Building and using ontologies

Elena Simperl, University of Southampton, UK
e.simperl@soton.ac.uk @esimperl

With contributions from "Linked Data: Survey of Adoption", Tutorial at the 3rd Asian Semantic Web School ASWS 2011, Incheon, South Korea, July 2011 by Aidan Hogan, DERI, IE
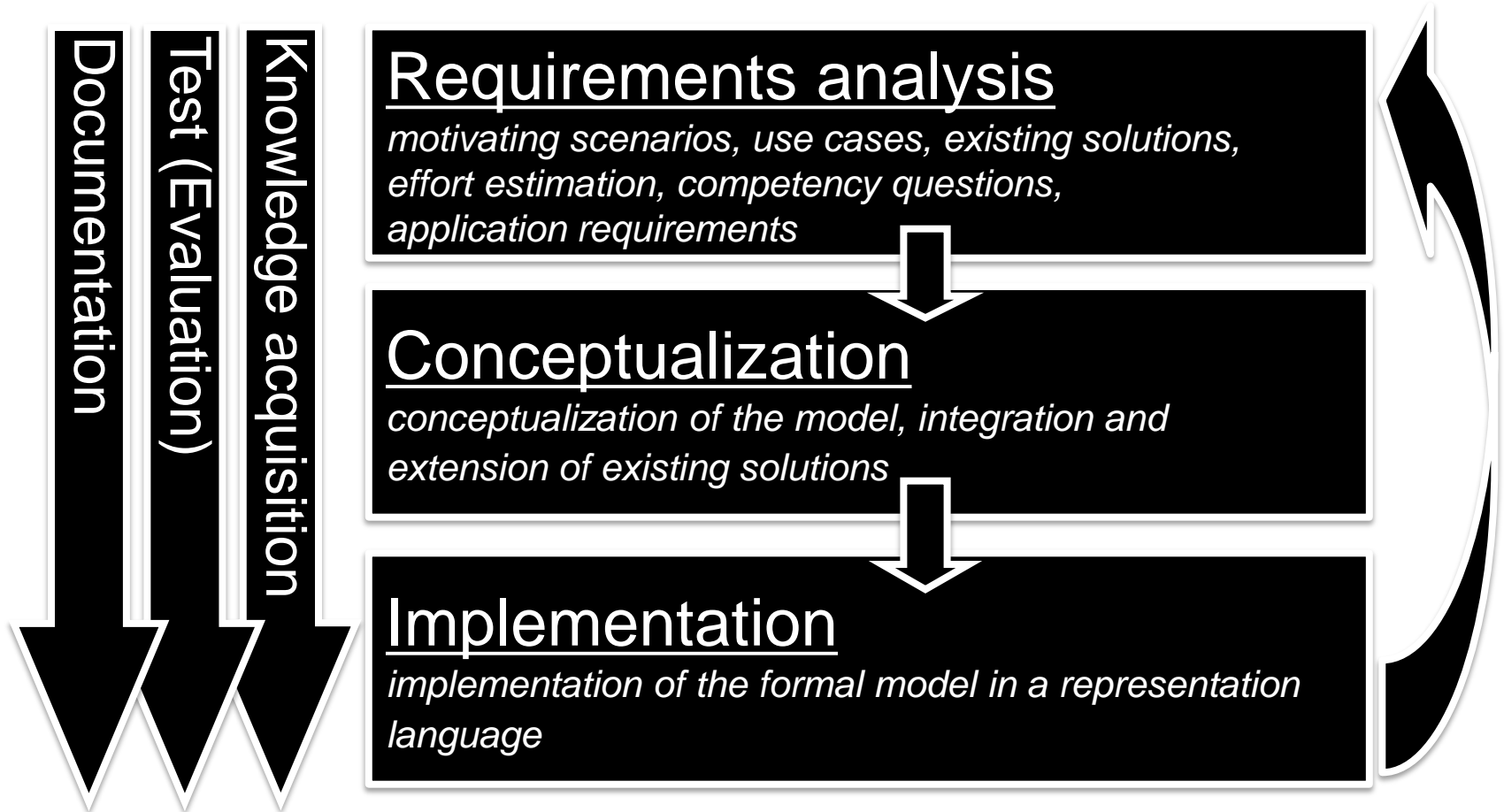
FUNDAMENTALS

Ontologies in Computer Science

- An ontology defines a **domain of interest**
  - … in terms of the **things** you talk about in the domain, their **attributes,** as well as **relationships** between them
- Ontologies are used to
  - Share a **common understanding** about a domain among people and machines
  - Enable **reuse** of domain knowledge

ontology *vocabulary*

microformat *conceptual graph*

topic map *thesaurus*

schema

**classification** object model

semantic network

glossary **taxonomy**

class relationship *entity* **type**

slot *particular* object *instance*

universal **attribute** association

individual *property* **role**

Classical ontology engineering process

EUCLID

EdUcational Curriculum for the usage of LInked Data

Requirements analysis
motivating scenarios, use cases, existing solutions, effort estimation, competency questions, application requirements

Conceptualization
conceptualization of the model, integration and extension of existing solutions

Implementation
implementation of the formal model in a representation language

Documentation

Test (Evaluation)

Knowledge acquisition

Example: Project Halo

- Knowledge acquisition from text (books)

- Professional and crowdsourced annotation

- Question analysis and answering through a combination of NLP and reasoning techniques
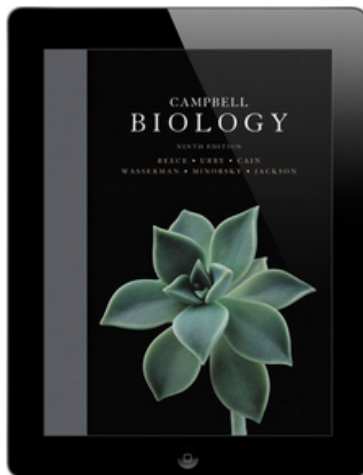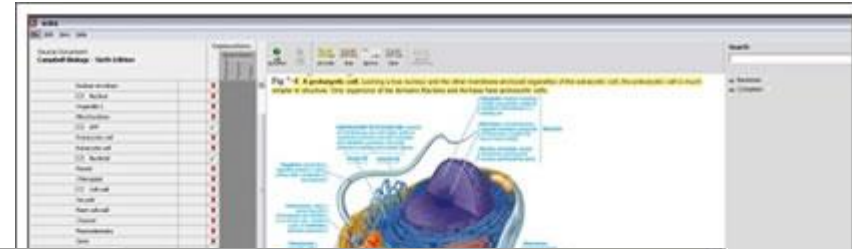
What is the relationship between plant cells and photosynthesis

define	Define **cellular respiration**
structure	What is the structure of a **chloroplast**?
function	What is the function of a **plasma membrane** in a **eukaryotic cell**?
compare	What are the differences between **chloroplasts** and **mitochondria**?
relate	If the **chloroplasts** were removed from a **plant**, what events would be affected?
search	Search book for **photosynthesis**

Images from http://www.projecthalo.com and http://www.inquireproject.com/

More examples

Semantic technologies are not **THE** solution to creating intelligent applications, but only one (essential) component

The Linked Data movement has promoted one approach to create and publish semantic data

- They created momentum for the Semantic Web, as well as several useful data sets

Rich knowledge representations can be extremely valuable, but are costly to achieve

| System	Scope	Input	Result	Core Technology
Inquire intelligent textbook	single textbook	simple English queries	formatted data and relevant textbook content	symbolic AI
Wolfram Alpha computational knowledge engine	curated data from "primary sources"	word phrases with mathematical operators	formatted data	Mathematica
Siri virtual personal assistant	emails, calendar, weather, maps, movies, etc.	voice commands	performs tasks	service integration via speech dialog
Google Search index of world's information	open domain text on the web	keywords and search queries	web documents	statistical AI, PageRank

