



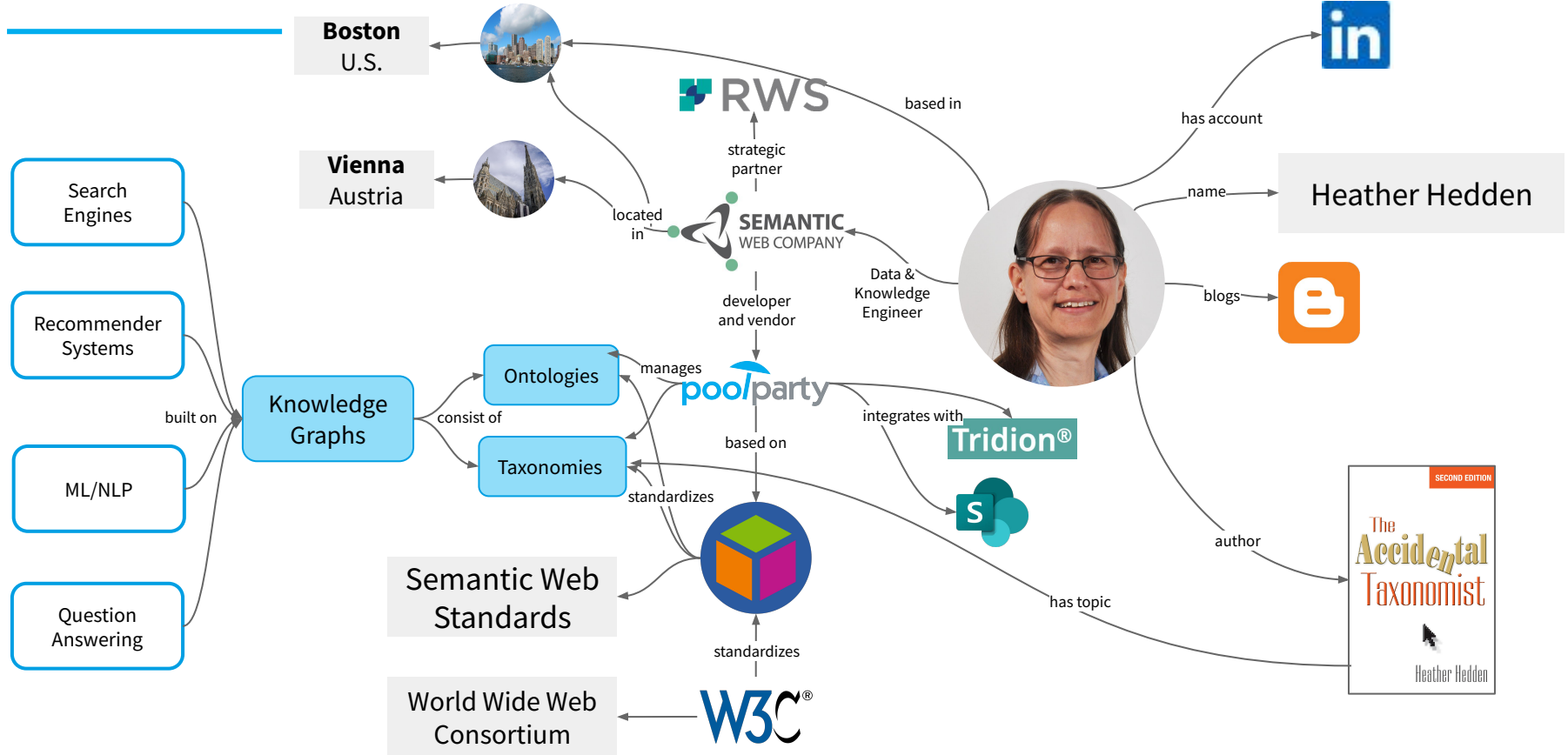
Turning a Taxonomy into an Ontology

KM World Connect: Taxonomy Boot Camp
November 17, 2021



Heather Hedden
Data & Knowledge Engineer
Semantic Web Company

Click the Graph—get in contact with us!



- ▶ Experience as vocabulary editor at Gale (1995 - 2004, 2014 - 2019) involved managing vocabularies with customized “cross-object” relationships and “extended attributes” on terms.



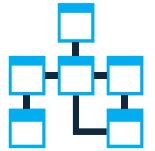
- ▶ Use of different thesaurus management tools with capabilities to customize relationships and attributes.
- ▶ At Taxonomy Boot Camp November 2012, presented [“From Accidental Taxonomist to Accidental Ontologist,”](#) a 10-minute, 10-slide “pecha kucha” talk.



- ▶ Why extend a taxonomy
- ▶ Knowledge organization system types
- ▶ What is an ontology
- ▶ Standards for taxonomies and ontologies
- ▶ Extending a taxonomy: different approaches to creating an ontology
- ▶ Applying an ontology as a semantic layer to a taxonomy/thesaurus
- ▶ Example implementation of ontology-based facets

What you can do with a taxonomy

- ▶ **Tagging:** index content consistently so that retrieval is comprehensive and accurate
- ▶ **Normalization:** Bring together different names, localizations, languages for concepts
- ▶ **Standard search:** find content about.... (search string matches taxonomy concepts)
- ▶ **Topic browse:** explore subjects arranged in a hierarchy and then content on the subject
- ▶ **Faceted (filtering/refining) search:** find content meeting a combination of basic criteria
- ▶ **Discovery:** find other content tagged with same concepts as tagged to found content; explore broader, narrower, and (sometimes) related taxonomy topics
- ▶ **Content curation:** create feeds or alerts based on pre-set search terms
- ▶ **Metadata management:** for retrieval, identification, comparison, analysis, etc.



What you cannot do with a taxonomy alone, but can with an added ontology

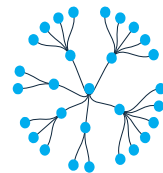
- ▶ Modeling complex interrelationships (e.g. in product approval or supply chain processes) and also connect to content
- ▶ Complex multi-part searches: e.g. find contacts in a specific location, who are employed by companies which belong to certain industries
- ▶ Exploring explicit relationships between concepts (not just broader, narrower, related)
- ▶ Search across datasets, not just search for content
- ▶ Search on more specific criteria that vary based on category (class)
- ▶ Visualization of concepts and semantic relationships
- ▶ Reasoning and inferencing
- ▶ Build a knowledge graph...



Knowledge graphs: taxonomy + ontology + instance data in a graph database

What you can do with a knowledge graph

- ▶ Search expanded to data in different repositories, platforms, applications, etc.
- ▶ Search and discovery across heterogeneous content and data, both structured and unstructured
- ▶ Better personalization and recommendation systems
- ▶ Visualizations of curated data



Enterprise knowledge graphs are becoming the dominant implementation for business ontologies.


Knowledge Organization System Types

Knowledge Organization system (KOS)

- ▶ Any system of terms, terminology, classification, etc.
- ▶ To organize, define, manage, and/or retrieve information.
- ▶ Broader, includes more than just “controlled vocabularies.”

