CptS 415 Big Data

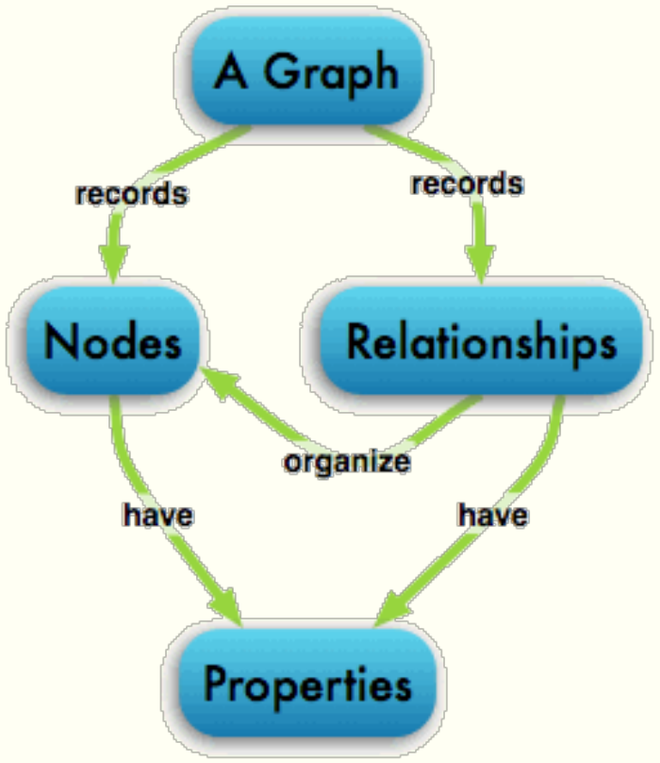
Graph and RDF

Srini Badri

Acknowledgements: Tinghui Wang

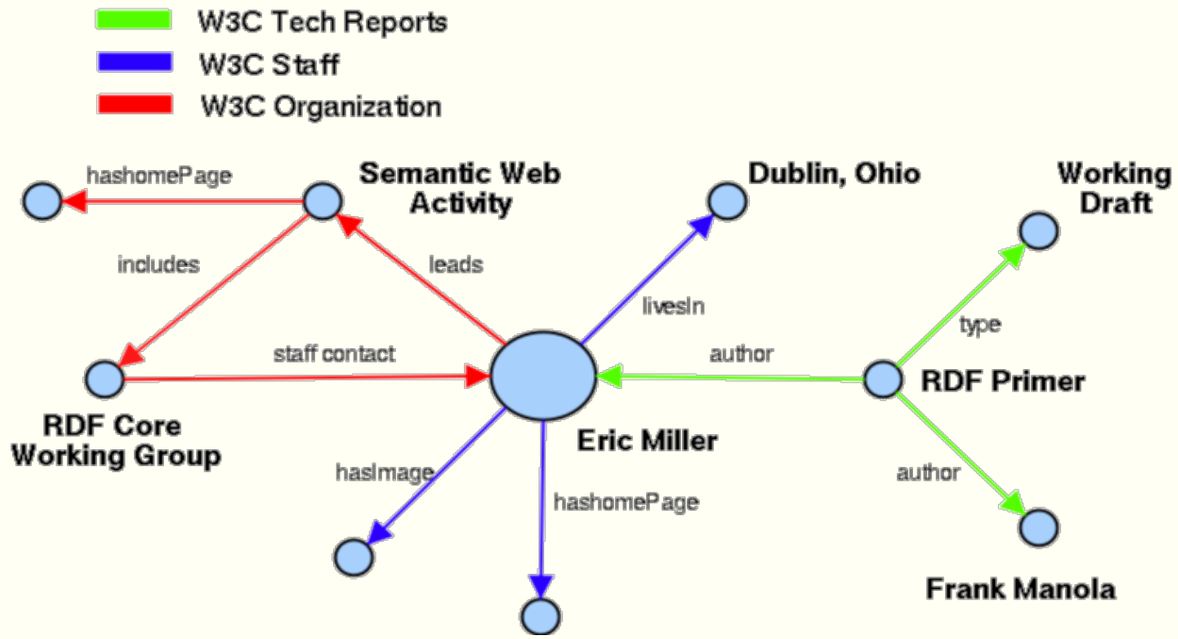
What is a Graph?

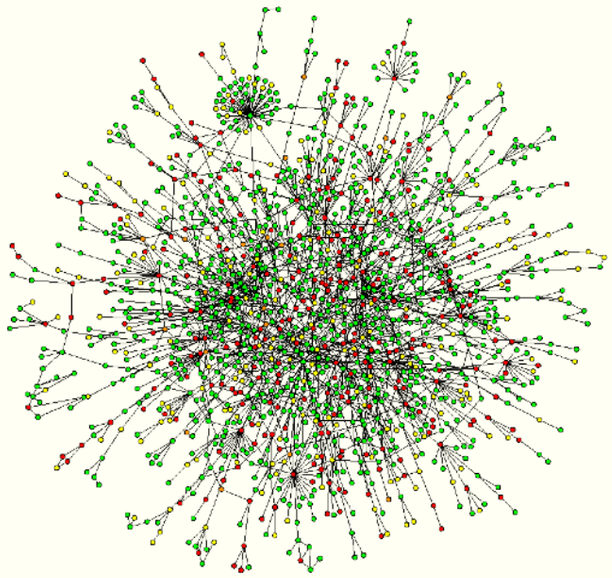
- $G = (V,\ E)$
  - V: a set of vertices (nodes) in the graph
  - E: a set of edges (links, relationships) in the graph
  - Both vertices and edges may contain additional information
- Type of graph
  - Directed Vs. Undirected
  - Simple Vs. Multi-graph
  - Weighted Vs. Unweighted
- Networks, Linked data, Web, …



Ubiquitous Network (Graph) Data

- Social Network
- Biological Network
- Road Network/Map
- WWW
- Sematic Web/Ontologies
- XML/RDF
- ….





Can We Use XML?

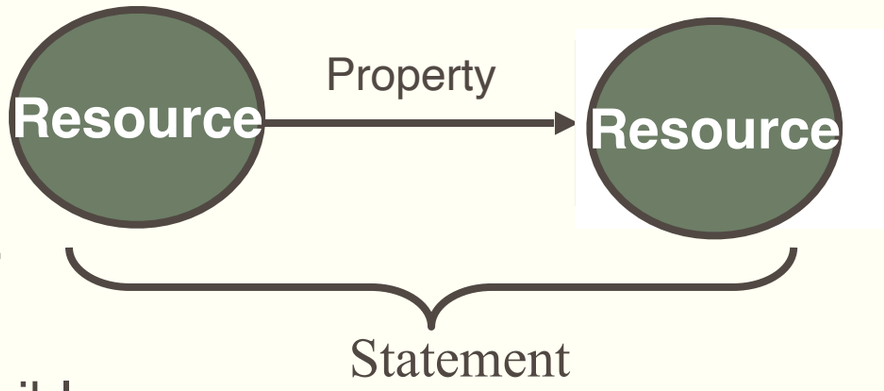
- XML provides a uniform framework for interchange of data and metadata between applications
- However, XML does not provide any means of the semantics (meaning) of data
  - E.g., there is no intended meaning associated with the nesting of tags
    - It is up to each application to interpret the nesting.

