Preserving Linked Data: Challenges and Opportunities

Vassilis Christophides
University of Crete & FORTH-ICS
Heraklion, Crete

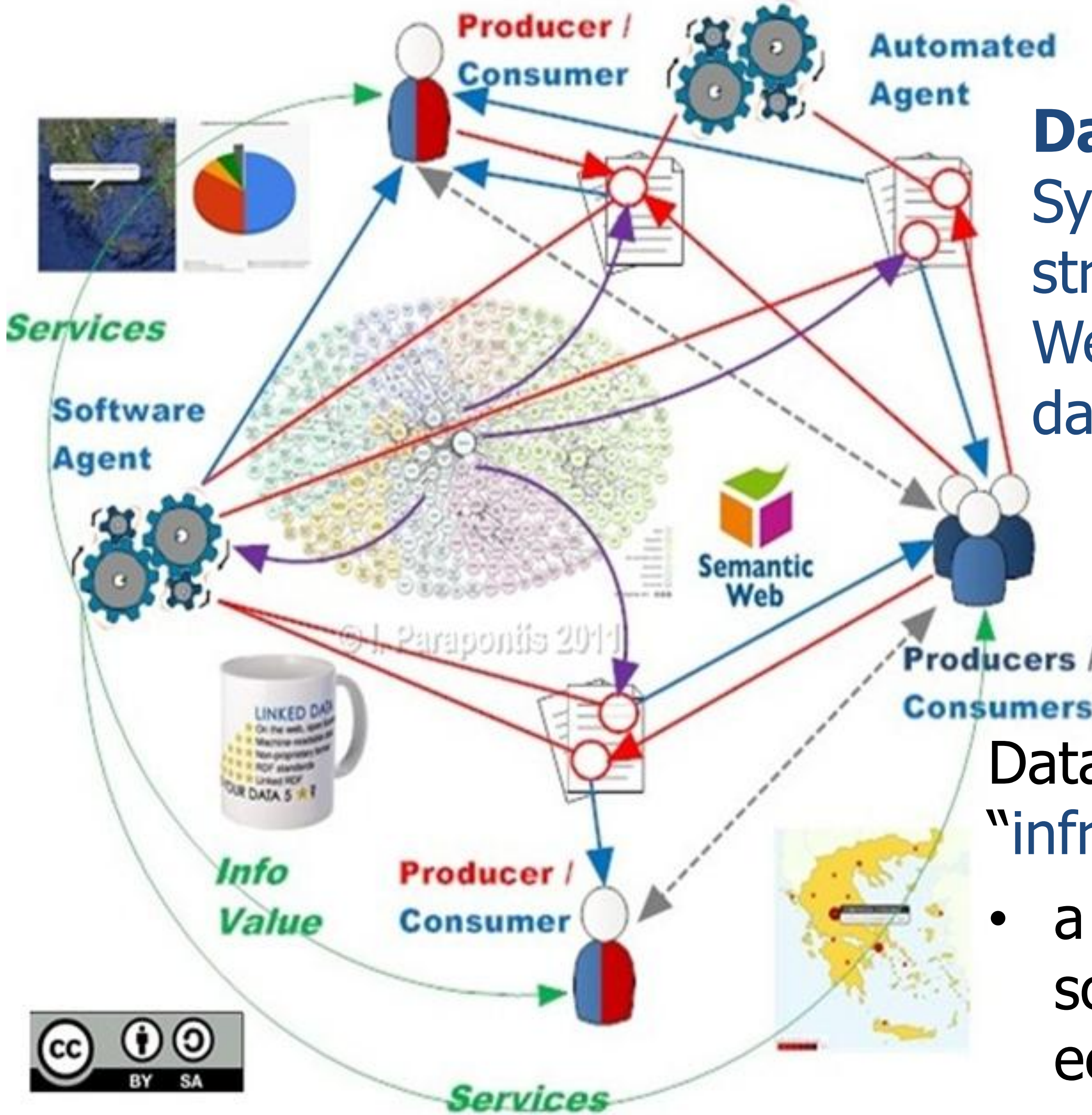
A bit of History: from Web 1.0 & 2.0 …

Many Web sites containing *unstructured, textual* content

Few large Web sites are specialized on *specific content types*

- *Semi-structured/xml* content floating around e-services

A bit of History: …to Web 3.0



Data as Service (DaaS)
Syndicating arbitrarily semi-structured content across Web sites using higher-level data abstractions (entities)

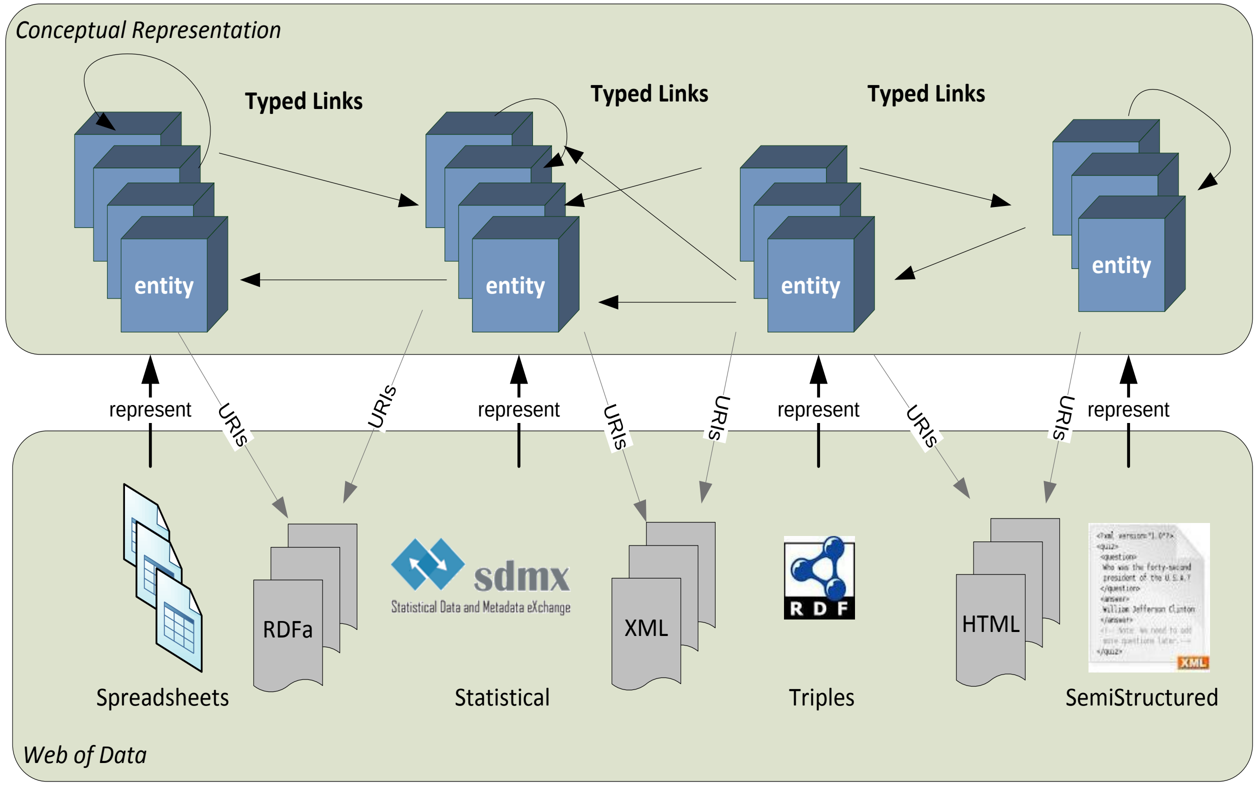
Data themselves become "infrastructure"

- a valuable asset, on which science, technology, the economy and society can advance

The Emerging Web of Data

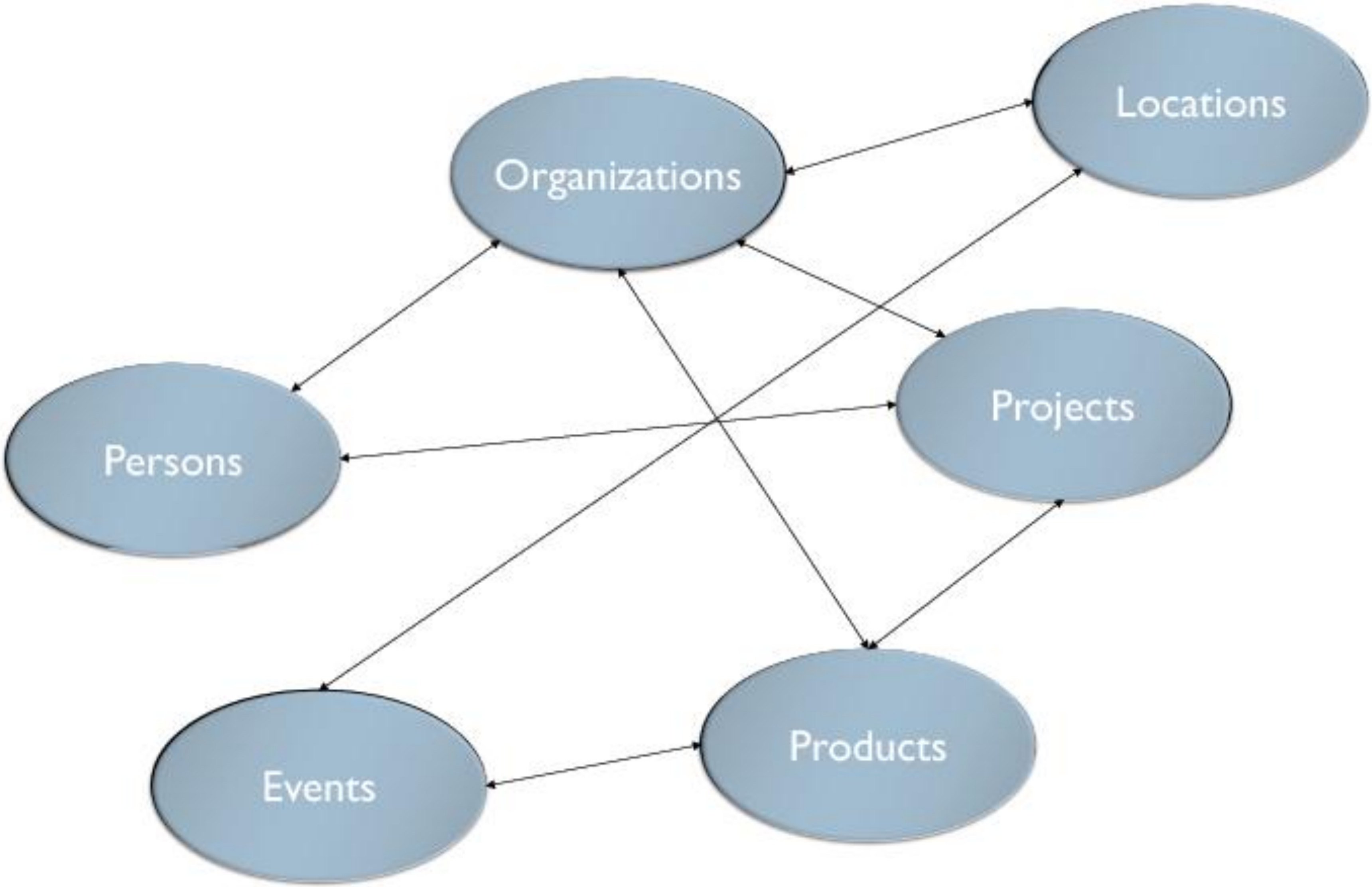
Adapted from Chris Bizer, Richard Cyganiak, Tom Heath, available at http://linkeddata.org/guides-and-tutorials

A Web of things in the world, described by data on the Web



- Global data space connecting data from diverse domains and sources
  - Primary objects: “things” (or description of things)
  - Links between “things”
- Granularity of information: from entire datasets to atomic data

Entities: an Invaluable Asset



- "Entities" is what a large part of our knowledge is about

Web Data of Increasing Standardization

Not all linked data is open and not all open data is linked!

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

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

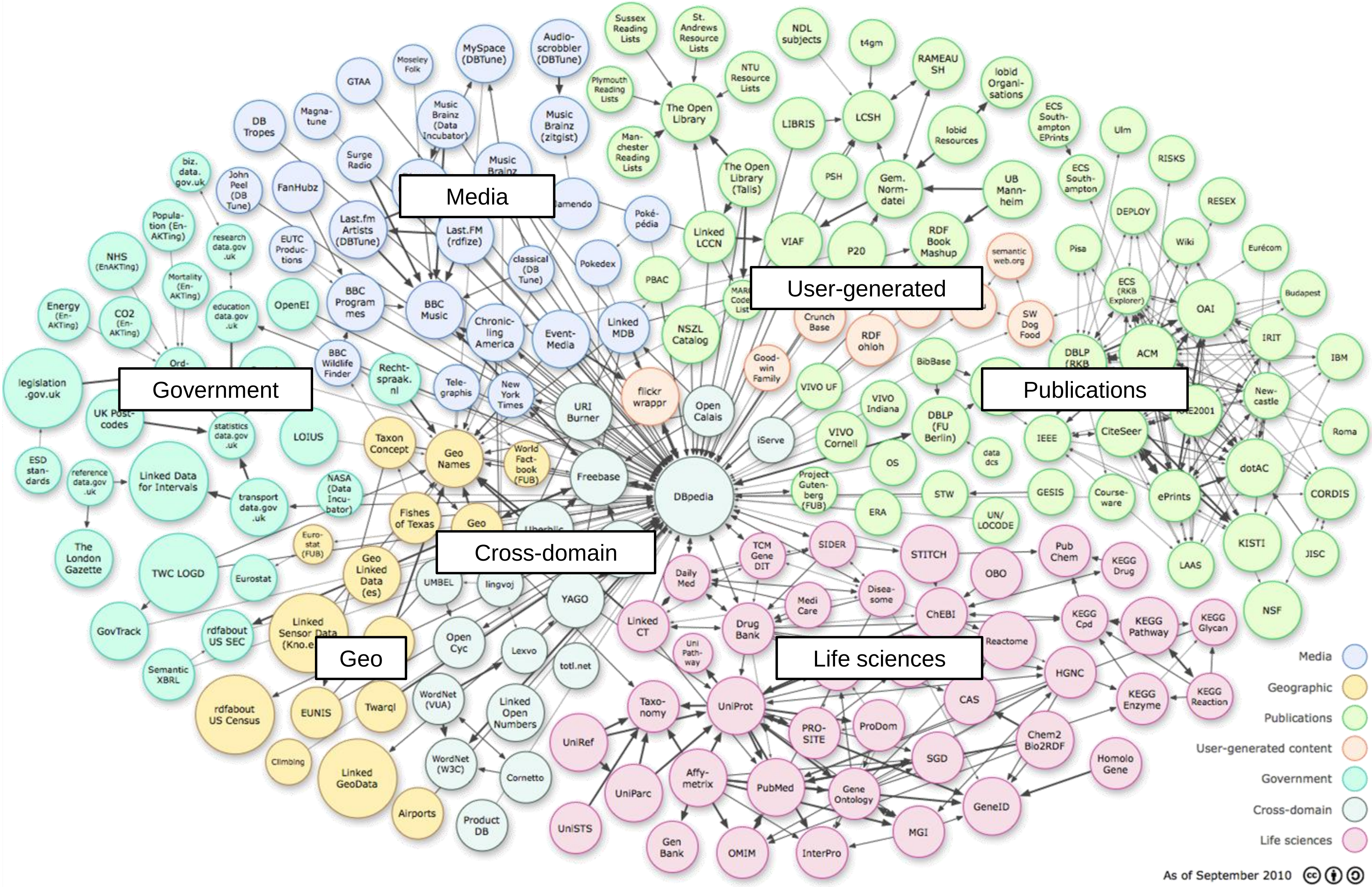
★★★ as (2) plus non-proprietary format (e.g. CSV instead of excel)

