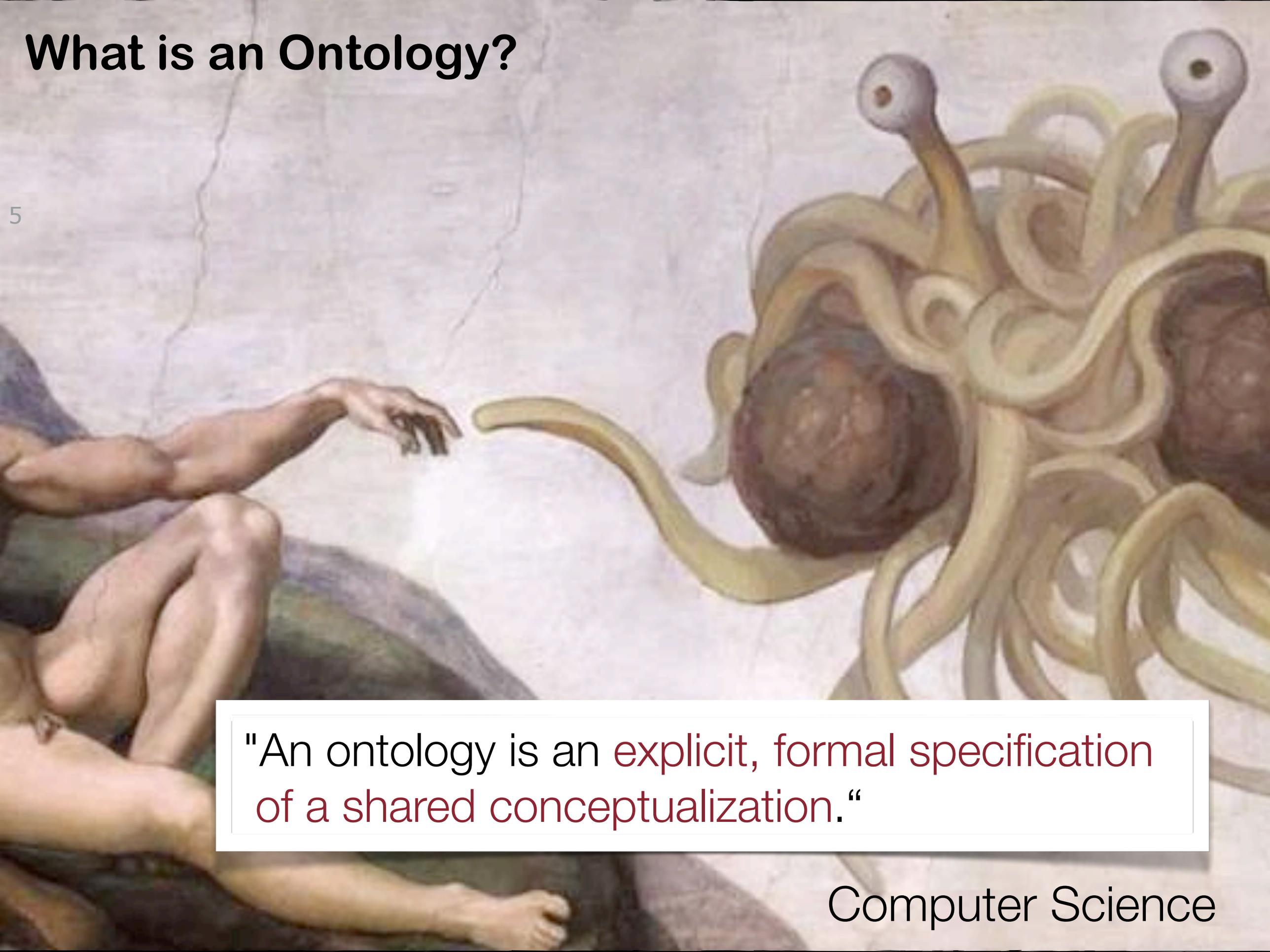
Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

What is an Ontology?

„A theory of being, which tries to explain the being itself, by developing a system of universal categories and their intrinsic relationships...“

Philosophy

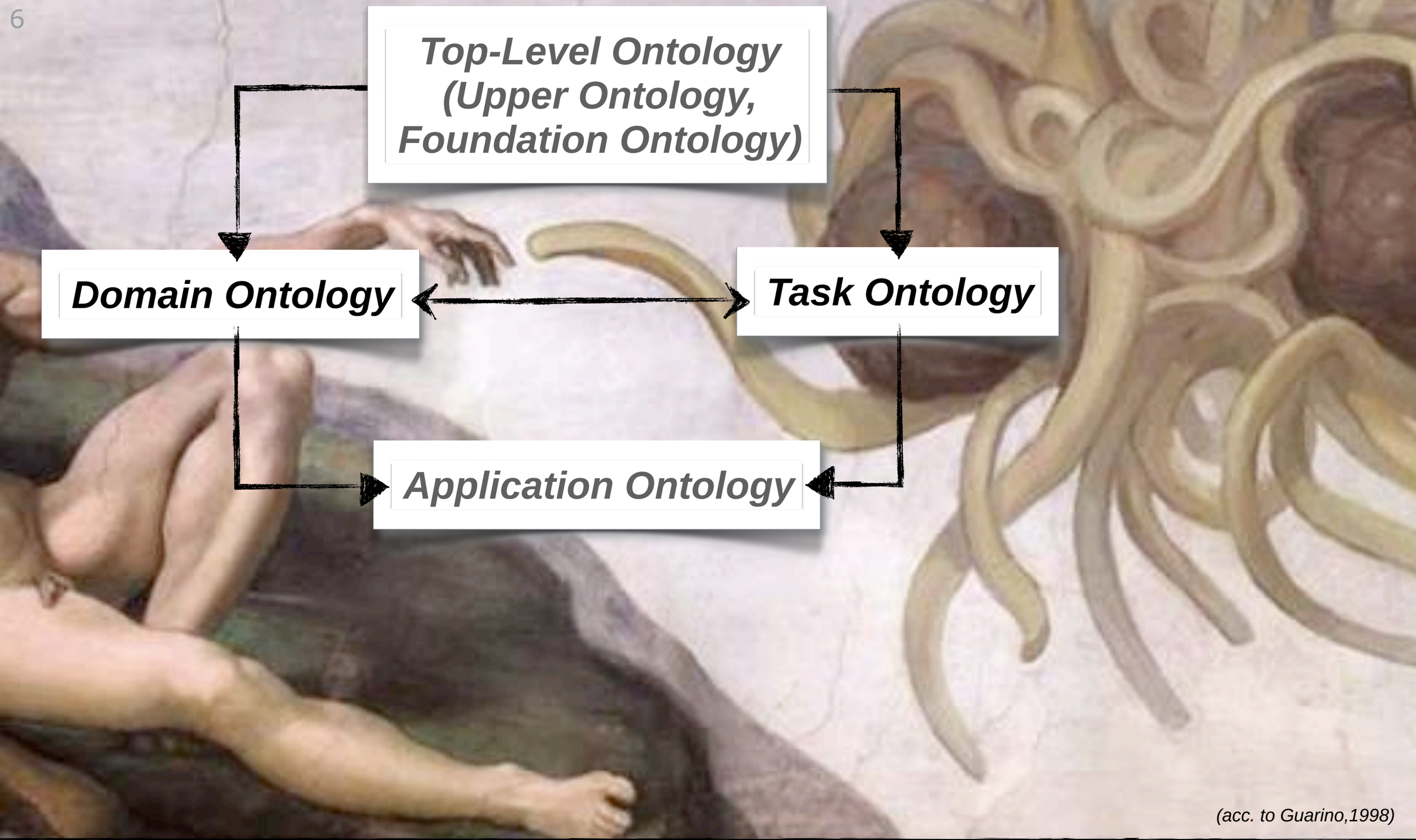
What is an Ontology?



"An ontology is an explicit, formal specification of a shared conceptualization.“

Computer Science

What is an Ontology?

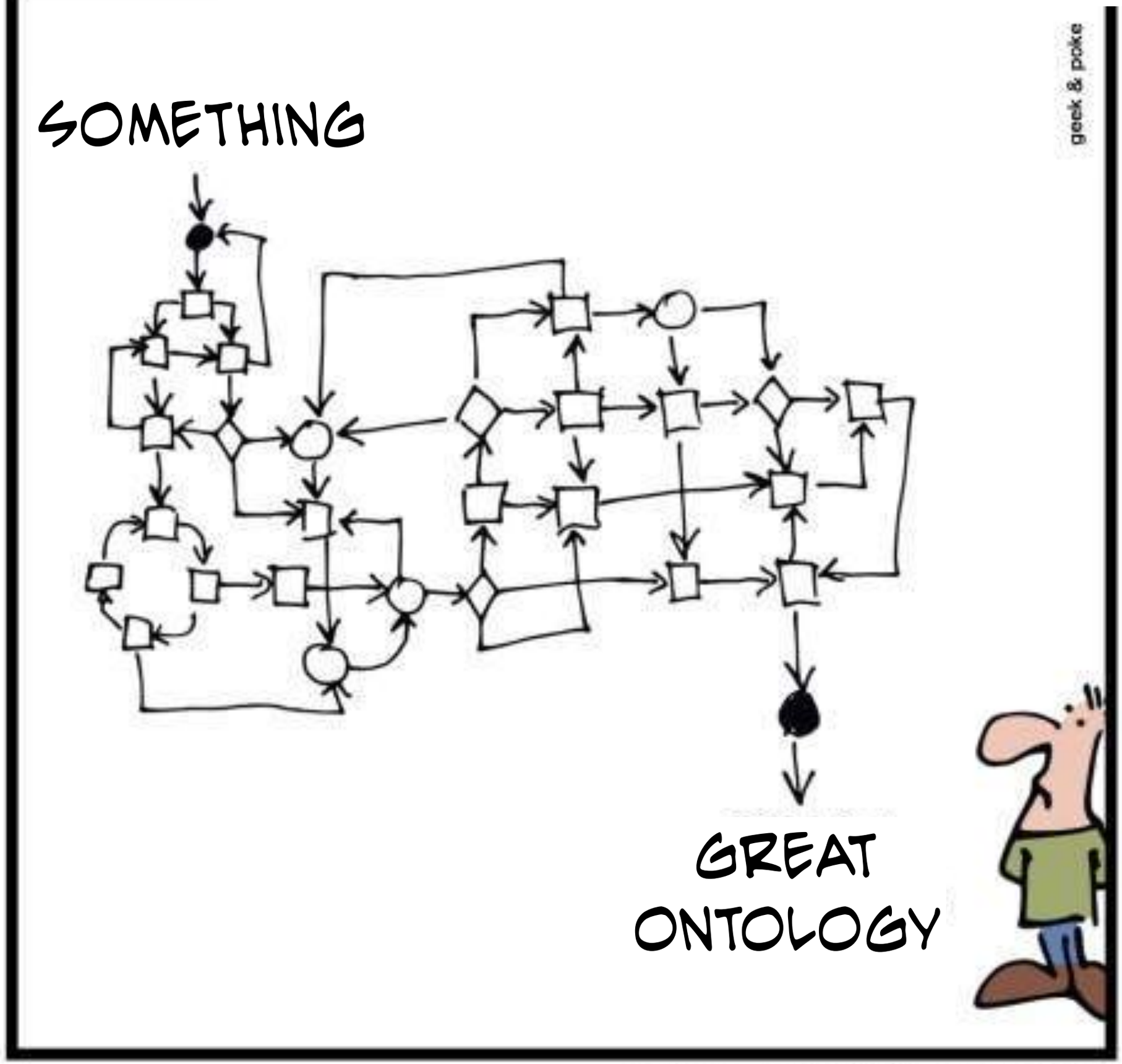


(acc. to Guarino,1998)

Ontological Engineering

- Ontologies enable **interoperability** among metadata
- Therefore, we need
  - Methods for efficient development of ontologies (**Ontology Design**)
  - Methods for efficient comparison of ontologies (**Ontology Mapping**)
  - Methods for efficient combination of ontologies (**Ontology Merging**)
- There are automated methods to support Ontological Engineering:
  - Learning new ontologies from a given set of information resources (**Ontology Learning**)
  - Populating existing ontologies with individuals from information resources

ONTOLOGY DEVELOPMENT PROCESS



acc. to http://geekandpoke.typepad.com/geekandpoke/2012/01/simply-explained-dp.html

Methodologies of Ontology Design

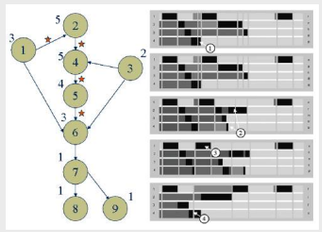
- *A methodology of Ontology Design describes all activities necessary for the construction of an ontology.*

- Why do we need a formal methodology?
  - development of **consistent** ontologies
  - **efficient** development of complex ontologies
  - **distributed** development of ontologies

- We distinguish (acc. to Fernandez-Lopez et. al., 1997)
  - Ontology management activities
  - Ontology development oriented activities
  - Ontology support activities

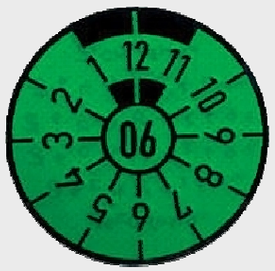
Ontological Engineering

Management



scheduling

control

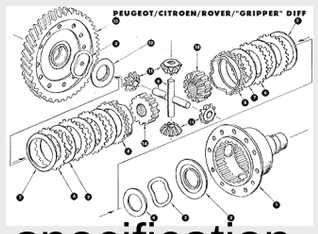


quality assurance

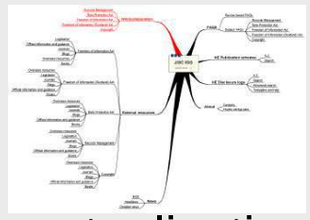
Development Oriented

environment study

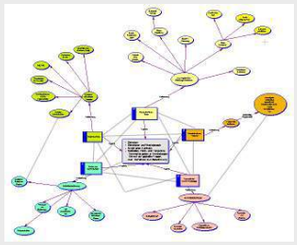
feasibility study



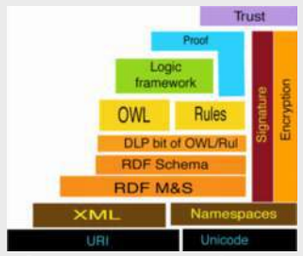
specification



conceptualization



formalization



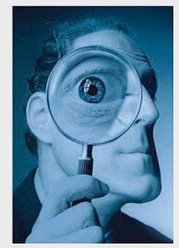
implementation

maintenance

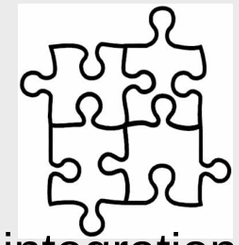
use / reuse

Support

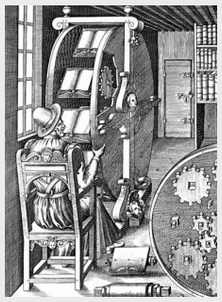
knowledge acquisition



evaluation



integration



documentation

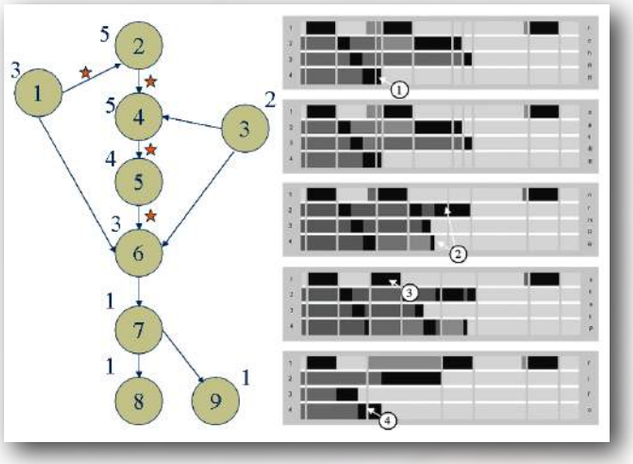
merging

configuration management

alignment

acc. to Fernandez-Lopez et. al., 1997

Ontology Management Activities



- **Scheduling**
  - Identification of tasks/problems to solve
  - Arrangement/planning of tasks/problems to solve
  - Identification of required resources (time, memory, etc…)

- **Control**
  - Guaranteeing correct execution of tasks/problems to solve

- **Quality Assurance**
  - Quality assurance of all produced resources during development (ontologies, software, documentation)

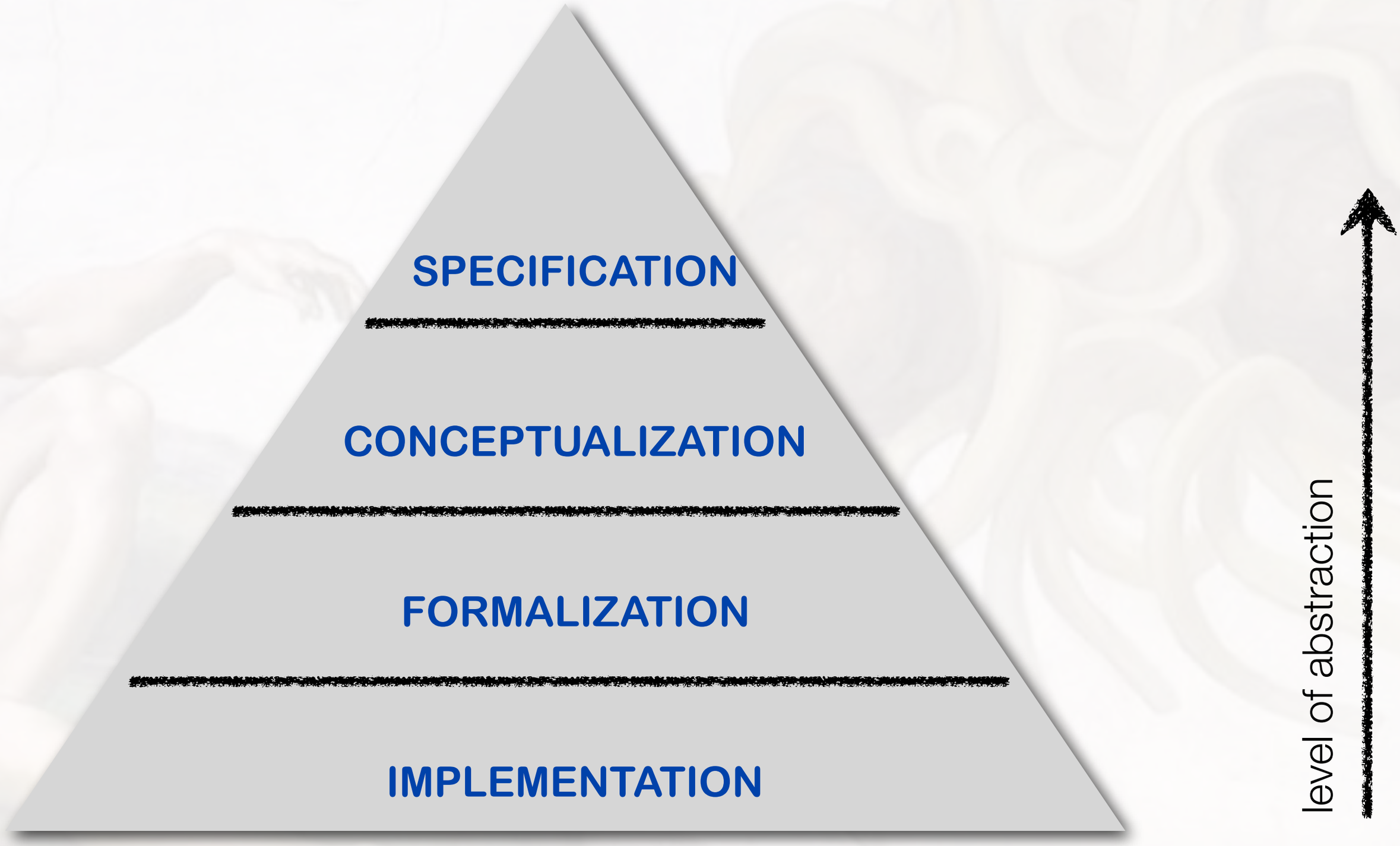
Ontology Development Oriented Activities

1. Pre-Development
2. Development
3. Post-Development

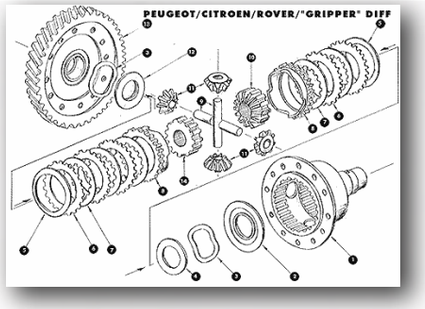
1.Pre-Development

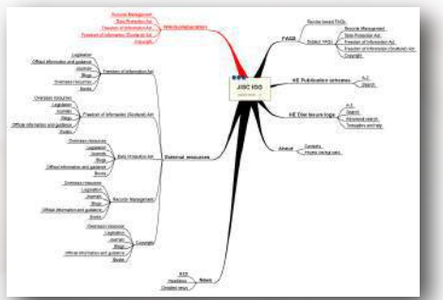
- **Environment Study**
  - What is the designated software platform for the ontology?
  - Which applications should use the ontology?
- **Feasibility Study**
  - Can the ontology really be developed?
  - Does it make sense to develop the ontology?

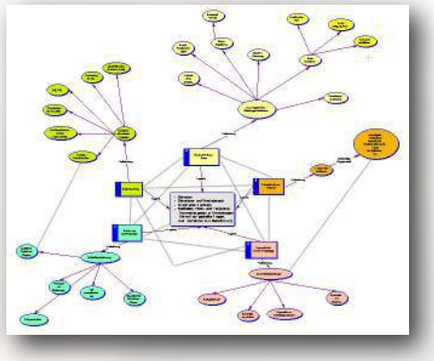
Ontology Development Oriented Activities

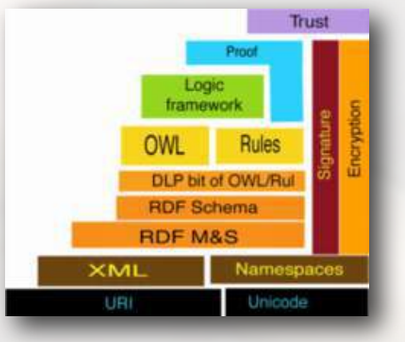


Ontology Development Oriented Activities









2.Development

- **Specification**
  - Why is the ontology developed, what is the benefit and who are the end-users?
- **Conceptualization**
  - Structuring domain knowledge in a conceptual model
- **Formalization**
  - Formalize conceptual model in (semi-)computable model
- **Implementation**
  - Construction of a computable model in an ontology representation language

Ontology Development Oriented Activities

3.Post-Development

- **Maintenance**
  - Update and adjustment of the ontology (if necessary)

- **Use / Reuse**
  - Usage of the ontology within the designated applications as well as in unplanned applications/ontologies

Ontology Support Activities

- **Knowledge Acquisition**
  - Gather knowledge from experts (Ontology Learning)

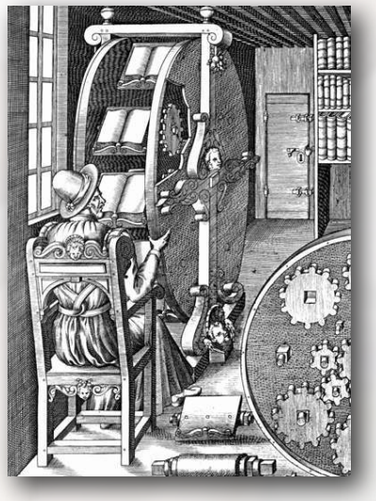
- **Evaluation**
  - Technical evaluation of the ontology in each step of the development process

- **Integration**
  - Reuse of existing ontologies (Ontology Reuse)

- **Merging**
  - Construction of a new ontology from already existing ontologies within a specific domain

Ontology Support Activities

- **Alignment**
  - Design/apply mapping rules for involved ontologies

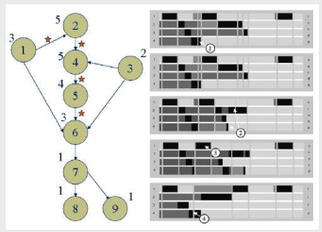


- **Documentation**
  - Each step of the ontology development must be accurately documented

- **Configuration Management**
  - Manages all versions of documentation and of the developed ontology

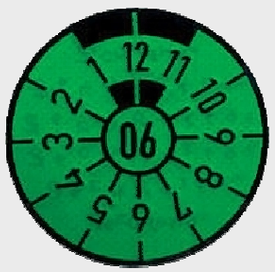
Ontological Engineering

Management



scheduling

control

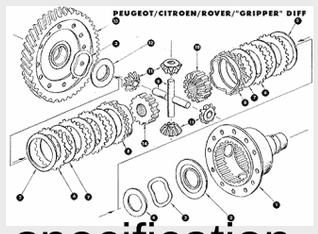


quality assurance

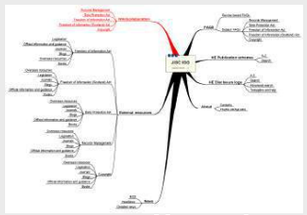
Development Oriented

environment study

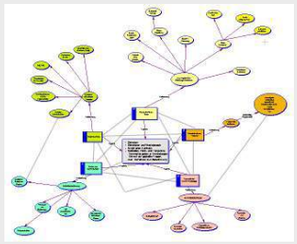
feasibility study



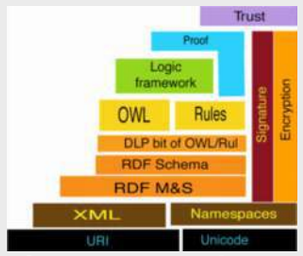
specification



conceptualization



formalization



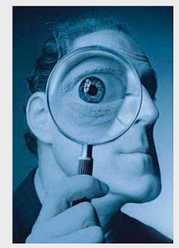
implementation

maintenance

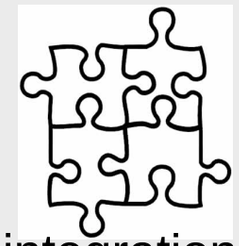
use / reuse

Support

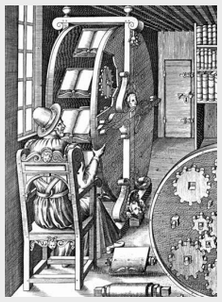
knowledge acquisition



evaluation



integration



documentation

merging

configuration management

alignment

acc. to Fernandez-Lopez et. al., 1997

02 - Ontology Design Methodologies

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

L'EXPOSITION UNIVERSELLE DE 1889. — ÉTAT DES TRAVAUX DU CHAMP DE MARS. — VUE PRISE DU PALAIS DU TROCADÉRO

Knowledge Engineering with Semantic Web Technologies

Lecture 6: Knowledge Engineering

02: Ontology Design Methodologies

Dr. Harald Sack
Hasso-Plattner-Institut for IT Systems Engineering
University of Potsdam
Spring 2014

Lecture 6: Knowledge Engineering

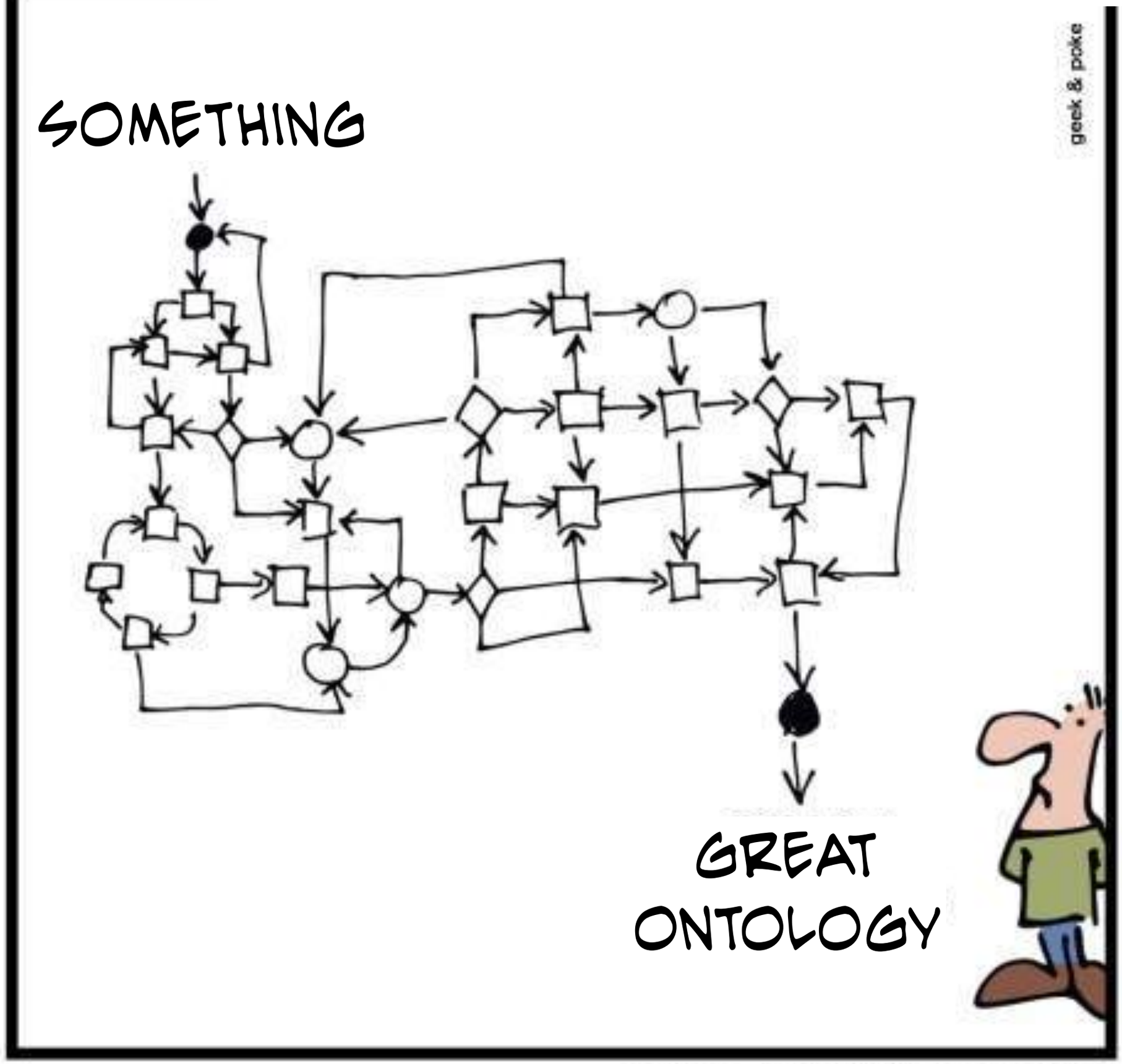
Open HPI Course: Knowledge Engineering with Semantic Web Technologies

02 - Ontology Design Methodologies

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

L'EXPOSITION UNIVERSELLE DE 1889. — ÉTAT DES TRAVAUX DU CHAMP DE MARS. — VUE PRISE DU PALAIS DU TROCADÉRO

ONTOLOGY DEVELOPMENT PROCESS



acc. to http://geekandpoke.typepad.com/geekandpoke/2012/01/simply-explained-dp.html

Unified Process for Ontology Building

De Nicola, Missikoff, Navigli (2005)