Table from http://www.inquireproject.com/

Our scenario: the Linked Data management life cycle
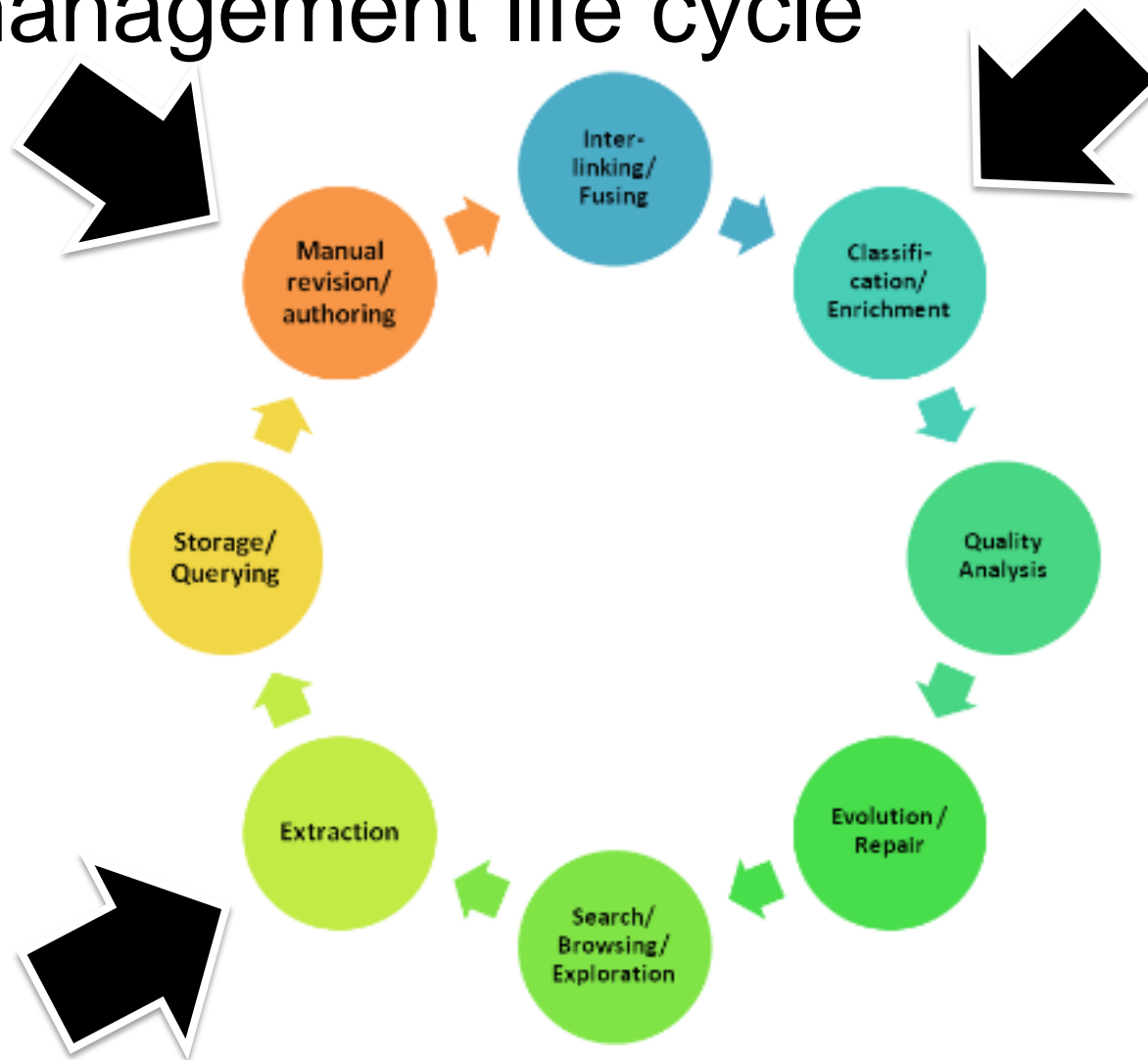
Example: BBC

- Various micro-sites built and maintained manually

- No integration across sites in terms of content and metadata

- Use cases
  - Find and explore content on specific (and related) topics
  - Maintain and re-organize sites
  - Leverage external resources

- Ontology: One page per thing, reusing DBpedia and MusicBrainz IDs, different labels

http://www.slideshare.net/reduxd/beyond-the-polar-bear
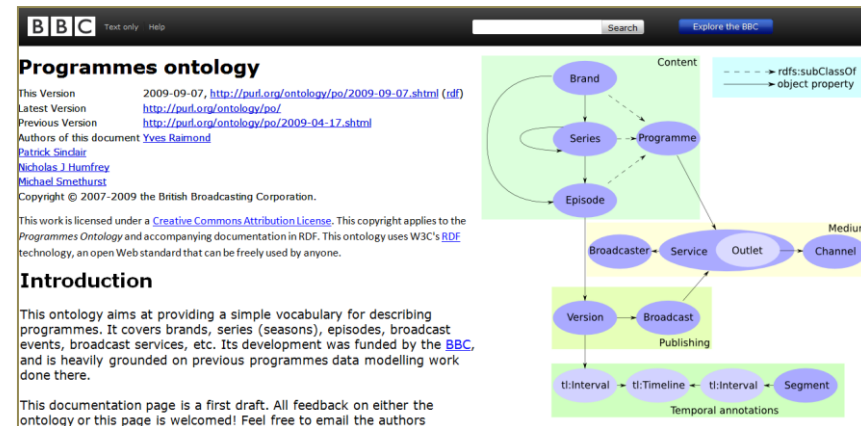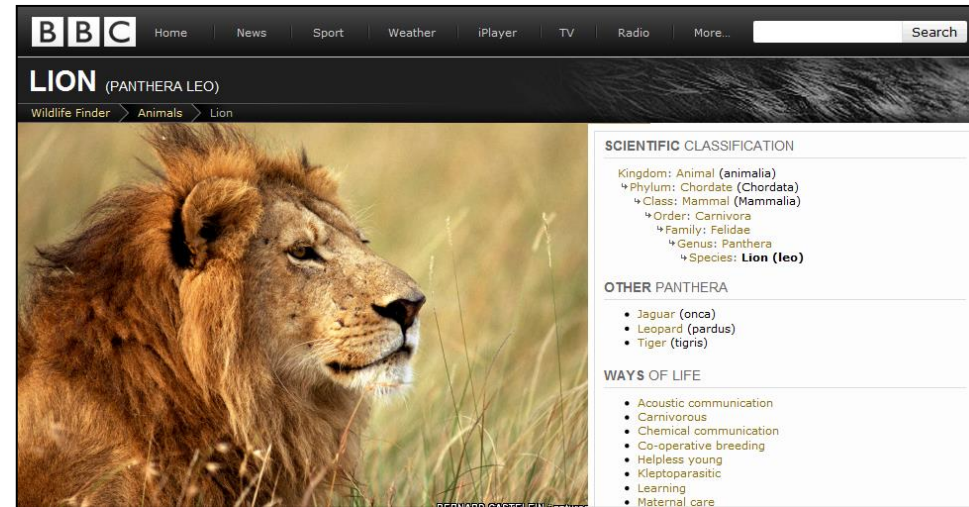
Core ontology engineering activities in our scenario

- **Find ontologies**
- **Select ontologies**
- **Adjust/extend ontologies**

- Popular activities we do **not** consider here
  - Requirements analysis
  - Knowledge representation
  - Ontology learning
  - Ontology alignment
  - ...
- See previous summer schools http://videolectures.net/eswc2012_summer_school/

- This is **not** a tutorial about
  - Ontology engineering tools e.g., **Protégé** (see http://protege.stanford.edu/)
  - Ontology languages e.g., RDFS, OWL

FIND ONTOLOGIES

Finding existing ontologies

- Linked Open Vocabularies: over 400 vocabularies, used in the LOD cloud
  - http://lov.okfn.org
- Protégé Ontologies: several hundreds of ontologies, cross-domain
  - http://protegewiki.stanford.edu/index.php/Protege_Ontology_Library#OWL_ontologies
- Open Ontology Repository: life sciences and other domains
  - http://ontolog.cim3.net/cgi-bin/wiki.pl?OpenOntologyRepository
- Dumontier Lab: life sciences ontologies
  - http://dumontierlab.com/index.php?page=ontologies
- Tones: ontologies used mainly for testing purposes
  - http://rpc295.cs.man.ac.uk:8080/repository/
- OBO Foundation Ontologies: hundreds of life sciences ontologies, including mappings
  - http://www.obofoundry.org/
- NCBO Bioportal: hundreds of medical ontologies
  - http://bioportal.bioontology.org/
- VoCamps
  - http://vocamp.org/wiki/Main_Page

Linked Open Vocabularies

See http://lov.okfn.org

Linked Open Vocabularies (2)

EUCLID
EdUcational Curriculum for the usage of LInked Data

Information about mo

Metrics:

Overall score with current parameters	0.405
Best ratio search words in labels	0.357
Nb occurences in main labels	2
Nb occurences in secondary labels	2
Nb occurences in LOD datasets	144239
Nb occurences in LOV datasets	18
Nb of vocabularies which reference this element	5

Element information:

URI	http://purl.org/ontology/mo/
rdf:type	owl:Thing > owl:Ontology > voaf:Vocabulary
vocabulary	null
is dc:hasPart of	All > Life > Music
dcterms:title	Music Ontology @en
dce:title	The Music Ontology
dcterms:description	The Music Ontology Specification provides main concepts and properties fo describing music (i.e. artists, albums and tracks) on the Semantic Web @en
dce:description	The Music Ontology Specification provides main concepts and properties fo describing music (i.e. artists, albums and tracks) on the Semantic Web.