★★★★ as (3), plus using open standards from W3C (RDF and SPARQL) to identify things through dereferenceable HTTP URIs, to ensure effective access

★★★★★ as all the above plus establishing links between data of different sources

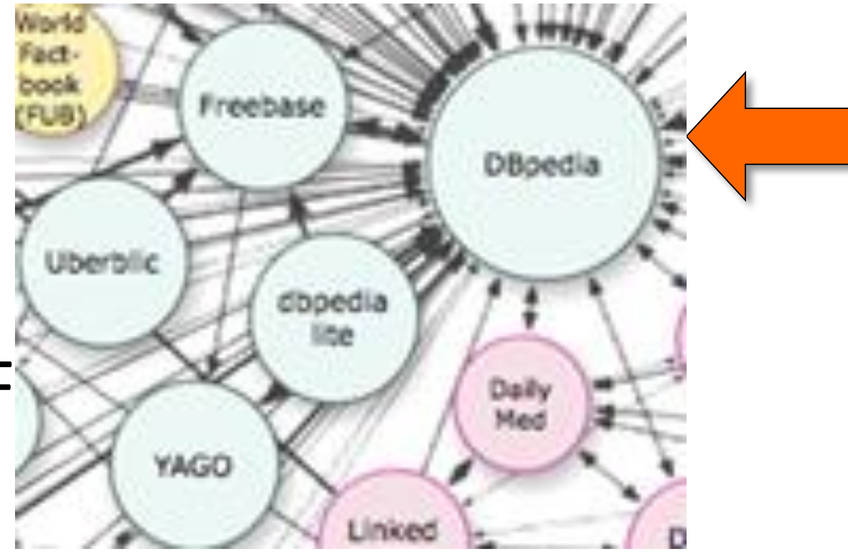
File format	Recommendations (on a scale of 0-5)
csv	★★★
xls	★
pdf	★
doc	★
xml	★★★★
rdf	★★★★★
shp	★★★
ods	★★
tiff	★
jpeg	★
json	★★★
txt	★
html	★★

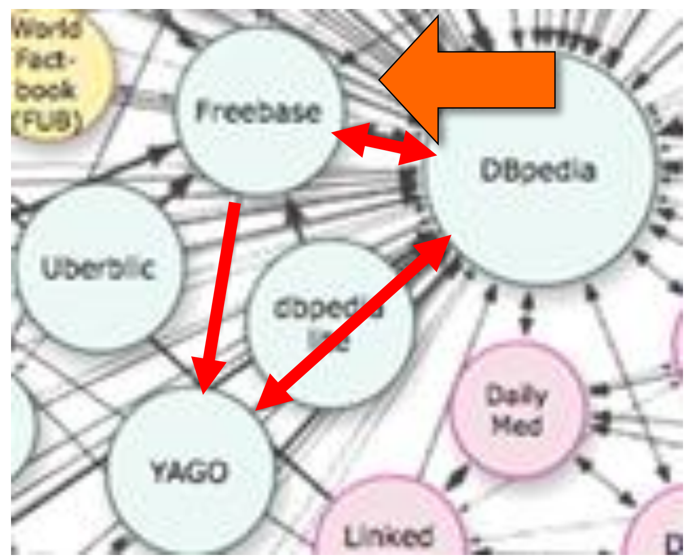
The LOD Cloud



Basic Terminology

- A dataset is a set of RDF triples that are published, maintained or aggregated by a single provider
- A linkset is a collection of RDF links between two datasets i.e. triples whose subject & object are described in different datasets
- But what do we really know about the production and curation processes of the sources publishing in RDF?





What is Digital Preservation (DP)?

- Ensure accessibility and usability of digital objects *over time* and *across domains*, and protect them from *media failure*, *physical loss*, & *hardware/software obsolescence*
- Traditionally, the objective is to preserve on the long run the *authenticity* of a digital object as originally recorded against any technological change
  - extensive annotation of digital objects with information related to their significant properties
    - *content format*
    - *context of production*
    - *structural meta-data*
    - *current behavior, …*

Linked Data vs Digital Objects

- DP techniques proposed for memory institutions and data centers concern fixed digital objects featuring mostly unstructured data
  - raw data sets held in files, scholarly data held in papers…
- Linked Data are digitally-born objects which
  - are graph-structured optionally satisfying integrity constraints (expressed in higher logic formalisms)
  - exhibit complex interdependencies across sources as well as varying data quality (curated knowledge bases vs extracted from text or Web 2.0 sources)
  - change without notification at different granularity levels to keep them fit for contemporary purposes, and be available for discovery and re-use in the future

Linked Data: Behind the scenes!

Different Descriptions of the same Entity

DBpedia	dbpedia:Statue_of_Liberty
rdfs:label	Statue of Liberty, Freiheitsstatue, …
dbpprop:location	New York City, New York, U.S., dbpedia:Liberty_Island
dbpprop:sculptor	dbpedia:Frédéric_Auguste_Bartholdi
dcterms:subject	dbpedia_category:1886_sculptures, …
foaf:isPrimaryTopicOf	http://en.wikipedia.org/wiki/Statue_of_Liberty
dbpprop:beginningDate	1886-10-28 (xsd:date)
dbpprop:restored	19381984 (xsd:integer)
dbpprop:visitationNum	3200000 (xsd:integer)
dbpprop:visitationYear	2009 (xsd:integer)
http://www.w3.org/ns/prov#wasDerivedFrom	http://en.wikipedia.org/wiki/Statue_of_Liberty?oldid=494328330

Freebase	fb:m.072p8
fb:art_form	fb:m.06msq (Sculpture)
fb:media	fb:m.025rsfk (Copper)
fb:architect	fb:m.0jph6 (F. Bartholdi), fb:m.036qb (G. Eiffel), fb:m.02wj4z (R. Hunt)
fb:height_meters	93
fb:opened	1886-10-28

yago select knowledge	yago:Statue_of_Liberty
skos:prefLabel	Statue of Liberty
rdf:type	yago:History_museums_in_NY, yago:GeoEntity
yago:hasHeight	46.0248
yago:wasCreatedOnDate	1886-##-##
yago:isLocatedIn	yago:Manhattan, yago:Liberty_Island,
yago:hasWikipediaUrl	http://en.wikipedia.org/wiki/Statue_of_Liberty

Linked Datasets Depend on Vocabularies

DBpedia	dbpedia:Statue_of_Liberty
rdfs:label	Statue of Liberty, Freiheitsstatue, …
dbpprop:location	New York City, New York, U.S., dbpedia:Liberty_Island
dbpprop:sculptor	dbpedia:Frédéric_Auguste_Bartholdi
dcterms:subject	dbpedia_category:1886_sculptures, …
foaf:isPrimaryTopicOf	http://en.wikipedia.org/wiki/Statue_of_Liberty
dbpprop:beginningDate	1886-10-28 (xsd:date)
dbpprop:restored	19381984 (xsd:integer)
dbpprop:visitationNum	3200000 (xsd:integer)
dbpprop:visitationYear	2009 (xsd:integer)
http://www.w3.org/ns/prov#wasDerivedFrom	http://en.wikipedia.org/wiki/Statue_of_Liberty?oldid=494328330

Freebase	fb:m.072p8
fb:art_form	fb:m.06msq (Sculpture)
fb:media	fb:m.025rsfk (Copper)
fb:architect	fb:m.0jph6 (F. Bartholdi), fb:m.036qb (G. Eiffel), fb:m.02wj4z (R. Hunt)
fb:height_meters	93
fb:opened	1886-10-28

yago select knowledge	yago:Statue_of_Liberty
skos:prefLabel	Statue of Liberty
rdf:type	yago:History_museums_in_NY, yago:GeoEntity
yago:hasHeight	46.0248
yago:wasCreatedOnDate	1886-##-##
yago:isLocatedIn	yago:Manhattan, yago:Liberty_Island,
yago:hasWikipediaUrl	http://en.wikipedia.org/wiki/Statue_of_Liberty

Linked Datasets Have Varying Quality

DBpedia	dbpedia:Statue_of_Liberty
rdfs:label	Statue of Liberty, Freiheitsstatue, …
dbpprop:location	New York City, New York, U.S., dbpedia:Liberty_Island
dbpprop:sculptor	dbpedia:Frédéric_Auguste_Bartholdi
dcterms:subject	dbpedia_category:1886_sculptures, …
foaf:isPrimaryTopicOf	http://en.wikipedia.org/wiki/Statue_of_Liberty
dbpprop:beginningDate	1886-10-28 (xsd:date)
dbpprop:restored	19381984 (xsd:integer)
dbpprop:visitationNum	3200000 (xsd:integer)
dbpprop:visitationYear	2009 (xsd:integer)
http://www.w3.org/ns/prov#wasDerivedFrom	http://en.wikipedia.org/wiki/Statue_of_Liberty?oldid=494328330

Freebase	fb:m.072p8
fb:art_form	fb:m.06msq (Sculpture)
fb:media	fb:m.025rsfk (Copper)
fb:architect	fb:m.0jph6 (F. Bartholdi), fb:m.036qb (G. Eiffel), fb:m.02wj4z (R. Hunt)
fb:height_meters	93
fb:opened	1886-10-28

yago select knowledge	yago:Statue_of_Liberty
skos:prefLabel	Statue of Liberty
rdf:type	yago:History_museums_in_NY, yago:GeoEntity
yago:hasHeight	46.0248
yago:wasCreatedOnDate	1886-##-##
yago:isLocatedIn	yago:Manhattan, yago:Liberty_Island,
yago:hasWikipediaUrl	http://en.wikipedia.org/wiki/Statue_of_Liberty

Linked Datasets Evolve Over Time

Current version of DBpedia

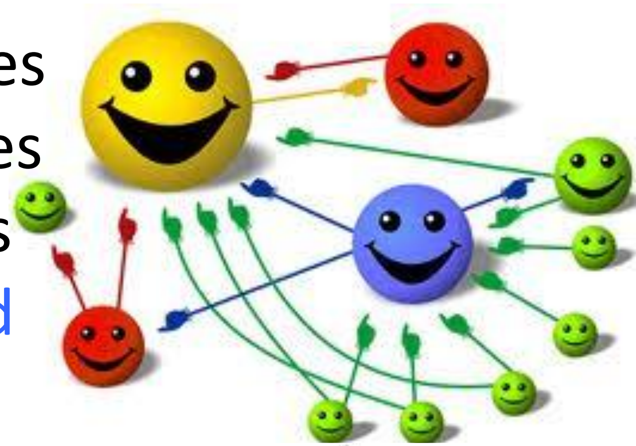
DBpedia	dbpedia:Statue_of_Liberty
rdfs:label	Statue of Liberty, Freiheitsstatue, …
dbpprop:location	New York City, New York, U.S., dbpedia:Liberty_Island
dbpprop:sculptor	dbpedia:Frédéric_Auguste_Bartholdi
dcterms:subject	dbpedia_category:1886_sculptures, …
foaf:isPrimaryTopicOf	http://en.wikipedia.org/wiki/Statue_of_Liberty
dbpprop:beginningDate	1886-10-28 (xsd:date)
dbpprop:restored	19381984 (xsd:integer)
dbpprop:visitationNum	3200000 (xsd:integer)
dbpprop:visitationYear	2009 (xsd:integer)
http://www.w3.org/ns/prov#wasDerivedFrom	http://en.wikipedia.org/wiki/Statue_of_Liberty?oldid=494328330

Previous version of DBpedia

DBpedia	dbpedia:Statue_of_Liberty
rdfs:label	Statue of Liberty, Freiheitsstatue, …
dbpprop:location	New York City, New York, U.S., dbpedia:Liberty_Island
dbpprop:sculptor	dbpedia:Frédéric_Auguste_Bartholdi
dcterms:subject	dbpedia_category:1886_sculptures, …
foaf:isPrimaryTopicOf	http://en.wikipedia.org/wiki/Statue_of_Liberty
dbpprop:built	1886-10-28 (xsd:date)
dbpprop:restored	19381984 (xsd:integer)
dbpprop:hasHeight	151 (xsd:integer)
http://www.w3.org/ns/prov#wasDerivedFrom	http://en.wikipedia.org/wiki/Statue_of_Liberty?oldid=494328330

DP Challenges for Linked Data

- The 'publish-first-refine-later' philosophy of the Linked Data movement, complemented by the open, decentralized nature of the Web results in Data
  - Incompleteness: real world entitles are usually partially described in data sources
  - Redundancy: the same real world entities are represented in multiple data sources
  - Inconsistency: various forms of inter and intra source data conflicts
  - Incorrectness: errors can be propagated from one source to the other due to copying



- Mastering the varying data quality is a prerequisite for trusting preserved data originating from various sources

DP Challenges for Linked Data

- Still data publishing and preservation are two largely separated processes which are addressed by SW and DP communities.
  - Shouldn't the "publishers" worry about preserving all their hard work?
  - Shouldn't the "preservers" be concerned about the way they organize, link, and annotate their linked data?
- Need to break down the traditional boundaries between the data creators & publishers and the data archivists & brokers
  - Integrate curation [when high current/ongoing interest] and preservation [when fall off in interest] activities
  - Distribute preservation costs over the life-cycle of linked datasets among data stewards (pay-as-you-go data preservation)

Vision 2030: High-Level Group on Scientific Data

"Researchers and practitioners from any discipline are able to find, access and process the data they need. They can be confident in their ability to use and understand data and they can evaluate the degree to which the data can be trusted"

- How we support future users in Trusting Data?
  - By assessing their quality wr.t to the entities they describe
  - By recording from which sources did they originate from
  - By understanding how their identity and integrity has evolved over time
  - By ensuring that they has been preserved properly

Frame Linked Data Preservation as a Research Problem

- How can we identify that different resource descriptions within or across datasets refer to the same real-world entity (the entity resolution problem) to convey various aspects of the quality of the harvested datasets (e.g., redundancy, completeness, freshness)?

