Building Blocks for Distributed Knowledge Graphs

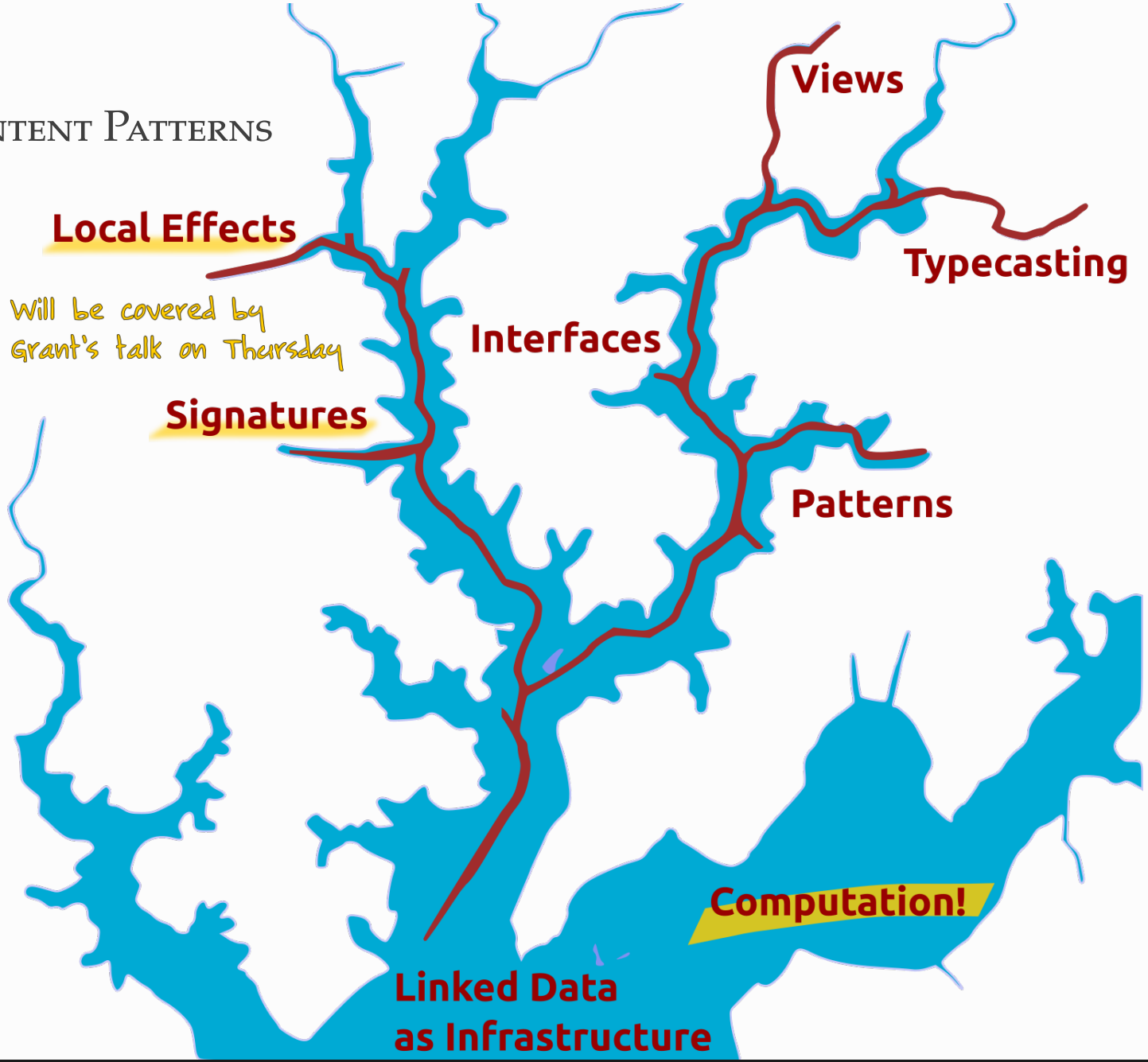
GeoVoCamp DC 2016

Krzysztof Janowicz

STKO Lab, University of California, Santa Barbara, USA

The Big Picture

More Than Just Content Patterns



What is Interoperability Anyway?

(Technical) Interoperability **measures the degree** to which different **systems** (e.g., Web Services/APIs) are able to work in concert to reach a common task.

Some (subtle) consequences

- Data do **not** inter-operate.
- Operating requires an **intension**, e.g., a goal.
- Interoperability is about a **degree** of meaningful exchange.

Syntactic interoperability relies on **common protocols and data formats** to ensure the proper exchange of data. We can ensure a **perfect** syntactic interoperability, e.g., via rigid standardization.

Semantic interoperability relies on a **common understanding** of the exchanged data, i.,e., **meaning** remains invariant during the exchange between multiple systems. This requires common **reference systems**.

What is Meaning Anyway?

Semantics is the study of meaning, i.e., the study of how and what **signifiers denote**. Meaning is an **emergent** property of interaction.

Humans use complex processes such as **situated simulation** to negotiate the **approximate**, common meaning of terms during interaction.

Unfortunately, we cannot do so with data.

The driving, often implied assumption behind the call for **raw** data is the idea that once collected data can be reused outside its original creation context and is free of interpretation.

However, there are no raw data. Data are always created following particular workflows, **procedures**, sampling strategies, are derived using specific instrumentation, are pre-processes in specific ways, and so forth.

One way to approach this problem are **ontologies**. However, they cannot fix meaning. Rather, they formally restrict the **possible interpretations** of domain terminology towards their intended meaning.

Consequences?

