SPARQL
Field Guide for Taxonomists

by

Heather Hedden and Lisa Dawn Colvin

Taxonomy Boot Camp
Washington, DC

November 16, 2026



About the Speakers

Heather Hedden

- Independent taxonomy consultant, Hedden Information Management
- Instructor of online and corporate taxonomy courses and workshops
- Previous taxonomy roles with Enterprise Knowledge, Semantic Web Company, Gale/Cengage, and others.
- Author of *The Accidental Taxonomist, 3rd ed.* (2022, Information Today, Inc.)



Lisa Dawn Colvin

- Sr. Ontologist, Genpact
- Women in AI Ethics Fellow
- 20+ years across foundational AI research (CYC, with Doug Lenat), federal semantic infrastructure (NASA Constellation, Library of Congress UDFR), and Fortune 100 enterprise ontologies and recommendation systems (Capital One, Pandora).

Outline

- Introduction: Why and What is SPARQL
- Taxonomy Maintenance Scenarios
- Taxonomy Management Software Capabilities
- SKOS Elements and Triples
- Field Guide: SPARQL Basics
- Example SPARQL Queries for Taxonomy Maintenance Scenarios
- LLM-Generation of SPARQL

Introduction: Why SPARQL for Taxonomies

Taxonomies Need Updating and Improving

- Involves tracking, identifying, completing and/or fixing issues and properties of concepts.
  - Includes labels, relationships, notes, possibly ontology connections

Objectives:

- Identification of issues with highly customizable reports
  - Extent of customization of reports varies by taxonomy software vendors.
- Fixing of issues more efficiently with batch editing
  - Usually cannot be done in taxonomy management software UI

SPARQL can be used for both custom reports and making edits.

Introduction

SPARQL

- **S**PARQL **P**rotocol and **R**DF **Q**uery **L**anguage
- For querying and editing RDF or RDF-*based* data,
  including data in RDFS, OWL, and SKOS (ontologies and taxonomies)



RDF (Resource Description Framework)

- "A standard model for data interchange on the Web" published by the W3C
- Information is expressed in statements of ***triples***: subject – predicate – object
- Requires the use of ***URIs*** for all resources within triples
- In various machine-readable forms: Turtle, JSON-LD, N-Triples, RDF/XML

SKOS (Simple Knowledge Organization System)

- "A common data model for sharing and linking knowledge organization systems via the Web," especially taxonomies
- Is based on RDF, so encodes data as triples with URIs
- Has become the standard data model for taxonomies and thesauri
- Supported by all dedicated taxonomy management systems



Taxonomy Maintenance - Report Scenarios

Getting *counts* of the number of concepts with/without certain properties

To get an understanding of the state of the taxonomy and plan for work

How many concepts:

- Have or lack definitions
- Are in polyhierarchies (having more than one broacer concept)
- Have or lack preferred labels in another designated language
- Have alternative labels and/or hidden labels
- Have or lack certain note types (Editorial, Change, Scope notes)
- Are top concepts
- And/or belong to a certain concept scheme, collection, or class

Taxonomy Maintenance - Report Scenarios

Identifying concepts that have certain properties/features

To check to see if they need editing, so that they conform to best practices, the taxonomy's policy, and consistency.

Concepts that have the following and list them:

- Have definitions
- Are in polyhierarchies
- Have related-concept relationships
- Have both an alternative label *and* a hidden label
- Have both history notes *and* change notes
- Have scope notes
- Are top concepts

Additionally: Concepts that have any of the above properties/features *and* belong to a specific concept scheme, hierarchical branch, collection, or class.

Taxonomy Maintenance - Report Scenarios

Identifying concepts that are *missing* certain properties/features

To add them or evaluate if they need to be added

Concepts that are missing:

- Definitions
- Narrower concepts
- Alternative labels
- Preferred labels in a designated language
- A type of note (Editorial, Change, Scope notes)
- Membership in any collection
- Membership in any class

Additionally: Concepts missing these *and* belong to a specific concept scheme, hierarchical branch, collection, or class.

Taxonomy Maintenance - Batch Editing Scenarios

Making *changes* to multiple concepts or their properties

- More efficiently than through the software managemt software's UI
- For corrections, if the software's quality settings were not sufficiently strict.
- To adapt a taxonomy for import for the software's preferred layout, (when it's not a spreadsheet, but in RDF/XML, JSON, Turtle, etc.)