KOS types:

term lists
synonym rings
name authorities
taxonomies
thesauri
glossaries
dictionaries
gazetteers
terminologies
categorization schemes
classification systems
subject heading schemes
semantic networks
ontologies



Controlled
Vocabularies
for tagging and
information
retrieval

Knowledge Organization System Types

Controlled vocabularies

- ▶ Term lists/Pick lists
- ▶ Synonym rings
- ▶ Authority files
 - ▶ Name authorities
- ▶ Taxonomies
- ▶ Subject heading schemes
- ▶ Thesauri

Defined vocabularies

- ▶ Dictionaries
- ▶ Glossaries
- ▶ Gazetteers
- ▶ Terminologies

Classification systems

- ▶ Cataloging systems
- ▶ Categorization schemes
- ▶ Classification schemes

Semantic models

- ▶ Mind maps
- ▶ Topic maps
- ▶ Semantic networks
- ▶ Ontologies

Knowledge Organization System Types



Term List	Name Authority	Taxonomy	Thesaurus	Ontology
Ambiguity control	Ambiguity control Synonym control (Attributes)	Ambiguity control (Synonym control) Hierarchical relationships	Ambiguity control Synonym control Hierarchical relationship Associative relationships	Ambiguity control Semantic relationships Classes Attributes

Knowledge Organization System Types

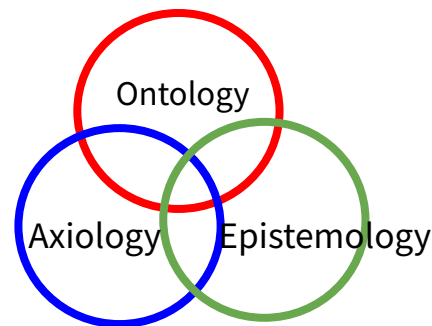
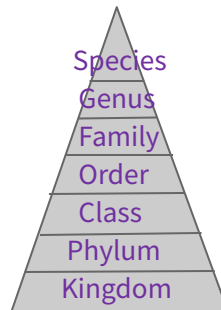


Ontology (model/layer)			
Term List	Name Authority	Taxonomy	Thesaurus
Ambiguity control	Ambiguity control Synonym control (Attributes)	Ambiguity control (Synonym control) Hierarchical relationships	Ambiguity control Synonym control Hierarchical relationship Associative relationships

What is an Ontology

Traditional meanings

- ▶ **Taxonomy** - an ordered naming system or classification system (especially in biology)
- ▶ **Ontology** - branch of metaphysics concerned with the nature and relations of being



Information and knowledge management usage:

- ▶ **Taxonomy** - a structured set of terms/concepts in a subject domain, used for tagging content to support information findability and retrieval
- ▶ **Ontology** - a structured set of entities, relationships and attributes in a subject domain, with semantic expressiveness, used for information exploration and analysis, in addition to findability and retrieval

Definition by Tom Gruber:

An ontology defines a set of representational primitives with which to model a domain of knowledge or discourse.

The representational primitives are typically classes (or sets), attributes (or properties), and relationships (or relations among class members).

The definitions of the representational primitives include information about their meaning and constraints on their logically consistent application.

In the context of database systems, ontology can be viewed as a level of abstraction of data models, analogous to hierarchical and relational models, but intended for modeling knowledge about individuals, their attributes, and their relationships to other individuals.

<https://tomgruber.org/writing/definition-of-ontology.pdf>

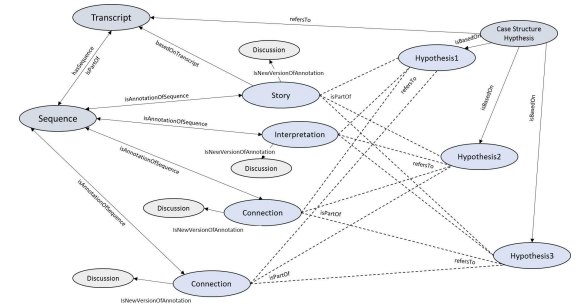
Gruber T. (2016) Ontology. In: Liu L., Özsu M. (eds) *Encyclopedia of Database Systems*. Springer, New York, NY, based on his original definition in 1993.



What is an Ontology

Other definition components of an ontology

- ▶ A form of knowledge representation
 - ▷ Not just knowledge organization
- ▶ A set of precise descriptive statements about a particular domain.
 - ▷ Statements as subject-predicate-object are expressed as triples.
- ▶ A formal naming and definition of the types, properties and interrelationships of entities in a particular domain.
 - ▷ Classes, custom attributes, and semantic relationships
- ▶ A more abstract layer in describing a knowledge organization system
 - ▷ Overlays and connects to a taxonomy or other controlled vocabulary to add semantics

