

Ontology Editing and Creating Semantic Applications

Sixth Annual VIVO Conference

August 12-14, 2015

Boston, MA

The plan

- Welcome, introductions, and brief overview
- Basics of a semantic web application
- Installing the new Vitro runtime
- Examples -- SKOS, Gedcom
- Ideas and explorations as a group
 - Ideas and/or examples using Vitro
 - How do we apply these ideas to VIVO projects?
- Improving Vitro and VIVO as semantic web applications

Jon Corson-Rikert

WHAT IS THE SEMANTIC WEB AND WHAT IS LINKED DATA?

What is the Semantic Web?

A distributed body of interoperable data

- Cross-referenced via stable identifiers (URIs)
- Interpretable by machines
- Bearing human-understandable meaning
- Encoded in a simple standard format
- Resolvable using standard Web protocols

What is Linked Open Data?

Data

- Structured information, not just documents and text
- A common, simple format

Open

- Available, visible, mineable
- Anyone can post, consume, and reuse

Linked

- By direct reference
- Indirectly via common references and inference

Basic principles of linked data



- Use URIs as names for things
- Use HTTP URIs so people can look up those names
- Provide useful information from a URI in a standard format
- Include links to other URIs

What is data?

Information about *things* of interest to you

- What they are
 - Including different types of things
- Attributes about them
 - Including distinct characteristics that may only apply for certain types
- Relationships among things of the same or different types

Muhammad Javed

WHAT IS RDF AND WHAT IS AN ONTOLOGY?

Size of the current web

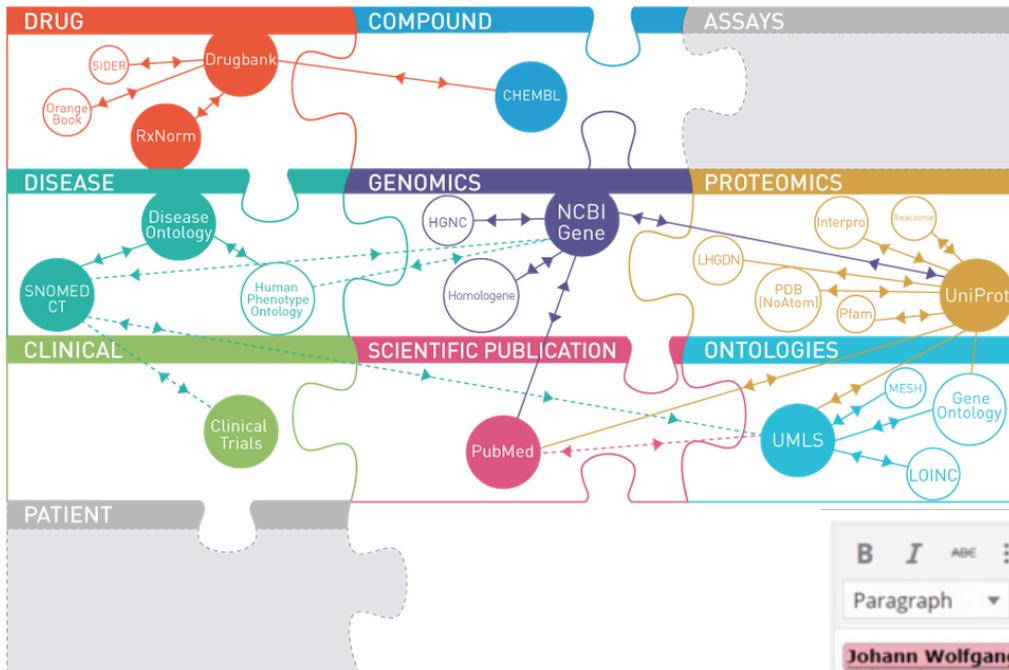


4.76 billion pages *(August, 11, 2015)*

Limitations of Current Web

- Too much information with too little incorporated meanings.
 - Data vs. String
- Made for human consumption.
 - Human can derive new information from given piece of data but not the machines)
- Content is heterogeneous
 - In terms of languages
 - In terms of structure

- Require open standards for describing information on the web.



<http://ontotext.com/insights-platform/linked-life-data/>

Rich text editor interface showing a text document with a context menu open over a highlighted name.

Johann Wolfgang von Goethe was a German writer, pictorial artist, biologist, theoretical physicist, and polymath. He is considered the supreme genius of modern German literature. His works span the fields of poetry, drama, prose, philosophy, and science.

Goethe studied law in University of Leiznin from 1765 to 1768. He detested learning age-old judicial rules by heart, preferring instead to... In Leipzig, Goethe fell in love with Kathchen Sch... genre.

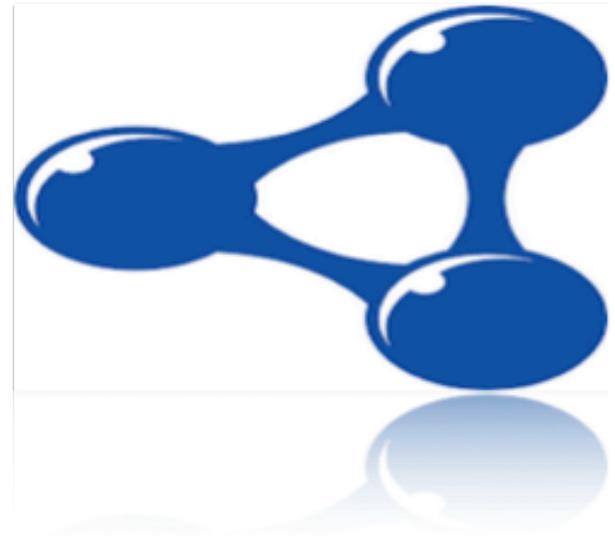
The restaurant Auerbachs Keller and its legend of Faust... Auerbachs Keller became the only real place in his close progress, Goethe was forced to return to Frankfurt at... on the nineteenth century.

The Goethe Institute is a non-profit German cultural as... of the German language abroad and encouraging inter... Institut fosters knowledge about Germany by providi...

Context Menu:

- Add as Entity
 - News Article
 - Organization
 - Person
 - Place
 - Postal Address
 - More...

RDF



WHAT IS RDF?

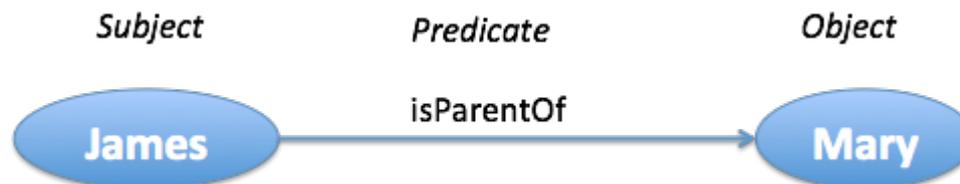
WHAT IS RDF?

- Resource Description Framework
- Use URIs to represent the resource and the relationship among them.

