

JSON-LD for Linked Open Data and Knowledge Graphs

Dr. Jans Aasman
(allegrograph.com)



Contents

- The world of Semantics and RDF and Graphs
- The world of JSON and Documents
- A powerful intersection: JSON-LD: JSON + RDF
- Store it in a document store or in a semantic graph database?

The world of Semantics and RDF

- Tim Berners-Lee in 1994: only humans can read the context on the web, computers can't. We need a meta data language that gives meaning to objects on the web.
- Invented RDF to add Semantics to objects.



RDF at the core is simple

[1] Use IRIs as a universal identifier mechanism

[2] Describe any type of (meta) data as triples

<subject> <predicate> <object>

subjects* and predicates are always IRIs

objects can be IRIs or any XSD datatype

[3] Add meaning to data by a self-describing logic called OWL

* ok, ok, subjects can also be blank nodes

NTRIPLES format for computers

- There is a person born in 1958 with first name “Jans” and last name “Aasman” who lives in a place with the name “Moraga” (pop 16000) that is part of state “California (pop 34 M)

```
<http://abc.com/person1> <http://abc.com/firstName> "Jans" .
<http://abc.com/person1> <http://abc.com/lastName> "Aasman" .
<http://abc.com/person1> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://abc.com/Person> .
<http://abc.com/person1> <http://abc.com/birthYear> "1958" .
<http://abc.com/person1> <http://abc.com/livesIn> <http://abc.com/place1> .
<http://abc.com/place1> <http://abc.com/name> "Moraga" .
<http://abc.com/place1> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://abc.com/Place> .
<http://abc.com/place1> <http://abc.com/population> "16000"^^<http://www.w3.org/2001/XMLSchema#integer> .
<http://abc.com/place1> <http://abc.com/partOf> <http://abc.com/statel> .
<http://abc.com/statel> <http://abc.com/name> "California" .
<http://abc.com/place1> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://abc.com/Place> .
<http://abc.com/statel> <http://abc.com/population> "34000000"^^<http://www.w3.org/2001/XMLSchema#integer> .
```

TURTLE for humans

```
# Turtle, same information...

@prefix : <http://abc.com/> .
:person1 a :Person ;
         :firstName 'Jans' ;
         :lastName 'Aasman' ;
         :birthYear 1958 ;
         :livesIn :place1 .
:place1  a :Place
         :name 'Moraga' ;
         :population 16000 ;
         :partOf :state1.
:state1  a :State;
         :name 'California' ;
         :population 34000000 .
```

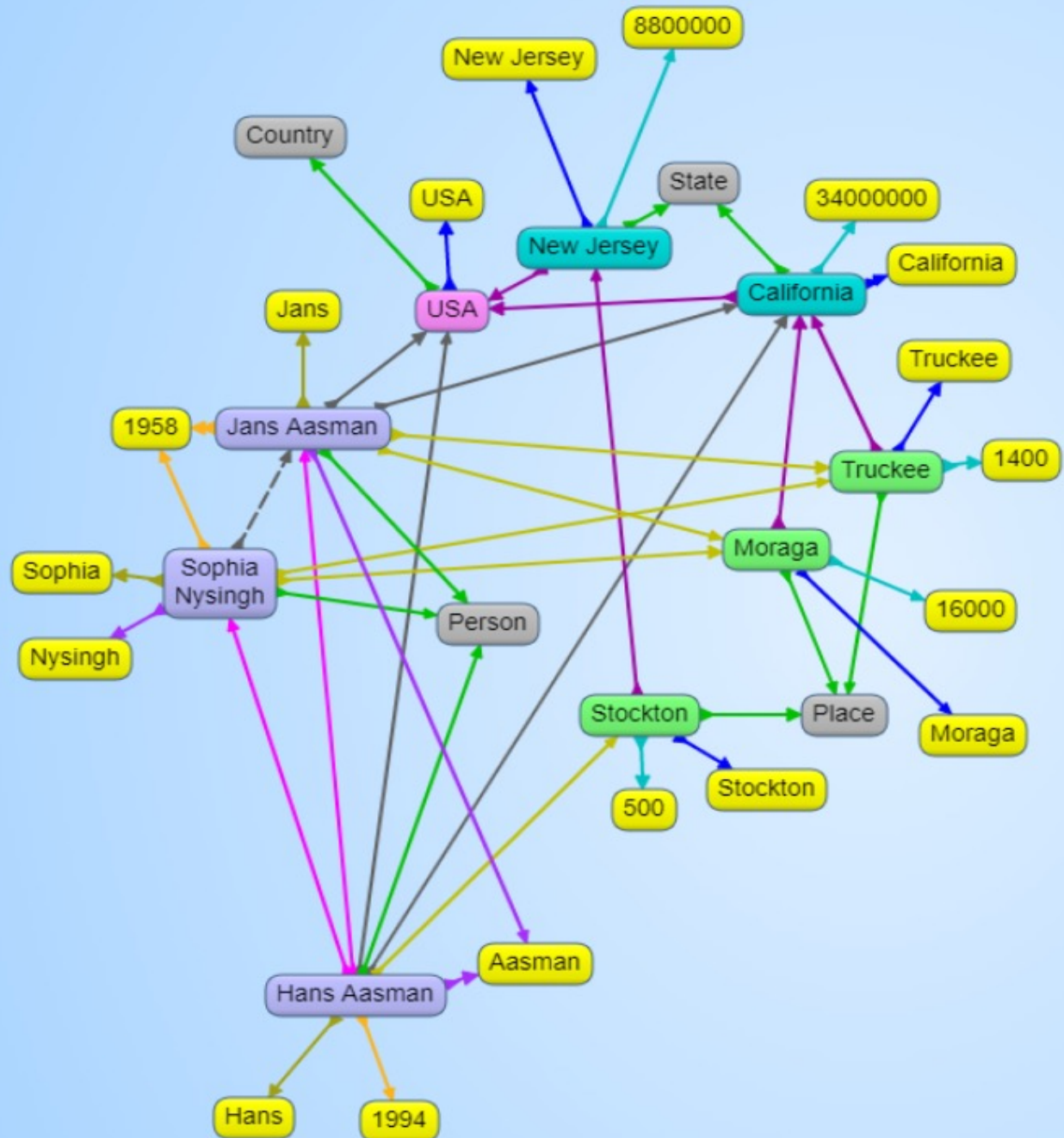
Make it slightly more interesting

```
@prefix : <http://abc.com/> .
:person1 a :Person ;
    :firstName 'Jans';
    :lastName 'Aasman' ;
    :birthYear '1958' ;
    :livesIn :place1 , :place2 .
:person2 a :Person ;
    :firstName 'Sophia';
    :lastName 'Nysingh' ;
    :birthYear '1958' ;
    :livesIn :place1 , :place2 ;
    :marriedTo :person1 .
:person3 a :Person ;
    :firstName 'Hans';
    :lastName 'Aasman' ;
    :birthYear '1994' ;
    :livesIn :place3 ;
    :hasParents :person1 , :person2 .
:place1 a :Place ; :name 'Moraga' ; :population 16000 ; :partOf :state1 .
:place2 a :Place; :name 'Truckee' ; :population 1400 ; :partOf :state1 .
:place3 a :Place; :name 'Stockton' ; :population 500 ; :partOf :state2 .
:state1 a :State ; :name 'California' ; :population 34000000 ; :partOf :country1 .
:state2 a :State ; :name 'New Jersey' ; :population 8800000 ; :partOf :country1 .
:country1 a :Country ; :name 'USA' .
:person1 :paysTaxesTo :country1, :state1 .
:person3 :paysTaxesTo :country1, :state1 .
```

Jans has two places, is married to Sophia. Jans pays taxes to California and USA and they have a son in Stockton, New Jersey

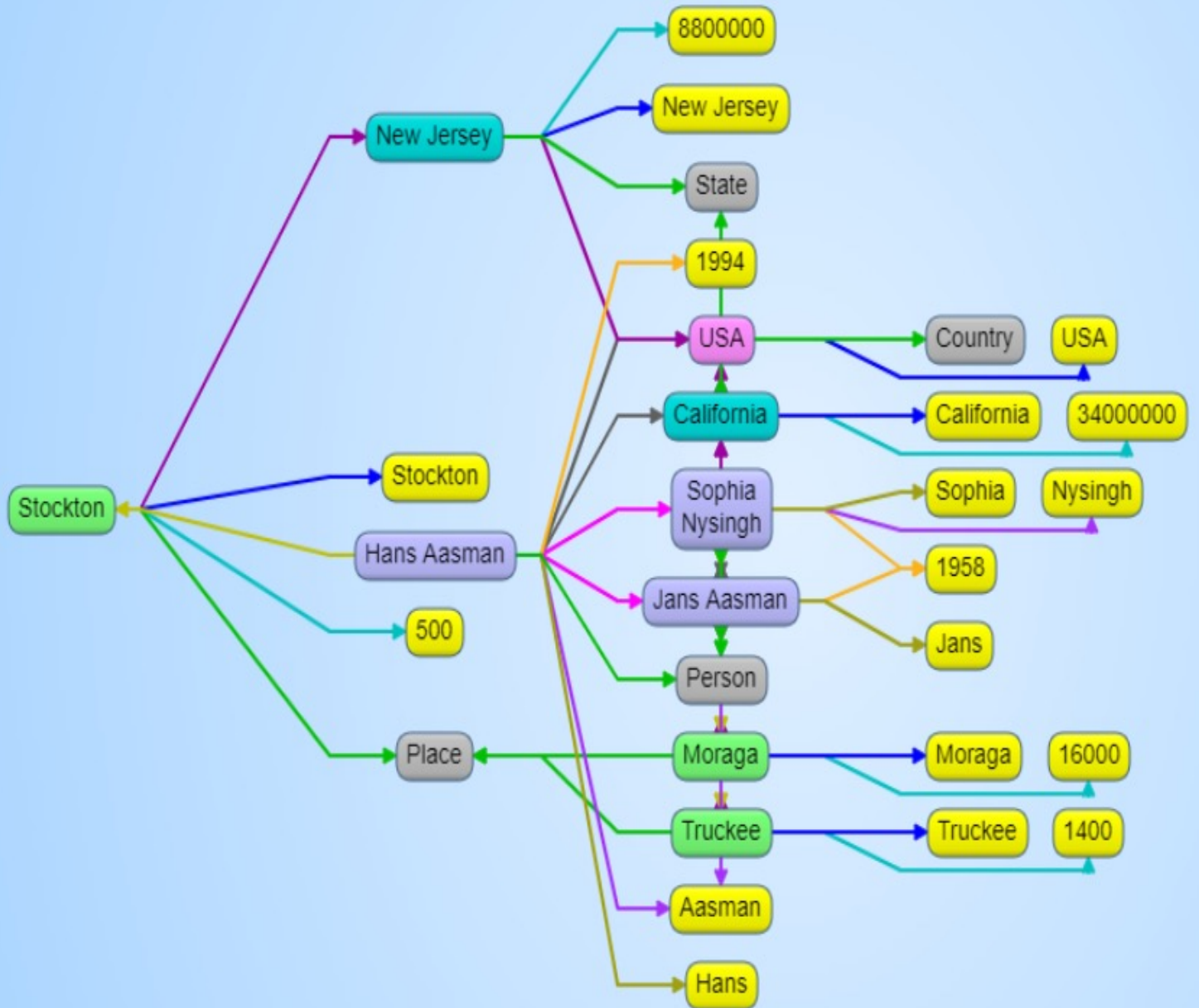
- Birth Year
- First Name
- Has Parents
- Last Name
- Lives In
- Married To
- Name
- Part Of
- Pays Taxes To
- Population
- Type