Examples

- Where the same label is duplicated as a prefLabel in one concept and an altLabel in another concept, delete the altLabel
- Change the URI pattern in a taxonomy
- Add a text string to all concept labels or notations in a subtree
- Add a linked relation to all concepts in a subtree
- Delete all values in a specific attribute field

Taxonomy Management Software Capabilities

Taxonomy management software cannot support all possible reports or edits.

Taxonomy management software varies in their capabilities:

- May offer pre-defined reports, but options might be limited.
- May provide counts of concepts, labels, relations, but not everything nor full customized combinations of criteria.
- May offer customized reports, but not support batch editing capabilities

Most taxonomy manage systems have a “SPARQL endpoint” feature to enter queries:

Graphwise, Synaptica, Progress Semaphore, TopBraid EDG, Graphologi, VocBench

Simpler, free SKOS editors have even fewer capabilities

All reporting, in addition to batch editing, must be done with SPARQL

No tool?

Use GenAI to convert your spreadsheet taxonomy into SKOS RDF.
Then query and fix the SKOS file with SPARQL queries.

SKOS Overview

SKOS Elements

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

Initial Upper case (ConceptScheme, Collection, and Concept) may be **subjects** or **objects** of triples.
Initial lower case (all the rest, e.g. definition, altLabel) are the **predicates** of triples.

Machine-readable examples: skos:prefLabel skos:broader

SKOS Overview

Triples in SKOS

Example triple for the *relationship* between two concepts:

Subject: https://server-name.com/HealthCare/23 (Health care regulations)

Predicate: http://www.w3.org/2004/02/skos/core#narrower

Object: https://server-name.com/HealthCare/25 (Affordable Care Act)

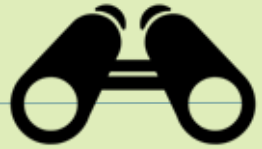
Example triple for a *label* of a concept:

Subject: https://server-name.com/HealthCare/23

Predicate: http://www.w3.org/2004/02/skos/core#prefLabel

Object: Health care regulations

Field Guide to SPARQL Basics: How Does SPARQL Work?



SPARQL specifies a question, with variables as placeholders for what you are asking

Questions	Query Patterns
What is a broader parent of Diabetes?	ex:diabetes skos:broader ?parent.
What are the children of Chronic Diseases?	?child skos:broader ex:chronicDiseases.
What does this taxonomy state about Diabetes?	ex:diabetes ?relationship ?value.

- You can name a variable what you wish.
- It will be the column heading for your results.
- Any use of it within a particular scope within a query must have the same referent.

SPARQL Basics - Recognizing Common Query Components

Component	Purpose	Examples
PREFIX	Sets a short name for long URI. One line per vocabulary	PREFIX skos: <http://www.w3.org/2004/02/skos/core#> PREFIX ex: <http://example.org/scheme/>
SELECT	Names the columns you want in results	SELECT ?concept ?prefLabel ?definition ?status
WHERE	Describes what to look for, the variables must match your data	WHERE {?concept a skos:Concept; skos:prefLabel ?prefLabel . }
OPTIONAL	Adds additional information if it exists, but you still get results if it doesn't.	OPTIONAL {?concept skos:definition ?definition . }
FILTER	Keeps only the matches which you want	FILTER (LANG(?PREFLABEL) = "en")
BIND	Creates a new column from calcuation	BIND (IF(BOUND(?definition), "With definition", "without definition") AS ?status)
ORDER BY	Sorts results	ORDER BY ?prefLabel

SPARQL Basics – Forming a Query

```
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>

SELECT ?concept ?prefLabel ?definition ?status
WHERE {
  ?concept a skos:Concept ;
           skos:prefLabel ?prefLabel .
OPTIONAL {
    ?concept skos:definition ?definition .
    FILTER (LANG(?definition) = "en")
  }
  FILTER (LANG(?prefLabel) = "en")
  BIND (IF(BOUND(?definition), "With definition", "Without definition") AS ?status)
}
 ORDER BY ?prefLabel
```

Query for getting concepts with their definitions

SPARQL Basics – Forming a Query

```
PREFIX skos: <http://www.w3.org/2004/02/skos/core#> #create my graph abbreviation

SELECT ?concept ?prefLabel ?definition ?status #pick what I want my query to return
WHERE {                                        # constrain context to only concepts with labels
  ?concept a skos:Concept ;
           skos:prefLabel ?prefLabel .
OPTIONAL {                                     # if a definition exists, show it, otherwise, blank
    ?concept skos:definition ?definition .
    FILTER (LANG(?definition) = "en")          # only if in English
  }
  FILTER (LANG(?prefLabel) = "en")             # ensure label is in English
  BIND (IF(BOUND(?definition), "With definition", "Without definition") AS ?status)
}                                              # create another column, showing def status
 ORDER BY ?prefLabel                           # order by label, alphabetically
```

