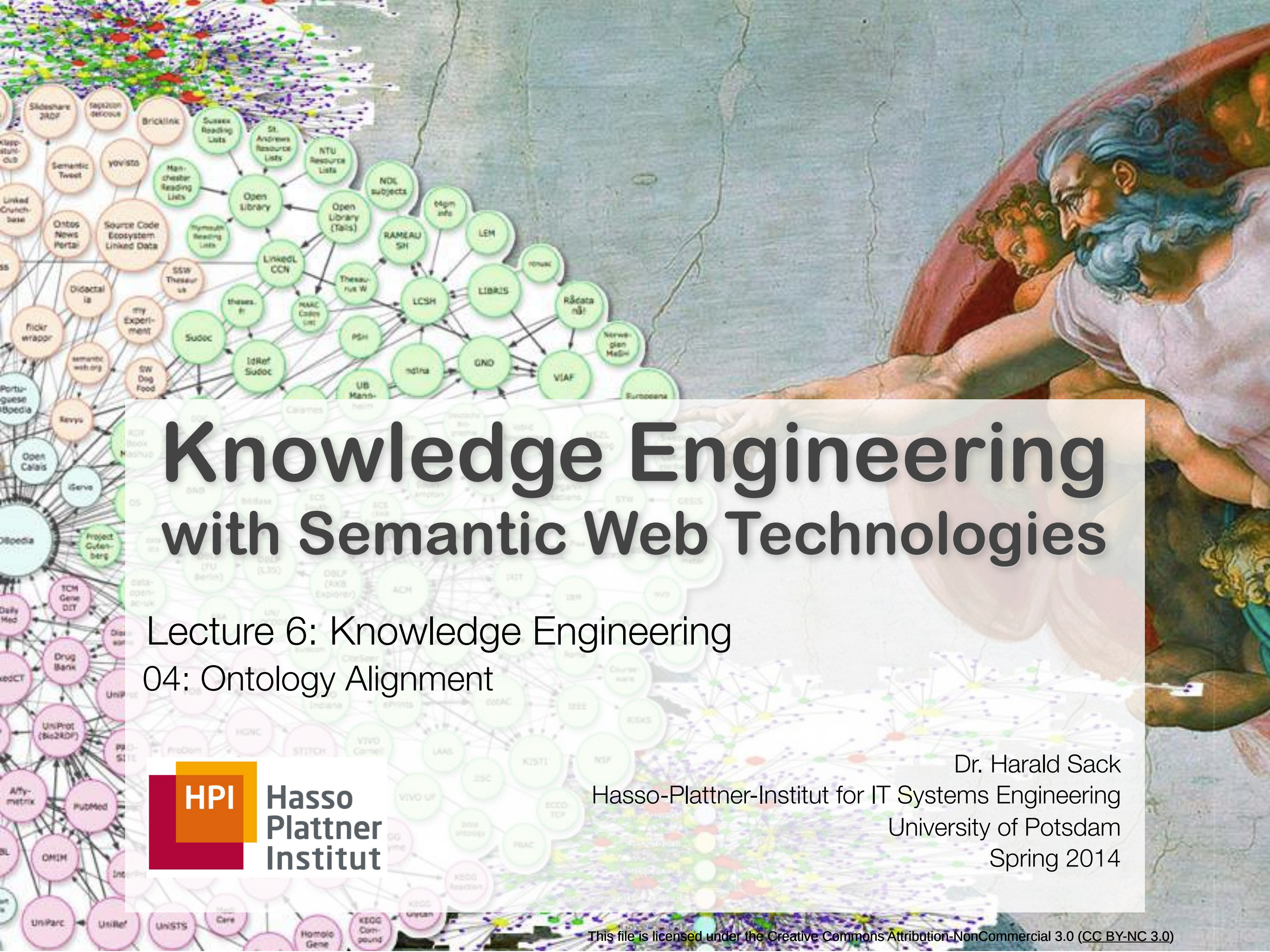




Knowledge Engineering with Semantic Web Technologies

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

- **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

6 • **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)