- Country
- Person
- Place
- State
- Literal
- No Type



- Birth Year
- First Name
- Has Parents
- Last Name
- Lives In
- Married To
- Name
- Part Of
- Pays Taxes To
- Population
- Type

- Country
- Person
- Place
- State
- Literal
- No Type



Visual SPARQL: find two people that lives in two places in California where one person also pays taxes in California

Gruff 8.1.1 on AllegroGraph 7.2.0 test read / write 46 triples server 127.0.0.1:10035

File View Text Search Display Remove Layout Select Edit Global Options Query Options Visual Graph Options Help

SPARQL

Prolog

Name Query

Revisit

Show Text Query

Run Query

Full Cardinality

REDUCED

DISTINCT

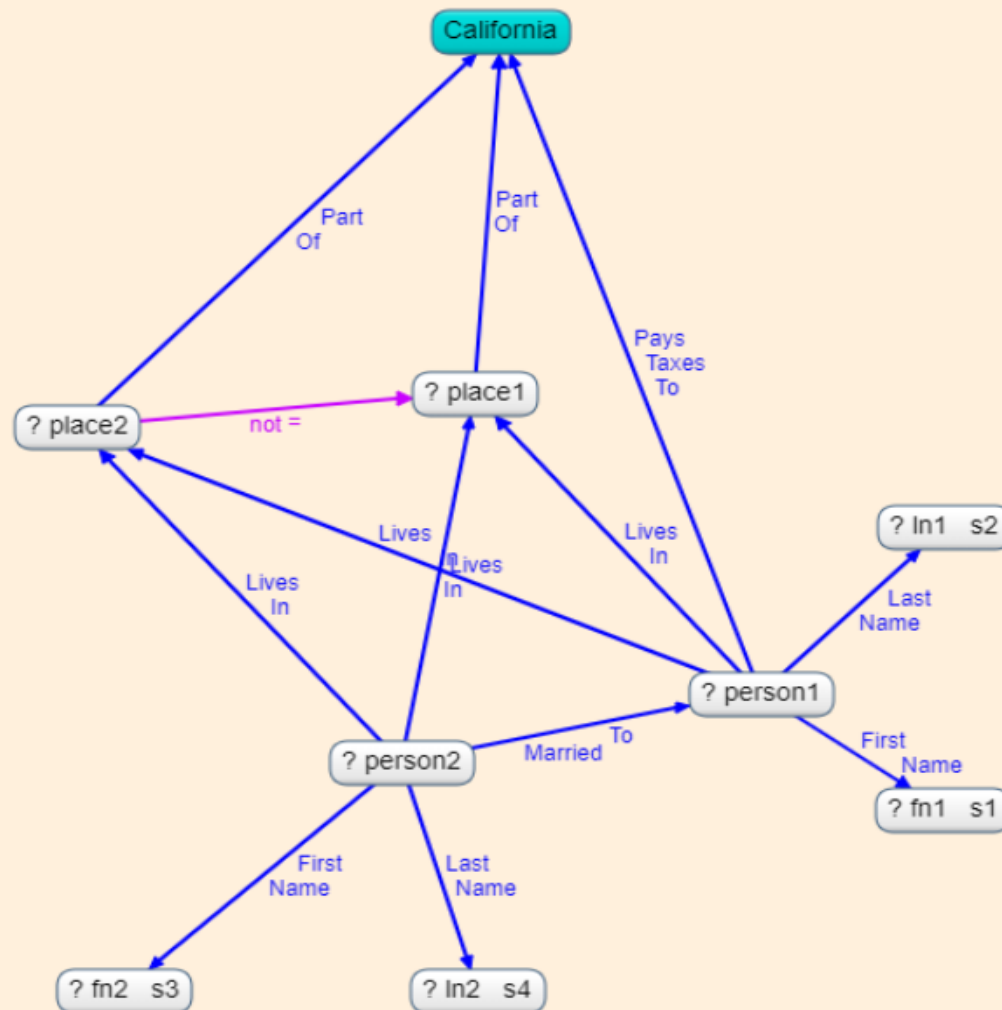
Limit

Offset

FROM

FROM NAMED

Remember to right-click in this view.



Generated SPARQL

Gruff 8.1.1 on AllegroGraph 7.2.0 test read / write 46 triples server 127.0.0.1:10035

File View Text Search Display Edit Global Options Query Options Table Options Help

SPARQL

Run Query

Reindent

Name Query

Revisit



Graph

Prolog

Select All

Query

```
prefix : <http://abc.com/>
select ?fn1 ?ln1 ?fn2 ?ln2 where
{ ?place1 :partOf :state1 .
  ?place2 :partOf :state1 .
  ?person1 :paysTaxesTo :state1 ;
           :livesIn ?place1 ,
                ?place2 ;
           :lastName ?ln1 ;
           :firstName ?fn1 .
  ?person2 :livesIn ?place1 ,
                ?place2 ;
           :marriedTo ?person1 ;
           :firstName ?fn2 ;
           :lastName ?ln2 .
  filter ( ?place2 != ?place1 ) }
limit 1
```

1 Result

Create Visual Graph

Add to Visual Graph

Write Text Report

Save as CSV

?fn1	?ln1	?fn2	?ln2
Jans	Aasman	Sophia	Nysingh

And a tiny ontology in OWL

```
:Person a owl:Class ; rdfs:subClassOf :Mammal .  
:Mammal a owl:Class .  
:hasParents a owl:ObjectProperty .  
:hasChild owl:inverseOf :hasParents .  
:marriedTo rdfs:domain :Person ;  
           rdfs:range   :Person .
```

Is Jans a Mammal

- And now we can reason too

Gruff 8.1.1 on AllegroGraph 7.2.0 test read / write 80 triples server 127.0.0.1:10035

File View Text Search Display Edit Global Options Query Options Table Options Help

SPARQL Prolog

Run Query Reindent Name Query Revisit ← →

Select All

```
prefix : <http://abc.com/>
select * { ?p :firstName 'Jans' ; a :Mammal }
```

1 Result Create Visual Graph Add to Visual Graph Write Text Report

?p
Jans Aasman

Does Jans have Children?

Gruff 8.1.1 on AllegroGraph 7.2.0 test read / write 80 triples server 127.0.0.1:10035

File View Text Search Display Edit Global Options Query Options Table Options Help

SPARQL Prolog

Run Query Reindent Name Query Revisit ← → Graph View Table View

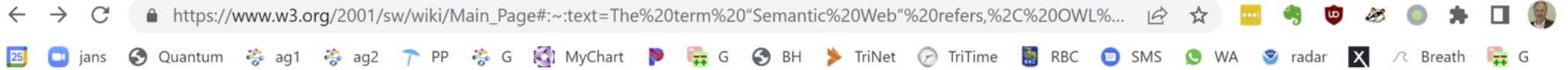
Select All

```
prefix : <http://abc.com/>
select * { ?person1 :hasChild ?person2 }
```

2 Results Create Visual Graph Add to Visual Graph Write Text Report Save as CSV

?person1	?person2
Sophia Nysingh	Hans Aasman
Jans Aasman	Hans Aasman

Semantic Technology Standards



Log in

Main page Discussion

Read

View source

View history

Search

Main Page

In addition to the classic "Web of documents" W3C is helping to build a technology stack to support a "Web of data," the sort of data you find in databases. The ultimate goal of the Web of data is to enable computers to do more useful work and to develop systems that can support trusted interactions over the network. The term "Semantic Web" refers to W3C's vision of the Web of linked data. Semantic Web technologies enable people to create data stores on the Web, build vocabularies, and write rules for handling data. Linked data are empowered by technologies such as [RDF](#), [SPARQL](#), [JSON-LD](#), [OWL](#), [SHACL](#) and [SKOS](#).



The goal of this wiki is to provide a "first stop" for more information on Semantic Web technologies, in particular on *Semantic Web Standards* published by the W3C. It does not aim to give a complete set on information on Semantic Web related events, conferences, ontologies or community efforts. There are already a number of sites maintained by the community that users can refer to (see some below).

Learn

[Books](#), [Presentations](#), [FAQ](#)

Main areas of SW

[Linked Data](#), [Vocabularies](#), [Queries](#), [Inference](#), [Vertical Applications](#)

The Standards

[RDF](#), [OWL](#), [SPARQL](#), [RDFa](#), [JSON-LD](#), [SKOS](#), [RDFS](#),
[GRDDL](#), [POWDER](#), [PROV](#), [RIF](#), [SAWSDL](#), [RDB2RDF](#), [SHACL](#)

Usage Examples

[Look at what others have done](#)

Develop

[Consult the list of available development tools](#)

Official publications

[W3C SW Publications](#)



[Main Page](#)
[Recent changes](#)
[Tools](#)
[Books](#)
[Validators](#)

Other W3C resources

[Activity news](#)
[Publications](#)
[Logos, buttons](#)
[Activity home page](#)

W3C RSS feeds

[Activity newsfeed](#)
[W3C blogs](#)
[Use cases, case studies](#)

Account request

[W3C Member](#)
[Public](#)

Tools

[What links here](#)
[Related changes](#)

Where is RDF Used

- Nearly every Fortune 500 company is building SKOS taxonomies
- In Linked Open Data for public data and enterprise data
- Nearly every Fortune 500 company is building Knowledge Graphs

RDF for taxonomies and ontologies

- All Fortune 500 companies are now building taxonomies, almost all of them RDF based
 - To improve search over documents
 - To improve NLP processes
 - To harmonize documents, databases, and processes
 - To build knowledge graphs
- Most taxonomies and ontologies are meant to be cross-enterprise and therefore mostly open source
- Two examples:

RDF based open source taxonomies and ontologies: Healthcare and Life Sciences are furthest ahead

The screenshot shows the BioPortal website interface. At the top, there is a navigation bar with the BioPortal logo and a menu icon. Below the navigation bar, a welcome message reads: "Welcome to BioPortal, the world's most comprehensive repository of biomedical ontologies".

There are two main search sections:

- Search for a class:** Includes a search input field with the placeholder "Enter a class, e.g. I" and a search button. Below the input field is a link for "Advanced Search".
- Find an ontology:** Includes a search input field with the placeholder "Start typing ontol" and a search button. Below the input field is a button labeled "Browse Ontologies" with a dropdown arrow.