SPARQL Basics – Forming a Query: Results from a SPARQL Endpoint

concept	prefLabel	definition	status
https://hedden-information.poolparty.biz/HealthCare/74	"3D printing"@en		"Without definition"
https://hedden-information.poolparty.biz/HealthCare/75	"AI in health care"@en	"The integration of artificial intelligence into health technology to enhance clinical decision-making, patient outcomes, and operational efficiency in medical settings."@en	"With definition"
https://hedden-information.poolparty.biz/HealthCare/94	"Adults"@en		"Without definition"
https://hedden-information.poolparty.biz/HealthCare/86	"Affordable Care Act"@en		"Without definition"
https://hedden-information.poolparty.biz/HealthCare/93	"Age groups"@en		"Without definition"
https://hedden-information.poolparty.biz/HealthCare/106	"Alternative medicine"@en		"Without definition"
https://hedden-information.poolparty.biz/HealthCare/48	"Anxiety disorders"@en		"Without definition"
https://hedden-	"Asthma"@en	"A condition in which a person's airways become inflamed, narrow, and	"With

SPARQL Examples - Get Counts of Concepts with Definitions

```
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX skos:<http://www.w3.org/2004/02/skos/core#>

SELECT (COUNT(DISTINCT ?c) AS ?withDefinition)
WHERE {
      ?c a skos:Concept .
FILTER EXISTS { ?c skos:definition ?d }
              }
```

withDefinition
"48"^^<http://www.w3.org/2001/XMLSchema#integer>

SPARQL Examples - Get Counts of Concepts Without Definitions

```
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX skos:<http://www.w3.org/2004/02/skos/core#>

SELECT (COUNT(DISTINCT ?c) AS ?withoutDefinition)
WHERE {
      ?c a skos:Concept .
FILTER NOT EXISTS { ?c skos:definition ?d }
                }
```

withoutDefinition
"67"^^<http://www.w3.org/2001/XMLSchema#integer>

SPARQL Examples - Get Counts of Concepts in a Concept Scheme

```
PREFIX skos:<http://www.w3.org/2004/02/skos/core#>
PREFIX health:<https://myhealth.io/HealthCare/>

  SELECT (COUNT(DISTINCT ?c) AS ?count)
   WHERE {
      ?c skos:inScheme health:Diseases
                  }
```

SPARQL Examples - Get Concepts That Are in Polyhierachies

```
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX skos:<http://www.w3.org/2004/02/skos/core#>


SELECT  ?label  (GROUP_CONCAT(DISTINCT ?btLabel; separator=" | ") AS
?broaderTerms)
WHERE {
  ?c a skos:Concept ; skos:broader ?bt .
  ?c skos:prefLabel ?label . FILTER(LANG(?label)="en")
  ?bt skos:prefLabel ?btLabel . FILTER(LANG(?btLabel)="en")
            }
GROUP BY ?c ?label
HAVING (COUNT(DISTINCT ?bt) > 1)
ORDER BY DESC(?broaderCount)
```

label	broaderTerms
"Type 1 diabetes"@en	"Autoimmune diseases | Diabetes"
"Coronary artery disease"@en	"Cardiovascular diseases | Chronic diseases"
"Asthma"@en	"Chronic diseases | Respiratory diseases"
"Chronic obstructive pulmonary disease"@en	"Chronic diseases | Respiratory diseases"
"Cystic fibrosis"@en	"Chronic diseases | Single-gene disorders"
"Long COVID"@en	"Chronic diseases | COVID-19"
"Sickle cell anemia"@en	"Chronic diseases | Single-gene disorders"

SPARQL Examples - Get Concepts that Have Both Alt and Hidden Labels

```
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
SELECT ?c ?label ?alt ?hid
 WHERE {
      ?c a skos:Concept ;
                skos:prefLabel ?label .
           FILTER (LANG(?label) = “en”)
      OPTIONAL {
           ?c skos:altLabel ?alt;
           FILTER (LANG(?alt) = “en”)
             }
            OPTIONAL {
           ?c skos:hiddenLabel ?hid ;
           FILTER (LANG(?hid) = “en”)
           }
}
ORDER BY ?label
```

As is: returns all concepts whether they have either alternative or hidden labels. Will return those that have neither.