Resource Description Framework (RDF)

- Developed by the World Wide Web Consortium (W3C)
- Provide a standard for defining an architecture for supporting the vast amount of web metadata
- Human and Machine readable
  - Machine readable: it maintains the structure of data
- History:
  - Metadata: begins in 1995
  - Platform of Internet Content Selection (PICS)
    - Mechanism for communicating ratings of web pages from server to clients
  - Interned resource description based on PICS architecture
  - PIC-NG working group: RDF, in 2004

Basic Ideas of RDF

- Basic building block: Statement
  - It is an object-attribute-value triple
  - A.k.a.: Subject-predicate-object
- RDF has been given a syntax in XML
  - Inherits the benefits of XML
  - Other syntactic representations of RDF possible



fundamental concepts

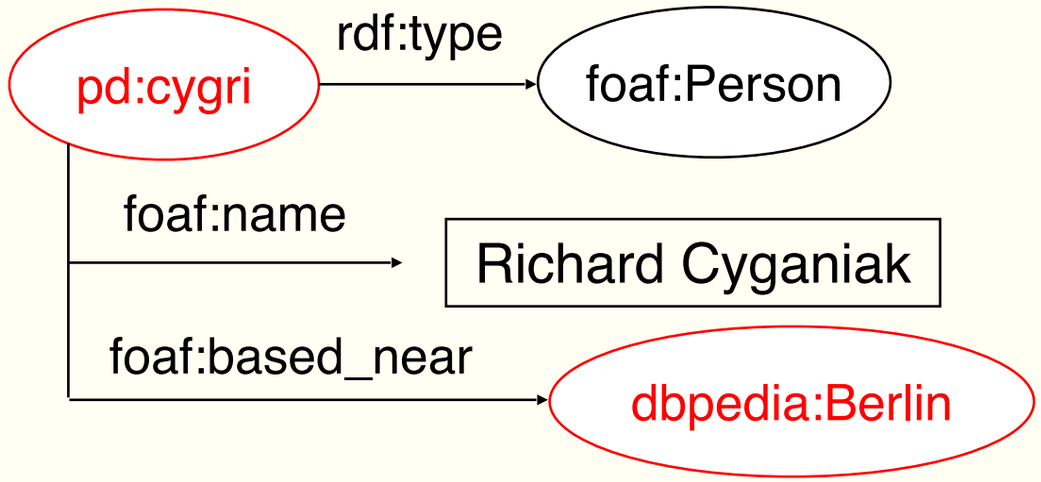
Resources

- We can think of a resource as an object
  - E.g. authors, books, publishers, places, people, hotels
- Every resource has a Universal Resource Identifier (URI)
- A URI can be
  - A Universal Resource Locator (URL: Web address)
  - Some kind of unique identifiers (e.g. ISBN)

URIs are a foundation

- URI = Uniform Resource Identifier
  - "The generic set of all names/addresses that are short strings that refer to resources"
  - URLs (Uniform Resource Locators) are a subset of URIs, used for resources that can be *accessed* on the web
- URIs look like URLs, often with fragment identifiers pointing to a document part:
  - http://foo.com/bar/mumble.html#pitch

Resources identified with HTTP URIs



dbpedia:Berlin = http://dbpedia.org/resource/Berlin

pd:cygri = http://richard.cyganiak.de/foaf.rdf#cygri

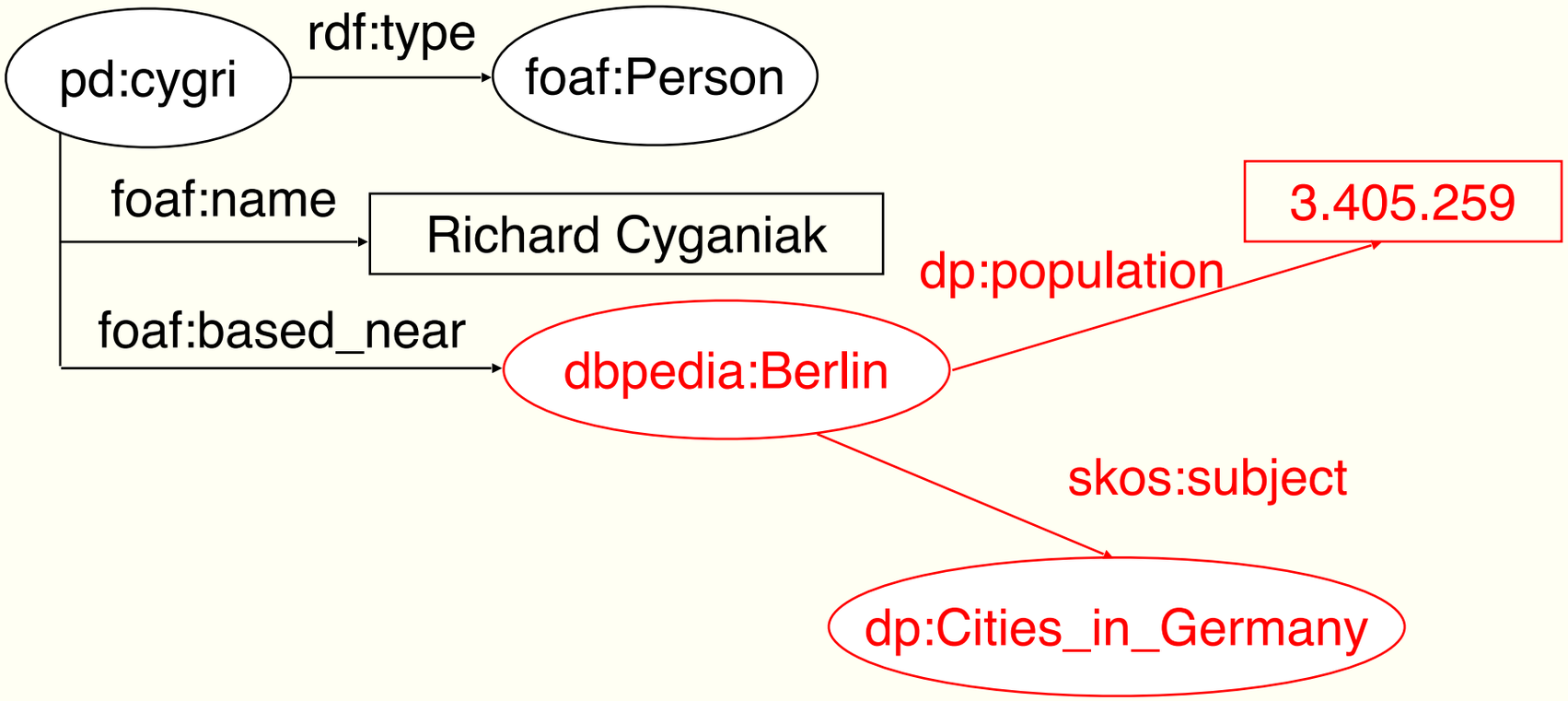
Properties

- Properties are a special kind of resources
- Describe relations between resources
  - E.g. “written by”, “age”, “title”, etc.
- Properties are also identified by URIs
- Advantages of using URIs:
  - A global, worldwide, unique naming scheme
  - Reduces the homonym problem of distributed data representation

