poolparty®

SEMANTiCS
Karlsruhe 2019

# Fast Track to Knowledge Graphs and Semantic AI

## How Semantic Knowledge Models and Machine Learning Enable Semantic AI Applications

**Heather Hedden**
Hedden Information Management

**Andreas Blumauer**
CEO of Semantic Web Company

# Agenda

- **Module 1: Taxonomies and Ontologies - Theory and Practice**
  - Part 1: 09:00 - 10:30
    - Short break
  - Part 2: 11:00 - 12:30

- **Lunch Break**

- **Module 2: Knowledge graphs**
  - 13:30 - 15:30
- **Module 3: Semantic AI Applications**
  - 15:45 - 17:00

# Fast Track to Knowledge Graphs and Semantic AI

## Module 1:
## Taxonomies and Ontologies - Theory and Practice

**presented by**
**Heather Hedden**

# About Heather Hedden

- Taxonomy consultant
  – Independent, through Hedden Information Management (since 2004)
  – Employed, through Project Performance Corporation, and contract
- Former staff taxonomist
  – At various companies: Gale/Cengage Learning, Viziant, First Wind
- Instructor of online and onsite taxonomy courses
  – Independently through Hedden Information Management
  – Previously at Simmons University - Library & Information Science School
- Author of *The Accidental Taxonomist* (2010, 2016, Information Today, Inc.)
- Former indexer of books and database content (articles, images, etc.)

# Outline

1. Introduction to taxonomies and ontologies
2. Purposes and benefits
3. Types of knowledge organization systems
4. Creating concepts
5. Creating relationships
6. Taxonomy structural design: hierarchies and facets
7. Standards: SKOS, RDF, RDF Schema, and OWL
8. Creating ontologies
9. Implementation issues
10. Linked data and the Semantic Web

# Outline

1. Introduction to taxonomies and ontologies
2. Purposes and benefits
3. Types of knowledge organization systems
4. Creating concepts
5. Creating relationships
6. Taxonomy structural design: hierarchies and facets
7. Standards: SKOS, RDF, RDF Schema, and OWL
8. Creating ontologies
9. Implementation issues
10. Linked data and the Semantic Web

# Introduction

## Taxonomy

- From ancient Greek "taxis," meaning arrangement + "nomia," meaning method.
- Originally meant the science and practice of naming and classifying.
- Originally in biology and then in any field.
- Aspect of "classification" is still important to the definition of taxonomies.

## Ontology

- From ancient Greek “onto,” meaning being + “logia,” meaning logical discourse.
- Originally meant the philosophical study of being.
- In information science, the naming of concepts, categories, properties, entities, relations, etc. that make up a domain of knowledge.

Definitions are not rigid. Taxonomies may merge into ontologies.

# Introduction

## Taxonomies and Ontologies are types of Knowledge Organization Systems (KOS)

- Any system of terms, concepts, terminology, classification, etc. to organize and define knowledge.
- Comprises concepts, labels, relationships between them, *and* models of how information/knowledge can be managed and organized. (specifications, policies, etc.)
- Sometimes called a “vocabulary” or “controlled vocabulary,” but is more than a simple list of terms.

KOS types:

authority files
categorization schemes
classification schemes
dictionaries
gazetteers
glossaries
**ontologies**
semantic networks
subject heading schemes
synonym rings
**taxonomies**
terminologies
thesauri

# Outline

1. Introduction to taxonomies and ontologies
2. Purposes and benefits
3. Types of knowledge organization systems
4. Creating concepts
5. Creating relationships
6. Taxonomy structural design: hierarchies and facets
7. Standards: SKOS, RDF, RDF Schema, and OWL
8. Creating ontologies
9. Implementation issues
10. Linked data and the Semantic Web

# Purposes and Benefits

## KOS uses

1. Management and retrieval/findability/discoverability of *internal* content by users within an organization
    - Intranet/SharePoint, content management system, document management system, digital asset management system, records management, linking structured and unstructured data, data warehouses, data lakes, etc.
2. Enabling retrieval/findability/discoverability of information by *external* users
    - Databases of published articles, information resource websites, products/services for sale, government website public information, business exchanges, search engine optimization (SEO), etc.

# Purposes and Benefits

## KOS uses and applications

1. Indexing/tagging support
    a) Manual indexing
    b) Automated indexing
2. Retrieval support
    a) In search
    b) In browse
        - Alphabetical browse
        - Hierarchical browse
        - Faceted browse (usually to limit search)

# Purposes and Benefits

## Indexing/tagging support

- For indexing documents, images, or other digital assets.
- For manual indexing/tagging, as an aid to indexers
- For automated indexing, as a basis for rules or examples
  (two different methods of auto-categorization)
- For ensuring consistent indexing across multiple content items of different sources/creators with different wording

# Purposes and Benefits

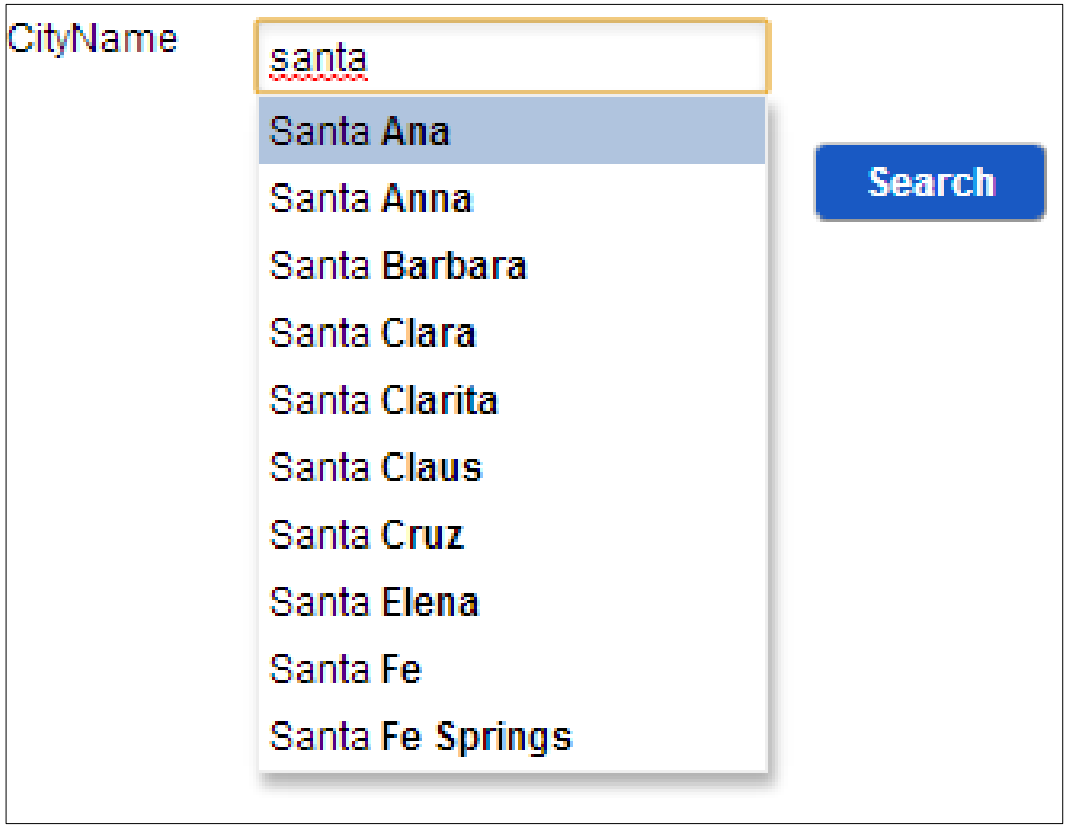
## Retrieval support: in search

Controlled list of terms and their synonyms/equivalents to aid online retrieval

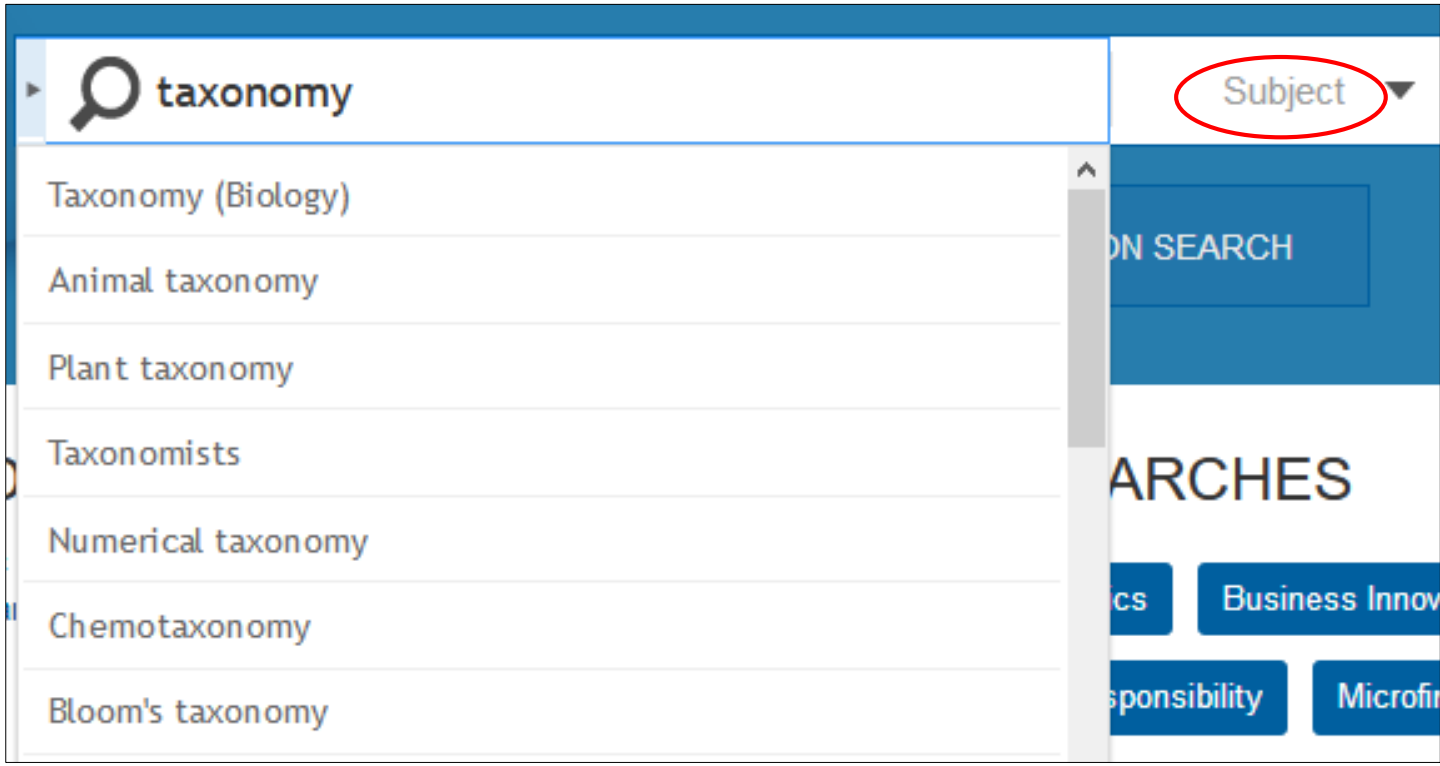
- For website or intranet search engines, online databases, online directories, enterprise search
- Might be displayed as type-ahead auto-suggest terms, or might not be displayed at all.
- Does not matter how content was indexed (manual or automated)

# Purposes and Benefits

## Retrieval support: in search




Type-ahead




Search-suggest

# Purposes and Benefits

## Retrieval support: in browse

### a) Alphabetical browse

Display method for thesauri, name/proper noun lists, and book-style indexes

Example of an alphabetical browse thesaurus:

ERIC (Educational Resources Information Center) Thesaurus
Institute of Education Sciences

https://eric.ed.gov/?ti=all

ERIC

Collection | Thesaurus

Search thesaurus descriptors

Search | Browse Thesaurus

☐ Include Synonyms ☐ Include Dead terms

## Purpose and Scope

The ERIC Thesaurus is a list of terms representing research topics in the field of education. Descriptors from the ERIC Thesaurus are assigned to every document in the ERIC digital library to describe its subject content.