If you want ONLY those that have both and alternative label and a hidden label, take out the OPTIONAL from those clauses.

Identify Concepts without Narrower Concepts (Are Leaf Nodes)

```
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX skos:<http://www.w3.org/2004/02/skos/core#>

SELECT ?label WHERE {
  ?c a skos:Concept ; skos:prefLabel ?label . FILTER(LANG(?label)="en")
  FILTER NOT EXISTS { ?c skos:narrower ?n }
  FILTER NOT EXISTS { ?child skos:broader ?c }
}
ORDER BY ?label
```

label
"3D printing"@en
"AI in health care"@en
"Adults"@en
"Affordable Care Act"@en
"Alternative medicine"@en
"Anxiety disorders"@en
"Asthma"@en
"Breast cancer"@en
"Cancer surgery"@en
"Candidiasis"@en

Identify Concepts That Are Missing Alternative Labels

```
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX skos:<http://www.w3.org/2004/02/skos/core#>

SELECT ?label WHERE {
  ?c a skos:Concept ; skos:prefLabel ?label . FILTER(LANG(?label)="en")
  FILTER NOT EXISTS { ?c skos:altLabel ?alt }
}
ORDER BY ?label
```

label
"3D printing"@en
"Adults"@en
"Age groups"@en
"Alternative medicine"@en
"Anxiety disorders"@en
"Asthma"@en
"Breast cancer"@en
"COVID-19"@en
"Cancer surgery"@en
"Cardiac surgery"@en

Identify Concepts Missing Preferred Labels in Spanish

```
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX skos:<http://www.w3.org/2004/02/skos/core#>


SELECT ?englishLabel WHERE {
  ?c a skos:Concept .
  OPTIONAL { ?c skos:prefLabel ?englishLabel .
FILTER(LANGMATCHES(LANG(?englishLabel),"en")) }
  FILTER NOT EXISTS { ?c skos:prefLabel ?l .
FILTER(LANGMATCHES(LANG(?l),"es")) }
}
```

englishLabel
"Seronegative rheumatoid arthritis"@en
"Seropositive rheumatoid arthritis"@en
"Health maintenance organizations"@en
"Public insurance"@en
"Medicaid"@en
"Medicare"@en
"3D printing"@en
"AI in health care"@en
"Electronic health records"@en
"Medical devices"@en
"Telemedicine"@en
"Affordable Care Act"@en
"Health Insurance Portability and Accounta

Show all Concepts with Their English and Spanish Labels

```
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX skos:<http://www.w3.org/2004/02/skos/core#>

SELECT ?c (STR(?enLabel) AS ?en) (STR(?esLabel) AS ?es)
WHERE {
  ?c a skos:Concept ;
     skos:prefLabel ?enLabel .
  FILTER(LANGMATCHES(LANG(?enLabel), "en"))
  OPTIONAL {
    ?c skos:prefLabel ?esLabel .
    FILTER(LANGMATCHES(LANG(?esLabel), "es"))
  }
}
ORDER BY ?en
```

c	en	es
https://hedden-information.poolparty.biz/HealthCare/74	"3D printing"	
https://hedden-information.poolparty.biz/HealthCare/75	"AI in health care"	
https://hedden-information.poolparty.biz/HealthCare/94	"Adults"	"Adultos"
https://hedden-information.poolparty.biz/HealthCare/86	"Affordable Care Act"	
https://hedden-information.poolparty.biz/HealthCare/93	"Age groups"	"Grupos de edad"
https://hedden-	"Alternative medicine"	"Medicina alternativ