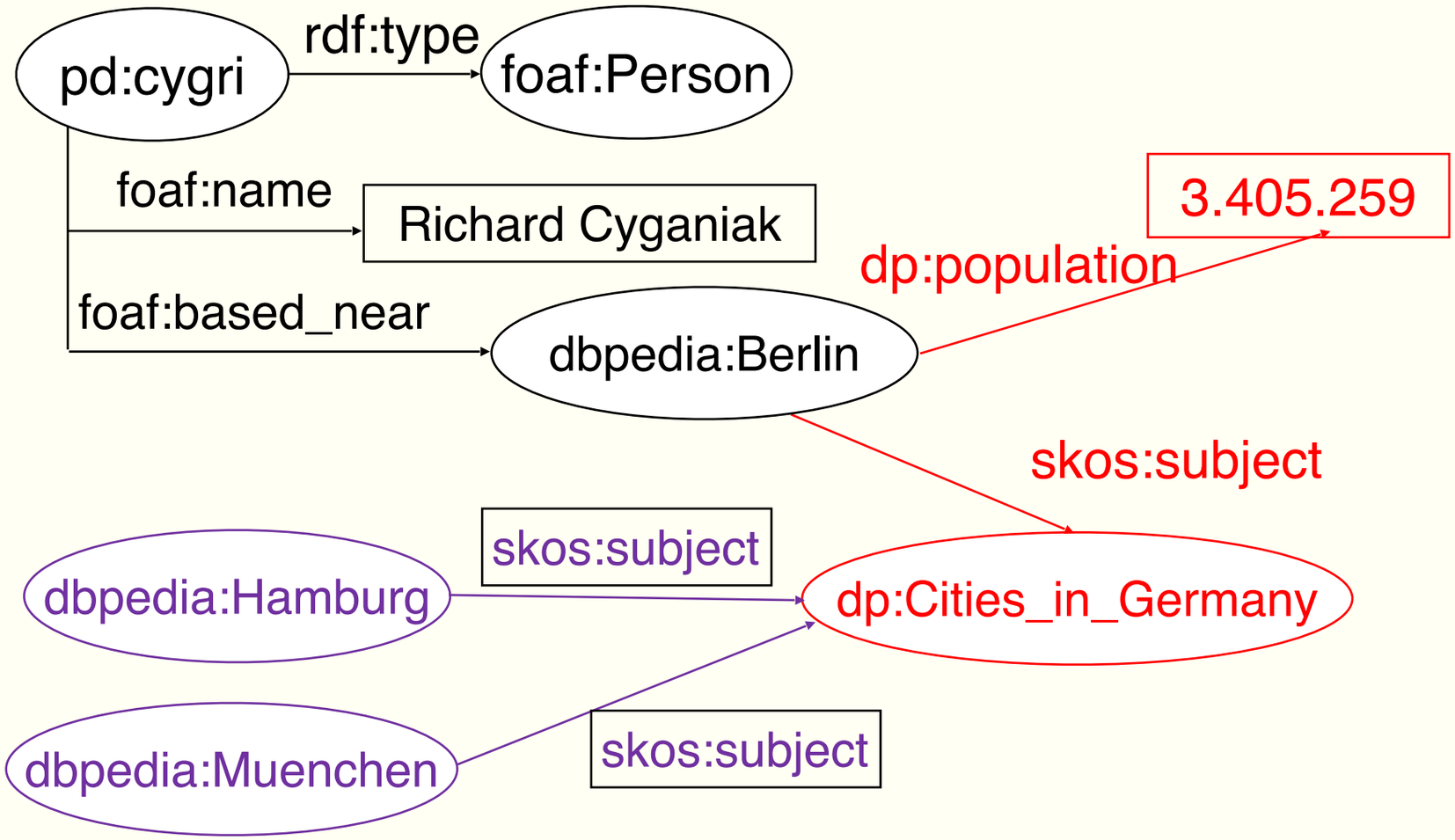
Statements

- Statements assert the properties of resources
  - It consists of a resource, a property and a “value”
- Values can be resources or literals
  - Literals are atomic values (strings)
- A statement can be viewed as
  - A Triple
  - A piece of graph
  - A piece of XML code
- Hence, an RDF document is
  - A set of triples
  - A graph (semantic Web)
  - An XML document

Resolving URIs over the Web



Dereferencing URIs over the Web



XML-based Syntax of RDF

- An RDF document consists of an rdf:RDF element
  - The content of that element is a number of descriptions

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-
syntax-ns#"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns:uni="http://www.mydomain.org/uni-ns">
    <rdf:Description rdf:about="949356">
        <uni:name>Srini Badri</uni:name>
        <uni:title>Instructor</uni:title>
        <uni:office rdf:datatype="&xsd:string">Remo
te<uni:office>
    </rdf:Description>
    <rdf:Description rdf:about="CPTS 415">
        <uni:courseName>Big Data</uni:courseName>
        <uni:isTaughtBy>Srini Badri</
uni:isTaughtBy>
    </rdf:Description>
```

Property Elements

- Content of rdf:Description elements

```
    <rdf:Description rdf:about="CPTS 415">
        <uni:courseName>Big Data</
uni:courseName>
        <uni:isTaughtBy>Srini Badri</
uni:isTaughtBy>
    </rdf:Description>
```

- uni:courseName and uni:isTaughtBy defines two property-value pairs for CPTS 415 (two RDF statements)

The rdf:resource Attribute

- We can denote that two entities are the same using the

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-
ns#"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns:uni="http://www.mydomain.org/uni-ns">
    <rdf:Description rdf:about="949356">
        <uni:name>Srini Badri</uni:name>
        <uni:title>Instructor</uni:title>
        <uni:office rdf:datatype="&xsd:string">Remote<uni
:office>
    </rdf:Description>
    <rdf:Description rdf:about="CPTS 415">
        <uni:courseName>Big Data</uni:courseName>
        <uni:isTaughtBy rdf:resource="949356" />
    </rdf:Description>
</rdf:RDF>
```

RDF Containers

- Collect a number of resources or attributes about which we want to make a statement as a whole
- Permit aggregation of several values for a property
- Different container semantics
  - Bag (rdf:Bag)
    - Unordered grouping (e.g. students in the class)
  - Sequence (rdf:Seq)
    - Ordered grouping (e.g. authors of a paper)
  - Alternatives (rdf: Alt)
    - Alternative values (e.g. measurement in different units)

Example of a Bag and Alternative

```
<uni:lecturer rdf:ID="949356" uni:name=“Srini Badri"
              uni:title= "Instructor">
    <uni:coursesTaught>
        <rdf:Bag>
            <rdf:_1 rdf:resource="#CPTS 131”/>
            <rdf:_2 rdf:resource="#CPTS 215"/>
            <rdf:_3 rdf:resource="#CPTS 415"/>
        </rdf:Bag>
    </uni:coursesTaught>
</uni:lecturer>
<uni:course rdf:ID="CPTS 415" uni:courseName="Big Data">
    <uni:lecturer>
        <rdf:Alt>
            <rdf:li rdf:resource=“#949356”/>
            <rdf:li rdf:resource=“#949352"/>
        </rdf:Alt>
    </uni:lecturer>
</uni:course>
```