Terms in the ERIC Thesaurus represent the vocabulary used in the documents that comprise the ERIC digital library collection. L
updated in this archived webinar.

The ERIC Thesaurus contains a total of 11,761 terms. There are 4,539 Descriptors and 7,089 Synonyms. There are also 133 De
Descriptors but remain in the Thesaurus to aid in searching older records. The ERIC Thesaurus was last updated in March 2019
new Descriptors, 19 new Synonyms, and changes to 184 existing terms. For more information, view the full list of updates and do

Terms in italics are variants/alternative labels redirecting to the preferred label of the concept.

## Browse Alphabetically

# A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

*Abbreviations (2004)*
Ability
Ability Grouping
Ability Identification
*Able Students (1966 1978)*
*Ableism (2004)*
*Abnormal Psychology*
*Aboriginal Australians (2004)*

African American Education
African American Employment
African American Family
African American History
African American Influences
African American Institutions
African American Leadership
African American Literature

*Applied Reading (1966 1980)*
*Applied Research*
*Applied Sciences*
*Appraisal*
Apprenticeships
Appropriate Technology
*Appropriations (Federal)*
*Appropriations (State)*

# Purposes and Benefits

## Retrieval support: in browse

### b) Hierarchical browse

Categorization scheme for information organization, classification, guided search

- For web site structural design, online information services, intranet content organization, content management system “folders”

# Purposes and B

## Retrieval Support

Hierarchical browse taxonomy

Example: Craigslist Boston

https://boston.craigslist.org

### community

activities
artists
childcare
classes
events
general
groups
local news
lost+found
missed connections
musicians
pets
politics
rants & raves
rideshare
volunteers

### services

automotive
beauty
cell/mobile
computer
creative
cycle
event
farm+garden
financial
household
labor/move
legal
lessons
marine
pet
real estate
skilled trade
sm biz ads
travel/vac
write/ed/tran

### discussion forums

android
apple
arts
frugal
gaming
garden
pets
philos
photo

### housing

apts / housing
housing swap
housing wanted
office / commercial
parking / storage
real estate for sale
rooms / shared
rooms wanted
sublets / temporary
vacation rentals

### for sale

antiques
appliances
arts+crafts
atv/utv/sno
auto parts
aviation
baby+kid
barter
beauty+hlth
bike parts
bikes
boat parts
boats
farm+garden
free
furniture
garage sale
general
heavy equip
household
jewelry
materials
motorcycle parts
motorcycles
music instr
photo+video

### jobs

accounting+finance
admin / office
arch / engineering
art / media / design
biotech / science
business / mgmt
customer service
education
etc / misc
food / bev / hosp
general labor
government
human resources
legal / paralegal
manufacturing
marketing / pr / ad
medical / health
nonprofit sector
real estate
retail / wholesale
sales / biz dev
salon / spa / fitness
security
skilled trade / craft
software / qa / dba
systems / network

# Purposes and Benefits

## Retrieval Support: in Browse

### a) Faceted browse/search

Multiple term lists of different types, also called facets/filters/refinements

- Browsed-for facet terms are often used in in combination with entering something into a search field.
- Example of a faceted taxonomy

  NCSU Libraries catalog (browse new titles) www.lib.ncsu.edu/catalog

# Purposes and Benefits

## Benefits of knowledge organization systems

1. Controlled vocabulary aspect

   Brings together different wordings (synonyms) for the same concept

   - Helps people search for information by different names
   - Content is not missed, due to varied names/labels

   Disambiguates identical works with different meanings (homographs) into separate concepts

   - Incorrect content is not retrieved merely because of matching words

2. Classification and structure aspect

   Organizes information into a logical structure

   - Helps people browse or navigate and find topics they did not know existed or how to describe or discover new related topics

# Purposes and Benefits

What is the purpose of a taxonomy, ontology or other KOS in your organization or work?

# Outline

1. Introduction to taxonomies and ontologies
2. Purposes and benefits
3. Types of knowledge organization systems
4. Creating concepts
5. Creating relationships
6. Taxonomy structural design: hierarchies and facets
7. Standards: SKOS, RDF, RDF Schema, and OWL
8. Creating ontologies
9. Implementation issues
10. Linked data and the Semantic Web

# Types of Knowledge Organization Systems

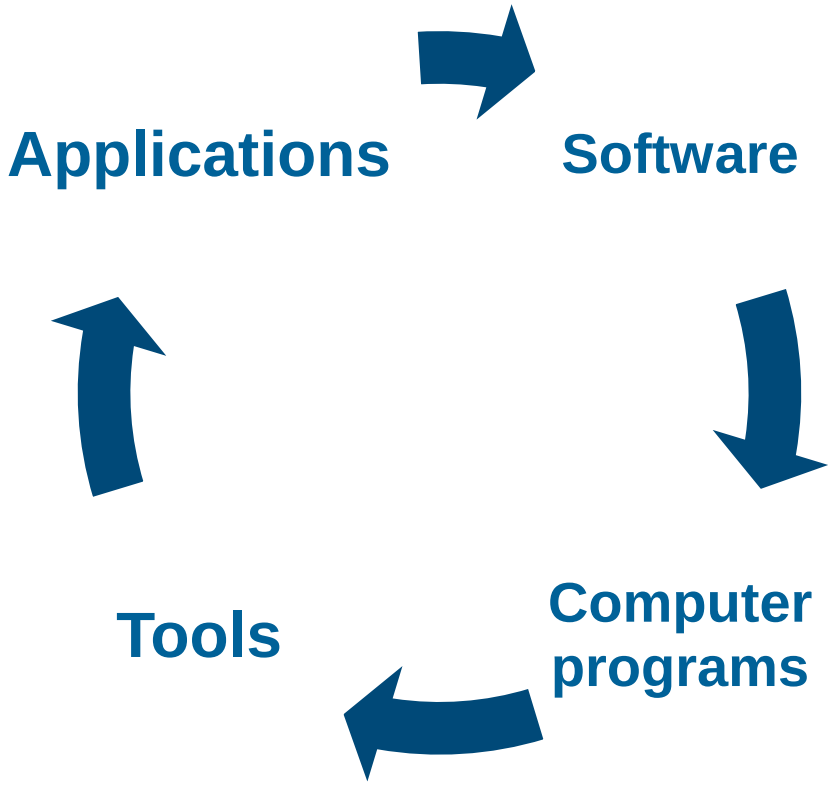
## Common types of knowledge organization systems

- Synonym ring (search-support “thesaurus”)
- Name authority file
- Taxonomy
  - Hierarchical taxonomy
  - Faceted taxonomy
- Thesaurus
- Ontology

# Types of Knowledge Organization Systems

## Synonym ring / search thesaurus

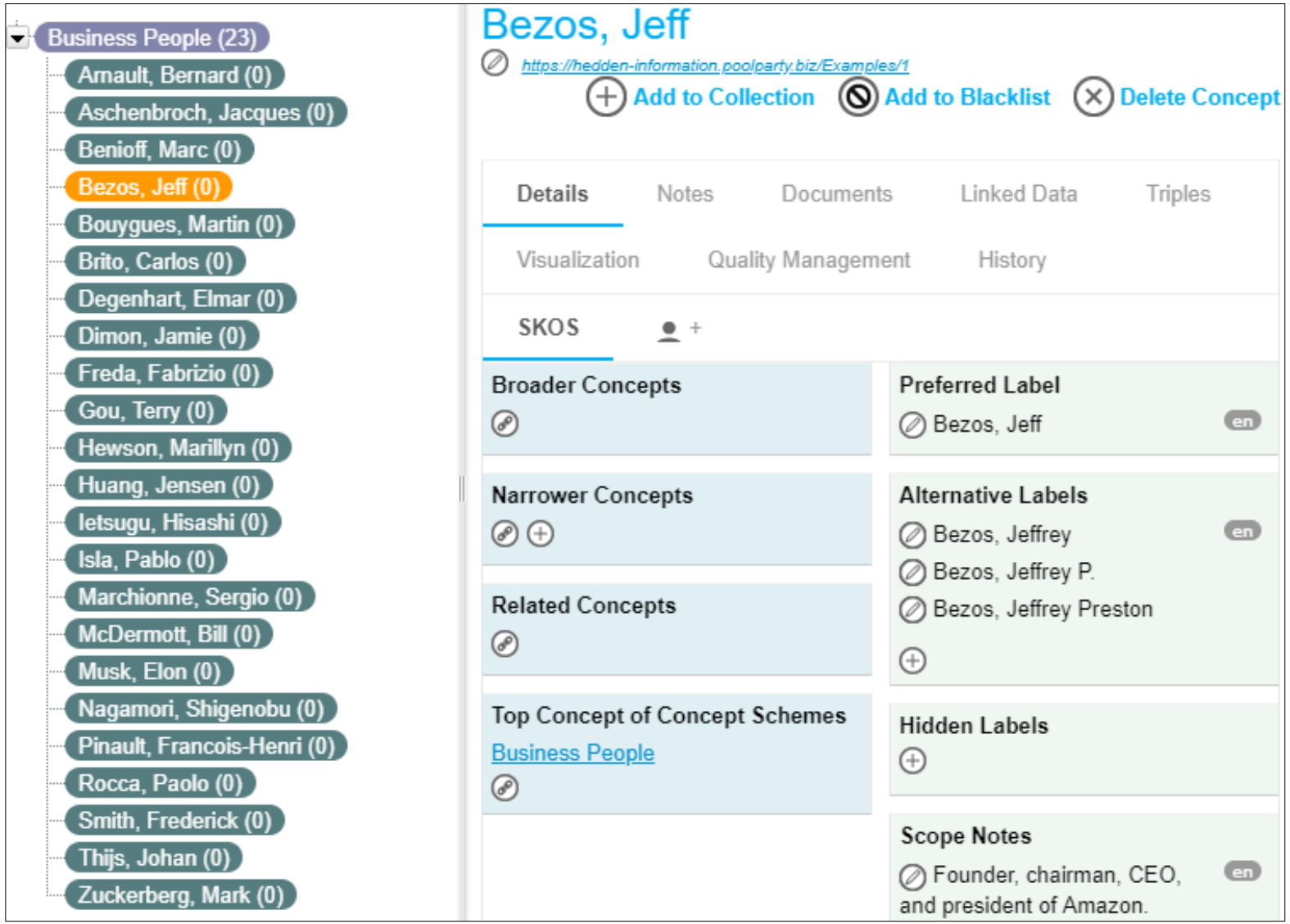
- A controlled vocabulary with synonyms or near-synonyms for each concept
- No designated “preferred” label: All labels are equal and point to each other.
- Concepts/labels are not displayed to the end user.
- Used to support search, where there is no browsing the taxonomy.
- Sometimes called “search thesaurus.”