<http://vitro.cornell.edu/individual/james>

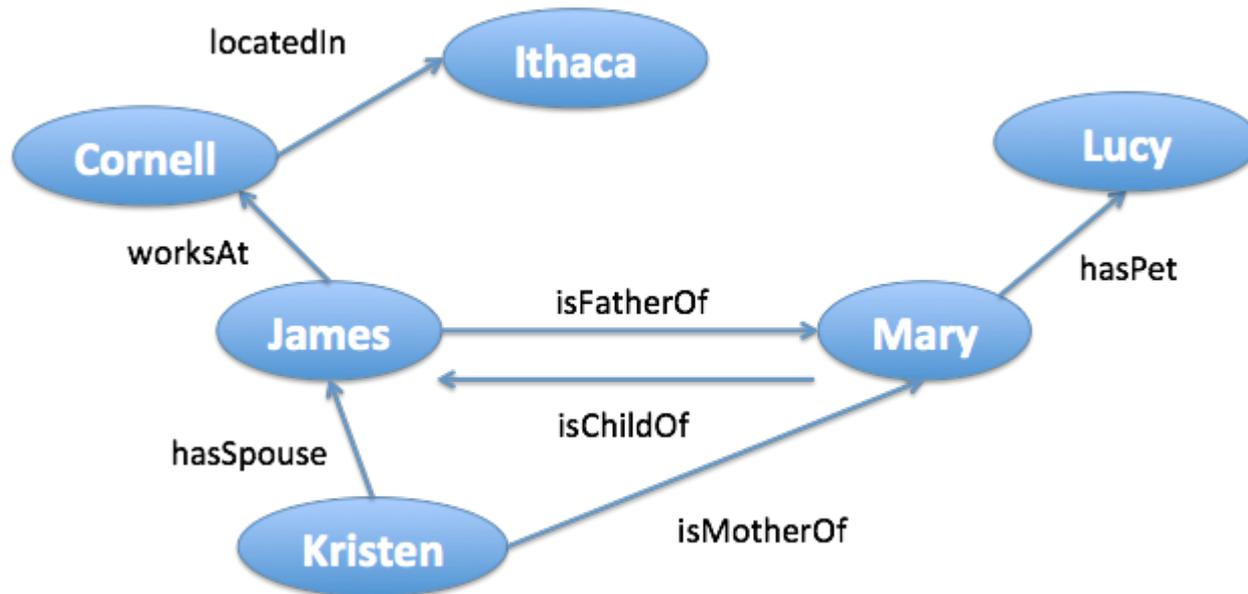
<http://vitro.cornell.edu/individual/isParentOf>

- Use triple format. <http://vitro.cornell.edu/individual/mary>



WHAT IS RDF?

- Linkages of triples creates a “Labeled Directed Graph”.



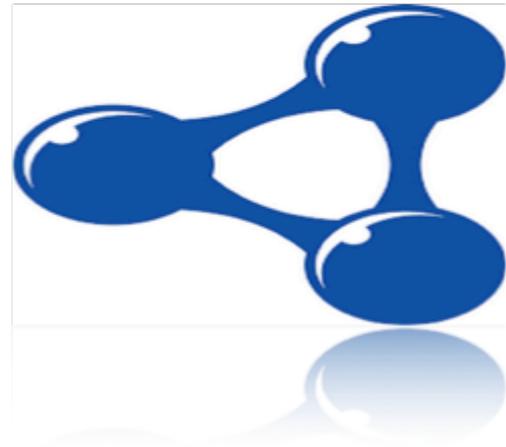
BENEFITS OF RDF

- Benefits of defining everything as resource:
 - Facilitate data merging from different sources even if underlying schema differ.
 - Supports schema evolution without requiring all data to be changed.

SEMANTICS OF RDF ASSERTIONS

- RDF provides ways to write assertions about URIs.
- But, what about semantics of such assertions? e.g. specifying that
 - A person cannot have more than one value for property “hasAge”
 - Two URIs <http://vitro.cornell.edu/Tony> and <http://vitro.cornell.edu/Anthony> represent the same thing.
 - Two properties (such as [isParentOf](#) and [isChildOf](#)) are inverse properties.

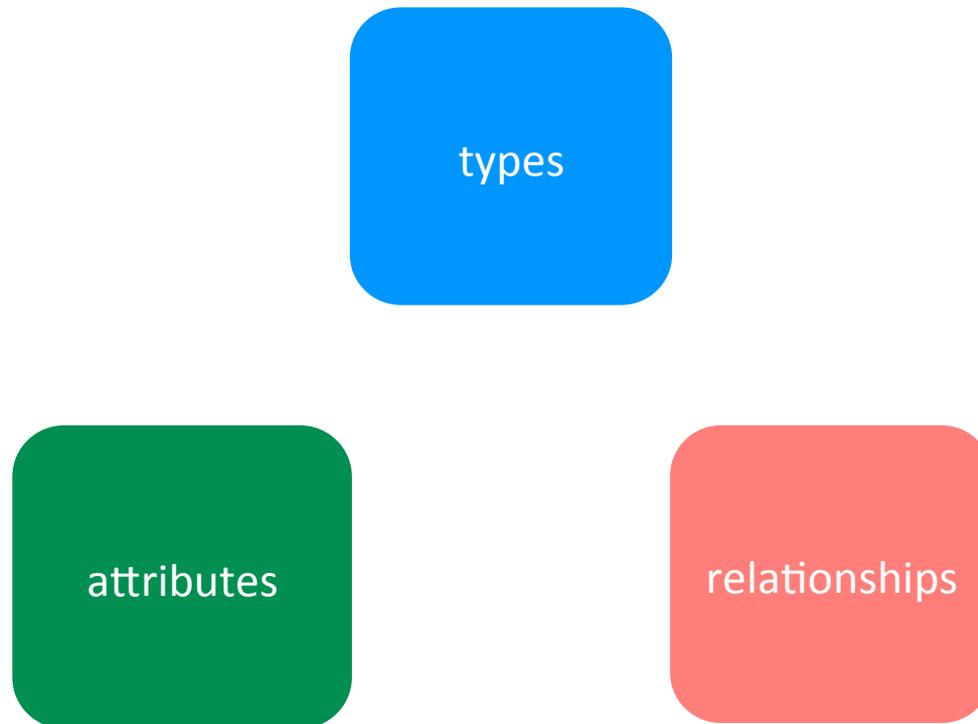
ONTOLOGY (OWL)



WHAT IS ONTOLOGY?

What is an ontology?

- A set of representational primitives with which to model a domain of knowledge or discourse (Tom Gruber, 2007)