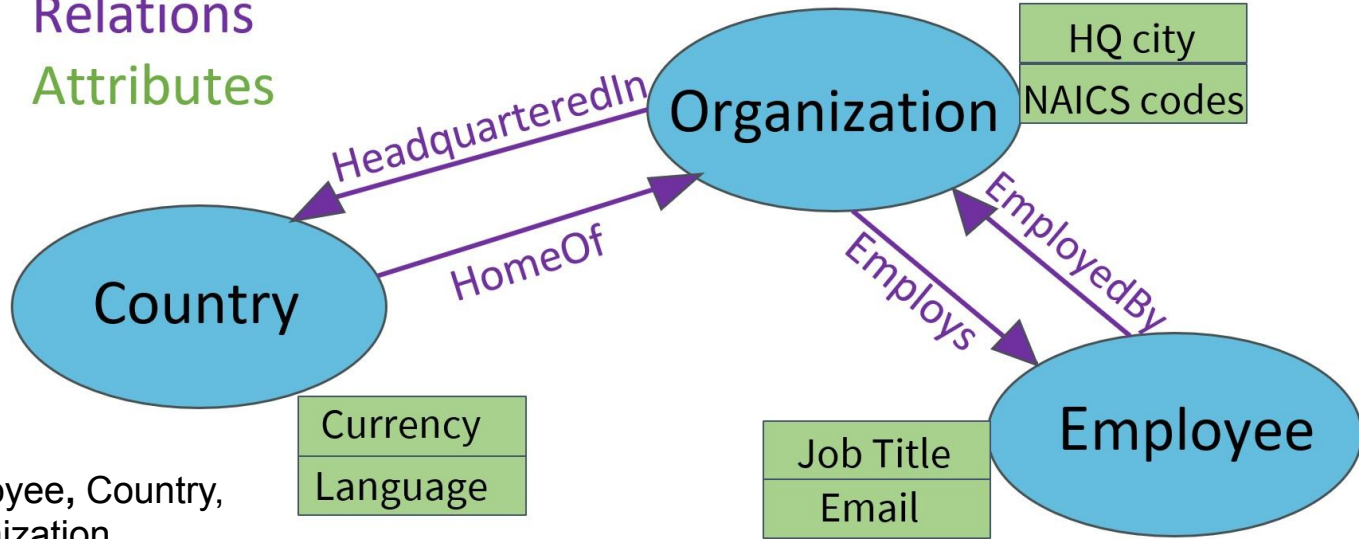


What is an Ontology: Example

Classes

Relations

Attributes



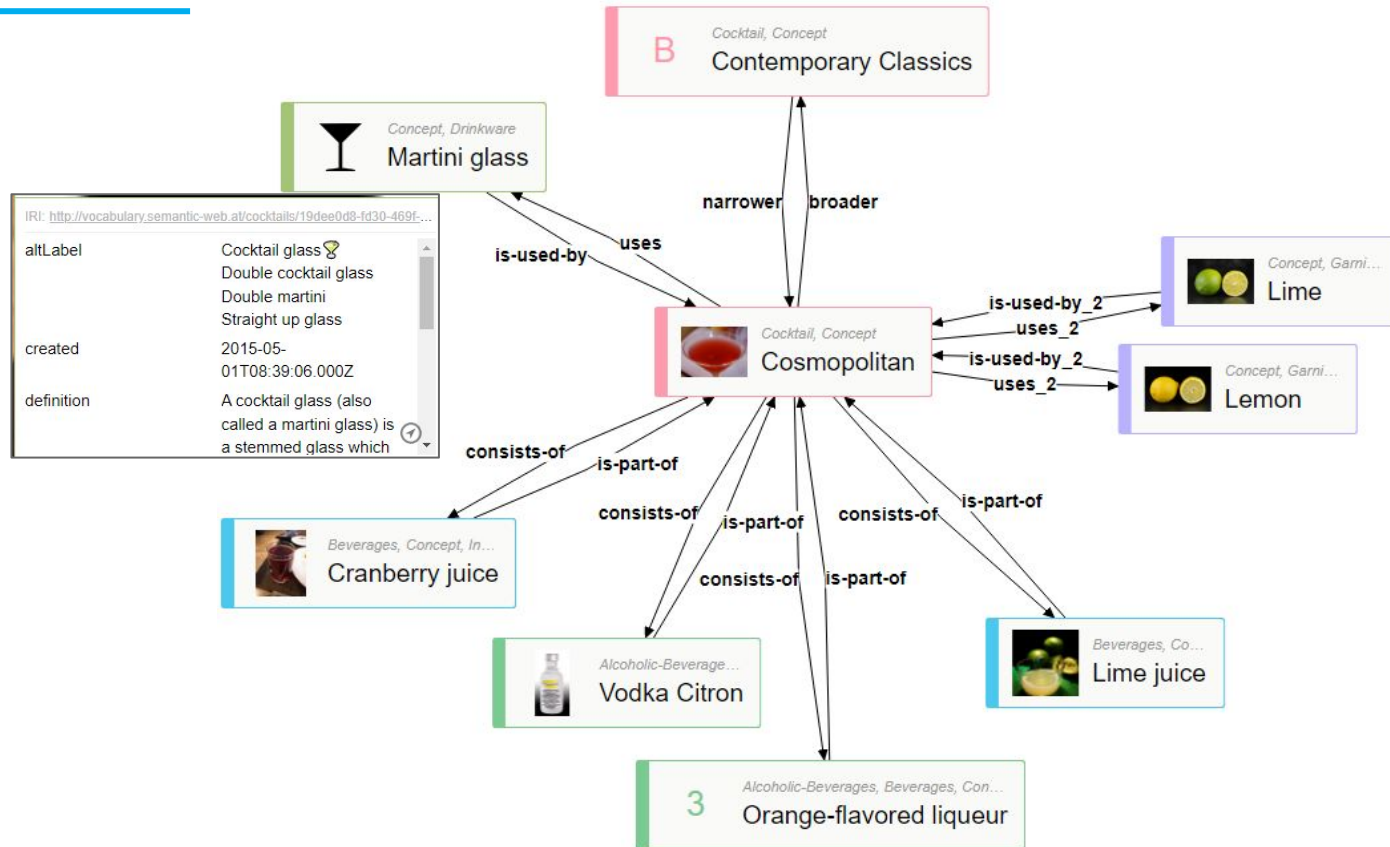
Classes

Employee, Country,
Organization

Relations: HeadquarteredIn < > HomeOf
EmployedBy < > Employs

Attributes: Email address, Job title, HQ city, NAICS codes, Currency, Language

What is an Ontology: Example



What is an Ontology

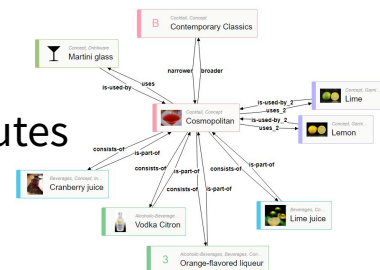
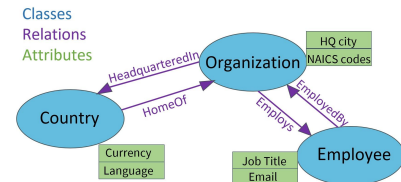
Ontology definition questions

Is it:

- ▶ A knowledge model, a semantic layer, a form of knowledge representation, that describes the classes, relationship types, and attribute types in a domain,

or:

- ▶ A knowledge organization system, that includes both: the classes, relationship types, and attribute types *and* the specific concepts, entities/individuals, and their specific attributes




Both definitions exist.

Turning a taxonomy into an ontology, refers to the latter definition of ontology.


An ontology can also be defined as a semantic model that conforms to OWL standards.

Standards for taxonomies and ontologies

Taxonomy standards

- ▶ ANSI/NISO Z39.19 or ISO 25964
Standards or design best practices, especially for thesauri 
 - ▷ *preferred term, nonpreferred term, broader term, narrower term, and related term relationships; scope notes*
- ▶ W3C: SKOS (Simple Knowledge Organization System)
Standard for machine-readability and interoperability
 - ▷ *concept schemes, concepts; labels, relationships, etc.*

Ontology standards

- ▶ W3C: RDF-S (Resource Description Framework - Schema) 
- ▶ W3C: OWL (Web Ontology Language)
Both are languages for representing ontologies on the web - for machine readability and interoperability

RDF (Resource Description Framework)

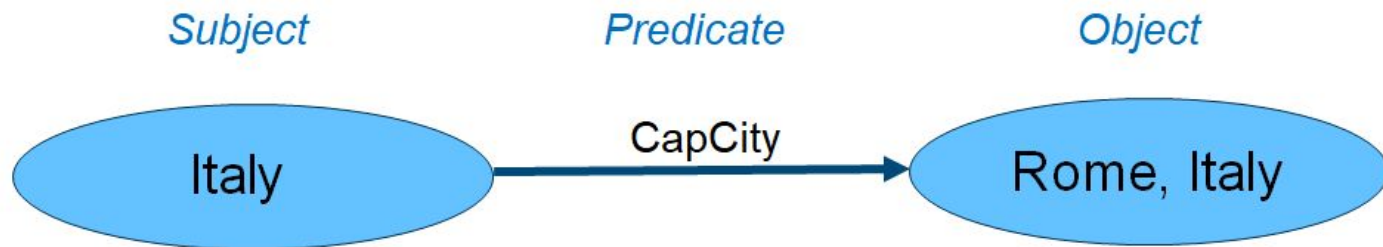
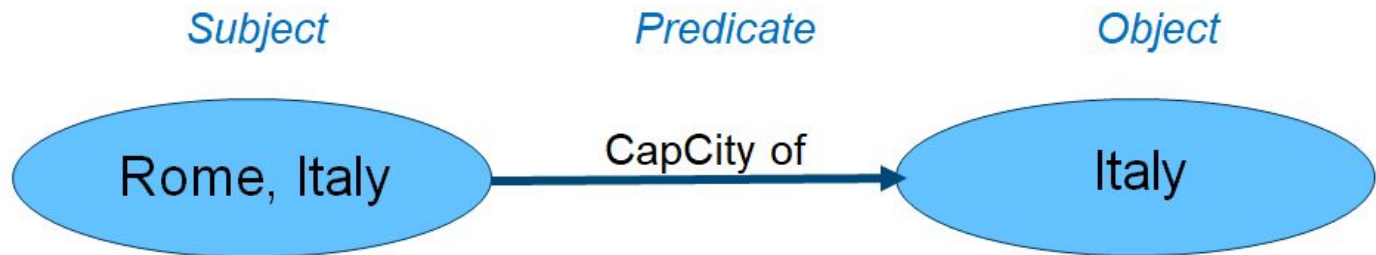
- ▶ A World Wide Web (W3C) recommendation <https://www.w3.org/TR/rdf11-concepts>
- ▶ Started in 1997, adopted by the W3C as a recommendation in 1999, RDF 1.1 specification in 2014.
- ▶ “A standard model for data interchange on the Web.”
- ▶ Requires the use of URIs (uniform resource identifiers) to specify things and to specify relationships.
- ▶ Models information as triples: subject – predicate – object
- ▶ Models information on a graph-based model (as nodes and edges).
- ▶ Facilitates data merging even if the underlying schemas differ.
- ▶ More fundamental, basic, and generic than SKOS or OWL.
RDF serves as a basic standard for *both* taxonomies (in SKOS) and ontologies.