Close

Dublin Core

Properties in the /terms/ namespace	abstract, accessRights, accrualMethod, accrualPeriodicity, accrualPolicy, alternative, audience, available, bibliographicCitation, conformsTo, contributor, coverage, created, creator, date, dateAccepted, dateCopyrighted, dateSubmitted, description, educationLevel, extent, format, hasFormat, hasPart, hasVersion, identifier, instructionalMethod, isFormatOf, isPartOf, isReferencedBy, isReplacedBy, isRequiredBy, issued, isVersionOf, language, license, mediator, medium, modified, provenance, publisher, references, relation, replaces, requires, rights, rightsHolder, source, spatial, subject, tableOfContents, temporal, title, type, valid
Properties in the legacy /elements/1.1/ namespace	contributor, coverage, creator, date, description, format, identifier, language, publisher, relation, rights, source, subject, title, type
Vocabulary Encoding Schemes	DCMIType, DDC, IMT, LCC, LCSH, MESH, NLM, TGN, UDC
Syntax Encoding Schemes	Box, ISO3166, ISO639-2, ISO639-3, Period, Point, RFC1766, RFC3066, RFC4646, RFC5646, URI, W3CDTF
Classes	Agent, AgentClass, BibliographicResource, FileFormat, Frequency, Jurisdiction, LicenseDocument, LinguisticSystem, Location, LocationPeriodOrJurisdiction, MediaType, MediaTypeOrExtent, MethodOfAccrual, MethodOfInstruction, PeriodOfTime, PhysicalMedium, PhysicalResource, Policy, ProvenanceStatement, RightsStatement, SizeOrDuration, Standard
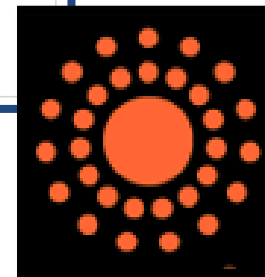
Table from http://dublincore.org/documents/dcmi-terms/

Friend Of A Friend

PROV-O

Image from http://www.w3.org/TR/2012/WD-prov-o-20120724/
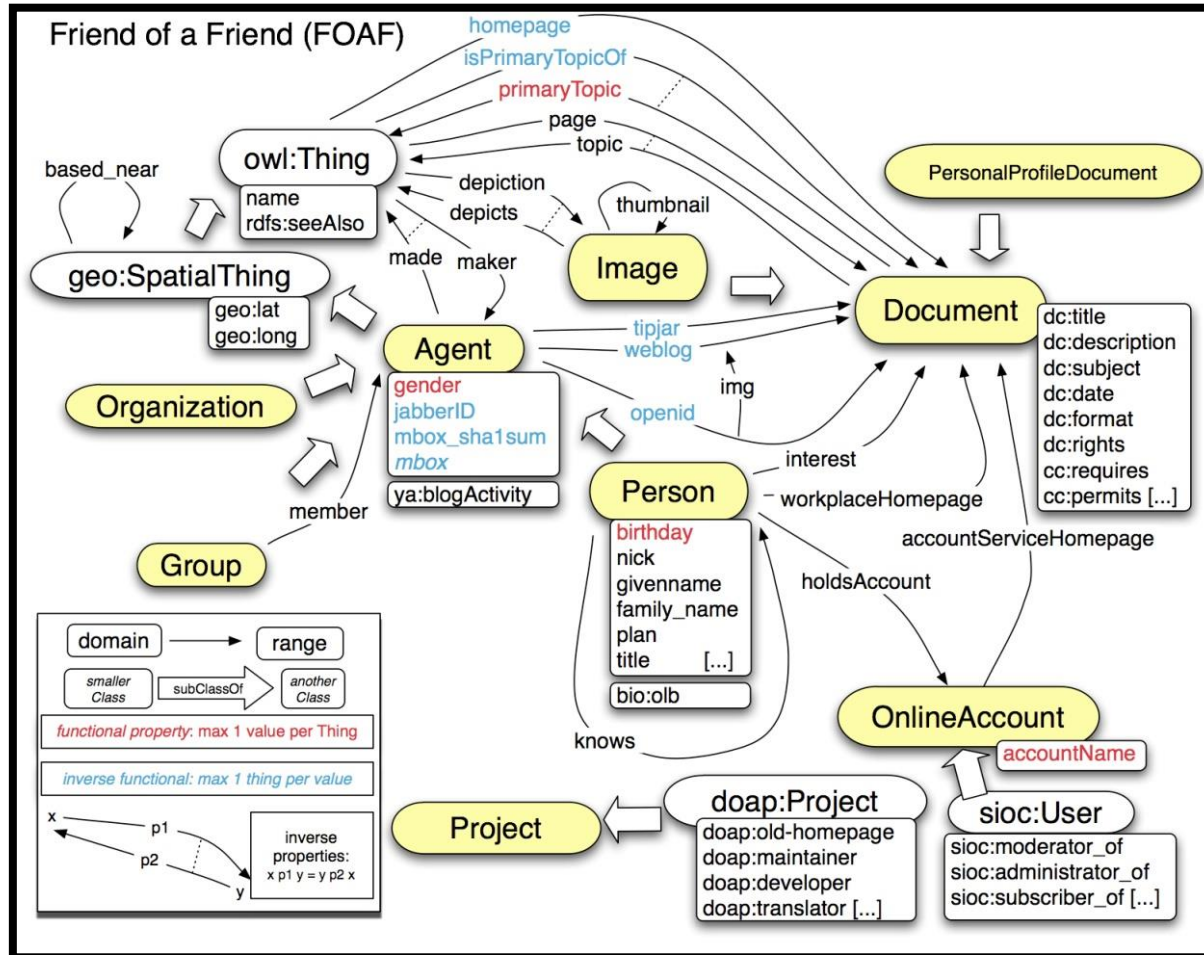
DBpedia

- **Classes and properties for Wikipedia export (infoboxes), regularly updated**

```
{{Infobox_Town_AT |
  name = Innsbruck |
  image_coa =  InnsbruckWappen.png |
  image_map = Karte-tirol-I.png |
  state = [[Tyrol]] |
  regbzk = [[Statutory city]] |
  population = 117,342 |
  population_as_of = 2006 |
  pop_dens = 1,119 |
  area = 104.91 |
  elevation = 574 |
  lat_deg = 47 |
  lat_min = 16 |
  lat_hem = N |
  lon_deg = 11 |
  lon_min = 23 |
  lon_hem = E |
  postal_code = 6010-6080 |
  area_code = 0512 |
  licence = I |
  mayor = Hilde Zach |
  website = [http://innsbruck.at] |
}}
```

Innsbruck

Country	Austria
State	Tyrol
Administrative region	Statutory city
Population	117,342 (2006)
Area	104.91 km²
Population density	1,119 /km²
Elevation	574 m
Coordinates	47°16' N 11°23' E
Postal code	6010-6080
Area code	0512
Licence plate code	I
Mayor	Hilde Zach
Website	www.innsbruck.at

About: Innsbruck

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

Innsbruck is the capital city of the federal state of Tyrol in western Austria. It is located in the Inn Valley at the junction with the Wipptal, which provides access to the Brenner Pass, some 30 kilometers (19 mi) south of Innsbruck.