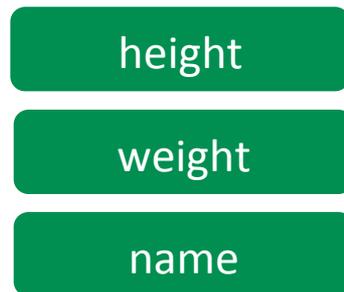


Dog ontology elements

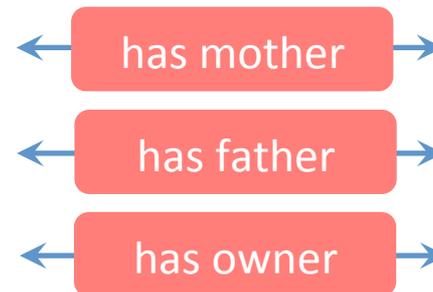
types of dogs



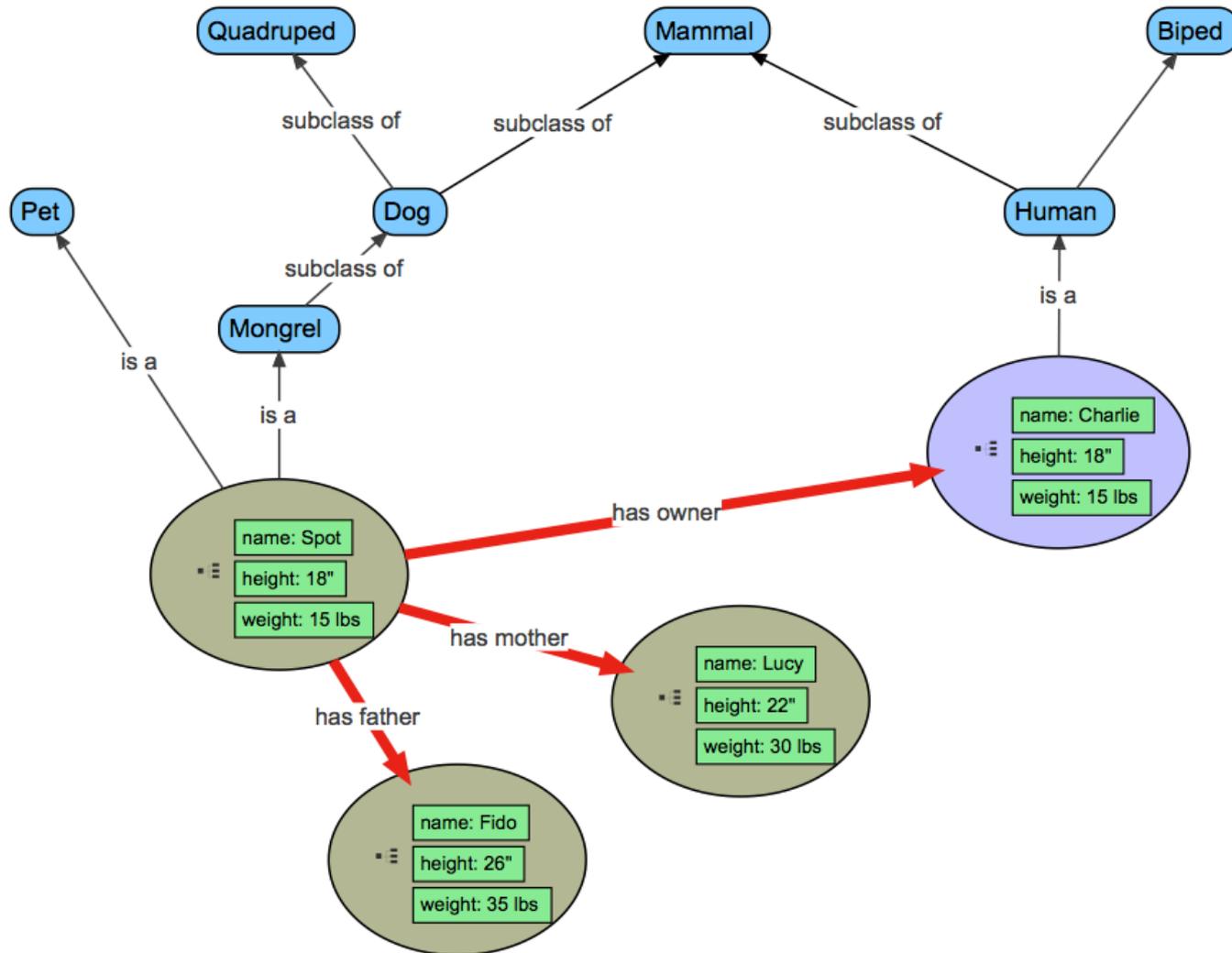
attributes dogs have



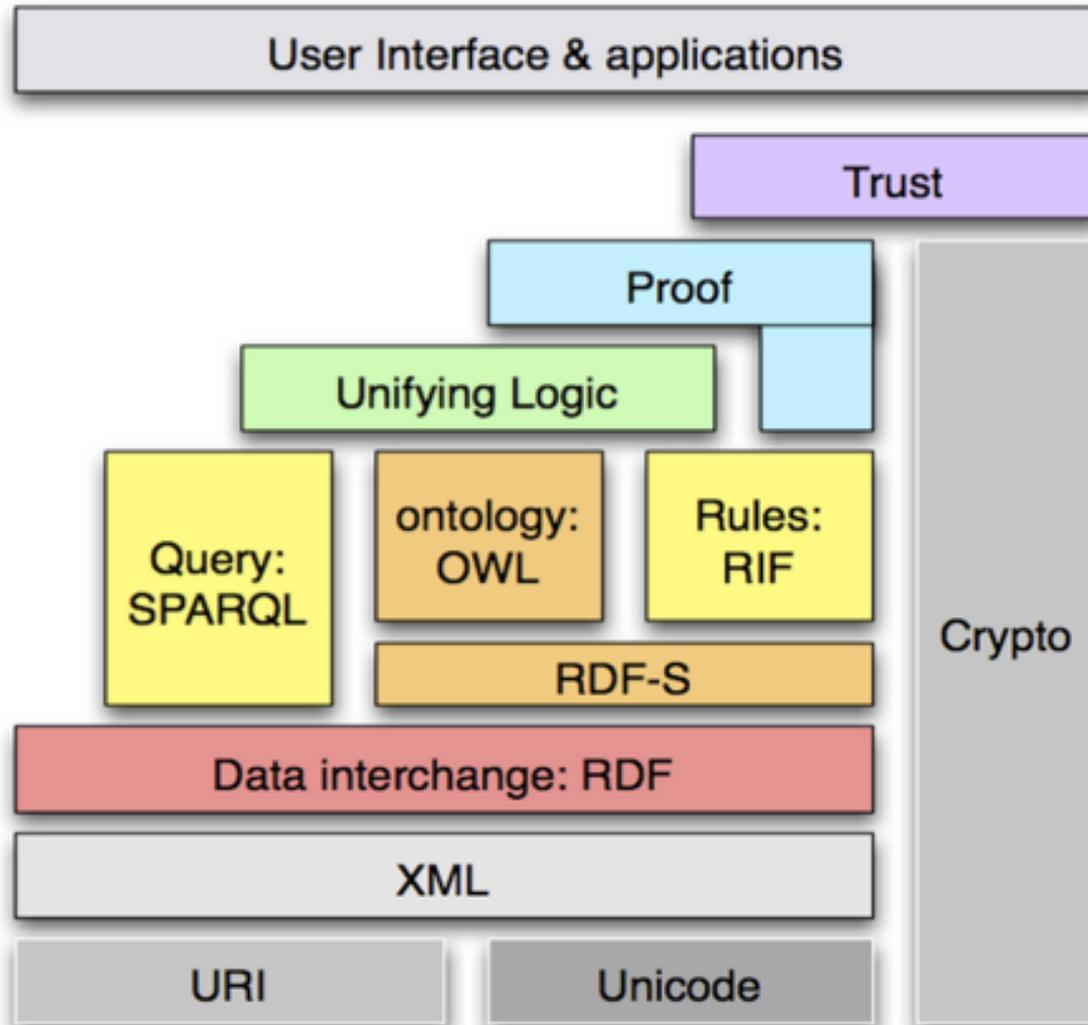
relationships dogs have



Populating the dog ontology



SEMANTIC WEB STACK



Vocabulary of OWL

- OWL offers additional vocabulary for expressing the semantics of the defined assertions.