Below the search sections, there are two data visualization components:

- Ontology Visits (June 2022):** A horizontal bar chart showing the number of visits for various ontologies. The x-axis represents the number of visits, ranging from 0 to 25,000. The y-axis lists ontologies: MEDDRA, SNOMEDCT, RXNORM, NDDF, and DTC. MEDDRA has the highest number of visits, followed by SNOMEDCT.
- BioPortal Statistics:** A table showing the following statistics:

Ontologies	1,004
Classes	14,686,181
Properties	36,286
Mappings	79,636,946

20 banks worked together to get a Financial Industry Business Ontology

About FIBO

The Financial Industry Business Ontology



The Financial Industry Business Ontology (FIBO) defines the sets of things that are of interest in financial business applications and the ways that those things can relate to one another. In this way, FIBO can give meaning to any data (e.g., spreadsheets, relational databases, XML documents) that describe the business of finance.

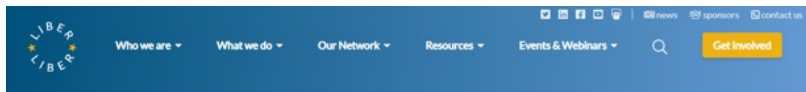
FIBO is hosted and sponsored by the Enterprise Data Management Council (EDMC) and is published in a number of formats for operating use and business definitions. FIBO is a trademark of EDM Council, Inc. It is also standardized by the Object Management Group (OMG).

FIBO is developed as an ontology in the Web Ontology Language (OWL). The language is codified by the World Wide Web Consortium (W3C), and it is based on Description Logic. The use of logic ensures that each FIBO concept is framed in a way that is unambiguous and that is readable both by humans and machines.



Linked Open Data (public)

- (semi) public (meta) data available in various RDF formats or even fully formed Knowledge Graphs



Linked Open Data: Impressions & Challenges Among Europe's Research Libraries

Posted: 04-02-2020 Topics: Strategy



A screenshot of the National Science Foundation (NSF) website. The header features the NSF logo and the tagline 'WHERE DISCOVERIES BEGIN'. A search bar is on the right. A navigation menu includes 'Research Areas', 'Funding', 'Awards', 'Document Library', 'News', and 'About NSF'. The main content area displays a 'Dear Colleague Letter: Encouraging Research on Open Knowledge Networks' dated November 18, 2021. The letter discusses the importance of open knowledge networks (OKNs) and invites principal investigators (PIs) to submit proposals for research and practice in this area. The letter is signed by the Office of Advanced Cyberinfrastructure (OAC).

Linked Open Data (enterprise)

- Create enterprise digital asset catalogs to get a handle on your data lakes

data.world powers the next-generation of



Data Discovery

Find and understand the data and analysis you need, wherever it lives. Across data sources and formats.

[MORE ABOUT DATA DISCOVERY](#)



Agile Data Governance

Enable your teams to safely access data and actively work together to transform data into knowledge.

[MORE ABOUT AGILE DATA GOVERNANCE](#)



DataOps

Accelerate the flow of information between data producers and consumers and improve the resiliency of your data supply chain.

[MORE ABOUT DATAOPS](#)

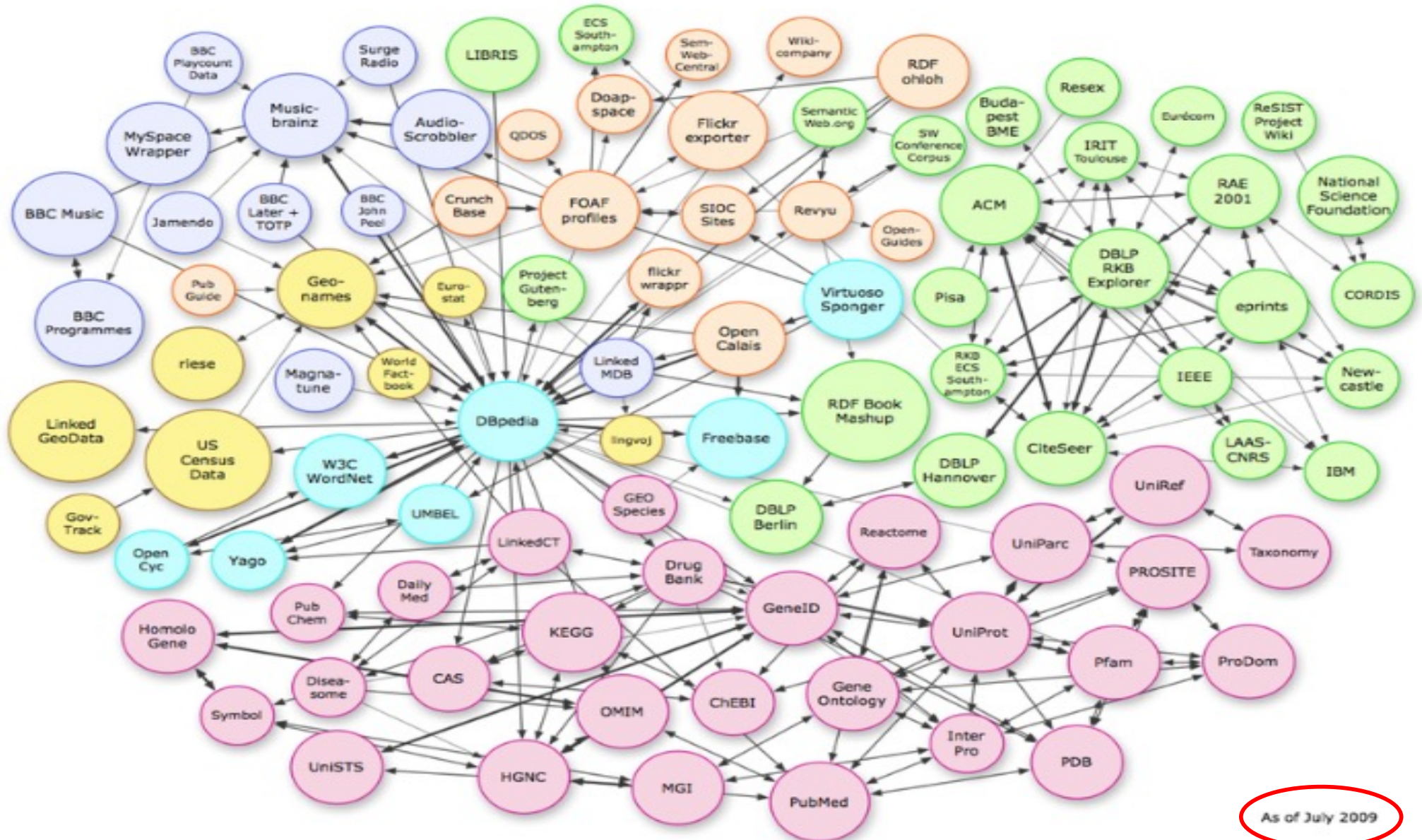
Explore the world's largest collection of open data

As part of our Public Benefit Mission, we host the world's largest free and open data resource. Discover and share impactful data, run queries using SQL or SPARQL in your browser, connect to your favorite tools, and work together to solve problems faster.

Do you have a problem or project that could benefit from a network of data problem solvers? We'd love to hear from you!

[JOIN OUR MISSION](#)

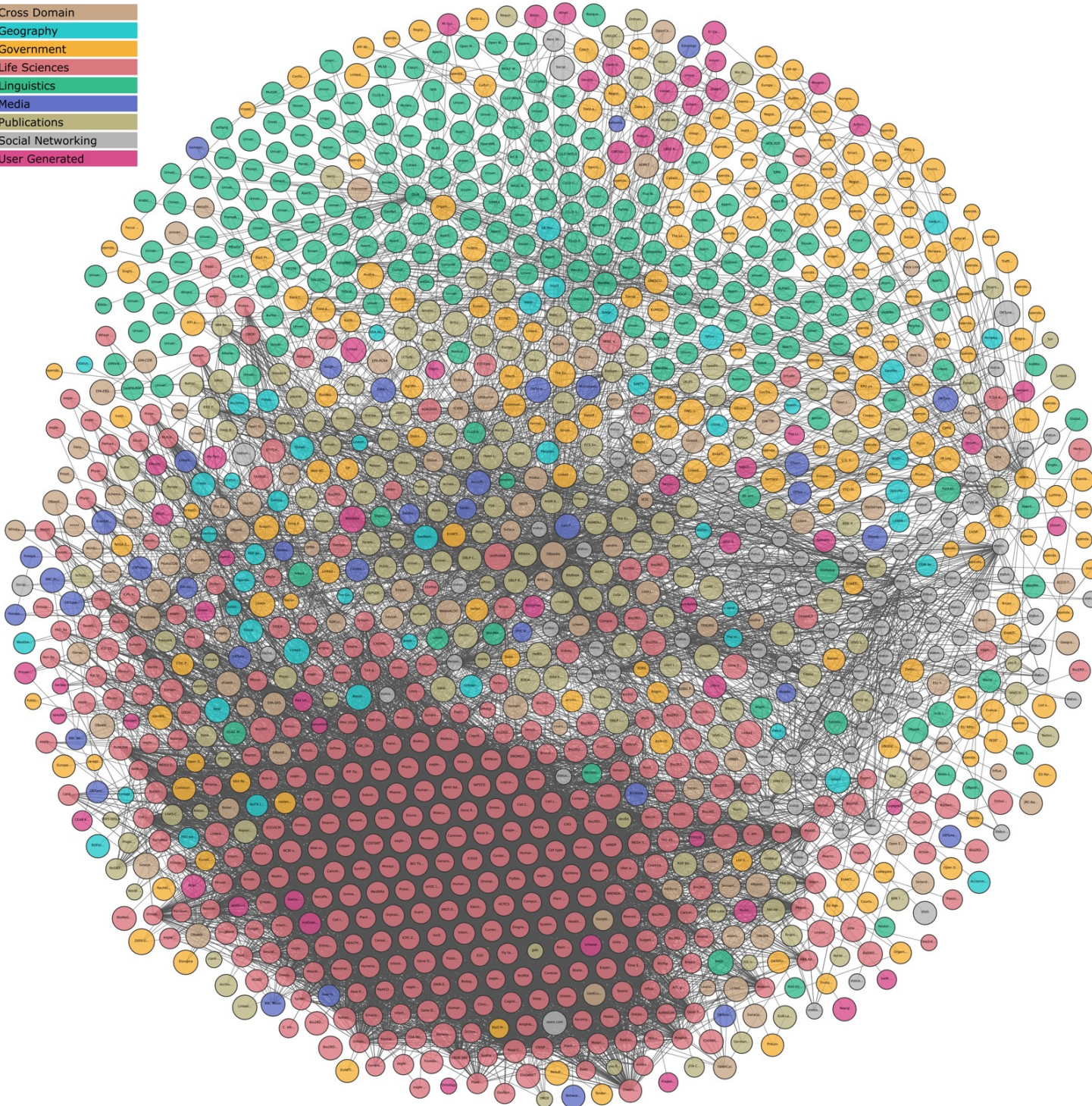
It started in the Semantic Web Community