# Types of Knowledge Organization Systems

## Name authority file

- For named entities, proper nouns
- A controlled vocabulary with preferred names and variant/alternative names.
- Might not have hierarchical relationships between named concepts.
- Usually has additional information for each named concept.



# Types of Knowledge Organization Systems

## Taxonomy

- A KOS with broader/narrower (parent/child) relationships that include all concepts to create a hierarchical structure
- Has focus on categorizing and organization concepts
- May or may not have “synonyms” to point to the correct, preferred terms/labels
- May comprise several hierarchies or facets
  (A facet can be considered a hierarchy.)

➢ “Taxonomy” may refer to any controlled vocabulary (term lists, synonym rings, authority files, classification schemes, thesauri, etc.),
but does not include ontologies

# Types of Knowledge Organization Systems

Taxonomy Examples

Leisure and culture
. Arts and entertainment venues
. . Museums and galleries
. Children's activities
. Culture and creativity
. . Architecture
. . Crafts
. . Heritage
. . Literature
. . Music
. . Performing arts
. . Visual arts
. Entertainment and events
. Gambling and lotteries
. Hobbies and interests
. Parks and gardens
. Sports and recreation
. . Team sports
. . . Cricket
. . . Football
. . . Rugby
. . Water sports
. . Winter sports
. Sports and recreation facilities
. Tourism
. . Passports and visas
. Young people's activities

Hierarchical Taxonomy Example

**Career Level**
- Student
- Entry Level
- Experienced
- Manager
- Director
- Executive

**Function**
- Customer Service & Support
- Delivery
- Engineering
- Finance
- General Management
- Legal & Regulatory Affairs
- Marketing & Advertising

[more]

**Industry**
- Agriculture
- Apparel & Fashion
- Automotive
- Aviation & Aerospace
- Banking
- Biotechnology
- Broadcast Media
- Chemicals

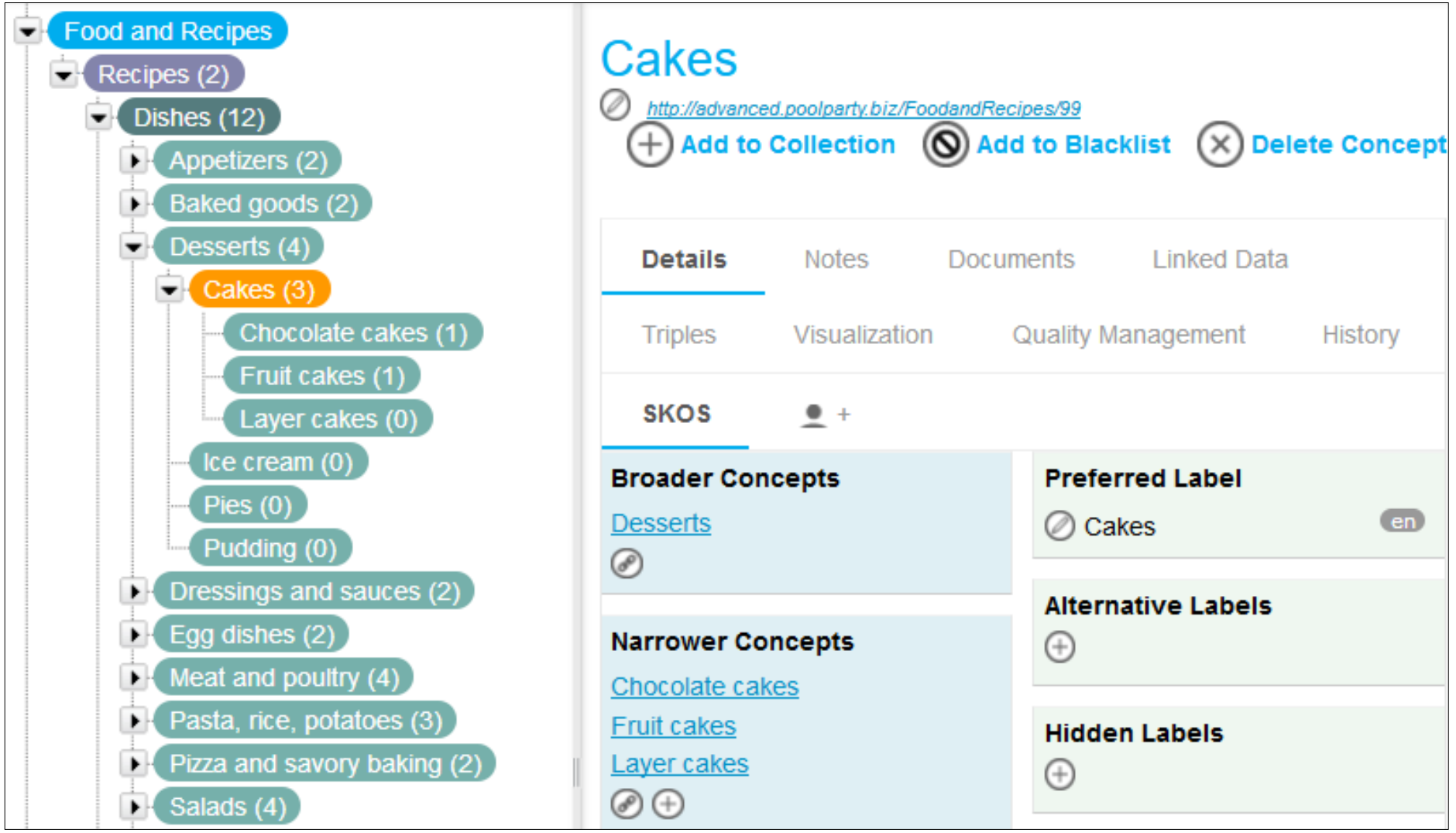
[more]

Faceted Taxonomy Example

# Types of Knowledge Organization Systems

## Hierarchical taxonomy

Concepts have broader concepts and narrower concepts.



# Types of Knowledge Organization Systems

## Thesaurus

- A KOS that has standard structured relationships between terms/concepts
  - Hierarchical: broader term/narrower term (BT/NT)
  - Associative: related terms (RT)
  - Preferred terms and nonpreferred terms (as equivalence relationship USE/UF) or preferred labels and alternative labels.
- Created in accordance with standards:
  - ISO 25964-1 Part 1, *Thesauri and interoperability with other vocabularies*
  - ANSI/NISO Z39.19 *Guidelines for Construction, Format, and Management of Monolingual Controlled Vocabularies*
- The kind of KOS most used in indexing articles for library/academic research
- Have existed, originally in print, since 1960s

# Types of Knowledge Organization Systems

## Ontology

- A more abstract layer in describing a KOS.
- A formal naming and definition of the types, properties and interrelationships of entities in a particular domain.
- A set of precise descriptive statements about some part of the world.
- A form of “knowledge representation.”
- If created according to W3C guidelines (OWL: Web Ontology Language) can enable knowledge linking on the web/Semantic Web.

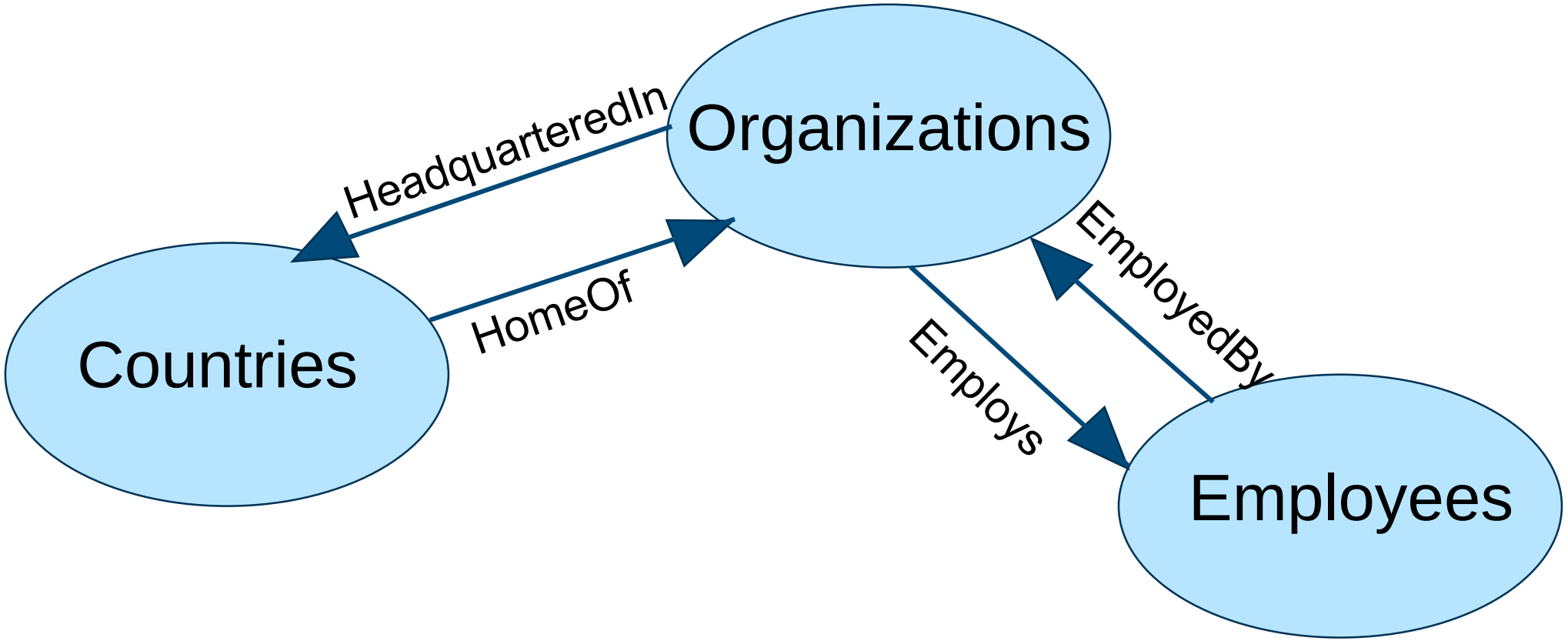
# Types of Knowledge Organization Systems

## Ontology Types

- Upper or core ontologies (top-level ontology, upper model, foundation ontology)
  - A generic, standard framework to serve as a model for a domain ontology, taxonomy, or other KOS
  - Examples: Basic Formal Ontology (BFO), gist, SUMO (Suggested Upper Merged Ontology), SKOS, BIBFRAME, FOAF
- Domain or custom ontologies
  - Concepts belong to a specific subject domain
  - Examples: Systems Biology Ontology, Gene Ontology, BBC Ontology, Financial Industry Business Ontology (FIBO)
- "Ontology" may also refer to a combination of a taxonomy with a custom ontology layer.