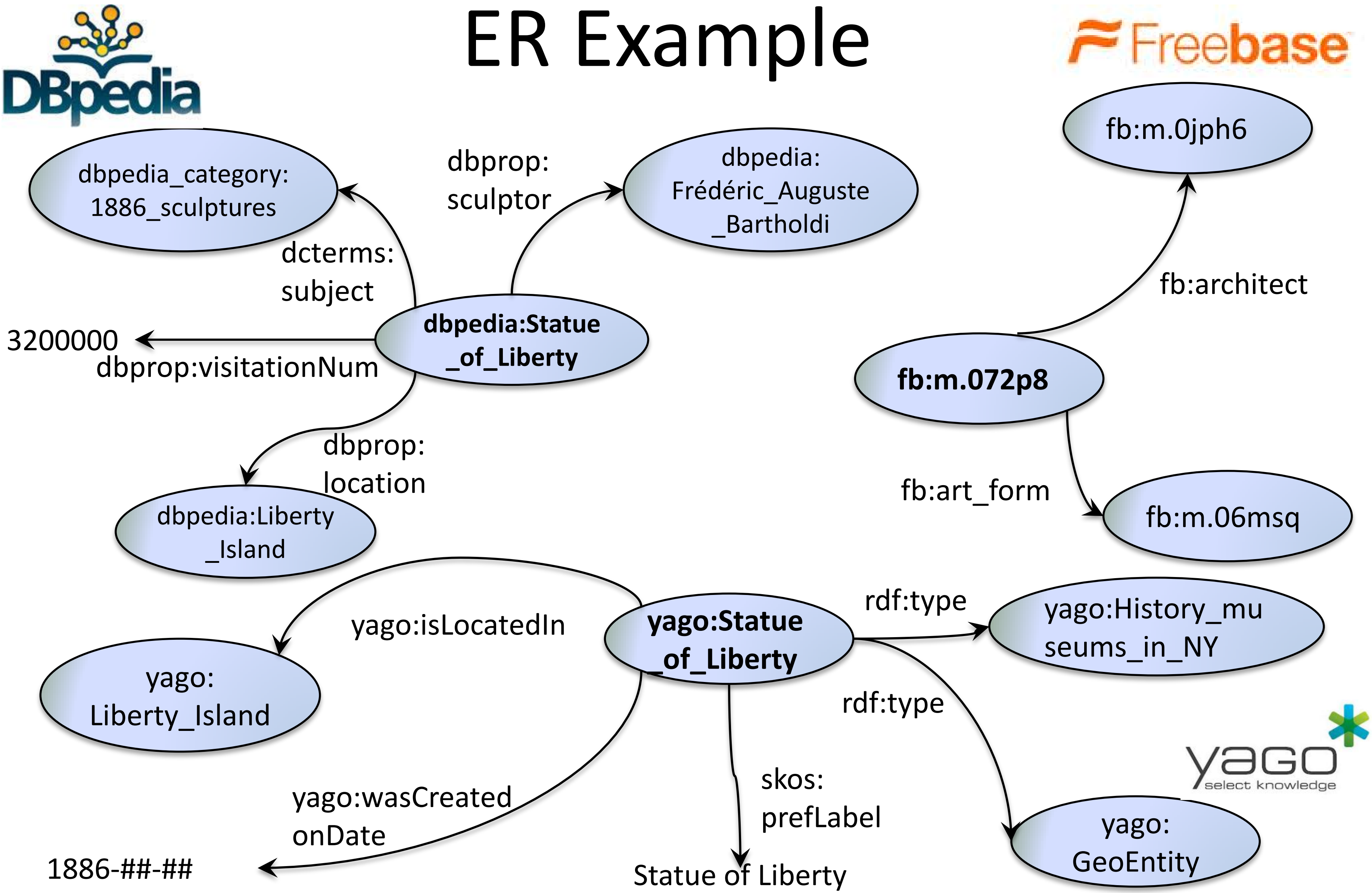
- How can we record dependencies of datasets (the provenance problem) and how they can smoothly represented along with other (temporal, spatial, thematic) metadata (the annotation problem)?

Frame Linked Data Preservation as a Research Problem

- How can we monitor changes of third-party datasets (the evolution tracking problem) or how can local/remote data imperfections (e.g., due to change propagation) can be repaired (the curation problem)?

- How can we appease what versions of linked datasets should to be preserved for future use (the multi-version archive consistency problem) and how we will be able to ask a query not only about any past state of the dataset but also about the evolution of some part of it (the longitudinal querying problem)?

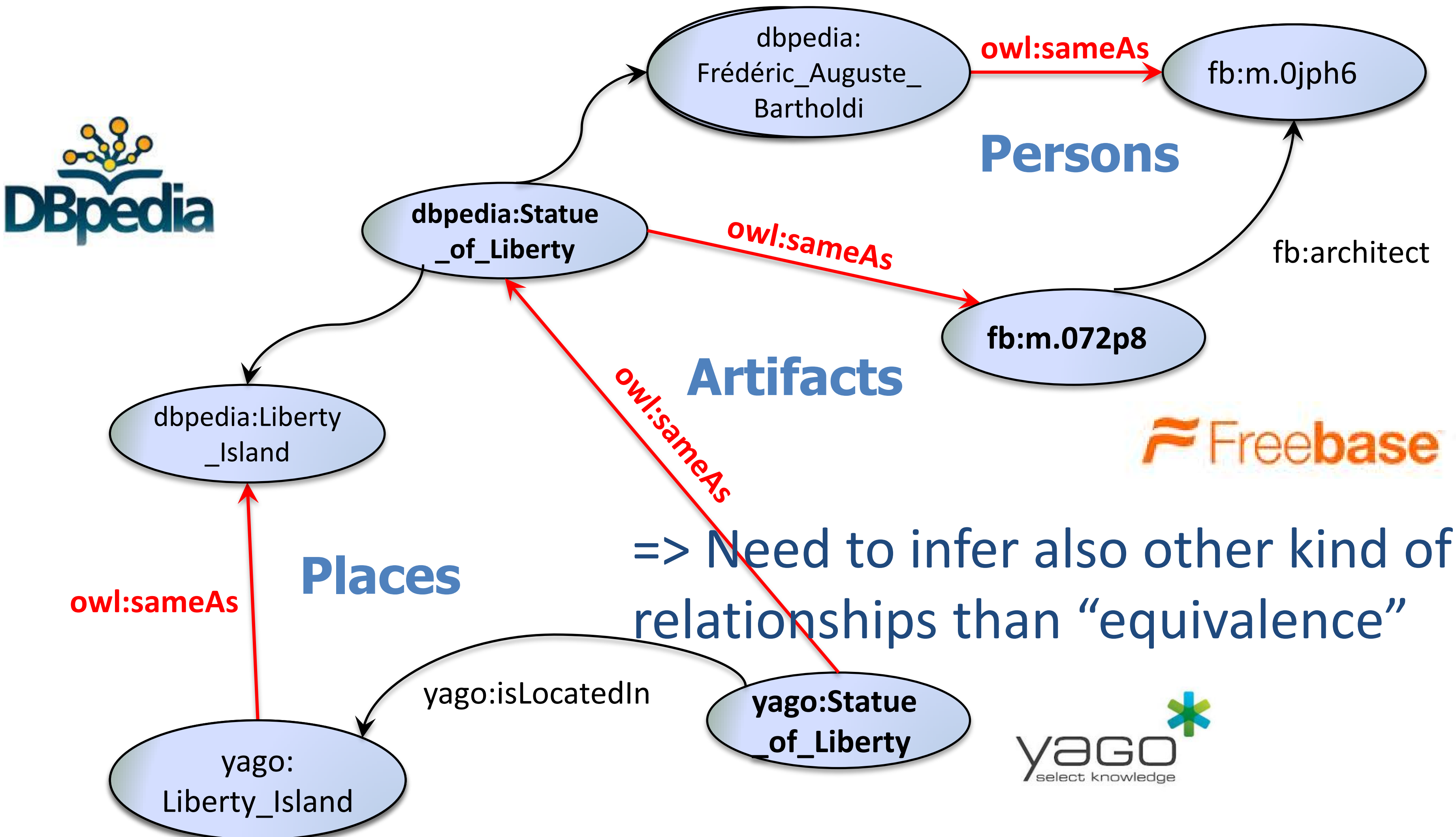
ER Example



Entity resolution: The problem of identifying descriptions of the same entity within one or across multiple data sources wrt. a match function

- No longer just matching of entity names

ER Example



=> Need to infer also other kind of relationships than "equivalence"

An entity resolution is a partition of a set of entity descriptions, such that:

1. Matching entity descriptions are placed in the same subset
2. All the descriptions of the same subset match

What Makes ER Difficult for Linked Data

=> Deal with loosely structured entities

- Linked Data are inherently semi-structured
  – several semantic types (see rdf:type properties in Yago) could be simultaneously employed resulting to entity descriptions even of the same type (persons, places, …) with quite different structures

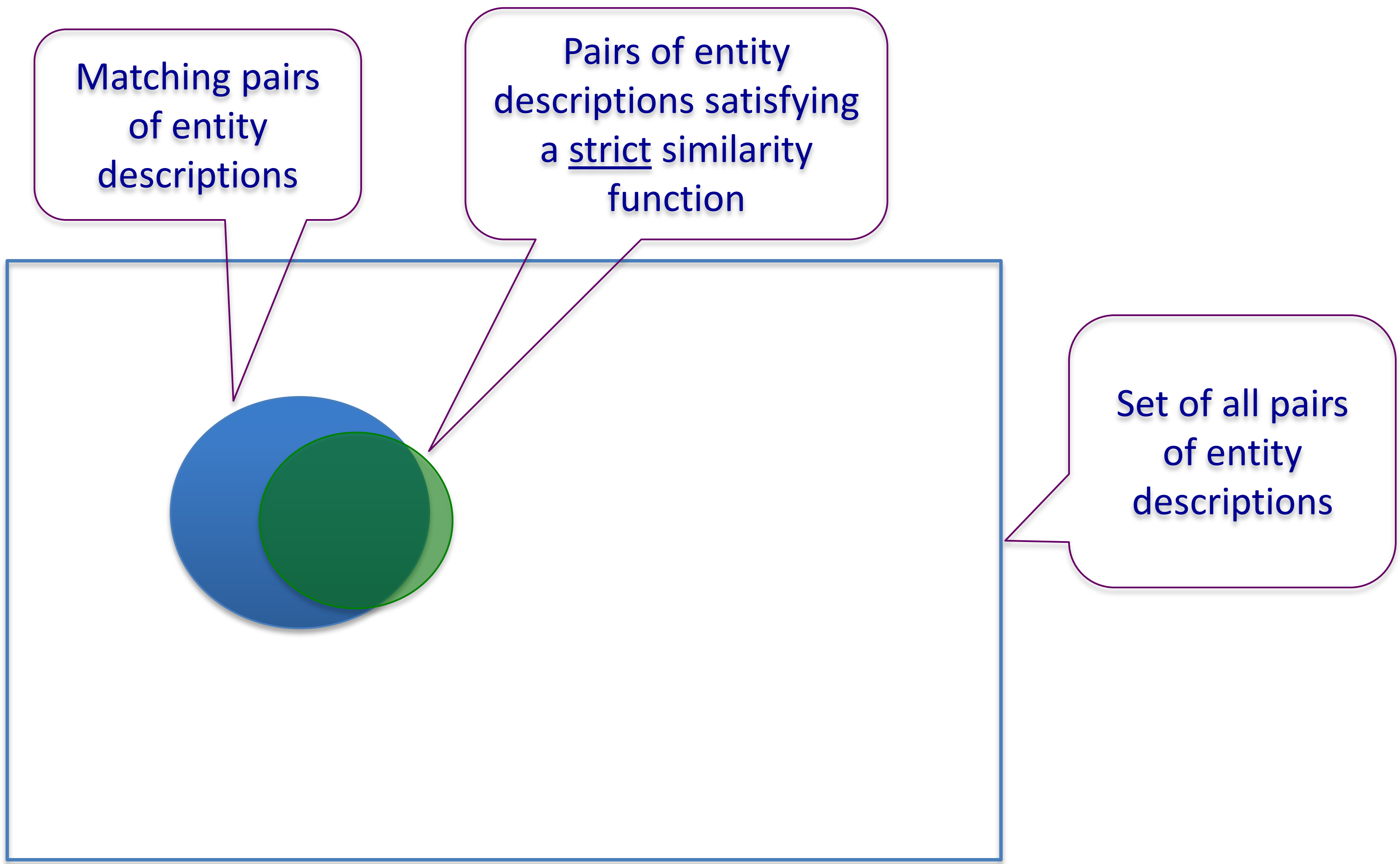
=> Need for cross-domain techniques

- Linked Data heavily rely on heterogeneous vocabularies
  – DBPedia 3.4: 50,000 properties
  – Google Base:100,000 schemata and 10,000 entity types