As of July 2009

Legend

Cross Domain
Geography
Government
Life Sciences
Linguistics
Media
Publications
Social Networking
User Generated



Linked Open Knowledge 2022

Using service calls in SPARQL

Gruff 8.1.2 on AllegroGraph 7.3.0 dbpedia read / write 442,953,741 triples server 127.0.0.1:10000

File View Text Search Display Edit Global Options Query Options Table Options Help

SPARQL Example Reindent Name Query Revisit Graph View

Prolog Run Query Select All

```
PREFIX geo: <http://franz.com/ns/allegrograph/3.0/geospatial/>
PREFIX geonames: <http://sws.geonames.org/>
PREFIX dbpedia_rsrc: <http://dbpedia.org/resource/>
PREFIX dbpedia_onto: <http://dbpedia.org/ontology/>
PREFIX dbpedia_prop: <http://dbpedia.org/property/>
PREFIX census: <tag:govshare.info,2005:rdf/census/>
PREFIX census_samp: <tag:govshare.info,2005:rdf/census/details/samp/>

SELECT distinct ?censusplace ?income {
  dbpedia_rsrc:Barack_Obama dbpedia_onto:birthPlace ?birthplace .
  ?birthplace dbpedia_prop:hasGeonamesID ?geonamesresource .

  SERVICE <https://localhost:10000/catalogs/demos/repositories/geonames>
  { ?geonamesresource geonames:isAt5 ?location .

    ?otherplace geo:inCircleMiles (geonames:isAt5 ?location 10) .
    ?otherplace geonames:feature_code "PPL" .
    ?geonamesresource geonames:feature_code "PPL" .

  }

  SERVICE <https://localhost:10000/catalogs/demos/repositories/census>
  { ?censusplace dbpedia_prop:hasGeonamesID ?otherplace .

    ?censusplace census:details ?detail .
    ?detail census_samp:population15YearsAndOverWithIncomeIn1999 ?d .
    ?d census_samp:medianIncomeIn1999 ?income .

  }
}
```

4 Results [Create Visual Graph](#) [Add to Visual Graph](#) [Write Text Report](#) [Save as](#)

?censusplace	?income
Iroquois point	33005
Pearl city	25776
Waimalu	34777
Maunawili	41886

Knowledge Graphs on the rise

Gartner Hype Cycle for Emerging Technologies, 2019

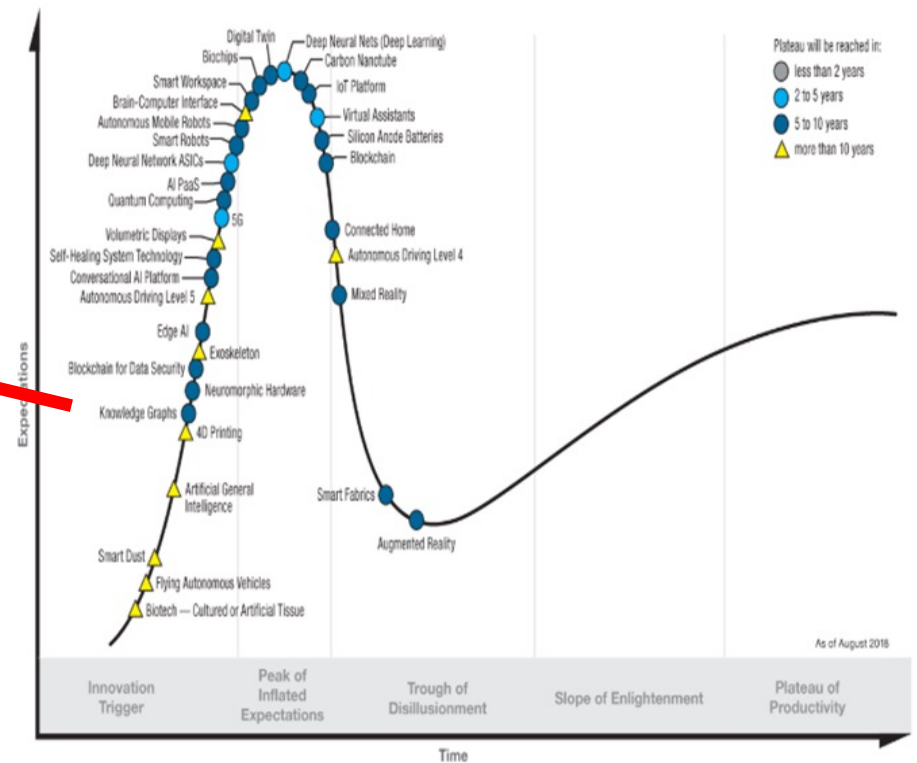


gartner.com/SmarterWithGartner

Source: Gartner
© 2019 Gartner, Inc. and/or its affiliates. All rights reserved.

Gartner

Hype Cycle for Emerging Technologies, 2018



gartner.com/SmarterWithGartner

Source: Gartner (August 2018)
© 2018 Gartner, Inc. and/or its affiliates. All rights reserved.

Gartner

All the big ones in the US heavily investing in it

Day 1

9:00	Christos Boutsidis	Goldman Sachs	Pythia: the Goldman Sachs Social Graph
9:20	Patricia Branum	Capital One	Knowledge Graph Pilot Improves Data Quality While Providing a Customer 360 View
9:40	David Newman	Wells Fargo	Knowledge Graphs and AI: The Future of Financial Data
10:00	Tim Baker	Refinitiv	Financial Crime
break 10:20 - 11:00	<i>Coffee and snacks will be provided</i>		
11:00	Denny Vrandecic	Google AI	Wikidata, Knowledge Graphs, and Beyond
11:20	Pierre Haren	Causality Link	Graphs
11:40	Dieter Fensel	OnLim	Talking Knowledge Graphs
12:00	Chris Brockmann	Eccenca	Knowledge Graph for Digital Transformation in the Supply-Chain
12:20	Tom Plasterer	Astrazeneca	FAIR Data Knowledge Graphs – From Theory to Practice
lunch 12:40 - 1:40	<i>A light lunch will be provided</i>		
1:40	Subhabrata Mukherjee	Amazon	Deep Learning for Knowledge Extraction and Integration to build the Amazon Product Graph
2:00	Teresa Tung	Accenture	Using a Domain Knowledge Graph to Manage AI at Scale
2:20	Alfio Gliozzo	IBM Research	Extending Knowledge Graphs using Distantly Supervised Deep Nets
2:40	Michael Tung	Diffbot	Knowledge Graphs for AI
break 3:00 - 3:30	<i>Coffee and snacks will be provided</i>		
3:30	Xiaoya Wei	Airbnb	Knowledge Graph at Airbnb
3:50	Amy Hodler	Neo4j	A Real-World Guide to Building Your Knowledge Graphs
4:10	Juan F. Sequeda	Capsenta	Designing and Building Enterprise Knowledge Graphs from Relational Databases in the Real V
4:30	Ron Snyder	ITHAKA / JSTOR	Why Wikibase? Why not?
4:50	Sören Auer	TIB	Creating a knowledge graph based Enterprise Data Innovation Architecture



Data Silos ➡ Trapped Knowledge

- Enterprises lack a **single source of truth** on their customers and their data
- Data scientists are expensive and waste 90% of their time on prep
- Current solutions (ETL, Data Lakes, MDM, Neo4J) **exacerbate the Silo problem**





Graph Market will grow from \$1.9 billion in 2021 to \$5.1 billion by 2026, CAGR of 22.5%

Gartner Client inquiry interest in data fabric in 2021 is approx. 900% of that in 2019



Graph > Knowledge Graph > Data Fabric

#3 Watch out for graph databases



Carl Olofson, Research Vice President, IDC

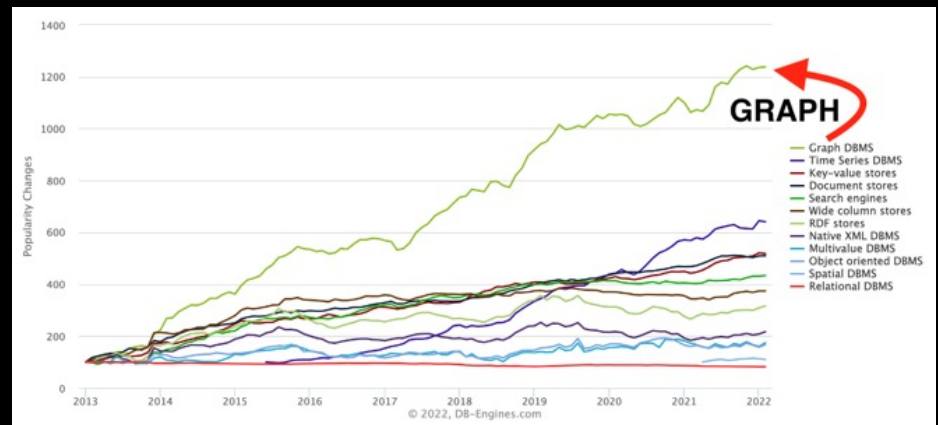
- Next revolution in database technology
- Potential for 600% growth over the decade
- Will slowly gain traction over the next five years
- Lots of use cases but not well understood
 - Semantic use cases (lots of AI examples)
 - Property graphs with dozens of use cases
- Graph databases dramatically different from traditional RDBMS

Gartner.