# Types of Knowledge Organization Systems

Domain ontology excerpt example



# Types of Knowledge Organization Systems

## Summary of common KOS Types


| Synonym Ring | Authority File | Taxonomy | Thesaurus | Ontology |
|---|---|---|---|---|
| Synonym control | Ambiguity control<br>Synonym control | Ambiguity control<br>(Synonym control)<br>Hierarchical relationships | Ambiguity control<br>Synonym control<br>Hierarchical relationship<br>Associative relationships | Ambiguity control<br>(Synonym control)<br>Semantic relationships<br>Classes<br>Linked data |

# Types of Knowledge Organization Systems

## Quiz

What kind of KOS is most suitable for:

- An ecommerce website
- A database of scholarly articles
- Enterprise search (search box)
- A digital asset management system
- An internal repository of researchers and projects
- A government agency public website
- Data for pharmaceutical product development

# Outline

1. Introduction to taxonomies and ontologies
2. Purposes and benefits
3. Types of knowledge organization systems
4. Creating concepts
5. Creating relationships
6. Taxonomy structural design: hierarchies and facets
7. Standards: SKOS, RDF, RDF Schema, and OWL
8. Creating ontologies
9. Implementation issues
10. Linked data and the Semantic Web

# Creating Concepts

## Concepts in a taxonomy or ontology

- A concept is a unique, unambiguous entity in a KOS/knowledge model, with its own definition and usage.
- The same concept may have multiple names, and the same name/word may refer to multiple concepts, so the focus should be on concepts, not names/words/terms.
- Concepts are tagged/indexed/assigned to content items.
- It should be clear to both those tagging/indexing and those browsing and searching for content what the concept means.
- Concepts are grouped into sets or hierarchies, called Concept Schemes.

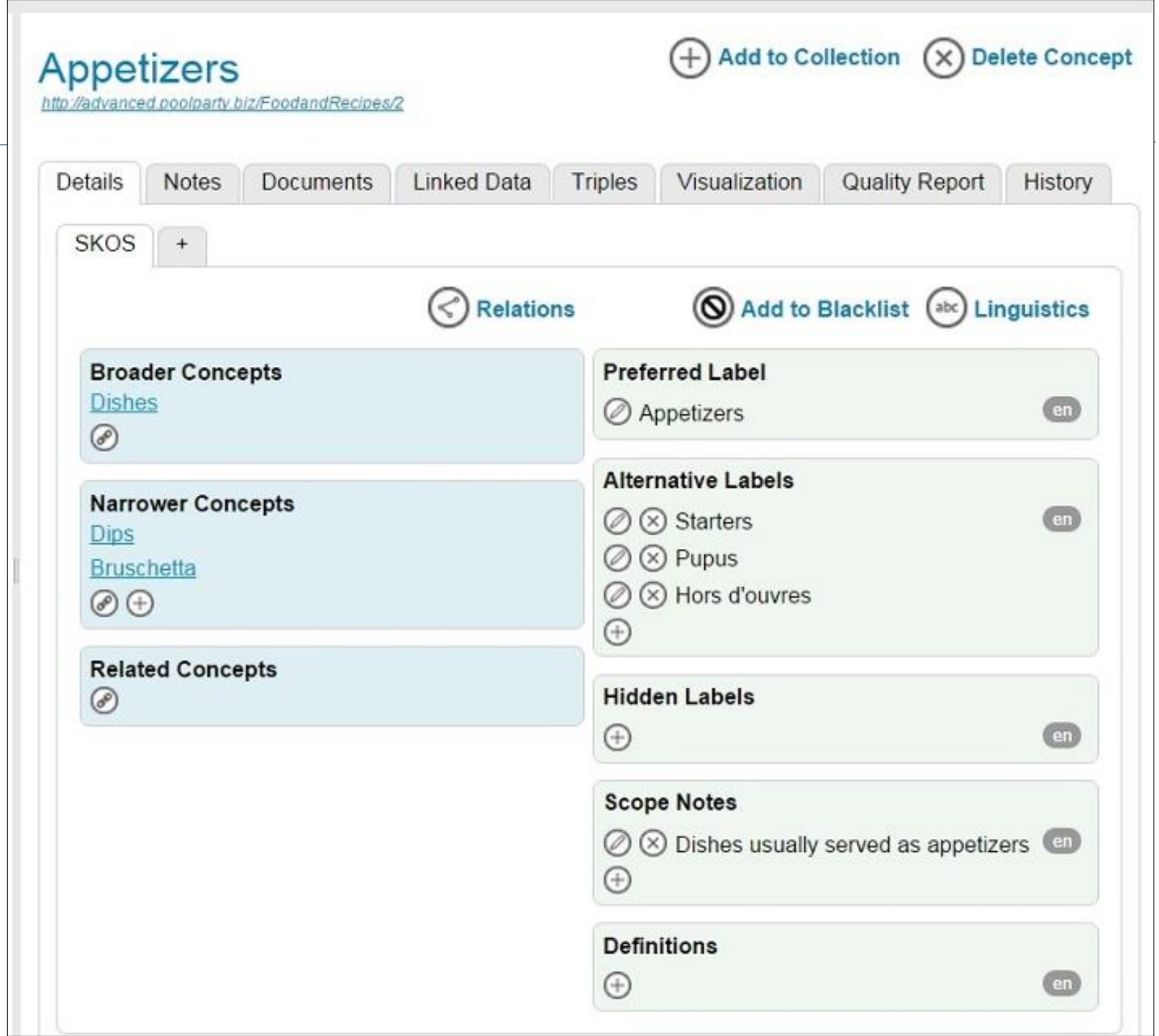
# Creating Concepts

## A concept has:

- Labels
  - A single **preferred label** (in each language, if in a multilingual KOS)
    - The displayed label, when concepts are displayed for browsing in hierarchies or other visualizations
  - Any number of **alternative labels**
    - The labels that support searching by other names
- Metadata
  - A unique identifier number; often a URI
  - Optional notes and other attributes: definition, notes, etc.
  - Other: creation date, last update date, creator, approval status, etc.
- Relationships (of various types) with other concepts

# Creating Concepts

A concept,
its labels,
relationships,
and notes,
as maintained in
taxonomy/ontology
management
software, PoolParty



# Creating Concepts

## Label format and style

- Consistent capitalization
- Single words or multi-word phrases
- Nouns or noun phrases
- Adjectives alone can be concepts only in small navigational taxonomies, where the noun is obvious from context, or in facets (such as colors).
- Countable nouns are usually plural.
- Parenthetical qualifiers may be used for disambiguation, not modification.
- Avoid inversions with commas (e.g. noun, adjective).

# Creating Concepts

## Alternative Labels

- Defined: Approximately synonymous words or phrases to refer to an equivalent concept, for the *context* of the KOS and content (knowledge model)
- Purpose: To capture different wordings of how different people might describe or look up the same concept or idea.
  - Differences between that of the author and the end-user
  - Differences between that of the indexers and the end-users
  - Differences between different indexers, people doing tagging
  - Differences among different end-users
- Serving as “multiple entry points” to look up and retrieve the desired content.
- Enabling consistent indexing/tagging

# Creating Concepts: Alternative Labels

## Guidelines for using alternative labels

- A concept may have any number of (multiple) alternative labels, or it may have no alternative labels.
- An alternative label is associated with only a single concept.
  - Alternative labels cannot be re-used in different concepts (unless there is some weighting scheme, and they are not displayed)
- Alternative labels may be displayed to the end-user or they may not be.
- Alternative labels, may redirect the end-user to the concept with the preferred label (before getting to the content), or they can link directly to the content.

# Creating Concepts: Alternative Labels

## Displayed vs. non-displayed alternative labels

Even when alternative labels are displayed, some may be specially designated for *not* displaying:

- Common misspellings, slang, or deprecated, or potentially offensive terms not displayed to users but can match searches.
- Auto-categorization support but not intended for manual indexing.
- Search support but not intended for type-ahead display.

SKOS model has **Hidden Label** (skos:hiddenLabel) for these uses.

# Creating Concepts

## Concept metadata: notes/documentation

- Concepts may have notes.
- If utilized, not all concepts need notes.
- Free text field associated with the concept in the taxonomy/thesaurus management system.
- May have multiple types/purposes of notes: for end-user, indexer, or both

Types:

- Standard thesaurus note: **Scope Note**
- SKOS-supported notes: **Scope Note**, **Editorial Note**, **Change Note**, **History Note**
- Other SKOS-supported documentation: **Definition**, **Example**

# Creating Concepts

## Concept metadata: additional attributes

- A KOS management system can store additional attribute data about a concept.
- Not part of the SKOS model, but is a standard feature of ontologies
- Typically used for named entities, not so much for subjects

Examples:

- *For Companies*: address, industry code, private/public status
- *For Person names*: title/occupation, birth date, nationality
- *For Products*: part number, price, introduction date
- *For Places*: latitude and longitude

# Outline

1. Introduction to taxonomies and ontologies
2. Purposes and benefits
3. Types of knowledge organization systems
4. Creating concepts
5. Creating relationships
6. Taxonomy structural design: hierarchies and facets
7. Standards: SKOS, RDF, RDF Schema, and OWL
8. Creating ontologies
9. Implementation issues
10. Linked data and the Semantic Web

# Creating Relationships

## Types of relationships between concepts

1. Hierarchical: Broader concept / Narrower concept
2. Associative: Related concept
3. Specific, customized relationships

Relationships are reciprocal between concepts.

Best practices for creating hierarchical and associative relationships are in the thesaurus standards:

*ISO 25964-1 Part 1, Thesauri* and interoperability with other vocabularies
www.iso.org/standard/53657.html, or

*ANSI/NISO Z39.19-2005 Guidelines for the Construction, Format, and Management of Monolingual Controlled Vocabularies*
www.niso.org/publications/ansiniso-z3919-2005-r2010

Good to follow even for taxonomies (not merely thesauri).

# Creating Relationships

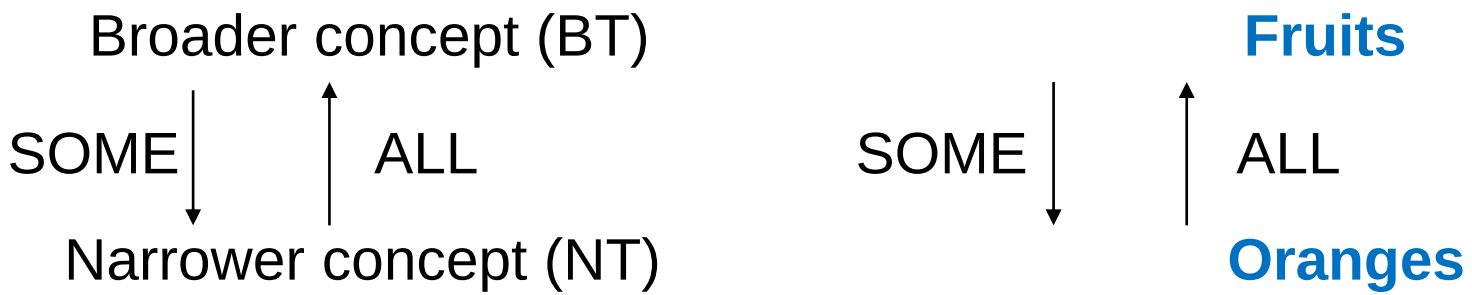
## Hierarchical relationships

- Broader-narrower / Topic-subtopic / Parent-child / Superordinate-Subordinate
- Required feature of both thesauri and taxonomies
- Thesaurus designation of BT / NT (broader term / narrower term)
- SKOS designation: Broader concept / Narrower concept
- Concepts usually have more than one narrower concept, unless they are the most specific concept in the vocabulary.
  (More so in taxonomies than thesauri.)
- On occasion, a concept may have more than one broader concept, referred to as polyhierarchy.