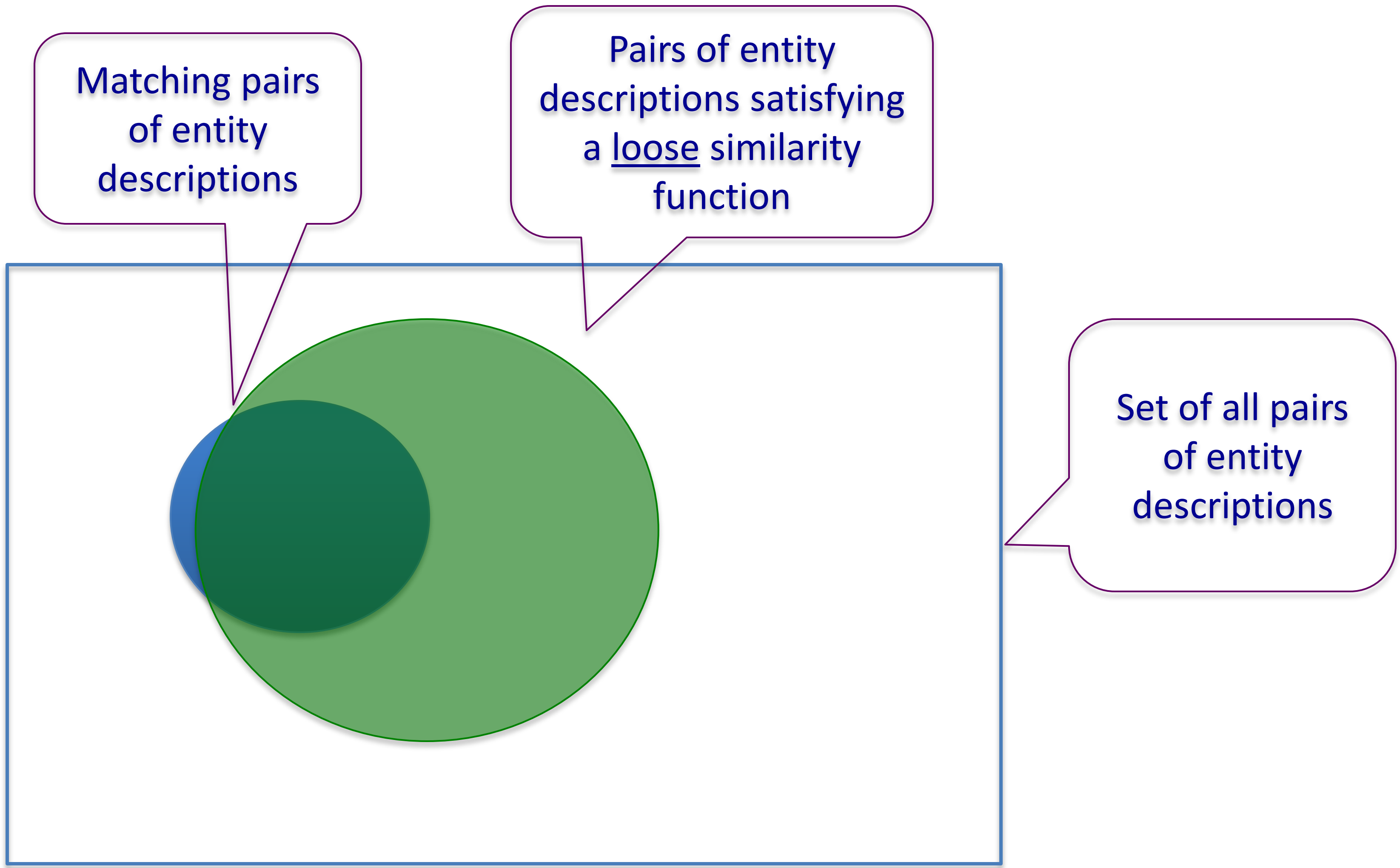
=> Calls for efficient parallel techniques

- Linked Data are Big Data
  – The LOD cloud consists of 32 billion RDF triples (last update: 2011)
  – DBPedia 3.4: 36.5 million triples, 2.1 million entity descriptions
  – BTC09: 1.15 billion triples, 182 million entity descriptions

The Role of Similarity Functions



The Role of Similarity Functions



Entity Collections and ER Types

- 2 kinds of entity collections given as input to an ER task:
  - Clean, which are duplicate-free (e.g.,DBPedia,Freebase,DBLP)
  - Dirty, which contain duplicate entity descriptions in themselves (e.g., Google Scholar, Citeseer)
- An ER task that receives as input two entity collections can be of the following types:
  - Clean-Clean ER: Given two clean, but overlapping entity collections, identify the common entity descriptions (a.k.a. the Record Linkage in databases)
  - Dirty-Clean ER:
  - Dirty-Dirty ER: Identify unique entity descriptions contained in union of the input entity collections (a.k.a. the Deduplication problem in databases)
- In the Web of Data we encountering more Clean-Clean ER

Scaling ER to the Web of Data

- Blocking to reduce the number of comparisons:
  - Split entity descriptions into blocks
  - Compare each description to the descriptions within the same block
- Desiderata
  - Similar entity descriptions in the same block
  - Dissimilar entity descriptions in different blocks
- Blocking approaches are distinguished between:
  - Partitioning, where each description is placed in exactly one block : Fewer comparisons
  - Overlapping, where each description can be placed in more than one block : More identified matches

Blocking techniques for Linked Data

- Multi-relational and cross-domain entity resolution
  - Token blocking
  - Property clustering
  - Prefix-Infix(-Suffix)

- Large-scale entity resolution
  - Choose a computationally not expensive similarity function
  - Process in parallel partitions of the entity graph in Map/Reduce nodes

Token Blocking [Papadakis et al. 2011]

- Ignores (semantic or structured) types of entities
  - String similarity of tokens of property literal values
- Distinct tokens of each property value of each entity description corresponds to a block
  - Each block contains all entities with the corresponding token
- High recall at the cost of low precision and low efficiency:
  - Most true matches are placed in the same block
  - Many non-matches are also placed in the same block
  - The same pair of descriptions is contained in many blocks

Token Blocking - Example

Entity descriptions:

e_1= {(name, Eiffel Tower), (architect, Sauvestre), (year, 1889), (location, Paris)}

e_2= {(name, Statue of Liberty), (architect, Bartholdi Eiffel), (year, 1886), (located, NY)}

e_3= {(about, Lady Liberty), (architect, Eiffel), (location, NY)}

e_4= {(about, Eiffel Tower), (architect, Sauvestre), (year, 1889), (located, Paris)}

e_5= {(name, White Tower), (year-constructed, 1450), (location, Thessaloniki)}

Generated blocks:

Eiffel	Tower	Statue	Liberty	White	1889	Bartholdi
e_1, e_2, e_3, **e_4**	**e_1**, **e_4**, e_5	e_2	e_2, e_3	e_5	**e_1**, **e_4**	e_2

NY	Paris	1886	1450	Lady	Sauvestre	Thessaloniki
e_2, e_3	**e_1**, **e_4**	e_2	e_5	e_3	**e_1**, **e_4**	e_5

The pair (e_1, e_4) is contained in 5 different blocks!

Property clustering [Papadakis et al. 2013]

- Assuming two duplicate-free datasets
  - Recognize similarity of properties based on the string similarity of their literal values occurring in entity descriptions
- Two main blocking steps:
  1. Similar properties are placed together in non-overlapping clusters
  2. Token blocking is performed on the descriptions of each cluster

Clustering Entity Properties

1. For each property of dataset D_1:
   - Find the most similar property of dataset D_2
2. For each property of dataset D_2:
   - Find the most similar property of dataset D_1
3. Compute the transitive closure of the generated pairs of similar properties
4. Similar properties form clusters
5. All singleton clusters are merged into a common one

Clustering Entity Properties: Example

e_1= {(about, Eiffel Tower), (architect, Sauvestre), (year, 1889), (located, Paris)}

e_2= {(about, Statue of Liberty), (architect, Bartholdi Eiffel), (year, 1886), (located, NY)}

e_3= {(about, Auguste Bartholdi), (born, 1834)}

e_4= {(about, Joan Tower), (born, 1938)}

e_5= {(work, Lady Liberty), (artist, Bartholdi), (location, NY)}

e_6= {(work, Eiffel Tower), (year-constructed, 1889), (location, Paris)}

e_7= {(work, Bartholdi Fountain), (year-constructed,1876), (location, Washington D.C.)}

Clustering Entity Properties: Example

e_1= {(about, Eiffel Tower), (architect, Sauvestre), (year, 1889), (located, Paris)}

e_2= {(about, Statue of Liberty), (architect, Bartholdi Eiffel), (year, 1886), (located, NY)}

e_3= {(about, Auguste Bartholdi), (born, 1834)}

e_4= {(about, Joan Tower), (born, 1938)}

e_5= {(work, Lady Liberty), (artist, Bartholdi), (location, NY)}

e_6= {(work, Eiffel Tower), (year-constructed, 1889), (location, Paris)}

e_7= {(work, Bartholdi Fountain), (year-constructed,1876), (location, Washington D.C.)}

Finding the property of D2 that is the most similar to the property "about" of D1:

values of about: {Eiffel, Tower, Statue, Liberty, Auguste, Bartholdi, Juan, Tower}

compared to (with Jaccard similarity) :

values of work: {Lady, Liberty, Eiffel, Tower, Bartholdi, Fountain} → Jaccard = 4/10

values of artist: {Bartholdi} → Jaccard = 1/8

values of location: {NY, Paris, Washington, D.C.} → Jaccard = 0

values of year-constructed: {1889, 1876} → Jaccard = 0

Clustering Entity Properties: Example

e_1= {(about, Eiffel Tower), (architect, Sauvestre), (year, 1889), (located, Paris)}

e_2= {(about, Statue of Liberty), (architect, Bartholdi Eiffel), (year, 1886), (located, NY)}

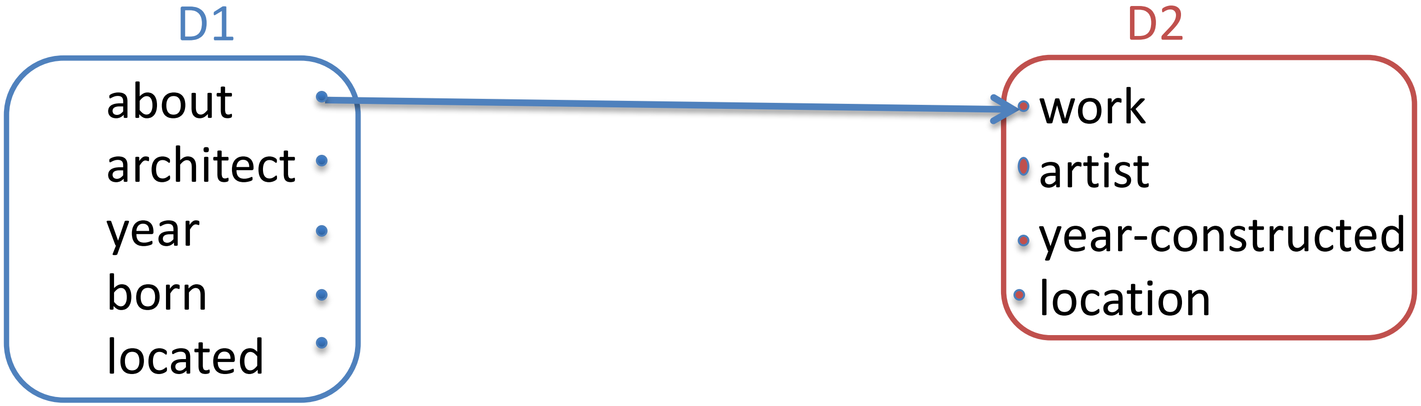
e_3= {(about, Auguste Bartholdi), (born, 1834)}

e_4= {(about, Joan Tower), (born, 1938)}

e_5= {(work, Lady Liberty), (artist, Bartholdi), (location, NY)}

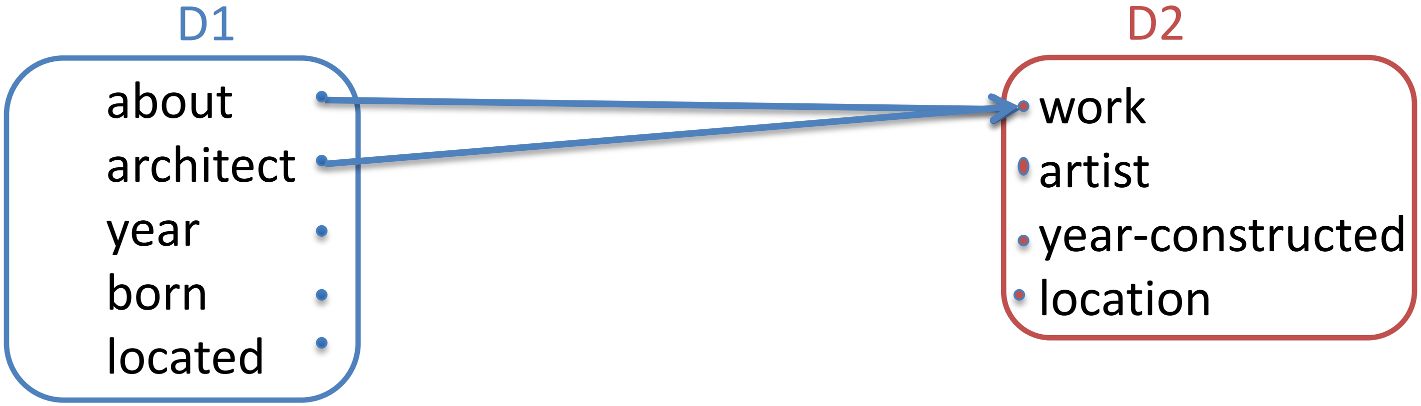
e_6= {(work, Eiffel Tower), (year-constructed, 1889), (location, Paris)}

e_7= {(work, Bartholdi Fountain), (year-constructed,1876), (location, Washington D.C.)}