owl:functionalProperty

owl:equivalentClass

owl:disjointClass

owl:intersectionOf

owl:sameAs

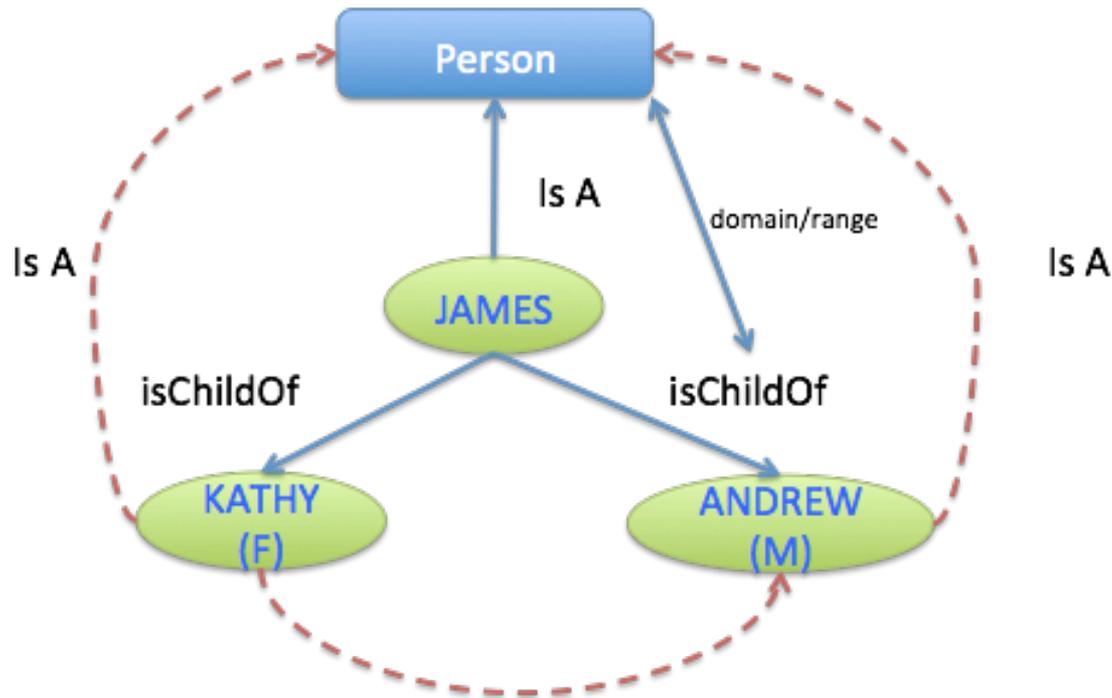
owl:unionOf

owl:symmetricProperty

owl:differentFrom

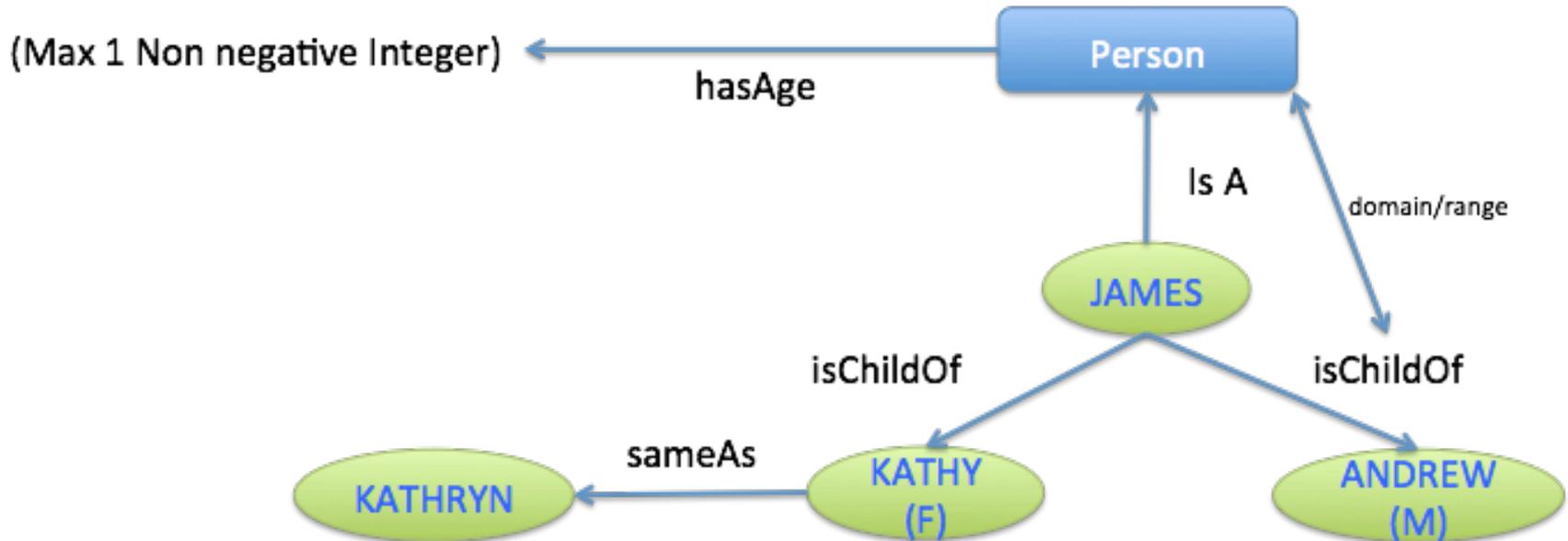
...

Inference/Reasoning in OWL



- If **X** isChildOf **Y** and **Y** hasSex 'M' => **Y** isFatherOf **X** **Y** is A Person
- If **X** isChildOf **Y** and **Y** hasSex 'F' => **Y** isMotherOf **X** **Y** is A Person
- If **X** isChildOf **Y** and **X** isChildOf **Z** => **Y** isSpouseOf **Z** **Z** isSpouseOf **Y**

Data Validation in OWL



- **KATHY** hasAge 35 => **OK**
- **KATHRYN** hasAge 41 => **Multiple values for the property hasAge**
- **ANDREW** hasAge -38 => **Negative value is not permissible**
- **JAMES** hasAge 13 => **OK**

Rebecca Younes

WHAT IS A SEMANTIC WEB APPLICATION?

What is a web application?

For our purposes today:

- Software accessed via the Web with a standard browser
 - In this case most of you will be running on a local web server on your own machines
- Software capable of entering, storing, and retrieving data
- Software accessible to end users for functions such as browsing and searching

What is a semantic web application?

- An application that consumes, displays, modifies, and outputs semantic web data
- *Hopefully*, an application that leverages semantic web technologies natively and not just via input and output
- *Ideally*, an application that leverages unique semantic web capabilities such as reasoning

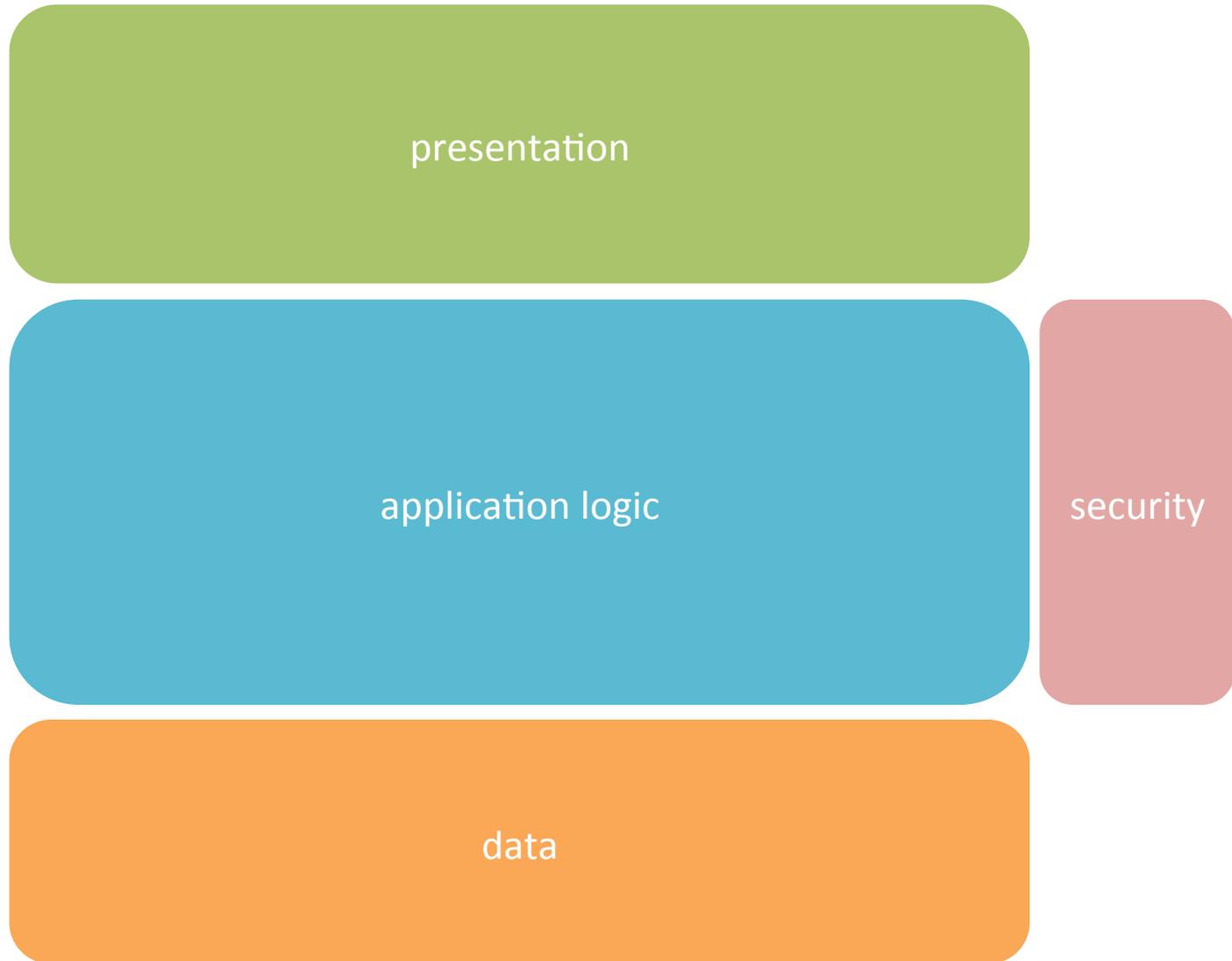
What is an ontology-driven application?