# Creating Relationships

## Hierarchical relationships

Reciprocal (bi-directional) relationships, but asymmetrical




**Fruits** NT **Oranges** **Oranges** BT **Fruits**

Three types:

1. Generic – Specific
2. Generic – Named entity instance: Common noun – Proper noun
3. Whole – Part

# Creating Relationships

## Associative relationships between concepts in different hierarchies:

- Process and agent: Skiing related Skiers
- Process and instrument: Ventilation related Fans (Equipment)
- Process and counter-agent: Bacterial infections related Antibiotics
- Action and property: Environmental protection related Pollution
- Action and product: Glassblowing related Glass containers
- Action and target: Appliance repair related Appliances
- Cause and effect: Hurricanes related Storm surges
- Object and property: Plastics related Elasticity
- Object and origins: Petroleum related Oil wells
- Raw material and product: Timber related Wood products
- Discipline and practitioner: Chemistry related Chemists
- Discipline and object: Literature related Books

# Creating Relationships

## Specific/customized relationships

- Relationships containing meaning: “semantic”
- Variations on hierarchical or associative relationships, but usually associative.
- Reciprocal, but asymmetrical or directional.
- Specific enough to convey the necessary meaning, but not uniquely specific.
- Relationships are between concepts of different types, across different designated *categories* or *classes,* or concept schemes.
- Taxonomist defines the relationships and the categories or classes.
- A required characteristic of ontologies.

# Creating Relationships

## Specific/customized relationships

Sample variations on the associative relationship (RT):

- Has produced the work (WRK) / Created by (CRE)
  - **Twain, Mark** WRK **The Adventures of Tom Sawyer**
  - **The Adventures of Tom Sawyer** CRE **Twain, Mark**
- Produces the product (PRD) / Is manufactured by (MAN)
  - **Apple Inc.** PRD **iPod**
  - **iPod** MAN **Apple Inc.**
- Has member affiliation with (AFF) / Has members (MEM)
  - **Saudi Arabia** AFF **OPEC**
  - **OPEC** MEM **Saudi Arabia**
- For treating (TRE) / Can be treated with the drug (DRUG)
  - **ACE inhibitors** TRE **Hypertension**
  - **Hypertension** DRUG **ACE inhibitors**

# Creating Relationships

Demonstration of creating concepts and relationships in PoolParty

# Outline

1. Introduction to taxonomies and ontologies
2. Purposes and benefits
3. Types of knowledge organization systems
4. Creating concepts
5. Creating relationships
6. Taxonomy structural design: hierarchies and facets
7. Standards: SKOS, RDF, RDF Schema, and OWL
8. Creating ontologies
9. Implementation issues
10. Linked data and the Semantic Web

# Structural Design: Hierarchies

## Hierarchies

- The extension of hierarchical relationships to include all concepts
- More important for taxonomies than other KOS types
- Emphasizes categorization, classification, sorting
- Users navigate from the top down
- Also known as “tree” structures

A single taxonomy may have one or more top-term hierarchies

Hierarchies should be designed to reflect the scope of the content and the view of the users

# Structural Des

## Examples of hierarchies

Higher education disciplines in the United States

Cengage Learning

www.cengage.com/all-disciplines

### Humanities & Social Sciences

- Anthropology
- Art & Humanities
- College Success and Study Skills
- Communication Studies
- Counseling
- Criminal Justice
- Developmental English
- Early Childhood Education
- Education
- English
- History
- Human Services
- Mass Communication
- Music
- Philosophy
- Political Science
- Psychology
- Radio Television & Film
- Religion
- Social Work
- Sociology
- Theatre
- World Languages

### Science, Technology & Mathematics

- Agriculture
- Astronomy
- Biology
- Chemistry
- Computing & Information Technology
- Developmental Math
- Earth Sciences
- Engineering
- Environmental Science
- Forensic Science
- Geography
- Health
- Mathematics
- Nutrition
- Physics
- Statistics
- Veterinary Technology

### Business & Economics

- Accounting
- Business Communication
- Business Law
- Career Success
- Decision Sciences
- Economics
- Finance
- General Business
- Introduction to Business
- Keyboarding
- Keyboard/Computer Education
- Management
- Marketing
- Office Technology
- Taxation

### Professional & Career

- Automotive & Trucking
- Beauty & Wellness
- Career Education
- Culinary, Hospitality, Travel & Tourism
- Emergency Services
- General Interests & Hobbies
- Health Care
- Media Arts & Design
- Paralegal
- Trades

# Structural Design: Hierarchies

## Depth vs. breadth of hierarchy levels: decision factors

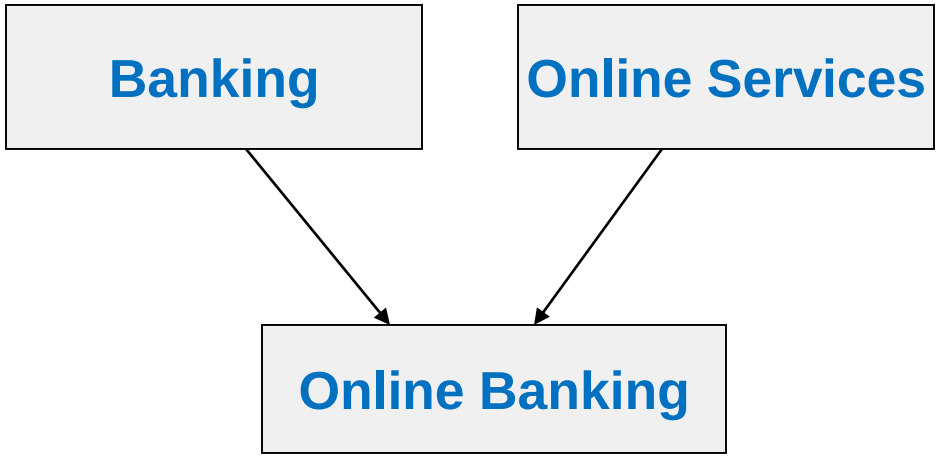
- Display interface horizontal and vertical space
- Multiple clicks to deeper levels on public websites
- More levels lead to less consistency across levels.
- User needs, and expectations
  Industry experts, internal employees, general public, students, etc.

# Structural Design: Hierarchies

## Polyhierarchies

Sometimes a concept can have two or more broader concepts.

- Polyhierarchy is permitted if the hierarchical relationship is valid in both/all cases
- Remember “All-and-Some” test for each generic hierarchical relationship
- Systems may or may not support it.



# Structural Design: Facets

## Facets

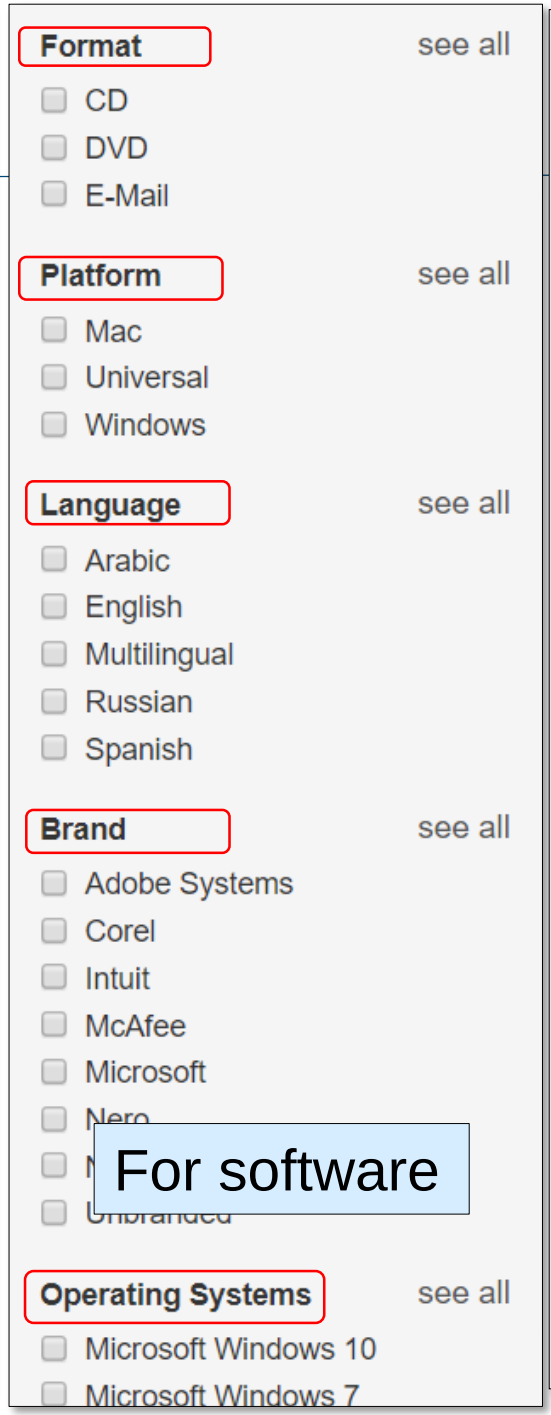
- For serving faceted classification, which allows the assignment of multiple classifications to an object
- A “dimension” of a query; a type of concept; an attribute of a thing; an aspect
- Intended for searching with multiple concepts in combination (post-coordination), one from each facet
- A refinement, filter, limit by, narrow by
- Can be for topics or for named entities
- Reflect the domain of content

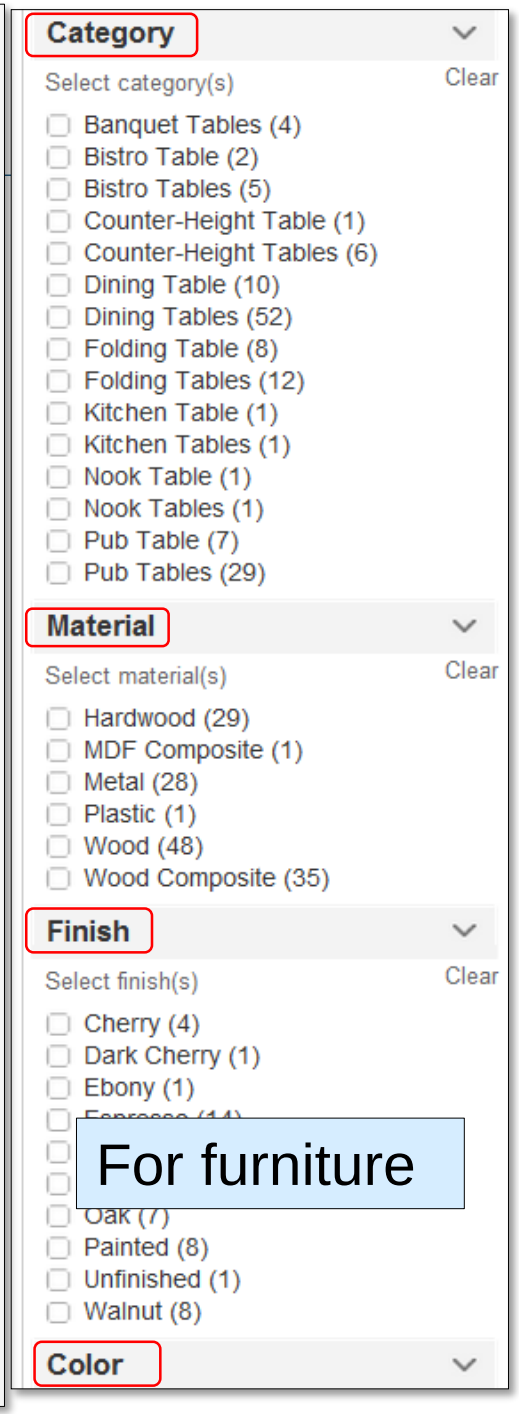
- Facets are dynamic and involve user interaction.
  Example: http://vocabulary.semantic-web.at/GraphSearch/