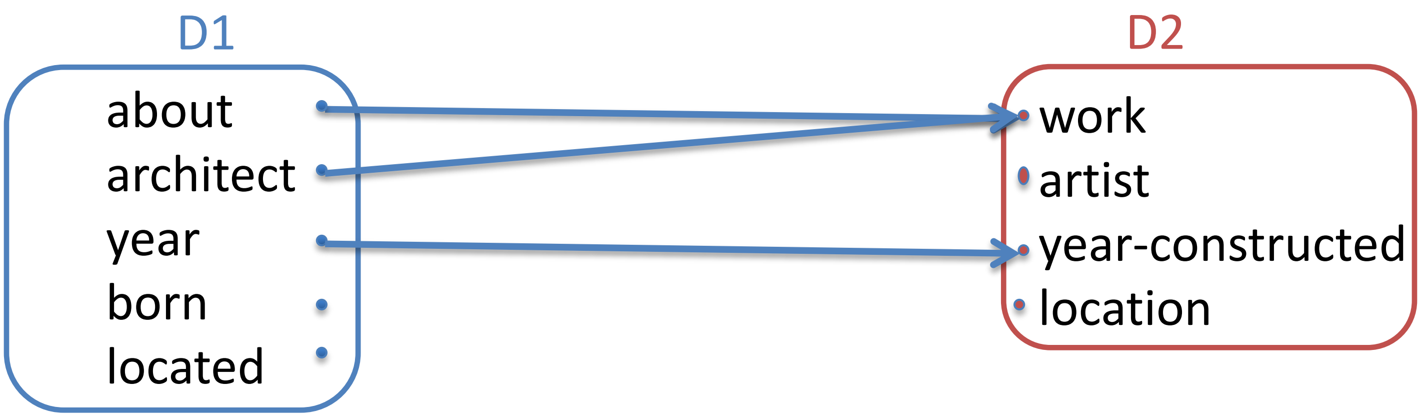
Clustering Entity Properties: Example

- Similarly for the rest of the properties…



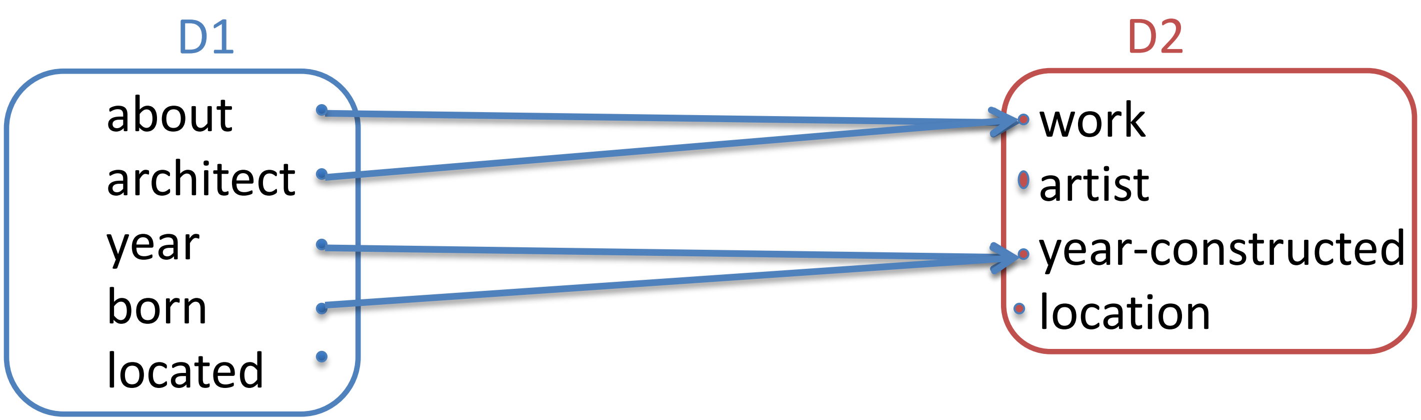
Clustering Entity Properties: Example

- Similarly for the rest of the properties…



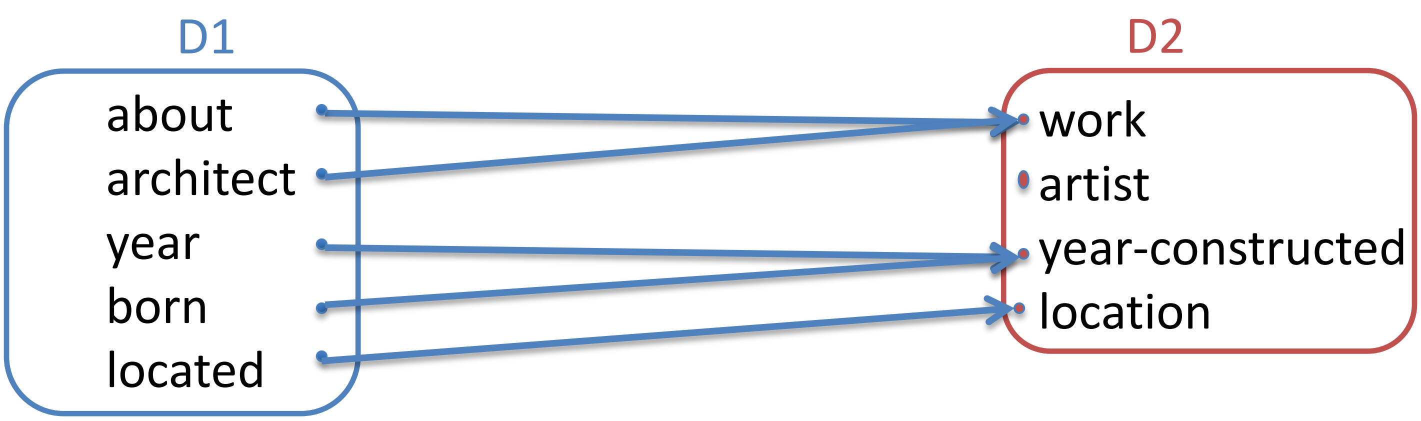
Clustering Entity Properties: Example

- Similarly for the rest of the properties…



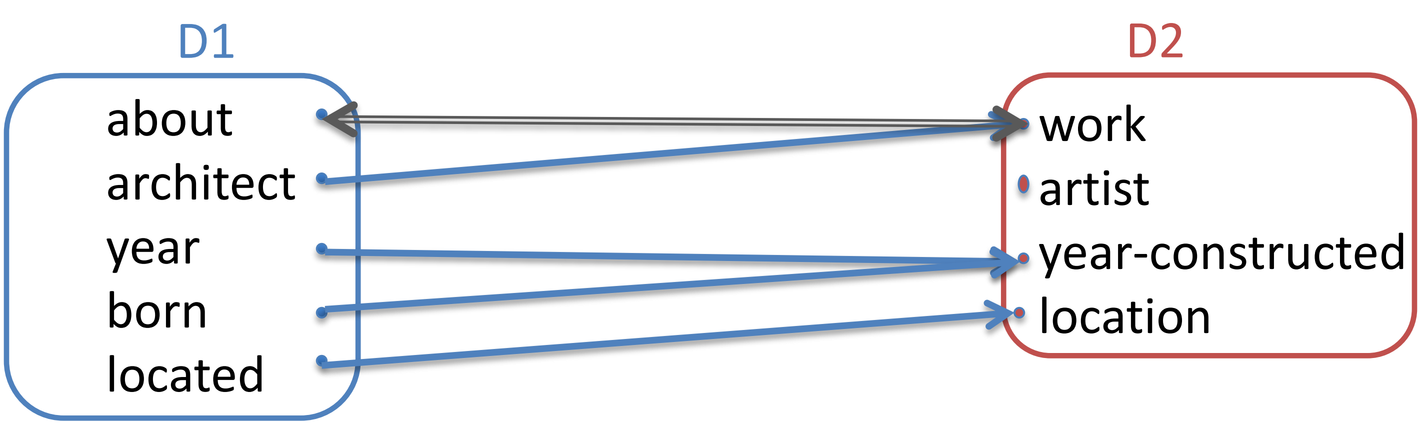
Clustering Entity Properties: Example

- Similarly for the rest of the properties…



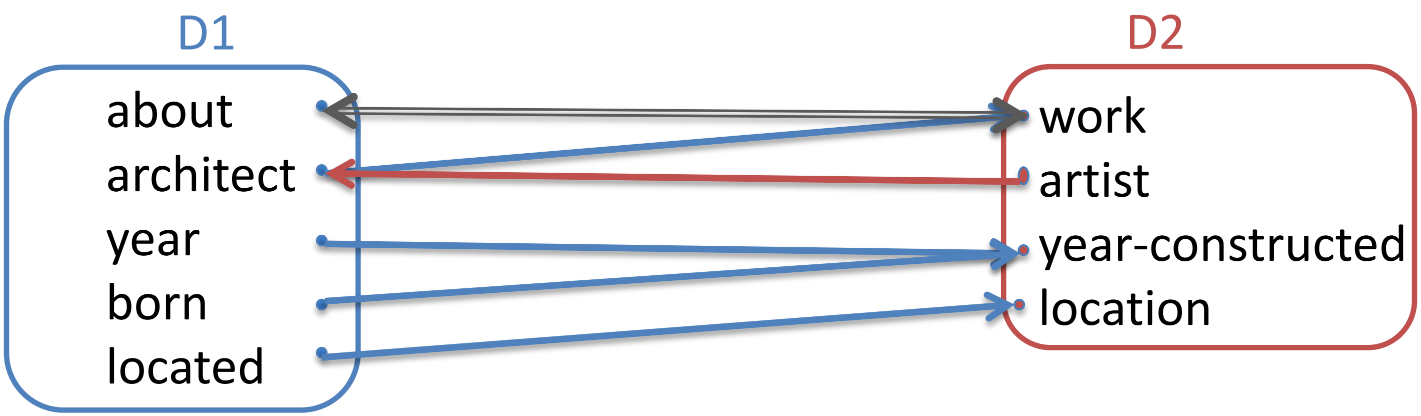
Clustering Entity Properties: Example

- Similarly for the rest of the properties…



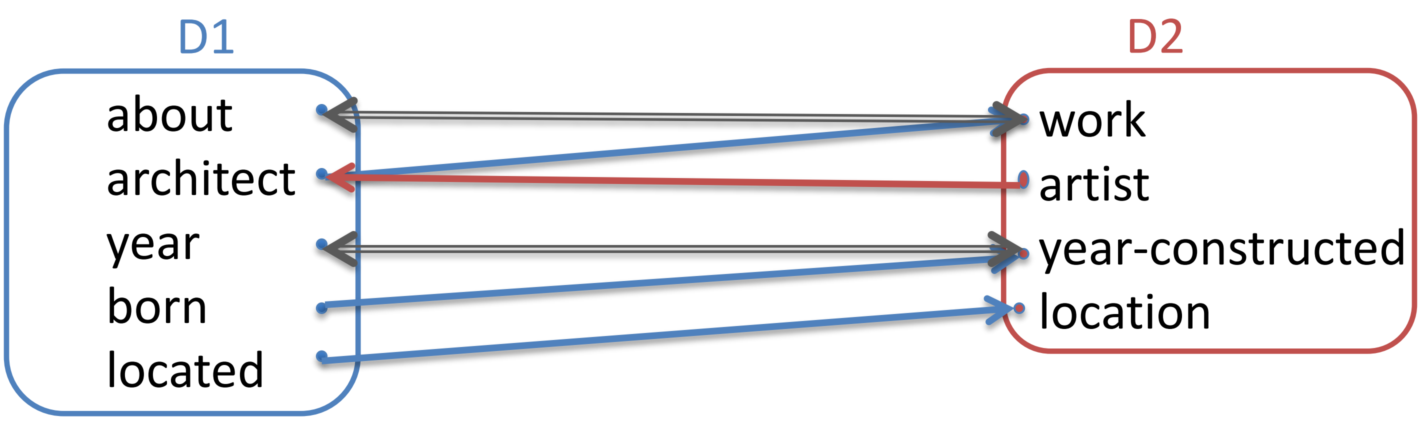
Clustering Entity Properties: Example

- Similarly for the rest of the properties…



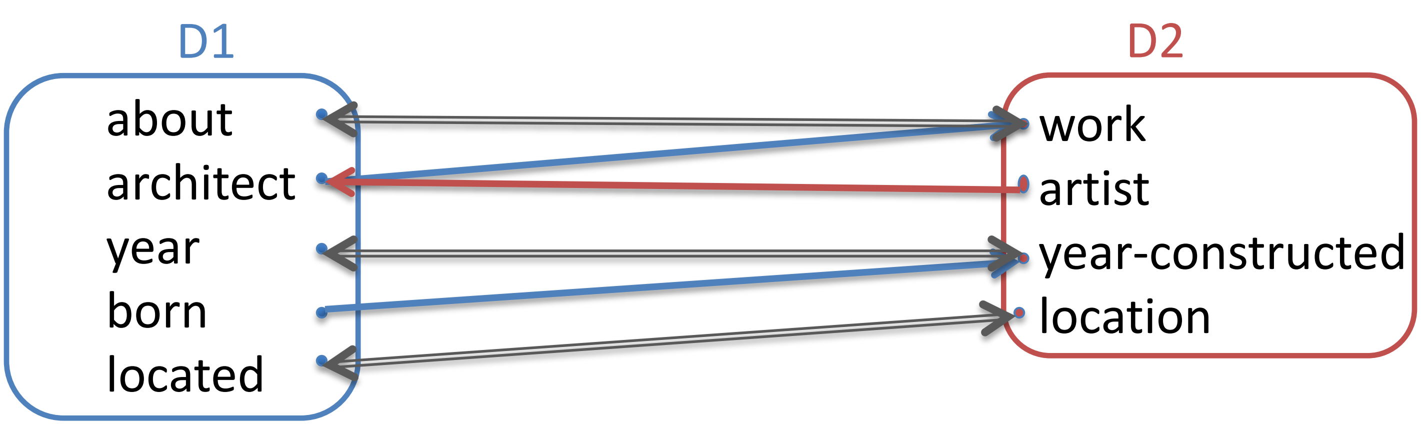
Clustering Entity Properties: Example

- Similarly for the rest of the properties…



Clustering Entity Properties: Example

- Similarly for the rest of the properties…



Clustering Entity Properties: Example

e_1= {(about, Eiffel Tower), (architect, Sauvestre), (year, 1889), (located, Paris)}

e_2= {(about, Statue of Liberty), (architect, Bartholdi Eiffel), (year, 1886), (located, NY)}

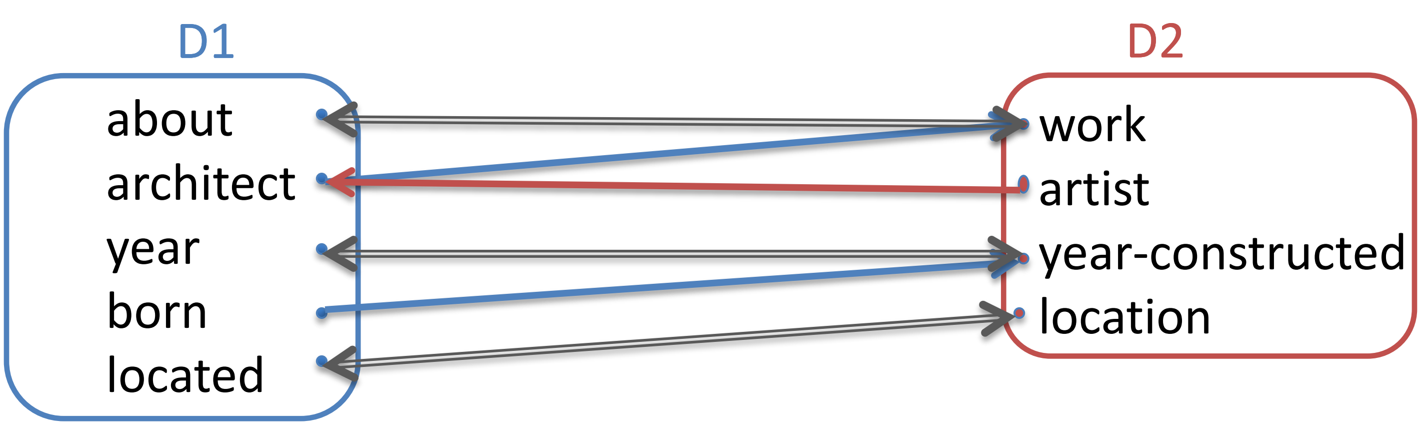
e_3= {(about, Auguste Bartholdi), (born, 1834)}

e_4= {(about, Joan Tower), (born, 1938)}

e_5= {(work, Lady Liberty), (artist, Bartholdi), (location, NY)}

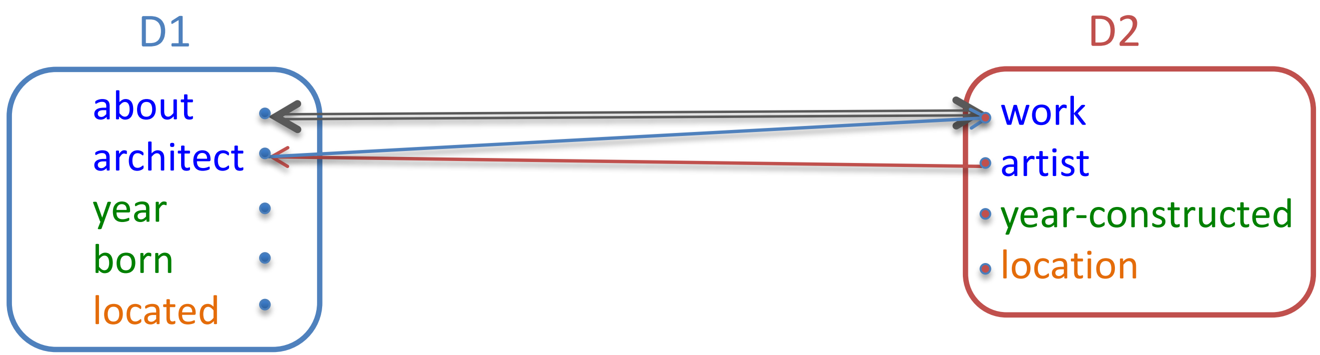
e_6= {(work, Eiffel Tower), (year-constructed, 1889), (location, Paris)}

e_7= {(work, Bartholdi Fountain), (year-constructed,1876), (location, Washington D.C.)}