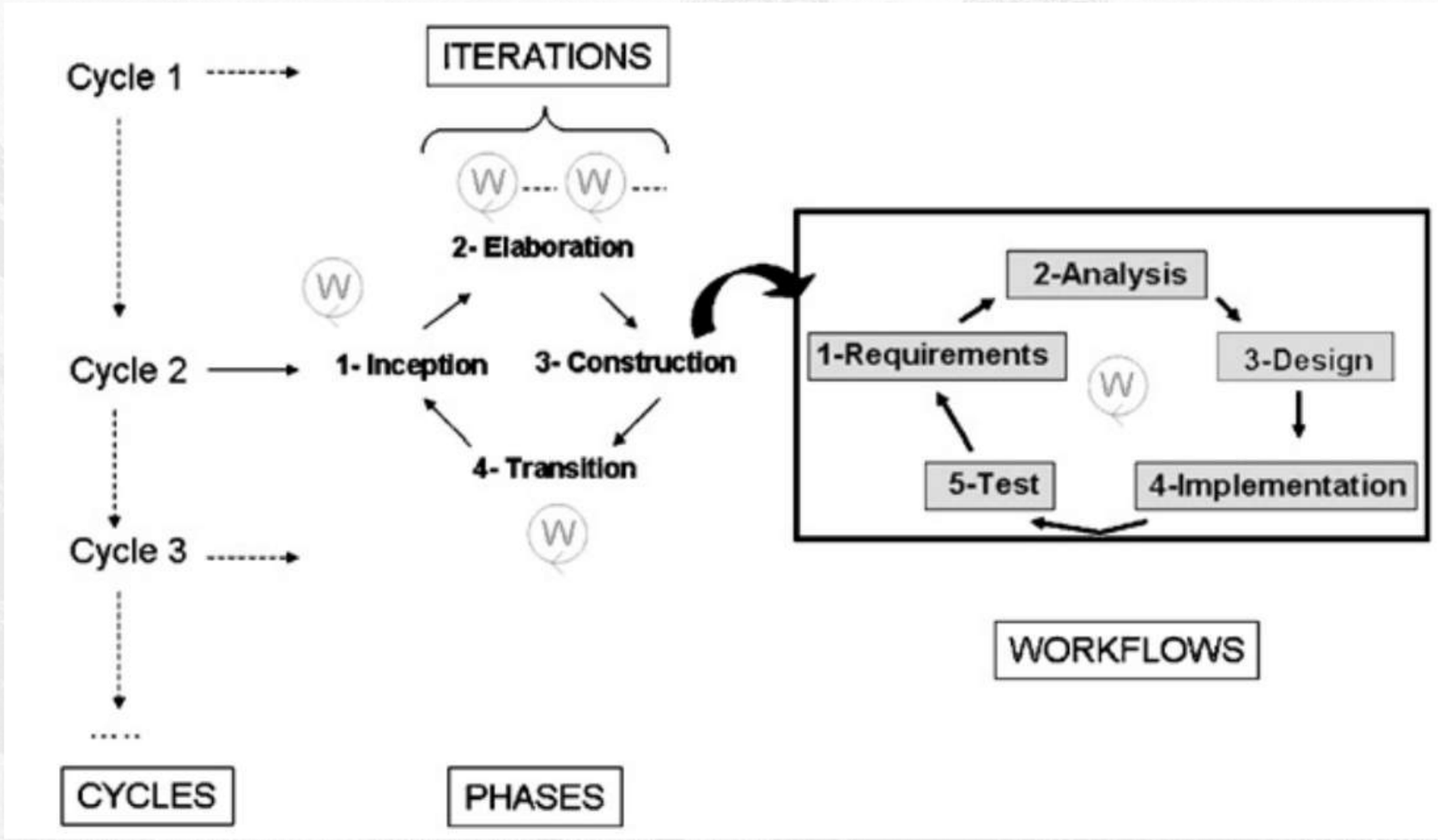
- Based on **Unified Process** (UP) methodology in software development and **Unified Modelling Language** (UML)
- Use-Case driven, i.e. more suitable for **application ontologies** than for domain ontologies

- **Goals**:
  - **Reduction of time and cost** in the development of large scale ontologies
  - **Quality improvement** of the developed ontology via progressive validation of intermediate results
  - Methodology for **efficient collaboration** of Knowledge Engineers and Domain Experts with clear separation of roles
  - Intermediate results can already be evaluated by the user

Unified Process for Ontology Building

De Nicola, Missikoff, Navigli (2005)

- Development is divided into **Cycles**, which are subdivided into 4 **Phases** of **Iterations** (*Inception, Elaboration, Construction, Transition*). Each iteration results in a new prototype

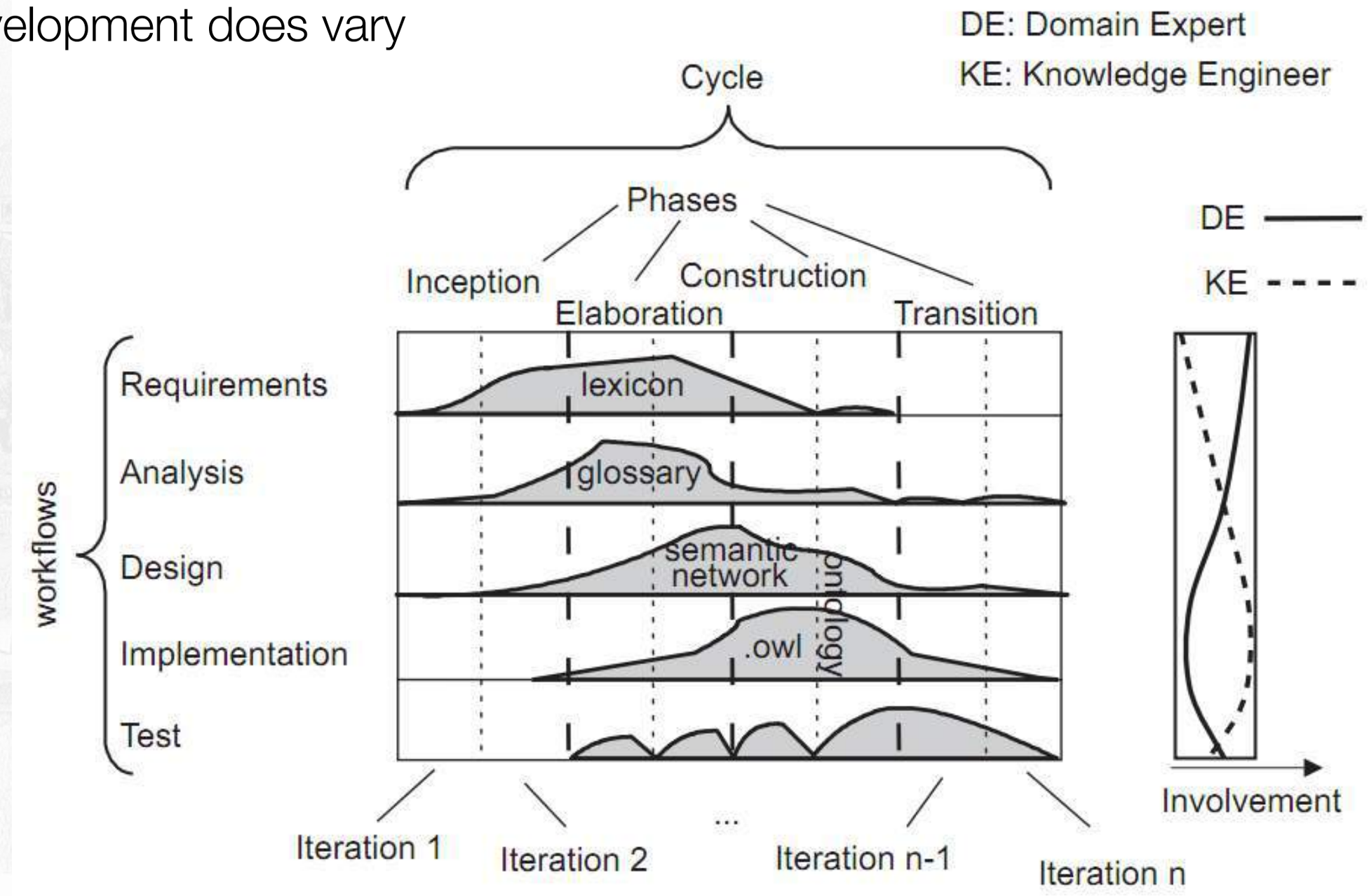


- Each iteration consists of 5 workflows (*Requirements, Analysis, Design, Implementation, Test*)

Unified Process for Ontology Building

De Nicola, Missikoff, Navigli (2005)

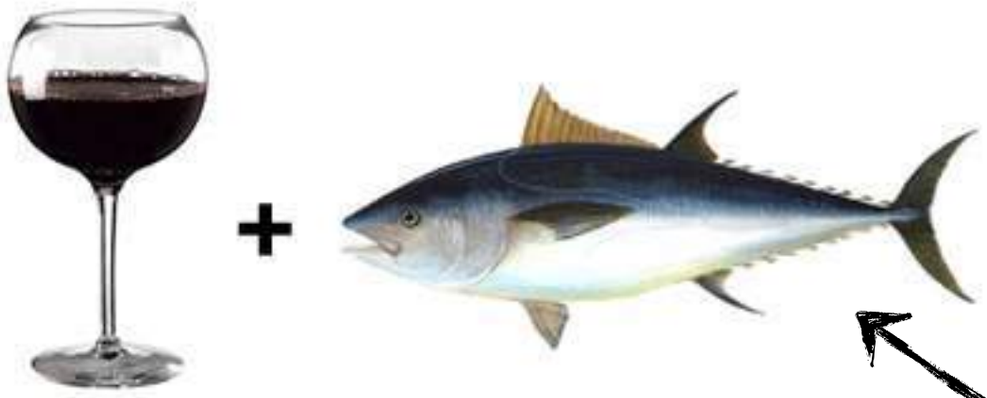
- Workflows and Phases are almost orthogonal, i.e. involvement of single workflows in different phases of ontology development does vary



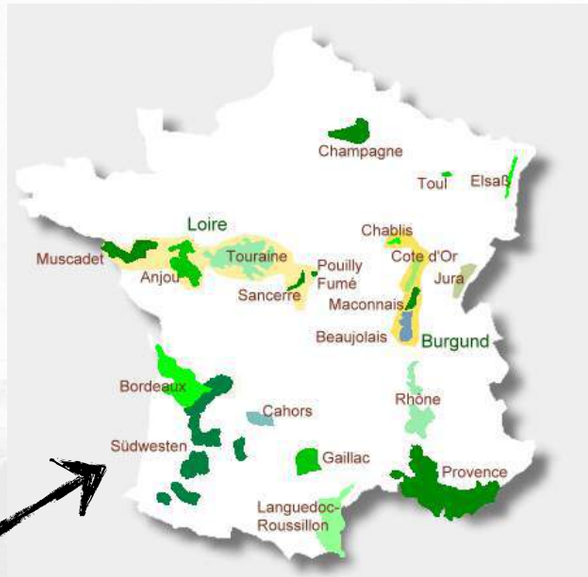
Ontology Development 101

(Noy, McGuinness, 2000)

- Example of a wine and food ontology

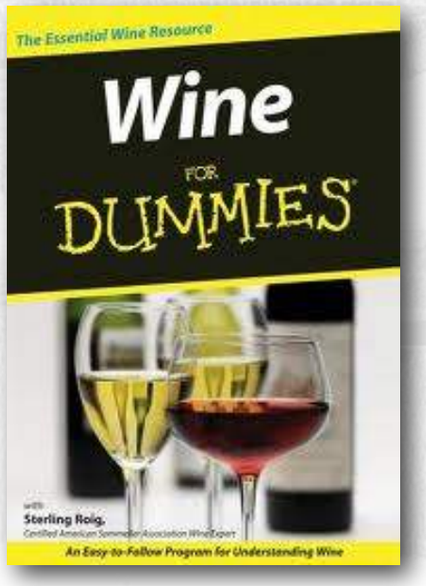


Which wine is the right one for fish?



French wine-growing regions and wines

A shared ontology on wine and food

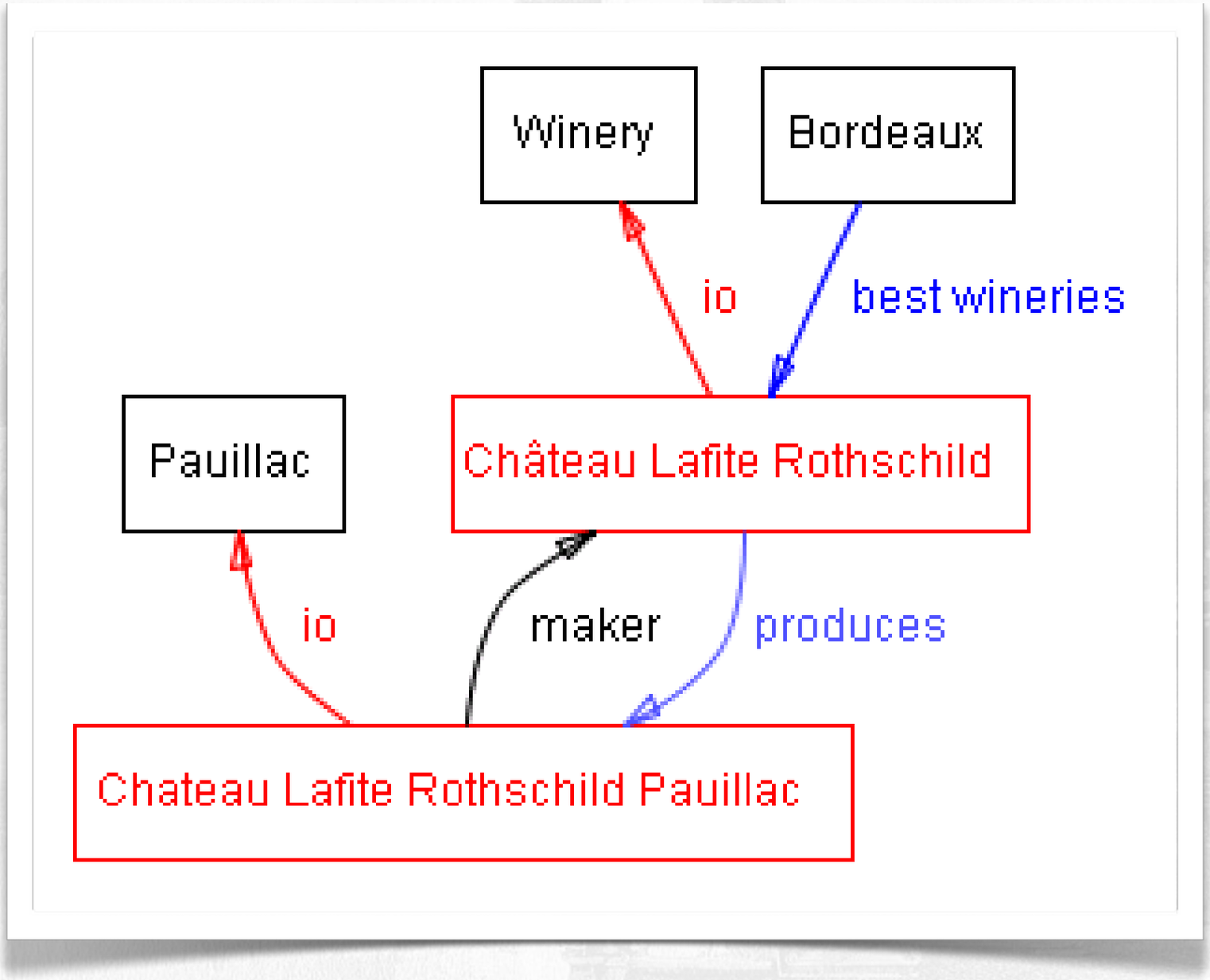


Californian wine-growing regions and wines

Ontology Development 101

(Noy, McGuinness, 2000)

- Example of a wine and food ontology



Ontology Development 101

(Noy, McGuinness, 2000)

- in practice an **iterative Process** that **repeats continuously** and improves the ontology
- there are always **different approaches** for modelling an ontology
- in practice the designated application decides about the modelling approach

„There is no one correct way to model a domain.
There are always viable alternatives."

Determine Domain and Focus

- Which **domain** should be covered by the ontology?
- **What** should the ontology be used **for**?
- What types of **Questions** should be answered by the knowledge represented in the ontology?
- **Who** will use and maintain the ontology?
- Formulation of **Competence Questions**

These Questions might change within the ontology life cycle

Determine Domain and Focus

Competence Questions *(Wine Ontology)*

- *Which properties of the wine should be considered for modelling?*
- *Is Bordeaux a white wine or a red wine?*
- *Does a Sauvignon Blanc match with fish?*
- *Which wine matches best for grilled vegetables?*
- *Which properties of a wine do influence whether it matches with a specific dish?*
- *Does the bouquet of a wine change with different vintages?*
- *Does the color of the wine depend on the color of the grape?*
- *...*

Consider Reuse

- Why should we consider reuse?
  - in order to save **cost**
  - in order to apply **tools** that are applied for other existing ontologies also for our own ontology
  - in order to reuse ontologies that have been validated by their application

If you don't find a suitable ontology or if the adaption is too complex then create a new ontology!

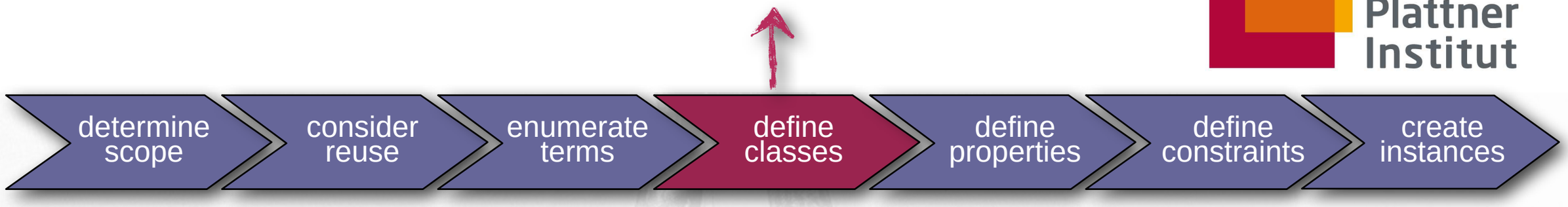
Develop a Terminology

- About which **concepts** are we talking?
- Which **properties** have these concepts?
- **What** do we want **to say** about these concepts?

*Example: **Wine Ontology***

- *wine, grape, winery, location,...*
- *a wine‘s color, body, flavor, sugar content,...*
- *subtypes of wine: white wine, red wine, Bordeaux wine,...*
- *types of food: seafood, fish, meat, vegetables, cheese,...*
- *...*

Develop Classes and Class Hierarchies



- **Classes** are concepts in the designated domain
  - *class of wines*
  - *class of wineries*
  - *class of red wines*
  - *...*
- Classes are collections of objects with **similar properties**
- Choose a **top-down** / **bottom-up** / **middle-out** approach to model class hierarchies

Define Properties

- **Properties** in a class definition describe attributes of instances
  - *every wine has a color, residual sugar, producer, etc...*

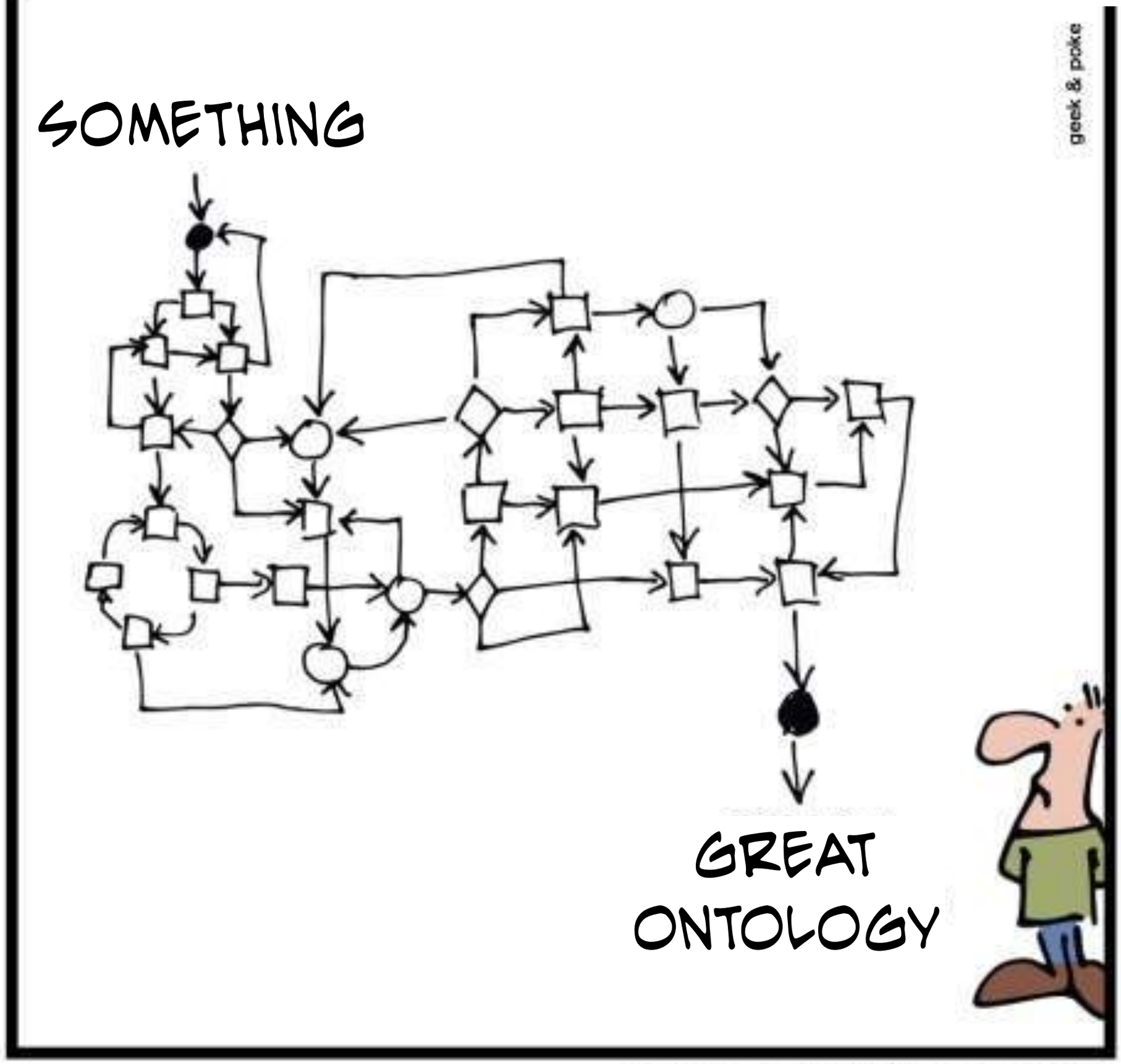
Define Property Constraints

- **Property constraints** (restrictions) describe or restrict the set of possible property values
  - *The name of a wine is a String*
  - *The producer is an instance of Winemaker...*

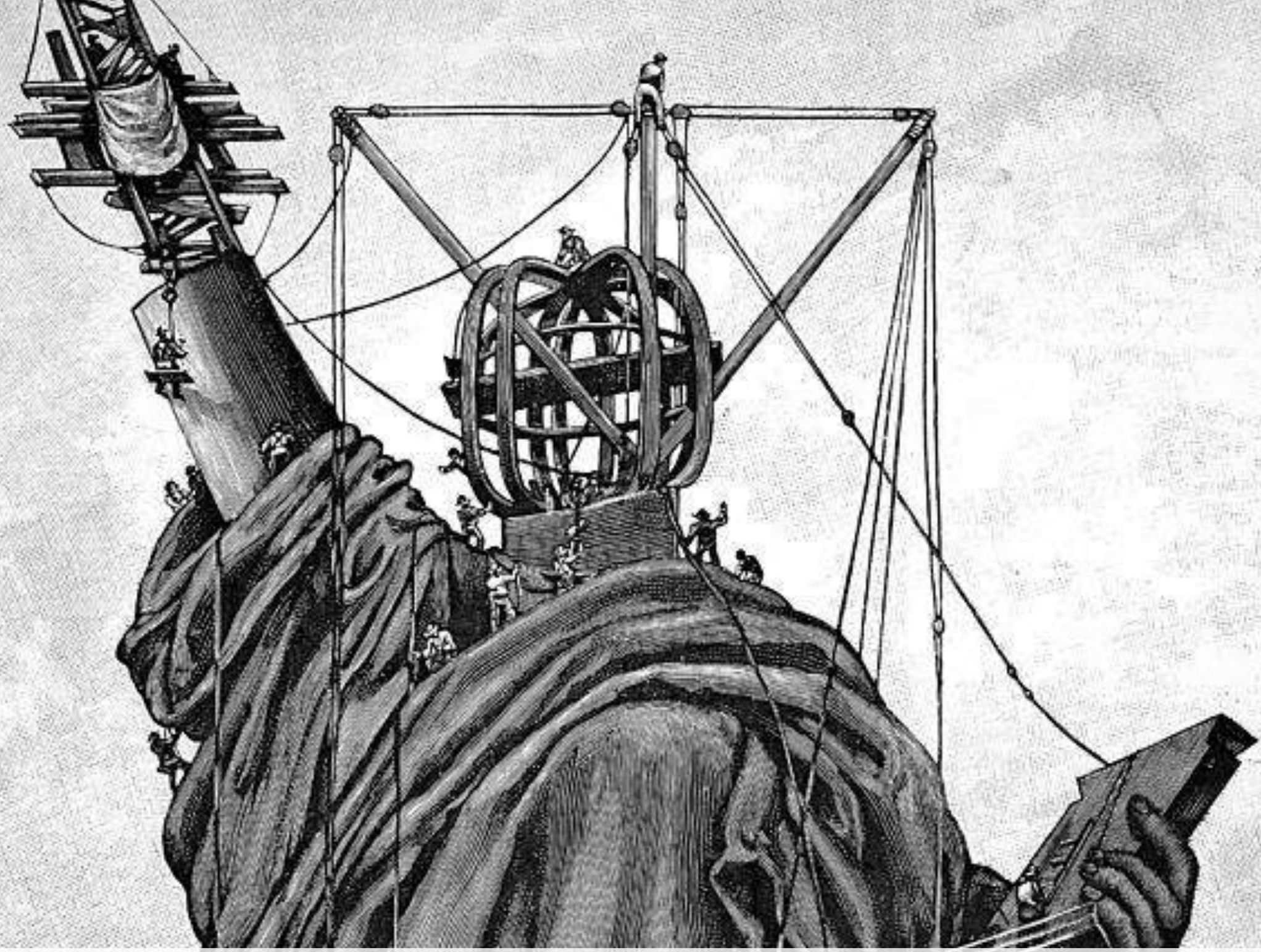
Definition of Instances

- Create **instances for the classes**
  - Every **class** directly becomes the **type** of its instances
  - Every **superclass** of a direct type is also type of its instances
- Create **instances for properties**, i.e. assignment of property values for the instances according to the given constraints
- *„the glass of red wine that I drank last supper...“*

ONTOLOGY DEVELOPMENT PROCESS



acc. to http://geekandpoke.typepad.com/geekandpoke/2012/01/simply-explained-dp.html



03 - Ontology Learning

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

Knowledge Engineering with Semantic Web Technologies

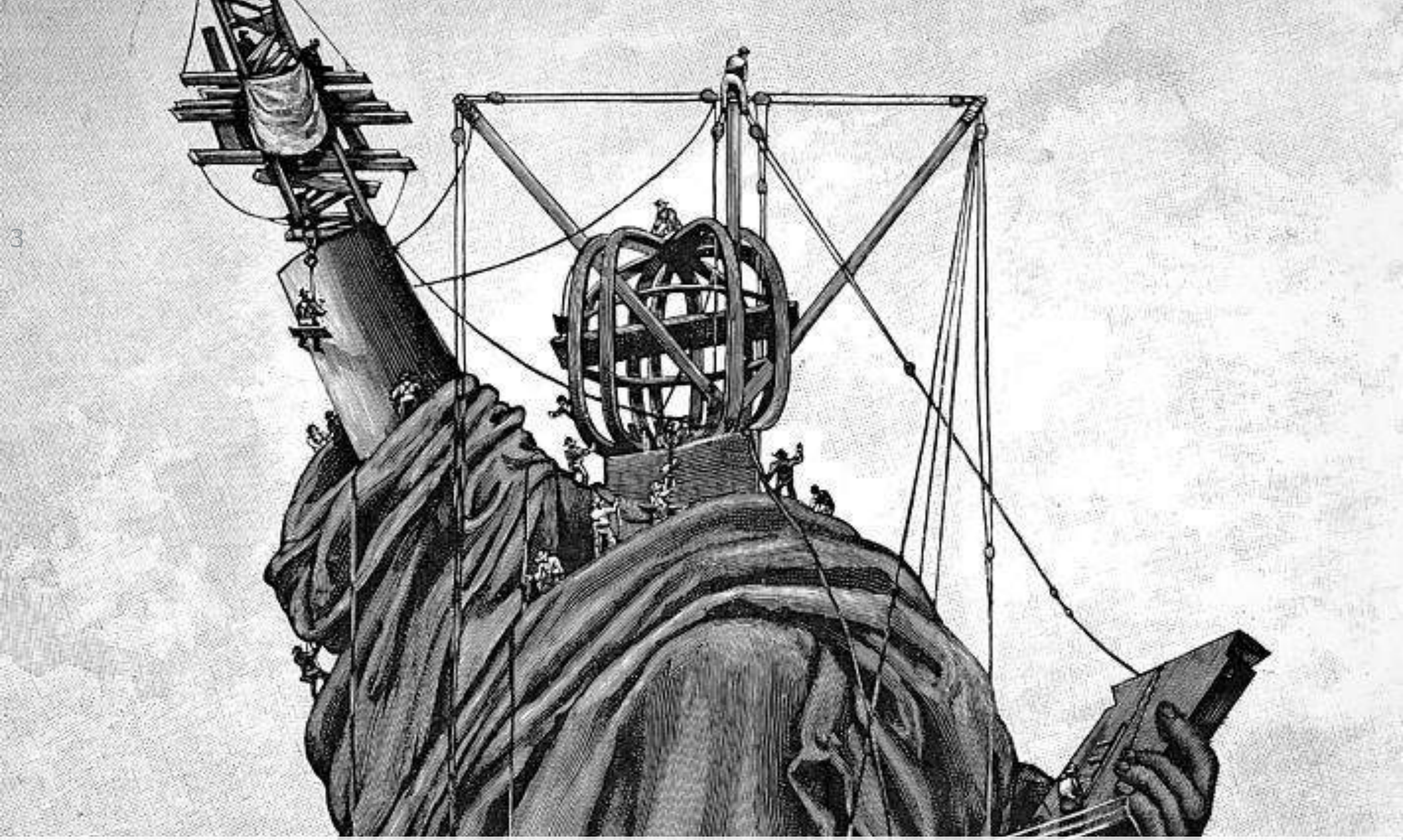
Lecture 6: Knowledge Engineering

03: Ontology Learning

Dr. Harald Sack
Hasso-Plattner-Institut for IT Systems Engineering
University of Potsdam
Spring 2014

Lecture 6: Knowledge Engineering

Open HPI Course: Knowledge Engineering with Semantic Web Technologies



03 - Ontology Learning

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

Ontology Learning

- Ontology Design is very expensive wrt. time and resources
  - *can we automate the process or at least some parts?*

- Ontologies can be „learned“ automatically

- **Ontology Learning** defines a set of methods and techniques
  - for **fundamental development** of new ontologies
  - for **extension or adaption** of already existing ontologies
- in a (partly) automated way from various resources.