Property	Value
dbpedia-owl:PopulatedPlace/populationDensity	1119.0
dbpedia-owl:abstract	Innsbruck ist die Landeshauptstadt des Bundeslandes Tiro Transit-Strecke Brenner (Auto- und Eisenbahn) nach Südti (Brücke über den Inn). Innsbruck ist mit 118.082 (Stand 1. und Salzburg die fünftgrößte Stadt Österreichs, im Ballung zu kommen ca. 30.000 Studenten und andere Nebenwo chtigungen von Städtetouristen. Innsbruck is the capital city of the federal state of Tyrol in the junction with the Wipptal, which provides access to the

See http://wiki.dbpedia.org/

Wikidata

See http://www.wikidata.org/

Freebase

http://www.freebase.com

Semantically Interlinked Online Communities
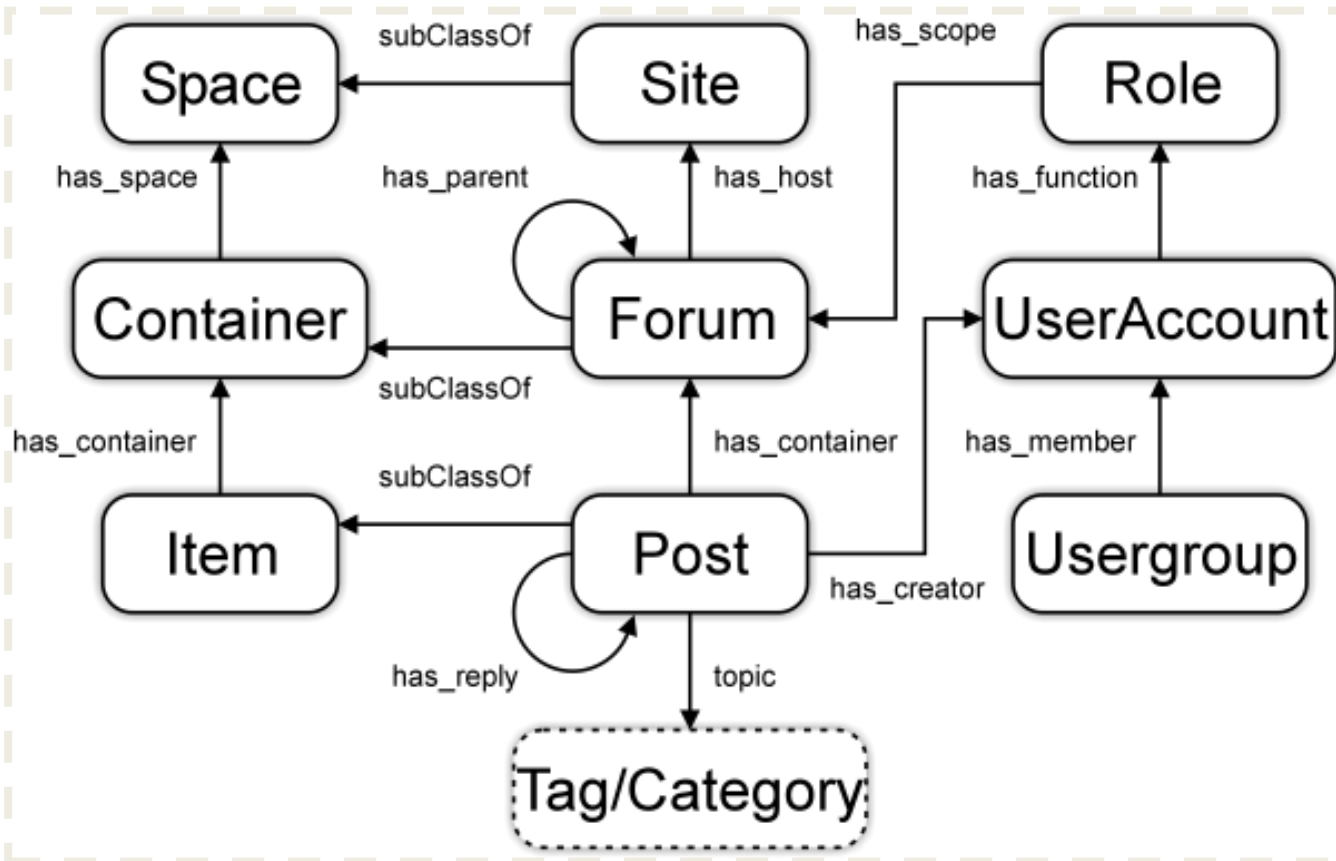
Image from http://rdfs.org/sioc/spec/: Bojārs, Breslin et al.

Simple Knowledge Organization System
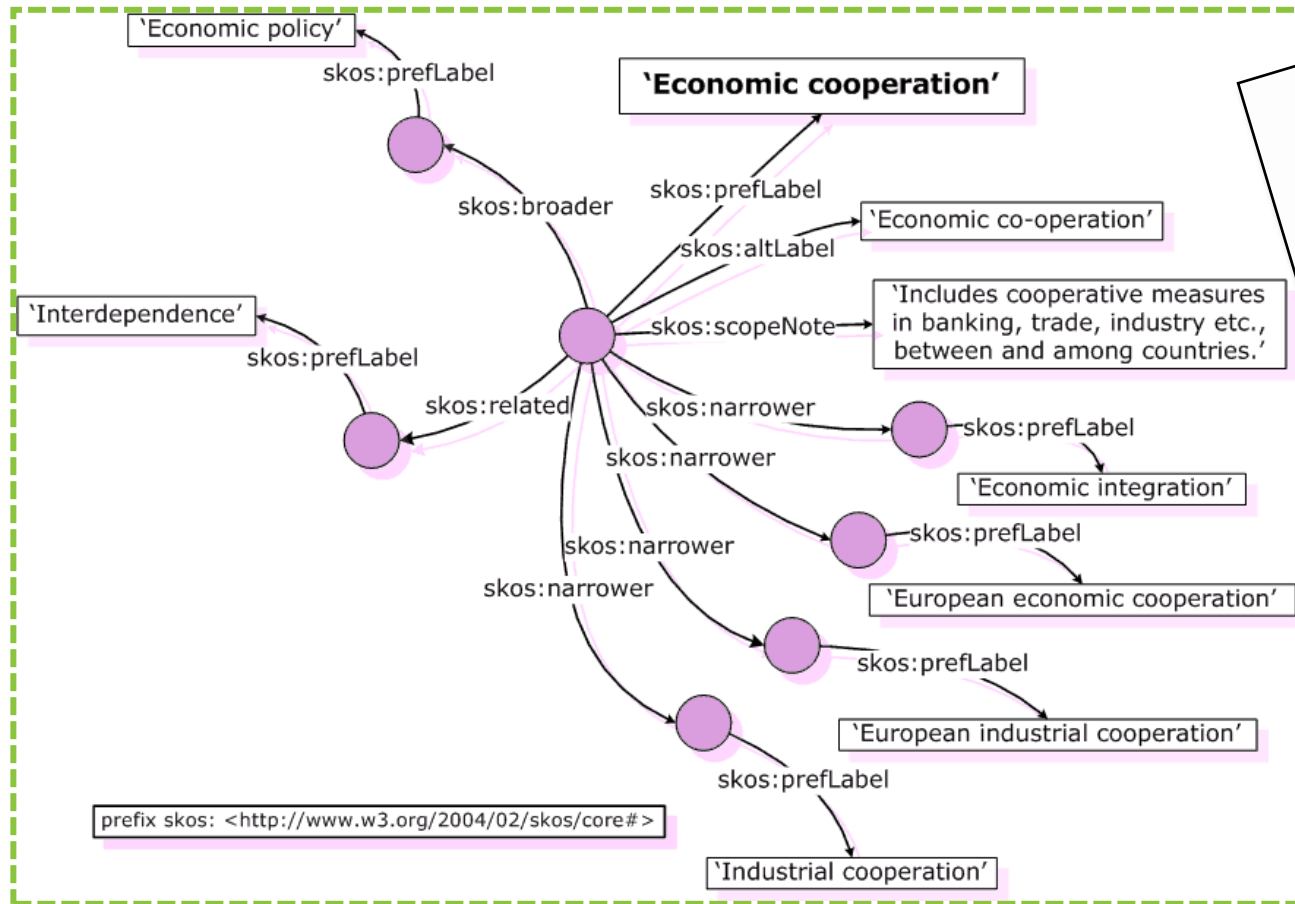
Image from http://www.w3.org/TR/swbp-skos-core-guide: Miles, Brickley