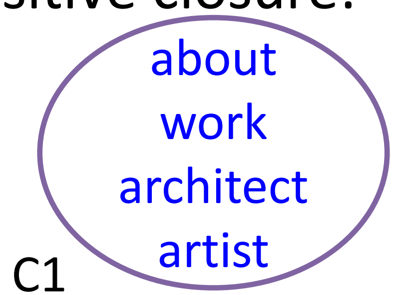
Clustering Entity Properties: Example

- Compute the transitive closure of the generated property name pairs
  - Connected properties form clusters



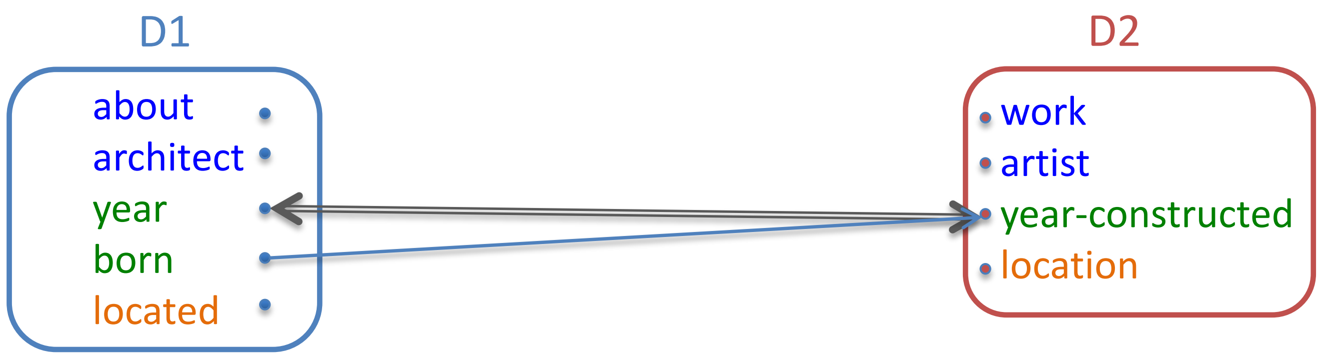
Pairs: (about, work), (work, about), (artist, architect), (architect, work)

Transitive closure:



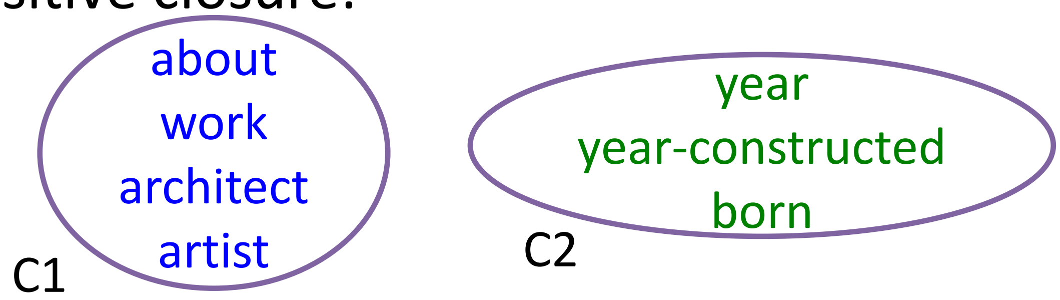
Clustering Entity Properties: Example

- Compute the transitive closure of the generated property name pairs
  - Connected properties form clusters



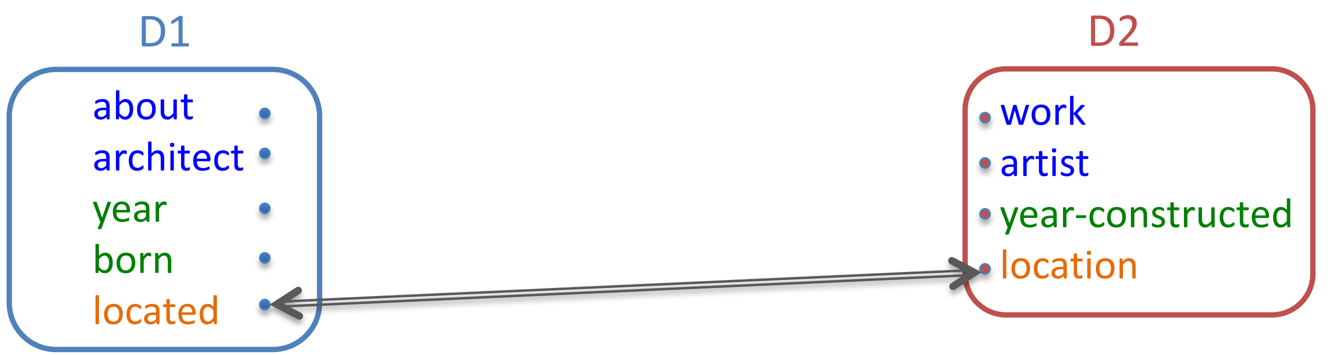
Pairs: (year, year-constructed), (year-constructed, year), (year-constructed, born)

Transitive closure:



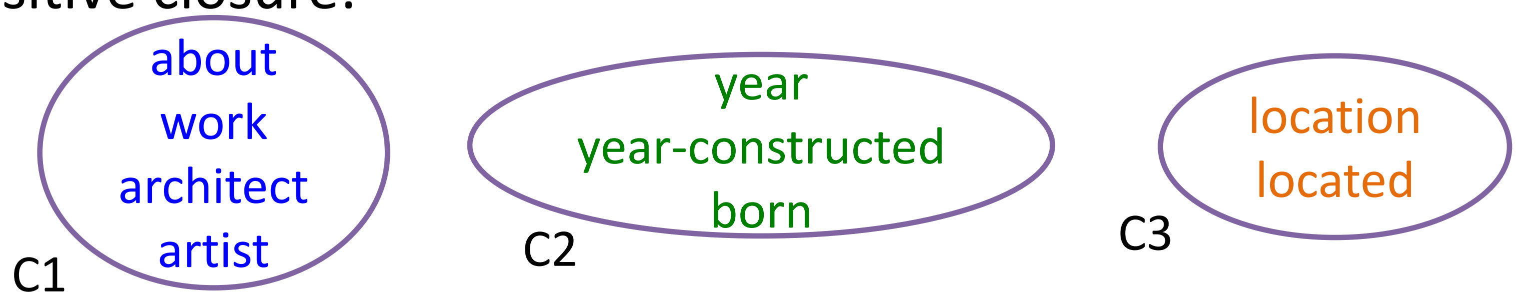
Clustering Entity Properties: Example

- Compute the transitive closure of the generated property name pairs
  - Connected properties form clusters



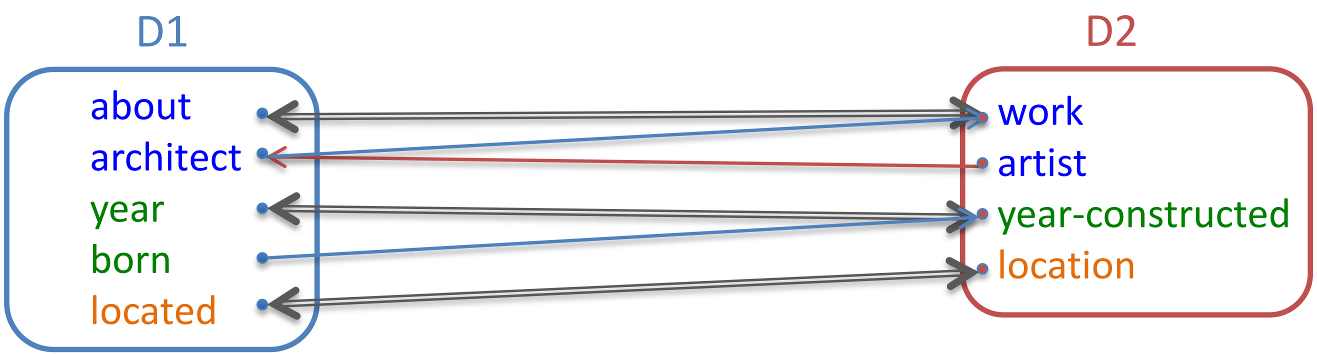
Pairs: (located, location), (location, located)

Transitive closure:

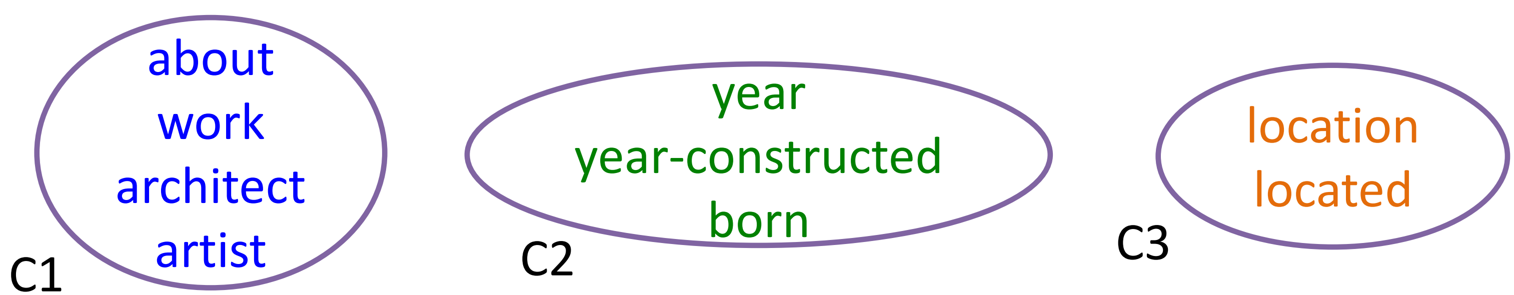


Clustering Entity Properties: Example

- Compute the transitive closure of the generated property name pairs
  - Connected properties form clusters



- Generated property clusters:



Token Blocking for Each Cluster

e_1= {(about, Eiffel Tower), (architect, Sauvestre), (year, 1889), (located, Paris)}

e_2= {(about, Statue of Liberty), (architect, Bartholdi Eiffel), (year, 1886), (located, NY)}

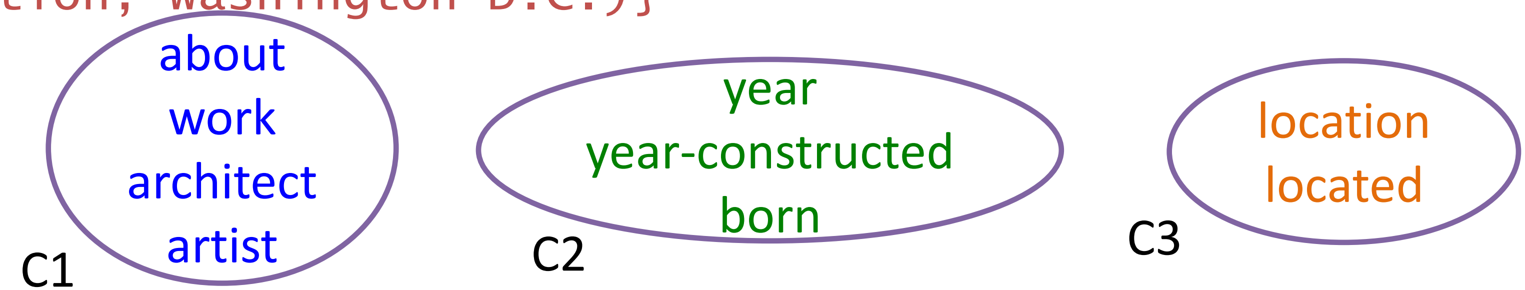
e_3= {(about, Auguste Bartholdi), (born, 1834)}

e_4= {(about, Joan Tower), (born, 1938)}

e_5= {(work, Lady Liberty), (artist, Bartholdi), (location, NY)}

e_6= {(work, Eiffel Tower), (year-constructed, 1889), (location, Paris)}

e_7= {(work, Bartholdi Fountain), (year-constructed,1876), (location, Washington D.C.)}



Some of the generated blocks:

C3.NY	C1.Tower	C1.Bartholdi
e_2, e_5	e_1, e_4, e_6	e_2, e_3, e_5, e_7

compare Lady Liberty to Auguste Bartholdi

Prefix-Infix(-Suffix) [Papadakis et al. 2012]

- How we can explore the semantics of URIs to better match entity descriptions?
  - E.g. 66% of the 182 million URIs of BTC09 (km.aifb.kit. edu/projects/btc-2009) follow the scheme: Prefix-Infix(-Suffix)
    - Prefix describes the source, i.e. domain, of the URI
    - Infix is a local identifier
    - The optional Suffix contains details about the format, e.g. .rdf and .nt, or a named anchor
- Token blocking on the Infixes appearing in the resource values of properties (as subject or object)