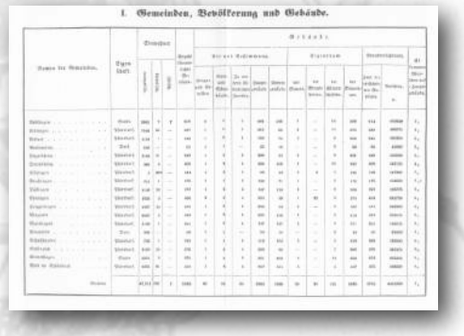
- also referred to as *Ontology Generation, Ontology Mining, or Ontology Extraction*

- Automatization requires help from
  - Natural Language Processing (NLP)
  - Data Mining
  - Machine Learning techniques (ML)

Ontology Learning

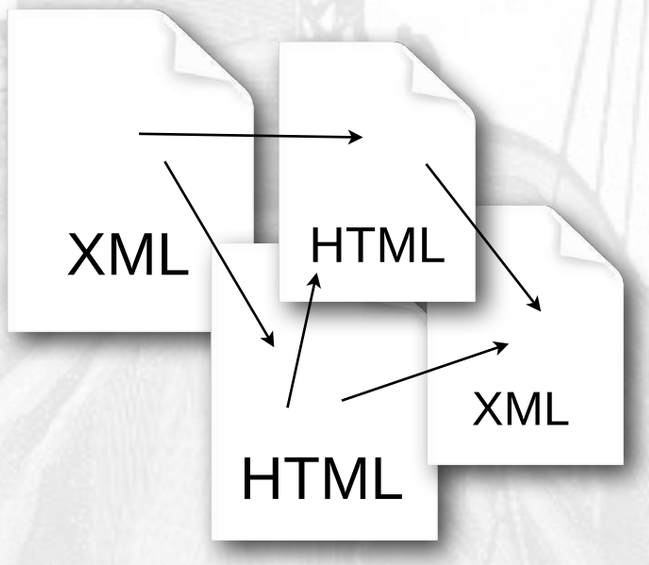
- Fundamental **(input) data** for ontology learning:

- **Structured Data**



Machine Learning

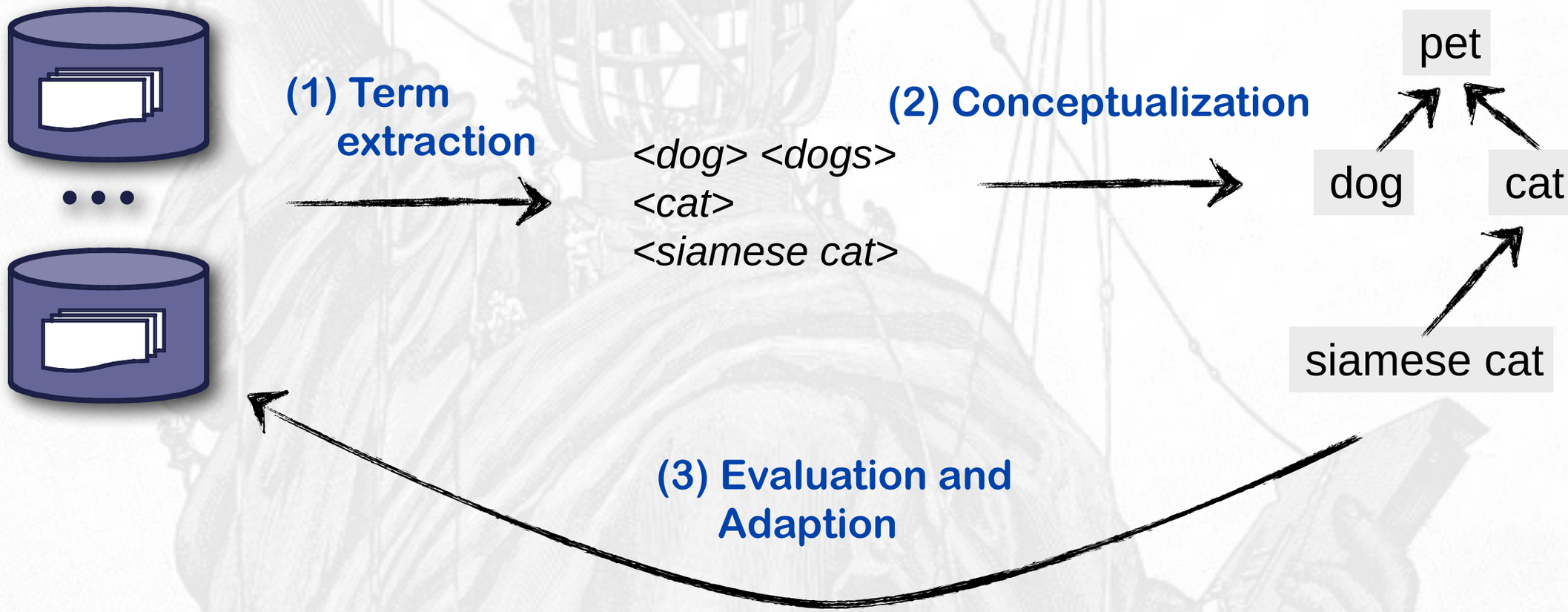
- **Semi-structured Data**



Natural Language Processing + Machine Learning

- **Unstructured Data**

Ontology Learning - Basic Approach (I)



***Term extractions**: requires linguistic processing (NLP) to identify important noun phrases and their internal semantic structure*

***Terms**: linguistic realizations of domain specific concepts*

***Concepts**: clusters of semantically related terms*

Ontology Learning - Basic Approach (II)

- **Natural Language Processing:**

(1) Tokenizer / Sentence Splitter

(2) Morphological Analysis

- Stemming
- Lemmatizer

(3) POS-Tagger

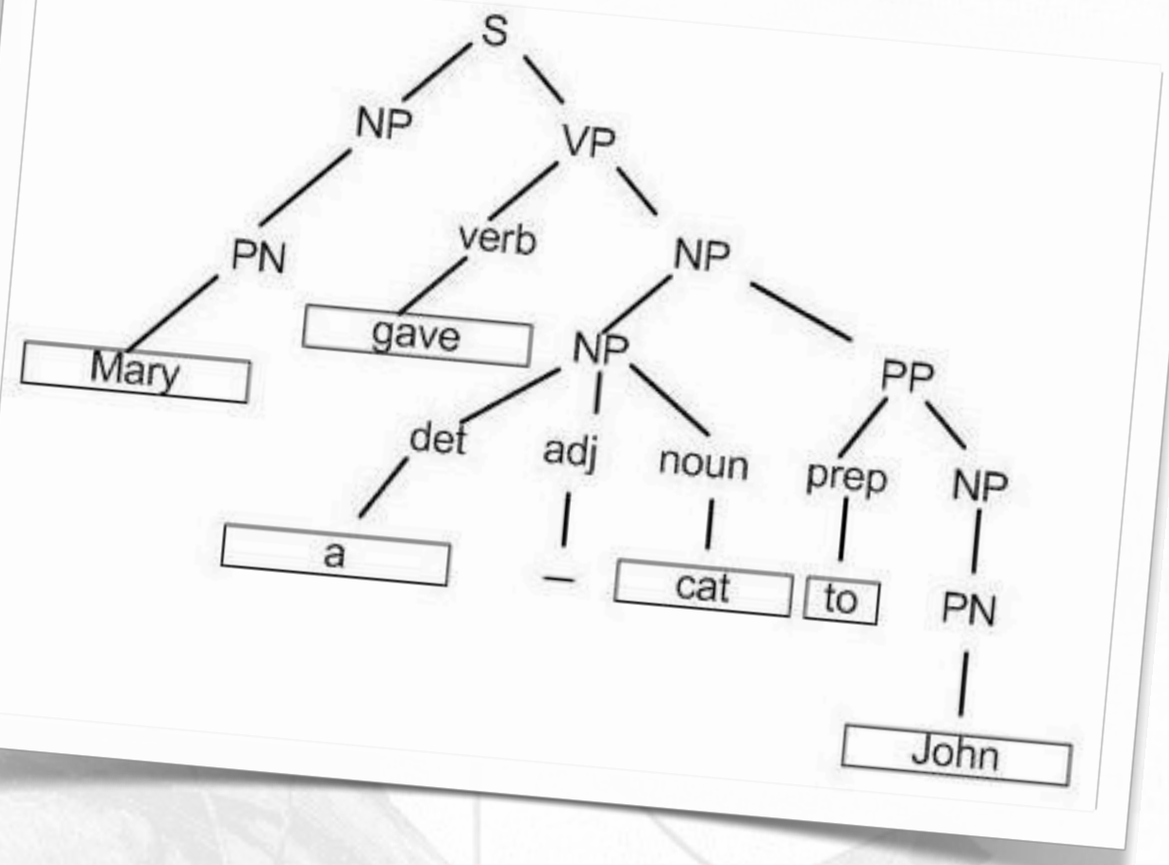
- Syntactic categories
  (verb, noun, preposition, etc...)

(4) Regular Expression Matching

(5) Chunks

- Detection of larger coherent structures in sentences

(6) Syntactic Parser



Ontology Learning - Basic Approach (III)

- **Machine Learning:**

(1) Association Rule Discovery

- Discovery of interesting associations between terms

(2) (Hierarchical) Clustering

- unsupervised learning, in particular clustering of terms

(3) Classification

- Classification of new concepts into an existing hierarchy
- e.g. with Support Vector Machines (SVM), Naive Bayes, knn

(4) Inductive Logic Programming

- Induction of rules from data, i.e. discovery of new concepts from extensional data

(5) Conceptual Clustering

- Formal Concept Analysis, learning concepts and concept hierarchies

Ontology Learning - Layer Cake

$\forall x((country(x) \rightarrow \exists y\ capitalOf(y,x)) \wedge (\forall z\ (capitalOf(z,x) \rightarrow y=z)))$

$river \sqcap mountain = \varnothing$

$capitalOf \sqsubseteq_R locatedIn$

flowThrough(dom:river, range:GeoEntity)

$capital \sqsubseteq_C city$, $city \sqsubseteq_C InhabitedGeoEntity$

c:=country:=<description(c), uri(c)>

{country, nation, Land}

river, country, nation, city, capital, ...

General Axions

Axiomatic Schemata

Hierarchy of Relations

Relations

Hierarchy of Concepts

Concept Description

Multilingual Synonyms

Terms

Ontology Learning Tasks

- which tasks from ontology development can be automated?

Ontology Learning Tasks

- Ontology creation
- Ontology schema extraction
- Extraction of ontology instances
- Ontology integration and navigation
- Ontology update
- Ontology enrichment

Ontology Learning Tasks

- **Ontology Creation**
  - Design of an ontology from the scratch by a team of experts
  - *Maschine Learning* (ML) supports the experts during the design phase by
    - Suggestions of well suited relations among concepts
    - Integrity / consistency checking of the designed ontology
    - ...

Ontology Learning Tasks

- **Ontology Schema Extraction**
  - Extraction of ontology schemata from heterogeneous documents, such as e.g. web documents or text documents
  - *Machine Learning* uses **input data** and **meta ontology** to create fully-fledged domain ontologies (with the help of human experts)

Ontology Learning Tasks

- **Extraction of Ontology Instances (Populate Ontologies)**
  - Extraction of ontology instances from semi-structured data or unstructured data to populate already existing ontology schemata with individuals
  - applies technologies from *Information Retrieval* and *Data Mining*

Ontology Learning Tasks

- **Ontology Integration and Navigation**
  - Reconstruction of existing knowledge bases and navigation in existing knowledge bases,
  - as e.g. translation of an existing knowledge base from FOL to OWL 2 DL

Ontology Learning Tasks

- **Ontology Update**
  - Extension, reconstruction and adaption of already existing ontologies, as e.g. adation of an ontology to a changed domain
  - relates to parts of ontologies that have been created in the way that they can be changed

Ontology Learning Tasks

- **Ontology Enrichment**
  - (also *Ontology tuning*) relates to automated update of smaller parts of existing ontologies
  - doesn‘t changes concepts and relations, but refines them (more precise)
  - in difference to ontology update only parts of the ontology are considered that usually shouldn‘t be changed

04 - Ontology Alignment

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

Knowledge Engineering with Semantic Web Technologies

Lecture 6: Knowledge Engineering

04: Ontology Alignment

Dr. Harald Sack
Hasso-Plattner-Institut for IT Systems Engineering
University of Potsdam
Spring 2014

Lecture 6: Knowledge Engineering

Open HPI Course: Knowledge Engineering with Semantic Web Technologies

04 - Ontology Alignment

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

Ontology Alignment

- **Ontologies** are not the **Reality**
- Ontologies are a **context-dependent projection (model) of the Reality**
- Therefore, different ontologies might represent the same (or similar) knowledge, as e.g. ontologies might
  - reflect different tasks and requirements for applications
  - follow different conventions or restrictions

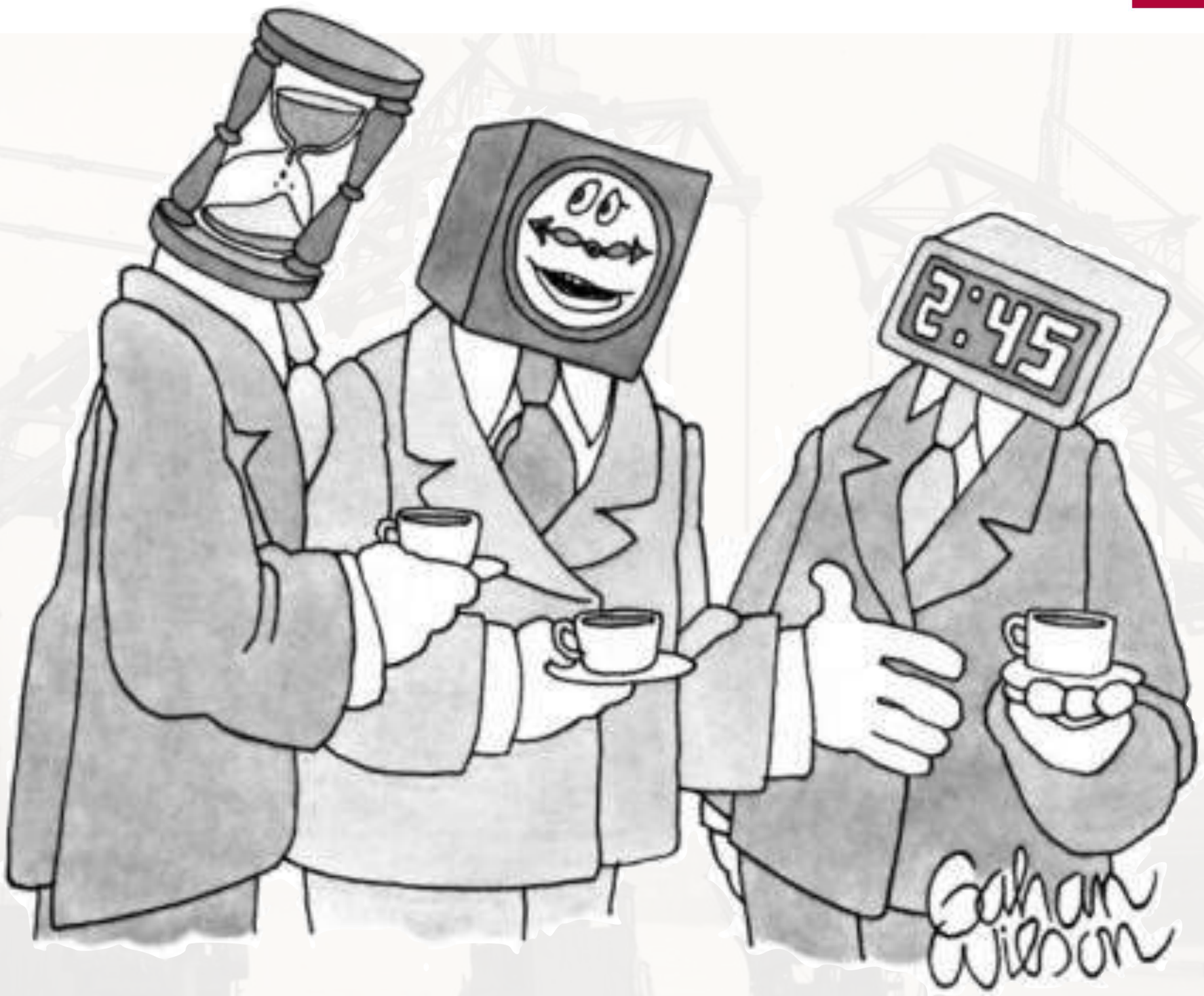
How Ontologies are Different?

- the **same term describes different concepts**
  - e.g. `Author` - *writer of a book vs. creator of a document*
- **different terms describe the same concept**
  - e.g. `Author` vs. `Writer`
- **different modeling conventions and paradigms**
  - e.g. `intervals` vs. `points` - *to describe temporal aspects*
- **different level of granularity**
  - e.g. `Fiction` vs. `PoliticalFiction ScienceFiction, RomanticFiction,` *etc. as literary Genres*
- **different coverage or different point of view**
- etc.

Heterogeneity of Ontology

- **Syntactical Heterogeneity:**
  - Ontologies are available in different ontology representation languages (E.g.: in OWL DL and F-Logic)
  - can be resolved on the conceptual level, most times preserving the semantics
- **Terminological Heterogeneity:**
  - Naming differences for the identification of entities in different ontologies (E.g.: `Author` vs. `Writer`)
  - Might occur because different (natural) languages are used
- **Conceptional (Semantic) Heterogeneity**
  - Ontologies model the same domain, but in different ways
  - Differences might occur in coverage, granularity, perspective, etc.
- **Semiotic (Pragmatic) Heterogeneity**
  - Differences in interpretation of the domain to be modelled by humans (difficult)

Ontology Alignment

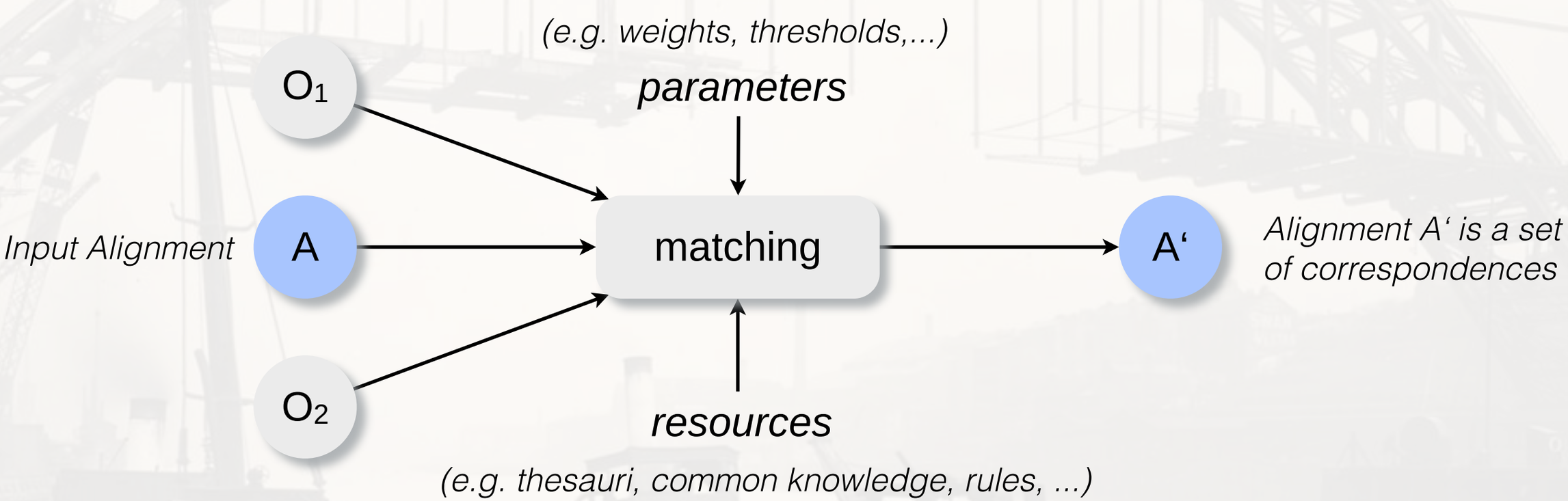


"Basically, we're all trying to say the same thing."

Author: Gahan Wilson
Source: Conde Nast

Ontology Alignment

- **Ontologies Alignment** or **Ontology Matching** is the process of determining *correspondences* between ontological concepts



Correspondence

- Given the ontologies O_1 and O_2, a **correspondence** among the entities e_1 and e_2 from O_1 respectivley O_2, is defined as

$$\langle \mathbf{id}, \mathbf{e_1}, \mathbf{e_2}, \mathbf{r}, \mathbf{n} \rangle$$

- with
  - **id** ... a unique **identifier** of the correspondence
  - **r** ... a **relation**, as e.g. equivalence (=), more general (⊒,≥), less general (⊑,≤), disjointness(⊥), part-of, etc...
  - n ... a **confidence measure** (typically in the range of [0,1]) holding for the correspondence between e_1 and e_2

- the correspondence $\langle id, e_1, e_2, r, n \rangle$ asserts that the relation ***r*** holds between the entities $\boldsymbol{e_1}$ and $\boldsymbol{e_2}$ with confidence ***n***

Complexity of Correspondences

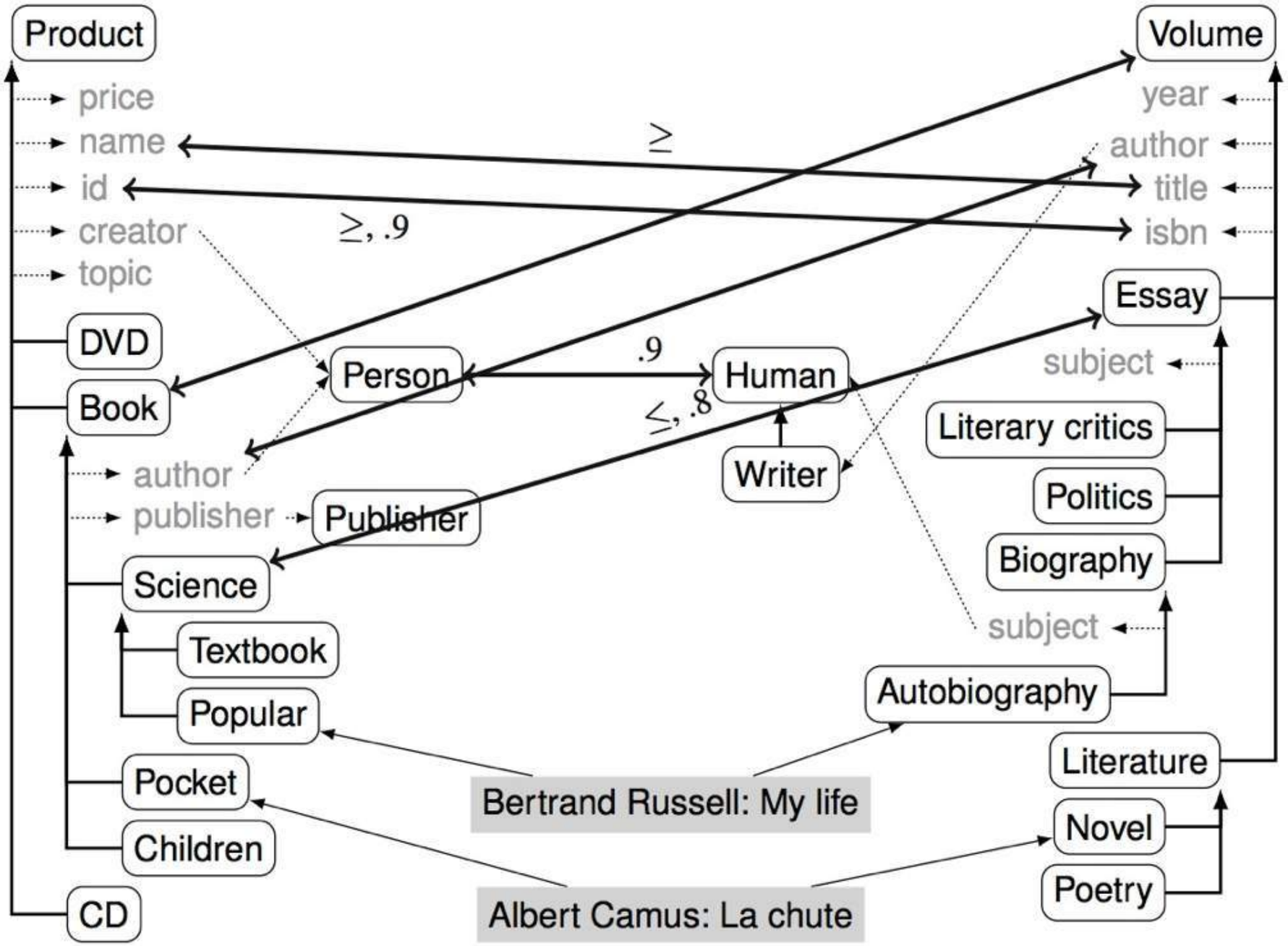
- Examples of **simple correspondences**:
  - `http://dbpedia.org/resource/George_Orwell = http://www.freebase.com/m/034bs`
  - `Author = Writer`
  - `Fiction` $\geq_{1.0}$ `ScienceFiction`
  - `rdfs:label` $\geq_{0.9}$ `dc:title`

Complexity of Correspondences

- Examples of more **complex correspondences**:
  - speed = velocity × 2.237
    0.477 × speed = velocity
  - Book(x)∧author(x,y)∧Writer(y) ⇒.85
    writtenBy(x,concat(y.firstname, y.lastname))

Ontology Alignment Example

- Book $=_{1.0}$ *Volume*
- id $\geq_{0.9}$ *isbd*
- Person $=_{.9}$ *Human*
- name $\geq_{1.0}$ *title*
- author $=_{1.0}$ *author*
- Science $\leq_{.9}$ *Essay*



Jérôme Euzenat, Pavel Shvaiko: Ontology Matching, Springer, 2007, p.48

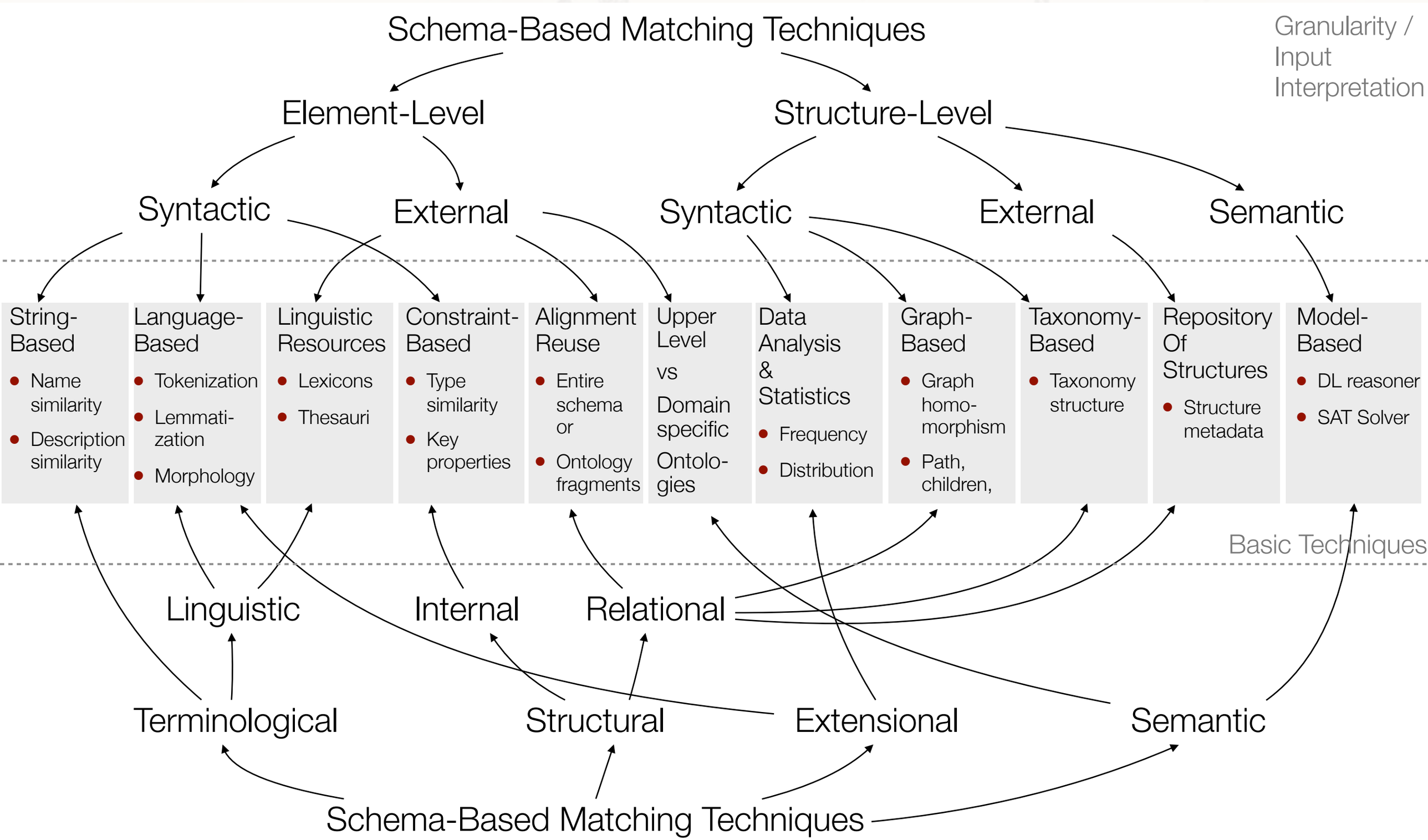
Ontology Matching Techniques

- **Element-level Ontology Matching Techniques** consider ontology entities or their instances in isolation from their relations with other entites or their instances
  - **String-Based** - *matching names or descriptions of entities*
  - **Linguistic-Based** - *use NLP, lexicons, or domain specific thesauri to match words based on linguistic relations (homonymy, synonymy, partonymy, etc.), or exploiting morphological properties*
  - **Constrained-Based** - *take into account internal constraints applied to the definitions of entities, as e.g. types, cardinality of properties, etc.*
  - **Extensional-Based** - *use individual representation of classes, i.e. classes are considered similar if they share many instances.*

Ontology Matching Techniques

- **Structure-level Ontology Matching Techniques** consider ontology entites or their instances to compare their relations with other entites or their instances
  - **Graph-Based** - *consider ontologies as labeled graphs, assumption: if nodes are similar, then also their neighbors must be similar*
  - **Taxonomy-Based** - *like graph-based algorithms, but consider only specialization/generalization relation*
  - **Method-Based** - *take into account semantic interpretation of the ontologies, assumption: if two entities are the same, then they share the same interpretation.*
  - **Data Analysis and Statistics** - *take a large sample, try to find regularities, discrepancies, allows grouping or determining distance metrics,...*

Ontology Alignment Process



Euzenat, Shvaiko: Ontology Matching, Springer 2007

Ontology Merging

- is a process, where from two or more start ontologies a **new ontology** is created.
- The new ontology **unifies** and **substitutes** the original start ontologies.

- **Union Approach**
  The new ontology is the union of all entities of the start ontologies, where conflicts arrising from different representations of identical concepts must be resolved.

- **Intersection Approach** (extensional)
  The new ontology only consists out of those parts of the start ontologies that overlap.

05 - Linked Data Engineering

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

Knowledge Engineering with Semantic Web Technologies

Lecture 6: Knowledge Engineering

05: Linked Data Engineering

Dr. Harald Sack
Hasso-Plattner-Institut for IT Systems Engineering
University of Potsdam
Spring 2014

Lecture 6: Knowledge Engineering

Open HPI Course: Knowledge Engineering with Semantic Web Technologies

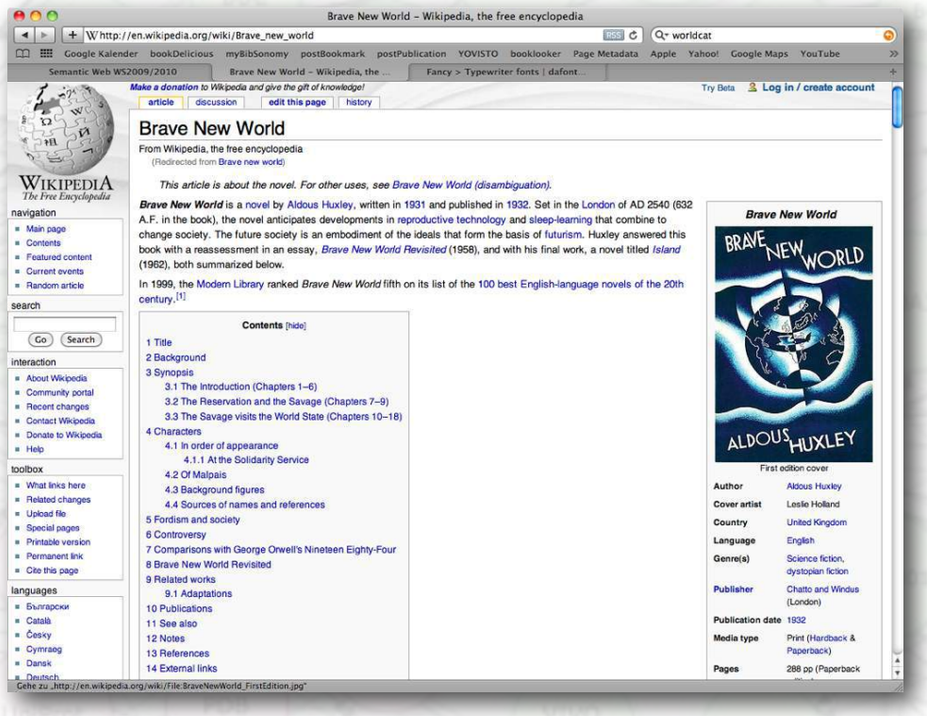
05 - Linked Data Engineering

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

How to get Data from the Web?