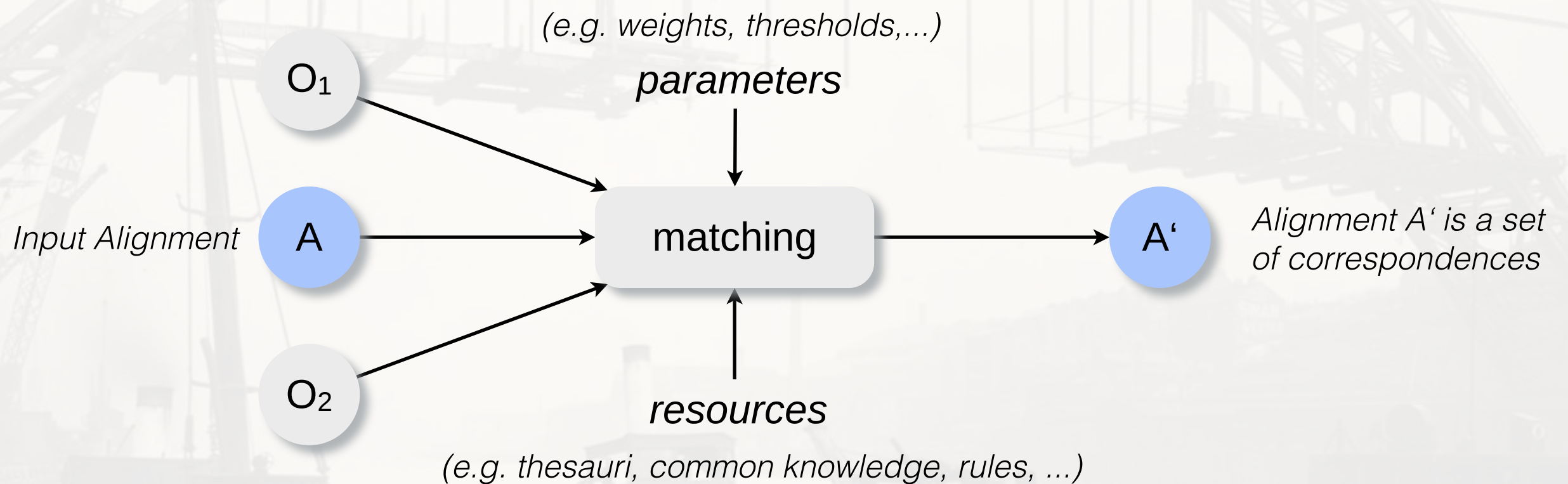
Ontology Alignment



“Basically, we’re all trying to say the same thing.”

Author: Gahan Wilson
Source: Conde Nast

- 8
- **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 respectively O_2 , is defined as

$\langle id, e_1, e_2, r, n \rangle$

- with
 - **id** ... a unique **identifier** of the correspondence
 - **r** ... a **relation**, as e.g. equivalence ($=$), more general ($\supseteq, \geq$), less general ($\sqsubseteq, \leq$), disjointness ($\perp$), 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 **e₁** and **e₂** 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**:

- $$\begin{aligned} \text{speed} &= \text{velocity} \times 2.237 \\ 0.477 \times \text{speed} &= \text{velocity} \end{aligned}$$