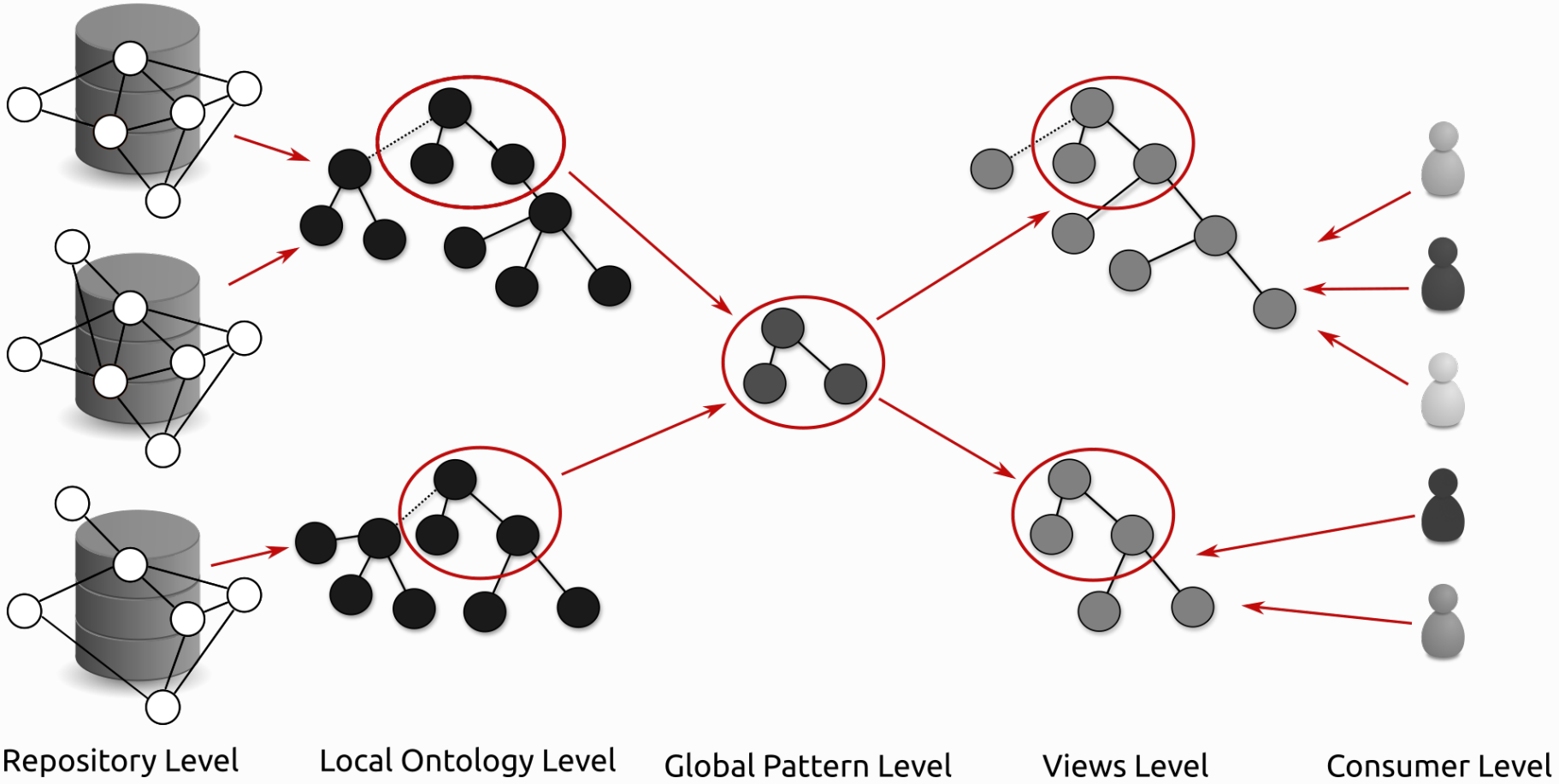
We can only measure semantic interoperability after the fact.

To move forward, we should measure our **progress in preventing** interoperability problems, not by trying to ensure semantic interoperability.

Our work should notify users that two datasets cannot be meaningfully combined or two systems will not meaningfully interact, instead of dreaming the **dream** of **fully-automated service orchestration** in heterogeneous environments.

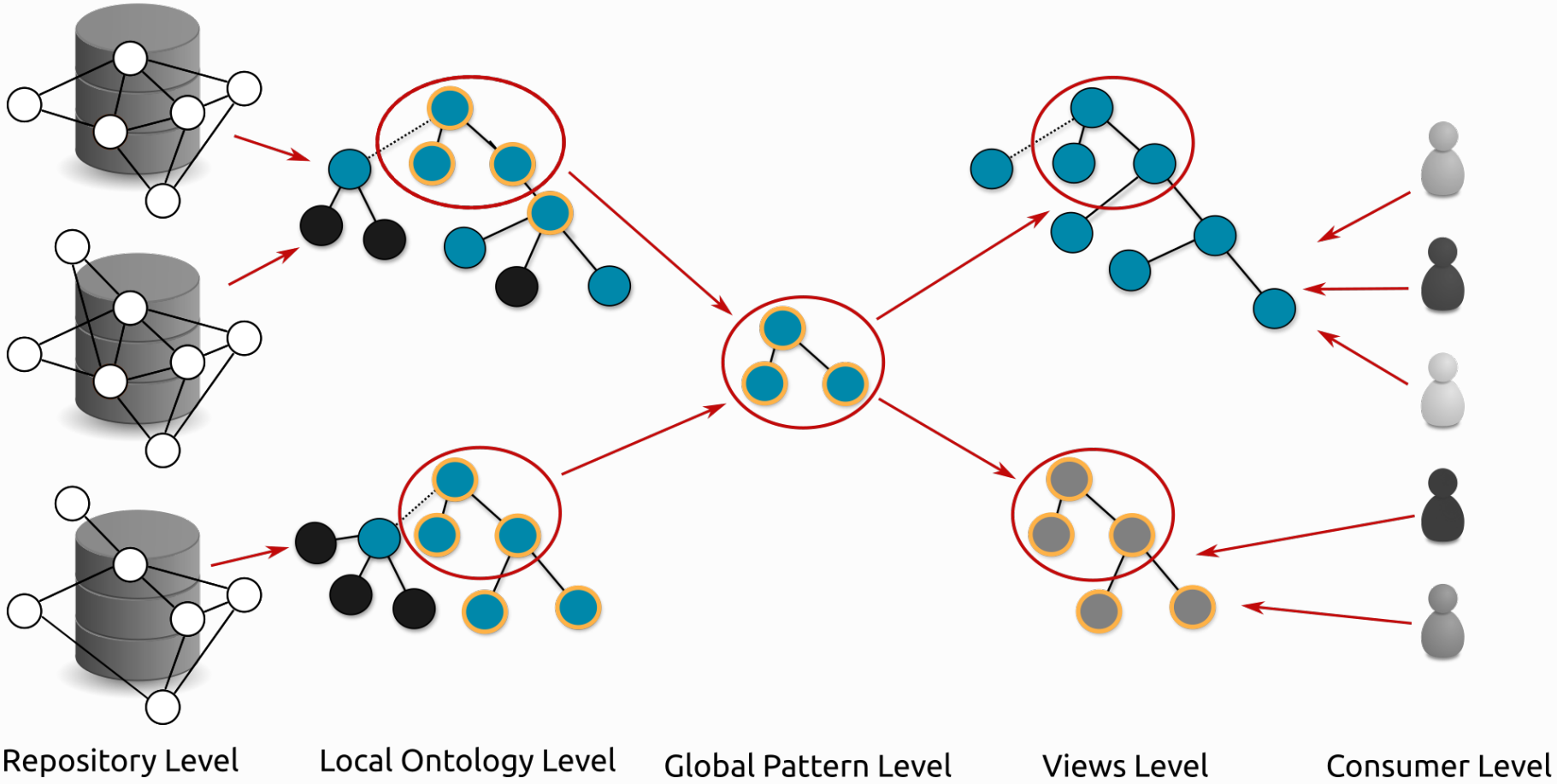
Example: If one application ontology models cars exclusively by fuel consumption and another ontology models cars exclusively by color, can data about cars be integrated across them?