- Data can only be found on the Web, if it is available at some website

HTTP
HTML



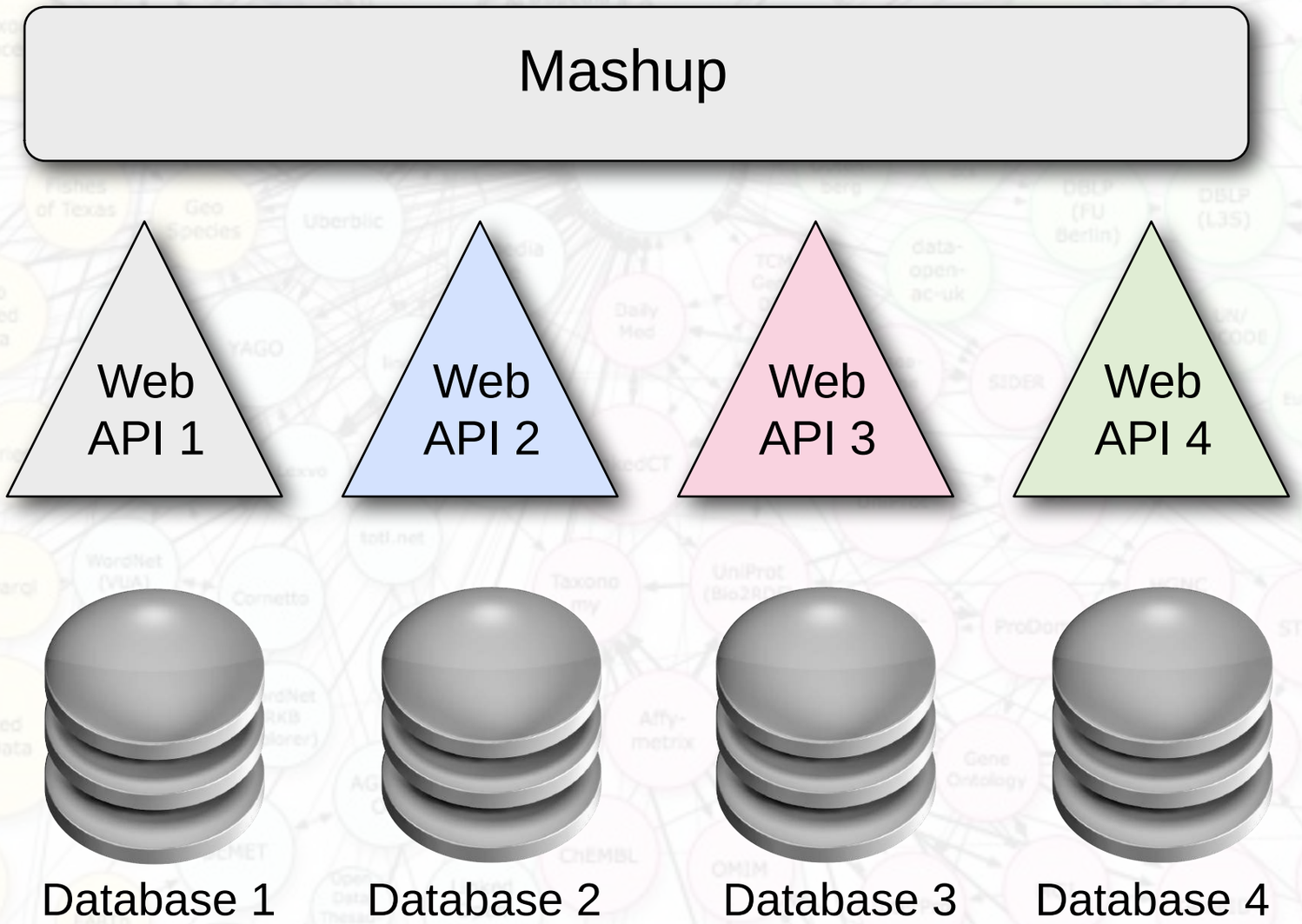
Browser

JDBC

Web-Server

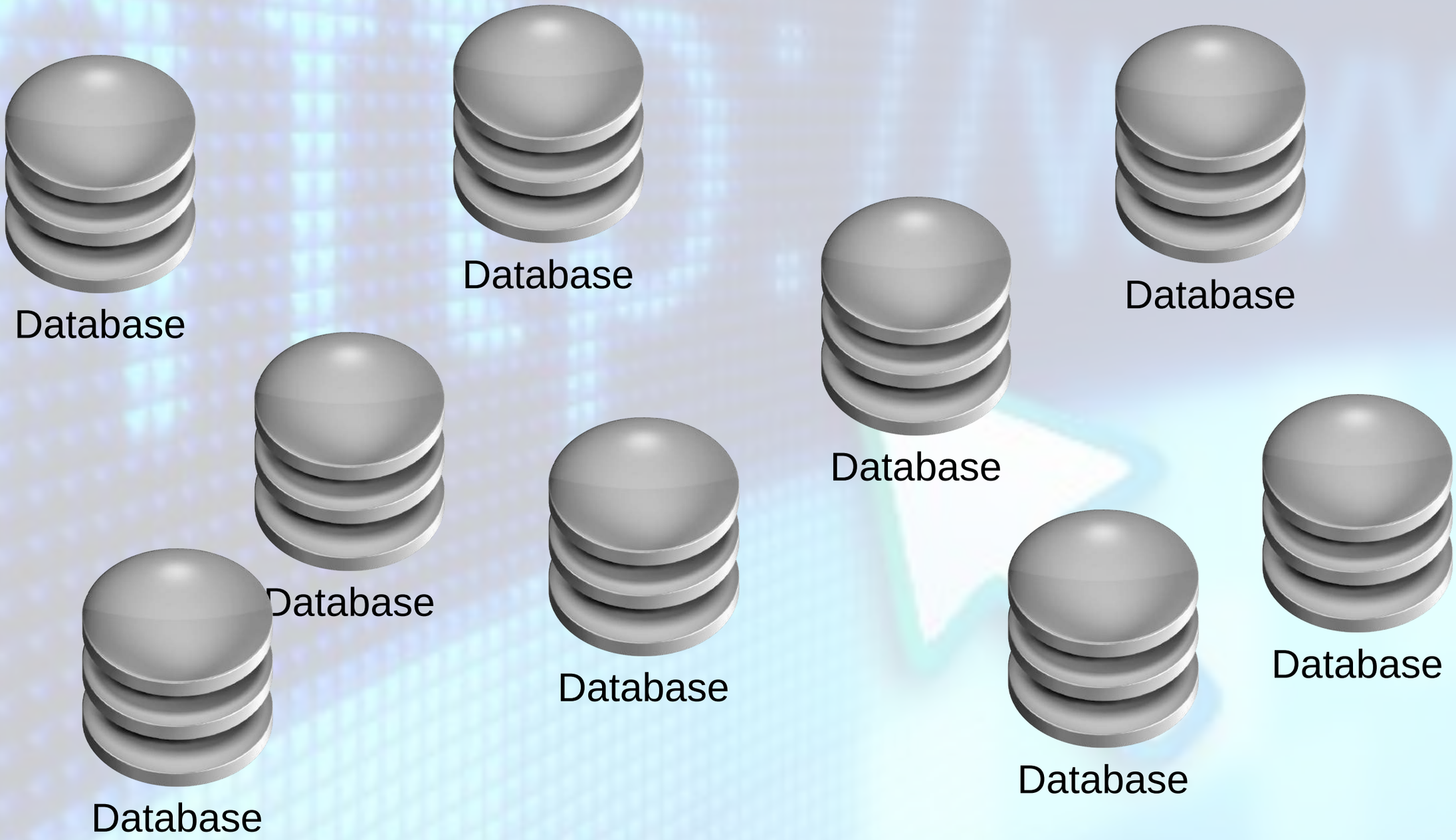
How to get Data from the Web?

- There is a a number of different (proprietary) **Web APIs**, data exchange formats and **Mashups** on top of that



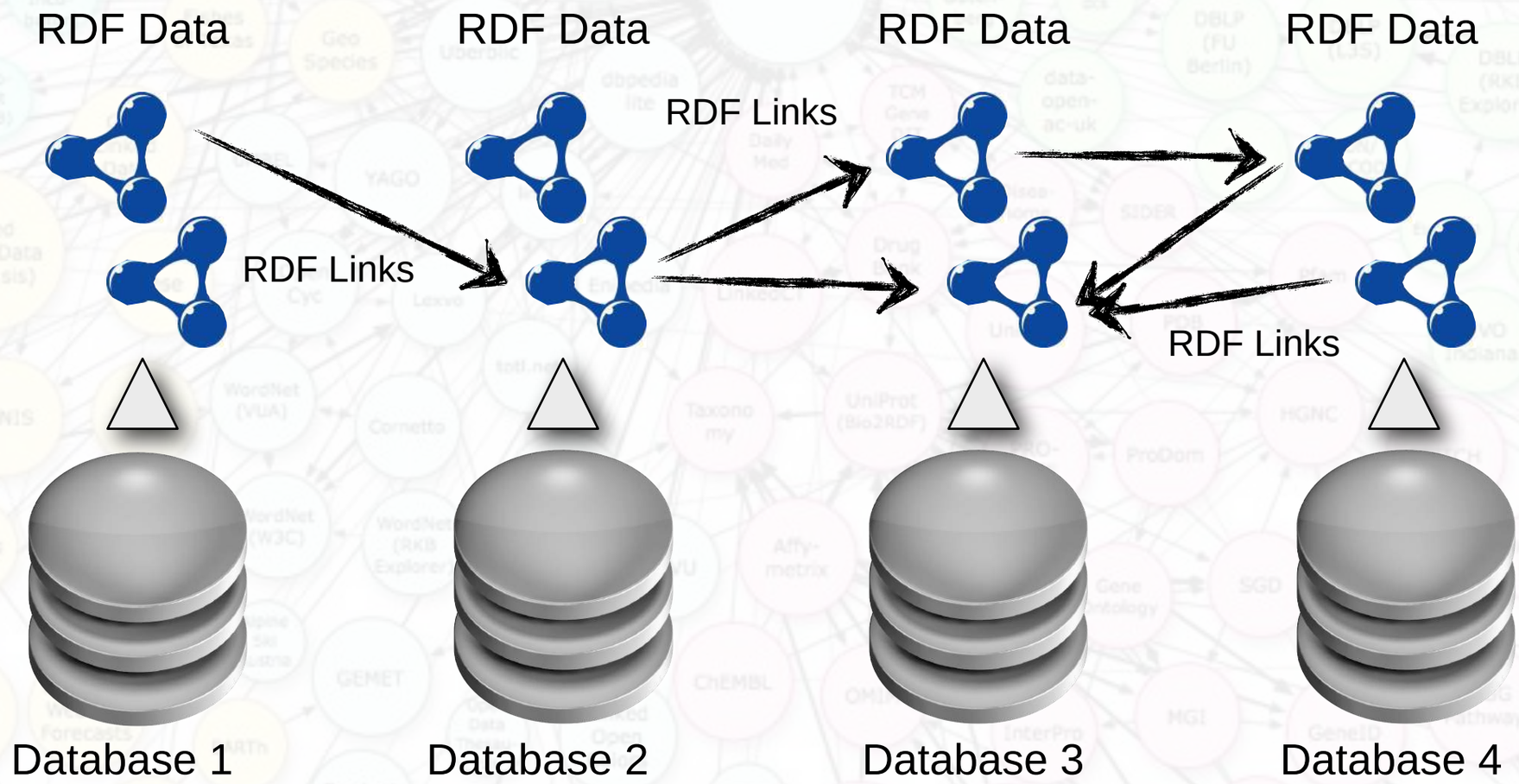
In the Web today...

- Data is locked up in small data islands
- Other applications usually cannot acces this data...



How to get of Closed Data Islands?

- **Apply semantic technologies:**
  - to publish structured data on the web
  - to draw connections from one data source to data from other data sources



Linked Data Principles

(1) Use **URIs** as names for things.

(2) Use **HTTP URIs**, so that people can **look up** those names.

(3) When someone looks up a URI, provide **useful information**, using the **standards** (RDF, SPARQL)

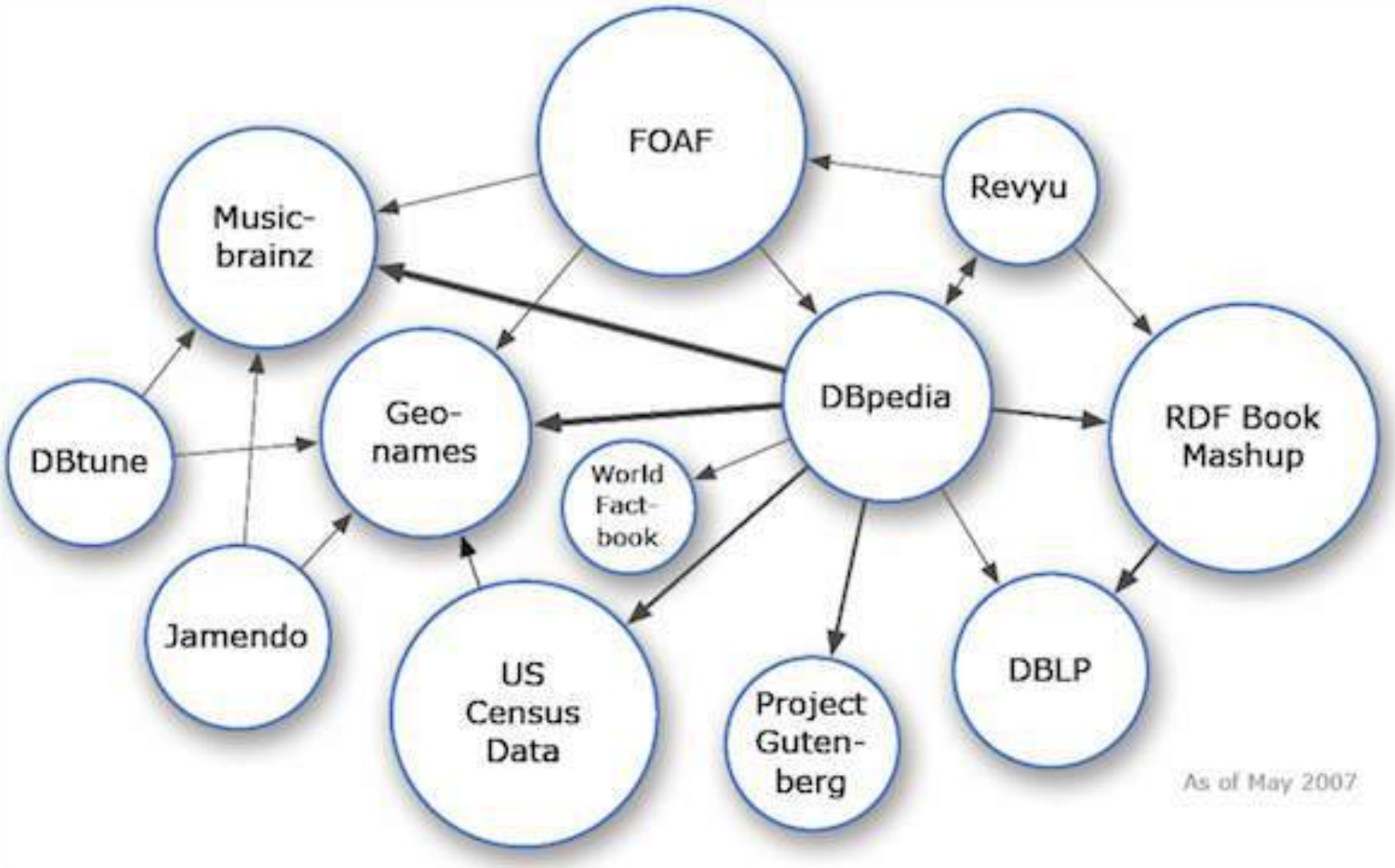
(4) Include **links to other URIs**, so that they can discover more things.

The application of the Linked Data Principles leads to a ‚**Web of Data**‘

964 Datasets
82B RDF Triples
808M Links
as of April 2014

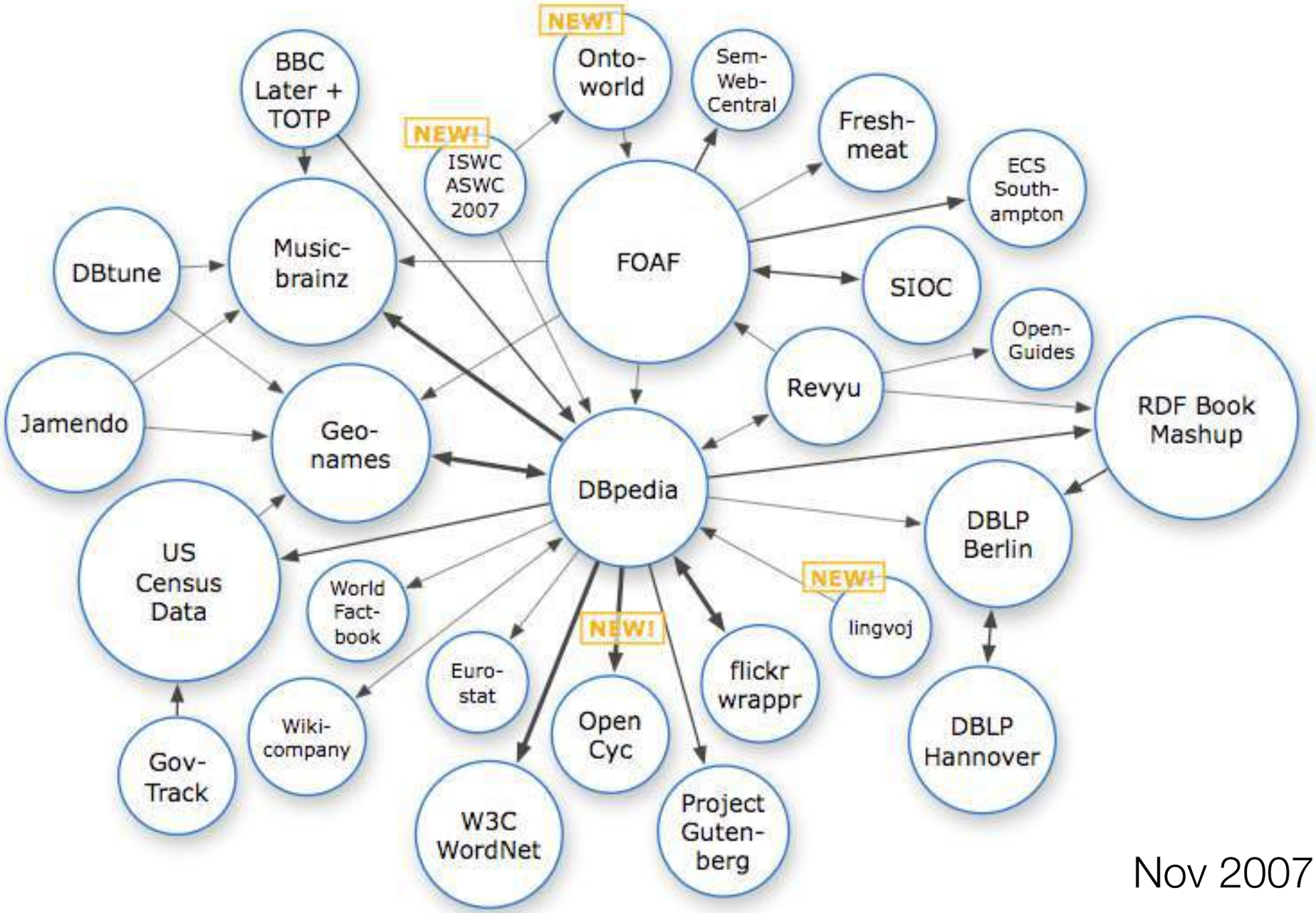
http://lod-cloud.net/versions/2011-09-19/lod-cloud_colored.html

Development of the Web of Data



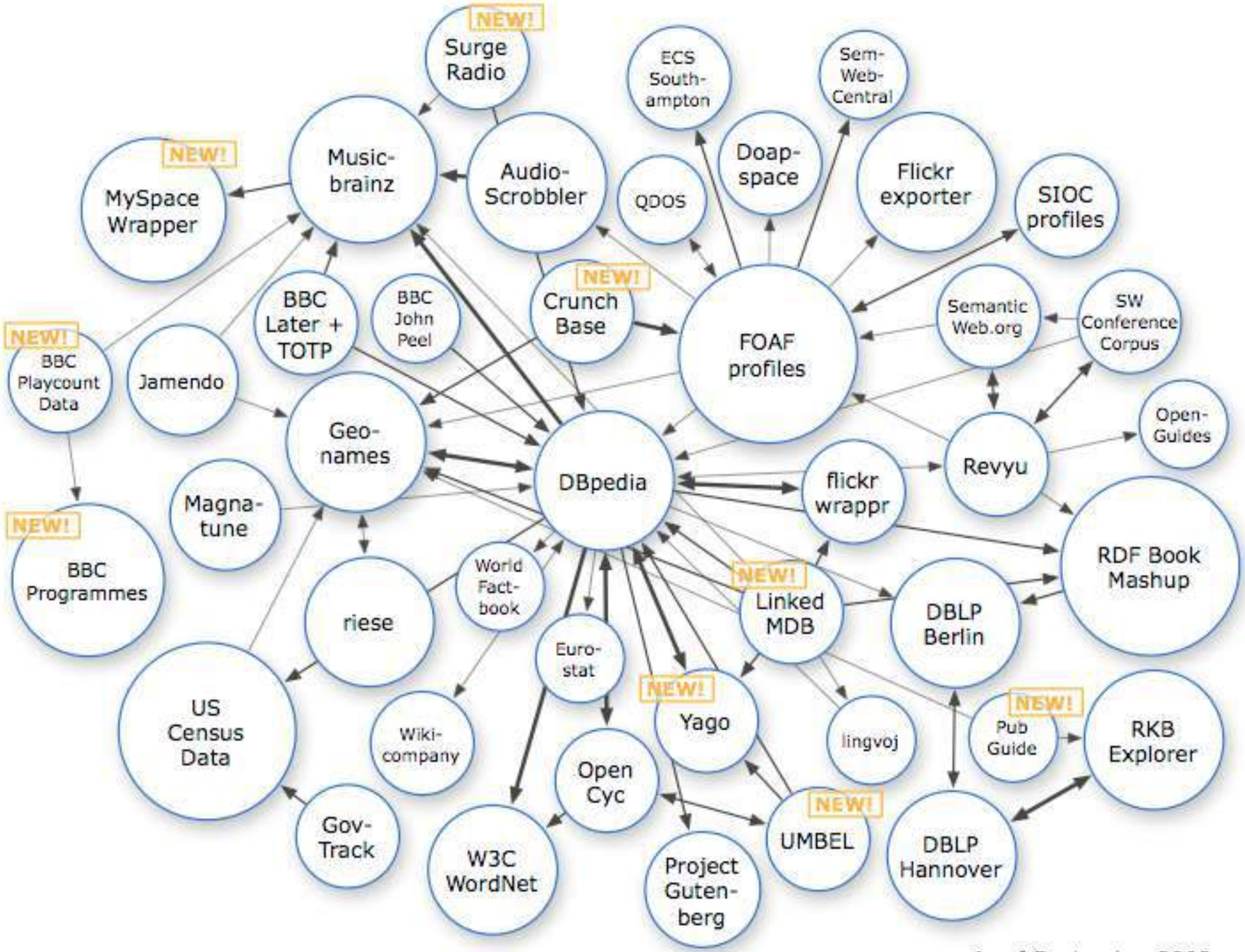
May 2007

Development of the Web of Data

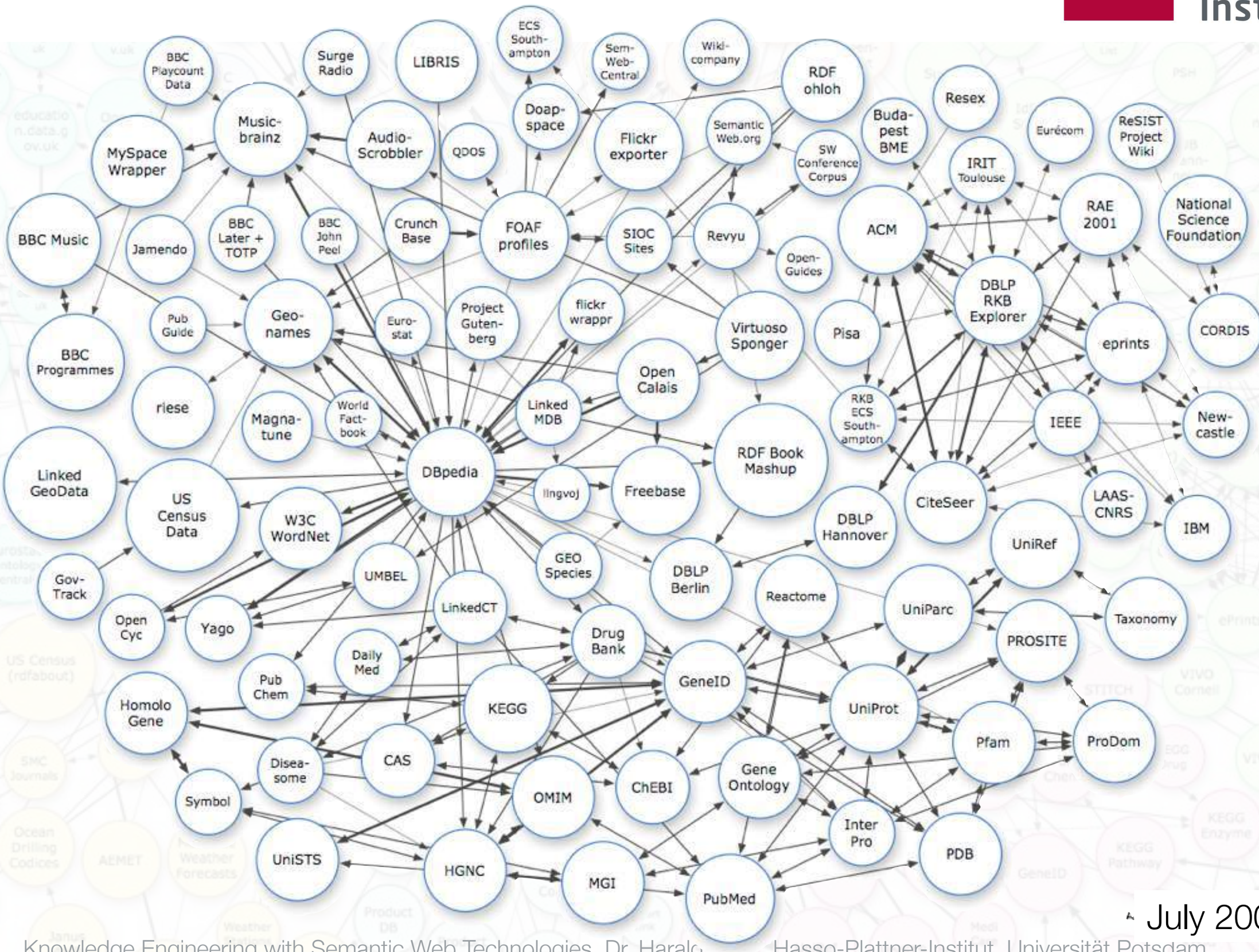


Nov 2007

Development of the Web of Data

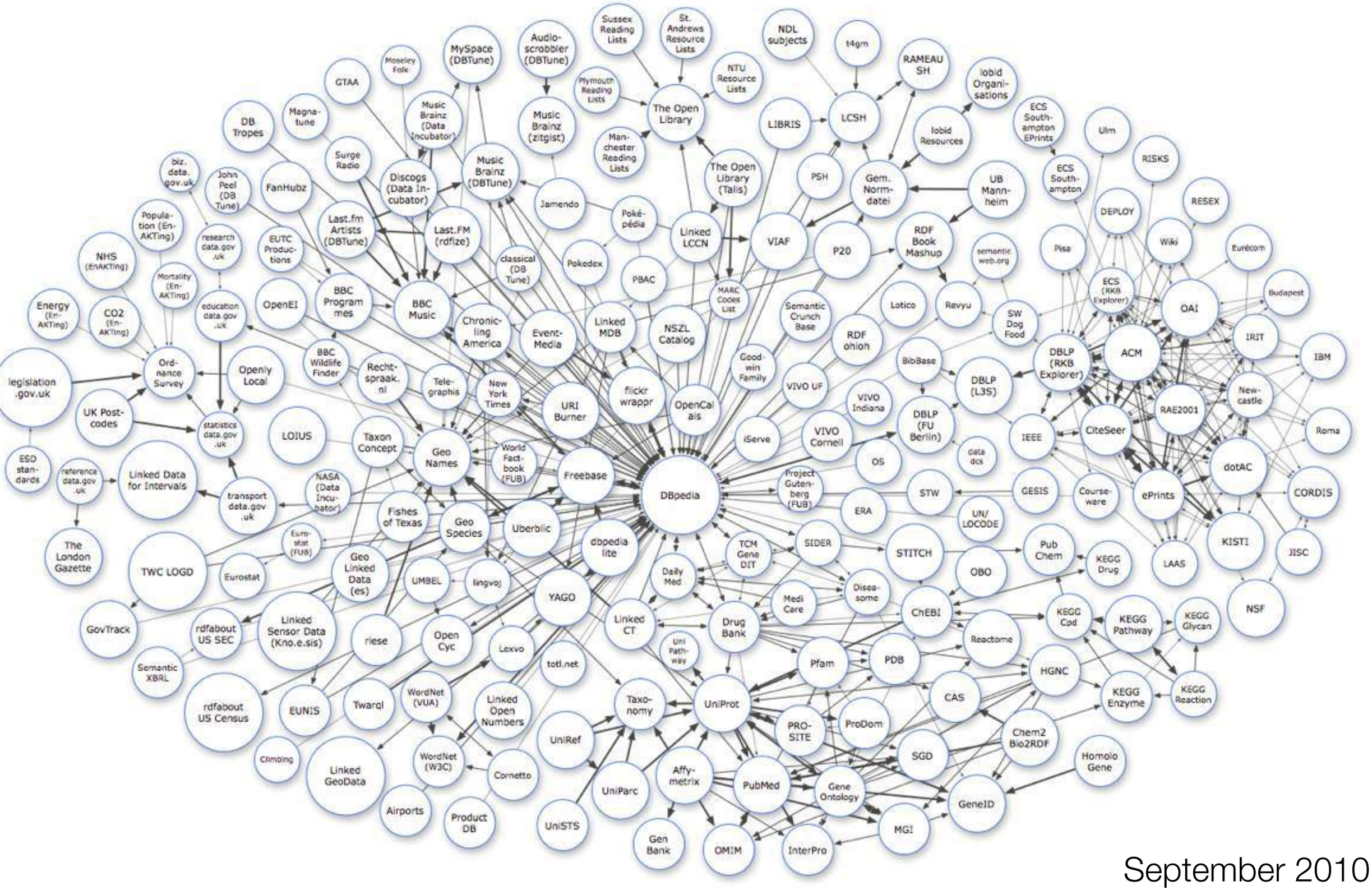


Development of the Web of Data



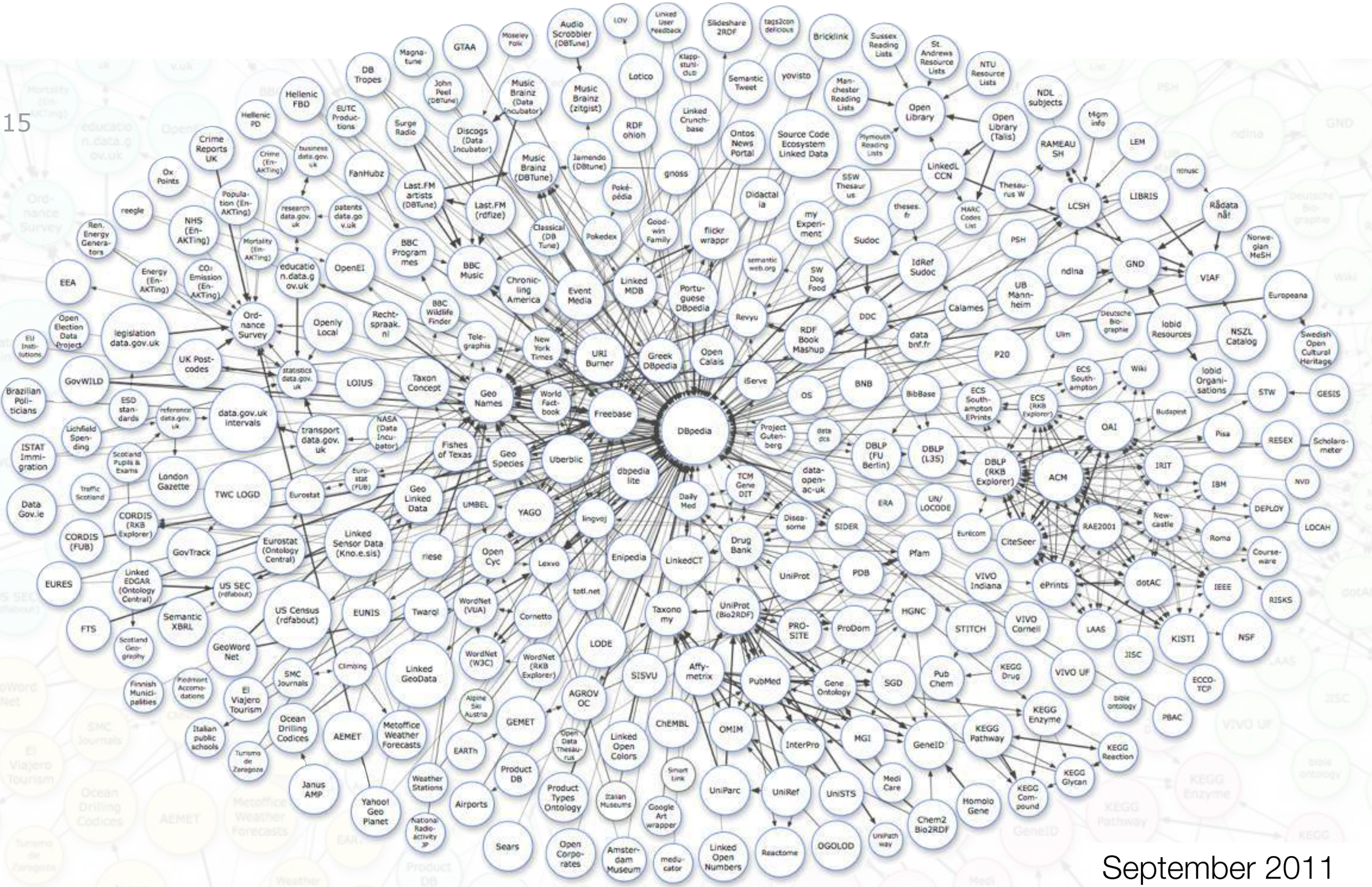
July 2009

Development of the Web of Data



September 2010

Development of the Web of Data



September 2011

Linked Open Data

- Public Linked Data resources in the Web, licensed as „Creative Common CC-BY“
- Tim Berners-Lee‘s 5-Star Criteria for Linked Open Data