Reification

- Sometimes, one want to make statements about other statements
- Idea:
  - Refer to a statement using an identifier
- RDF allows such reference through a reification mechanism which turns a statement into a resource

Example: Tom said his father is out

Reification

- To accesss parts of a statement:
- Properties
  - rdf:type - subject is an instance of that category or class defined by the value
  - rdf:subject, rdf:predicate, rdf:object – relate elements of statement tuple to a resource of type statement.
- Types (or classes)
  - rdf:Resource – everything that can be identified (with a URI)
  - rdf:Property – specialization of a resource expressing a binary relation between two resources
  - rdf:statement – a triple with properties rdf:subject, rdf:predicate, rdf:object

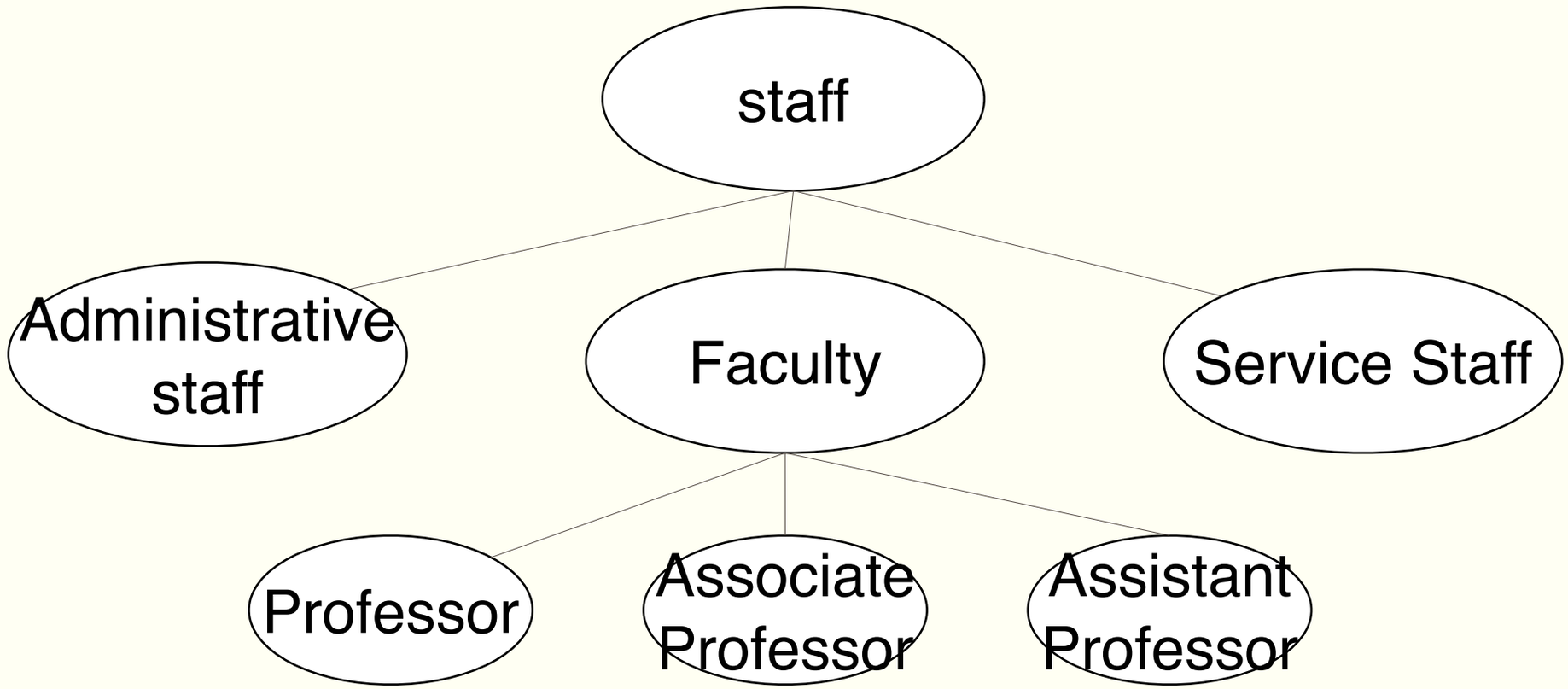
RDF Schema

- RDF is a universal language that lets users describe resources in their own vocabularies
  - RDF does not assume, nor does it define semantics of any particular application domain
- The user can do so in RDF Schema using:
  - Classes and Properties
  - Class Hierarchies and Inheritance
  - Property Hierarchies
- Enables communities to share machine readable tokens and locally define human readable labels.

Classes and Their Instances

- Distinguish between
  - Concrete “Things” (individual objects) in the domain: e.g. Big data, Tinghui Wang, etc.
  - Sets of individuals sharing properties called classes: e.g. Lectures, students, courses, etc.
- Individual objects that belong to a class are referred to as instances of that class
- The relationship between instances and classes in RDF is through rdf:type
  - So that the following statements are not allowed:
  - “Big data is taught by Graph Theory”
  - “Sloan 150 is taught by Big Data”