Prefix-Infix(-Suffix) [Papadakis et al. 2012]

E.g. (Infix-profile):

e1= {(skos:prefLabel, Statue of Liberty), (yago:isLocatedIn, yago:Liberty_Island)}

e2= {(rdfs:label, Statue of Liberty), (dbpprop:location, dbpedia:Liberty_Island)}

e3= {(freeb:official_name, Statue of Liberty), (freeb:containedby, freeb:m.026kp2)}

e4= {(geonames:name, Statue of Liberty), (geonames:nearby, geonames:5124330)}

yago select knowledge

DBpedia

Freebase

GeoNames

Generated blocks:

Liberty_Island	m.026kp2	5124330
e_1, e_2	e_3	e_4

Note: The effectiveness of the approach relies on the good naming practices of the data publishers

LINDA [Böhm et al. 2012]

- Works on an entity graph constructed from RDF triples by considering the URIs appearing in their subject, predicate and object positions

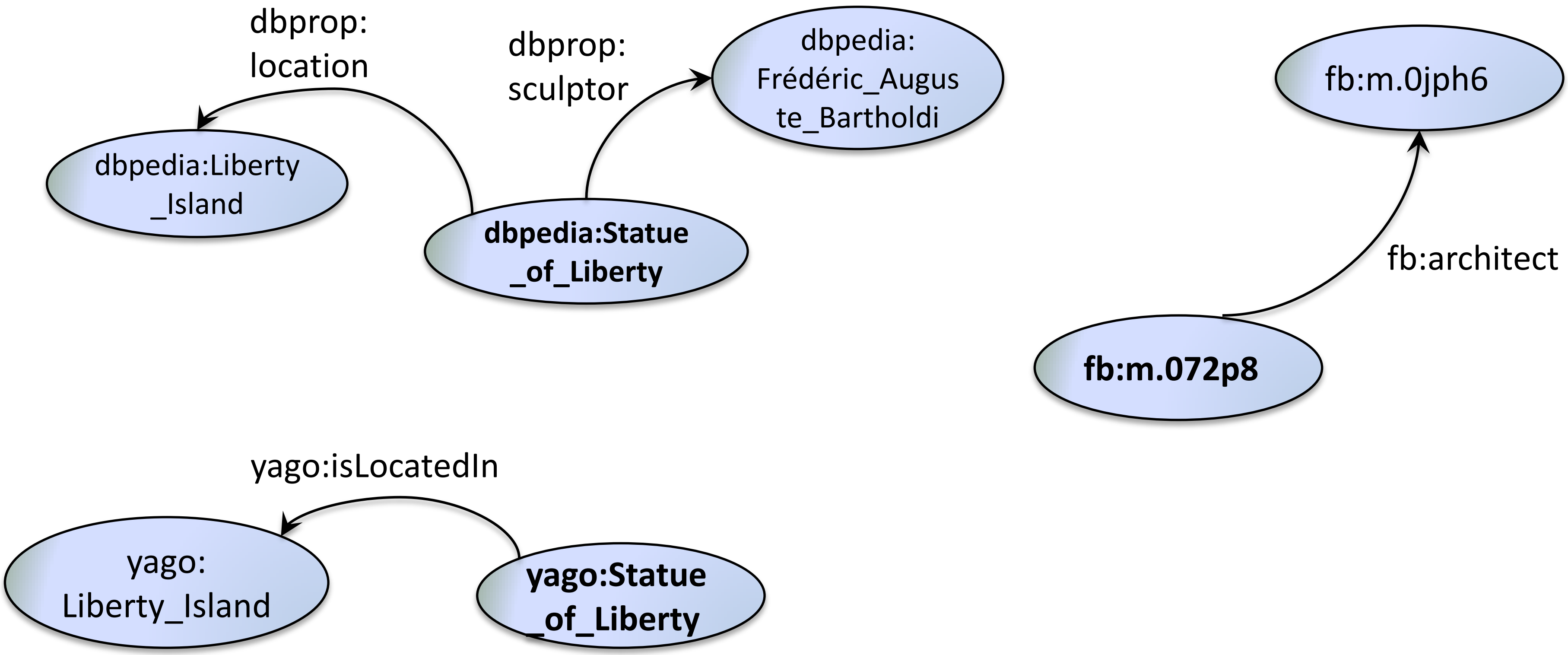
- Matches are identified using two kinds of similarities:
  - Descriptions are similar wrt. a string similarity of their literal values: *Checked once*
  - Descriptions have similar neighbours in the entity graph: *Checked iteratively*

LINDA [Böhm et al. 2012]

- Scalability: Entity graph partitions are processed in parallel
- Each Map/Reduce node holds:
  - A partition of the graph along with the similarities of the entity description pairs in this partition
    - description pairs are stored in a *priority queue* in descending order wrt. their similarity
      - *Fast merge-join-like access*
- Effectiveness: Messages from mappers to reducers, only for the pairs of descriptions that need similarity re-computation

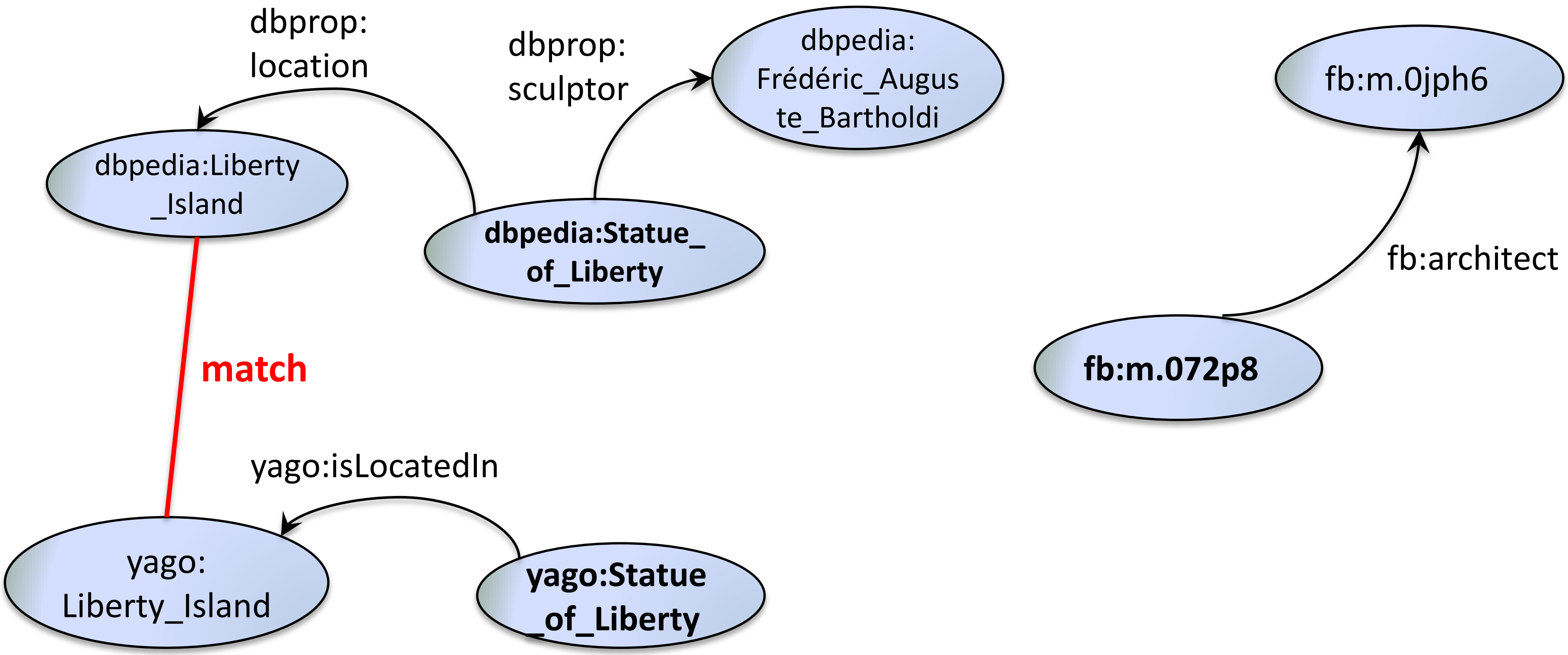
LINDA ER Algorithm

- Two matrices are used:
  - X captures the identified matches (binary matrix)
  - Y captures the pair-wise similarities (real values)
    - Initialization: common neighbors and string similarity of literals
    - Updates: Use the identified matches of X
- Until the priority queue (extracted from Y) becomes empty:
  - Get the pair (e_i, e_j) with the highest similarity
    - (e_i, e_j) match by default
      - Update X: matches of e_i are also matches of e_j
  - Update the queue wrt. the new matches



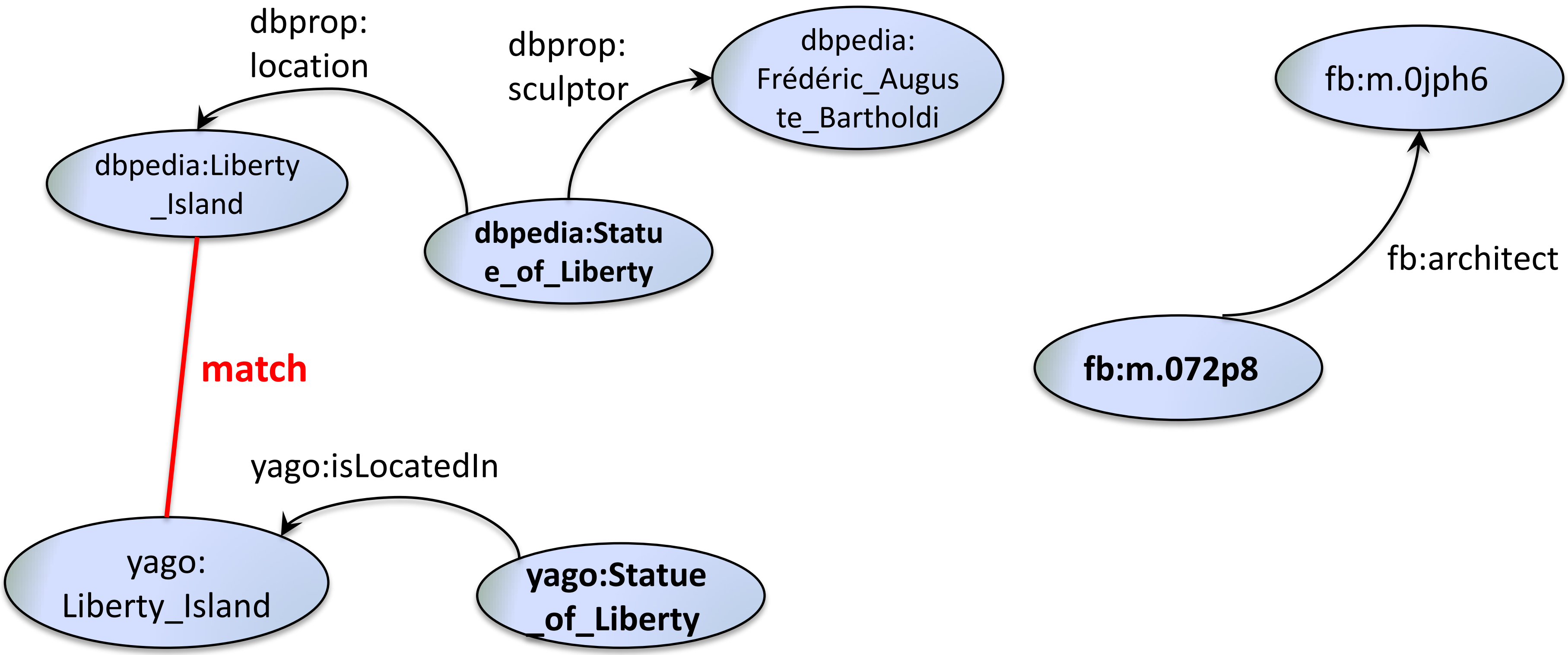
Priority Queue:

(dbpedia:Liberty_Island, yago:Liberty_Island)
(dbpedia:Statue_of_Liberty, yago:Liberty_Island)
(fb:m.072p8, dbpedia:Liberty_Island)



Priority Queue:

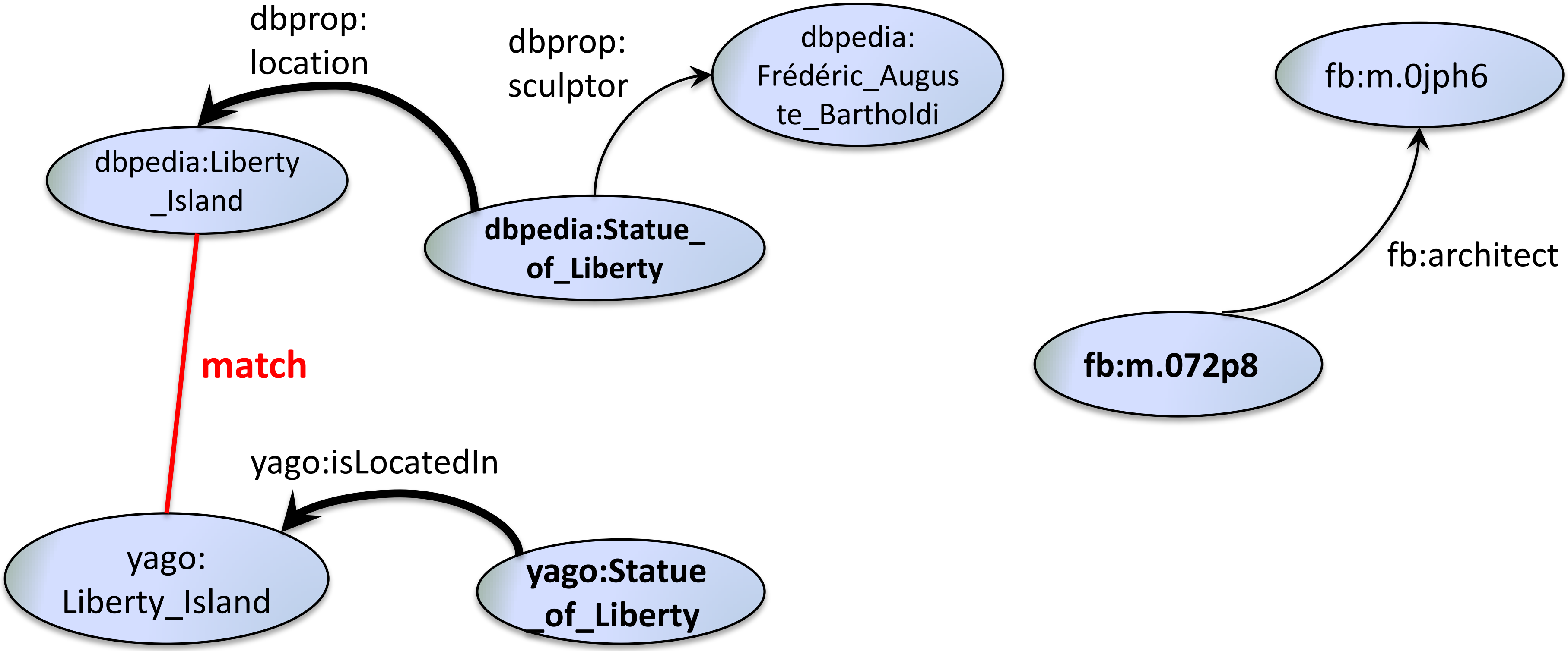
(dbpedia:Liberty_Island, yago:Liberty_Island)
(dbpedia:Statue_of_Liberty, yago:Liberty_Island)
(fb:m.072p8, dbpedia:Liberty_Island)