★ Available on the web (whatever format) but with an **open licence**, to be Open Data

★★ Available as **machine-readable structured data** (e.g. excel instead of image scan of a table)

★★★ as (2) plus **non-proprietary format** (e.g. CSV instead of excel)

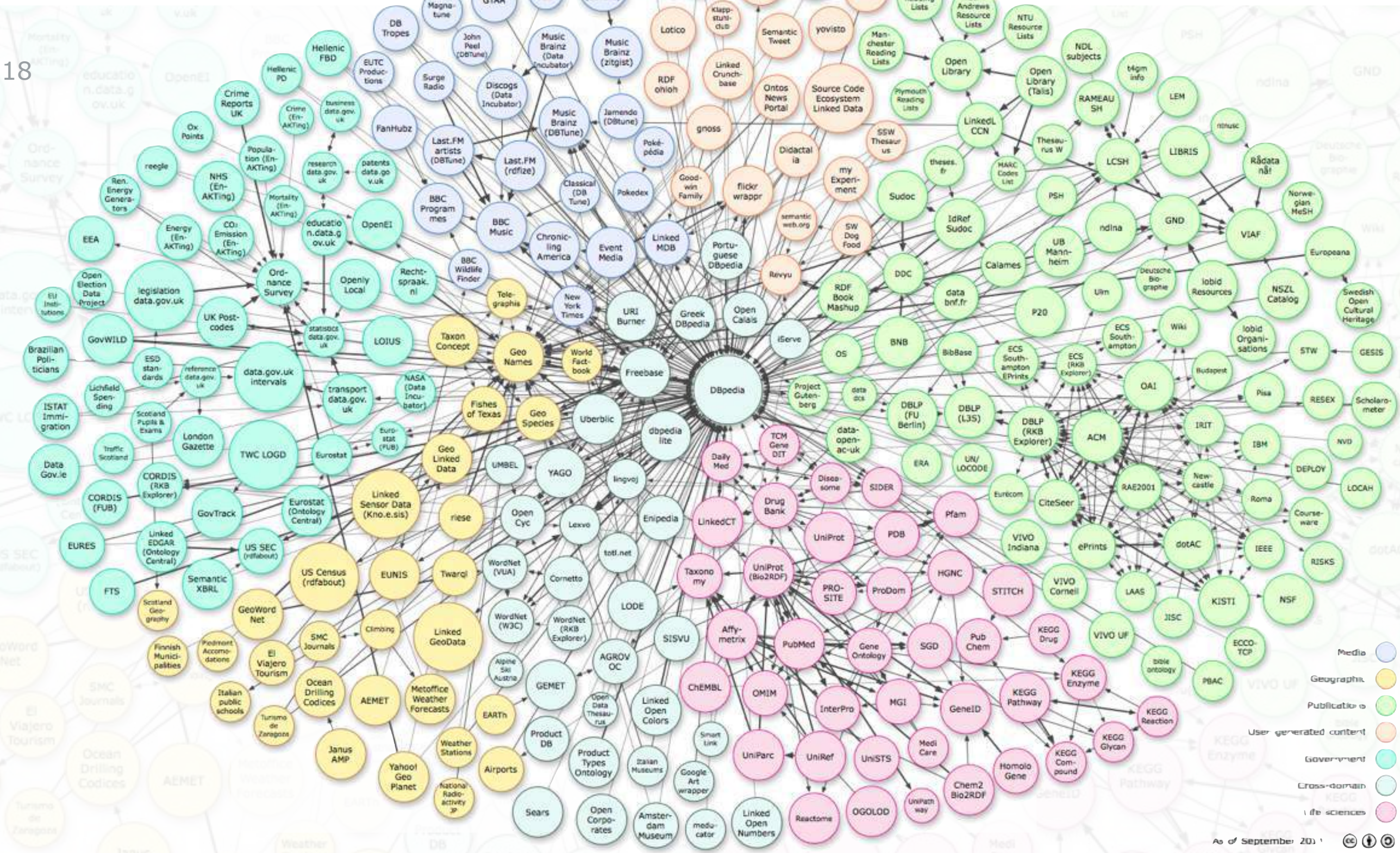
★★★★ All the above plus: use **open standards from W3C** (RDF and SPARQL) to identify things, so that people can point at your stuff

★★★★★ All the above, plus: **link your data to other people’s data** to provide context

Linked Open Data

Linked Data - A Guided Tour...

- ordered by categories

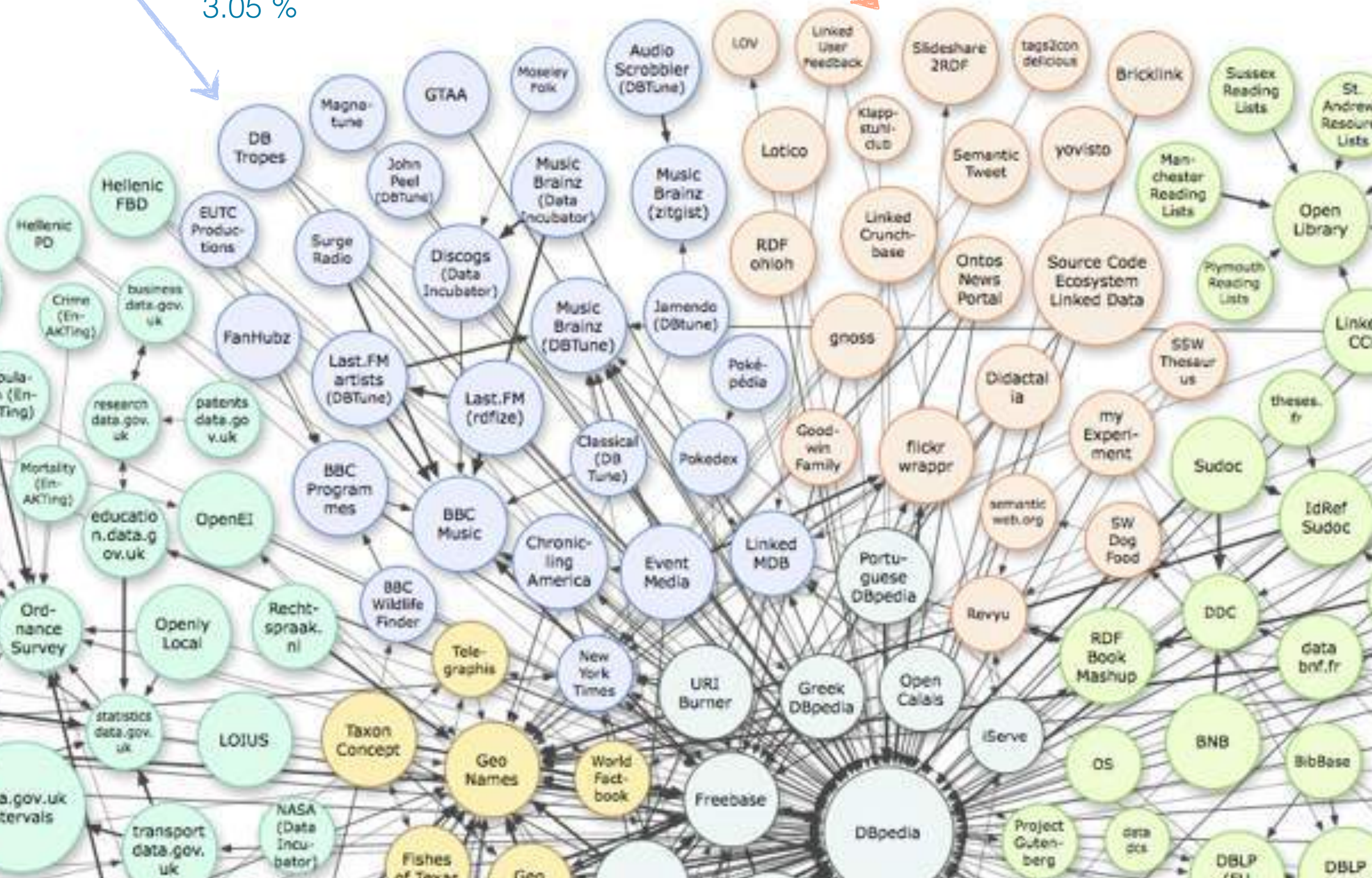


Media

41 data sets
2.568.964.515 RDF Triples
3.05 %

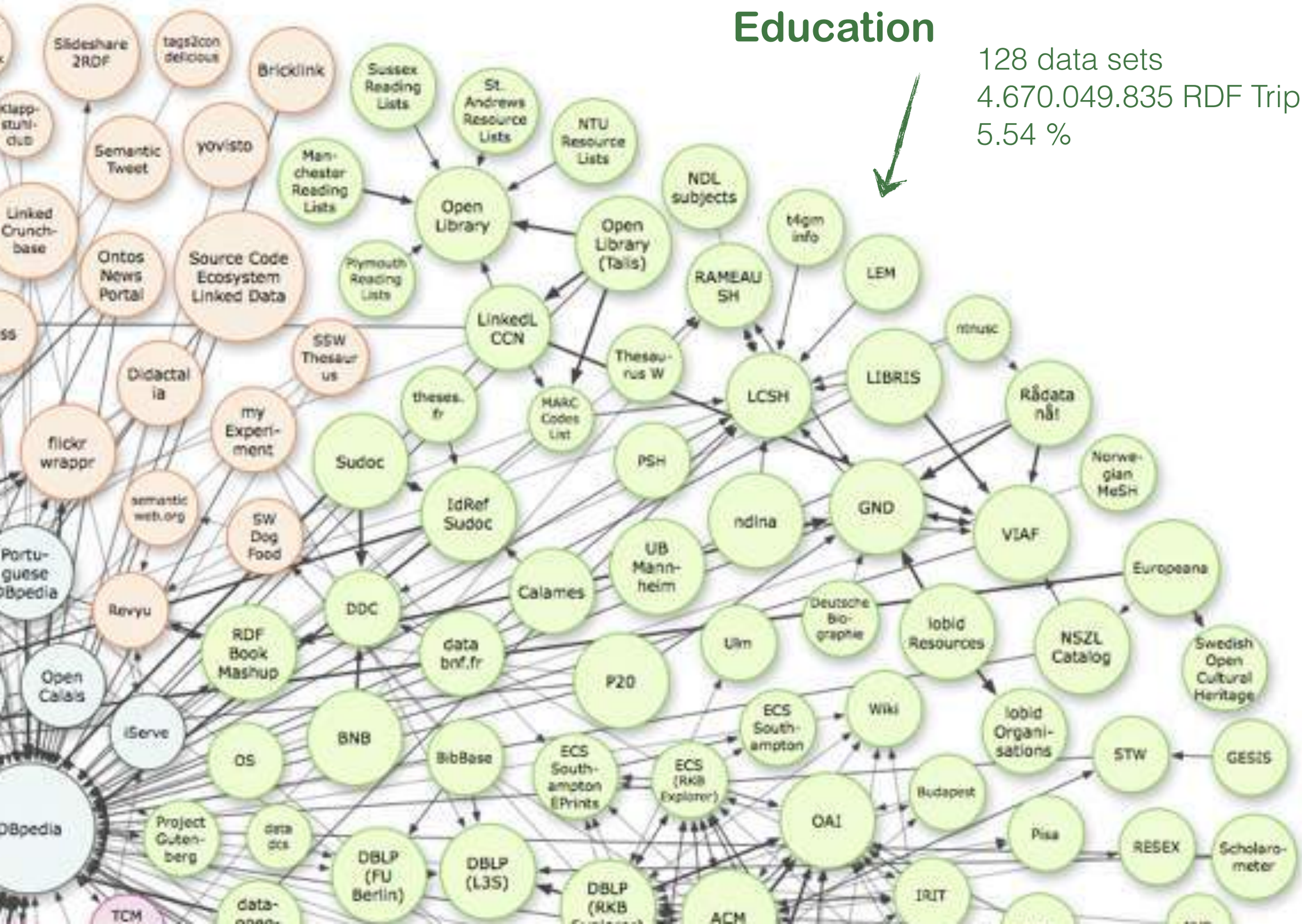
User Generated Content

49 data sets
2.283.111.223 RDF Triples
2.71 %



Libraries and Education

128 data sets
4.670.049.835 RDF Triples
5.54 %

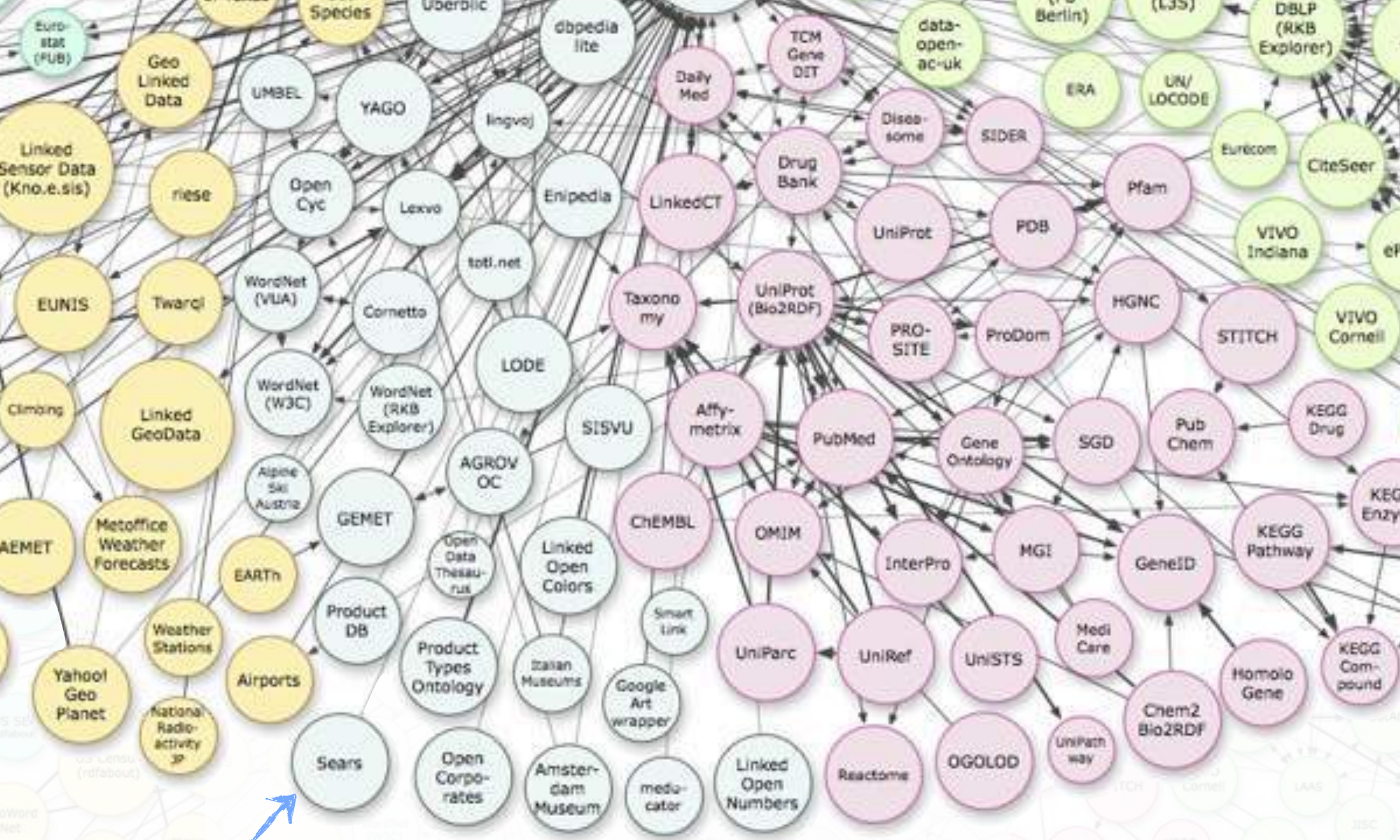


Government

139 data sets
38.246.327.678 RDF Triples
45,5 %

Geographic

46 data sets
4.830.360.509 RDF Triples
5,73 %


Cross-Domain
84 data sets
9.161.202.967 RDF Triples
10,87 %
Life Sciences
92 data sets
7.370.024.776 RDF Triples
8,75 %

Semantic Mashups

- **Semantic Mashups** are applications that use linked RDF data from various data sources.

- in difference to *interfaces* and *exchange formats* or *ordinary Web APIs*, **Linked Data** offers the following benefits:
  - a flexible and standardized data format (**RDF**)
  - standardized access mechanism (**http**)
  - possibility to put links (**RDF-Links**) among different data sources
    - » enables navigation
    - » is supported by search engines (crawler)
    - » enables expressive search facilities over the crawled data and beyond

06 - Linked Data Engineering - Pt.2

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

Knowledge Engineering with Semantic Web Technologies

Lecture 6: Knowledge Engineering

06: Linked Data Engineering Part 2

Dr. Harald Sack
Hasso-Plattner-Institut for IT Systems Engineering
University of Potsdam
Spring 2014

Lecture 6: Knowledge Engineering

Open HPI Course: Knowledge Engineering with Semantic Web Technologies

06 - Linked Data Engineering Pt.2

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

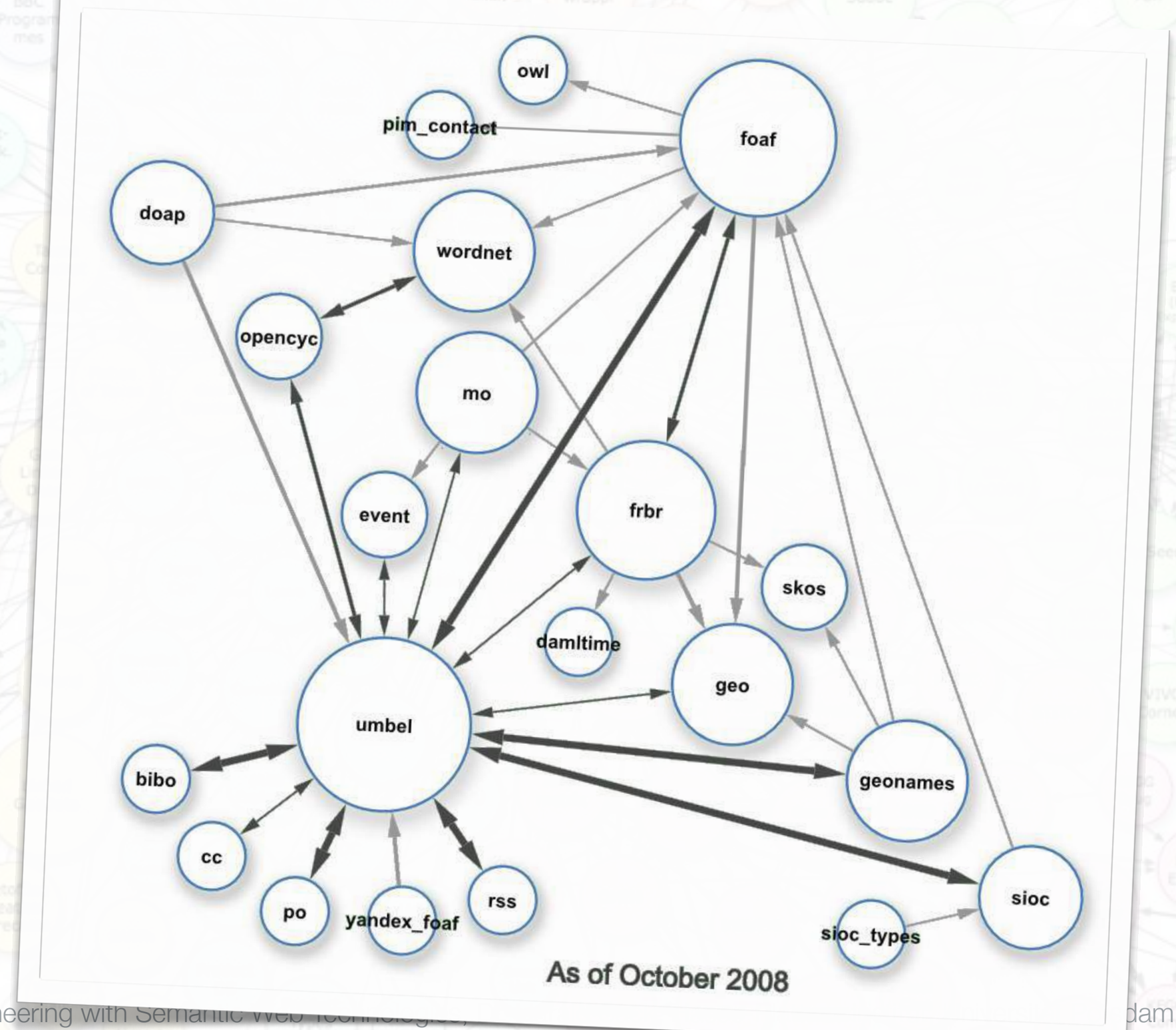
The **Web of Data** enables the use of Linked Data Resources on the Web

964 Datasets
82B RDF Triples
808M Links
as of April 2014

http://lod-cloud.net/versions/2011-09-19/lod-cloud_colored.html

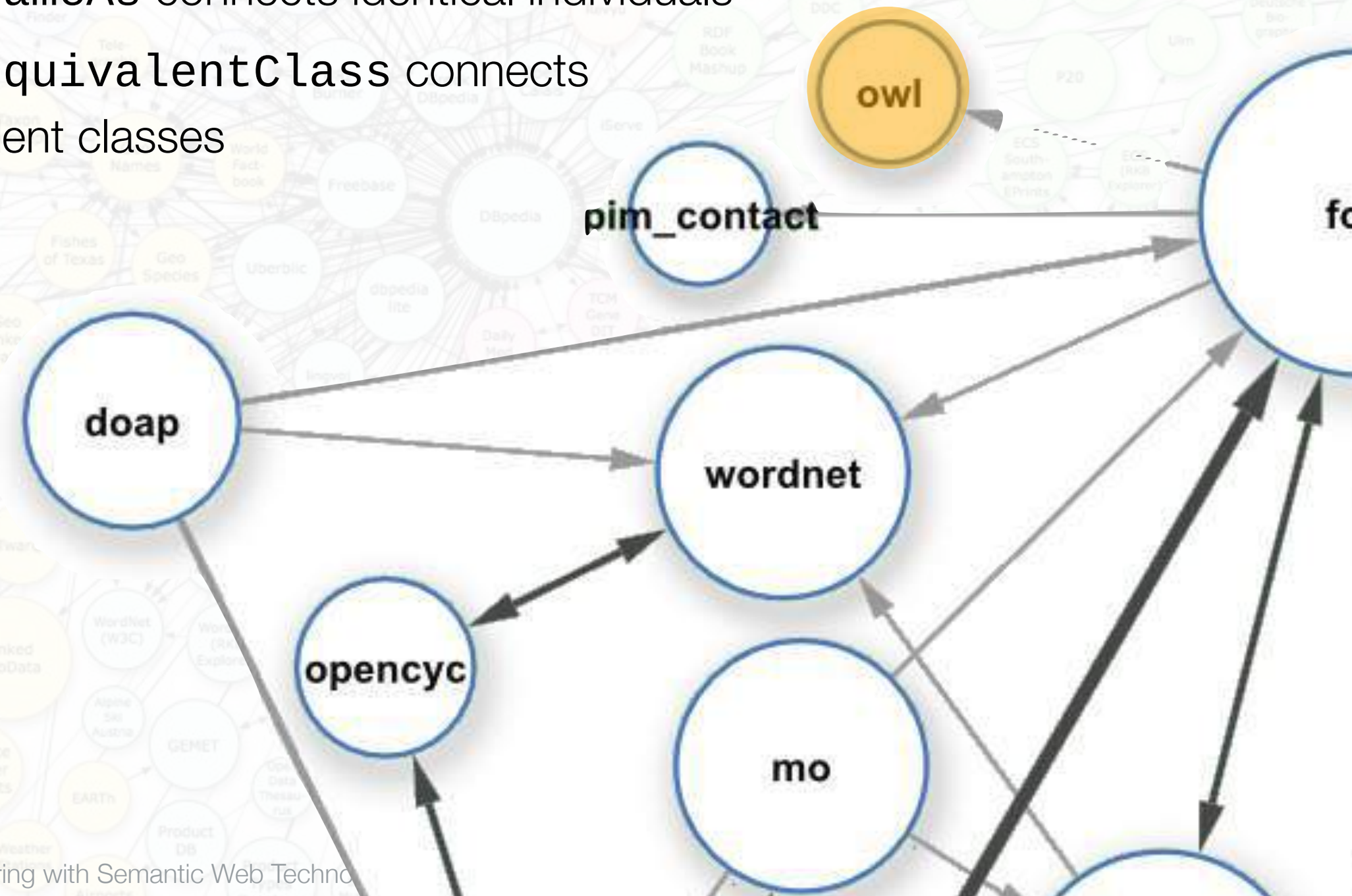
Linked Data Ontologies

- Ontologies hold the Linked Data Cloud together



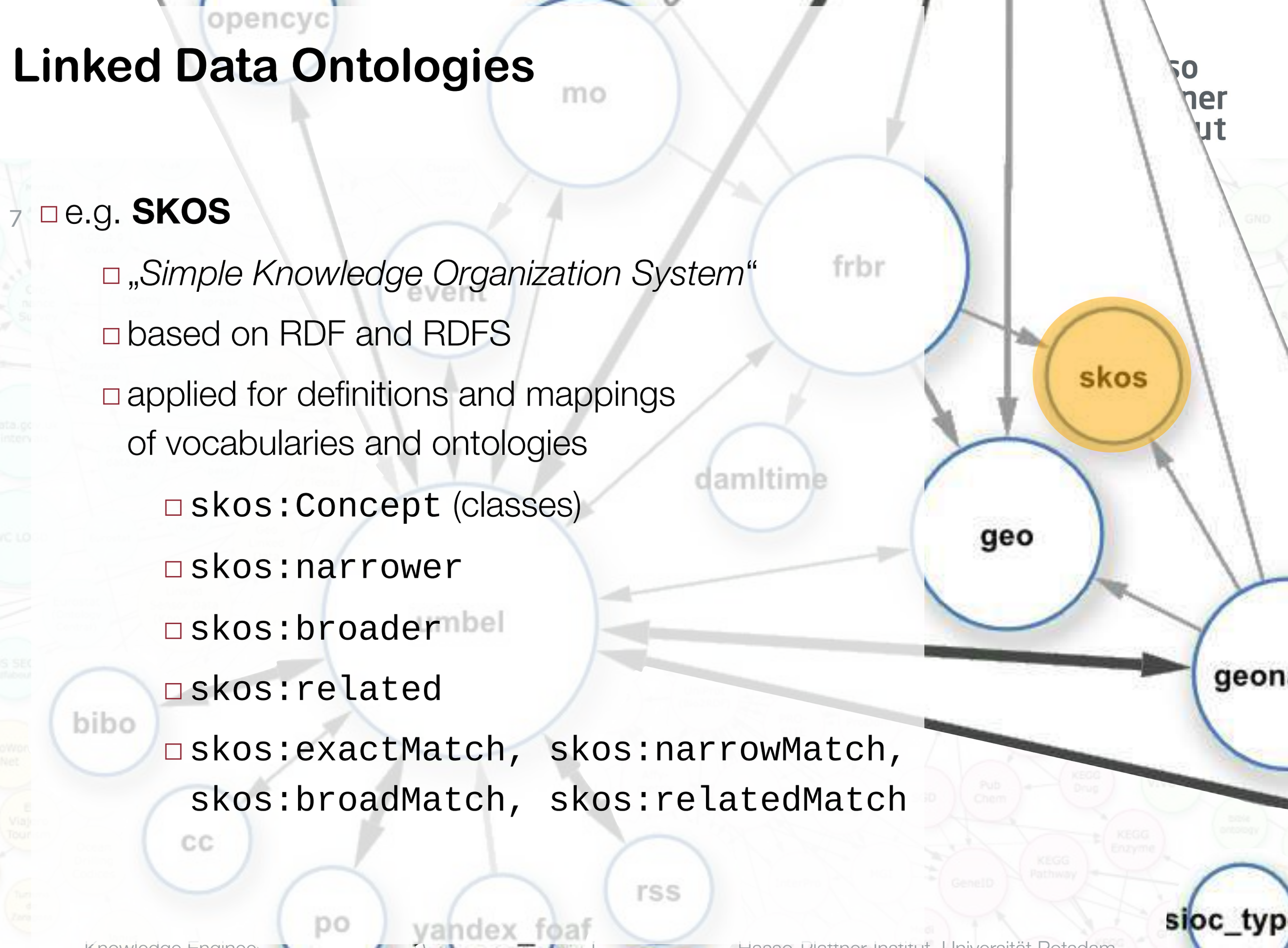
Linked Data Ontologies

- e.g. **OWL**
  - `owl:sameAs` connects identical individuals
  - `owl:equivalentClass` connects equivalent classes