Envisioned, Pattern-based Arcitecture (Horizontal)



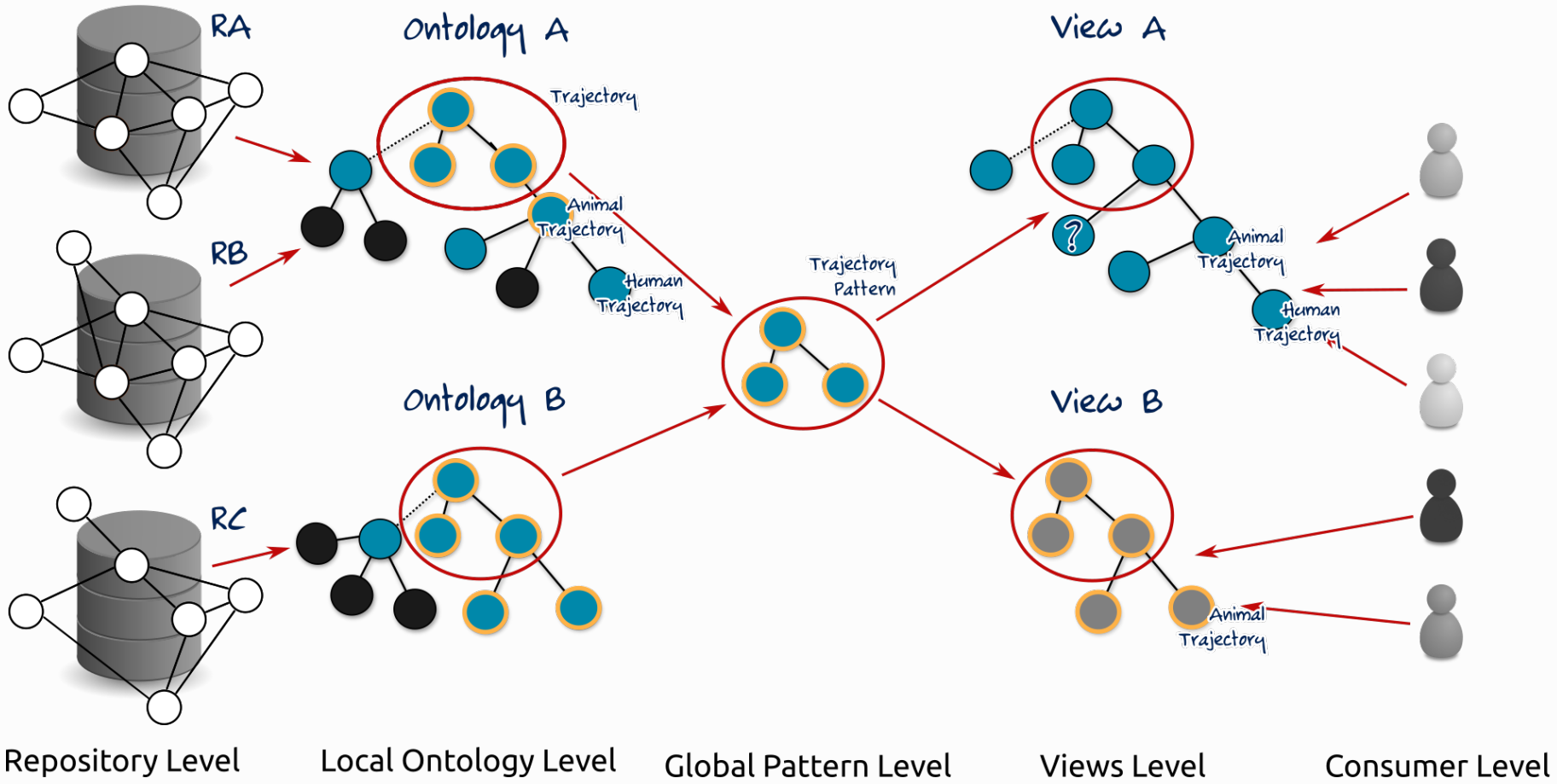
Patterns act as **fallback** level that ensures **minimal interoperability** while preserving **heterogeneity** (i.e., local, repository-specific ontologies can differ).

Envisioned, Pattern-based Arcitecture (Horizontal)



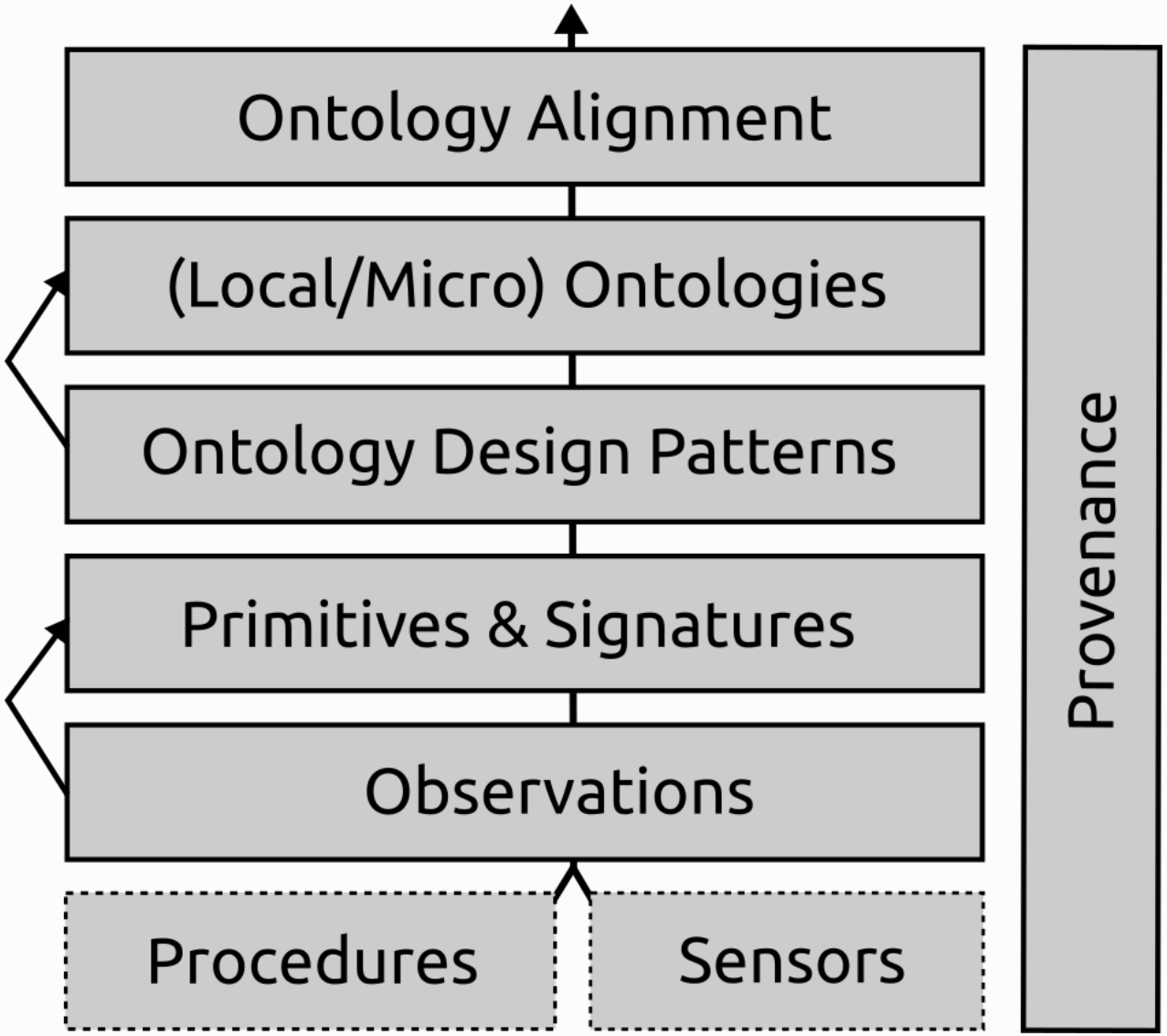
Views as virtual ontologies. 'All' provider- and user-perspectives agree on a common core; more specific results can differ.