Standards: RDF

RDF triple: (1) Subject – (2) Predicate – (3) Object

Example



SKOS (Simple Knowledge Organization System) elements

Concept Scheme & Collection	Concepts	Labels & Notation	Documentation	Semantic Relations	Mapping Relations
ConceptScheme	Concept	prefLabel	scopeNote	broader	exactMatch
inScheme	hasTopConcept	altLabel	definition	narrower	closeMatch
Collection	topConceptOf	hiddenLabel	example	related	broaderMatch
orderedCollection		notation	changeNote		narrowerMatch
member			editorialNote		relatedMatch
memberList			historyNote		

<https://www.w3.org/TR/skos-reference>



RDF Schema - RDFS or RDF/S or RDF(S)

- ▶ Also called: RDF Vocabulary Description Language 1.0
- ▶ A World Wide Web (W3C) recommendation <https://www.w3.org/2001/sw/wiki/RDFS>
- ▶ Published as part of the RDF Specification Suite Recommendations in 2004.
- ▶ “A general-purpose language for representing simple RDF vocabularies on the Web.”
- ▶ A flexible data model adaptable to specific needs.
- ▶ Goes beyond RDF to designate **classes** and **properties**.
- ▶ RDFS serves as a standard for ontologies.



OWL - Web Ontology Language

- ▶ A World Wide Web (W3C) specification <https://www.w3.org/OWL>
- ▶ First published in 2004; OWL 2 (with extended features), published in 2009 <https://www.w3.org/TR/owl2-overview>
- ▶ “A Semantic Web language designed to represent rich and complex knowledge about things, groups of things, and relations between things”
- ▶ To provide a common way to process the content of web Information.
- ▶ A computer-readable language, usually written in XML,
- ▶ A declarative language (not a programming or schema language)
- ▶ Enables knowledge linking on the web/Semantic Web
- ▶ Based on RDF and RDFS. OWL is W3Cs attempt to extend RDFS.
- ▶ The Semantic Web led to a requirement for a “web ontology language.”



OWL-Defined Ontology Components

Entities – subjects (domains) or objects (ranges) of properties, within RDF triples

- ▶ **Classes** (in SKOS: concepts)
 - ▷ Named sets of concepts that share characteristics and relations
 - ▷ May contain subclasses or individuals (instances of the class)
- ▶ **Individuals** (in SKOS: concepts)
 - ▷ Members or instances of a class. Unique named entities.

Properties – predicates about individuals (instances)

- ▶ **Object properties** (in SKOS: relations)
 - ▷ **Relations** between individuals
 - ▷ May be directed (single direction), symmetric, or with an inverse (different in each direction)
- ▶ **Datatype properties**
 - ▷ **Attributes** or characteristics of individuals
 - ▷ The object of a datatype property is a *value*.



Literals – values of attributes, with just a *lexical form* and a *datatype*.

<https://www.w3.org/TR/2012/REC-owl2-primer-20121211/>

- ▶ Names in OWL are international resource identifiers (IRIs).
- ▶ Syntaxes used in OWL: RDF/XML, OWL XML, and Manchester syntax.
- ▶ OWL modeling features also include:
 - ▷ Class disjointness
 - ▷ Complex classes
 - ▷ Property hierarchies
 - ▷ Property characteristics
 - ▷ Property restrictions
 - ▷ Property cardinality restrictions
 - ▷ Domain and range restrictions
 - ▷ Equality and inequality of individuals
 - ▷ Enumeration of individuals
 - ▷ Datatypes



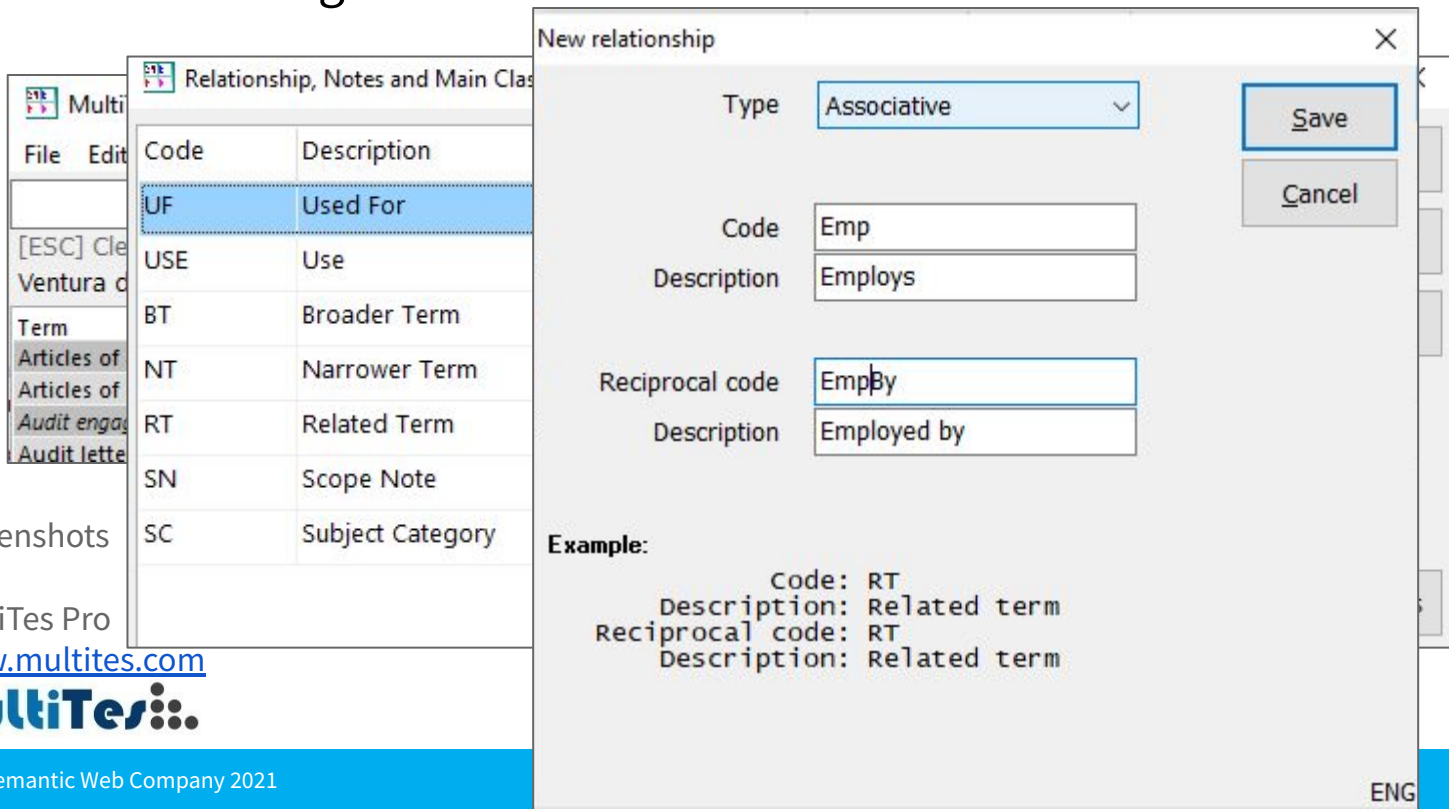
The background of the slide is a light grey color with a network diagram. It consists of numerous white circles of varying sizes connected by thin white lines, creating a complex web of connections. The circles represent nodes, and the lines represent edges or relationships between them.