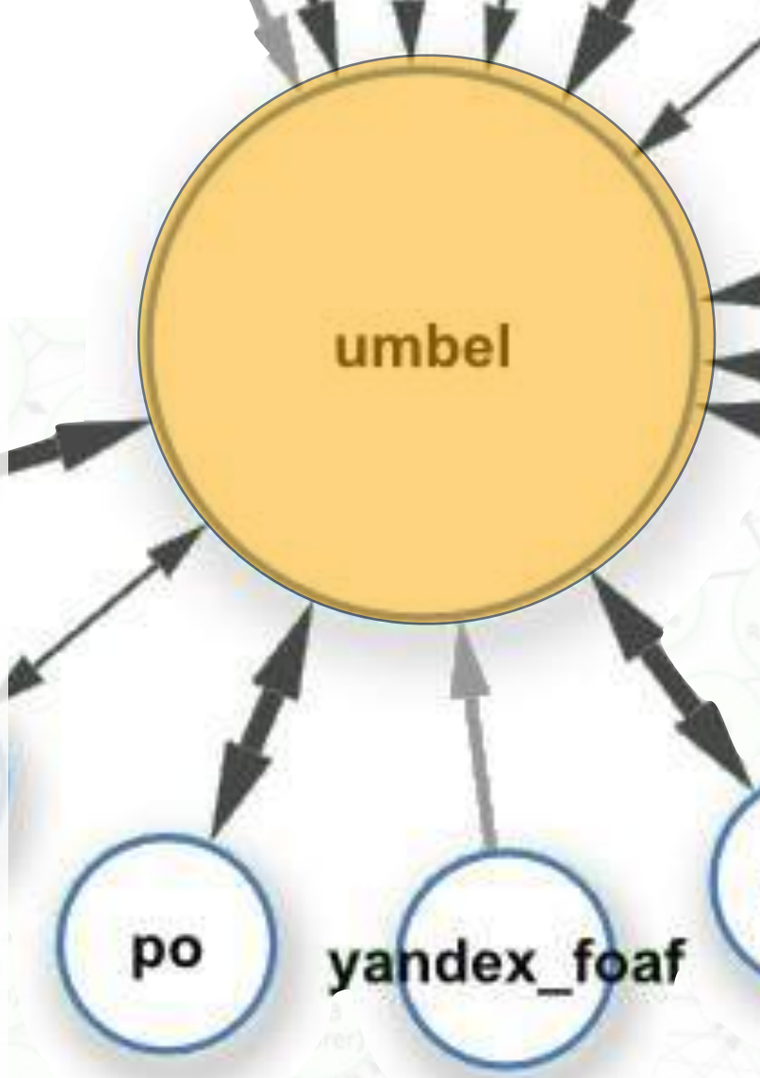
Linked Data Ontologies

- e.g. **SKOS**
  - „*Simple Knowledge Organization System*“
  - based on RDF and RDFS
  - applied for definitions and mappings of vocabularies and ontologies
    - `skos:Concept` (classes)
    - `skos:narrower`
    - `skos:broader`
    - `skos:related`
    - `skos:exactMatch, skos:narrowMatch, skos:broadMatch, skos:relatedMatch`



Linked Data Ontologies

- e.g.. **umbel** (version 1.0, Feb. 2011)
  - „*Upper Mapping and Binding Exchange Layer*“
  - Subset of OpenCyc as RDF Triples based on SKOS and OWL2
  - Upper Ontology with 28.000 concepts (`skos:Concept`)
  - 46.000 Mappings into DBpedia, geonames, e.a. (`owl:equivalentClass, rdfs:subClassOf`)
  - Links to more than 2 Mio Wikipedia pages



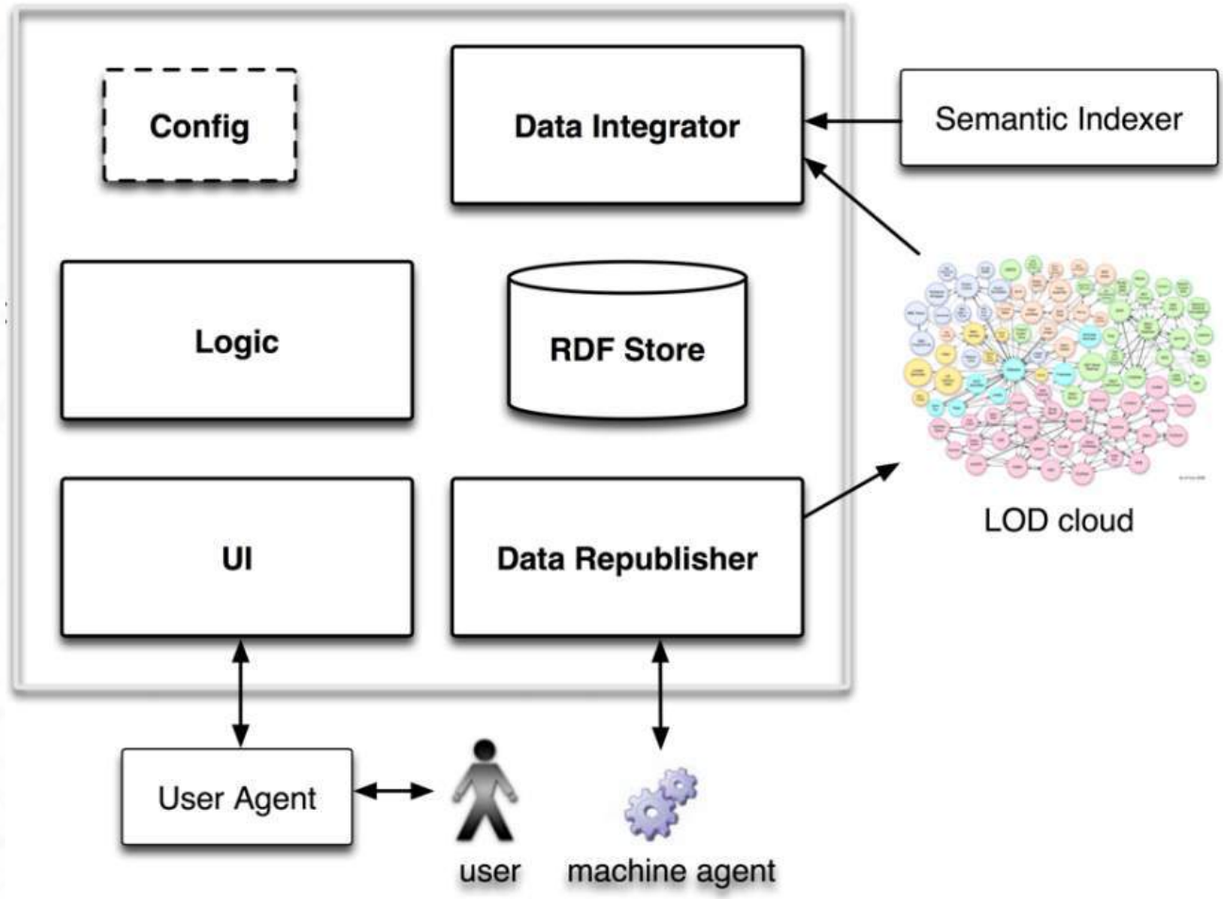
umbel

Linked Data Sources in the Web

- **Native publication**
  - D2R-Server, OpenLink Virtuoso, Pubby, etc.
- **Implementation of Wrappers** around existing applications / APIs
  - SIOC Exporter for Wordpress, Drupal, phpBB,...
  - RDF Book Mashup (Amazon API, Google Base-API,...)
- **Linking Open Data Project**
  - Semantic Web Education and Outreach W3C working group
  - Catalogue of all known sources of linked data with an open source license
    - » DBPedia, Flickr, Open-Cyc, FOAF, SIOC, GeoNames, ...

Linked Data Driven Web Applications

- Required Components:
  - **Local RDF Store**
    - caching of results
    - permanent storage
  - **Logic** (Controller) and **User Interface** (=Business Logic)
    - (not LOD specific)
  - **Data Integration component**
    - get data directly from LOD-Cloud or
    - via Semantic Indexer
  - **Data Republishing component**
    - write back application dependent data into the Web of Data



M.Hausenblas:
Linked Data Applications, DERI Technical Report, 2009

Linked Data Driven Web Applications

- Access to Linked Data via **SPARQL endpoints**
- ...but where do I find SPARQL endpoints?
  - W3C: Currently Alive SPARQL Endpoints
    http://esw.w3.org/SparqlEndpoints

- SPARQL endpoints are a **RESTful Web Services**
  - **`HTTP GET`** Request with SPARQL query
  - Result available as
    - XML, JSON, plaintext (SPARQL Select/Ask)
    - RDF/XML, NTriples, Turtle, N3
      (SPARQL Describe/Construct)
    - Data format can be determined via HTTP Accept Header
      z.B. `Accept: application/sparql-results+json`
    - (or via parameters of the SPARQL query)

Linked Data Driven Web Applications

- The easiest way is to make use of a suitable library:
  - SPARQL Javascript Library
    http://www.thefigtrees.net/lee/blog/2006/04/sparql_calendar_demo_a_sparql.html
  - ARC for SPARQL (PHP)
    http://arc.semsol.org/
  - RAP - RDF API für PHP
    http://www4.wiwiss.fu-berlin.de/bizer/rdfapi/index.html
  - Jena/ARQ (Java)
    http://jena.sourceforge.net/
  - Sesame (Java)
    http://www.openrdf.org/
  - **SPARQL Wrapper (Python)**
    http://sparql-wrapper.sourceforge.net/
  - …

07 - Linked Data Programming Example

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

Knowledge Engineering with Semantic Web Technologies

Lecture 6: Knowledge Engineering

07: Linked Data Programming Example

Dr. Harald Sack
Hasso-Plattner-Institut for IT Systems Engineering
University of Potsdam
Spring 2014

Lecture 6: Knowledge Engineering

Open HPI Course: Knowledge Engineering with Semantic Web Technologies

07 - Linked Data Programming Example

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

The **Web of Data** enables the use of Linked Data Resources on the Web

964 Datasets
82B RDF Triples
808M Links
as of April 2014

http://lod-cloud.net/versions/2011-09-19/lod-cloud_colored.html

Linked Data Driven Web Applications

- The easiest way is to make use of a suitable library:
  - **SPARQL Wrapper (Python)**
    http://sparql-wrapper.sourceforge.net/

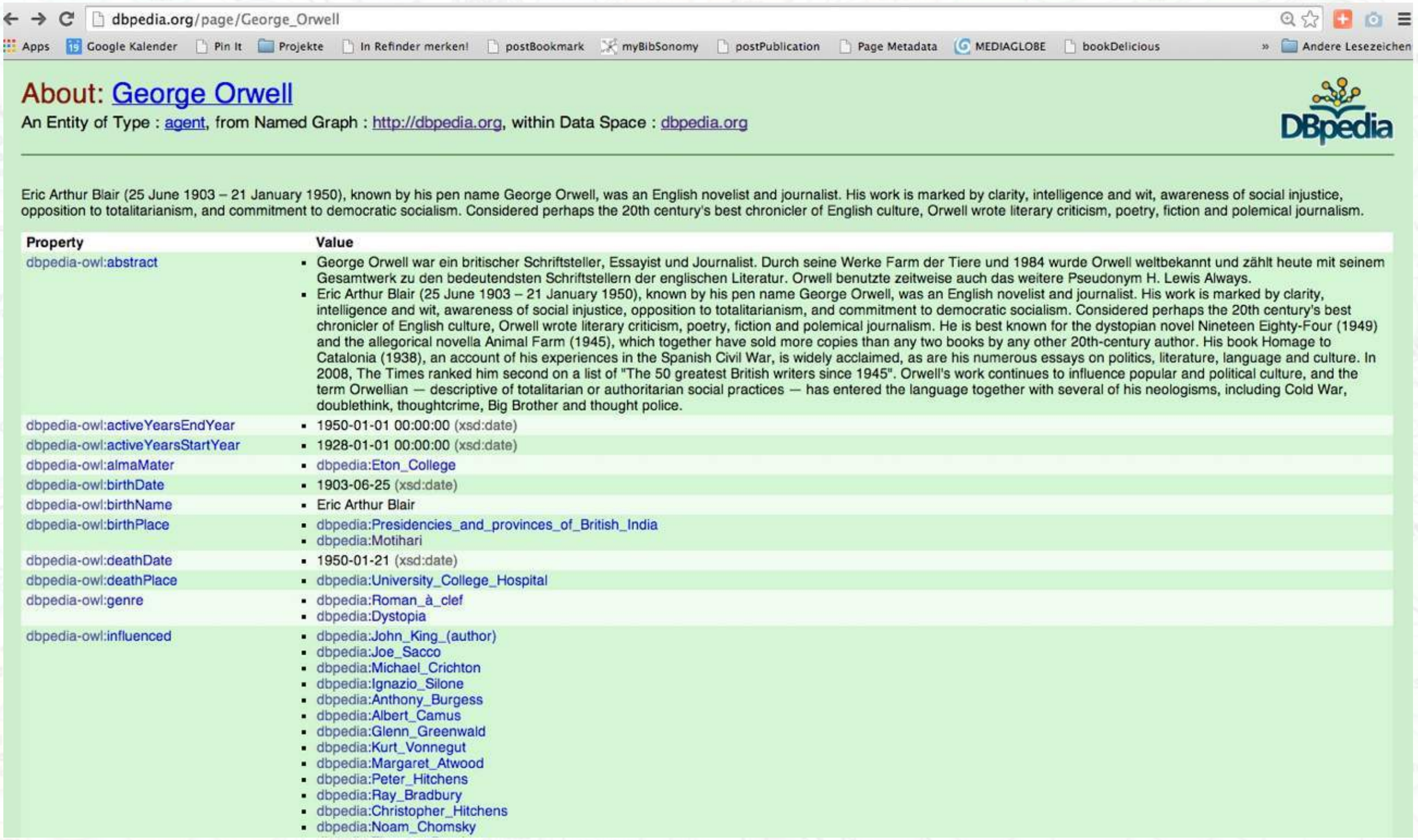
- Access to Linked Data via **SPARQL endpoints**
  - let‘s choose **DBpedia** (just for simplicity...)
    http://dbpedia.org/sparql

- ...now we have to think of a simple example...

Linked Data Programming Example

- **Example Application:**
  - Build a simple application that looks for **today‘s birthdays** of famous people, as e.g. **authors**
  - Create a **list** of authors, whose birthday is today including some **additional information**, as e.g.
    - Year of Birth
    - Short description
    - ...
  - Let‘s create a simple (local) **web page** for the task, which can be displayed in the browser
  - we use **Python** and the **SPARQL Wrapper library**

Linked Data Programming Example



Linked Data Programming Example

Virtuoso SPARQL Query Editor

Default Data Set Name (Graph IRI)

http://dbpedia.org

Query Text

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>
PREFIX dcterms: <http://purl.org/dc/terms/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>

SELECT ?person ?name ?date ?indegree ?descr
WHERE
{
  ?person rdf:type dbpedia-owl:Writer .
  ?person dbpedia-owl:birthDate ?date FILTER (SUBSTR(STR(?date),6)=SUBSTR(STR(bif:curdate('')), 6)) .
  ?person dbpedia-owl:wikiPageInLinkCount ?indegree .
  ?person rdfs:label ?name FILTER(lang(?name)="en") .
  ?person dc:description ?descr .
}
ORDER BY DESC(?indegree)
```

(Security restrictions of this server do not allow you to retrieve remote RDF data, see details.)

Results Format: HTML *(The CXML output is disabled, see details)*

Execution timeout: 30000 milliseconds *(values less than 1000 are ignored)*

Options: ☑ Strict checking of void variables

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

Run Query Reset

```python
#!/usr/bin/python
# -*- coding: utf-8 -*-

from datetime import datetime
from SPARQLWrapper import SPARQLWrapper, JSON, XML, N3, RDF

sparql = SPARQLWrapper("http://dbpedia.org/sparql")

sparql.setQuery("""
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>
PREFIX dcterms: <http://purl.org/dc/terms/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>

SELECT ?person ?name ?date ?indegree ?descr
WHERE
{
  ?person rdf:type dbpedia-owl:Writer .
  ?person dbpedia-owl:birthDate ?date FILTER (SUBSTR(STR(?date),6)=SUBSTR(STR(bif:curdate('')), 6)) .
  ?person dbpedia-owl:wikiPageInLinkCount ?indegree .
  ?person rdfs:label ?name FILTER(lang(?name)="en") .
  ?person dc:description ?descr .
}
ORDER BY DESC(?indegree)
""")

sparql.setReturnFormat(JSON)
results = sparql.query().convert()

# Create HTML output
print '<html><head><title>Literary Birthdays of Today</title></head>'

#extract Weekday %A / Month %B / Day of the Month %d by formatting today's date accordingly
datum = datetime.today().strftime("%A  %B %d")
print '<body><h1>Literary Birthdays of {}</h1>'.format(datum)

print '<ul>'

for result in results["results"]["bindings"]:
    if ("person" in result):
        #Create Wikipedia Link
        url = "http://en.wikipedia.org/wiki/" + result["person"]["value"].encode('ascii', 'ignore').split('/')[-1]
    else:
        url = 'NONE'
    if ("name" in result):
        name = result["name"]["value"].encode('ascii', 'ignore')
    else:
        name = 'NONE'
    if ("date" in result):
        date = result["date"]["value"].encode('ascii', 'ignore')
    else:
        date = 'NONE'
    if ("descr" in result):
        description = result["descr"]["value"].encode('ascii', 'ignore')
    else:
        description = ' '

    #parse date as datetime
    dt = datetime.strptime(date, '%Y-%m-%d+02:00')
    print '<li><b>{}</b> --  <a href="{}">{}</a>, {} </li>'.format(dt.year, url, name, description)

print '</ul>'
print '</body></html>'
```

Linked Data Programming Example

Literary Birthdays of Thursday June 19

- **1947** -- Salman Rushdie, British-Indian novelist and author
- **1572** -- John Donne, English poet
- **1945** -- Tobias Wolff, Fiction writer, memoirist
- **1909** -- Osamu Dazai, Japanese author
- **1928** -- Barry Took, British comedian
- **1952** -- Angie Sage, English author and illustrator
- **1921** -- Patricia Wrightson, Australian children's writer
- **1950** -- Rosie Shuster, Canadian-born comedy writer
- **1932** -- Mark Adlard, Novelist
- **1969** -- Thomas Breitling, Entrepreneur
- **1909** -- Henryka azowertwna, Polish writer
- **1872** -- Sutton E. Griggs, Novelist, clergyman, social activist
- **1988** -- Sarah Kay (poet), American poet
- **1975** -- Rekha Waheed, Novelist
- **1912** -- Luu Trong Lu, Poet
- **1969** -- Hlne Boudreau, Canadian children's author
- **1891** -- Carlos Wyld Ospina, Guatemalan writer
- **1980** -- Sadaf Fayyaz, Journalist

Linked Data Driven Web Applications

- Alternative: simple example with Jena ARQ and Java:

```
import com.hp.hpl.jena.query.*;

String service = "..."; // address of the SPARQL endpoint
String query = "SELECT ..."; // your SPARQL query
QueryExecution e = QueryExecutionFactory.sparqlService(service, query)

ResultSet results = e.execSelect();
while ( results.hasNext() ) {
        QuerySolution s = results.nextSolution();
        // ...
}