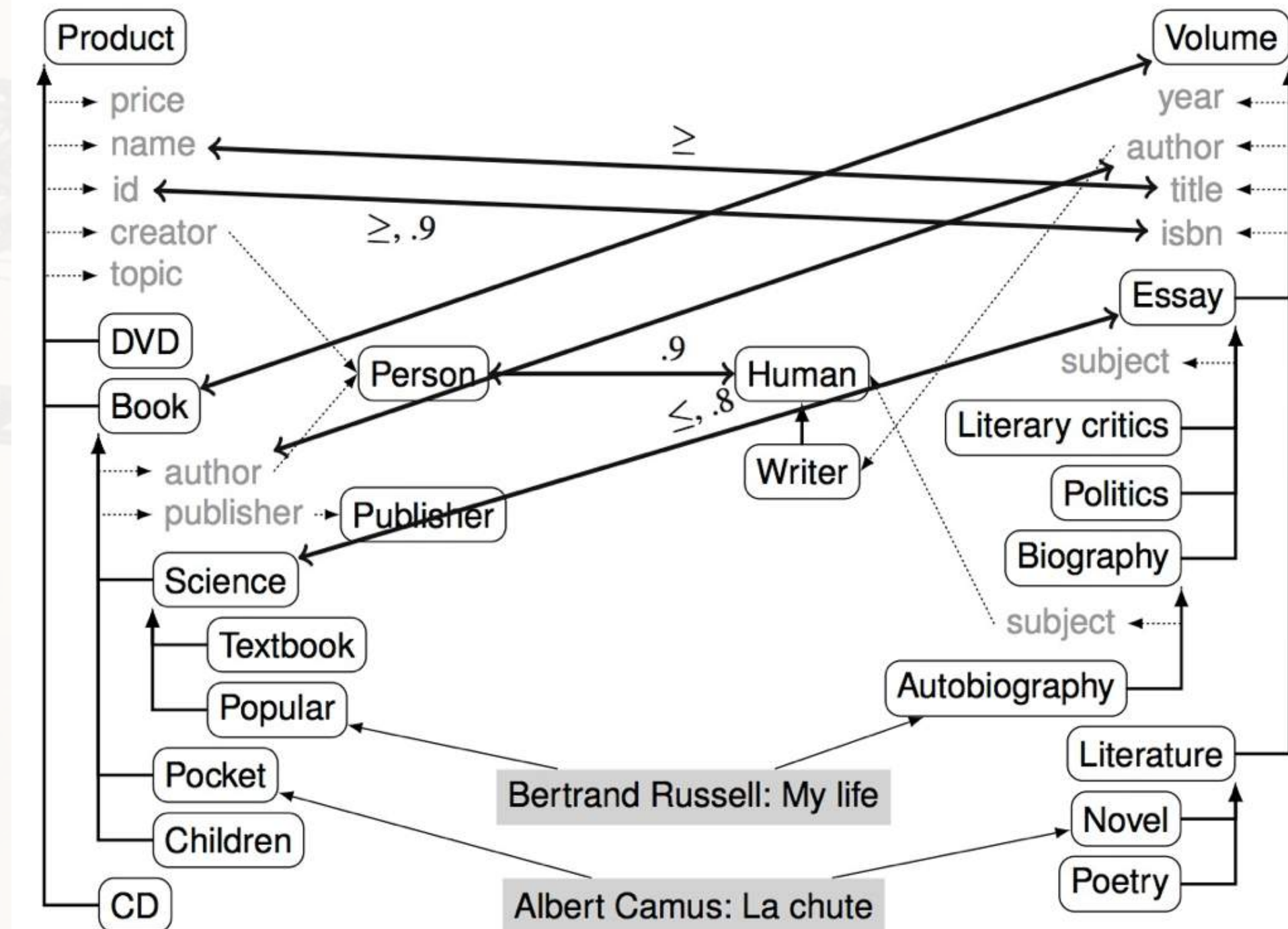
- $$\text{Book}(x) \wedge \text{author}(x, y) \wedge \text{Writer}(y) \Rightarrow .85$$