- Uses an ontology as the basis for its (semantic) data model
- Changes in response to changes in the underlying ontology it uses, with minimal reprogramming
- Successfully balances demands for data modeling with requirements for presentation and user interaction

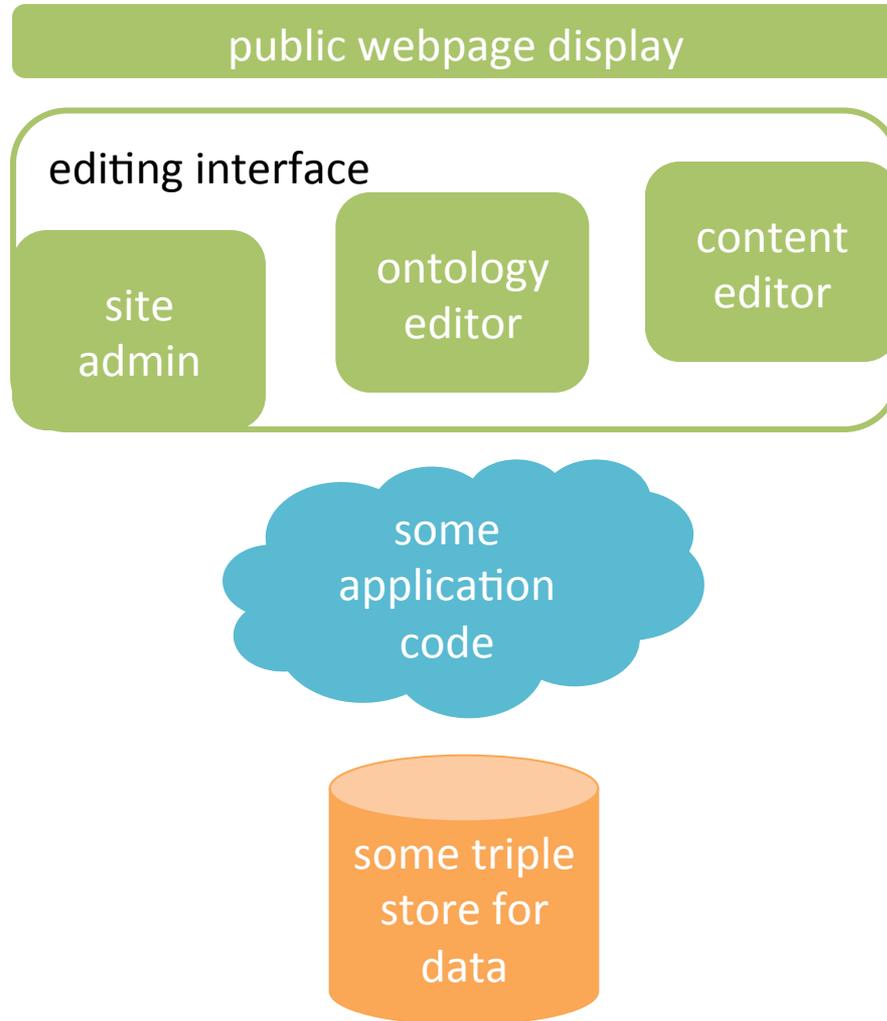
Vitro as ontology-driven application

- Starts out with a very minimal data model and no content
- You as application owner create or import an ontology
- You enter or import data
- You iterate to improve your ontology or add new imported ontologies
- You configure display and interaction as far as possible without coding
- You consider further customizations that may require coding

Typical application architecture



All you need to know about Vitro architecture today



Jim Blake

INSTALLING VITRO

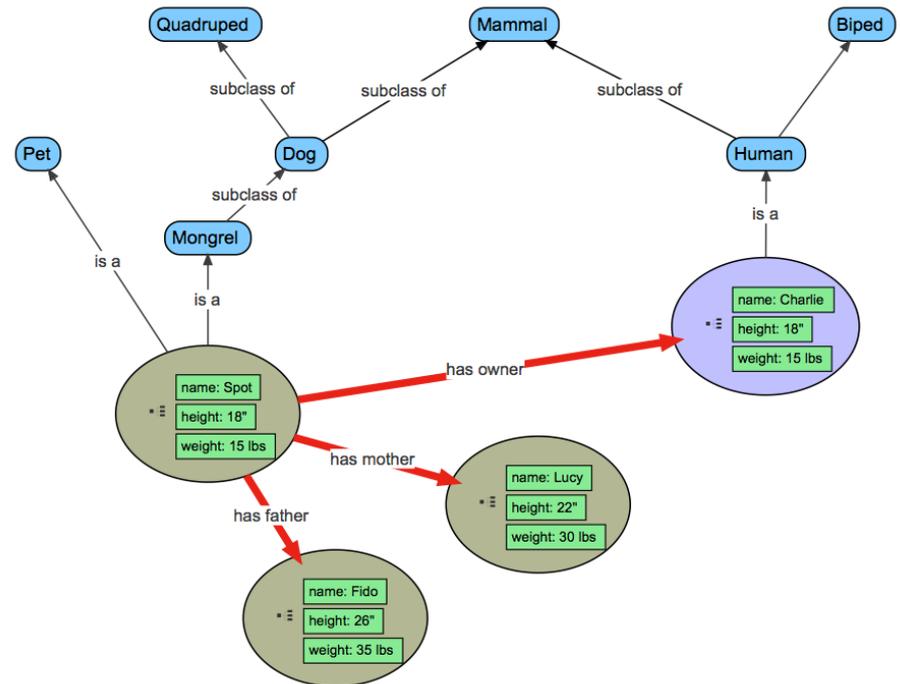
COFFEE BREAK

Rebecca Younes

EXPLORING AN EMPTY VITRO

Exploring Vitro

- Add some classgroups
- Add some classes
- Add data properties
- Enter 1-3 individuals, using different classes
- Add object properties
- Link up your data