e.close();
```

- http://jena.sourceforge.net/

08 - More Applications

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

Knowledge Engineering with Semantic Web Technologies

Lecture 6: Knowledge Engineering

08: More Applications

Dr. Harald Sack
Hasso-Plattner-Institut for IT Systems Engineering
University of Potsdam
Spring 2014

Lecture 6: Knowledge Engineering

Open HPI Course: Knowledge Engineering with Semantic Web Technologies

08 - More Applications

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

How do we determine the semantics of a word?

"Now! *That* should clear up
a few things around here!"

Gary Larson's - The Far Side - http://www.thefarside.com/

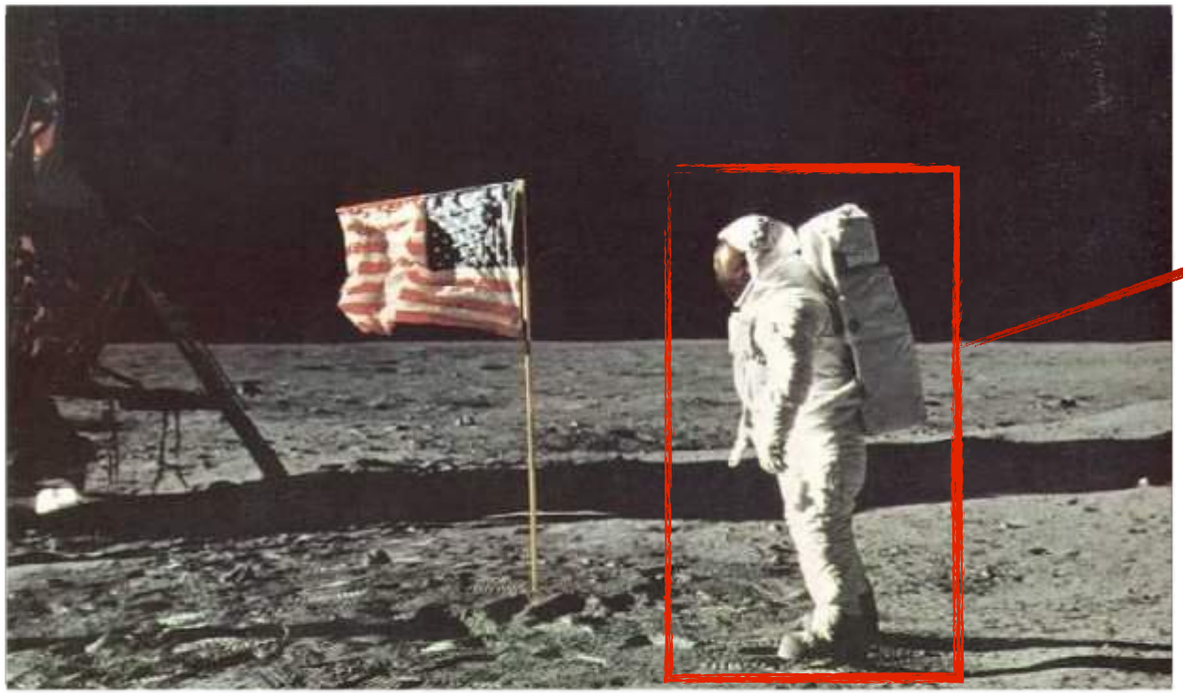
Armstrong

Armstrong

About: Neil Armstrong

An Entity of Type : NASA, from Named Graph : http://dbpedia.org, within Data Space : dbpedia.org

DBpedia



Neil Armstrong

is a

is a

Astronaut

Person

subClassOf

Science Occupation

subClassOf

Employment

Named Entity Recognition

„locating and classifying atomic elements...into predefined categories such as **names**, **persons**, **organizations**, **locations**, **expressions of time**, **quantities**, **monetary values**, etc.“

C.J.Rijsbergen, Information Retrieval (1979)

Entity Resolution

„...**identifying** and linking/grouping **different manifestations of the same real world object**“

also Disambiguation, Record Linkage, Object Identification, etc

Named Entity Resolution

Named Entity Recognition → http://dbpedia.org/ontology/Person

Class

„Armstrong was the first man on the Moon.“

Text

Named Entity Resolution → http://dbpedia.org/resource/Neil_Armstrong

Individual

How do I find the right entity?

Meaning

sender

receiver

Experience

Context

Concept

symbolizes

refers to

Experience

Symbol

stands for

Objec

Armstrong

http://commons.wikimedia.org/wiki/User:McSmit

Pragmatics

Ogden, Richards: The Meaning of Meaning:
A Study of the Influence of Language upon Thought and of the Science of Sy... (1923)

Armstrong landed the Eagle on the Moon.

Determine all possible Entity Mapping Candidates

- linguistic analysis (POS tagging)
- normalization
- encoding and spelling
- special (language dependent) characters
- language dependent spellings
- abbreviations, acronyms
- type dependent spellings
- alternative names and synonyms
- fuzzy string mapping
- ...

Armstrong landed the Eagle on the Moon.

Determine all possible Entity Mapping Candidates

DBpedia

- Armstrong, Ontario
- Anton Armstrong
- Armstrong, Ontario
- Armstrong Tools
- Ian Armstrong
- Armstrong (car)
- Armstrong, Florida
- Edward Armstrong
- Neil Armstrong
- Armstrong (moon crater)
- Armstrong County, Texas
- Gary Armstrong
- The Armstrongs
- Armstrong Tunnel
- George Armstrong
- Armstrong Bridge
- Louis Armstrong
- Craig Armstrong
- The Armstrong Twins
- Lance Armstrong

+ 400 more...

Semantic Analysis

Named Entity Resolution

Consider all entities within the same context

HPI Hasso Plattner Institut
IT Systems Engineering | Universität Potsdam

„Armstrong landed the Eagle on the Moon.“

Armstrong

448 entities

- George Armstrong Custer
- Neil Armstrong
- The Armstrong Twins
- Armstrong, Florida
- Craig Armstrong
- Armstrong, Ontario
- Armstrong (Moon Crater)
- Armstrong Gun
- Armstrong‘s Theorem
- Louis Armstrong International Airport
- Armstrong County, Texass
- Joe Armstr
- Ian Armstrong
- Armstrong Tunnel
- Armstrong Automobile
- Sir Thomas Armstrong
- Louis Armstrong
- rmstrong (British Columbia)
- Karen Armstrong
- Curtis Armstrong
- Hilary Armstrong
- Gillian Armstrong
- William L. Armstrong

Eagle

95 entities

- Eagle (Bird)
- Eagle (heraldry)
- USCGC Eagle
- The Eagle (2011 fil
- Eagle (comic)
- gle (song)
- Eagle (lunar module)
- The Eagle (newspaper)
- War Eagle
- Eagle (Moon Crater)
- The Eagle (Pub)
- Eagle TV
- Eagle Falls (Washington)
- Eagle (racehorse)
- Armstrong Tunnel
- John H. Eagle
- Eagle (typeface)
- Angela Eagle
- Linda Eagle
- James Philipp Eagle

Moon

156 entities

- Man on the Moon (film)
- Moon (song)
- Moon Son-Ri
- C Moon
- Moon 44
- The Moon (Tarot card)
- Man on the Moon (soundtrack)
- Moon
- Man on the Moon (musical)
- Mr. Moon (song)
- Moon (Band)
- Moon OS
- Moon 83
- Lottie Moon
- Edgar Moon
- Darvin Moon
- Gary Moon
- William Moon
- Francis Moon
- Robert Charles Moon
- Allan Moon
- Fly me to the Moon (song)
- Black Moon
- Ban-Ki Moon

Semantic Analysis

Named Entity Resolution

Entity Selection Process

HPI Hasso Plattner Institut
IT Systems Engineering | Universität Potsdam

Select matching entities from all possible candidate entities:

- Popularity based strategies
- Linguistical strategies
- Statistical strategies
- Semantic based strategies

- reference text corpus (wikipedia)
- link graph (wikipedia)
- semantic graph (dbpedia)

General Approach

1. Make an assumption
2. Do the strategies support or contradict your assumption
3. Make decision according to logical and probabilistic rules/constraints

N. Ludwig, H. Sack, "Named entity recognition for user-generated tags,TIR 2011

Armstrong landed the Eagle on the Moon.

Consider all entities within the same context

Article Talk | Read View source View history | Search

Louis Armstrong

From Wikipedia, the free encyclopedia

Louis Armstrong[1] (August 4, 1901 – July 6, 1971),[2] nicknamed
from New Orleans, Louisiana.

Coming to prominence in the 1920s as an "inventive" trumpet and
the focus of the music from collective improvisation to solo performa
also an influential singer, demonstrating great dexterity as an impro
purposes. He was also skilled at scat singing (vocalizing using sour

Renowned for his charismatic stage presence and voice almost as
beyond jazz music, and by the end of his career in the 1960s, he wa
general. Armstrong was one of the first truly popular African-America
his music in an America that was severely racially divided. He rarely
Americans, but took a well-publicized stand for desegregation durin
socially acceptable access to the upper echelons of American socie

Contents [hide]

1 Early life
2 Career
2.1 The All Stars
3 Death
4 Personal life
4.1 Personality
4.2 Nicknames
4.3 Armstrong and race
4.4 Religion
4.5 Personal habits
4.5.1 Purging
4.5.2 Love of food
4.6 Claimed offspring
4.7 Writings
4.8 Social organizations
5 Music
5.1 Horn playing and early jazz
5.2 Vocal popularity
5.3 Colleagues and followers
5.4 Hits and later career
5.5 Stylistic range
6 Literature, radio, films and TV
7 Awards and honors
7.1 Grammy Awards
7.2 Grammy Hall of Fame
7.3 Rock and Roll Hall of Fame
7.4 Inductions and honors
8 Legacy
8.1 House
9 Discography
10 See also
11 Notes
12 Further reading
13 External links
13.1 Video clips

Early life

Article Talk | Read Edit View history | Search

Eagle

From Wikipedia, the free encyclopedia

This article is about the bird. For other uses, see Eagle (disambiguation).

Eagle is a common name for some members of the bird family Accipitridae, and bel
related to each other. Most of the more than sixty species occur in Eurasia and Afric
two species (the Bald Eagle and Golden Eagle) in the United States and Canada,
three species in Australia.

Contents [hide]

1 Description
2 Species
3 Eagles in culture
3.1 Religion
3.2 Eagles as national symbols
4 References
5 Further reading
6 External links

Description

Eagles are large, powerfully built birds of prey, with a heavy head and beak. Even t
Common Buzzard (*Buteo buteo*) or Red-tailed Hawk (*B. jamaicensis*)), have relativ
aerodynamic feathers. Most eagles are larger than any other raptors apart from son
at 450 g (1 lb) and 40 cm (16 in). The largest species are discussed below. Like all
legs, and powerful talons. The beak is typically heavier than most other birds of pre
which enables them to spot potential prey from a very long distance.[2] This keen e
(scattering) of the incoming light. The female of all species of eagle known are larg

Eagles normally build their nests, called eyries, in tall trees or on high cliffs. Many s
The dominant chick tends to be the female, as they are bigger than the male. The p

Due to the size and power of many eagle species, they are ranked at the top of the
The *Haliaeetus* and *Ichthyophaga* eagles prefer to capture fish, though the species
kleptoparasites of other birds. The snake and serpent eagles of the genera *Circaet*
tropics of Africa and Asia. The eagles of the genus *Aquila* are often the top birds of
eagles are absent, other eagles, such as the buteonine Black-chested Buzzard-Ea
eagles, including the species-rich *Spizaetus* genus, live predominantly in woodlan
which are often unsuspectingly ambushed in such dense, knotty environments. Hu
in quite varied techniques based their environment and prey at any given time. Mos
torn apart.[7] The Bald Eagle is noted for having flown with the heaviest load verifie
However, a few eagles may target prey considerably heavier than themselves; such
a perch or nest. Golden and Crowned Eagles have killed ungulates weighing up to
predating eagle.[7][9] Authors on birds David Allen Sibley, Pete Dunne and Clay Su
this case the Bald and Golden Eagles as compared to other North American raptors

> *They have at least one singular characteristic. It has been observed that mos*
> *predation is after all a two-edged sword. All hawks seem to have this habit, f*

Among the eagles are some of the largest birds of prey; only the condors and some

Article Talk | Read Edit View history | Search

Fly Me to the Moon

From Wikipedia, the free encyclopedia

For other uses, see Fly Me to the Moon (disambiguation).

This article **needs additional citations for verification**. Please help improve this article by adding citations to reliable sources.
Unsourced material may be challenged and removed. *(February 2012)*

"**Fly Me to the Moon**" is a popular standard song written by Bart Howard in 1954. It was originally titled "**In Other Words**", and was introduced by Felicia Sanders in cabarets. The song became known popularly as "Fly Me to the Moon" from the first line of the B verse, and after a few years the publishers changed the title to that officially.[*citation needed*]

Contents [hide]

1 History of notable recordings
2 Association with space travel
3 Appearance in film, television and other media
4 References
5 External links

"Fly Me to the Moon"
Song by Frank Sinatra from the album *It Might as Well Be Swing*

Released	1964
Recorded	June 9, 1964
Genre	Traditional pop
Length	2:30
Label	Reprise
Writer	Bart Howard
Composer	Bart Howard
Producer	Sonny Burke

It Might as Well Be Swing track listing

"Fly Me to the Moon" (1)	"I Wish You Love" (2)

History of notable recordings [edit]

- Kaye Ballard recorded this song first. Her recording was released by Decca Records in 1954 on a 45 rpm record (catalog number 29114). Billboard (magazine) said that it was "A love song sung with feeling..." in a brief review published on 8 May 1954. The record included one other song "Lazy Afternoon" which Kaye Ballard was then singing in the stage musical The Golden Apple.
- Chris Connor recorded it in April 1955 for Bethlehem Records who released it on EP (catalog number BCP-127) in 1955 and on the LP "Chris" (catalog number BCP-56) in 1957.
- Portia Nelson recorded it for her album *Let Me Love You* (1956)
- Johnny Mathis also recorded it in 1956, which was the first time the title "Fly Me to the Moon" appeared on a record label.
- Eydie Gormé recorded the song in 1958 for her Grammy nominated album 'Eydie in Love' (released by ABC Paramount in September 1958 catalogue number ABC/ABCS 246).
- Felicia Sanders recorded the song in 1959. (Released by Decca Records, catalog number 30937.)
- April Stevens recorded the song in 1960 on her album Teach Me Tiger. (Imperial 12055).
- Nancy Wilson featured the song on her 1960 album Like in Love (Capitol St 1319).
- Peggy Lee recorded the song for her 1960 album Pretty Eyes released by Capitol Records in 1960.
- Peter Nero included a piano version of the song on his album "Piano Forte) released in April 1961 (RCA Victor LPM2334).
- Nat King Cole recorded the song for the album Nat King Cole Sings/George Shearing Plays (1961). (Released on Capitol Records, catalog #1675)
- Sarah Vaughan recorded the song for her 1962 album You're Mine You. This album was orchestrated and conducted by Quincy Jones.
- Mel Tormé recorded the song in 1962 at live performance released by Atlantic on the album 'Mel Tormé at the Red Hill'.
- Petula Clark recorded the song on her 1962 album *In Other Words*.
- Joe Harnell recorded an instrumental version in 1962, as "Fly Me to the Moon – Bossa Nova". This version became the biggest chart hit, reaching #14 (#4 easy listening) on the U.S. pop singles charts.
- Jazz drummer Roy Haynes, accompanied by Rahsaan Roland Kirk, Tommy Flanagan on piano, and Henry Grimes on double bass recorded the song as a brisk instrumental waltz for the album *Out of the Afternoon* (1962). (Released by Impulse! Records, AS-23).
- Dinah Washington recorded it for her album *In Love* (1962)
- Joni James included it on her setlist for *I Feel a Song Coming On* (1962)
- Jack Jones recorded the song for his 1962 album 'Gift of Love' (Kapp Records KS 3292).
- Count Basie included an instrumental version of the song, arranged by Quincy Jones, on his album "This Time by Basie - Hits Of The 50's & 60's!" released in January 1963 (Reprise Records catalog number R-6070).
- Brenda Lee performed this song on her album All Alone Am I released in February 1963.
- Connie Francis recorded it in Italian as *Portami con te* (MGM K 2078) released in April 1963. She also recorded a Spanish version *Llevame a la luna* (MGM HT 057-78) released in 1963 also.

Armstrong landed the Eagle on the Moon.

Consider all entities within the same context

Neil Armstrong

From Wikipedia, the free encyclopedia
(Redirected from Neil armstrong)

For other people named Neil Armstrong, see Neil Armstrong (disambiguation).

Neil Alden Armstrong (August 5, 1930 – August 25, 2012) was an American astronaut and the first person to walk on the Moon. He was also an aerospace engineer, naval aviator, test pilot, and university professor. Before becoming an astronaut, Armstrong was an officer in the U.S. Navy and served in the Korean War. After the war, he earn
National Advisory Committee for Aeronautics High-Speed
over 900 flights. He later completed graduate studies at the

A participant in the U.S. Air Force's Man in Space Soonest
Astronaut Corps in 1962. He made his first space flight, as
in space. On this mission, he performed the first docking of

Armstrong's second and last spaceflight was as mission co
and Buzz Aldrin descended to the lunar surface and spent
the Command Module. Along with Collins and Aldrin, Arms
Nixon; in 1978, President Jimmy Carter presented Armstro
crewmates received the Congressional Gold Medal in 200

Armstrong died in Cincinnati, Ohio, on August 25, 2012, at

Neil Armstrong

Contents [hide]
1 Early years
2 Navy service
3 College years
4 Test pilot
5 Astronaut career
5.1 Gemini program
5.1.1 *Gemini 8*
5.1.2 *Gemini 11*
5.2 Apollo program
5.2.1 Apollo 11
5.2.1.1 Voyage to the Moon
5.2.1.2 First Moon walk
5.2.1.3 Return to Earth
6 Life after Apollo
6.1 Teaching
6.2 NASA accident investigations
6.3 Business activities
6.4 Voice actor
7 Personal life
8 Illness and death
9 Legacy
10 See also
11 References
12 Bibliography
13 Further reading
14 External links

Early years

Neil Armstrong was born on August 5, 1930, in Wapakonet
Scottish, Irish, and German ancestry and had two younger

Article Talk Read Edit View history Search

Apollo Lunar Module

From Wikipedia, the free encyclopedia

The **Apollo Lunar Module** (**LM**), previously known as the **Lunar Excursion Module** (**LEM**), w
spacecraft built for the US Apollo program by Grumman to carry a crew of two from lunar orb
successfully landed on the Moon between 1969–1972.

The LM, consisting of an ascent stage and descent stage, was ferried to lunar orbit by its cor
(CSM), a separate spacecraft of approximately twice its mass, which also took the astronauts
mission, the LM was discarded. In one sense it was the world's first true spacecraft in that it v
space, structurally and aerodynamically incapable of flight through the Earth's atmosphere.

Its development was also plagued with several hurdles which delayed its first unmanned flig
manned flight by about three months. Despite this, the LM eventually became the most reliab
system, the only one never to suffer any failure that significantly impacted a mission,[1] and i
Aquarius) greatly exceeded its design requirements by maintaining life support for astronaut
Service Module.

Contents [hide]
1 Operational profile
2 History
2.1 Design phase
2.2 Astronaut training
2.3 Development flights
2.4 Production flights
2.5 Extended J Class Missions
3 Specifications
3.1 Ascent stage
3.2 Descent stage
4 Lunar Modules produced
5 Proposed derivatives
5.1 Apollo Telescope Mount
5.2 LM truck (aka Lunar Payload Module)
6 Depiction in fiction
7 Media
8 See also
9 References
10 Further reading
11 External links

Operational profile

At launch, the Lunar Module sat directly beneath the Command/Service Module (CSM) with
Adapter (SLA) attached to the S-IVB third stage of the Saturn V rocket. There it remained thro
Lunar Injection (TLI) rocket burn to send the craft toward the Moon.

Soon after TLI, the SLA opened and the CSM separated, turned around, came back to dock
from the S-IVB. During the flight to the Moon, the docking hatches were opened and the LM
up and test its systems (except for propulsion). Throughout the flight, he performed the role o
monitoring the systems of both spacecraft.

After achieving a lunar parking orbit, the Commander and LM Pilot entered and powered up
docking equipment, unfolded and locked its landing legs, and separated from the CSM, flyin
operated the flight controls and engine throttle, while the Lunar Module Pilot operated other
Commander informed on systems status and navigational information. After inspection of the
Pilot, the LM was withdrawn to a safe distance, then the descent engine was pointed forward

Article Talk Read View source View history Search

Moon

From Wikipedia, the free encyclopedia

This article is about Earth's Moon. For moons in general, see Natural satellite. For other uses, see Moon (disambiguation).

The **Moon** is the only natural satellite of the Earth,[d][7] and the fifth largest satellite in the Solar System. It is the largest natural satellite of a planet in the Solar System relative to the size of its primary,[e] having 27% the diameter and 60% the density of Earth, resulting in $^{1}/_{81}$ its mass. The Moon is the second densest satellite after Io, a satellite of Jupiter.

The Moon is in synchronous rotation with Earth, always showing the same face with its near side marked by dark volcanic maria that fill between the bright ancient crustal highlands and the prominent impact craters. It is the brightest object in the sky after the Sun, although its surface is actually very dark, with a reflectance similar to that of coal. Its prominence in the sky and its regular cycle of phases have, since ancient times, made the Moon an important cultural influence on language, calendars, art and mythology. The Moon's gravitational influence produces the ocean tides and the minute lengthening of the day. The Moon's current orbital distance, about thirty times the diameter of the Earth, causes it to appear almost the same size in the sky as the Sun, allowing it to cover the Sun nearly precisely in total solar eclipses. This matching of apparent visual size is a coincidence. The Moon's linear distance from the Earth is currently increasing at a rate of 3.82±0.07cm per year, however this rate is not constant.[8]

The Moon is thought to have formed nearly 4.5 billion years ago, not long after the Earth. Although there have been several hypotheses for its origin in the past, the current most widely accepted explanation is that the Moon formed from the debris left over after a giant impact between Earth and a Mars-sized body. The Moon is the only celestial body other than Earth on which humans have set foot. The Soviet Union's Luna programme was the first to reach the Moon with unmanned spacecraft in 1959; the United States' NASA Apollo program achieved the only manned missions to date, beginning with the first manned lunar orbiting mission by Apollo 8 in 1968, and six manned lunar landings between 1969 and 1972, with the first being Apollo 11. These missions returned over 380 kg of lunar rocks, which have been used to develop a geological understanding of the Moon's origins, the formation of its internal structure, and its subsequent history.

After the Apollo 17 mission in 1972, the Moon has been visited only by unmanned spacecraft, notably by the final Soviet Lunokhod rover. Since 2004, Japan, China, India, the United States, and the European Space Agency have each sent lunar orbiters. These spacecraft have contributed to confirming the discovery of lunar water ice in permanently shadowed craters at the poles and bound into the lunar regolith. Future manned missions to the Moon have been planned, including government as well as privately funded efforts. The Moon remains, under the Outer Space Treaty, free to all nations to explore for peaceful purposes.