$$\text{writtenBy}(x, \text{concat}(y.\text{firstname}, y.\text{lastname}))$$

Ontology Alignment Example

12

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



Ontology Matching Techniques

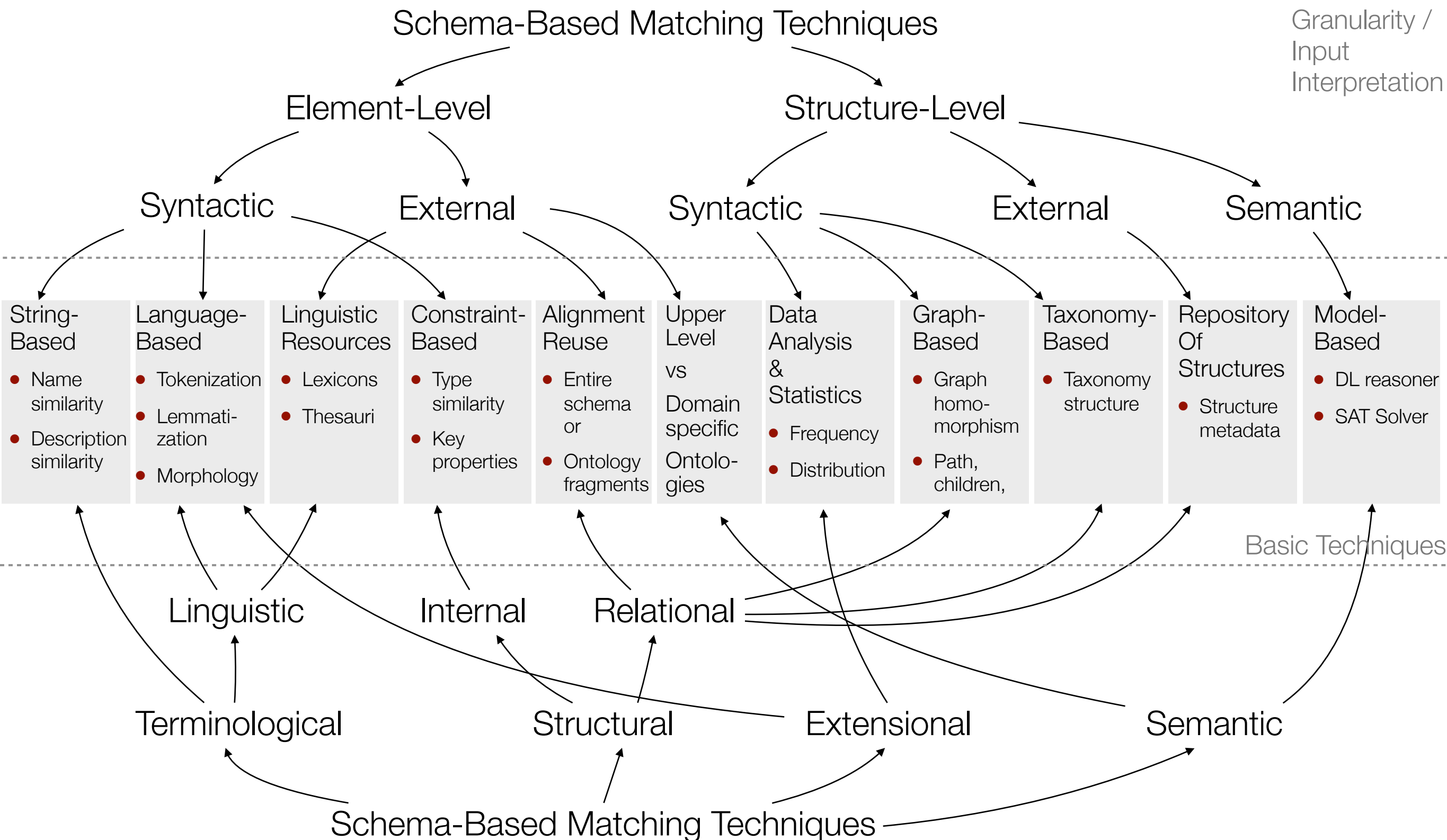
- 13 ● **Element-level Ontology Matching Techniques** consider ontology entities or their instances in isolation from their relations with other entities or their instances
 - **String-Based** - *matching names or descriptions of entities*
 - **Linguistic-Based** - *use NLP, lexicons, or domain specific 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

14

- **Structure-level Ontology Matching Techniques** consider ontology entities or their instances to compare their relations with other entities 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