# Structural Design: Facets

Examples of ecommerce facets for different kinds of products

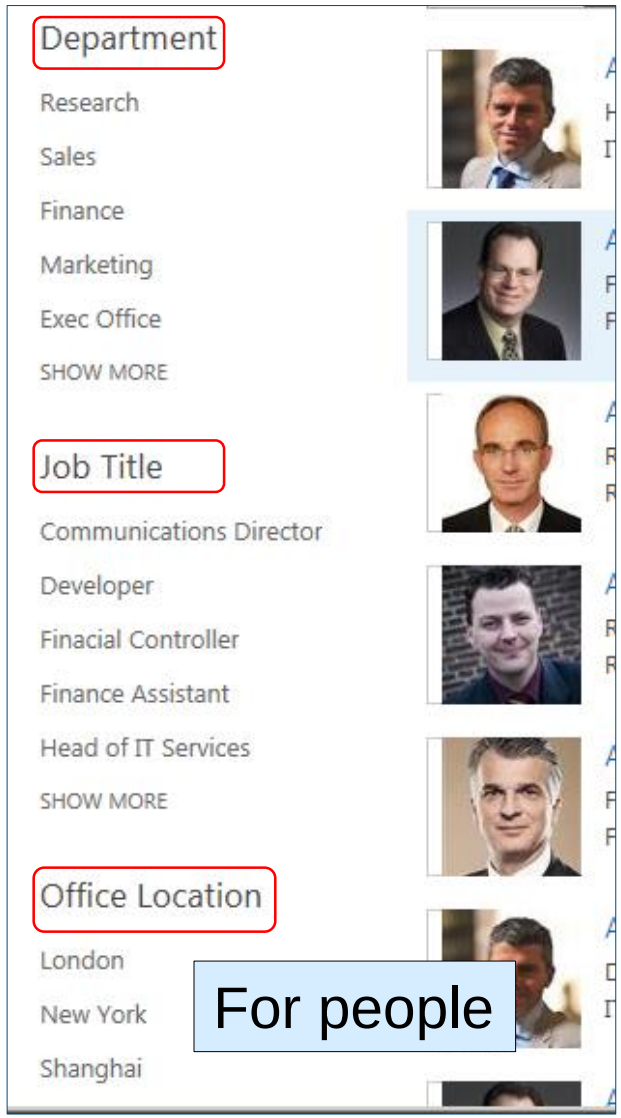




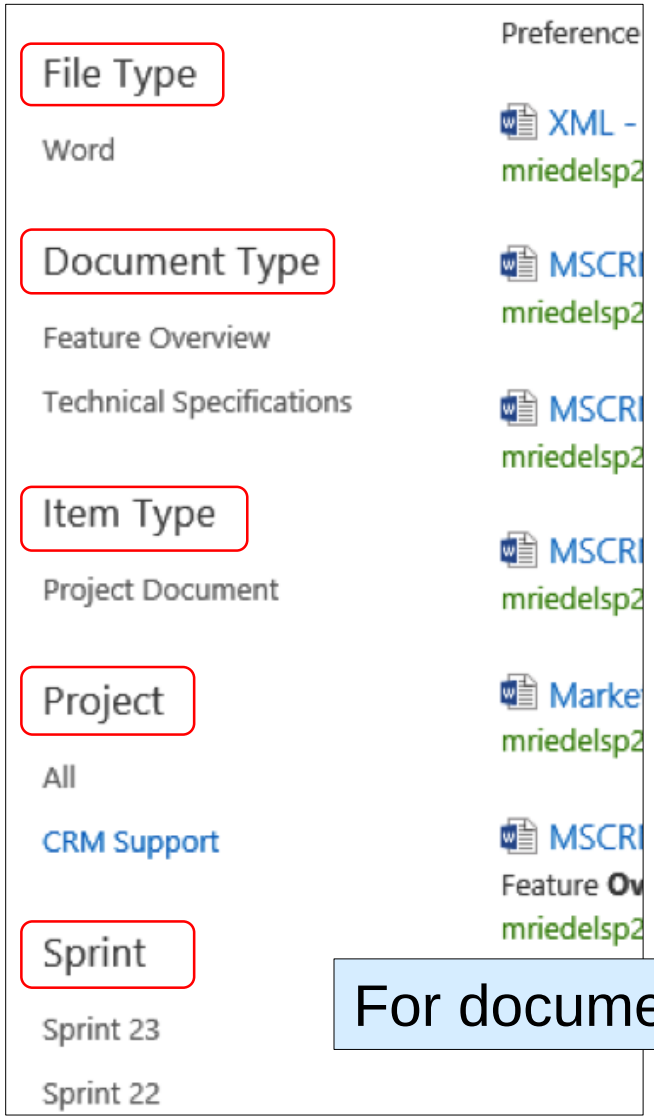



# Structural Design: Facets

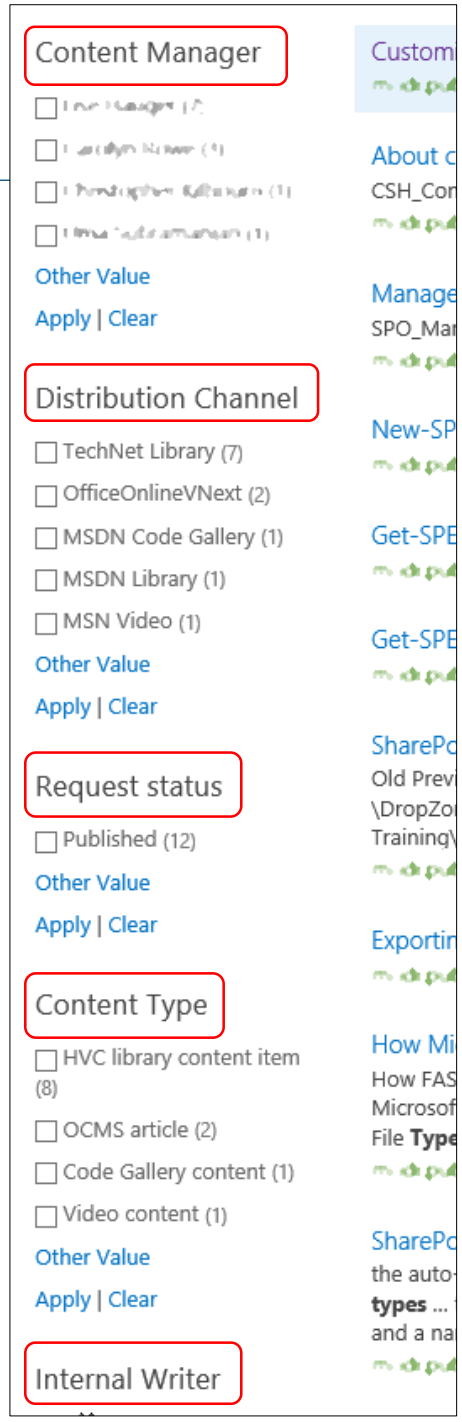
## Examples of internal content facets




For people




For documents

# Structural Design: Facets

## Facet advantages

- Supports more complex search queries by users
- Allows users to control the search refinement, narrowing or broadening in any manner or order
- Familiar to novice users; suitable for expert users

## Facet disadvantages

- Only suitable for somewhat structured, unified type of content that all share the same multiple facets
- Not practical for extremely large topical taxonomies
- Requires investment of thorough indexing/tagging

# Structural Design

Demonstration of the comparison of hierarchies and facets in PoolParty

# Outline

1. Introduction to taxonomies and ontologies
2. Purposes and benefits
3. Types of knowledge organization systems
4. Creating concepts
5. Creating relationships
6. Taxonomy structural design: hierarchies and facets
7. Standards: SKOS, RDF, RDF Schema, and OWL
8. Creating ontologies
9. Implementation issues
10. Linked data and the Semantic Web

# Standards: SKOS

## SKOS (Simple Knowledge Organization System)

- A data model to representation knowledge organization systems
- A World Wide Web (W3C) recommendation.
- Released in 2005 as a working draft and in 2009 as a recommendation.
- “A common data model for sharing and linking knowledge organization systems via the Web” https://www.w3.org/TR/skos-reference/
- Encoded using XML and RDF (Resource Description Framework).
- To enable easy publication and use of such vocabularies as linked data.
- A KOS built on SKOS is machine-readable and interchangeable.

# Standards: SKOS

## SKOS principles

- A KOS is a group of Concepts identified with URIs and grouped into a Concept scheme.
- Concepts can be labeled with any number of lexical strings (labels) in any natural language, such as prefLabel and altLabel.
- Concepts can be documented with notes of various types: scope notes, definitions, editorial notes, etc.
- Concepts can be linked to each other using hierarchical and associative semantic relations.
- Concepts can be grouped into Collections, which can be labeled and/or ordered.
- Concepts of different concept schemes can be mapped using four basic types of mapping links.

# Standards: SKOS

SKOS Elements

| Concepts | Labels & Notation | Documentation | Semantic Relations | Collections | Mapping Properties |
|---|---|---|---|---|---|
| Concept | prefLabel | note | broader | Collection | broadMatch |
| ConceptScheme | altLabel | changeNote | narrower | orderedCollection | narrowMatch |
| inScheme | hiddenLabel | definition | related | member | relatedMatch |
| hasTopConcept | notation | editorialNote | broaderTransitive | memberList | closeMatch |
| topConceptOf | | example | narrowerTransitive | | exactMatch |
| | | historyNote | semanticRelation | | mappingRelation |
| | | scopeNote | | | |



Example URI: skos:prefLabel

# Standards: RDF

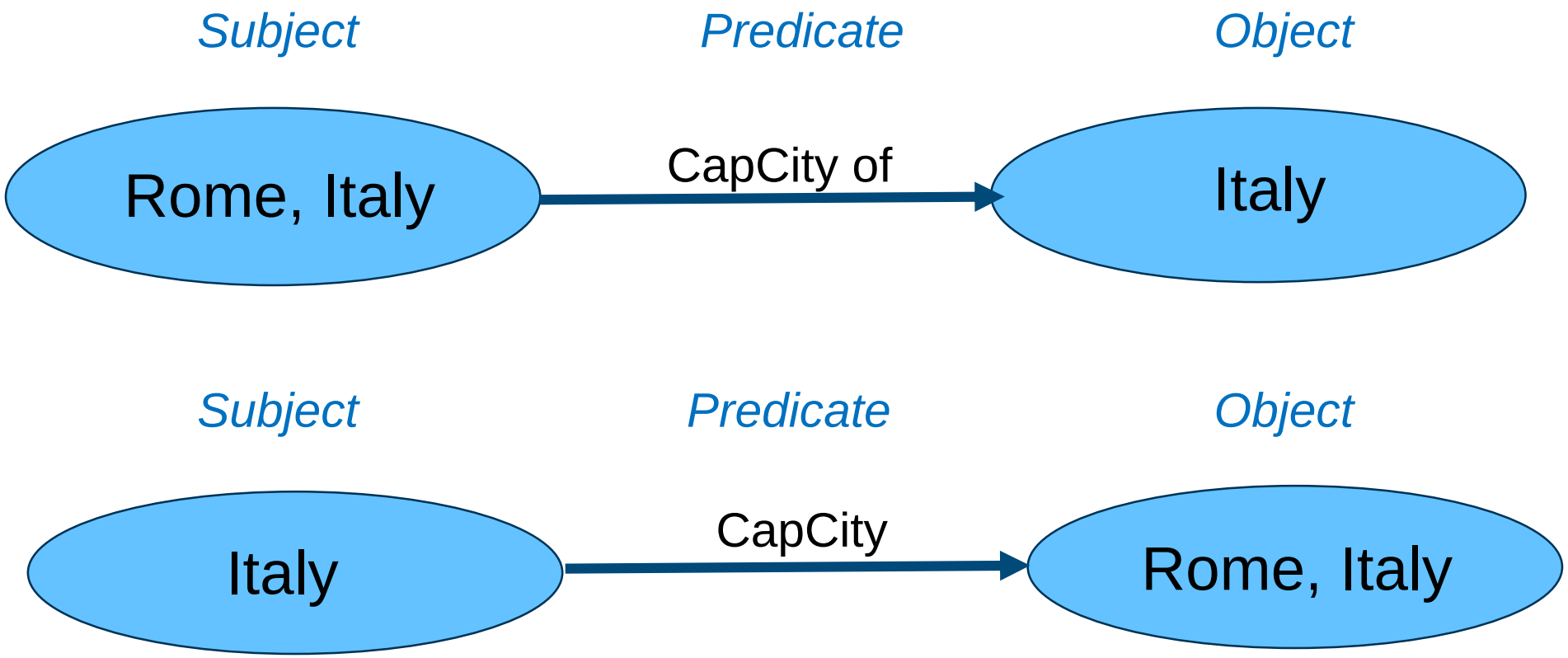
## RDF (Resource Description Framework)