Top Strategic Technology Trends for 2022: Data Fabric

Published 18 October 2021 - ID G00749680 - 15 min read

By Mark Beyer, Ehtisham Zaidi, and 2 more



Knowledge Graphs are the Key to Data Fabrics

Connectible:

- KGs can bridge all data and metadata "Silos" for seamless data integration and management

Flexible:

- Express any data and metadata: this is important as new data sources get added to the fabric and new metadata is collected

Integratable:

- Offer the most complete, open and flexible APIs

Expressive:

- Capture semantically rich models to assist with the understanding of data and metadata

Composable:

- Make it easy to incrementally evolve a data fabric

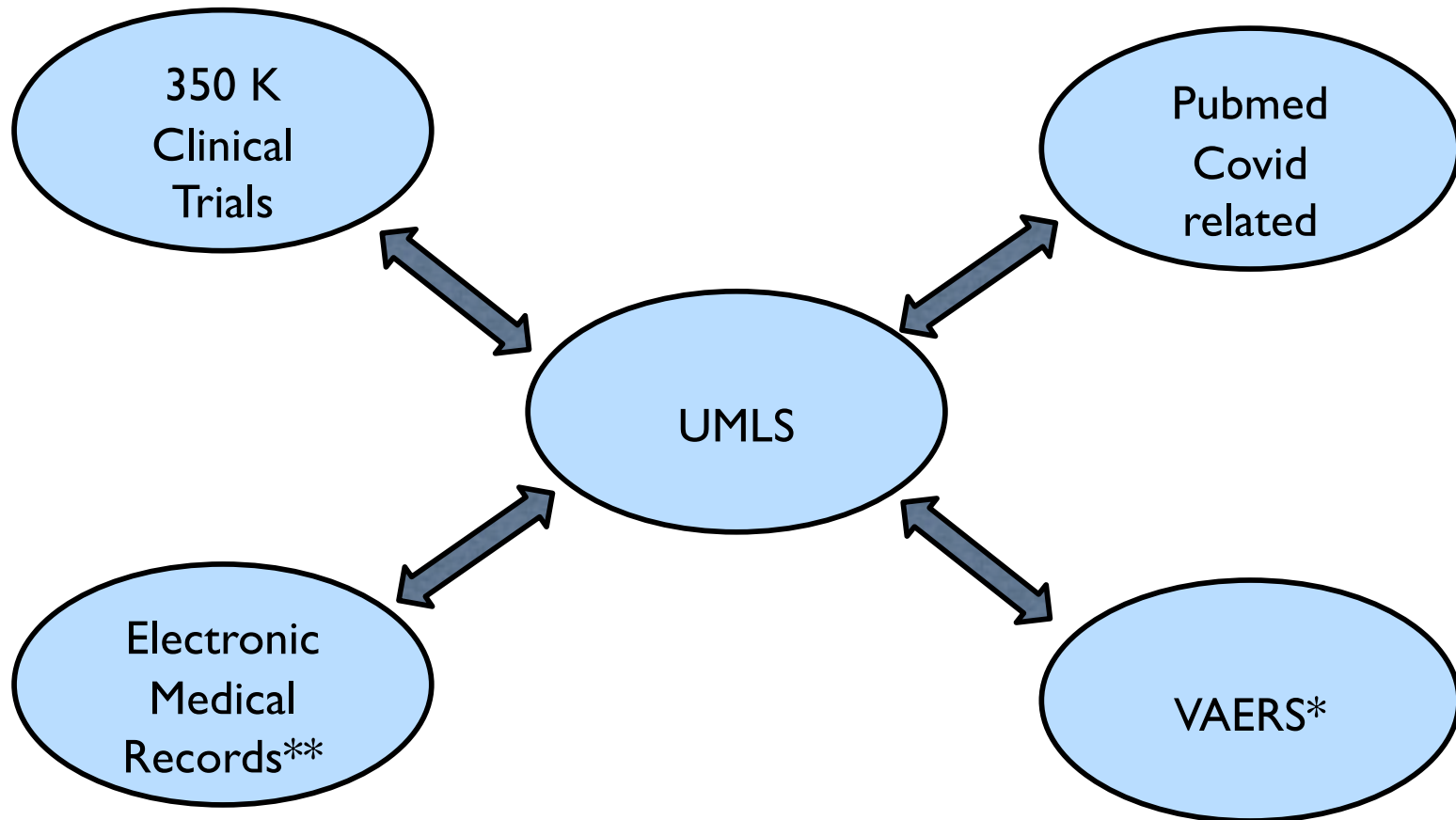
Future Proof:

- Based on Standards

Intelligent:

- Integrate rules-based reasoning and machine learning: KGs provide the semantics layer that adds additional context and meaning enabling better and more informed analytics and AI/ML

An example open source knowledge graph for Long Covid Research



* Adverse vaccine reactions

** Synthea

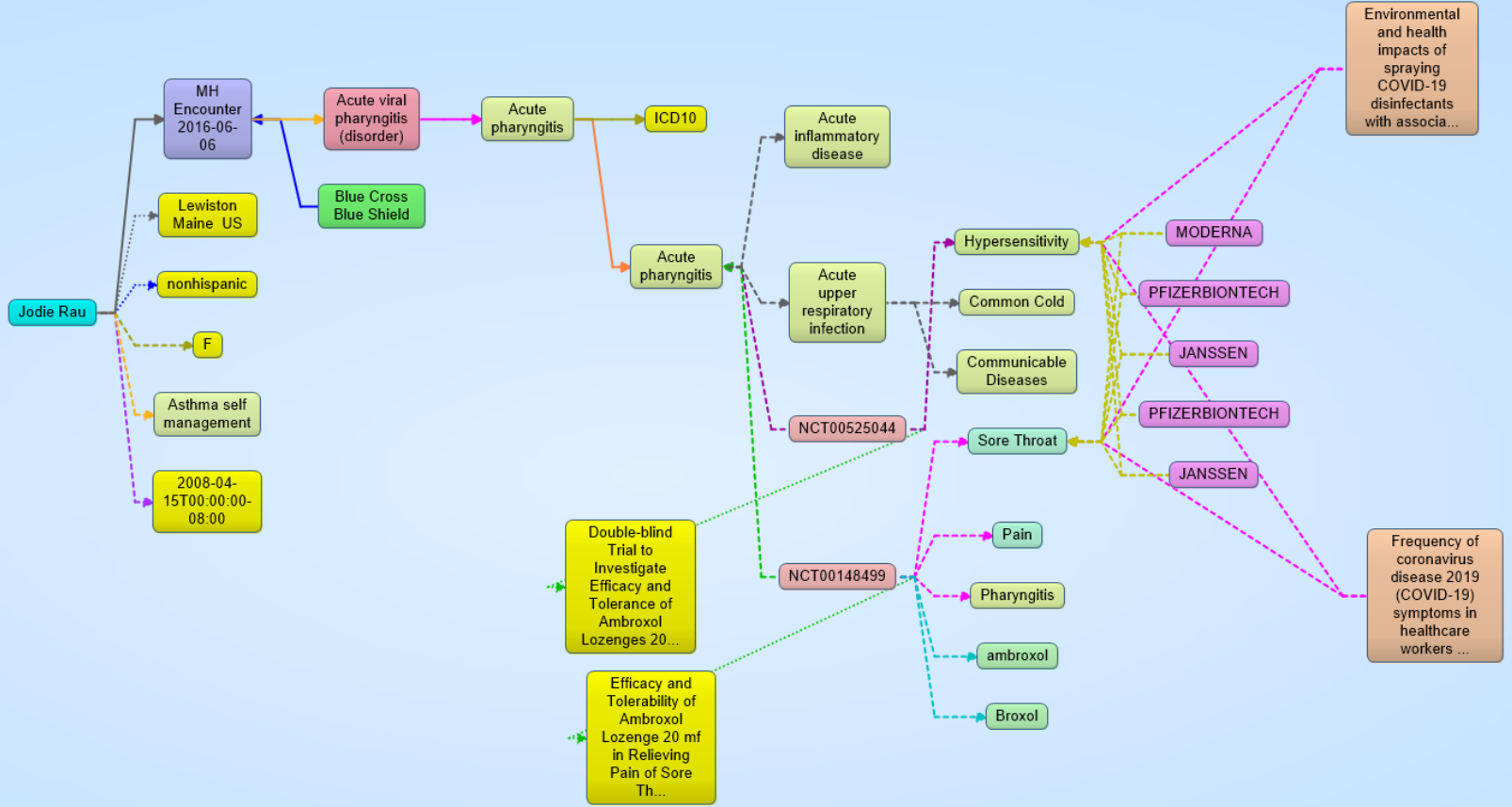
Used by Long Covid Researchers

Gruff 8.1.0 on AllegroGraph 7.2.0 ag-healthcare read / write 470,831,141 triples server localhost:10035

File View Text Search Display Link Remove Layout Select Edit Global Options Visual Graph Options Help

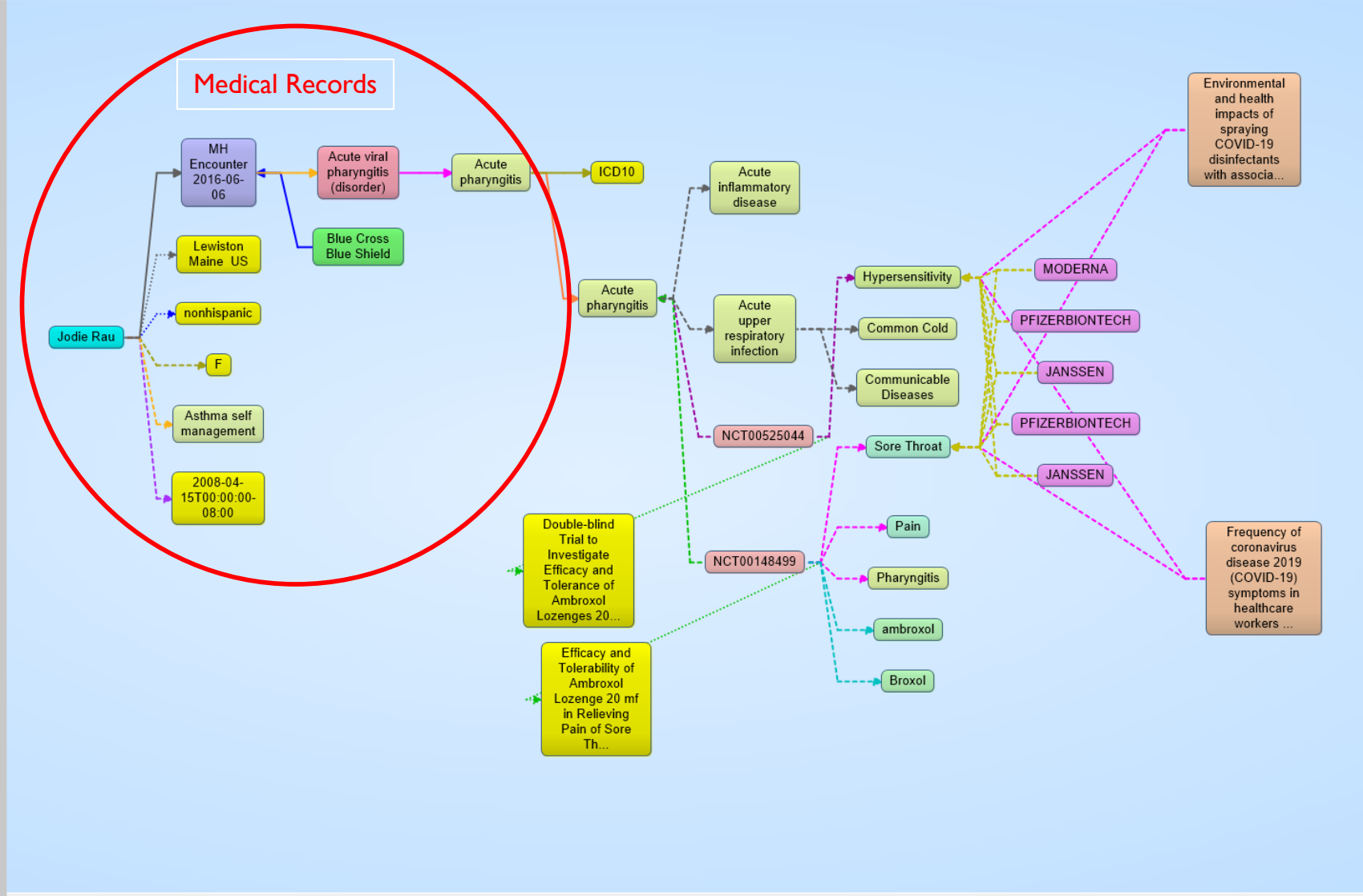
Brief title
 Broader
 Code
 Eligibility criteria mentions disease
 Encounter Condition
 Ethnicity
 Exact Match
 Exclusion criteria mentions disease
 Gender
 In Scheme
 Mentions disease
 Mentions disease
 Mentions drug
 Patient Careplan
 Patient Encounter
 Payer Encounter