Description Of A Project

Music Ontology

WordNet

schema.org

- Collection of schemas to mark-up structured content in HTML pages

Thing: additionalType, alternateName, description, image, name, sameAs, url

Action: agent, endTime, instrument, location, object, participant, result, startTime

AchieveAction

LoseAction: winner

TieAction

WinAction: loser

AssessAction

ChooseAction: option

VoteAction: candidate

IgnoreAction

ReactAction

AgreeAction
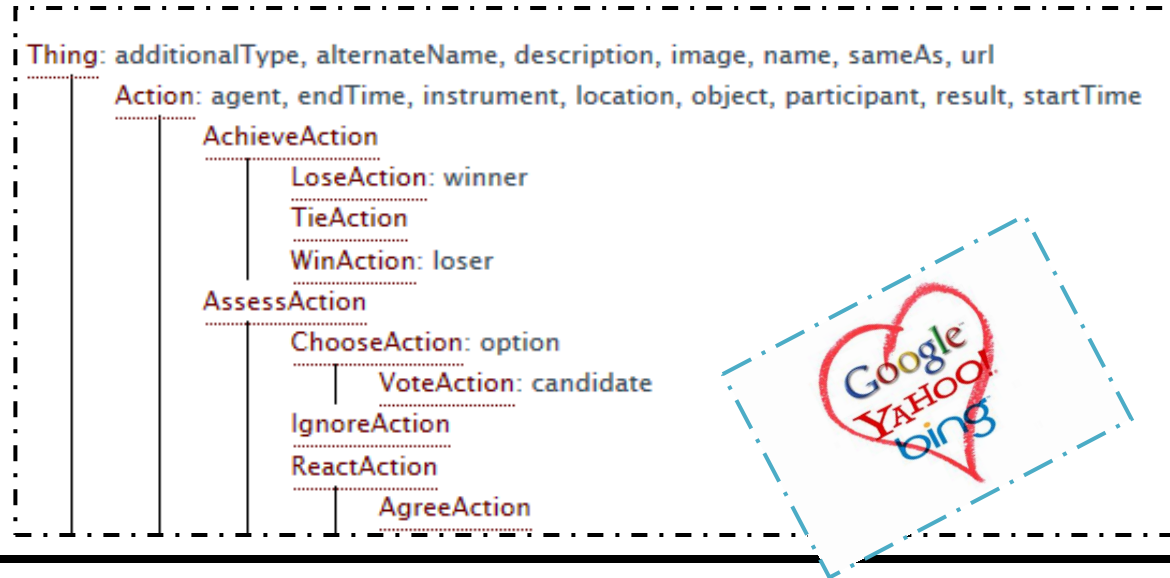
Note: as of 2014-04-04 this tree is not entirely up to date. Additional types have been added: see EmailMessage, Reservation, Question and Answer, added in version 1.1. Version 1.2 added the Potential Actions vocabulary, see potentialAction, EntryPoint, target, actionStatus, ActionStatusType, ActiveActionStatus, CompletedActionStatus, PotentialActionStatus.

semantic web
structured data
markup
schema.org
SEO
search

See also http://schema.org/docs/full.html
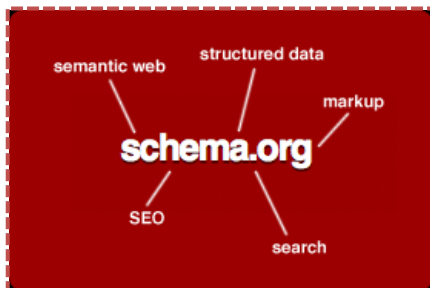
GoodRelations
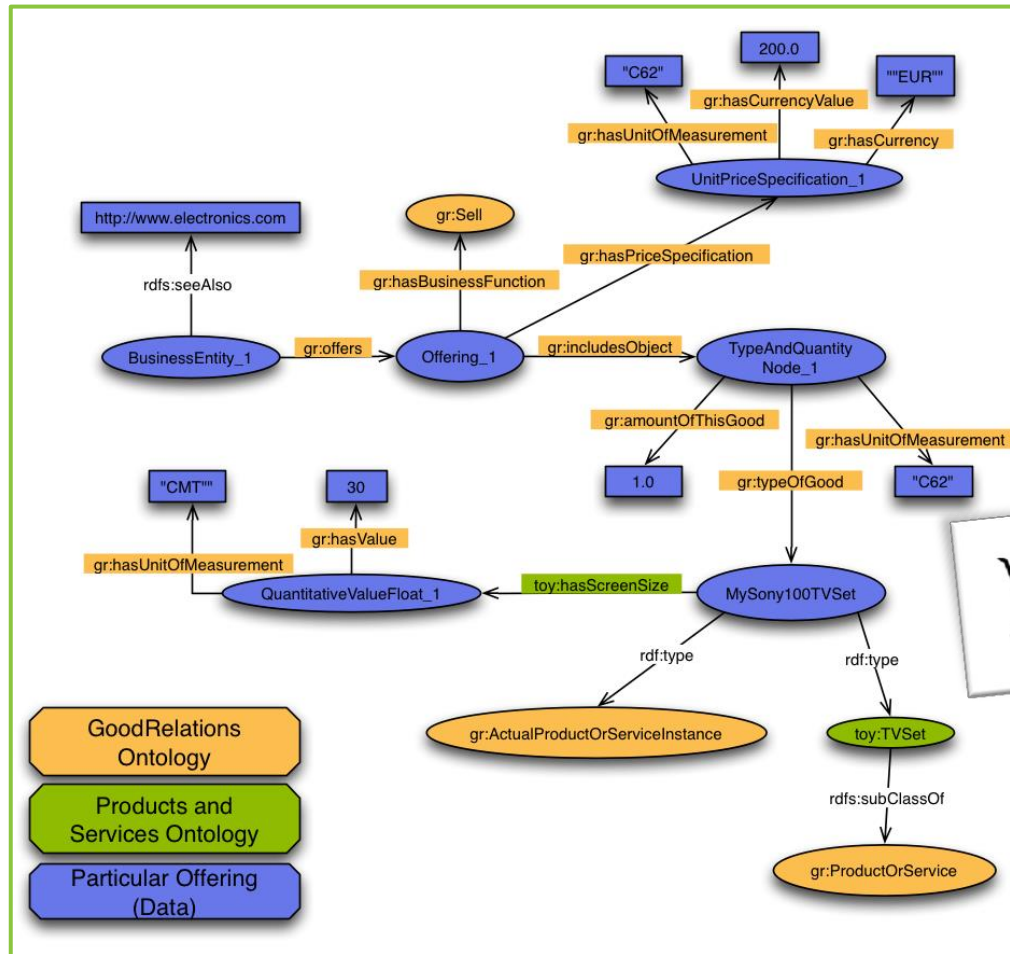
Image from http://www.heppnetz.de/projects/goodrelations/primer/: **Hepp**

Life sciences and healthcare

http://www.obofoundry.org/

Getty vocabularies

http://www.getty.edu/research/tools/vocabularies/lod/index.html

SELECT ONTOLOGIES

Selecting relevant ontologies

- Key: domain and usage
  - There are many different points of view upon a domain
  - Use popular ontologies

- You might need to adjust/expand an existing ontology to
  - Lexicalization
  - Implementation language (e.g., RDFS, OWL, frames, SKOS)
  - Level of granularity
  - Level of expressivity
  - Instance data

- Be aware of/that
  - Imports: transitive dependency between ontologies
  - Changes in imported ontologies can result in inconsistencies and changes of meanings and interpretations, as well as computational aspects