- 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”
- Facilitates data merging even if the underlying schemas differ.
- Requires the use of URIs (Uniform Resource Identifiers) to specify **things** and to specify **relationships.**
- Models information as **subject – predicate – object** triples.
- Models information on a graph-based model.
- More fundamental, basic, and generic than SKOS or OWL.

# Standards: RDF

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

### Example



# Standards: RDF

## RDF is an abstract framework.

As a standard format for exchange/interoperability of data, there are various *serialization formats*:

- RDF/XML – XML-based syntax, the first standard format for serializing RDF
- Turtle – compact, human-friendly format
- N-Triples – very simple, easy-to-parse, line-based format, not as compact as Turtle
- N-Quads – superset of N-Triples, for serializing multiple RDF graphs
- JSON-LD – JSON-based serialization
- RDF/JSON – alternative syntax for expressing RDF triples using a simple JSON notation
- N3 (Notation3) – non-standard serialization similar to Turtle, but has additional features

# Standards: RDF Schema

## 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**
- A vocabulary for describing properties and classes of RDF resources.

# Standards: RDF Schema

## RDF Schema (RDFS) define classes and properties

Class:

- A type or category of resources or things.
- RDFS also describes subclasses and instances.

Property:

- Used to describe characteristics of things.
- Properties are also resources, so can be subjects of RDF triples.

Classes and properties are features of ontologies.

RDFS serves as a standard for ontologies.

# Standards: OWL

## 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.

# Standards: OWL

## OWL basic components

- **Classes** – *subjects or objects (domains and ranges) of RDF triples*
  - May contain individuals (instances of the class) and other subclasses
  - Sets of concepts that share characteristics and relationships
  - In SKOS: Concept schemes, top concept in a scheme, or concepts with narrower concepts
- **Individuals** – *subjects or objects (domains and ranges) of RDF triples*
  - Members or instances of a class.
  - In SKOS: Concepts
- **Properties** – *predicates of RDF triples*
  - Relations between instances or classes (2-way)
  - Attributes of instances or of classes (1-way)
  - In SKOS: Relationships *or* Attributes

# Standards: OWL

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

# Outline

1. Introduction taxonomies and ontologies
2. Purposes and benefits
3. Types of knowledge organization systems
4. Creating concepts
5. Creating relationships
6. Taxonomy structural design: hierarchies and facets
7. Standards: SKOS, RDF, RDF Schema, and OWL
8. Creating ontologies
9. Implementation issues
10. Linked data and the Semantic Web

# Creating Ontologies

## Creating a domain ontology, as a KOS

- Use a tool (such as PoolParty) that builds ontologies in SKOS, RDFS and OWL, so it's interoperable with other SKOS vocabularies and the Semantic Web.
- Consider starting with a core (upper) ontology as a model.
- Knowledge modeling is the initial task:
  - Define the scope
  - Identify the various classes (e.g. people, places, organizations, products)
  - Identify the relationships between classes
  - Identify the attributes for classes
- Create specific instances within the classes and apply the relationships
  - As combining an ontology with the specifics of a taxonomy
  - The taxonomy can already exist and be made more expressive, or be created along with the ontology as an integrated project.

# Creating Ontologies

## Knowledge modeling for a domain ontology example

## Language services business: match contractors to projects

Identify classes (groupings):

- Contractor
- Service type
- Language
    - etc.

# Creating Ontologies

## Demonstration of creating an ontology in PoolParty

# Outline

1. Introduction taxonomies and ontologies
2. Purposes and benefits
3. Types of knowledge organization systems
4. Creating concepts
5. Creating relationships
6. Taxonomy structural design: hierarchies and facets
7. Standards: RDF, SKOS, RDF Schema, and OWL
8. Creating ontologies
9. Implementation issues
10. Linked data and the Semantic Web

# Implementation Issues

## Stumbling block to taxonomy and ontology implementation

- Lack of user-focused or use-case-focused design; failure to test
- Lack of maintenance and governance
- Lack of support and for manual tagging
- Inappropriate integration with end-user search

# Implementation Issues: Testing

## Taxonomy/ontology testing overview

- Taxonomies serve a purpose, and that purpose should be tested.
- All taxonomies, regardless of who created them, should be tested.
- Testing can be simple or complex, depending on time and budget.
- Testing involves participants, as sample or representative users.
- Different types of tests are appropriate for different stages of taxonomy development.
- An inappropriate test or inappropriately timed text can be a waste of time and money.

# Implementation Issues: Maintenance & Governance

A taxonomy/ontology is never finished; it needs to be maintained and updated.

- New content, bringing up new concepts
- Content that gets dropped
- New requirements, users, needs, trends, markets, etc.
- New concepts or changes in terminology
- User feedback suggesting improvements

# Implementation Issues: Maintenance & Governance

## Taxonomy/ontology governance comprises:

- Maintenance (updating): responsibility, roles, processes, procedures
- KOS descriptive documentation (purpose, type, scope, users, indexing method, history/sources)
- KOS editorial policy/guidelines for maintenance
- Indexing or tagging policy/guidelines
- Instructional/how-to documents (system-specific)

➢ Governance process starts with the start of creating the taxonomy/ontology. As issues come and get resolved, they get documented as policy.

➢ Taxonomy governance may be part of a larger metadata specification.

# Implementation Issues: Manual Tagging

## A KOS is only useful if correctly and comprehensively tagged to content.

Choice of auto-categorization or manual tagging depends on volume of content and content management workflows

- Auto-categorization software (or add-ons to KOS management software) provides a good solution for tagging.
- Good software for manual tagging does not exist. It's just a feature of some other software or custom-programmed.
- Manual tagging interfaces may lack usability features
  - Ease of and speed of use
  - Both hierarchical and alphabetical (with alternative labels) lookups
- Manual tagging interfaces should be customizable to support indexing policy rules or required fields, cardinality, etc.

# Implementation Issues: Integration with Search

## How the KOS is utilized in search impacts KOS design

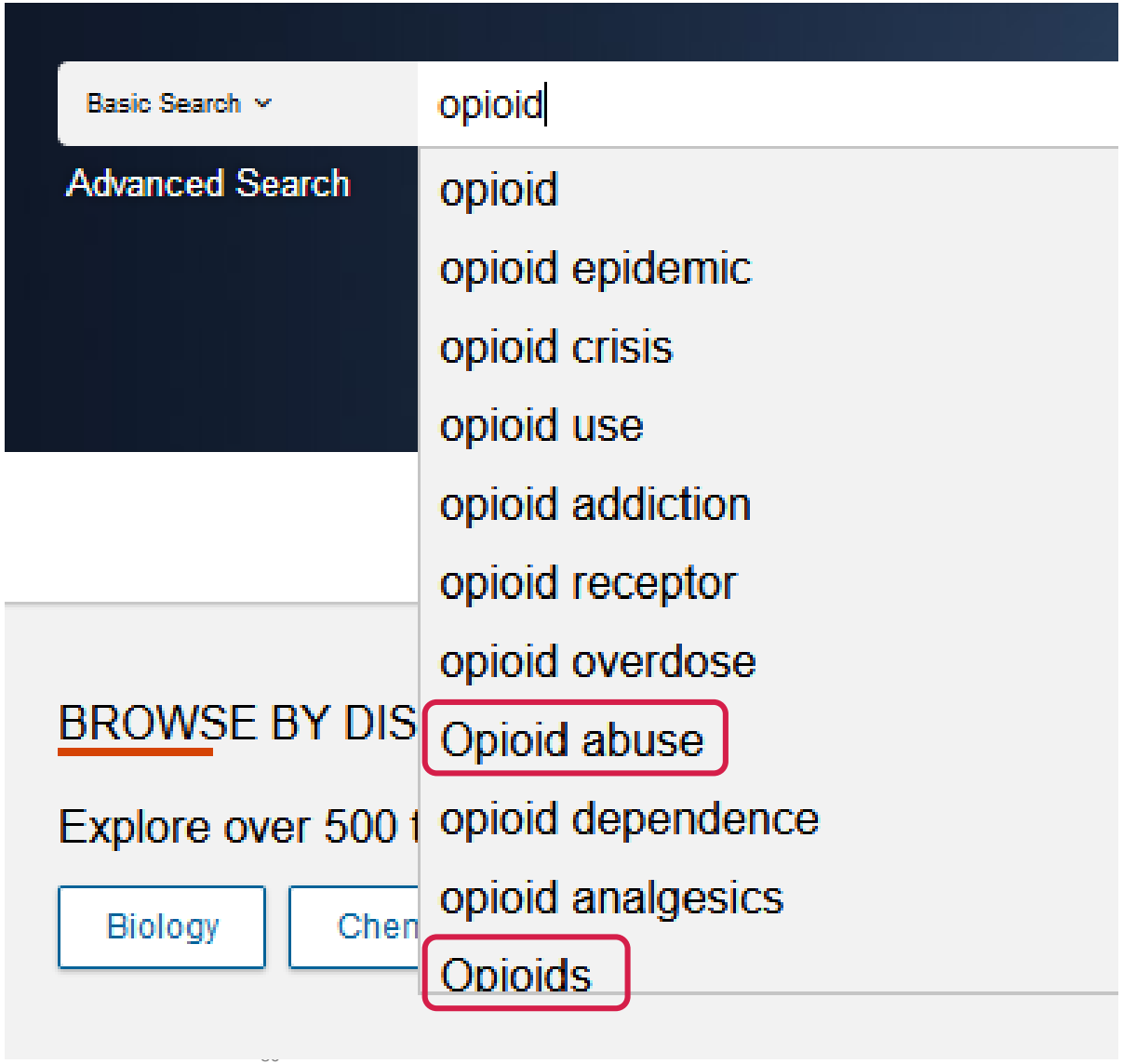
## Problems

- A KOS that is not displayed to end-users in any way
  (type-ahead display based on popular search keywords not the taxonomy)
- Lack of utilization of alternative labels in search
- Faceted taxonomy design without separate dynamic facets in the user interface
- A default keyword search and use of concepts in post-search filters