Show all Concepts with Their English and Spanish Labels

c	en	es
https://hedden-information.poolparty.biz/HealthCare/74	"3D printing"	
https://hedden-information.poolparty.biz/HealthCare/75	"AI in health care"	
https://hedden-information.poolparty.biz/HealthCare/94	"Adults"	"Adultos"
https://hedden-information.poolparty.biz/HealthCare/86	"Affordable Care Act"	
https://hedden-information.poolparty.biz/HealthCare/93	"Age groups"	"Grupos de edad"
https://hedden-information.poolparty.biz/HealthCare/106	"Alternative medicine"	"Medicina alternativa"
https://hedden-information.poolparty.biz/HealthCare/48	"Anxiety disorders"	"Trastornos de ansiedad"
https://hedden-information.poolparty.biz/HealthCare/18	"Asthma"	"Asma"
https://hedden-information.poolparty.biz/HealthCare/2	"Autoimmune diseases"	"Enfermedades autoinmunes"
https://hedden-information.poolparty.biz/HealthCare/33	"Bacterial infections"	"Infecciones bacterianas"
https://hedden-information.poolparty.biz/HealthCare/9	"Breast cancer"	"Cáncer de mama"
https://hedden-information.poolparty.biz/HealthCare/44	"COVID-19"	"COVID-19"
https://hedden-information.poolparty.biz/HealthCare/8	"Cancer"	"Cáncer"
https://hedden-	"Cancer surgery"	"Cirugía oncológica"

Batch Maintenance: Find concepts with duplicated pref & altLabels

```
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>

SELECT ?altOwner ?altOwnerPrefLabel ?altLabel ?prefOwner ?prefLabel
WHERE {
  ?altOwner a skos:Concept ;
            skos:prefLabel ?altOwnerPrefLabel;
           skos:altLabel ?altLabel.
FILTER(LANGMATCHES(LANG(?altOwnerPrefLabel), "en"))
  ?prefOwner a skos:Concept ;
            skos:prefLabel ?prefLabel .
  FILTER(?altOwner != ?prefOwner)
  FILTER(LANG(?altLabel) = LANG(?prefLabel))
  FILTER(LCASE(STR(?altLabel)) = LCASE(STR(?prefLabel)))
}
ORDER BY ?altLabel
```

altOwner	altOwnerPrefLabel	altLabel	prefOwner	prefLabel
https://hedden-information.poolparty.biz/HealthCare/63	"Health care services"@en	"Health maintenance organizations"@en	https://hedden-information.poolparty.biz/HealthCare/69	"Health maintenance organizations"@en

Batch Maintenance: Delete altLabels that Duplicate prefLabels

```
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>

DELETE ?altOwner ?altOwnerPrefLabel ?altLabel ?prefOwner ?prefLabel
WHERE {
  ?altOwner a skos:Concept ;
            skos:prefLabel ?altOwnerPrefLabel;
           skos:altLabel ?altLabel.
FILTER(LANGMATCHES(LANG(?altOwnerPrefLabel), "en"))
  ?prefOwner a skos:Concept ;
            skos:prefLabel ?prefLabel .
  FILTER(?altOwner != ?prefOwner)
  FILTER(LANG(?altLabel) = LANG(?prefLabel))
  FILTER(LCASE(STR(?altLabel)) = LCASE(STR(?prefLabel)))
}
ORDER BY ?altLabel
```

t's common in graph database systems to have separate query and update endpoints

Comparison of Using LLMs/GenAI for SPARQL

```
PREFIX skos:<http://www.w3.org/2004/02/skos/core#>
SELECT ?conceptLabel ?relatedLabel
WHERE {
  ?concept skos:related ?related .
  FILTER (STR(?concept) < STR(?related))
  OPTIONAL {
    ?concept skos:prefLabel ?conceptLabel .
    FILTER (LANG(?conceptLabel) = "en")
  }
  OPTIONAL {
    ?related skos:prefLabel ?relatedLabel .
    FILTER (LANG(?relatedLabel) = "en")
  }
}
ORDER BY ?conceptLabel ?relatedLabel
```

Get concepts that have related relationships

Comparison of Using LLMs/GenAI for SPARQL: ChatGPT

```
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
SELECT DISTINCT
    ?concept
    ?conceptLabel
    ?relatedConcept
    ?relatedConceptLabel
WHERE {
    ?concept a skos:Concept ;
            skos:related ?relatedConcept .
    OPTIONAL {
        ?concept skos:prefLabel ?conceptLabel .
        FILTER (LANG(?conceptLabel) = "en")
    }
    OPTIONAL {
        ?relatedConcept skos:prefLabel ?relatedConceptLabel .
        FILTER (LANG(?relatedConceptLabel) = "en")
    }
}
ORDER BY ?conceptLabel ?relatedConceptLabel
```

Create a SPARQL query to list all concepts in a SKOS taxonomy that have Related Concept relationships.