Envisioned, Pattern-based Arcitecture (Horizontal)

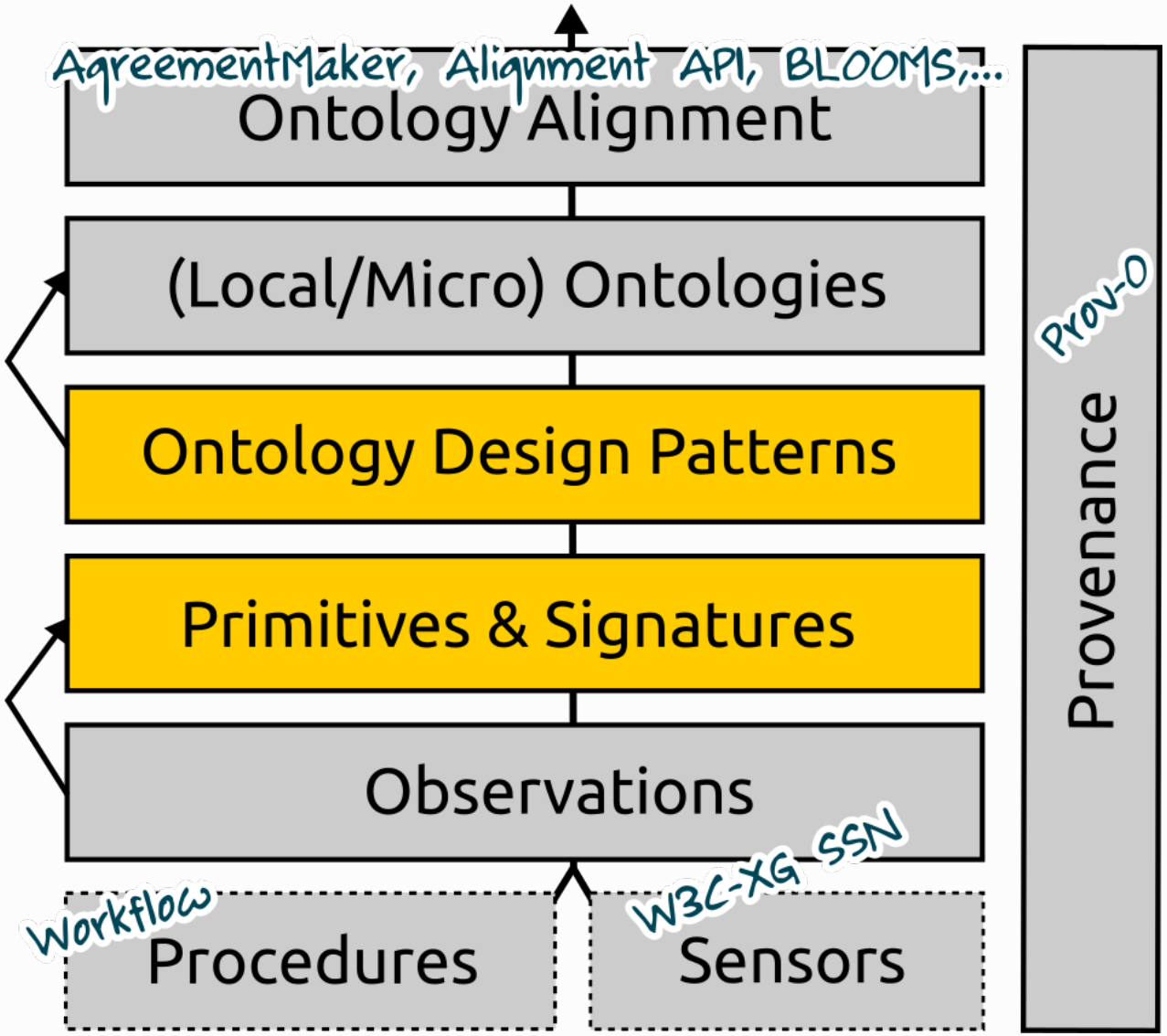


All users can query for data that correspond to the pattern, using view A one can retrieve data on human trajectories but these data will only come from RA and RB.

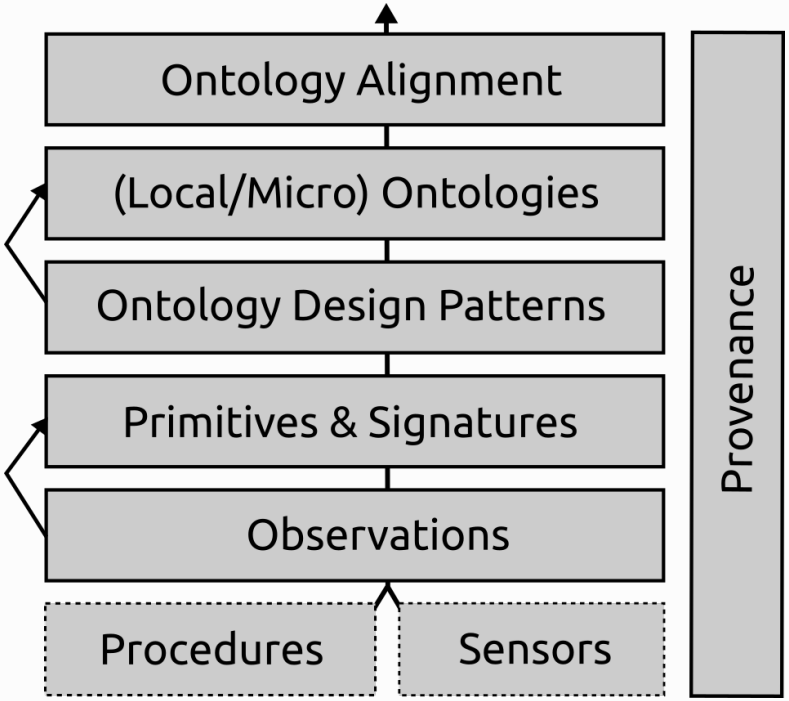
Envisioned, Pattern-based Arcitecture (Vertical)