Priority Queue:

(dbpedia:Liberty_Island, yago:Liberty_Island)
~~(dbpedia:Statue_of_Liberty, yago:Liberty_Island)~~
~~(fb:m.072p8, dbpedia:Liberty_Island)~~

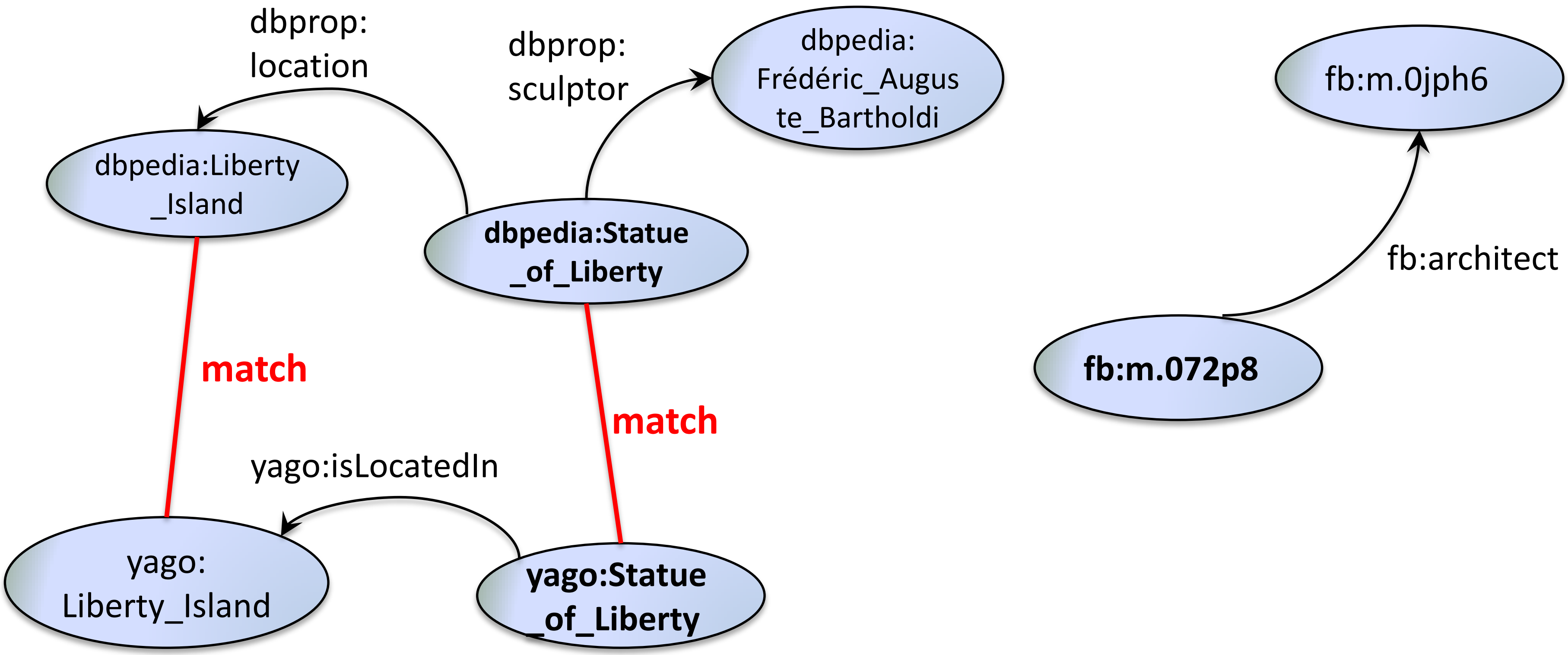
dequeue these pairs, as each entity can be mapped at most to one entity per data source →



Priority Queue:

enqueue this pair, as it is in the context of the newly-found match →

(dbpedia:Liberty_Island, yago:Liberty_Island)
(dbpedia:Statue_of_Liberty, yago:Statue_of_Liberty)



Priority Queue:

(dbpedia:Statue_of_Liberty, yago:Statue_of_Liberty)

LINDA

Distribute across a cluster the input entity graph

- A node i holds a portion Q_i of the priority queue and the respective part G_i of the graph

Map phase

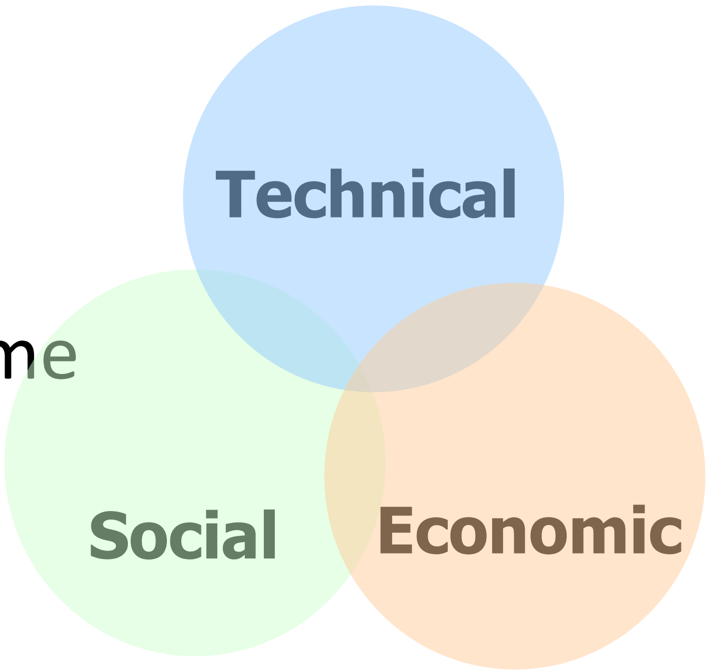
- Mapper i reads Q_i and forwards messages to reducers for similarities re-computations
  - Matrix X of identified matches is updated

Reduce phase

- Similarities re-computations (Matrix Y)
- Updates on priority queues

Frame Linked Data Preservation as a Sustainable Economic Activity

- Economic activity: deliberate allocation of resources
  - Cost of losing datasets
- Sustainable: ongoing resource allocation over long periods of time
  - Involved data subjects
- Articulate the problem/provide recommendations & guidelines
  - Economic and societal benefits



Sustainability Conditions

- Who benefits from use of the preserved data?
- Who selects what data to preserve?
- Who owns the data?
- Who preserves the data?
- Who pays both for data and preservation services?

- recognition of the benefits of preservation by decision makers
- selection of datasets with long-term value
- incentives for decision makers to act in the public interest or to elaborate new business models
- appropriate governance of preservation activities
- ongoing and efficient allocation of resources to preservation
- timely actions to ensure long-term data access and usability

Conclusions

- We need new abstractions bridging closer data creation, processing, publication and processing
  - Diachronic Data: Data annotated with temporal and provenance information self-describing their evolution history
  - Preserve (semi-)structured, interrelated, evolving data by keeping them constantly accessible & reusable from an open framework such as the Data Web
- We need new business models for spreading data publication and archiving costs among data stakeholders
  - Pay-as-you-go data preservation as data products are re-used through complex value making chains (both memory institutions and data market places)

Acknowledgements

- DIACHRON EU IP Project No 601043
  - http://www.diachron-fp7.eu
- PRELIDA EU CSA Project No 600663
  - http://www.prelida.eu

Collaborators

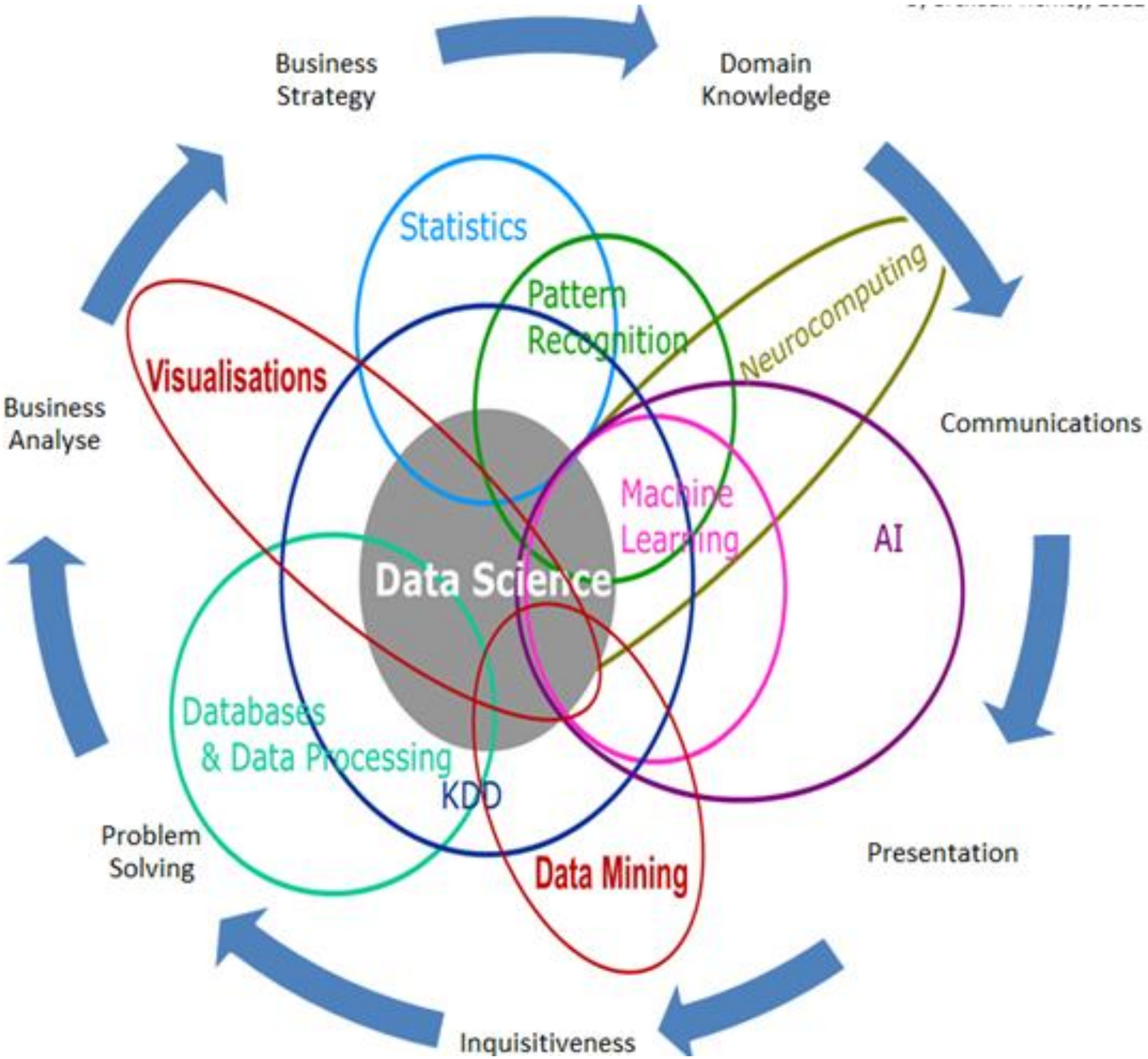
- Vassilis Efthimiou (University of Crete)
- Kostas Stefanidis (FORTH/ICS)
- Grigoris karvounarakis (LogicBox)
- Giorgos Flouris (FORTH/ICS)

Questions

References

- Bohm, C., de Melo, G., Naumann, F., Weikum, G.: Linda: distributed web-of-data-scale entity matching. In CIKM 2012
- Geerts, F., Karvounarakis, G., Christophides, V. and Fundulaki I.: Algebraic structures for capturing the provenance of SPARQL queries. In ICDT 2013.
- Papadakis, G., Ioannou, E., Niederee, C., Palpanas, T., Nejdl, W.: Beyond 100 million entities: large-scale blocking-based resolution for heterogeneous data. In WSDM 2012.
- Papadakis, G., Ioannou, E., Palpanas, T., Niederee, C., Nejdl, W.: A Blocking Framework for Entity Resolution in Highly Heterogeneous Information Spaces. IEEE Trans. Knowl. Data Eng. (2013) To appear
- Papavasileiou, V., Flouris, G., Fundulaki, I,. Kotzinos, D., Christophides, V. :High-level change detection in RDF(S) KBs. ACM Trans. Database Syst. 38, 1, April 2013
- High-Level Group on Scientific Data. “Riding the Wave: how Europe can gain from the raising tide of scientific data” © European Union, 2010 cordis.europa.eu/fp7/ict/e-infrastructure/docs/hlg-sdi-report.pdf
- Blue Ribbon Task Force on Sustainable Digital Preservation and Access, Final report 2010 brtf.sdsc.edu/biblio/BRTF_Final_Report.pdf

Data Science: A Multidisciplinary Challenge



Source www.oralytics.com/2012/06/data-science-is-multidisciplinary.html

Data Science Research Agenda

Acquisition, Storage, and Management of "Big Data"	Data Analytics	Data Sharing and Collaboration
Data representation, storage, and retrieval	Computational, mathematical, statistical, and algorithmic techniques for modeling high dimensional data	Tools for distant data sharing, real time visualization, and software reuse of complex data sets
New parallel data architectures, including clouds	Learning, inference, prediction, and knowledge discovery for large volumes of dynamic data sets	Cross disciplinary model, information and knowledge sharing
Data management policies, including privacy and secure access	Data mining to enable automated hypothesis generation, event correlation, and anomaly detection	Remote operation and real time access to distant data sources and instruments
Communication and storage devices with extreme capacities	Information infusion of multiple data sources	
Sustainable economic models for access and preservation		

Source Big Data R&D Initiative Howard Wactlar
NIST Big Data Meeting June, 2012

Towards Data Accountability