Dressing up Vitro

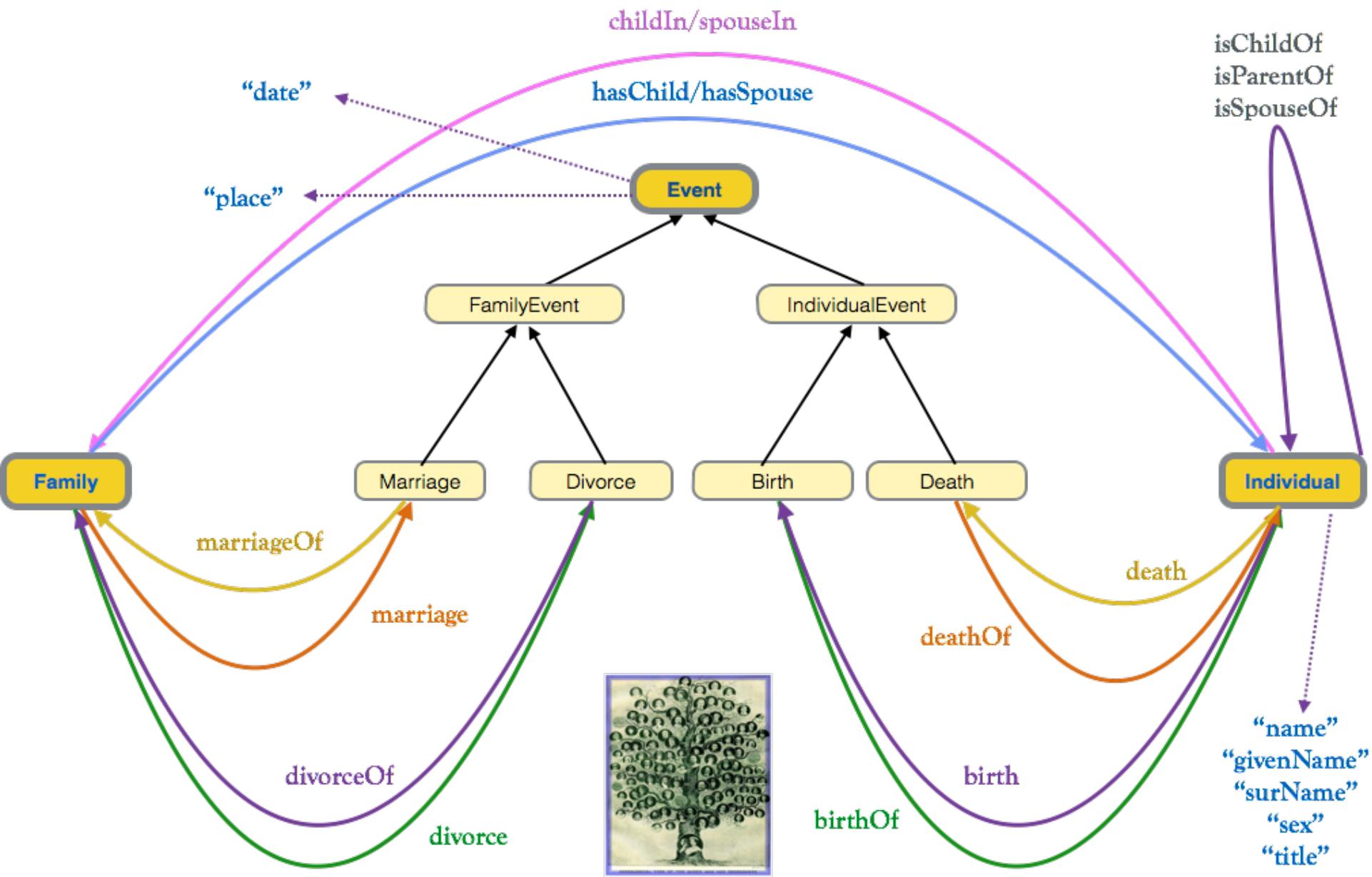
- Add menu pages
- Add more properties
- Add property groups

Rebecca Younes

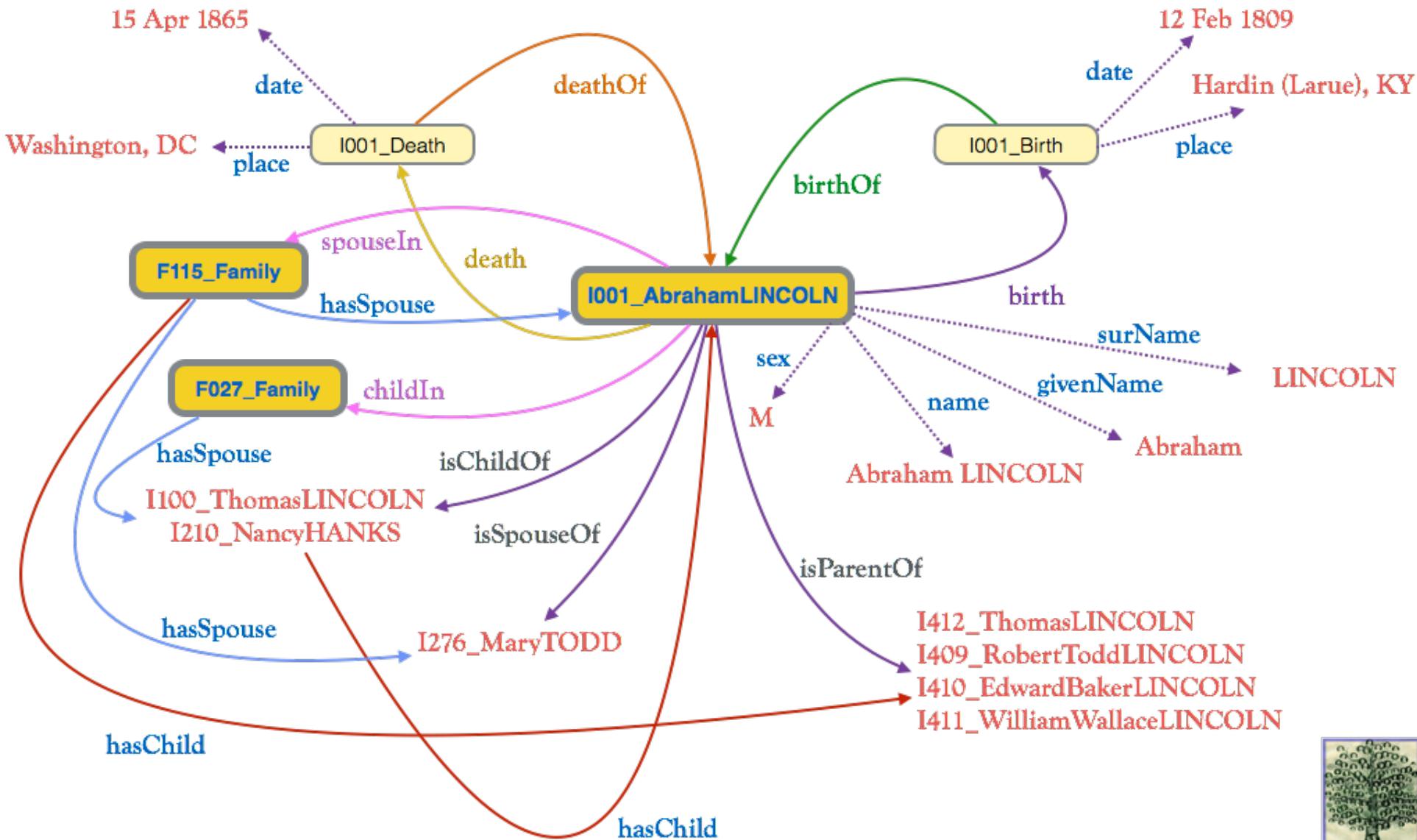
SKOS VITRO

Muhammad Javed

GEDCOM VITRO (BASIC)



Gedcom Vitro



Rebecca Younes

GEDCOM VITRO WITH CUSTOMIZATIONS

Vitro customizations

- Goals
 - Create a usable and meaningful public-facing user interface
 - Allow data curators to manage data without understanding all the complexities of the ontology
- How?
 - Add application configuration on top of the data model

Types of customizations

- Menus and pages
- Classgroups created
- Classes assigned to classgroups
- Property groups created
- Properties assigned to and ordered within property groups
- Meaningful labels for properties, classes, and groups
- Remove some classes from search, display, instantiation

Custom list views

- Motivation: hide the intricacies of the data model to display the data in a way that is most useful and meaningful to users

Example: gedcom:birth

- Data model provides for representation of event data such as date and place
- Gedcom model of a person's birth:
 - Person has-birth Birth
 - Birth has-date Date
 - Birth has-place Place
- Birth object should be hidden from users

Example: gedcom:marriage

- Family has-marriage Marriage
- Family has-spouse Person (x 2)
- Person has-name name (x 2)
- Marriage has-date Date
- Marriage has-place Place

What's important in a marriage?