Brief introduction to ontology conceptualization

ADJUST/EXPAND

Basics

- Ontological primitives in this tutorial
- Classes
- Instances
- Attributes
- Relationships
- Literals
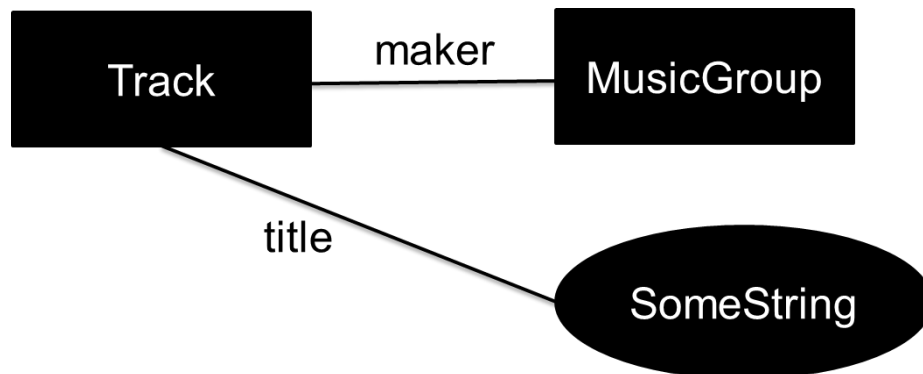
- In real applications
  - Ontology languages with different degrees of formality and support for
    - Different types of nodes
    - Different types of edges
    - Built-in features of nodes and edges
  - Nodes and edges may come from different ontologies
  - (Ideally) provenance metadata attached to nodes and edges

Example: OWL

- Classes
- Instances
  - Set of classes is not always disjunct from set of instances
- Datatype properties
- Object properties
- Constraints
  - Cardinality
  - Range constraints (all values, some values etc.)
- Others
  - Imports
  - Annotations
  - …

Classes

- A class represents a set of instances
- A class should be cohesive, meaninfully named, and relevant
- Classes represent domain concepts and not the words that denote these concepts
  - Synonyms for the same concept do not represent different classes

| MusicArtist | performs | MusicalWork |
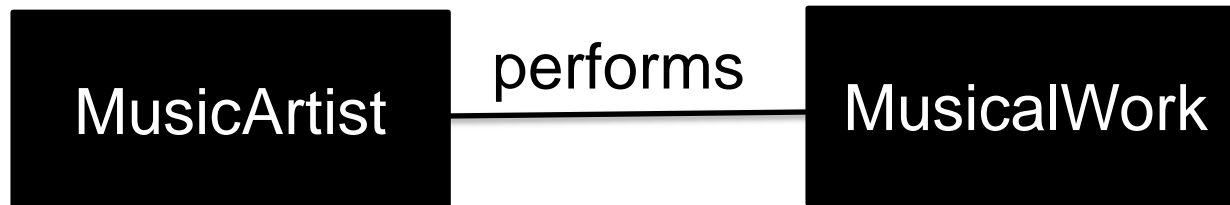
Classes (2)

- Typically nouns and nominal phrases, but not restricted to them
  - Verbs can be modeled as classes, if the emphasis is on the process as a whole rather than the actual execution
  - No pronouns

Person — isIll — 0/1

Person — isAffectedBy — IllnessEpisode
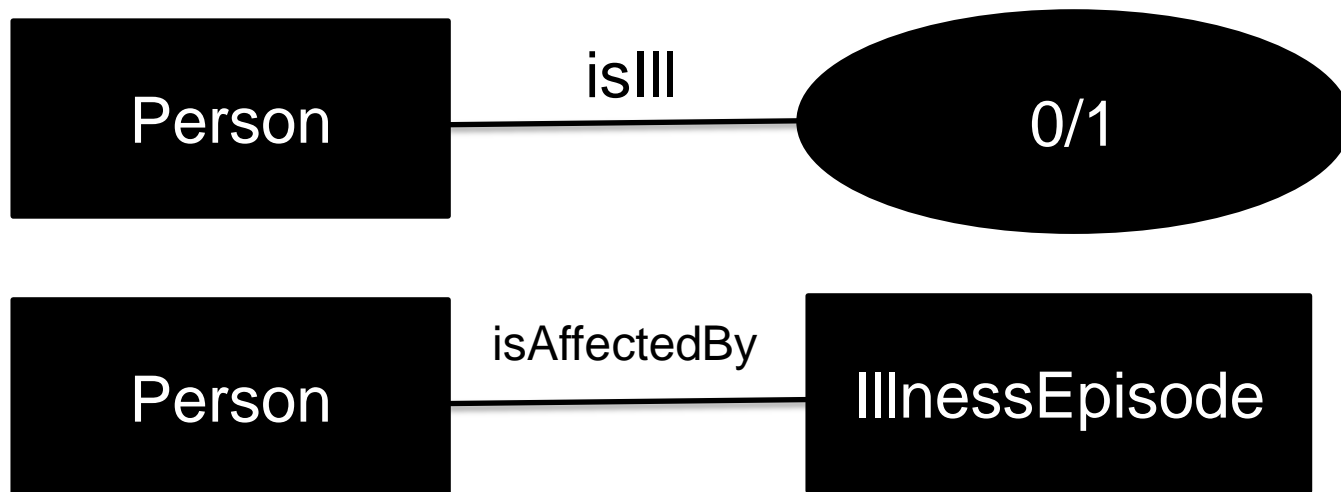
Cohesiveness

- A class should represent one thing, all of that thing and nothing but that thing
  - Why: Reusability, maintenance, see also OO design
- You can prove cohesion by giving the class a representative name, typically nouns
  - No plural form, e.g, Albums
  - No others, utilities etc.
- On a related note: avoid ambiguous terms
  - Manager, handler, processor, list, information, item, data etc.

Instances

- Entities of a certain type
  - Abstract entities (e.g., Jazz music) are allowed

- Issues
  - Distinction between classes and instances
    - Example: Stradivarius
  - Choice of the most appropriate class
    - Example: Violetta Valery

- Identity vs individuality: entities may change values, but remain members of the same class
  - Example: Age of child vs person

Characterizing classes

- Two types of principal characteristics
  - ‚Measurable' properties of a class: attributes
  - Inter-entity connections: relationships associations

Image	— hasColor —	Color

Image	— hasColor —	{red, blue, green,…}
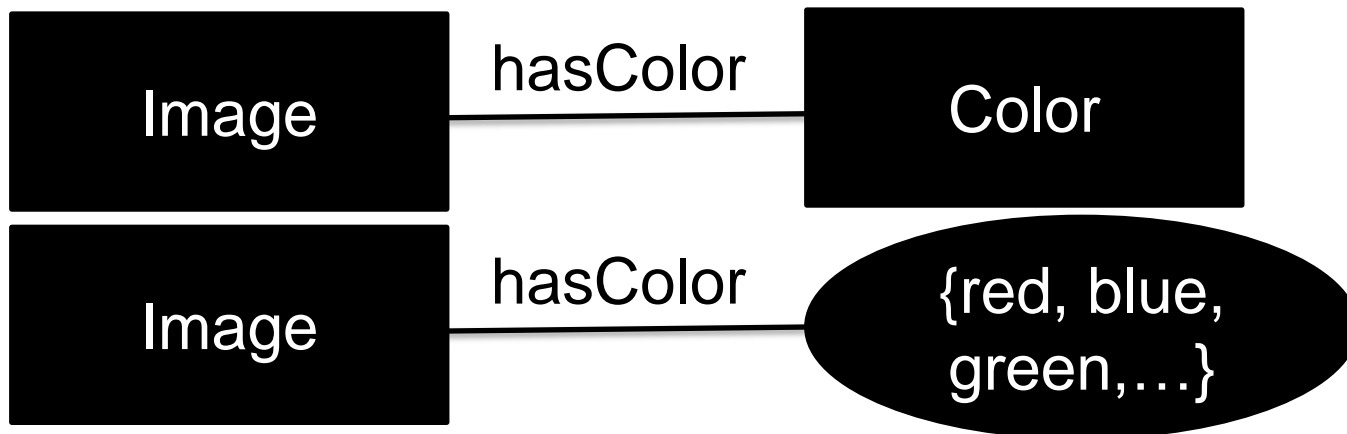
Attributes

- An attribute is a measurable property of a class
  - Scalar values: choice from a range of possibilities
  - Attributes do NOT exhibit identity