**Semantically enriching a
taxonomy to become an
ontology**

Different options for creating the semantic model + individual instances definition of an ontology:

1. Use a feature in thesaurus management tool to customize relationships, attributes, and categories to make an ontology-*like* semantic thesaurus
 - ▶ Follows thesaurus standards, but not ontology standards
2. Use a dedicated ontology tool or hand-coding OWL to build out the detailed taxonomic hierarchy (of subclasses and instances) within the ontology
 - ▶ Follows ontology standards, but not thesaurus standards
3. Use taxonomy/ontology combined tool to create taxonomy and an ontology and link them, with the ontology functioning as a semantic layer
 - ▶ Follows thesaurus/taxonomy and ontology standards

1. Customizing a thesaurus



Screenshots from MultiTes Pro www.multites.com

MultiTes

Code	Description
UF	Used For
USE	Use
BT	Broader Term
NT	Narrower Term
RT	Related Term
SN	Scope Note
SC	Subject Category

New relationship

Type: Associative

Code: Emp

Description: Employs

Reciprocal code: EmpBy

Description: Employed by

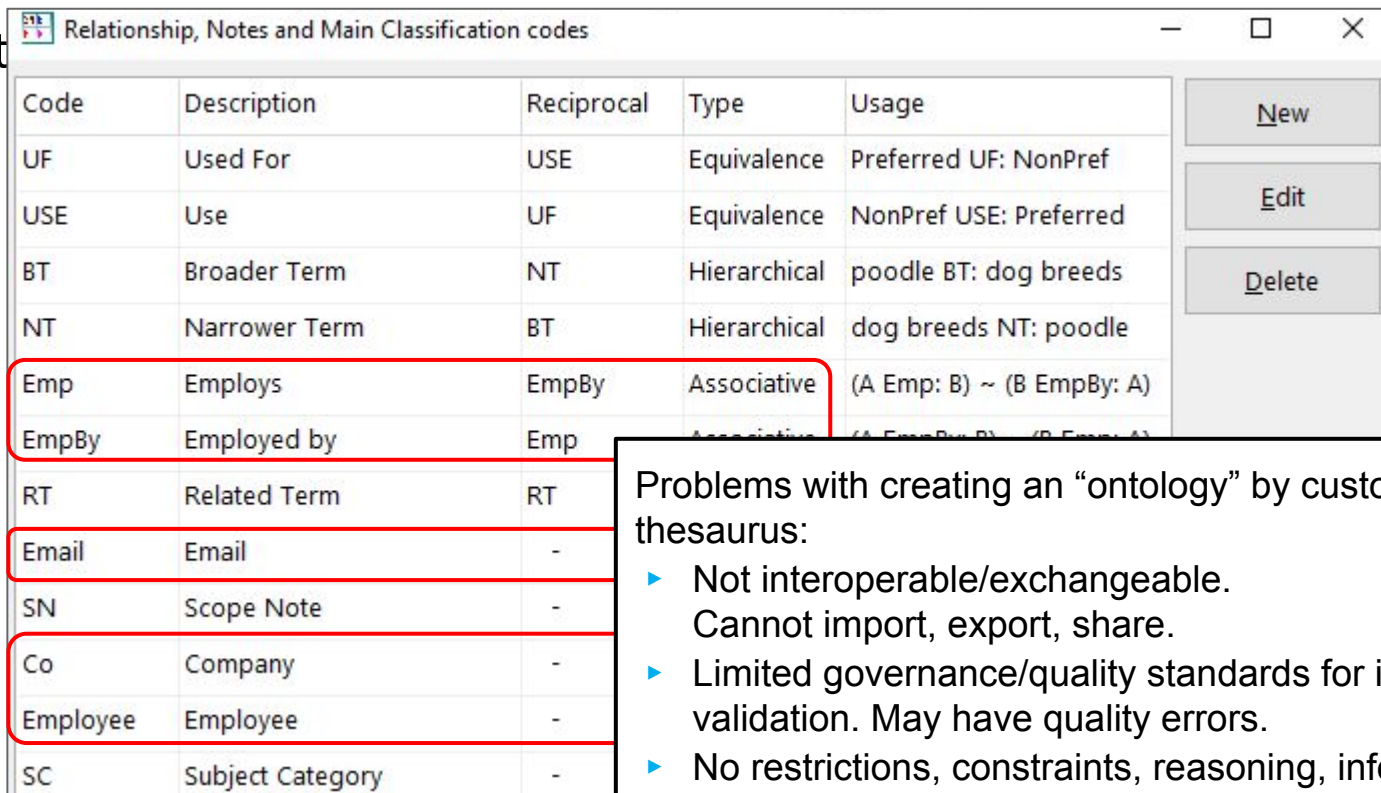
Buttons: Save, Cancel

Example:

```
Code: RT
Description: Related term
Reciprocal code: RT
Description: Related term
```

Ontology Creation Approaches

1. Custom



Code	Description	Reciprocal	Type	Usage
UF	Used For	USE	Equivalence	Preferred UF: NonPref
USE	Use	UF	Equivalence	NonPref USE: Preferred
BT	Broader Term	NT	Hierarchical	poodle BT: dog breeds
NT	Narrower Term	BT	Hierarchical	dog breeds NT: poodle
Emp	Employs	EmpBy	Associative	(A Emp: B) ~ (B EmpBy: A)
EmpBy	Employed by	Emp	Associative	(A EmpBy: B) ~ (B Emp: A)
RT	Related Term	RT		
Email	Email	-		
SN	Scope Note	-		
Co	Company	-		
Employee	Employee	-		
SC	Subject Category	-		

Problems with creating an “ontology” by customizing a thesaurus:

- ▶ Not interoperable/exchangeable. Cannot import, export, share.
- ▶ Limited governance/quality standards for internal validation. May have quality errors.
- ▶ No restrictions, constraints, reasoning, inferencing.

2. Using a dedicated ontology tool

- ▶ The ontology comprises all entities: classes, subclasses, and instances/individuals.
- ▶ The ontology is as detailed as any taxonomy or thesaurus, but has the addition of semantic relations and attributes.
- ▶ Classes have multiple levels of of subclasses.
- ▶ The ontology supports all OWL features: constraints, reasoning, inference, etc.

A method of building an ontology, but not by extending an existing taxonomy.

Screenshots from Protégé, free open-source ontology editor

<https://protege.stanford.edu>



Ontology Creation Approaches

2. Using a dedicated ontology tool: Protégé

Example of FIBO Financial Industry Business Ontology Classes

The screenshot displays the Protégé ontology editor interface. On the left, the 'Class Hierarchy' pane shows a tree structure starting from 'owl:Thing', with 'Agreement' as a top-level class. Under 'Agreement', there are several subclasses, including 'contract', 'credit agreement', 'debt instrument', 'mutual contractual agreement' (which is selected and highlighted in blue), 'service agreement', 'transferable contract', and 'unilateral contract'. The 'mutual contractual agreement' class has several subclasses listed below it, such as 'RepurchaseAgreement', 'agency agreement', 'non-tradable debt instrument', 'over-the-counter instrument', 'account-specific service agreement', 'underwriting arrangement', 'security underwriting arrangement', 'negotiable security', 'UnrestrictedShare', 'CommonNonVotingUnrestrictedFullyPaid', 'CommonVotingUnrestrictedFullyPaidReg', 'promissory note', 'verbal contract', 'written contract', 'collateral agreement', and 'financial instrument'.