Article
 Careplan
 Clinical Trials
 Condition
 Disease or Syndrome
 Encounter
 Pathologic Function
 Patient
 Payer
 Pharmacologic Substance
 Report



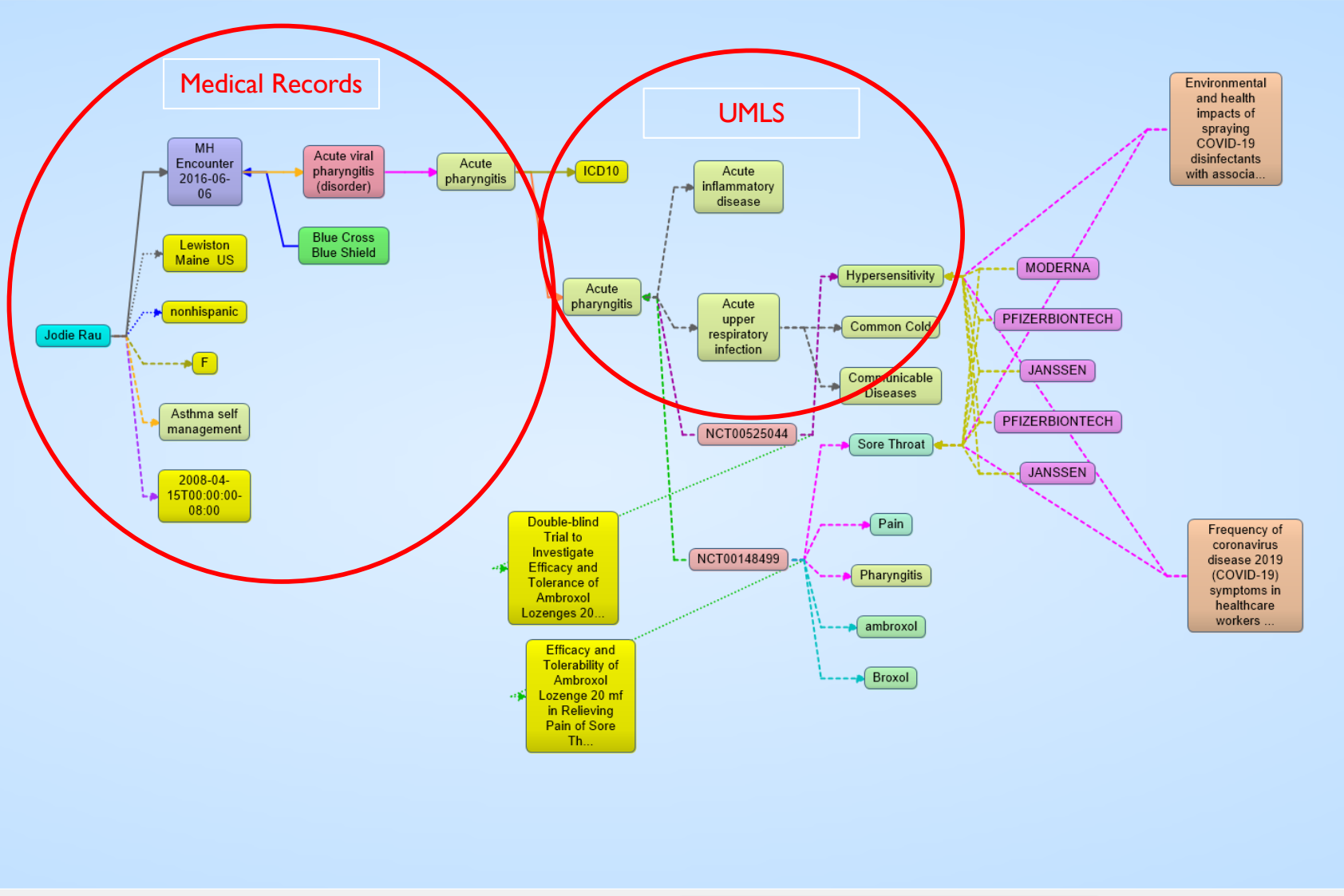
Brief title
Broader
Code
Eligibility criteria mentions disease
Encounter Condition
Ethnicity
Exact Match
Exclusion criteria mentions disease
Gender
In Scheme
Mentions disease
Mentions disease
Mentions drug
Patient Careplan
Patient Encounter
Payer Encounter

Article
Careplan
Clinical Trials
Condition
Disease or Syndrome
Encounter
Pathologic Function
Patient
Payer
Pharmacologic Substance
Report



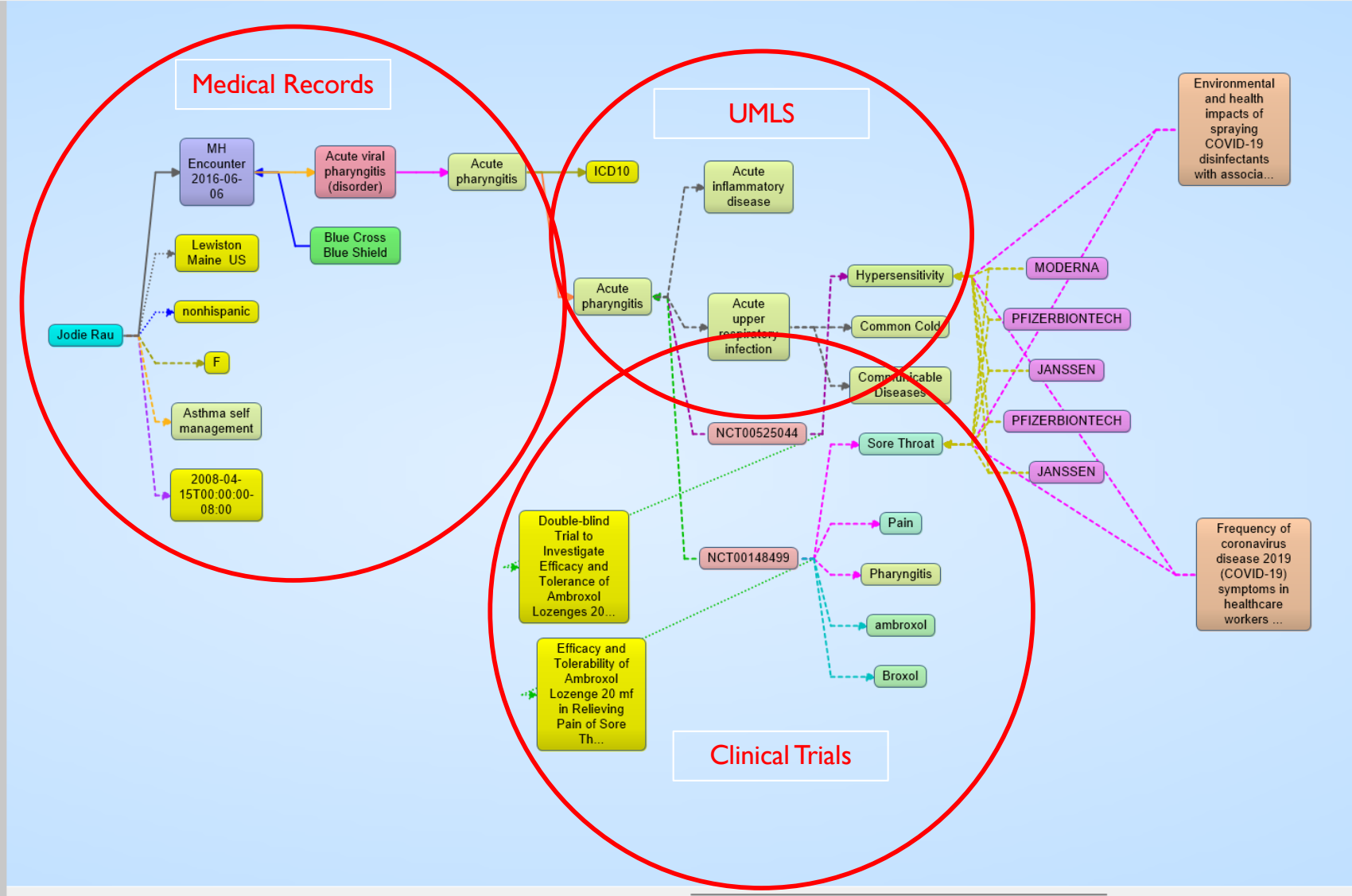
Brief title
Broader
Code
Eligibility criteria mentions disease
Encounter Condition
Ethnicity
Exact Match
Exclusion criteria mentions disease
Gender
In Scheme
Mentions disease
Mentions disease
Mentions drug
Patient Careplan
Patient Encounter
Payer Encounter

Article
Careplan
Clinical Trials
Condition
Disease or Syndrome
Encounter
Pathologic Function
Patient
Payer
Pharmacologic Substance
Report