Relationships

MusicArtist — MusicalWork

MusicalWork — similar_to — 1 ... *

Performance — 0..1 — 0..* — MusicalWork

MusicArtist — * — composes — * — Composition
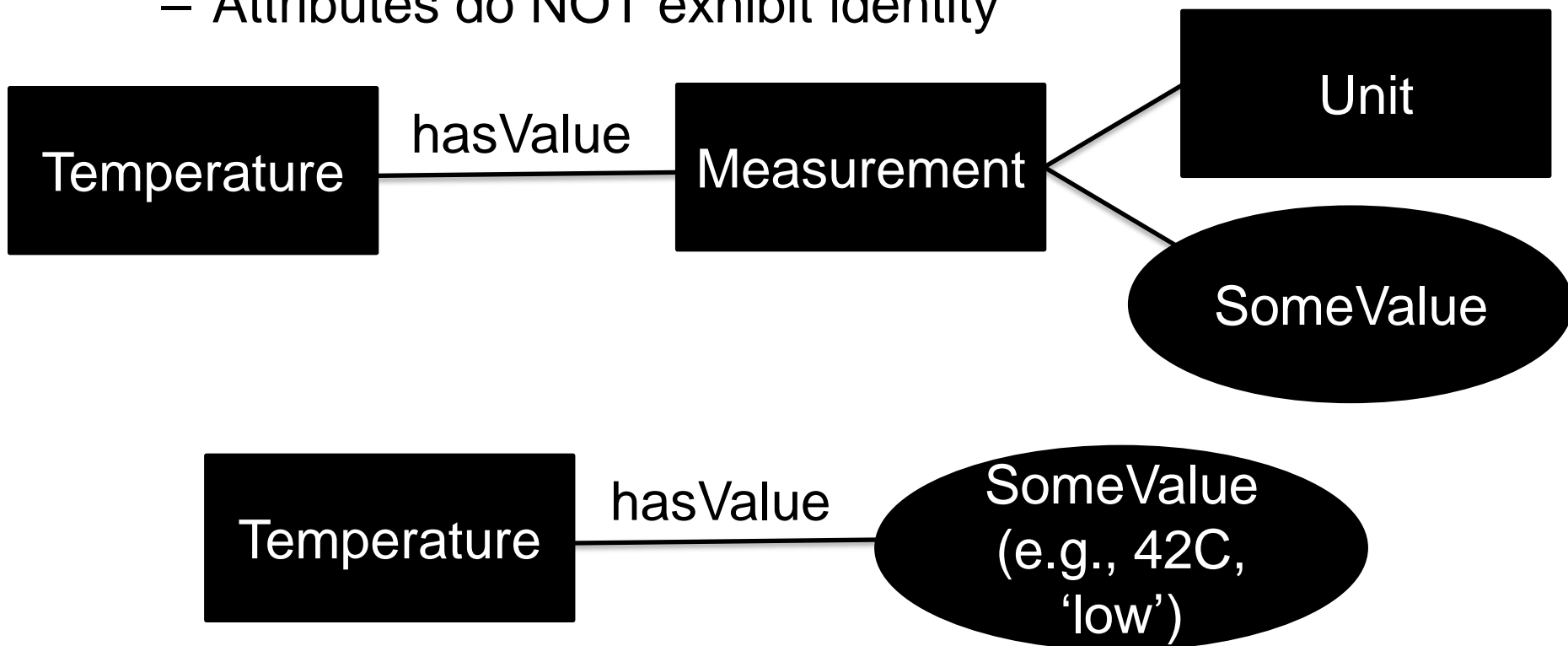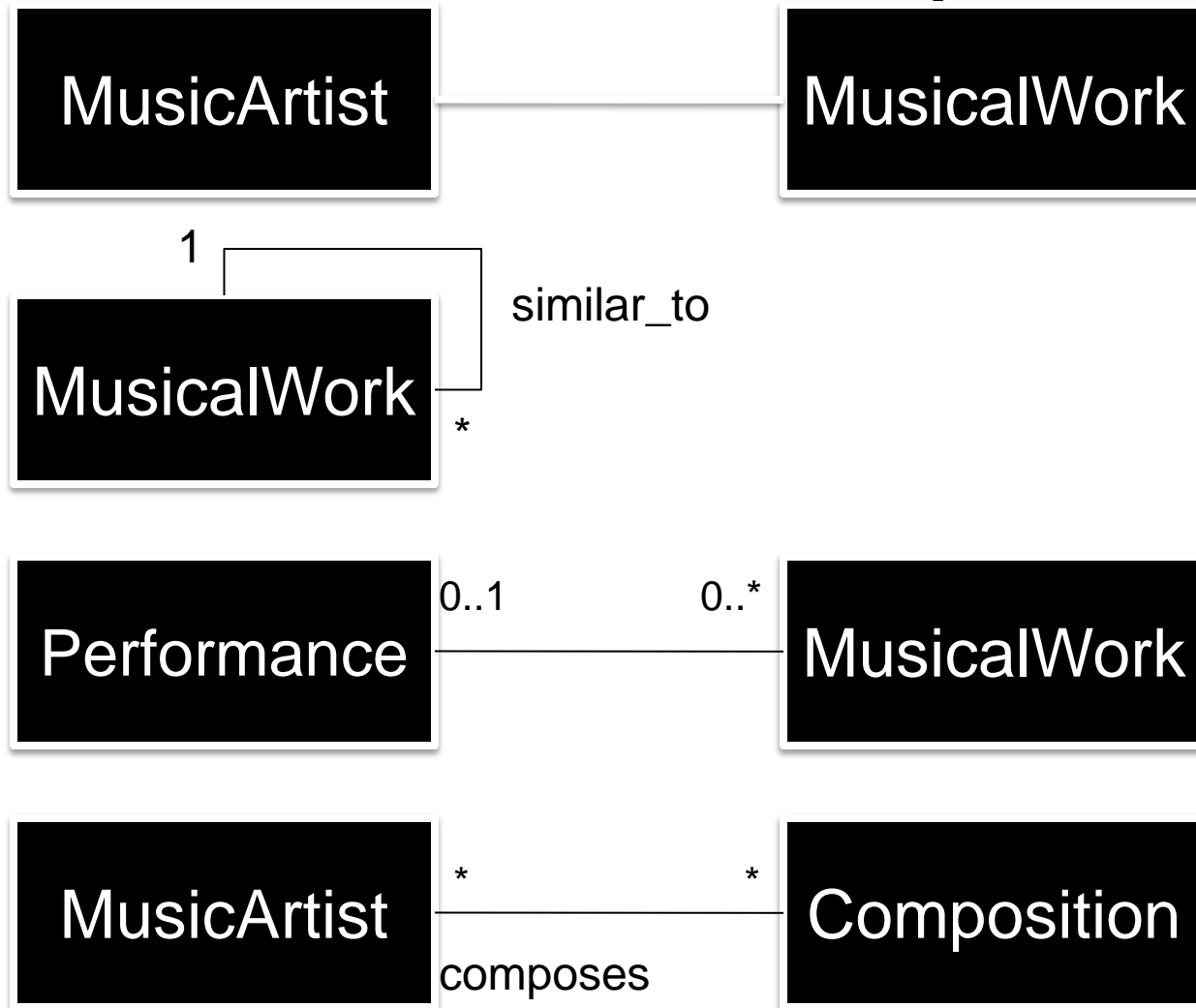
Some instances of a class hold a relationship with some instances of another class

Class hierarchy

- A subclass of a class represents a concept that is a "kind of" the concept that the superclass represents

- A subclass has
  - Additional properties
  - Restrictions different from those of the superclass, or
  - Participates in different relationships than the superclasses

- Multiple inheritance may be possible

Class hierarchy (2)

- All the siblings in the hierarchy (except for the ones at the root) must be at the same level of generality

- If a class has only one direct subclass there may be a modeling problem or the ontology is not complete

- If there are more than a dozen subclasses for a given class then additional intermediate categories might be necessary

- Roles are not subclasses

- Application dependent or subjective
  - Example: Artist and person
  - Example: Rectangle and square