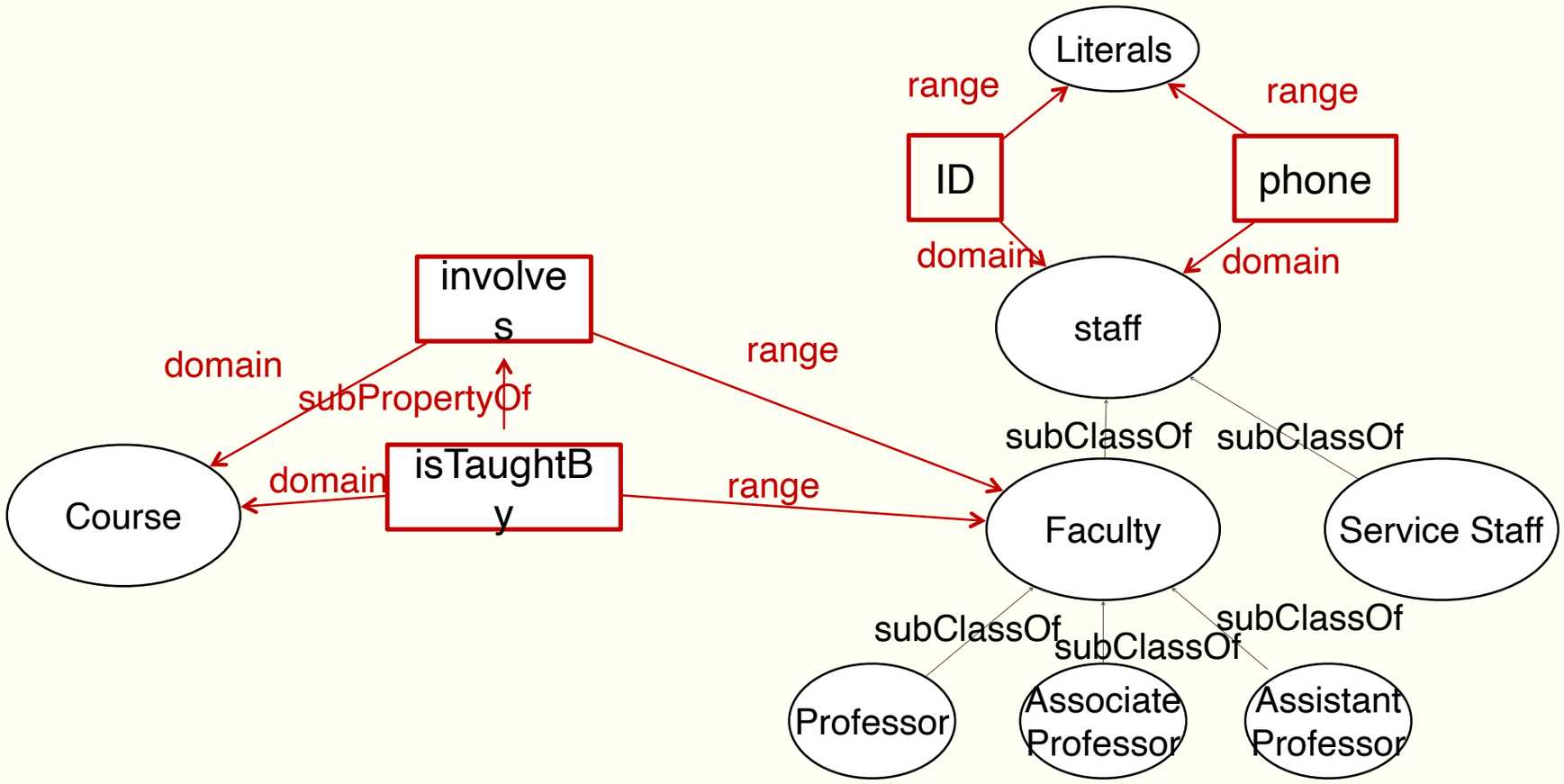
Class Hierarchy Example



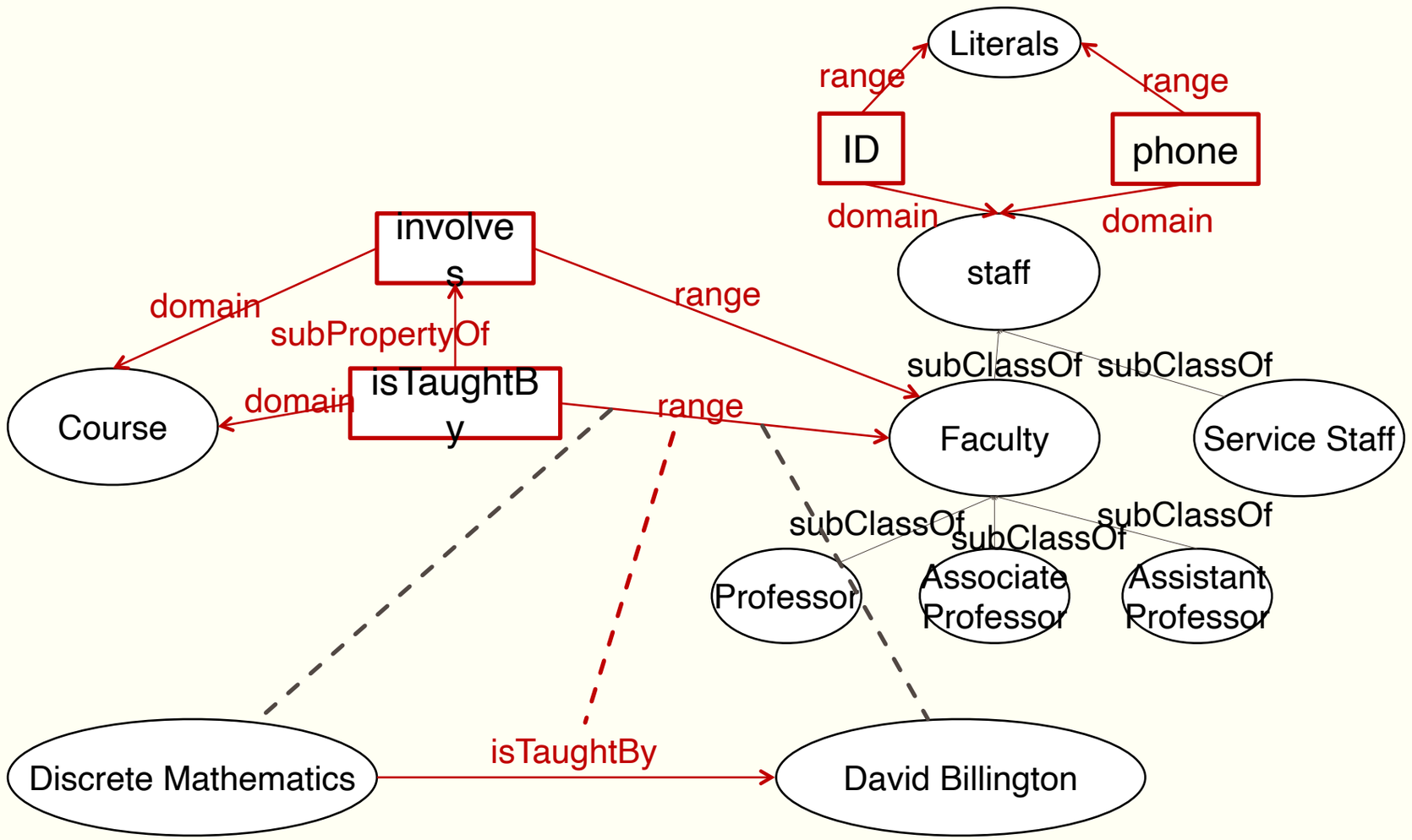
Property Hierarchies

- Property-related namespace
  - rdfs:subPropertyOf
  - rdfs:domain
  - rdfs:range
- Hierarchical relationships for properties
  - E.g. “isTaughtBy” is a subproperty of “involves”
  - In another word, if a course is taught by a faculty member, the course also involves the faculty member
- The converse is not necessarily true
  - E.g. A course involves a person, but the course may not be taught by that person.

Example: RDF Schema



Example: RDF Layer Vs. RDF Schema Layer



RDF Schema

- The modeling primitives of RDF schema are defined using resources and properties (an RDF!)
- To declare that “associate professor” is a subclass of “faculty”
  - Define resources “associate professor” and “faculty”
  - Define property subClassOf
  - Write triple (lecturer, subclassOf, faculty)

Example: RDF Schema 1

```
<rdfs:Class rdf:ID="faculty" />
<rdfs:Class rdf:ID="associatedProfessor">
    <rdfs:comment>
        The class of associate professors.
        All associate professors are faculties.
    </rdfs:comment>
    <rdfs:subClassOf rdf:resource="#faculty"/>
</rdfs:Class>
<rdfs:Class rdf:ID="course">
    <rdfs:comment>The class of courses</
rdfs:comment>
</rdfs:Class>
```

Example: RDF Schema 2

```
<rdf:Property rdf:ID="involves">
    <rdf:domain rdf:resource="#course" />
    <rdf:range rdf:resource="#faculty" />
</rdf:Property>
<rdf:Property rdf:ID="taughtBy">
    <rdfs:comment>
        Inherits its domain ("course") and range ("faculty")
        from its superproperty "involves"
    </rdfs:comment>
    <rdfs:subPropertyOf rdf:resource="#involves"/>
</rdf:Property>
```