- Decide what to display based on what users want to know
 - Family has-marriage Marriage
 - Family has-spouse Person (x 2)
 - Person has-name **name (x 2)**
 - Marriage has-date **date**
 - Marriage has-place **place**

Building a custom list view

- Register the custom list view in the application

`<vitrohome>/rdf/display/everytime/gedcomListViewConfig.rdf`

- Create the XML configuration file: SPARQL queries and template file
- Create the template

`<vitrohome>/themes/<themename>/templates/propStatement-marriage.ftl`

Custom forms

- Hide the complexities of the data model from data curators
- Components:
 - Java form generator class
 - Template
 - Register the form generator
- Example: entering a person's birth

Custom forms: next generation

- Form generators follow a general pattern, with variations
- Build a generic form generator class that reads configuration files to generate individual forms
- Same evolution as custom list views
- Would allow adding or updating forms without programming and rebuild

All

IDEAS FOR AN ONTOLOGY-DRIVEN APPLICATION

Local foods Vitro example



Semantic characteristics

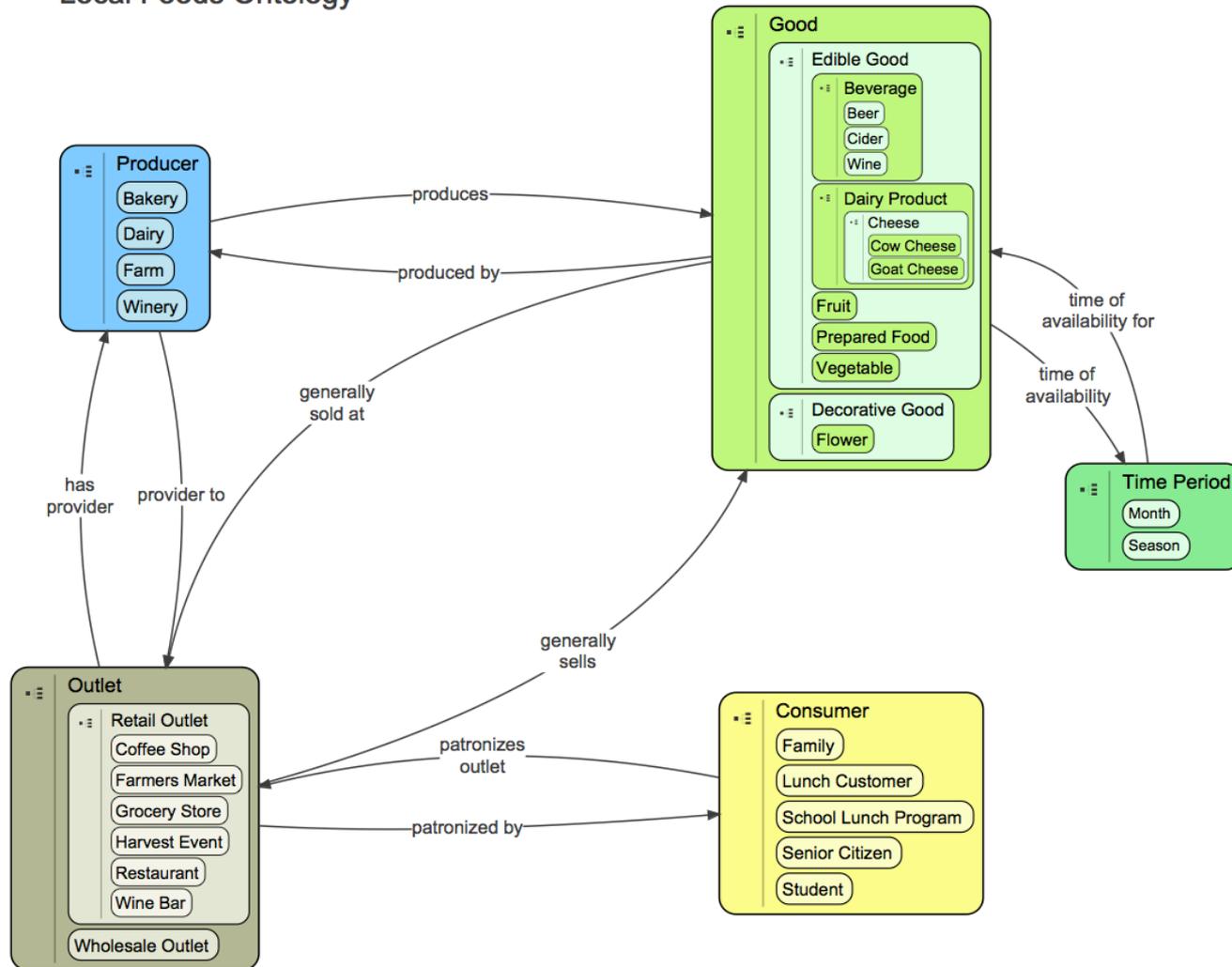
Illustrates the need to access the same information from multiple points of view