Envisioned, Pattern-based Arcitecture (Vertical)



Envisioned, Pattern-based Arcitecture (Vertical)



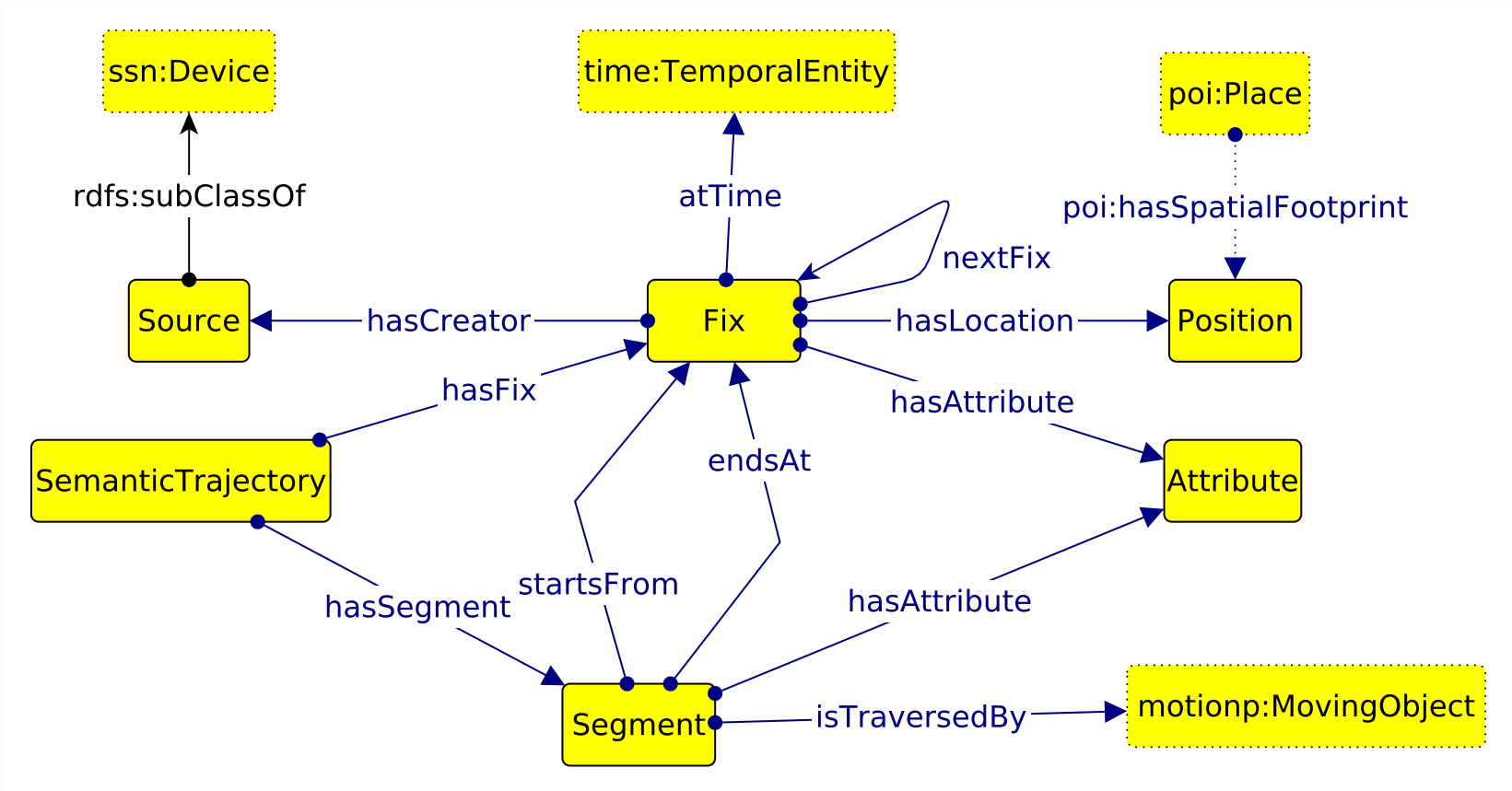
Cons

- *... speak now or forever hold your peace*

Pros

- **Defers** the introduction of classes that are heavy on ontological **commitments** (e.g., 'vulnerability')
- No need for (community-wide) **agreement** which is a key (social) challenge for other approaches. Preserves heterogeneity
- Mines ontological primitives out of real **observation data**
- Assists domain experts in becoming knowledge engineers by developing reusable **patterns**
- Moves ontology **reuse** to the layer where it belongs (and avoid *Frankenontologies*)
- Is driven by publishing, **discovery**, **reuse**, and integration needs.

A Semantic Trajectory Pattern



A pattern for **discrete** trajectories of people, wildlife, vessels, and so forth.

A Semantic Trajectory Pattern

$$\mathsf{Fix} \sqsubseteq \exists \mathsf{hasLocation}.\mathsf{Position} \sqcap \exists \mathsf{atTime}.\mathsf{time{:}TemporalEntity} \sqcap (=1\ \mathsf{hasFix}^{-}.\mathsf{Trajectory}) \sqcap (\leqslant 1\ \mathsf{nextFix}.\mathsf{Fix}) \sqcap \neg \exists \mathsf{nextFix}.\mathsf{Self}$$