Formal properties of ontologies

- Identity
  - Example: triangle as three edges of the same length vs edge length and angle
  - Example: the same clay vs the same statue
  - See also primary keys in ER modeling
- Types and roles
  - Roles hold because an instance happens to participate in some relationship with another instance (at some point in time), and not because they care essential to identify these instances
  - Example: Person vs student vs employee
- Dependence
  - Existence depends on other instance
  - Example: Student and university
- Concreteness
  - Has physical location (not necessarily real)
  - Example: Violetta Valery
- Unity
  - Is identified by the sum of its parts
  - Example: Piece of stone vs person vs pile of stones

- These properties are inherited along by subclasses and instances
- Used to
  - Test ontological consistency
  - Avoid unintended inferences
  - Improve extendibility
  - Improve reusability

- See also
  - OntoClean (http://en.wikipedia.org/wiki/OntoClean)

Ontology design paterns

Content from http://ontologydesignpatterns.org/
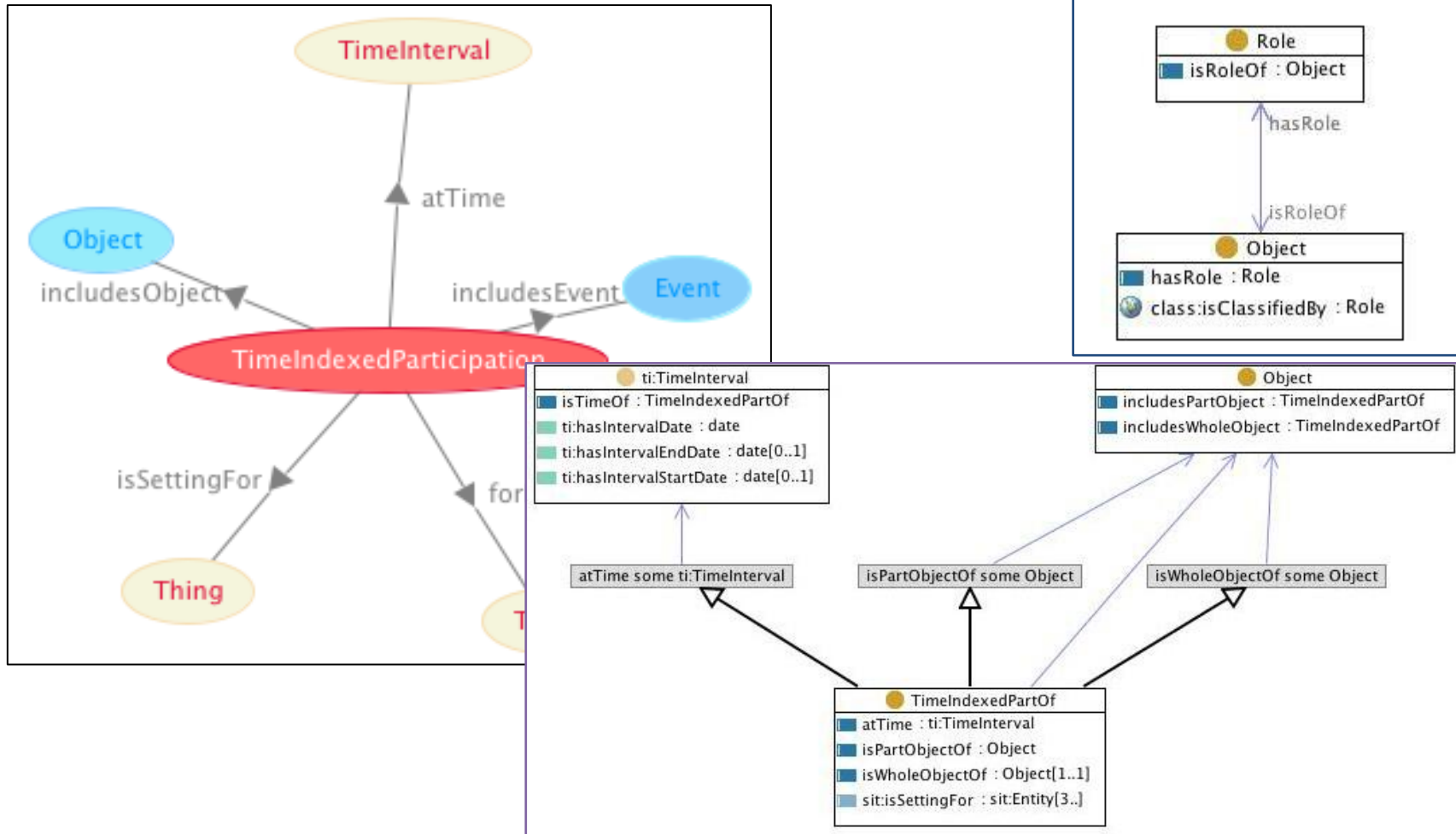
Assignments

(in your free time)

Modeling: Unstructured to structured

The current configuration of the "Red Hot Chili Peppers" are: Anthony Kiedis (vocals), Flea (bass, trumpet, keyboards, and vocals), John Frusciante (guitar), and Chad Smith (drums). The line-up has changed a few times during they years, Frusciante replaced Hillel Slovak in 1988, and when Jack Irons left the band he was briefly replaced by D.H. Peligo until the band found Chad Smith. In addition to playing guitars for Red hot Chili Peppers Frusciante also contributed to the band "The Mars Volta" as a vocalist for some time.

From September 2004, the Red Hot Chili Peppers started recording the album "Stadium Arcadium". The album contains 28 tracks and was released on May 5 2006. It includes a track of the song "Hump de Bump", which was composed in January 26, 2004. The critic Crian Hiatt defined the album as "the most ambitious work in his twenty-three-year career". On August 11 (2006) the band gave a live performance in Portland, Oregon (US), featuring songs from Stadium Arcadium and other albums.

Modeling: different encodings

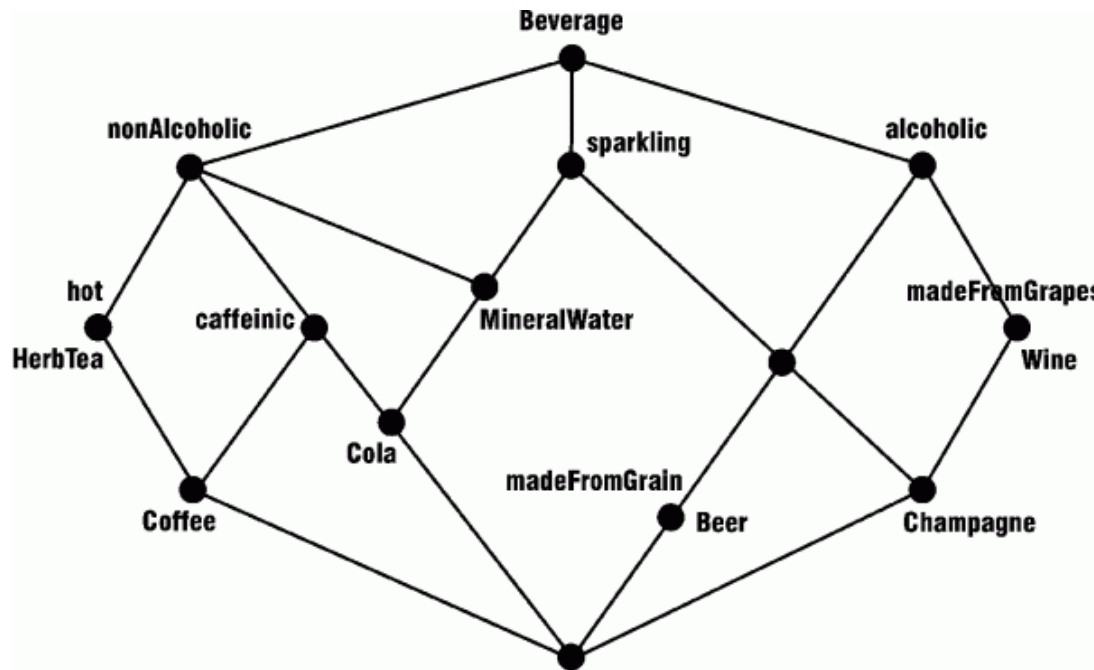- Encode using the notation introduced in the tutorial

Image from **http://www.jfsowa.com/ontology/**

http://www.euclid-project.eu

eBook

HTML

amazonkindle

ePUB

iBooks

Course

HTML

iTunes U

Other channels

@euclid_project

euclidproject

euclidproject