- Grower/producer
- Retailer
- Consumer

Interesting interplay of the abstract and the particular

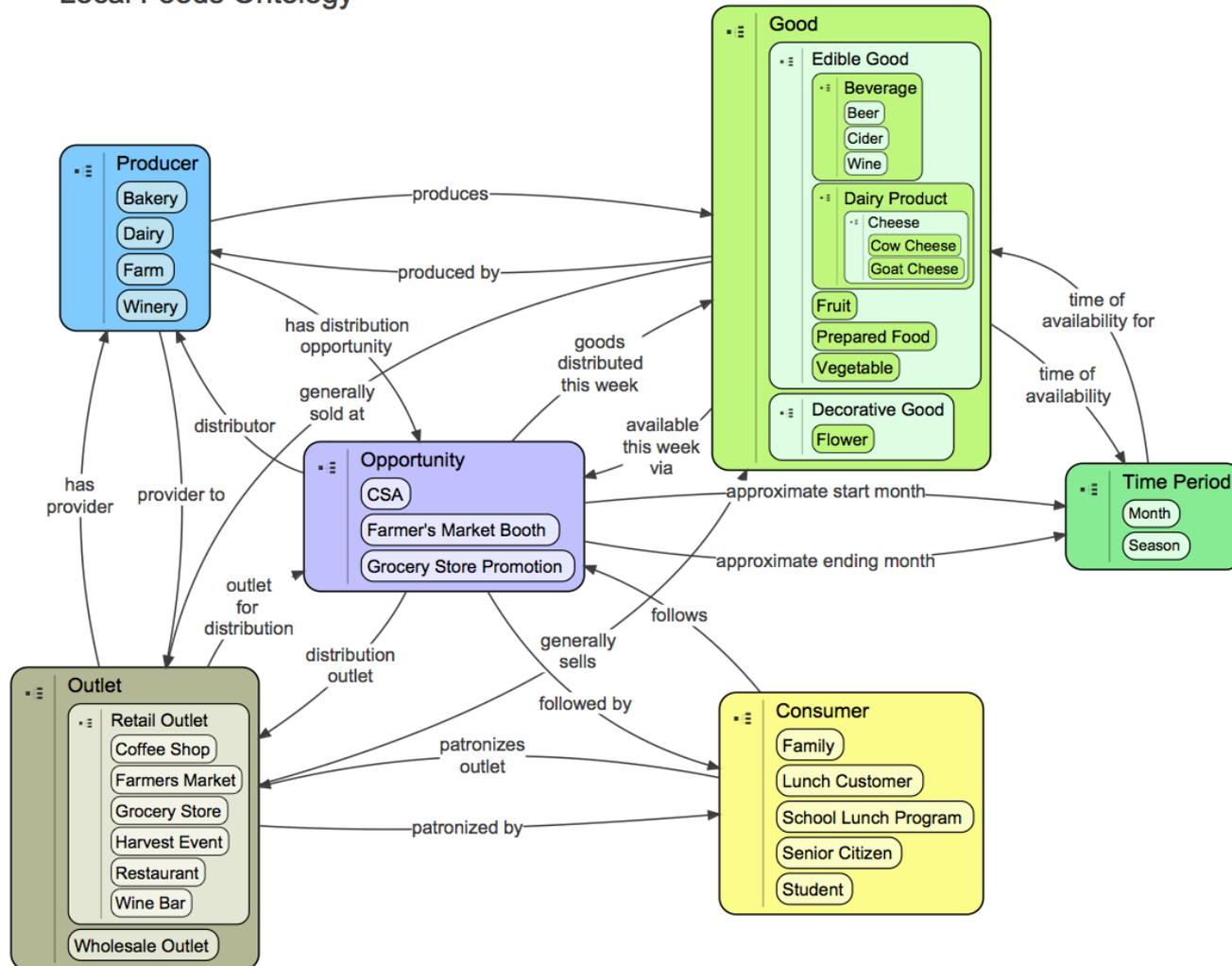
Basic classes and relationships

Local Foods Ontology



Adding the composite Opportunity class

Local Foods Ontology



User stories Vitro example

- Developed for the Linked Data for Libraries project, a collaboration among the Cornell, Harvard, and Stanford libraries
- Provided some value during the process of developing user stories
- Final use cases are presented on the project wiki
 - <https://wiki.duraspace.org/display/ld4l/LD4L+Use+Cases>

Story overview

01 Research guided by community usage | Active User Story [🔗](#)

Overview Related Comments View All

as a

[reference librarian](#)

[researcher](#)

in the course of

exploring a new field (researcher)

I would like to

find what is being used (read, annotated, bought by libraries, etc.) by the scholarly communities, not only at my institution but at others, and especially to find sources used elsewhere but not by my community

to gain the benefit that

a search for a subject or for a particular work makes suggestions for further exploration that are both relevant and surprising

votes

[01: highest \(dw\)](#)

Other views

Photo +

Admin Panel [Edit this individual](#)

Resource URI: <http://userstories.vivoweb.org/individual/n5654>

faculty member | [✎](#) | **User Type** [🔗](#)

Overview | Related | View All

protagonist in story +

- [02 Compose a syllabus](#) | [✎](#) [🗑️](#)
- [03 Build a virtual collection](#) | [✎](#) [🗑️](#)
- [28 Form-fill for faculty deposits to an IR](#) | [✎](#) [🗑️](#)
- [30 Find which works by an author are used in courses](#) | [✎](#) [🗑️](#)

Annotation

all 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

[collection level vs. item level](#)

[could do textual analysis perhaps](#)

[demonstrates scholarly communities learning from one another across institutions](#)

[heavily focused on the UI -- how central should that be to the use cases?](#)

[how do we geolocate items?](#)

[how will we align subject terms?](#)

[integrates person data -- good](#)

[LD enough?](#)

[maps are a gnarly type](#)

[may be more compelling if re-framed](#)

[out of scope](#)

course syllabi | [Concept](#) 🔗

Overview

concept applied to

active user story

- [02 Compose a syllabus](#)

data availability dependency

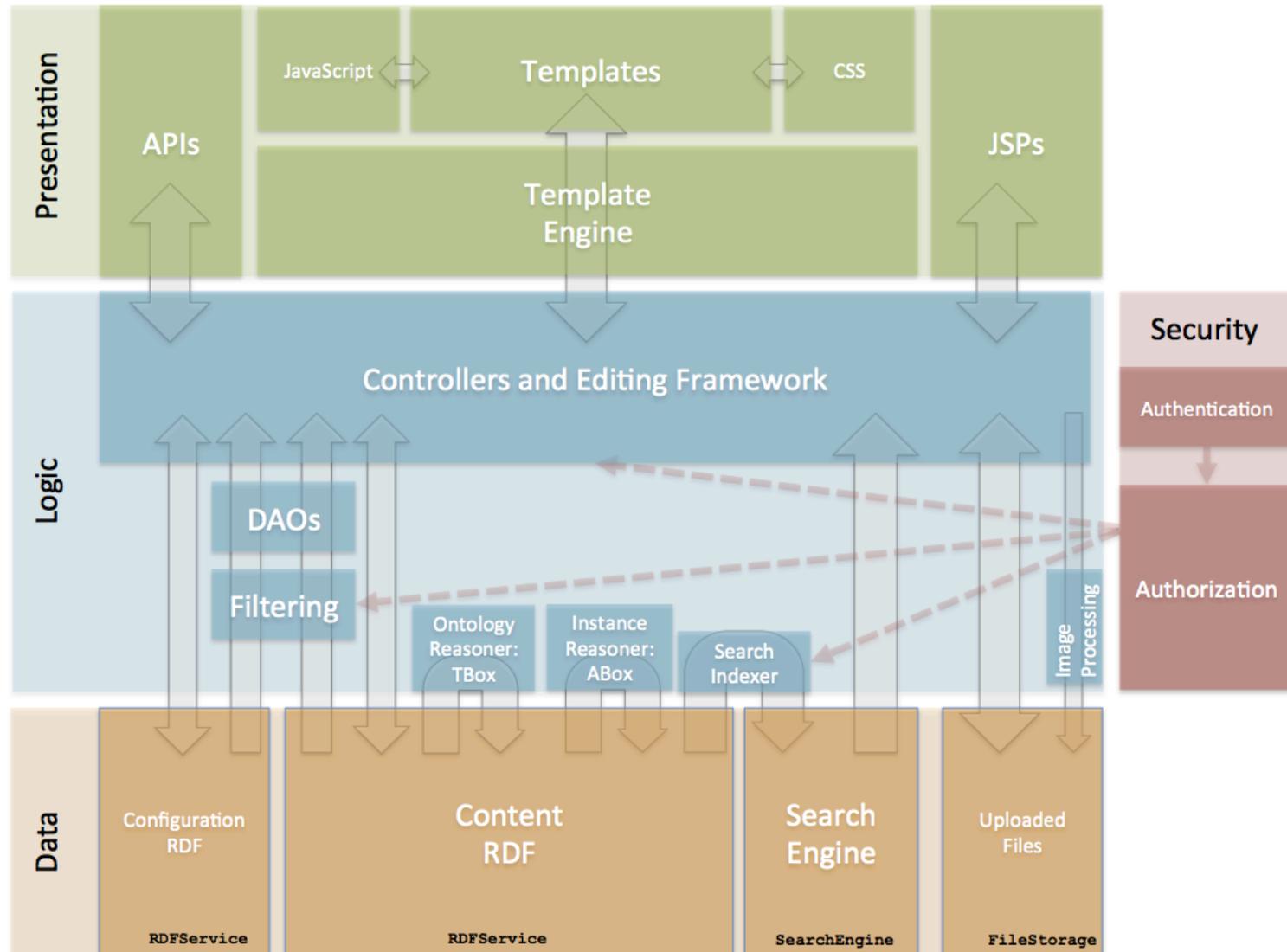
- [course information: courses taught, syllabi, and recommended/required readings](#)

TAKING VITRO FURTHER

Vitro vs. VIVO

	Vitro	VIVO
Purpose	General-purpose tool for working with Semantic Data	Specialized tool for Research Networking
Ontology	No ontology	Includes an ontology (VIVO-ISF) for Research Networking
Theme	Minimal theme	Elaborate theme, display and editing are customized for the ontology
Display Rules	Default display rules	Annotations are used to: <ul style="list-style-type: none">• Assign data properties to groups• Arrange property groups on the page
Form editing	Default editing forms	Editing is customized to the ontology
Search Index	Default search index	Search index contains additional fields specific to VIVO
Functionality	Default functionality	Additional functionality: visualizations, interface to Harvester, QR codes, etc.

Vitro-VIVO architecture



All

WRAP-UP AND QUESTIONS