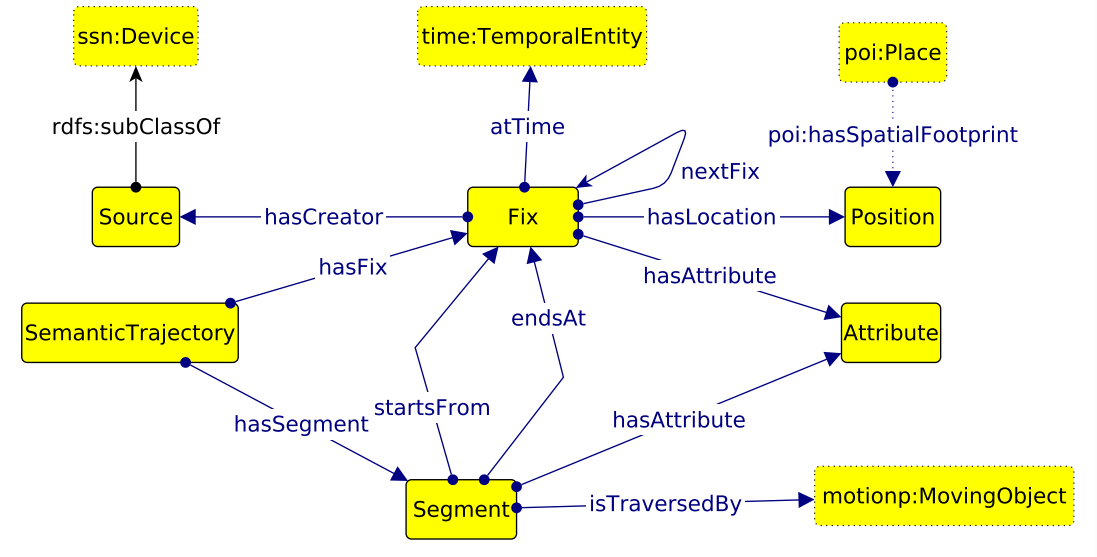
$$\mathsf{Segment} \sqsubseteq (=1\ \mathsf{startsFrom}.\mathsf{Fix}) \sqcap (=1\ \mathsf{endsAt}.\mathsf{Fix}) \sqcap (=1\ \mathsf{hasSegment}^{-}.\mathsf{Trajectory})$$

$$\exists \mathsf{nextFix}.\mathsf{Fix} \sqsubseteq (=1\ \mathsf{startsFrom}^{-}.\mathsf{Segment})$$

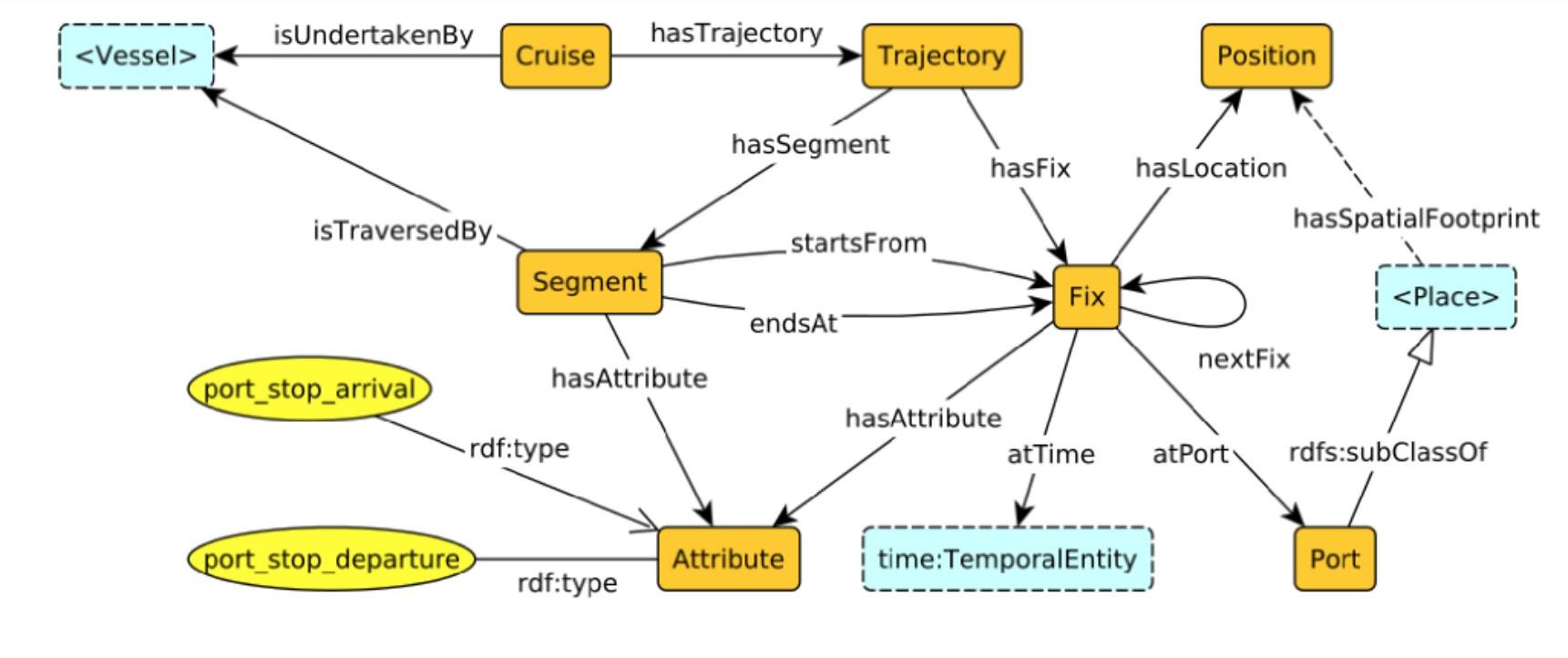
$$\exists \mathsf{nextFix}^{-}.\mathsf{Fix} \sqsubseteq (=1\ \mathsf{endsAt}^{-}.\mathsf{Segment})$$

$$\mathsf{startsFrom} \circ \mathsf{nextFix} \sqsubseteq \mathsf{endsAt}$$

$$\mathsf{hasFix} \circ \mathsf{startsFrom}^{-} \sqsubseteq \mathsf{hasSegment}$$