Comparison of Using LLMs/GenAI for SPARQL: Claude

```
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>

SELECT ?concept ?prefLabel
       (COUNT(DISTINCT ?related) AS ?relatedCount)
       (GROUP_CONCAT(DISTINCT ?relatedLabel; separator=" | ") AS ?relatedConcepts)
WHERE {
  ?concept a skos:Concept ;
           skos:related ?related .
  OPTIONAL { ?concept skos:prefLabel ?prefLabel
             FILTER(LANG(?prefLabel) = "en") }
  OPTIONAL { ?related skos:prefLabel ?relatedLabel
             FILTER(LANG(?relatedLabel) = "en") }
}
GROUP BY ?concept ?prefLabel
ORDER BY DESC(?relatedCount) ?prefLabel
```

Create a SPARQL query to list all concepts in a SKOS taxonomy that have Related Concept relationships.



Comparison of Using LLMs/GenAI for SPARQL: Claude

conceptLabel	relatedLabel
"Cancer surgery"@en	"Cancer"@en
"Cardiac surgery"@en	"Cardiovascular diseases"@en
"Chemotherapy"@en	"Cancer"@en
"Patient privacy"@en	"Health Insurance Portability and Accountability Act"@en

Manual query created by ontologist, Lisa

concept	conceptLabel	relatedConcept	relatedConceptLabel
https://hedden-information.poolparty.biz/HealthCare/8	"Cancer"@en	https://hedden-information.poolparty.biz/HealthCare/112	"Cancer surgery"@en
https://hedden-information.poolparty.biz/HealthCare/8	"Cancer"@en	https://hedden-information.poolparty.biz/HealthCare/116	"Chemotherapy"@en
https://hedden-information.poolparty.biz/HealthCare/112	"Cancer surgery"@en	https://hedden-information.poolparty.biz/HealthCare/8	"Cancer"@en
https://hedden-information.poolparty.biz/HealthCare/113	"Cardiac surgery"@en	https://hedden-information.poolparty.biz/HealthCare/13	"Cardiovascular diseases"@en
https://hedden-information.poolparty.biz/HealthCare/13	"Cardiovascular diseases"@en	https://hedden-information.poolparty.biz/HealthCare/113	"Cardiac surgery"@en
https://hedden-information.poolparty.biz/HealthCare/116	"Chemotherapy"@en	https://hedden-information.poolparty.biz/HealthCare/8	"Cancer"@en
https://hedden-information.poolparty.biz/HealthCare/87	"Health Insurance Portability and Accountability Act"@en	https://hedden-information.poolparty.biz/HealthCare/84	"Patient privacy"@en
https://hedden-information.poolparty.biz/HealthCare/84	"Patient privacy"@en	https://hedden-information.poolparty.biz/HealthCare/87	"Health Insurance Portability and Accountability Act"@en



concept	label
https://hedden-information.poolparty.biz/HealthCare/120	"Cancer"@en
https://hedden-information.poolparty.biz/HealthCare/63	"Cancer surgery"@en
https://hedden-information.poolparty.biz/HealthCare/64	"Cardiac surgery"@en
https://hedden-information.poolparty.biz/116HealthCare/116	"Chemotherapy"@en
https://hedden-information.poolparty.biz/HealthCare/97	"Coronary artery disease"@en
https://hedden-information.poolparty.biz/HealthCare/119	"Health Insurance Portability and Accountability Act"@en
https://hedden-information.poolparty.biz/HealthCare/16	"Patient privacy"@en



Resources on SPARQL

"Why a Taxonomist Should Know SPARQL" https://enterprise-knowledge.com/why-a-taxonomist-should-know-sparql/

"An Introduction to Querying Taxonomies Using SPARQL" (Graphologi) https://graphologi.com/basic-sparql-taxonomies

"What is SPARQL" (Graphwise) https://graphwise.ai/fundamentals/what-is-sparql/

"The Comprehensive Guide to SPARQL" tutorial https://sparql.dev/

SPARQL 1.1 spec https://www.w3.org/TR/sparql11-overview/

Learning SPARQL: Querying and Updating with SPARQL 1.1 by Bob DuCharme https://www.learningsparql.com

Questions/Contact

Heather Hedden
Taxonomy Consultant
Hedden Information Management
Carlisle, MA
heather@hedden.net
www.hedden-information.com
www.linkedin.com/in/hedden
accidental-taxonomist.blogspot.com

Lisa Dawn Colvin
Senior Ontologist, genpact
AI Literacy, Women in AI Ethics
Oakland, CA
lisacolvin@gmail.com
www.linkedin.com/in/lisacolvin
thedawnmachine.substack