On the right, the 'Class: mutual contractual agreement' pane shows the following details:

- IRI:** <https://spec.edmouncil.org/fibo/ontology/FND/Agreements/Contracts/MutualContractualAgreement>
- Annotations:**
 - rdfs:label:** mutual contractual agreement (lang)
 - skos:definition:** contract between named parties whose individual rights and obligations are not transferable to another party without prior written permission (lang)
 - rdfs:comment:** QName: fibo-fnd-agr-ctr:MutualContractualAgreement (lang)
 - rdfs:isDefinedBy:** <https://spec.edmouncil.org/fibo/ontology/FND/Agreements/Contracts/MutualContractualAgreement> (lang)
 - explanatory note:** A mutual contractual agreement involves an exchange of a promises in which the promises made by each party represent considerations supporting the promises of the other party(ies). (lang)
 - synonym:** bilateral contract (lang)
- Parents:**
 - contract
 - mutual agreement
- Relationships:** (Empty table with columns for property, value, and language)

Ontology Creation Approaches

2. Using a dedicated ontology tool: Protégé

Example of FIBO
Financial Industry
Business Ontology
Relations

The screenshot shows the Protégé ontology editor interface. The top navigation bar includes 'Classes', 'Properties', 'Individuals', 'Comments', 'Changes by Entity', and 'History'. The 'Properties' tab is active, showing a 'Property Hierarchy' tree on the left. The tree is expanded to show the 'authorizes' property under the 'owl:topObjectProperty' root. The 'authorizes' property is selected, and its configuration is shown in the main panel. The configuration includes the IRI, annotations, domain, and range.

Property Hierarchy

- owl:topObjectProperty
 - acts in
 - authorizes through
 - is controlling party in role
 - is owning party in role
 - acts on
 - authorizes**
 - delegates control to
 - designates signatory
 - has beneficiary
 - holds shares in
 - is controlling affiliate of
 - is parent company of
 - is general partner of
 - is limited partner of
 - is party controlling
 - advises
 - has portfolio company
 - is beneficial owner of
 - is controlling member of
 - is managing member of
 - is principal party of
 - is director of
 - is guarantor of
 - is officer of
 - is trustee of
 - applies in
 - buys

Property: authorizes

IRI: <https://spec.edmcouncil.org/fibo/ontology/BE/OwnershipAndControl/Executives/authorizes>

Annotations

Property	Value	Language
rdfs:label	authorizes	lang
skos:definition	endorses, enables, empowers, or gives permission to	lang
rdfs:comment	QName: fibo-be-oac-exec:authorizes	lang
rdfs:isDefinedBy	https://spec.edmcouncil.org/fibo/ontology/BE/Owl	lang

Enter property: Enter value: lang

Domain

authorizing party

Enter a class name:

Range

authorized party

Enter a class name:

2. Using a dedicated ontology tool: Protégé

Example of FIBO Financial Industry Business Ontology

Attributes

The screenshot displays the Protégé ontology editor interface for the FIBO ontology. The top navigation bar includes 'Classes', 'Properties', 'Individuals', 'Comments', 'Changes by Entity', and 'History'. The 'Properties' tab is active, and the 'Data Properties' sub-tab is selected. The left pane shows a tree view of the ontology hierarchy, with 'owl:topDataProperty' expanded to show various data properties, including 'allows payment in kind'. The right pane shows the details for the selected property, 'allows payment in kind', including its IRI, annotations, characteristics, domain, and range.

Property Hierarchy

- owl:topDataProperty
 - allows auto-reinvestment
 - allows payment in kind**
 - excludes energy and food
 - has URL
 - has website
 - has address line 1
 - has address line 2
 - has address line 3
 - has attention line
 - has best measure
 - has city
 - has collection size
 - has population size
 - has universe size
 - has count
 - has number of entries
 - has date specification
 - has date time stamp value
 - has date time value
 - has date value
 - has delisting date
 - has description
 - has duration value
 - has entity expiration date
 - has estate or death put feature
 - has event date value
 - has exponent

Property: allows payment in kind

IRI

https://spec.edmcouncil.org/fibo/ontology/SEC/Debt/DebtInstruments/allowsPaymentInK

Annotations

rdfs:label	allows payment in kind	lang	X
rdfs:comment	QName: fibo-sec-dbt-dbt:allowsPaymentInKind	lang	X
rdfs:isDefinedBy	https://spec.edmcouncil.org/fibo/ontology/SEC/I	lang	X
skos:definition	indicates whether the principal may be repaid in kind (i.e., replaced with another instrument) rather than in cash	lang	X

Characteristics

Functional

Domain

redemption provision

Enter a class name

Range

xsd:boolean

Enter a datatype name

Ontology Creation Approaches

2. Using a dedicated ontology tool: Protégé

Example of FIBO Financial Industry Business Ontology

Individuals of a class

The screenshot shows the Protégé ontology editor interface. The top navigation bar includes 'Classes', 'Properties', 'Individuals', 'Comments', 'Changes by Entity', and 'History'. The 'Individuals' tab is active, showing a list of individuals under the 'Jurisdiction' class. The 'Federal Reserve Fourth District' is selected. The right pane displays the details for this individual, including its IRI, annotations, types, and relationships.

Individuals by Class

Class: Jurisdiction

Individuals:

- American Samoa jurisdiction
- Canadian jurisdiction
- District of Columbia jurisdiction
- England and Wales jurisdiction
- European Union jurisdiction
- Federal Reserve Eighth District
- Federal Reserve Eleventh District
- Federal Reserve Fifth District
- Federal Reserve First District
- Federal Reserve Fourth District**
- Federal Reserve Ninth District
- Federal Reserve Second District
- Federal Reserve Seventh District
- Federal Reserve Sixth District
- Federal Reserve Tenth District
- Federal Reserve Third District
- Federal Reserve Twelfth District
- German jurisdiction
- Guam jurisdiction
- jurisdiction of Albania
- jurisdiction of Andorra

Individual: Federal Reserve Fourth District

IRI: <https://spec.edmcouncil.org/fibo/ontology/FBC/FunctionalEntities/NorthAmericanEntities/USRegulatoryAgencies/FederalReserveFourthDistrict>

Annotations

- rdfs:label: Federal Reserve Fourth District
- skos:definition: jurisdiction of the Fourth District of the Federal Reserve, which covers Ohio, western Pennsylvania, the northern panhandle of West Virginia, and eastern Kentucky
- rdfs:comment: QName: fibo-fbc-fct-usjrga:FederalReserveFourthDistrict
- rdfs:isDefinedBy: <https://spec.edmcouncil.org/fibo/ontology/FBC/FunctionalEntities/NorthAmericanEntities/USRegulatoryAgencies/FederalReserveFourthDistrict>
- adapted from: <https://www.clevelandfed.org/en/About%20Us/Who%20We%20Are.aspx>

Types

- Federal Reserve district

Relationships

- has reach: Kentucky
- has reach: Ohio
- has reach: Pennsylvania
- has reach: West Virginia

Importing taxonomies into an OWL-based ontology or dedicated ontology tool.

Not recommended approach to extending a taxonomy into an ontology:

Why not?

- ▶ All taxonomy hierarchies get converted to class-subclass hierarchies.
 - ▷ The class-subclass hierarchy in ontologies is of the hierarchical type generic-specific (“is a kind of”) only.
 - ▷ Importing taxonomies into ontologies will incorrectly treat:
 - whole-part taxonomy relations (e.g. geographic) as class-subclass relations
 - generic-instance taxonomy relations as class-subclass relations, not class-instance affiliations
- ▶ Alternative labels can import, but as “Annotation” properties, and do not function as alternative labels for tagging and search.