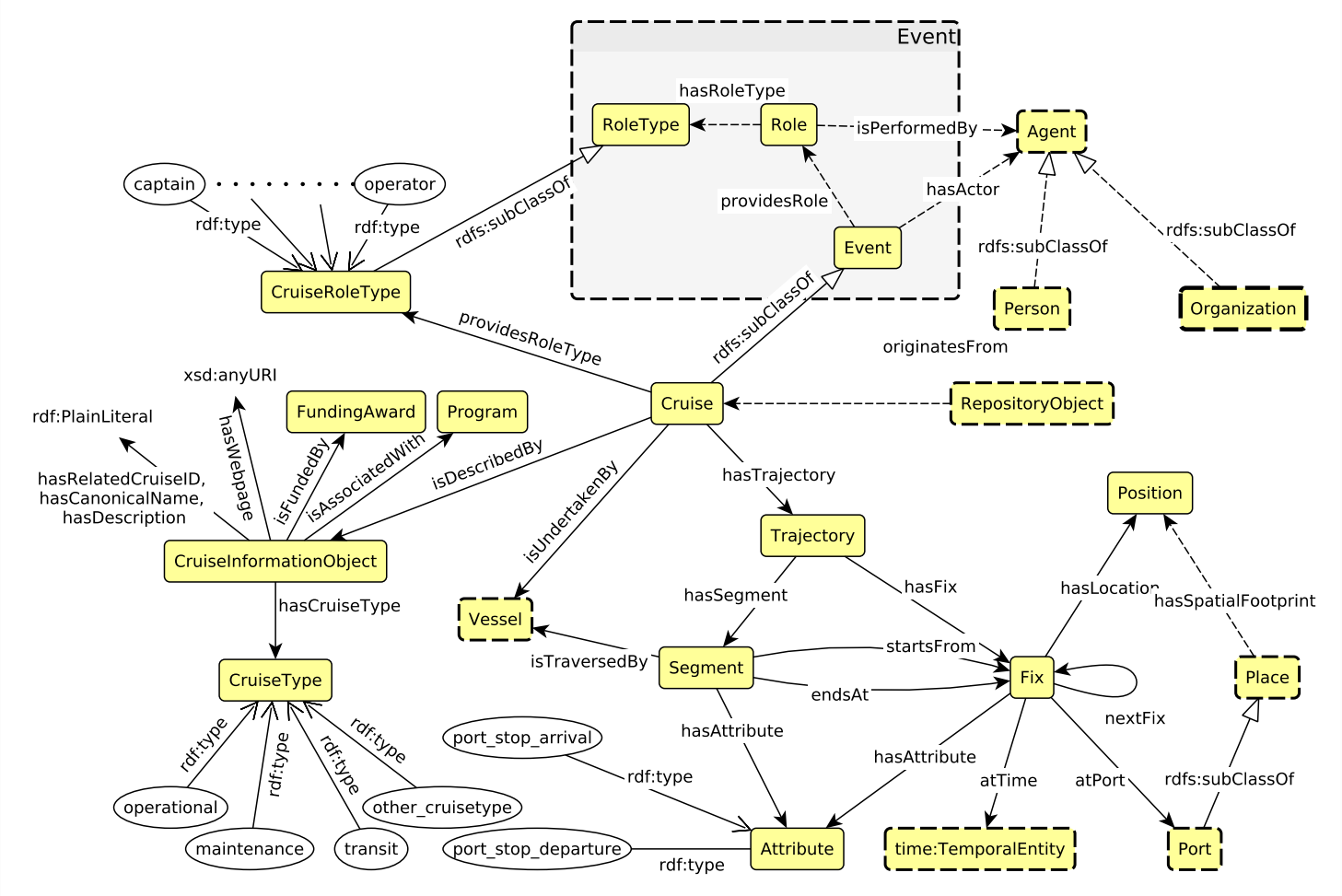


Ontology Design Patterns Can Be Specialized



- Trajectories that model the research **cruises** of scientific vessels
- Is this still an ontology design pattern?

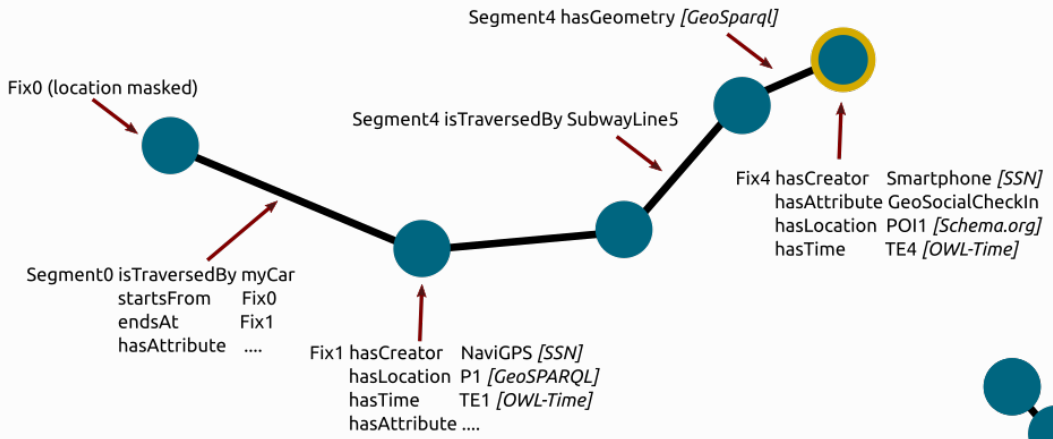
A Micro-Ontology for Cruises



Combining the InformationObject, Agent, Event, Vessel, and Trajectory patterns

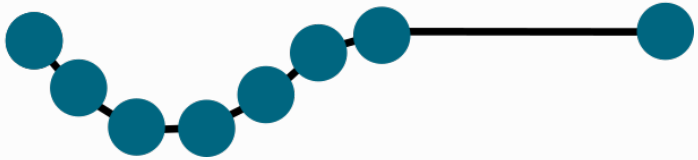
Idea Behind Semantic Trajectory Pattern

Human travel trajectory



Abstraction

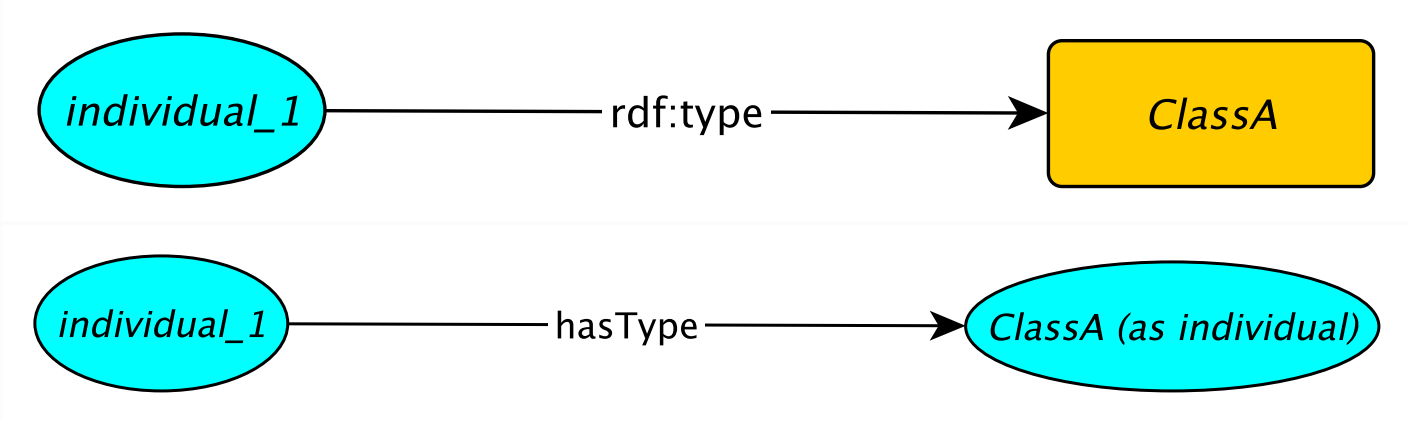
Discretization



- Can cover a wide range of **domains**
- Can be easily **extended**
- Supports multiple **granularities**
- **Axiomatization** beyond mere surface semantics
- Has various **hooks** to well-known ontologies / patterns.
- Only partially self-contained ☹