➢ Use of taxonomies in search that is desired and expected,
but perhaps not supported in 3rd-party systems

# Implementation Issues: Integration with Search

Type-ahead search display based on a combination of popular search keywords and controlled KOS concepts (with initial upper case)



# Outline

1. Introduction taxonomies and ontologies
2. Purposes and benefits
3. Types of knowledge organization systems
4. Creating concepts
5. Creating relationships
6. Taxonomy structural design: hierarchies and facets
7. Standards: RDF, SKOS, RDF Schema, and OWL
8. Creating ontologies
9. Implementation issues
10. Linked data and the Semantic Web

# Linked Data and the Semantic Web

## A KOS can take advantage of linked data and Semantic Web technologies.

### Linked Data

- Structured data which is interlinked with other data so that become more useful through semantic queries
- Collection of interrelated datasets on the Web, available in a standard format, reachable and manageable by Semantic Web tools
- Web sources based on the RDF scheme

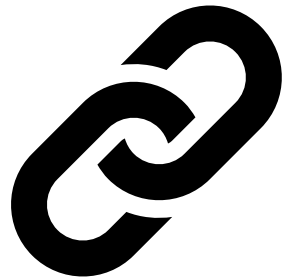

### The Semantic Web

- Large scale integration of, and reasoning on, data on the Web
- W3C's vision of the Web of linked data
- A technology stack to support a "Web of data," the sort of data you find in databases
- A common framework that allows data to be shared and reused across application, enterprise, and community boundaries

# Linked Data and the Semantic Web

## Linked Data Principles (on the web or in the enterprise)

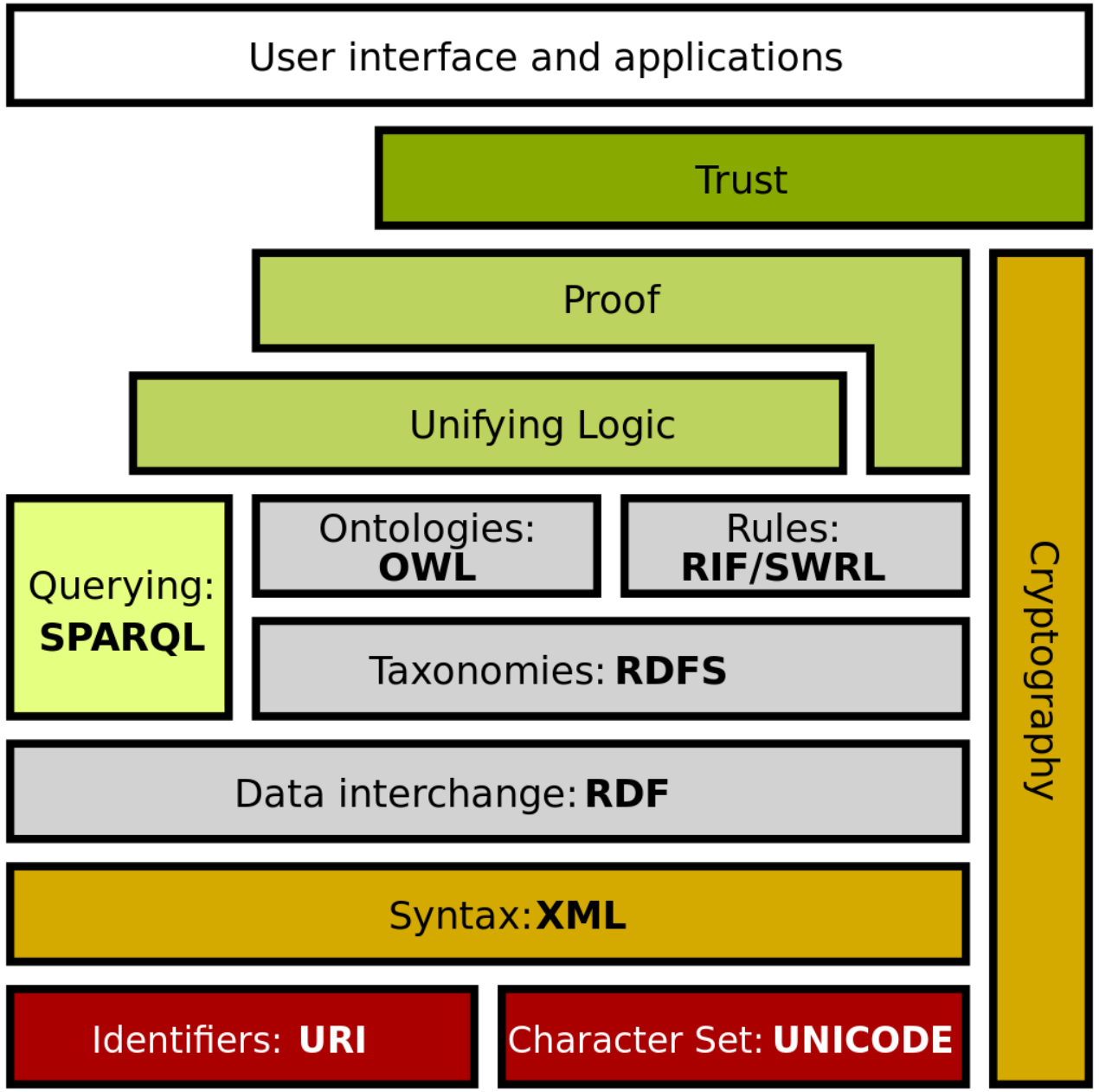
- Things are named with URIs - to identify and reference resources unambiguously.
- URIs are dereferenceable  - looking up a URI on the Web in order to get information about the referenced resource.
- RDF is used to represent information.
- Links to other things are included.



# Linked Data and the Semantic Web

Semantic Web Stack or
Semantic Web Layer Cake

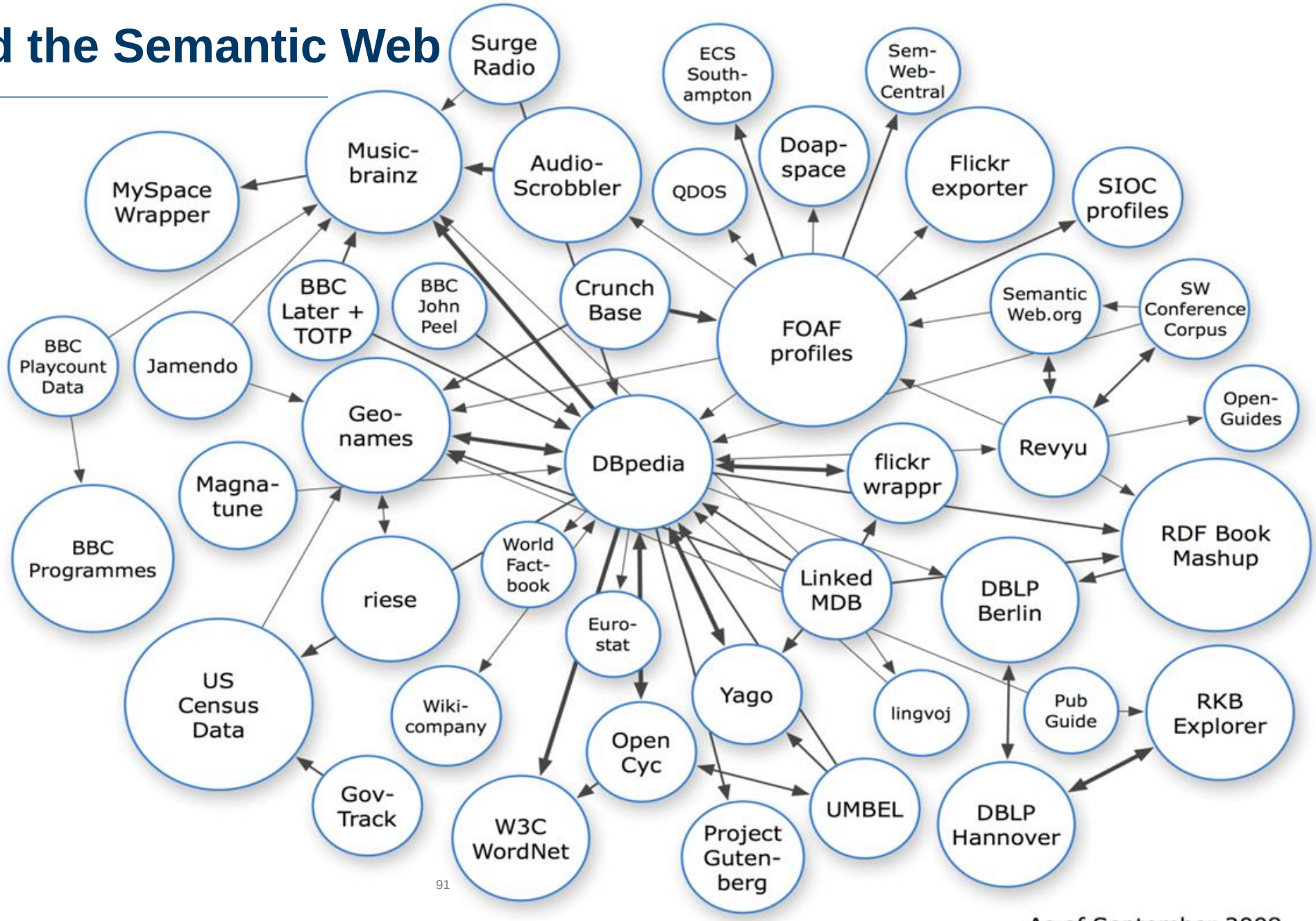
Illustration originally created by
Tim Berners-Lee, since revised.
https://en.wikipedia.org/wiki/Semantic_Web_Stack



# Linked Data and the Semantic Web

Popular Linked Data Datasets,

Part of the Semantic Web



# Linked Data and the Semantic Web

## Taxonomies and ontologies relate to linked data and the Semantic Web

### Have links going out

- Link out to add metadata to a concept (definitions, images, etc.).
- Link to equivalent concepts in linked vocabularies to obtain alternative labels.
- Link to equivalent concepts in linked open vocabularies to expand the set of linked content per concept.

### Have URLs for others to access your KOS

- Publish a taxonomy or ontology available for external reuse (with or without tagged content).
- Share the taxonomy or ontology and linked content with restricted access to external partners.

Utilize a taxonomy or ontology on the web on which to base yours.

# Linked Data and the Semantic Web

A KOS can take advantage of linked data and Semantic Web technologies.

Part of the Semantic Web set of technologies is a query language:

SPARQL

- **S**PARQL **P**rotocol **A**nd **R**DF **Q**uery **L**anguage
- The query language of the Semantic Web and knowledge graphs, or any data that follows the RDF specification, where data is stored as RDF triples
- Became a standard in of the W3C in 2008
- Allows for a query to consist of triple patterns, conjunctions, disjunctions, and optional patterns.
- Query types are: SELECT, ASK, CONSTRUCT, DESCRIBE

# Questions/Contact


## Heather Hedden

Taxonomy Consultant
Hedden Information Management
Carlisle, MA USA
+1 978-467-5195
www.hedden-information.com
accidental-taxonomist.blogspot.com
www.linkedin.com/in/hedden
Twitter: @hhedden