3. In a combined taxonomy/ontology tool, create a taxonomy and an ontology and link them, with the ontology functioning as a semantic layer.
 - ▶ The ontology comprises classes and subclasses to the extent needed to describe the generic characteristics of the semantic model.
 - ▶ The ontology does not include all possible levels of hierarchy, nor any instances; More of the hierarchy and the instances reside in the SKOS taxonomy.

Ontology Creation Approaches

3. In a combined taxonomy/ontology tool, create a taxonomy and an ontology and link them, with the ontology functioning as a semantic layer.

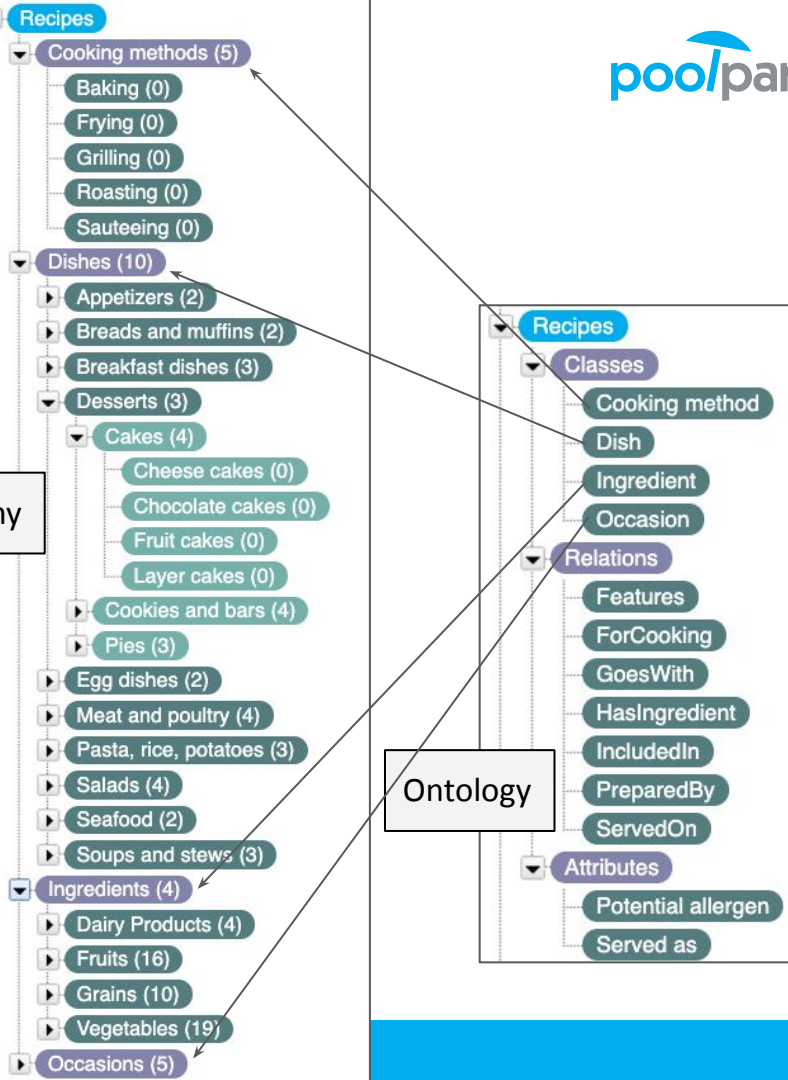
Taxonomy + ontology layer

The ontology tends to be smaller and simpler.

Taxonomy may be based on SKOS, whereas ontology is based on OWL, But they are compatible, based on RDF.

Taxonomy

Ontology

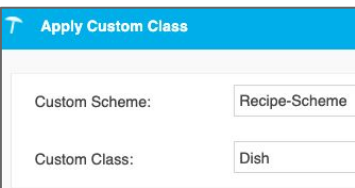
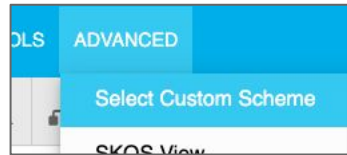


Screenshots from PoolParty

Extending SKOS taxonomy concepts to be part of an ontology

- ▶ Ontology class labels correspond/match the SKOS concept scheme or concept labels to which they will be applied
 - ▷ The ontology “layer” is not an upper hierarchical layer, but an **overlay** to the higher levels of the SKOS project.
 - ▷ Tip: consider using singular for ontology class names and plural for SKOS concept labels
- ▶ There is no dilemma in determining if an entity is an individual/instance or a class, since the ontology layer comprises only classes.

Extending SKOS taxonomy concepts to be part of an ontology



1. An ontology or custom scheme is applied/linked to the taxonomy project or to a specific concept scheme of a project.
2. Classes are applied to the levels of concept schemes, top concepts, or broad-level concepts, as appropriate. (Most often to concept schemes.)
3. Class properties (relations and attribute types) are then inherited by all narrower concepts.
4. Relevant relations and attributes are available for all concepts in the taxonomy, based on their class assigned to their broader concept or concept scheme.
5. Instantiating a relation between a pair of specific concepts or adding values to attributes may be done manually or in a batch mode through importing spreadsheet tables.

Applying Ontologies to Taxonomies

Extending SKOS taxonomy concepts to be part of an ontology

Instantiating relations between specific concepts and adding values to attributes done in a batch mode through importing a spreadsheet table

Concept	Class (URI)	Relation (URI)	Relation (URI)	Attribute (URI)	Attribute (URI)
concept	type	https://k2.poolparty.biz/Jobs/belongsToTo	https://k2.poolparty.biz/Jobs/has-HQ-in	https://k2.poolparty.biz/Jobs/Address	https://k2.poolparty.biz/Jobs/Phone-number
Volkswagen	https://k2.poolparty.biz/Jobs/Employer	Automotive	Germany	Dieselstraße 28, 38446 Wolfsburg, Germany	49-5361-90
Accenture	https://k2.poolparty.biz/Jobs/Employer	Information technology and services	Ireland	Unit 1, Grand Canal Square, Grand Canal Quay,	353 1 407 6000
BASF	https://k2.poolparty.biz/Jobs/Employer	Chemicals	Germany	Carl-Bosch-Straße 38, 67056 Ludwigshafen/Rh	49 621 600
Tata Consultancy Services	https://k2.poolparty.biz/Jobs/Employer	Information technology and services	India	TCS House, Raveline Street, Mumbai 400001, M	91-22-6778 9999
Nestlé	https://k2.poolparty.biz/Jobs/Employer	Food and beverages	Switzerland	Av. Nestlé 55, 1800 Vevey, Switzerland	41 21 924 1111
Royal Dutch Shell	https://k2.poolparty.biz/Jobs/Employer	Oil and gas	Netherlands	Carel van Bylandtlaan 16, 2596 HR The Hague,	31 70 3779111
Toyota Motor	https://k2.poolparty.biz/Jobs/Employer	Automotive	Japan	1 Toyota-Cho, Toyota City, Aichi Prefecture, Jap	81 565 282121
Apple Inc	https://k2.poolparty.biz/Jobs/Employer	Computer hardware	United States	1 Infinite Loop Cupertino, CA 95014	1 408 996-1010
Samsung Electronics	https://k2.poolparty.biz/Jobs/Employer	Consumer electronics	South Korea	416, Maetan 3-Dong, Yeongtong-Gu, Suwon, G	82 31 2001114

Applying Ontologies to Taxonomies

Concepts have both:
SKOS relationships and properties

The screenshot displays the Poolparty interface. On the left is a taxonomy tree under the 'Cooking' root. The tree includes categories like 'Cooking methods (5)', 'Dishes (10)', 'Desserts (4)', and 'Cakes (4)'. The 'Fruit cakes (0)' concept is highlighted in orange. On the right, the 'Fruit cakes' concept is detailed. It shows a URL, a 'Dish' type, and a 'SKOS' label. The 'SKOS' label is highlighted with a red box. Below this, there are sections for 'Broader Concepts' (Cakes), 'Narrower Concepts', 'Related Concepts', and 'Top Concept of Concept Schemes'. On the right side of the details panel, there are sections for 'Preferred Label' (Fruit cakes), 'Alternative Labels' (Christmas cake (dried fruit), Fruitcake), 'Hidden Labels', 'Scope Notes', and 'Definitions'.