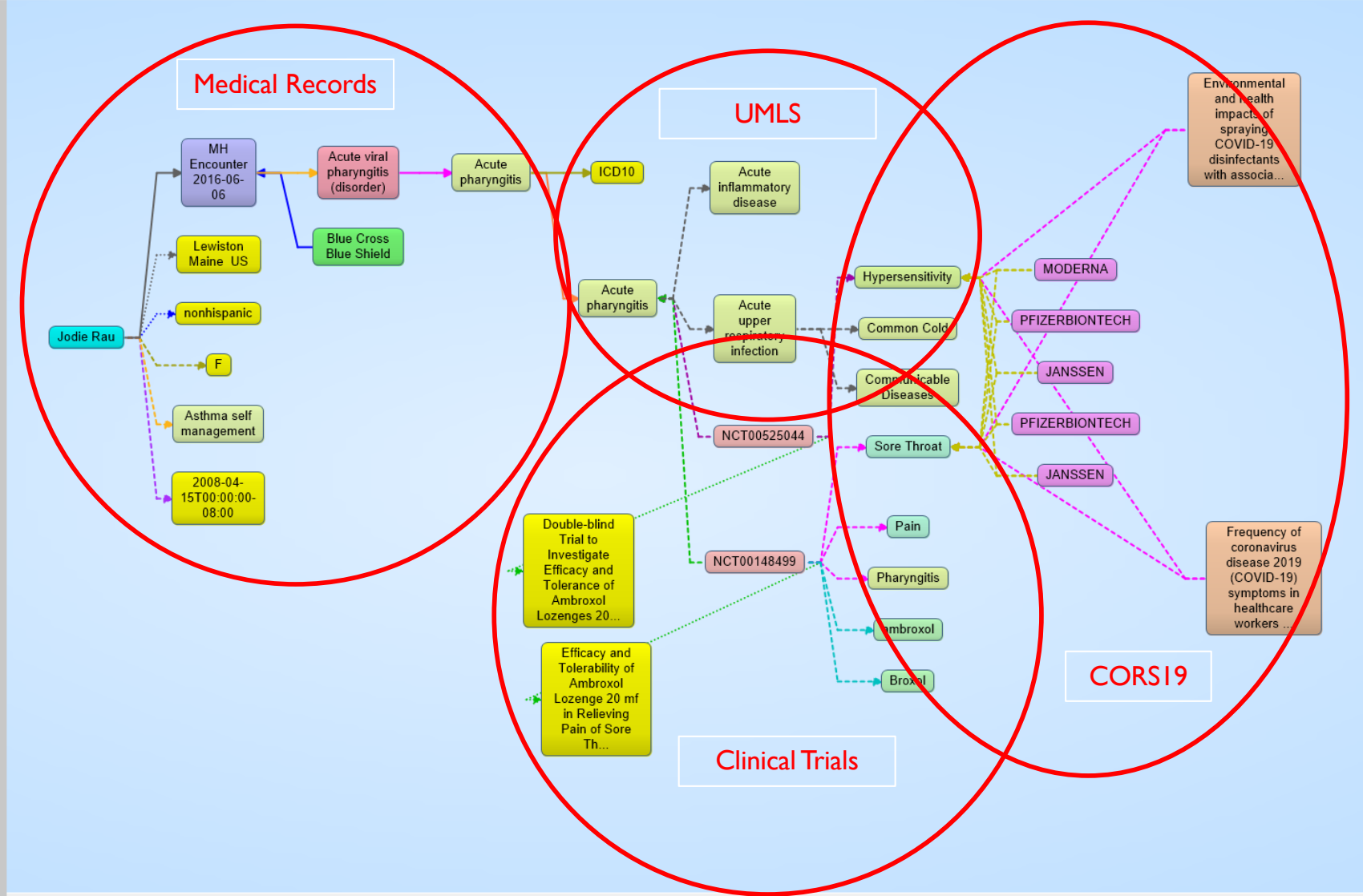
Brief title
Broader
Code
Eligibility criteria mentions disease
Encounter Condition
Ethnicity
Exact Match
Exclusion criteria mentions disease
Gender
In Scheme
Mentions disease
Mentions disease
Mentions drug
Patient Careplan
Patient Encounter
Payer Encounter

Article
Careplan
Clinical Trials
Condition
Disease or Syndrome
Encounter
Pathologic Function
Patient
Payer
Pharmacologic Substance
Report



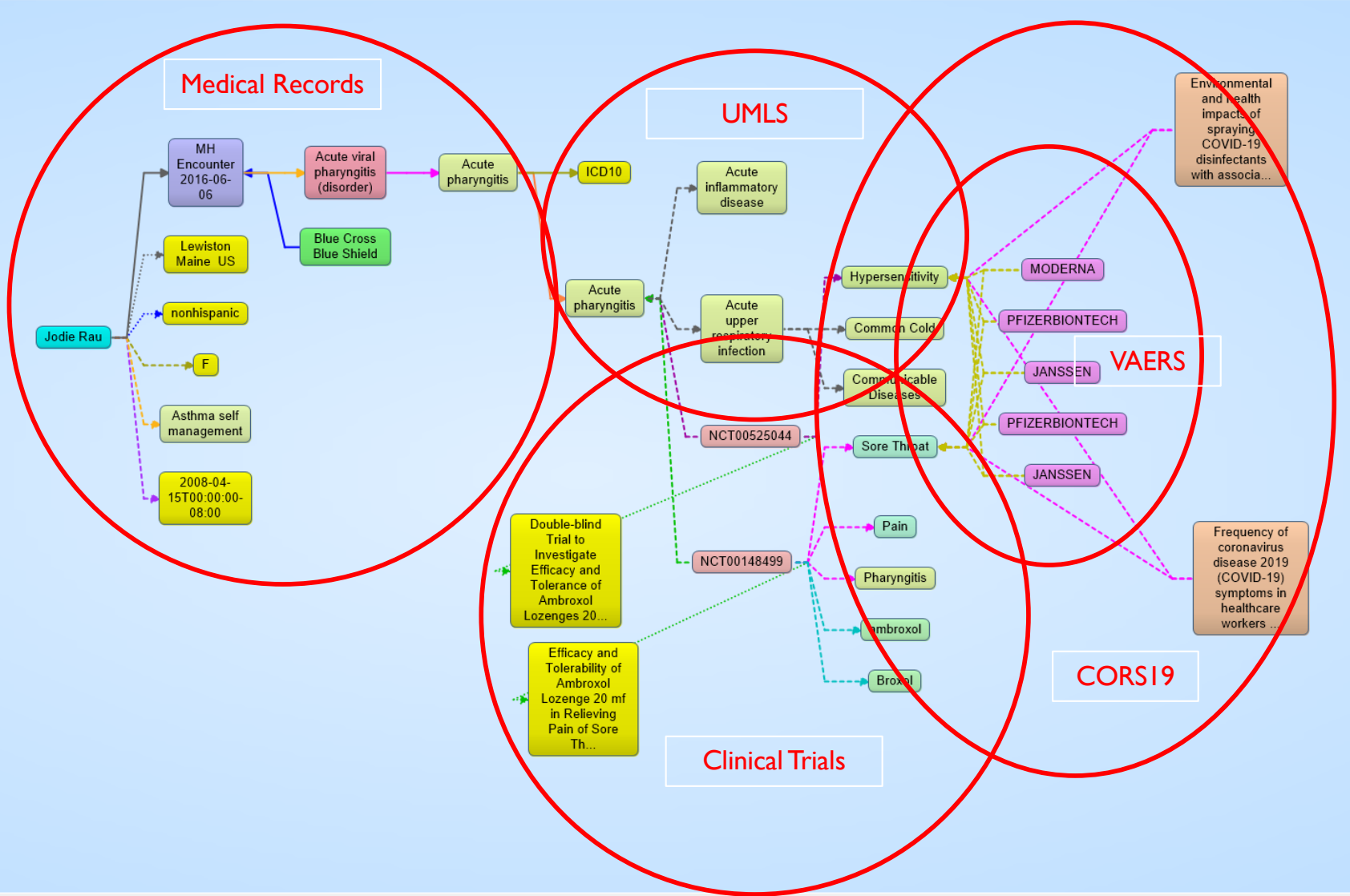
Brief title
Broader
Code
Eligibility criteria mentions disease
Encounter Condition
Ethnicity
Exact Match
Exclusion criteria mentions disease
Gender
In Scheme
Mentions disease
Mentions disease
Mentions drug
Patient Careplan
Patient Encounter
Payer Encounter

Article
Careplan
Clinical Trials
Condition
Disease or Syndrome
Encounter
Pathologic Function
Patient
Payer
Pharmacologic Substance
Report



Brief title
Broader
Code
Eligibility criteria mentions disease
Encounter Condition
Ethnicity
Exact Match
Exclusion criteria mentions disease
Gender
In Scheme
Mentions disease
Mentions disease
Mentions drug
Patient Careplan
Patient Encounter
Payer Encounter

Article
Careplan
Clinical Trials
Condition
Disease or Syndrome
Encounter
Pathologic Function
Patient
Payer
Pharmacologic Substance
Report



Now onto JSON-LD

- Knowledge Graphs are getting popular very fast but developers a little bit scared to learn the W3C Semantic Stack.

How do you make it easy as MongoDB to

- **Add data** to a knowledge graph
- **Retrieve data** from a knowledge graph
- **Validate** your data

Solution: JSON-LD

- JSON-LD will help you add, retrieve, validate, and delete objects to a Knowledge Graph as easy as MongoDB

First about JSON: it somehow won

- Messaging:
 - the lingua franca for messaging and data exchange
- Configuration:
 - JSON is replacing XML for configuration of nearly anything
- Document and key/value store:
 - JSON is the main data format stored in Document Stores (Couchbase, Mongo, etc...)

JSON – the good

- Simple standard:
 - Json.org spec is 5 pages, XML spec on W3C = 60 pages 😊
 - only a few datatypes and with arrays!
 - you can make your own complex data types if you want
- Easy to read and parse by humans and machines
- Easy to store in document stores
- Easy to program: support in every programming language

When JSON runs into trouble

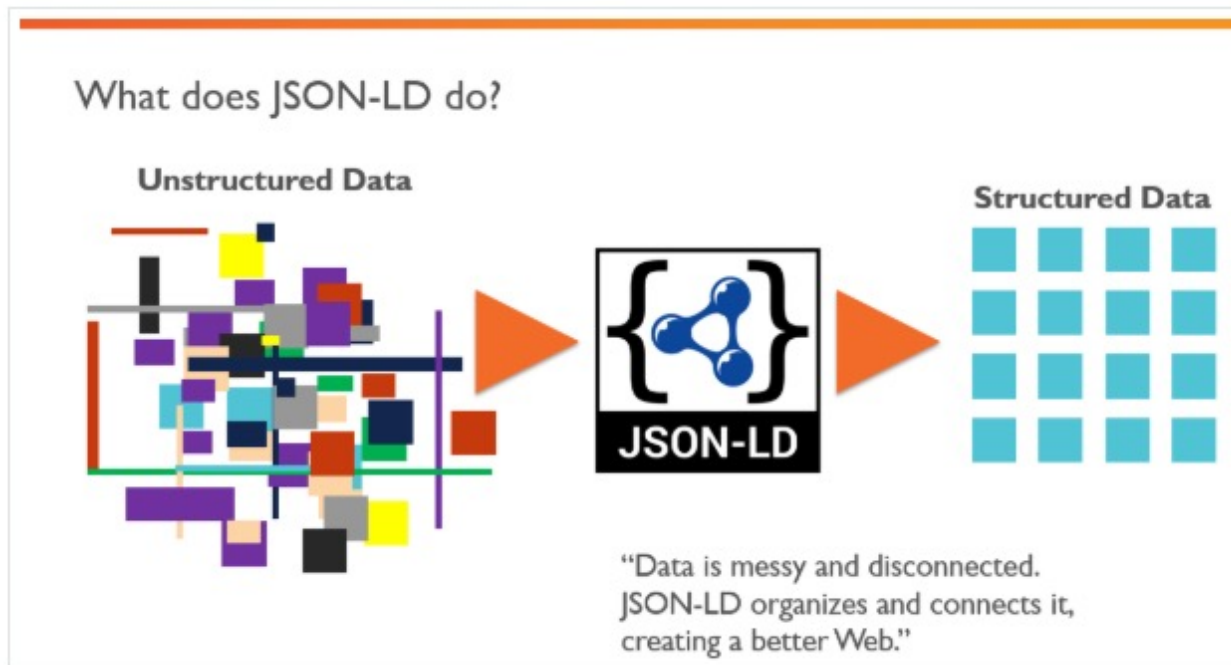
- In a hospital setting you might have > 1000 JSON types that stream between various sources and knowledge graphs
- All objects need to be persisted in a data lake but I don't want to create silos of document stores
- If I look at any random JSON object I have to find from the code to see what it means
- There are graphs in the data but how do I point from one object to the other in a standardized way?