RDF Schema: Core Classes

Class name	comment
rdfs:Resource	The class resource, everything.
rdfs:Literal	The class of literal values, e.g. textual strings and integers.
rdf:langString	The class of language-tagged string literal values.
rdf:HTML	The class of HTML literal values.
rdf:XMLLiteral	The class of XML literal values.
rdfs:Class	The class of classes.
rdf:Property	The class of RDF properties.
rdfs:Datatype	The class of RDF datatypes.
rdf:Statement	The class of RDF statements.
rdf:Bag	The class of unordered containers.
rdf:Seq	The class of ordered containers.
rdf:Alt	The class of containers of alternatives.
rdfs:Container	The class of RDF containers.
rdfs:ContainerMembershipProperty	The class of container membership properties, rdf:_1, rdf:_2, ..., all of which are sub-properties of 'member'.
rdf:List	The class of RDF Lists.

https://www.w3.org/TR/2014/REC-rdf-schema-20140225

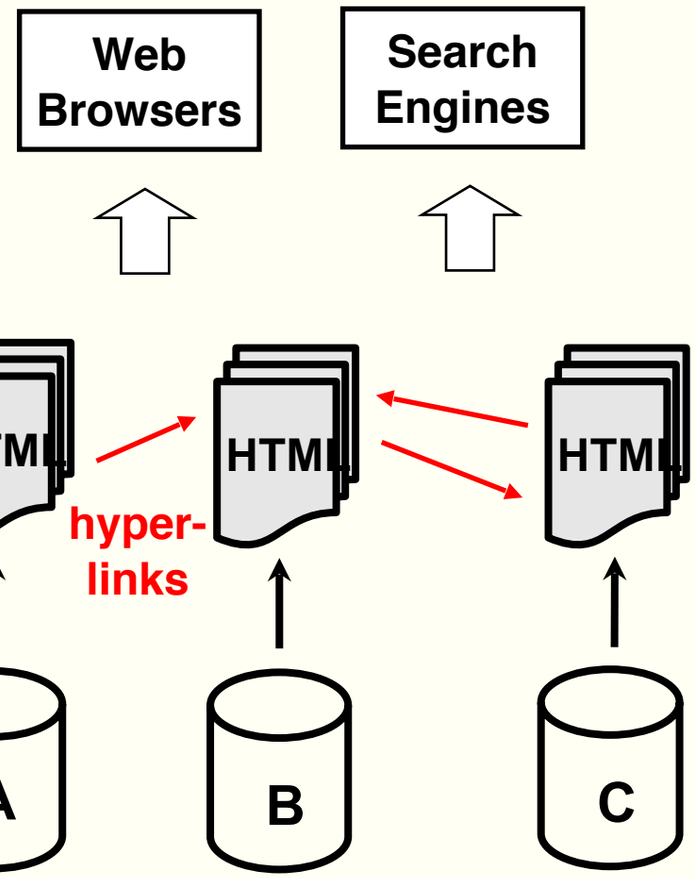
RDF Schema: Core Properties

Property name	comment	domain	range
rdf:type	The subject is an instance of a class.	rdfs:Resource	rdfs:Class
rdfs:subClassOf	The subject is a subclass of a class.	rdfs:Class	rdfs:Class
rdfs:subPropertyOf	The subject is a subproperty of a property.	rdf:Property	rdf:Property
rdfs:domain	A domain of the subject property.	rdf:Property	rdfs:Class
rdfs:range	A range of the subject property.	rdf:Property	rdfs:Class
rdfs:label	A human-readable name for the subject.	rdfs:Resource	rdfs:Literal
rdfs:comment	A description of the subject resource.	rdfs:Resource	rdfs:Literal
rdfs:member	A member of the subject resource.	rdfs:Resource	rdfs:Resource
rdf:first	The first item in the subject RDF list.	rdf:List	rdfs:Resource
rdf:rest	The rest of the subject RDF list after the first item.	rdf:List	rdf:List
rdfs:seeAlso	Further information about the subject resource.	rdfs:Resource	rdfs:Resource
rdfs:isDefinedBy	The definition of the subject resource.	rdfs:Resource	rdfs:Resource
rdf:value	Idiomatic property used for structured values.	rdfs:Resource	rdfs:Resource
rdf:subject	The subject of the subject RDF statement.	rdf:Statement	rdfs:Resource
rdf:predicate	The predicate of the subject RDF statement.	rdf:Statement	rdfs:Resource
rdf:object	The object of the subject RDF statement.	rdf:Statement	rdfs:Resource

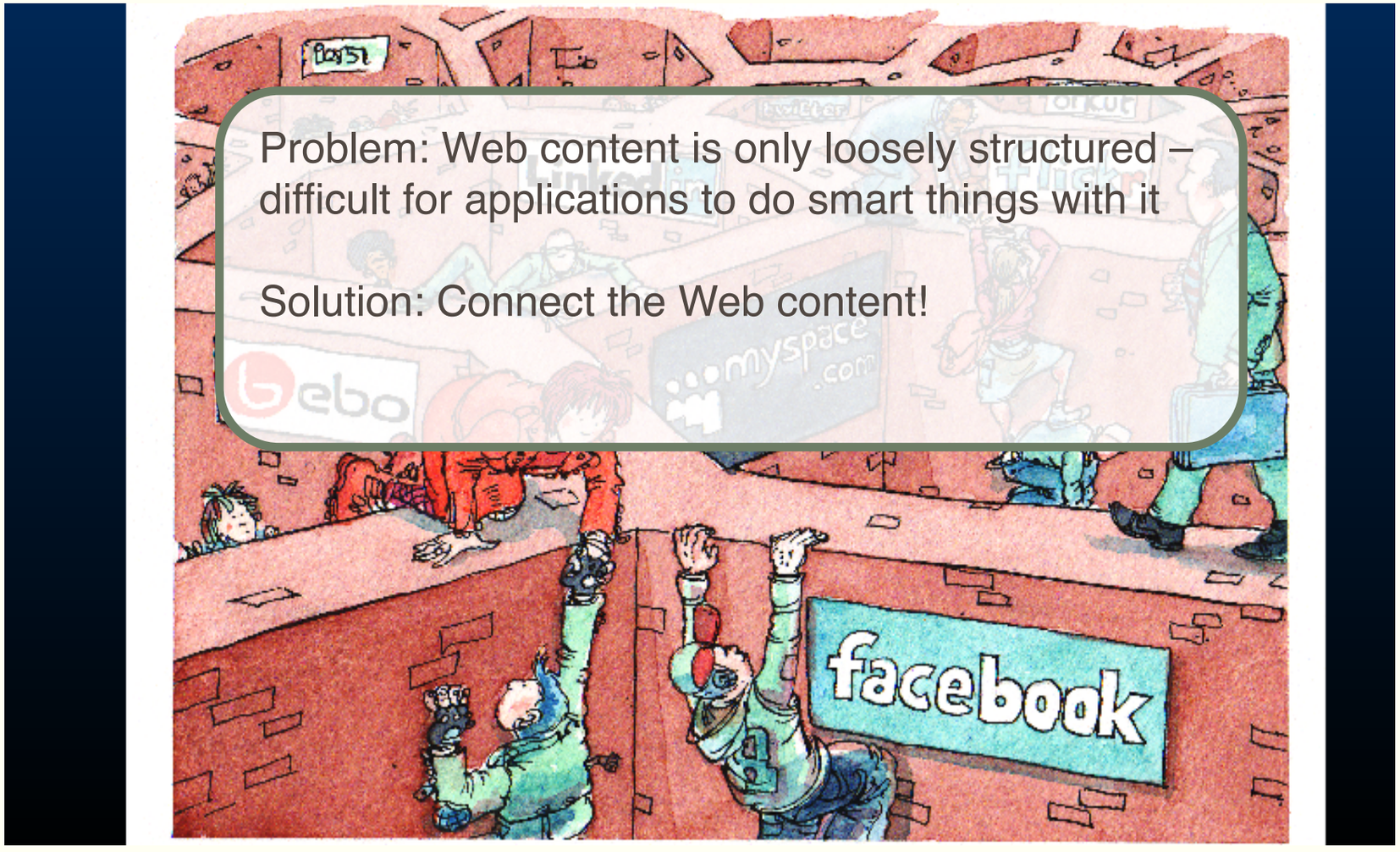
https://www.w3.org/TR/2014/REC-rdf-schema-20140225