Screenshots from Poolparty

Applying Ontologies to Taxonomies

Concepts have both:
SKOS relationships and properties *and* OWL-based semantic relationships and attributes from an ontology-based custom scheme.

Screenshots from Poolparty

The screenshot displays the Poolparty interface. On the left is a taxonomy tree under 'Cooking', with 'Fruit cakes (0)' highlighted. On the right is the 'Fruit cakes' concept detail view. The 'SKOS' tab is active, and the 'Recipe-Scheme' sub-tab is highlighted with a red box. The detail view shows various properties:

- For occasion:** Winter holidays
- Calories:** 200 (Integer)
- Goes with:** Chocolate cakes, Whipped cream
- Preparation time:** 2 hours 15 minutes (xsd:string)
- Has main ingredient:** Dried fruit
- Served:** Room temperature (xsd:string)
- Prepared by:** Baking

Benefits of combining a high-level ontology as a semantic layer with a taxonomy

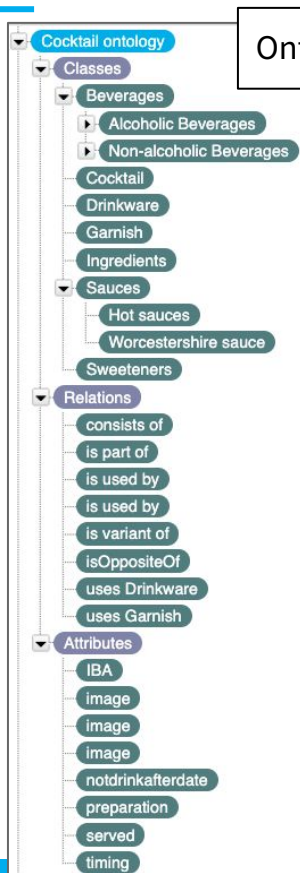
- ▶ Makes use of existing taxonomies, even multiple taxonomies
- ▶ Easier to model the ontology
 - ▷ Existing taxonomies provide a basis for knowledge modeling.
 - ▷ No need to distinguish between sub-classes and individuals.
- ▶ Supports expert specialization
 - ▷ Taxonomists develop and maintain taxonomies.
 - ▷ Ontologists develop and maintain the ontology.
- ▶ More flexible and adaptable
 - ▷ The taxonomy changes more frequently than does the ontology.
 - ▷ Taxonomies can easily be added.
- ▶ Different purposes served
 - ▷ The ontology is for modeling, reasoning, and analysis.
 - ▷ The taxonomy is for tagging and information/data retrieval.

Taxonomy + Ontology Implementation Example

Taxonomy



Ontology



Taxonomy enriched with an ontology in PoolParty

The PoolParty interface displays a taxonomy tree on the left, which is enriched with an ontology. The "Harvey Wallbanger" concept is highlighted in orange. The right pane shows the details for the "Cocktail" concept, including its SKOS scheme, "consists of" (Galliano, Orange juice, Vodka), "is variant of", "uses Drinkware" (Highball glass), and "uses Garnish" (Cherry, Orange). The "preparation" and "served" properties are also visible.

Harvey Wallbanger
<http://vocabulary.semantic-web.at/cocktails/018cc862-9056-49be-b636-cd228d7262d>

Cocktail

Details Notes Documents Linked Data Triples Visualization Quality Management History

SKOS Cocktail scheme

consists of

- Galliano
- Orange juice
- Vodka

is variant of

- Highball glass

uses Drinkware

- Highball glass

uses Garnish

- Cherry
- Orange

IBA http://www.iba-world.com/index.php?option=com_content&id=263&Itemid=532

image http://upload.wikimedia.org/wikipedia/commons/4/44/Harvey_Wallbanger.jpg

preparation
Stir the vodka and orange juice with ice in the glass, then float the Galliano on top. Garnish and serve.

served
On the rocks; poured over ice

GRAPHSEARCH

Search in All Facets EN

Search Space

Cocktail Search

Facets

Entity types

- Search Facet Values
- Ingredients - 198
- Cocktail - 91
- Garnish - 30
- Drinkware - 14

consists of

is part of

is used by

is used by

is variant of

uses Drinkware

uses Garnish

Selected facet values: empty

302 results

Kir-Breton Tommy's Marg... Americano Godmother White Lady B-52 Aviation French Connec... Ramos Gin Fizz Dirty Martini

Page: 1 Results per page: 10 1 - 10 of 302

Semantic relations as facets in PoolParty GraphSearch








<https://vocabulary.semantic-web.at/GraphSearch/>

GRAPHSEARCH

Search in All Facets EN

Selected facet values: Ingredients (Entity types) X Planter's Punch (is part of) X Clear all X

7 results

 Angostura bitt...	 Orange juice	 Pineapple juice	 Grenadine	 Dark rum	 Lemon juice	 Sugar syrup
--	---	--	---	---	--	--

Page: 1 X Results per page: 10 1 - 7 of 7

Semantic relations as facets in PoolParty GraphSearch

<https://vocabulary.semantic-web.at/GraphSearch/>

© Semantic Web Company 2021 47

Search Space

Cocktail Search

Facets

Entity types

Search Facet Values

Cocktail - 7

consists of

Search Facet Values

- Lemon juice - 7
- Gin - 4
- Triple sec - 4
- Bons Bois - 2
- Maraschino - 2
- Crème de violette - 1
- Egg white - 1
- Orange bitters - 1
- Raspberry syrup - 1
- Vodka Citron - 1

is part of

is used by

is used by

is variant of

uses Drinkware

Search Facet Values

Martini glass - 7

Selected facet values: Cocktail (Entity types) X Lemon juice (consists of) X Martini glass (uses Drinkware) X Clear all X

7 results

Casino

Lemon Drop M...

Clover Club Co...

Sidecar

Between the S...

White Lady

Aviation

<https://vocabulary.semantic-web.at/GraphSearch/>

Semantic relations as facets in PoolParty GraphSearch

Extending a taxonomy to be an ontology

- ▶ Provides more uses than a taxonomy alone or an ontology alone
- ▶ Makes use of existing taxonomies
- ▶ Can conform to both taxonomy and ontology standards for interoperability
 - ▷ Start with a taxonomy in SKOS
- ▶ Is best done in a combined taxonomy/ontology tool based on W3C/
Semantic Web standards

Questions/Contact

Heather Hedden

Data and Knowledge Engineer
Semantic Web Company Inc.
One Boston Place, Suite 2600
Boston, MA 02108

857-400-0183

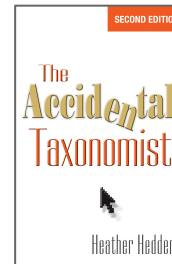
heather.hedden@semantic-web.com

www.linkedin.com/in/hedden

<https://twitter.com/hhedden>

Semantic Web Company www.semantic-web.com

PoolParty Semantic Suite www.poolparty.biz



The Accidental Taxonomist 25% off print edition
with coupon code KMW21 through December 15
<https://books.infotoday.com/books/Accidental-Taxonomist-2nd-Edition.shtml>