We need our JSON to have Semantics

- Every JSON object needs an Identity that other objects can point to
- Every JSON object needs a Type
- Every Type has an ontology (schema) that describes
 - The attributes and datatypes
 - Links to other types of objects
 - Version information, links to processes and apps that use it
- And a standard Validation Language

JSON-LD = 100 % JSON +

- Add basic [schema support](#) to JSON: (but SHACL more complete)
- Add [semantics](#) to JSON objects: what does this attribute mean
- Designed to [link](#) JSON objects together
- Enables [joins](#) and [graph search](#) in document stores



Learn from JSON-LD.ORG

JSON-LD Playground

Play around with JSON-LD markup by typing out some JSON below and seeing what gets generated from it at the bottom of the page. Pick any of the examples below to get started.

NOTE: The playground uses [jsonld.js](#) which conforms to JSON-LD 1.0 syntax, API, framing, and errata, the W3C Community Group JSON-LD 1.1 syntax, API, and framing drafts, and partial support of the W3C Working Group JSON-LD 1.1 syntax, API, and framing drafts. Also see the classic [JSON-LD 1.0 playground](#) and the [RDF Distiller](#).

Examples: [Person](#) [Event](#) [Place](#) [Product](#) [Recipe](#) [Library](#) [Activity](#)

[Permalink](#) [Gist](#) [Shortcuts](#)

[JSON-LD Input](#) [Options](#)

[Document URL](#)

```
{
  "@context": "http://schema.org/",
  "@type": "Person",
  "@id": "foaf:person-1",
  "name": "Jane Doe",
  "jobTitle": "Professor",
  "telephone": "(425) 123-4567",
  "url": "http://www.janedoe.com"
}
```

[Expanded](#) [Compacted](#) [Flattened](#) [Framed](#) [N-Quads](#) [Normalized](#) [Table](#) [Visualized](#) [Signed with RSA](#) [Signed with Bitcoin](#)

```
<http://xmlns.com/foaf/0.1/person-1> <http://schema.org/jobTitle> "Professor" .
<http://xmlns.com/foaf/0.1/person-1> <http://schema.org/name> "Jane Doe" .
<http://xmlns.com/foaf/0.1/person-1> <http://schema.org/telephone> "(425) 123-4567" .
<http://xmlns.com/foaf/0.1/person-1> <http://schema.org/url> <http://www.janedoe.com> .
<http://xmlns.com/foaf/0.1/person-1> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://schema.org/Person> .
```


It is everywhere: let's look at this product

Browser tabs: A Guide to JSON-LD for Begin... | JSON: The Fat-Free Alternativ... | Mad Hippie Antioxidant Facia... | facial oil - Google Search

Address bar: https://www.ulta.com/antioxidant-facial-oil?productId=xlsImpprod18731241&sku=2530018&cmpid=PS_

Page Header: **ULTA BEAUTY** Search [] FIND A STORE | EMAIL SIGNUP | GIFT CARDS | SIGN IN | REWARDS [] []


Shipping: **FREE STANDARD SHIPPING** on any \$50 purchase.

Breadcrumbs: Home / Skin Care / Moisturizers / Face Oil / Antioxidant Facial Oil

Navigation Menu:

- SHOP BY BRAND
- NEW ARRIVALS
- MAKEUP
- NAILS
- SKIN CARE
- HAIR
- TOOLS & BRUSHES
- FRAGRANCE
- BATH & BODY
- MEN
- ULTA COLLECTIONS
- GIFTS
- SALES & COUPONS
- CURRENT AD
- BEAUTY TIPS
- ALLURE WINNERS
- BEAUTY SERVICES
- BOOK APPOINTMENT

Product Images:



Product Name: **MAD HIPPIE** Antioxidant Facial Oil

Rating: ★★★★★ REVIEWS > | Q&A >

1.02 oz Item 2530018

\$24.99

Quantity: 1 [v]

Add To Bag []

Social Media: [f] [t] [p]

Details

Make peace with your inner hippie while hydrating & protecting against photo-aging....Mad Hippie preservative-

Search for @context in the source

```
href="https://www.ulta.com/stores" target="_self" tabindex="0" data-nav-description="bc - book appointment /book appointments/6/"/>  
</div></div>  
    <div id='js-mobileBody'><div class="ProductPage" data-reactroot=""><div class="BaseLayout"><div class="ProductPage__widthConstrainer">  
<div class="MetaData"><div class="ProductDetail"><div class="ProductDetail__wrapper"><script type="application/ld+json">  
{"@context":"http://schema.org", "@type":"Product", "aggregateRating":  
{"@type":"AggregateRating", "ratingValue":4.6, "reviewCount":66}, "description":"Make peace with your inner hippie while hydrating & protecting  
against photoaging...Mad Hippie's preservative-free Antioxidant Facial Oil is truly the most natural way to moisturize.", "brand":"Mad  
Hippie", "name":"Antioxidant Facial Oil", "image":"https://images.ultra.com/is/image/Ultra/2530018", "productID":"2530018", "offers":  
{"@type":"Offer", "availability":"http://schema.org/InStock", "price":"24.99", "priceCurrency":"USD"}}</script><div class="Breadcrumb"><script  
type="application/ld+json">{"@context":"http://schema.org", "@type":"BreadcrumbList", "itemListElement":[{"@type":"ListItem", "position":1, "item":  
{"@id":"//www.ultra.com/", "name":"Home"}], [{"@type":"ListItem", "position":2, "item":{"@id":"https://www.ultra.com/skin-care?N=2707", "name":"Skin  
Care"}], [{"@type":"ListItem", "position":3, "item":{"@id":"https://www.ultra.com/skin-care-moisturizers?N=2796", "name":"Moisturizers"}],  
{"@type":"ListItem", "position":4, "item":{"@id":"https://www.ultra.com/skin-care-moisturizers-face-oil?N=27hb", "name":"Face Oil"}],  
{"@type":"ListItem", "position":5, "item":{"@id":"https://www.ultra.com/antioxidant-facial-oil?productId=xlsImpprod18731241", "name":"Antioxidant  
Facial Oil"}]}]</script><ul><li><a class="Anchor Tertiary" href="//www.ultra.com/" target="_self" tabindex="0" data-nav-description="bc -  
home">Home</a><a class="Anchor Tertiary" href="https://www.ultra.com/skin-care?N=2707" target="_self" tabindex="0" data-nav-description="bc - skin  
care">Skin Care</a><a class="Anchor Tertiary" href="https://www.ultra.com/skin-care-moisturizers?N=2796" target="_self" tabindex="0" data-nav-  
description="bc - skin care:moisturizers">Moisturizers</a><a class="Anchor Tertiary" href="https://www.ultra.com/skin-care-moisturizers-face-oil?  
N=27hb" target="_self" tabindex="0" data-nav-description="bc - skin care:moisturizers:face oil">Face Oil</a></li></ul></div></div></div></div></div></div>
```

JSON alone would lead to confusion, JSON-LD and SCHEMA.ORG to the rescue

```
{ "aggregateRating":  
  { "ratingValue": 4.6,  
    "reviewCount": 66 },  
  "description": "Makepeace with your inner hippie while hydrating & protecting against  
    photoaging...Mad Hippie's preservative-free Antioxidant Facial Oil is  
    truly the most natural way to moisturize.",  
  "brand": "Mad Hippie",  
  "name": "Antioxidant Facial Oil",  
  "image": "https://images.ulta.com/is/image/Ulta/2530018",  
  "productID": "2530018",  
  "offers":  
    { "availability": "http://schema.org/InStock",  
      "price": "24.99",  
      "priceCurrency": "USD" } }
```

NO Meaning

```
{ "@context": "http://schema.org",  
  "@type": "Product",  
  "aggregateRating":  
    { "@type": "AggregateRating",  
      "ratingValue": 4.6,  
      "reviewCount": 66 },  
  "description": "Makepeace with your inner hippie while hydrating & protecting against  
    photoaging...Mad Hippie's preservative-free Antioxidant Facial Oil is  
    truly the most natural way to moisturize.",  
  "brand": "Mad Hippie",  
  "name": "Antioxidant Facial Oil",  
  "image": "https://images.ulta.com/is/image/Ulta/2530018",  
  "productID": "2530018",  
  "offers":  
    { "@type": "Offer",  
      "availability": "http://schema.org/InStock",  
      "price": "24.99",  
      "priceCurrency": "USD" } }
```

WITH Meaning


Massive e-commerce ontology – schema.org

AggregateRating

[Thing](#) > [Intangible](#) > [Rating](#) > [AggregateRating](#)

The average rating based on multiple ratings or reviews.

[more...]

- Canonical URL: <http://schema.org/AggregateRating>
- [Leave public feedback on this term](#) 
- [Check for open issues.](#)

Property	Expected Type	Description
Properties from AggregateRating		
itemReviewed	Thing	The item that is being reviewed/rated.
ratingCount	Integer	The count of total number of ratings.
reviewCount	Integer	The count of total number of reviews.
Properties from Rating		
author	Organization or Person	The author of this content or rating. Please note that author is special in that HTML 5 provides a special mechanism for indicating authorship via the rel tag. That is equivalent to this and may be used interchangeably.
bestRating	Number or Text	The highest value allowed in this rating system. If bestRating is omitted, 5 is assumed.
ratingExplanation	Text	A short explanation (e.g. one to two sentences) providing background context and other information that led to the conclusion expressed in the rating. This is particularly applicable to ratings associated with "fact check" markup using ClaimReview .
	Number or	The rating for the content



Demo JSON-LD in Python

- Based on crunch base data from early 2000 till 2014
 - Core objects: Investments, acquisitions, investors, companies
- For developers: how can you implement basic CRUD with AllegroGraph JSONLD
 - You can add and retrieve Python dictionaries directly
 - Like many other document databases
 - Objects are indexed with triples but can also be stored as blobs
 - You can retrieve parts of objects in a SPARQL queries
 - And you can retrieve as dictionaries.

Conclusion: JSON-LD and SHACL for Knowledge Graphs

- Make life easier for User Experience and Application Developers that need to work with Knowledge Graphs.
- JSON-LD hides complexity of semantics and graphs
- SHACL easy way to validate new data.

Thank you