RDF: Classic Web

- Single Global Information Space
- URLs as
  - Global unique IDs
  - Retrieval mechanism
- HTML as shared content format
- Hyperlinks

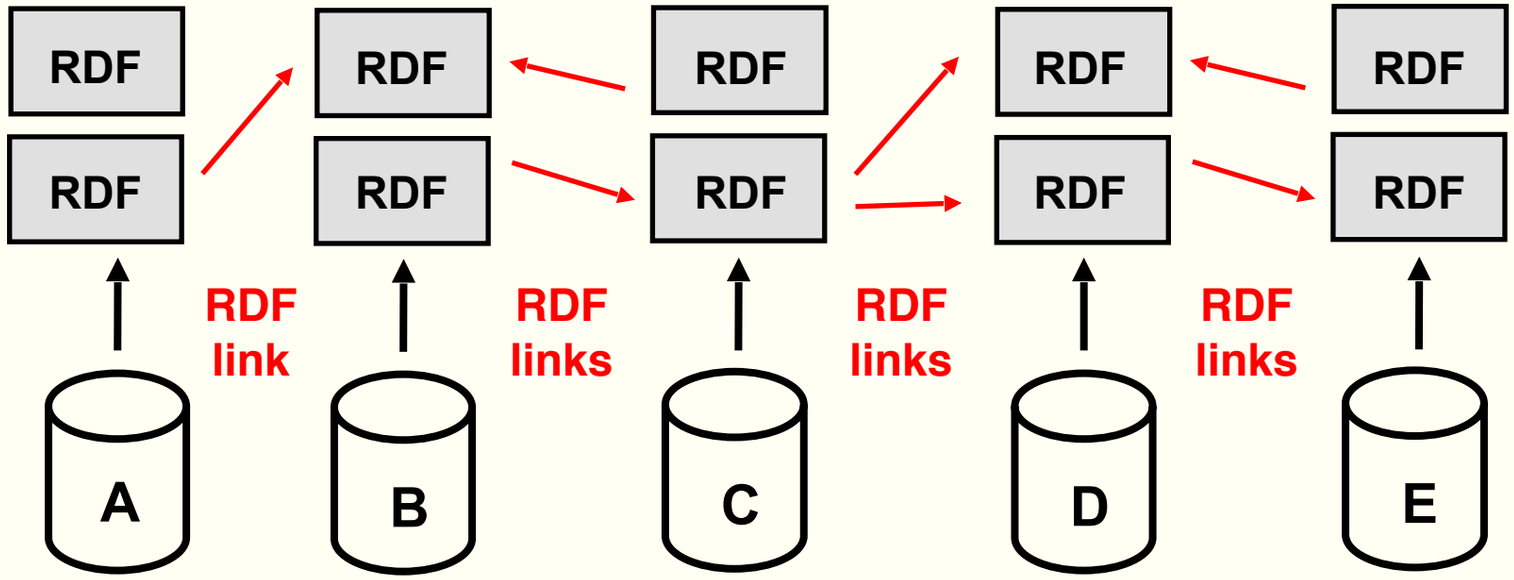


Background: The rise of linked data

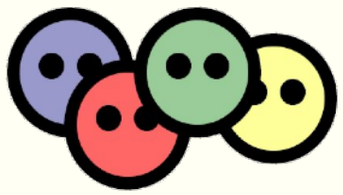


The Idea of Linked Data

- Use Semantic Web technologies to publish structured data on the Web,
- Set links between data from one data source to data within other data sources.



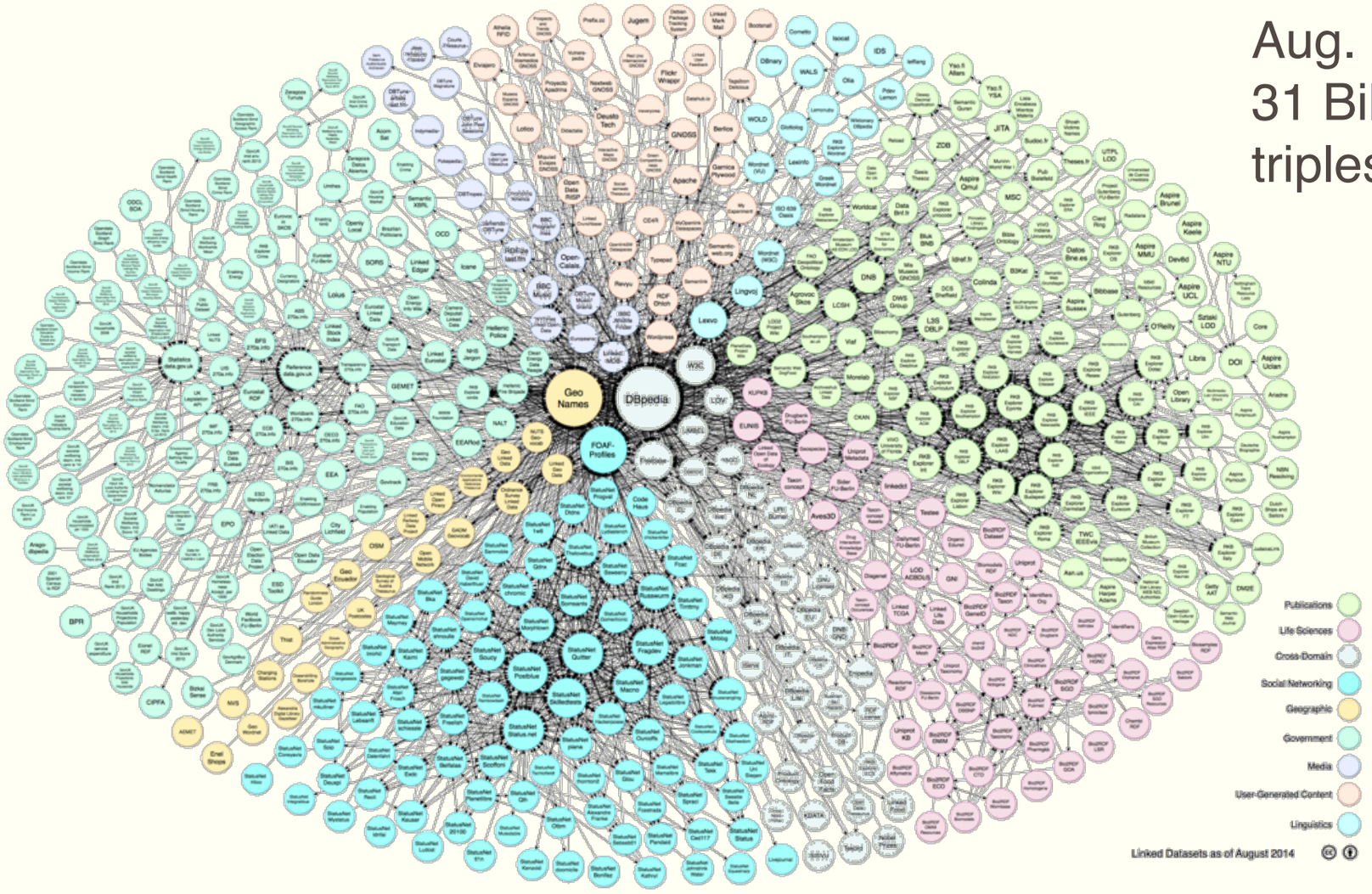
Friend of a Friend (FOAF)