Contents [hide]
1 Name and etymology
2 Formation
3 Physical characteristics
3.1 Internal structure
3.2 Surface geology
3.2.1 Volcanic features
3.2.2 Impact craters
3.2.3 Presence of water
3.3 Gravitational field
3.4 Magnetic field
3.5 Atmosphere
3.6 Seasons
4 Relationship to Earth
4.1 Orbit
4.2 Relative size
4.3 Appearance from Earth
4.4 Tidal effects
4.5 Eclipses
5 Study and exploration
5.1 Early studies

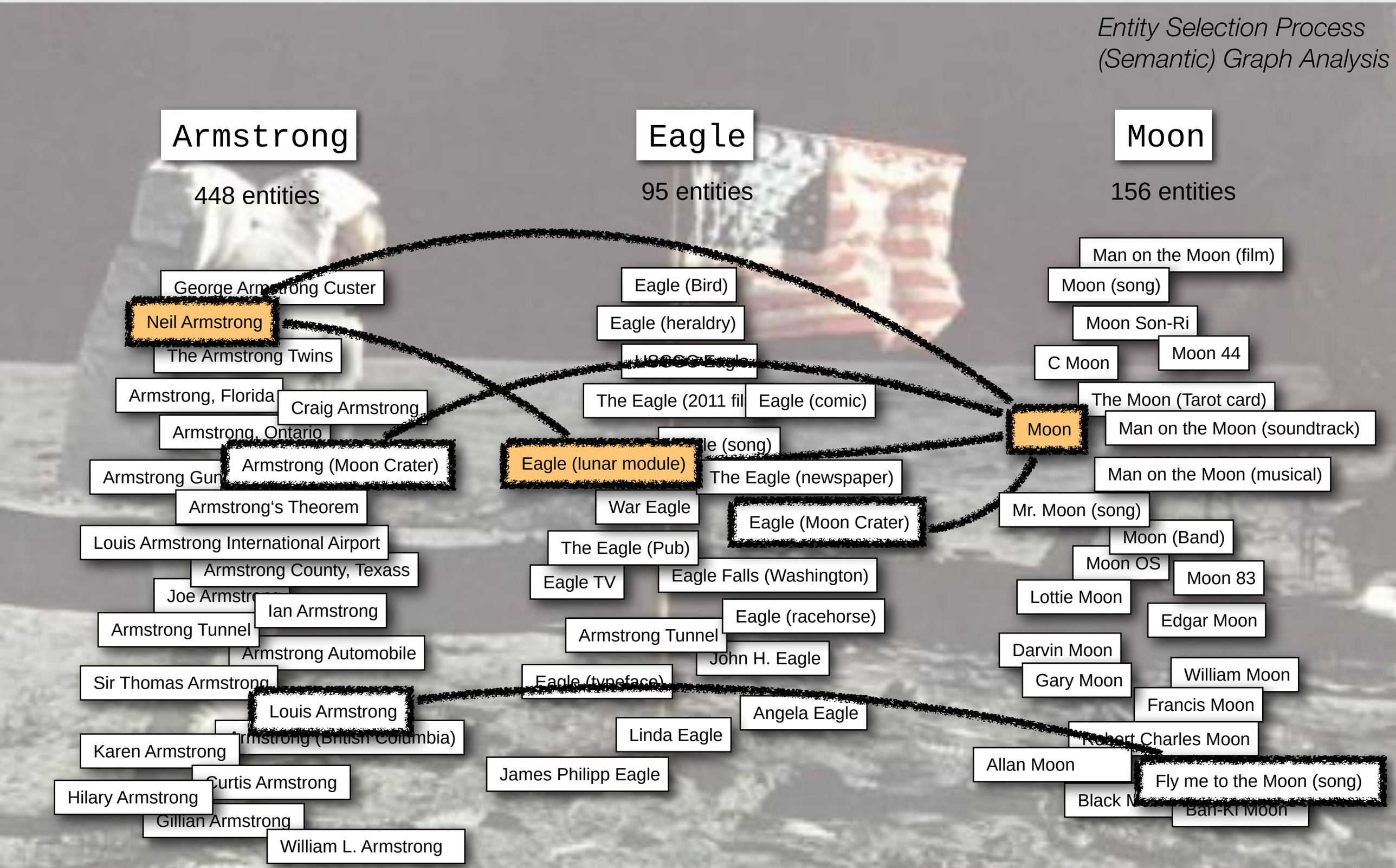
The Moon

Full moon as seen from Earth's northern hemisphere

Designations	
Adjective	lunar, selenic
Orbital characteristics	
Perigee	362,570 km (0.0024 AU)
Apogee	405,410 km (0.0027 AU)
Semi-major axis	384,399 km (0.002 57 AU)[1]
Eccentricity	0.0549[1]
Orbital period	27.321 582 d (27 d 7 h 43.1 min[1])
Synodic period	29.530 589 d (29 d 12 h 44 min 2.9 s)
Average orbital speed	1.022 km/s
Inclination	5.145° to the ecliptic[2] (between 18.29° and 28.58° to Earth's equator)[1]
Longitude of ascending node	regressing by one revolution in 18.6 years
Argument of perigee	progressing by one revolution in 8.85 years
Satellite of	Earth
Physical characteristics	
Mean radius	1,737.10 km (0.273 Earths)[1][3]
Equatorial radius	1,738.14 km (0.273 Earths)[3]
Polar radius	1,735.97 km (0.273 Earths)[3]
Flattening	0.001 25
Circumference	10,921 km (equatorial)
Surface area	3.793×10^7 km^2 (0.074 Earths)
Volume	2.1958×10^{10} km^3 (0.020 Earths)
Mass	7.3477×10^{22} kg (0.0123 Earths[1])

Armstrong landed the Eagle on the Moon.

Entity Selection Process
(Semantic) Graph Analysis



N. Steinmetz, H.Sack: Semantic Multimedia Information Retrieval Based on Contextual Descriptions, 2013

Armstrong landed the Eagle on the Moon.

http://dbpedia.org/resource/Neil_Armstrong

http://dbpedia.org/resource/Apollo_Lunar_Module

http://dbpedia.org/resource/Moon

09 - Even More Applications

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

Knowledge Engineering with Semantic Web Technologies

Lecture 6: Knowledge Engineering

09: Even More Applications

Dr. Harald Sack
Hasso-Plattner-Institut for IT Systems Engineering
University of Potsdam
Spring 2014

Lecture 6: Knowledge Engineering

Open HPI Course: Knowledge Engineering with Semantic Web Technologies

09 - Even More Applications

Open HPI - Course: Knowledge Engineering with Semantic Web Technologies
Lecture 6: Knowledge Engineering

Searching…


Google
Google Search
I'm Feeling Lucky
It's good to be found. Get your business on Google for free.

Semantic Search

Semantic metadata enable improvement of traditional keyword-based search and retrieval by

(1) **Query String Extension/Refinement**
enables more precise or more complete search results

(2) **Cross Referencing**
enables to complement search results with additional associated or similar information

(3) **Exploratory Search**
enables visualization and navigation of the search space

(4) **Reasoning**
enables to complement search results with implicitly given information

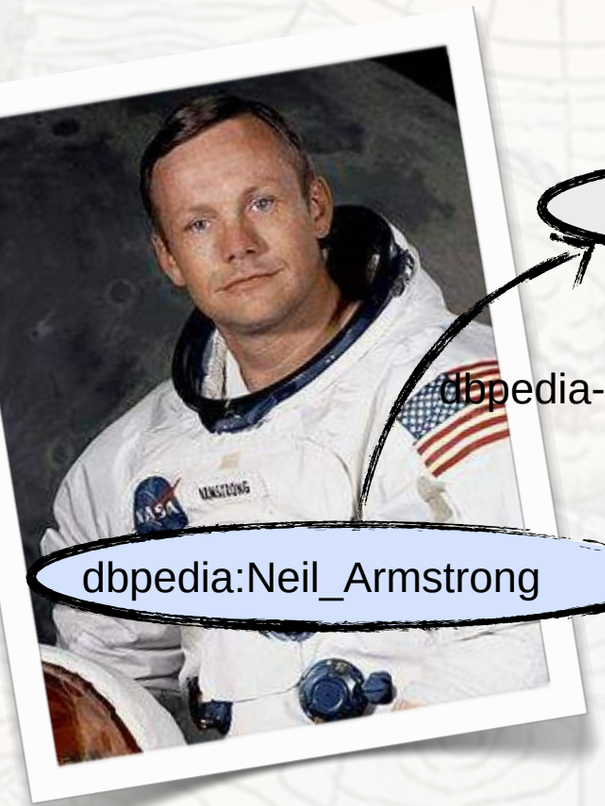
(5) etc.

Semantic Search
Cross Referencing

- Provide search results that do not literally contain the query string but are closely related to the query by content
  - Apply **domain ontologies** for determining related concepts
  - Appy **statistical analysis** of large (text) **document corpora**

Semantic Search
Exploratory Search

- Provide additional search results that do not necessarely contain the query string but are **related to the query by content** or also are **related to the search results achieved by the direct query**
  - Apply **domain ontologies** and **heuristics** to determine the relevance of facts



dcterms:subject

category:Apollo_program

dbpedia:Apollo_11

dcterms:subject

dbpedia-owl:mission

dbpedia:Apollo_13

dbpedia:Neil_Armstrong

rdf:type

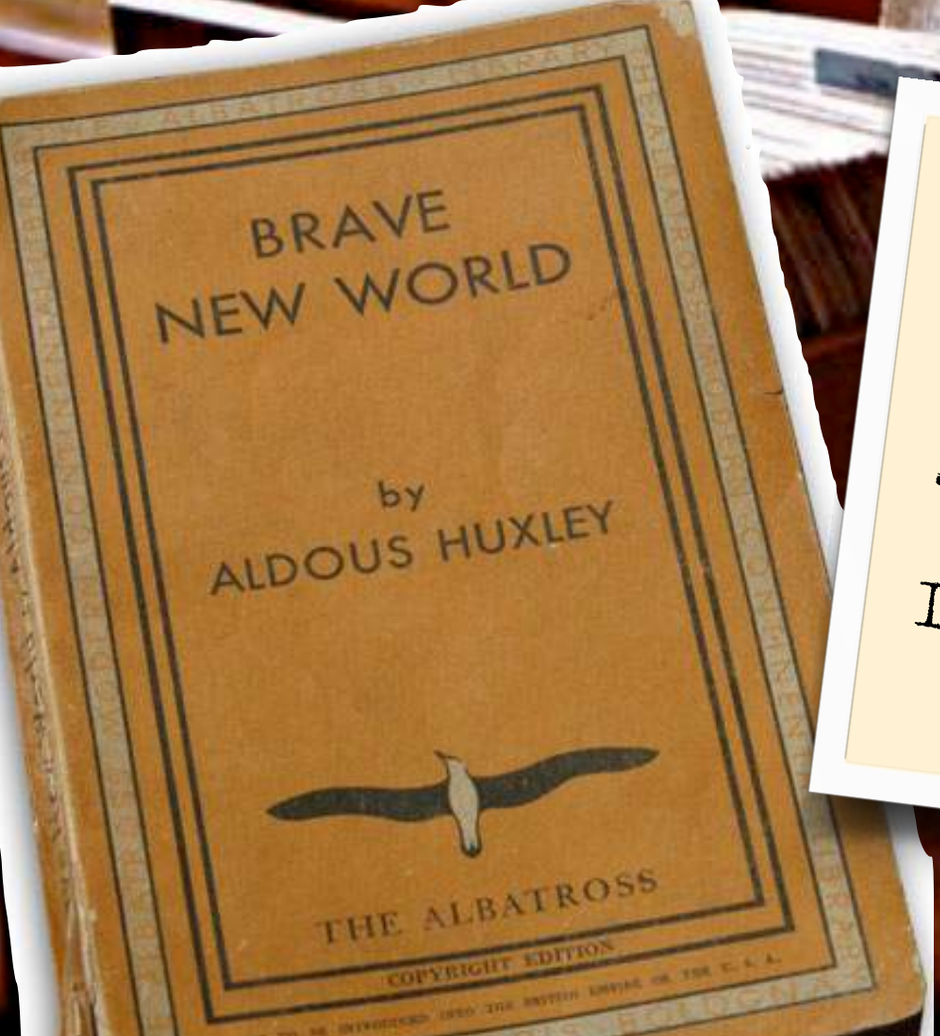
yago:Space_accidents_and_incidents

rdf:type

dbpedia:Space_Shuttle_Challenger

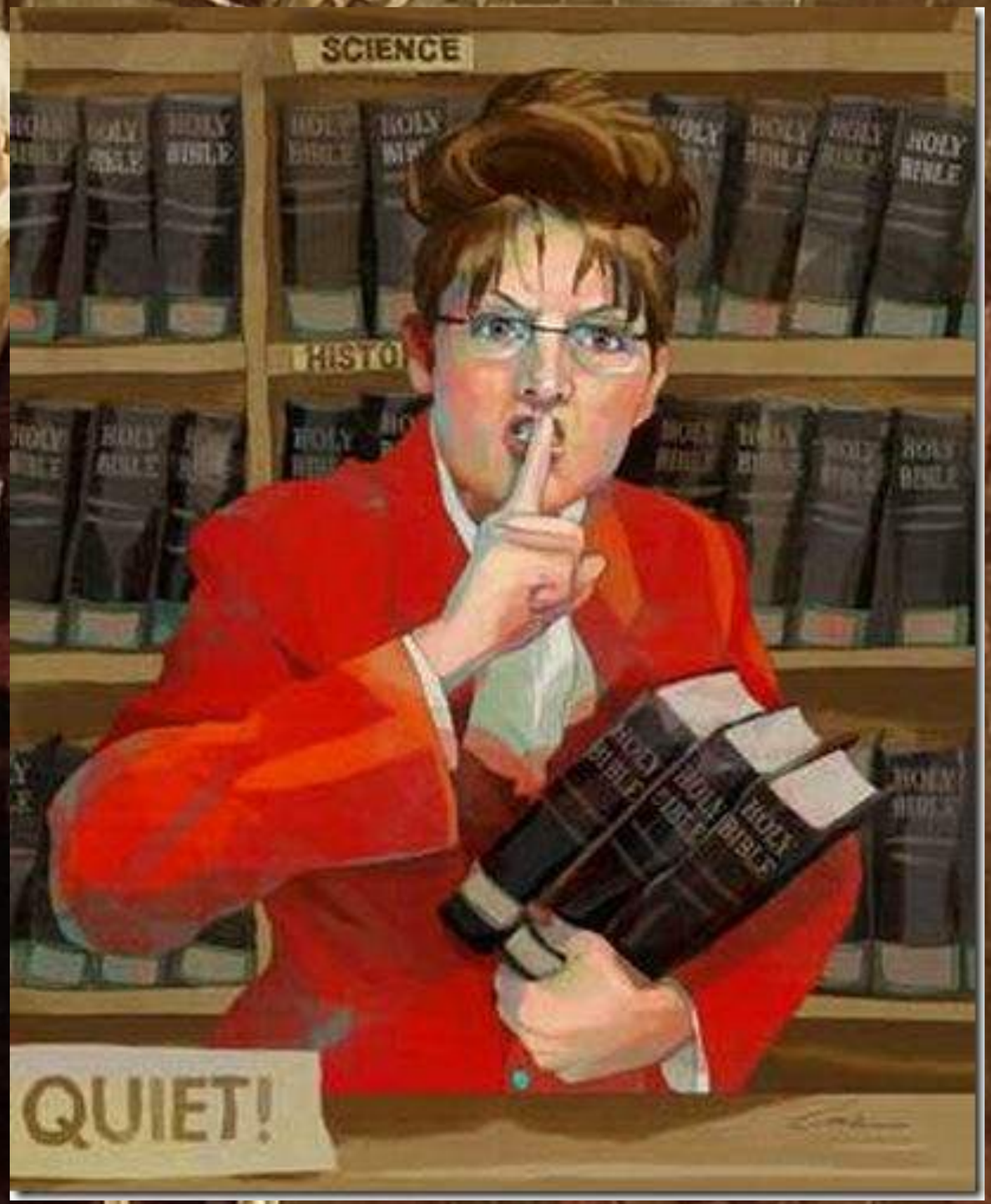
Exploratory Search

I‘m looking for the book „Brave New World“ by Aldous Huxley in the first German edition...



Brave New World. - Aldous H U X L E Y.
- The Albatros Continental Library, 47
(Hamburg usw., Albatros Verlag, 1933)
257 S. 8“

II 1, 2506, 34548



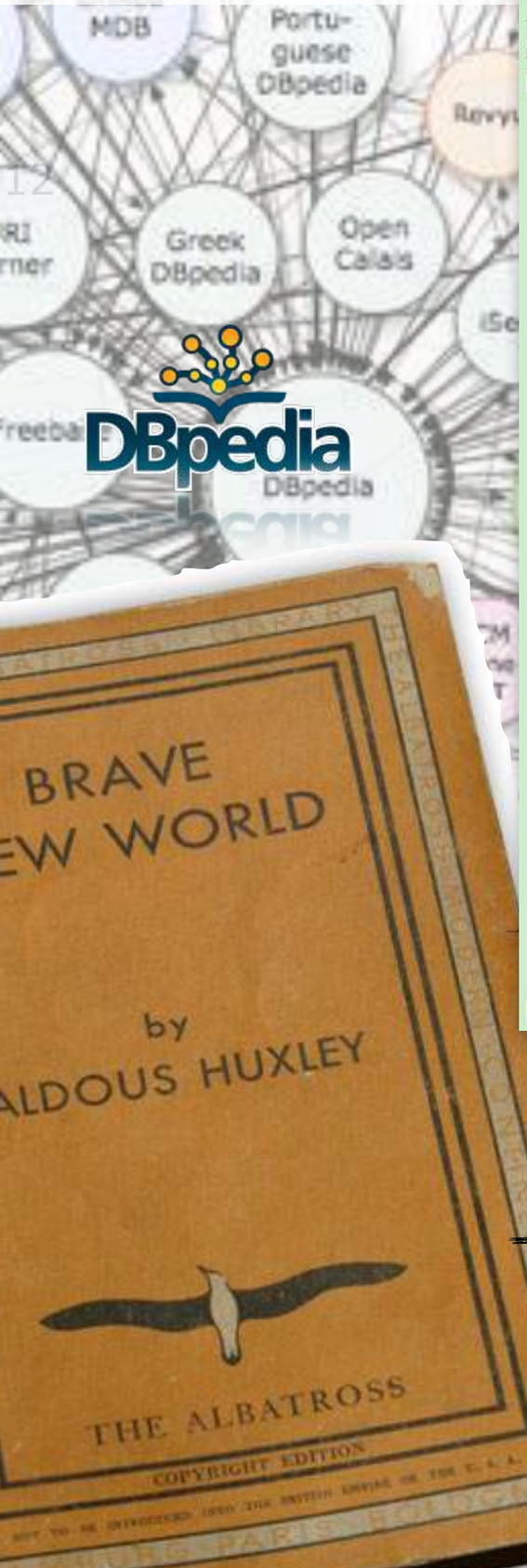
I really liked „Brave New World“ by Aldous Huxley but how should I find what to read next...?

Exploratory Search

- *What, if the user does not know,* ***which query string*** *to use?*
- *What, if the user is looking for* ***complex answers*** *?*
- *What, if the user* ***does not know the domain*** *he/she is looking for?*
- *What, if the user wants to know all(!) about a specific topic?*

- ...‚Browsing‘ instead of ‚Searching‘
- ...to find something by chance, i.e. Serendipity
- ...to get an overview
- ...enable content based navigation

Enable Exploratory Search based on Linked Open Data



About: Schöne neue Welt

An Entity of Type : work, from Named Graph : http://dbpedia.org, within Data Space : dbpedia.org

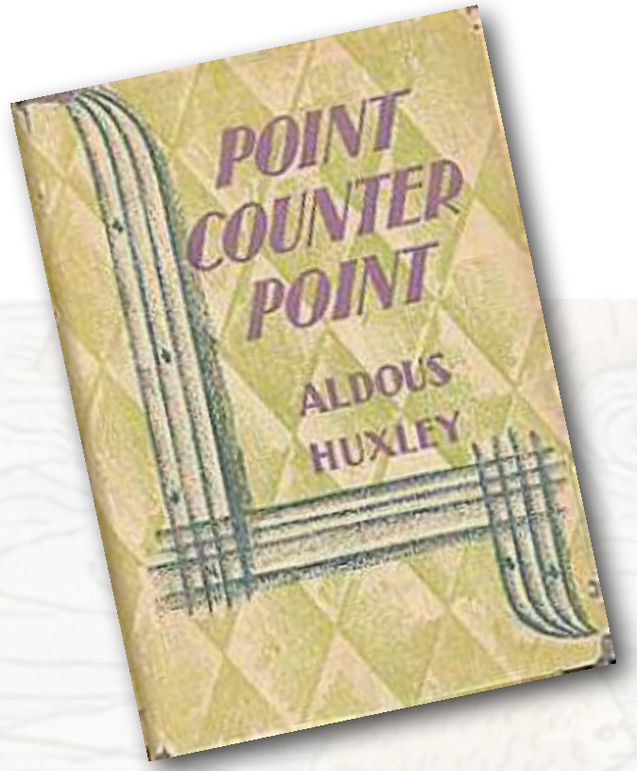
Schöne neue Welt (engl. Brave New World) ist ein 1932 erschienener dystopischer Roman von Aldous Huxley, der eine utopische Gesellschaft beschreibt, in der „Stabilität, Frieden und Freiheit“ gewährleistet scheinen.

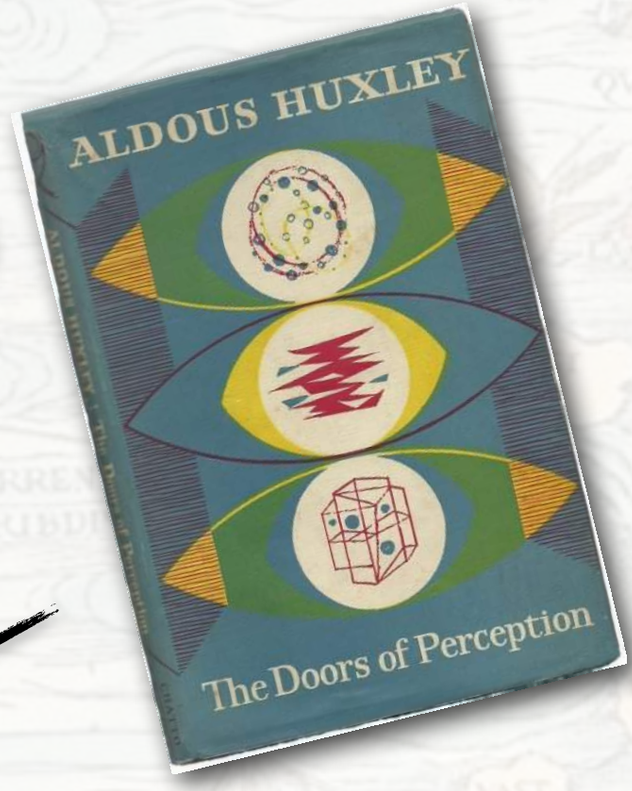
Property	Value
dbpedia-owl:abstract	▪ Brave New World is Aldous Huxley's fifth novel, written in 1931 and published in 1932. Set in London of AD 2540 (632 A.F. in the book), the novel anticipates developments in reproductive technology and sleep-learning that combine to change society. The future society is an embodiment of the ideals that form the basis of futurology. Huxley answered this book with a reassessment in an essay, Brave New World Revisited (1958), summarized below, and with his final work, a novel titled Island (1962). In 1999, the Modern Library ranked Brave New World fifth on its list of the 100 best English-language novels of the 20th century. ▪ Schöne neue Welt (engl. Brave New World) ist ein 1932 erschienener dystopischer Roman von Aldous Huxley, der eine utopische Gesellschaft beschreibt, in der „Stabilität, Frieden und Freiheit“ gewährleistet scheinen. Mittels physischer Manipulationen der Embryonen und Föten sowie der anschließenden mentalen Indoktrinierung der Kleinkinder werden die Menschen gemäß der jeweiligen gesellschaftlichen Kasten geprägt, denen sie angehören sollen und die von Alpha-Plus (für Führungspositionen) bis zu Epsilon-Minus (für einfachste Tätigkeiten) reichen. Allen Kasten gemeinsam ist die Konditionierung auf eine permanente Befriedigung durch Konsum, Sex und die Droge Soma, sodass den Mitgliedern dieser Gesellschaft das Bedürfnis zum kritischen Denken und Hinterfragen ihrer Weltordnung genommen wird. Die Regierung jener Welt bilden Kontrolleure, Alpha-Plus-Menschen, die von der Bevölkerung wie Idole verehrt werden. ▪ Un mundo feliz (Brave new world en inglés, literalmente «mundo nuevo y espléndido») es la novela más famosa del escritor británico Aldous Huxley, publicada por primera vez en 1932. El título tiene origen en una obra del autor William Shakespeare, La tempestad, en el acto V, cuando Miranda pronuncia su discurso. ▪ Uljas uusi maailma on Aldous Huxleyn vuonna 1932 julkaistu tieteisromaani. Se on George Orwellin romaanin Vuonna 1984 ohella yksi 1900-luvun merkittävimpiä tulevaisuusfiktioita. Se ennusti pyrkimyksen luoda parempi maailma muokkaamalla ihmisistä täydellisempiä geenitekniikan avulla. Uljas uusi maailma sai vaikutteita muun muassa filosofi Bertrand Russellin vuotta aiemmin julkaisemasta kirjasta The Scientific Outlook. Huxleyn maailma kuvaa dystopiaa noin viisisataa vuotta meidän ajastamme eteenpäin. Hänen maailmansa on totalitaristinen yhtenäisvaltio, jossa ihmisen tuotteistaminen on viety loogiseen loppuunsa saakka. Hänen luomassaan maailmassa lapset syntyvät tuotantolaitoksissa, joissa soveliaimpia geeniyhdistelmiä tuotetaan useita kappaleita tarkoin valvotuissa olosuhteissa. Ihmisten kohtalo on predestinoitu sen mukaan, mihin tuotantosarjaan (yhteiskuntaluokkaan) hän sattuu kuulumaan; tuotteistetut ihmiset jaetaan alfoihin, beetoihin, gammoihin, deltoihin ja epsiloneihin. Kulttuuri on täysin banalisoitu, taide ja korkeakulttuuri hävitetty ja historia muistomerkkeineen pyyhitty pois ihmisten mielistä. Siinä, missä Orwellin dystopia viittaa suoraan uhkaan ja totalitaristiseen ideologiaan vallankumouksen kautta, Huxleyn dystopiassa uhka on epäsuora ja viittaa totalitaristiseen ideologiaan konsumerismin ja monopolikapitalismin kautta. Uljas uusi maailma on muutenkin hyvin erilainen. Sivistyneet ihmiset hukuttautuvat keinotekoisiin nautintoihin ja vapaaseen seksiin haluten aina vain lisää. Koska ihmisten valmistaminen on tuotteistettu, perheen käsitys on hävitetty ja perhekäsitteistä (isä, äiti, veli, sisko jne) on tullut kirosanoja. Uskontoa ei enää ole siinä muodossa kuin me olemme tottuneet sen tulkitsemaan. Ihmisille esikuvaksi on noussut Henry Ford, jota he palvovat kuin puolijumalaa. Kristinuskon risti on korvattu Fordin T-mallin mukaan T-merkillä. ▪ Il mondo nuovo (Brave New World) è un romanzo di fantascienza di genere distopico scritto nel 1932 da Aldous Huxley. È il suo romanzo più famoso e ne sono stati tratti alcuni adattamenti cinematografici e televisivi. Il libro anticipa temi quali lo sviluppo delle tecnologie della riproduzione, l'eugenetica e il controllo mentale, usati per forgiare un nuovo modello di società. Il mondo che vi è descritto potrebbe essere un'auspicabile utopia ovvero un drammatico limbo esistenziale. Nei fatti il ritratto che ne fa l'autore è distaccato, sebbene a volta traspaia velatamente una cinica esaltazione degli aspetti grotteschi delle varie vicende; più spesso il narratore appare tuttavia interiormente rattristato dalla scena da lui descritta, ed è forse proprio questo il motivo per cui a volte indugia forzatamente negli improvvisi aspetti grotteschi del dramma. Il titolo originale si rifà alle parole pronunciate da Miranda ne La tempesta di William Shakespeare: « How beauteous mankind is! O brave new world that has such people in't! » « Com'è bello il genere umano! Oh mirabile e ignoto mondo che possiedi abitanti così piacevoli! » ▪ 『すばらしい新世界』（すばらしいしんせかい）は、オルダス・ハクスリーが1932年に発表したディストピア小説。原題は "Template:Lang"『ブレイヴ・ニュー・ワールド』。 ▪ Brave New World is een dystopische sciencefiction-roman uit 1932 van de Britse schrijver Aldous Huxley. Het boek is in het Nederlands vertaald als Het soma-paradijs en als Heerlijke nieuwe wereld. De titel is ontleend aan de woorden die Miranda spreekt in William Shakespeares The Tempest (De storm). In de eerste scène van de laatste akte zegt zij: O wonder! How many goodly creatures are there here! How beauteous mankind is! O brave new world, That has such people in it! Deze woorden worden in de roman aangehaald door John Savage (de Wilde) op het moment dat hij nog meent in een prachtige wereld te zijn beland. ▪ Vidunderlige nye verden er en dystopisk roman skrevet av Aldous Huxley. Romanen ble først utgitt i 1932 og ble Aldous Huxleys best kjente roman. Historien finner sted i London i det 26. århundre. Den forutser en utvikling innen reproduksjonsteknologi, bioteknologi, og "sleep-learning" (å lære mens en sover), som sammen bidrar til å forandre et samfunn. Boken beskriver en verden som også kan kalles en ironisk utopia: Menneskeheten er uten bekymringer, den er sunn og på et høyt teknologisk plan. Krig og fattigdom har blitt fjernet og alle er permanent lykkelige. Ironien ligger i at dette har blitt oppnådd ved å eliminere mye av det mennesker for øyeblikket får lykke fra: Familie, kunst, litteratur, religion, filosofi, vitenskap og kulturelt mangfold. Boken beskriver også et hedonistisk samfunn, der glede og nytelse tilfredsstilles av promiskuøs sex og stoffmisbruk - her spesielt bruken av "soma", en kraftig stimulant som gir fantasier i form av hallusinasjon for å komme vekk fra smerte og dårlige følelser. Boken, som var inspirert av H.G. Wells utopiske roman Men Like Gods, var Aldous Huxleys første forsøk på en dystopi. Siden dens utgivelse i 1932 har den hatt en bemerkelsesverdig innflytelse på den moderne verden. Tittelen kommer fra Mirandas tale i Shakespeares Stormen, 5. akt, scene I: "O wonder! How many goodly creatures are there here! How beautious mankind is! O brave new world, That has such people in't!" ▪ Nowy wspaniały świat (ang. Brave New World) – powieść Aldousa Huxleya napisana w 1932 roku. Jest to antyutopia, klasyfikowana do gatunku science-fiction. Kanon fantastyki posłużył w niej Huxleyowi jako narzędzie do konfrontacji idei wolności z porządkiem, pragmatyzmu z fantazją, indywidualizmu z kulturą społeczności masowej. Powieść rozpoczyna się od wykładu w "Ośrodku Rozrodu i Warunkowania w Londynie Centralnym", podczas którego grupa londyńskich studentów zapoznaje się z zasadami funkcjonowania Republiki Świata. Idee utopii ujęte zostały w trzech słowach przewodnich Republiki: Wspólność - czego wyrazem jest zasada "każdy jest dla każdego". Identyczność - upodobań i poglądów, a w niższych warstwach klasowego społeczeństwa Republiki także wyglądu, będąca efektem klonowania. Stabilność - wypływająca z zachowania wspólności i identyczności, osiągana ustawiczną kontrolą i tępieniem zachowań odbiegających od ustalonych norm. Aby powyższe zasady wcielić w życie stosowano różne metody, najważniejszą z nich było warunkowanie genetyczne i psychologiczne całego społeczeństwa. Zgodnie z twierdzeniem: "Wszelkie warunkowanie zmierza do jednej rzeczy: do sprawienia, by ludzie polubili swe nieuniknione przeznaczenie społeczne". Na obrzeżach Nowego Świata znajdował się Rezerwat, w którym ludzie żyli według starych, "naturalnych" zasad, nie przyjmując nowego porządku. Jeden z bohaterów, John (Dzikus), wychowując się tam poznał ideę Boga, potrzebę indywidualności, prawo do przeżywania cierpienia i smak wolności myśli. Kiedy został sprowadzony do Londynu, odkrywał stopniowo, że ten cudowny świat nie jest taki, jak tego oczekiwał. Porównanie świata szekspirowskich namiętności pełnego tragedii, miłości i cierpienia, piękna i brzydoty ze światem uporządkowanym od chwili narodzin aż do śmierci, wypadło na niekorzyść dla nowego, wspaniałego świata, doprowadziło Dzikiego do ucieczki w samotność i pokutę. Huxley stworzył powieść o charakterze proroctwa i przestrogi, którą próbuje ostrzec ludzkość przed utopią przedkładającą trwałą szczęśliwość i wygodę nad wzloty i upadki w świecie pełnym emocji. Krytykowane wartości dostrzegał u podstaw ery przemysłowej. Stworzył zatem Republikę, w której rozwinął i przerysował je ukazując ich mechanistyczną naturę. Stąd w Republice niechęć do wszelkich przejawów starości, włączając w to starą literaturę, przedmioty codziennego użytku, idee i religię. Raz zepsutych rzeczy nie naprawiano, gdyż byłoby to działanie aspołeczne. Z tych samych powodów nieaprobowane były także gry i sport nie wymagające skomplikowanego sprzętu. Tworzono doskonałych konsumentów o ustabilizowanych potrzebach. Krytyka cywilizacji przemysłowej sięga dalej - Nowy Świat hoduje idealnych robotników. Warunkowani psychicznie i fizycznie do pełnienia określonej roli w społeczeństwie nie mają wygórowanych wymagań, są zadowoleni ze swojej pracy. Aby nie mieli wyższych potrzeb celowo obniża się ich poziom umysłowy dostosowując go do minimalnych wymagań mechanicznej, odtwórczej pracy. Stanowią idealną załogę: nie strajkują, nie żądają podwyżki, nie intrygują w celu uzyskania awansu. Kolejna dawka somy, narkotyku, na którego używanie wszyscy byli warunkowani, uśmierza ewentualne niepokoje, zgodnie z maksymą: "Gdy jednostka czuje, wspólnota szwankuje". Huxley położył na szalę wyboru wygodną szczęśliwość i wybitną twórczość, tę drugą kosztem spokoju zarówno duchowego jak i społecznego jednostki. Z ideologii Republiki wyłania się niejako definicja ludzkiego szczęścia: "szczęśliwi ludzie to tacy, którzy nie są świadomi lepszych i większych możliwości, żyją we własnych światach odpowiednio skrojonych do ich predyspozycji". ▪ Admirável Mundo Novo (Brave New World na versão original em língua inglesa) é um livro escrito por Aldous Huxley e publicado em 1932 que narra um hipotético futuro onde as pessoas são pré-condicionadas biologicamente e condicionadas psicologicamente a viverem em harmonia com as leis e regras sociais, dentro de uma sociedade organizada por castas. A sociedade desse "futuro" criado por Huxley não possui a ética religiosa e valores morais que regem a sociedade atual. Qualquer dúvida e insegurança dos cidadãos era dissipada com o consumo da droga sem efeitos colaterais aparentes chamada "soma". As crianças têm educação sexual desde os mais tenros anos da vida. O conceito de família também não existe. ▪ «О дивный новый мир» («Прекрасный новый мир», англ. Brave New World) — антиутопический, сатирический роман английского писателя Олдоса Хаксли (1932). В заглавие вынесена строчка из трагикомедии Уильяма Шекспира «Буря». ▪ Brave New World omdirigerar hit. För albumet, se Brave New World (album). Du sköna nya värld är en berömd framtidsroman skriven 1932 av Aldous Huxley. Titeln är ett citat från Shakespeares pjäs Stormen. Boken utspelar sig i en avlägsen framtid, där tideräkningen utgår från Henry Ford. Barnen skapas på mer eller mindre industriell väg och anpassas under fosterstadiet på kemisk väg till sin plats i kastsamhället. Persontyperna Alfa och Beta har till uppgift att sköta de höga och mellanhöga positionerna i samhället, medan Gamma, Delta och Epsilon ska vara arbetare utan förmåga att tänka själva. Denna uppdelning förstärks genom den indoktrinering och propaganda som genomsyrar hela samhället, även medborgarnas sömn. Huvudpersonerna är Bernard Marx, Helmholtz Watson och vilden som är modersfödd till skillnad från de två förstnämnda. Vilden och hans mor Linda tas in i civilationen av Bernard. För att hålla människorna på plats tilldelas de dagligen en ranson av den ultimata drogen Soma. ▪ 《美麗新世界》（Brave New World），亦名「美妙的新世界」、「勇敢面對新世界」。為英國作家阿道斯·雷歇爾德·赫胥黎於1931年創作1932年發表的反烏托邦作品。與《一九八四》和《我們》並列為世界三大反烏托邦小說。書名靈感得自莎士比亞的《暴風雨》中，米蘭達的對白：「人類有多麼美！啊！美麗的新世界，有這樣的人在裡頭！（O brave new world, that has such people in it.）」。小說標題：「美麗新世界」是在諷刺新世界的好處。雖然外表似美，其實科技並沒有在社會的總體精神進步，反而讓小說中的社會文化倒退。作者以豐富的文辭和形容詞，做散文章不同的情界藉以突顯書所要描寫的美麗世界。在書中開頭的第一段，全都使用大寫字母，此寫作手法獨特，不但押韻，也給人震撼的感覺。 ▪ Le Meilleur des mondes (en anglais, Brave New World) est un roman d'anticipation dystopique, écrit en 1931 par Aldous Huxley. Il paraît en 1932. Modèle:Référence nécessaire Vingt-cinq ans plus tard, Huxley publie un essai dédié à ce livre, Retour au meilleur des mondes. Le titre original du roman, Brave New World, provient de La Tempête de William Shakespeare, acte 5 scène 1. Le titre français, Le Meilleur des mondes, est tiré d'une phrase en français présente au début de la version originale anglaise du livre de Huxley, empruntée à Candide de Voltaire. Le roman a été adapté au cinéma en 1998. Même s'il reprend beaucoup d'éléments, le film présente une histoire assez différente.
dbpedia-owl:author	▪ dbpedia:Aldous_Huxley
dbpedia-owl:isbn	▪ ISBN 0-06-080983-3 (Paperback edition)

http://dbpedia.org/page/Brave_New_World

Gather knowledge about dbpedia:Brave_New_World and decide, which interesting fact to follow....

Explorative Search

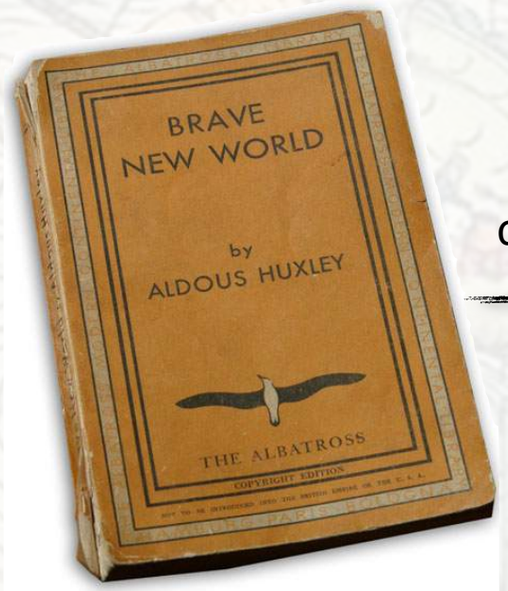




DBpedia

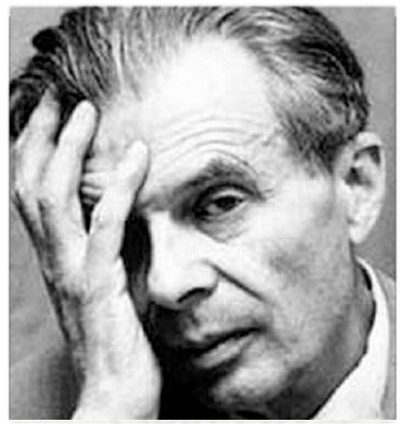
dbpedia-owl:author

dbpedia-owl:author



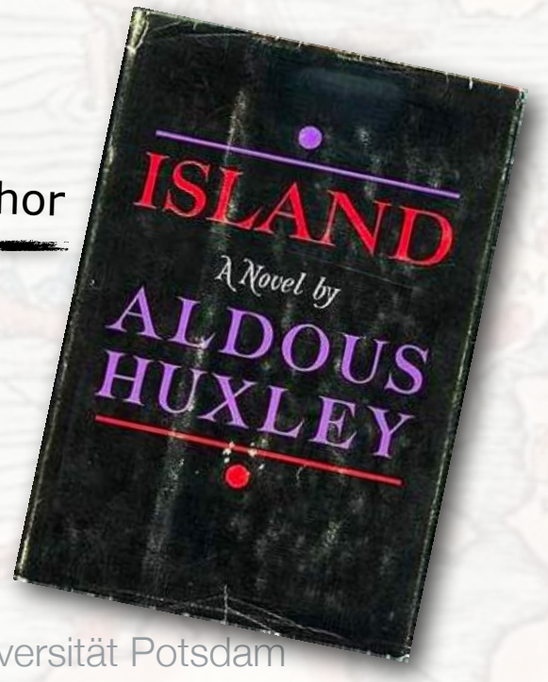
dbpedia:Brave_New_World

dbpedia-owl:author



dbpedia:Aldous_Huxley

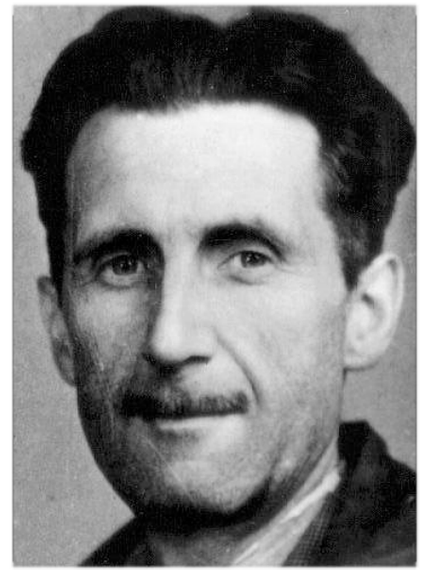
dbpedia-owl:author



Explorative Search

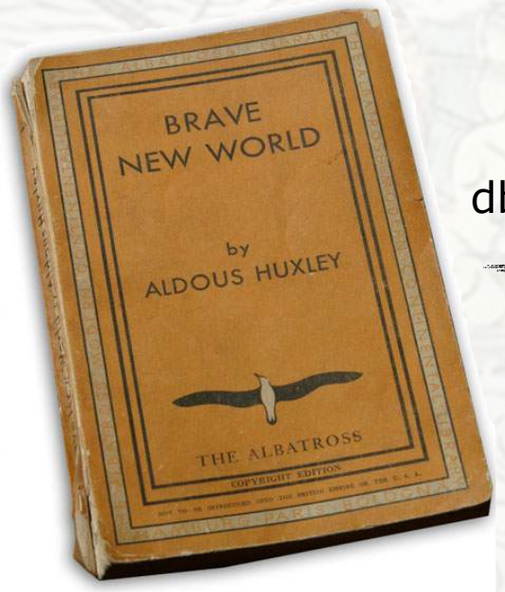
dbpedia:H._G._Wells

dbpedia:George_Orwell

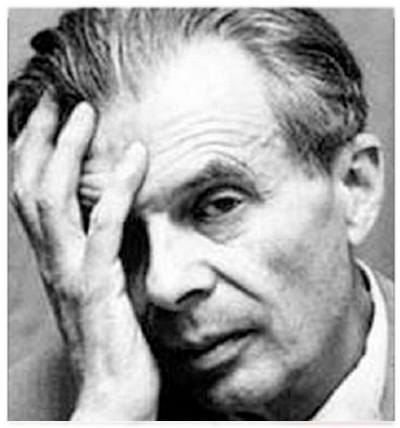


dbpedia:ontology/influences

dbpedia:ontology/influences



dbpedia-owl:author



dbpedia:Aldous_Huxley

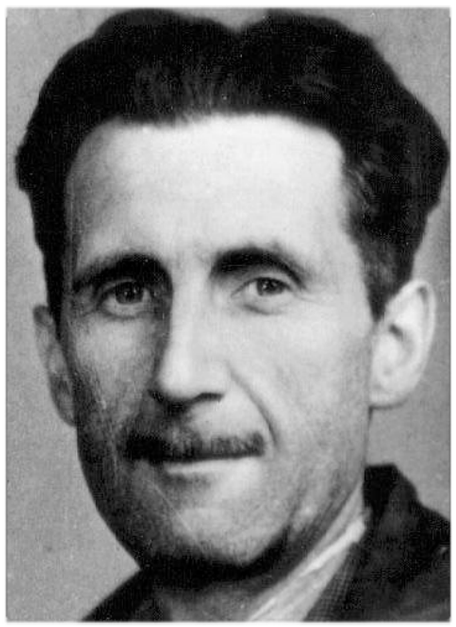
dbpedia:ontology/influences

dbpedia:Brave_New_World

dbpedia:Michel_Houellebecq

dbpedia:H._G._Wells

dbpedia:George_Orwell

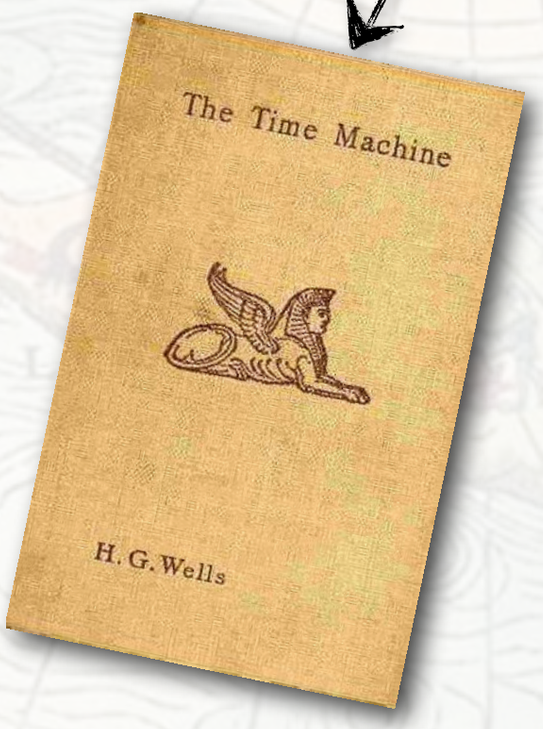


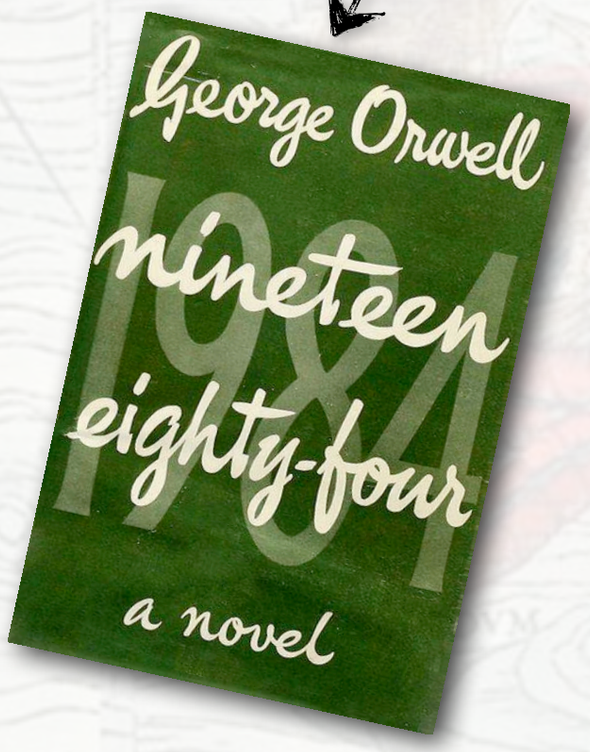
dbpedia:Michel_Houellebecq

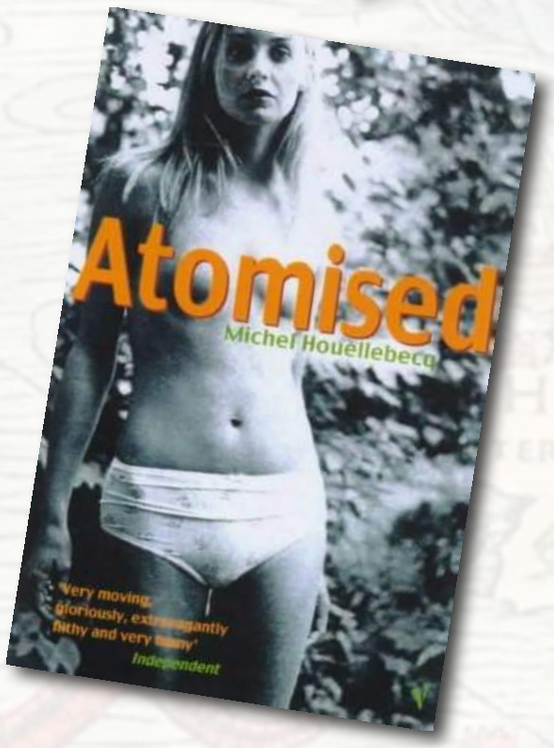
dbpedia-owl:notableWork

dbpedia-owl:notableWork

dbpedia-owl:notableWork





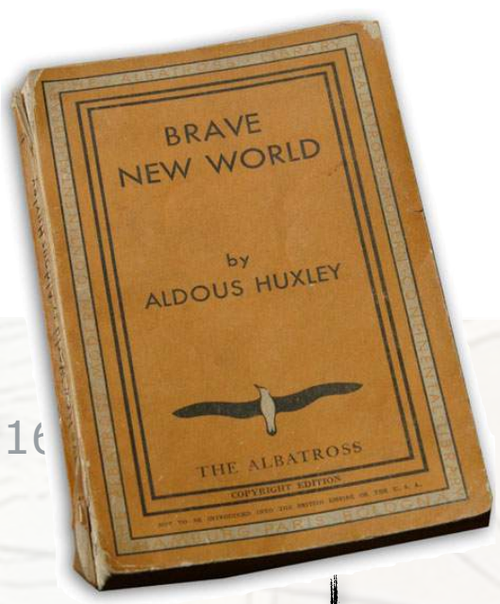


dbpedia:The_Time_Machine

dbpedia:Nineteen_Eighty-Four

dbpedia:Les_Particules_élémentaires

...and now please surprise me



Yago:EnglishExpatriatesInTheUnitedStates

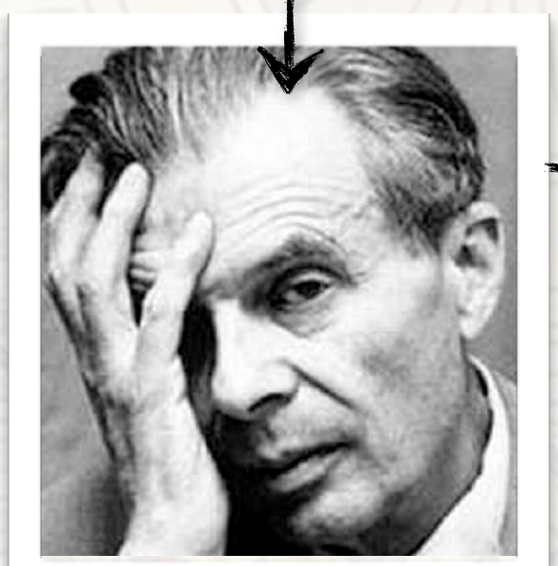
dbpedia-owl:author

rdf:type

rdf:type

rdf:type

dbpedia:Tim_Berners-Lee



dbpedia:Aldous_Huxley

dbpedia:Patrick_Stewart

dbpedia-owl:starring

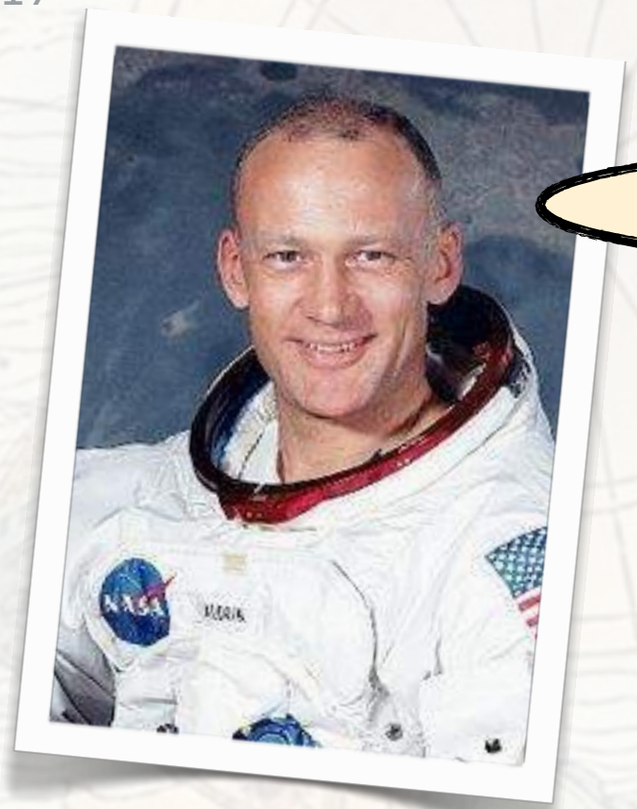
dbpprop:inventor

dbpedia:Star_Trek:_The_Next_Generation

dbpedia:World_Wide_Web

Exploratory Search and Serendipity

- Find something unexpected that you were not looking for on purpose ...



dbpedia:Buzz_Aldrin

dbpedia:Cookie_Monster

dbpedia:Strictly_Come_Dancing

dbpedia:Transformers

Knowledge Engineering with Semantic Web Technologies

Lecture 6: Knowledge Engineering