Granularity /
Input
Interpretation

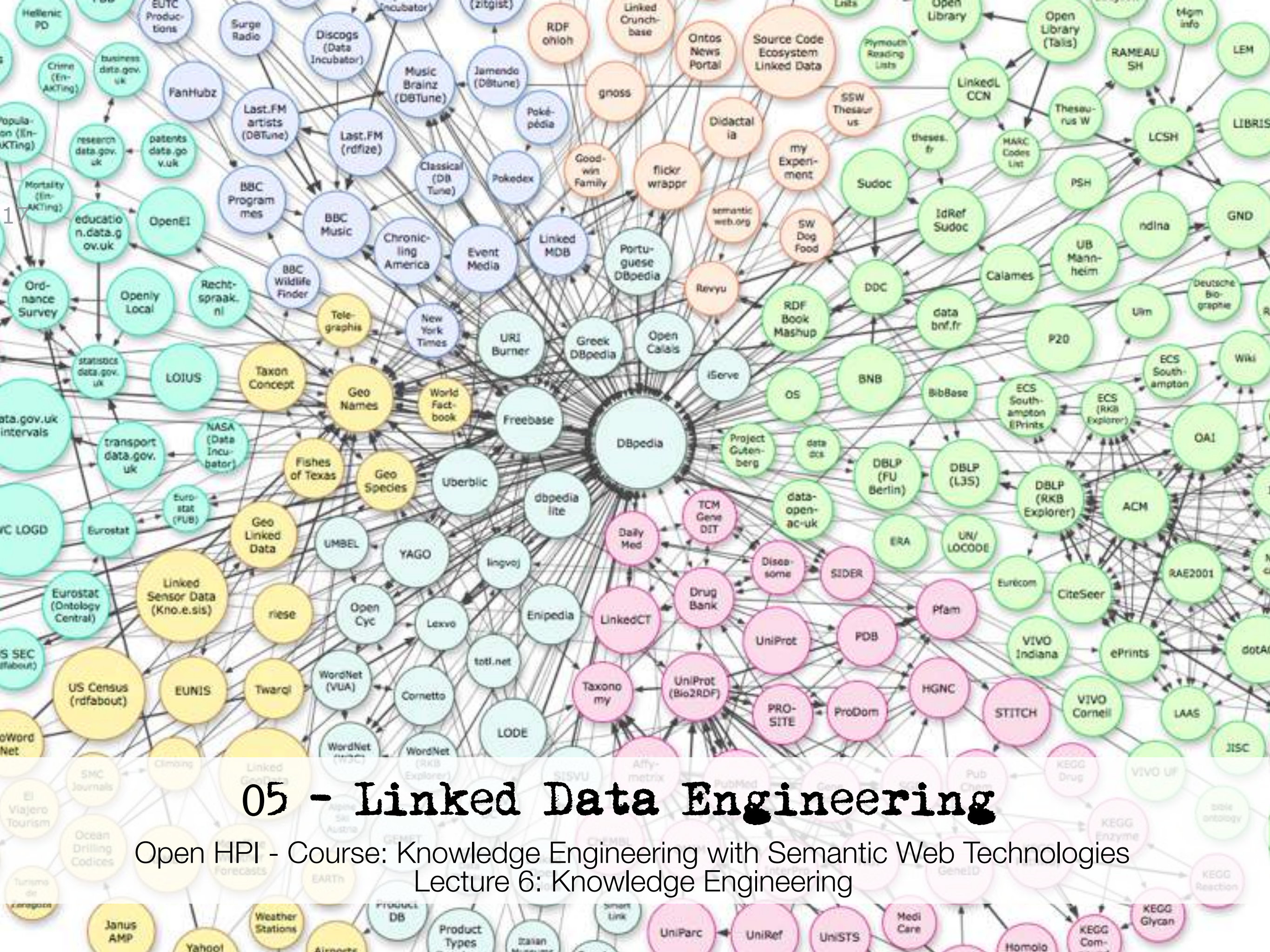
- is a process, where from two or more start ontologies a **new ontology** is created.
- The new ontology **unifies** and **substitutes** the original start ontologies.

- **Union Approach**

The new ontology is the union of all entities of the start ontologies, where conflicts arising 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