- FOAF (Friend of a Friend) is a simple ontology to describe people and their social networks.
  - the foaf project page: http://www.foaf-project.org/
- In 2008: over 1,000,000 valid RDF FOAF files.
  - Most of these are from the http://liveJournal.com/ blogging system which encodes basic user info in foaf
  - See http://apple.cs.umbc.edu/semdis/wob/foaf/

```
<foaf:Person>
    <foaf:name>Tim Finin</foaf:name>
    <foaf:mbox_sha1sum>2410…37262c252e</foaf:mbox_sha1sum>
    <foaf:homepage rdf:resource="http://umbc.edu/~finin/" />
    <foaf:img rdf:resource="http://umbc.edu/~finin/images/
passport.gif" />
</foaf:Person>
```

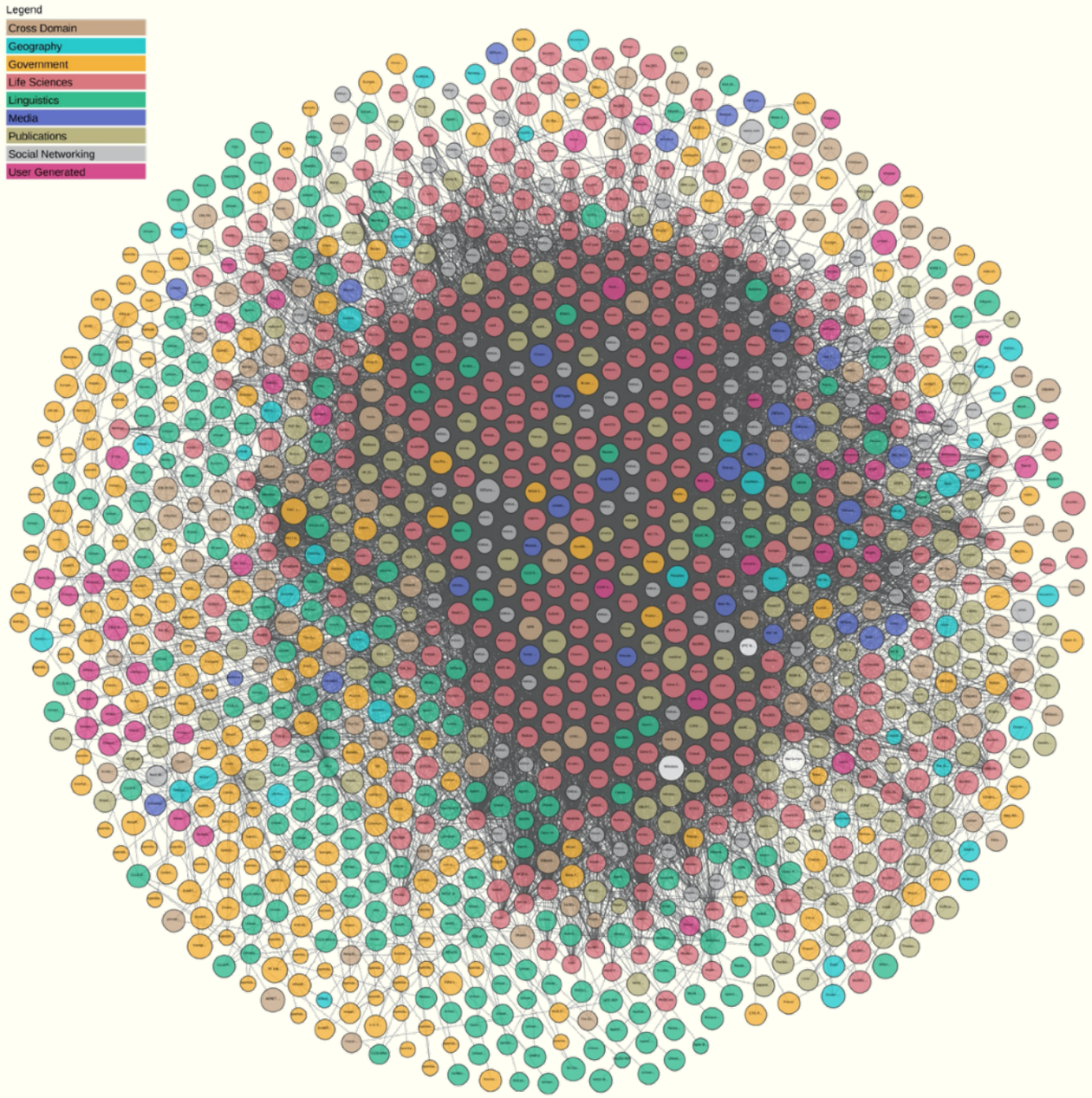
LOD Cloud on the web



Aug. 2014:
31 Billion
triples

Now

1,260 datasets, 16,187 links



The Linked Open Data Cloud from lod-cloud.net

Q: Is RDFs better than XML?

XML

- a tree, i.e., a strong hierarchy
- applications may rely on hierarchy position
- relatively simple syntax and structure
- not easy to combine trees

RDF

- a loose collections of relations
- applications may do database-like search
- not easy to recover hierarchy
- easy to combine relations in one big collection
- great for the integration of heterogeneous information

Summary

- RDF provides a foundation for representing and processing metadata
- RDF has a graph-based data model
- RDF has an XML-based syntax to support syntactic interoperability
  - XML and RDF complement each other because RDF supports semantic interoperability
- RDF has a decentralized philosophy and allows incremental building of knowledge, and its sharing and reuse

Summary

- RDF is domain-independent
  - RDF Schema provides a mechanism for describing specific domains
- RDF Schema is a primitive ontology language
  - It offers certain modelling primitives with fixed meaning
  - Key concepts of RDF Schema are class, subclass relations, property, subproperty relations, and domain and range restrictions
  - There exist query languages for RDF and RDFS, including SPARQL