Typecasting – Dealing with Styles

Typecasting Individual to Class and Back



$$\text{ClassName} \sqsubseteq \exists \text{hasType}.\{\text{classname}\} \quad (1)$$

$$\exists \text{hasType}.\{\text{classname}\} \sqsubseteq \text{ClassName} \quad (2)$$

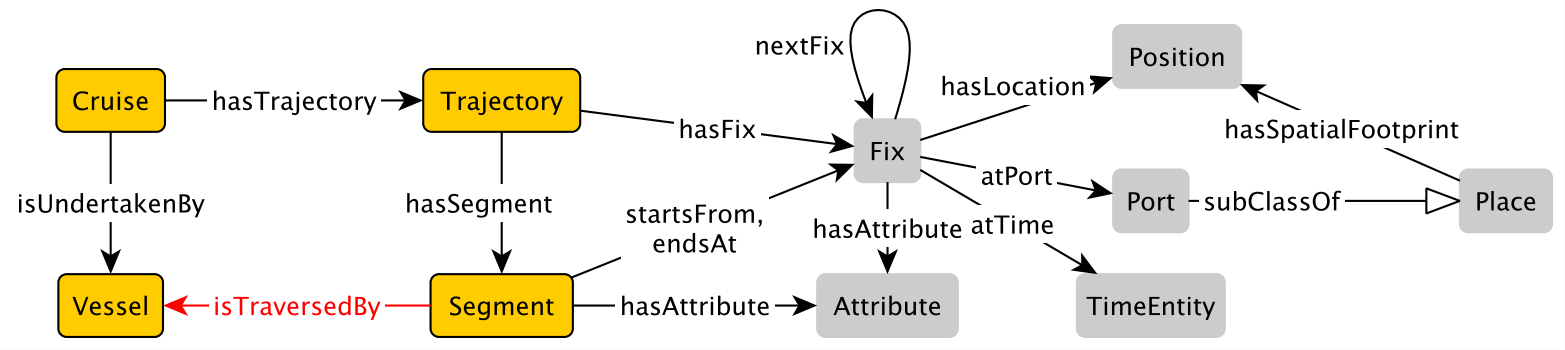
Rolification: Typecasting from Classes to Properties

...

Reification: Typecasting Properties into Classes

...

Views – Dealing with Granularity and Perspectives



$$\text{Vessel} \equiv \exists R_{\text{Vessel}}.\text{Self}$$

$$\text{Cruise} \equiv \exists R_{\text{Cruise}}.\text{Self}$$

$$\text{Trajectory} \equiv \exists R_{\text{Trajectory}}.\text{Self}$$

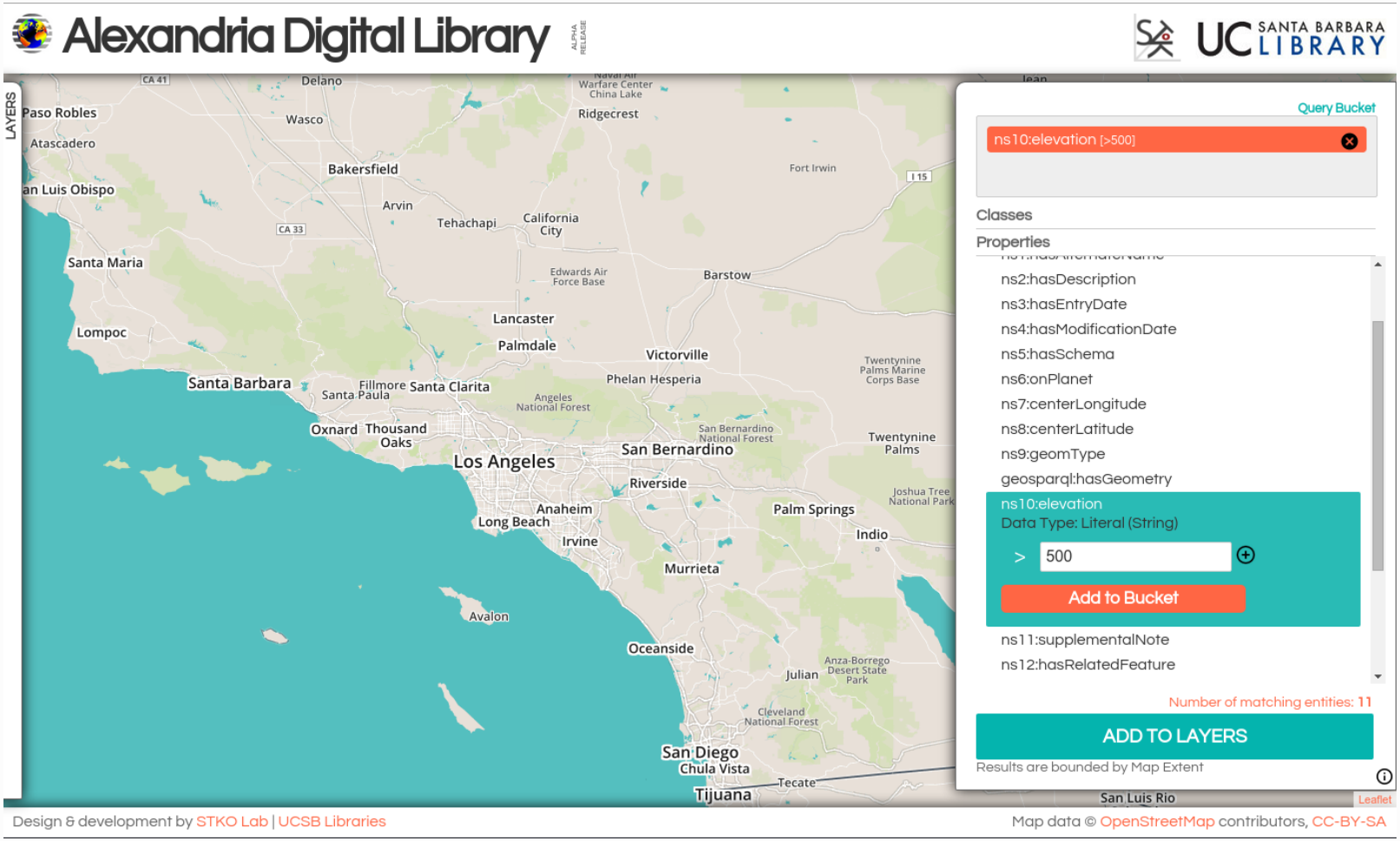
$$\text{Segment} \equiv \exists R_{\text{Segment}}.\text{Self}$$

$$R_{\text{Segment}} \circ \text{hasSegment}^{-} \circ R_{\text{Trajectory}} \circ \text{hasTrajectory}^{-} \circ R_{\text{Cruise}} \circ \text{isUndertakenBy} \sqsubseteq \text{isTraversedBy}$$

Ontologies as Interfaces

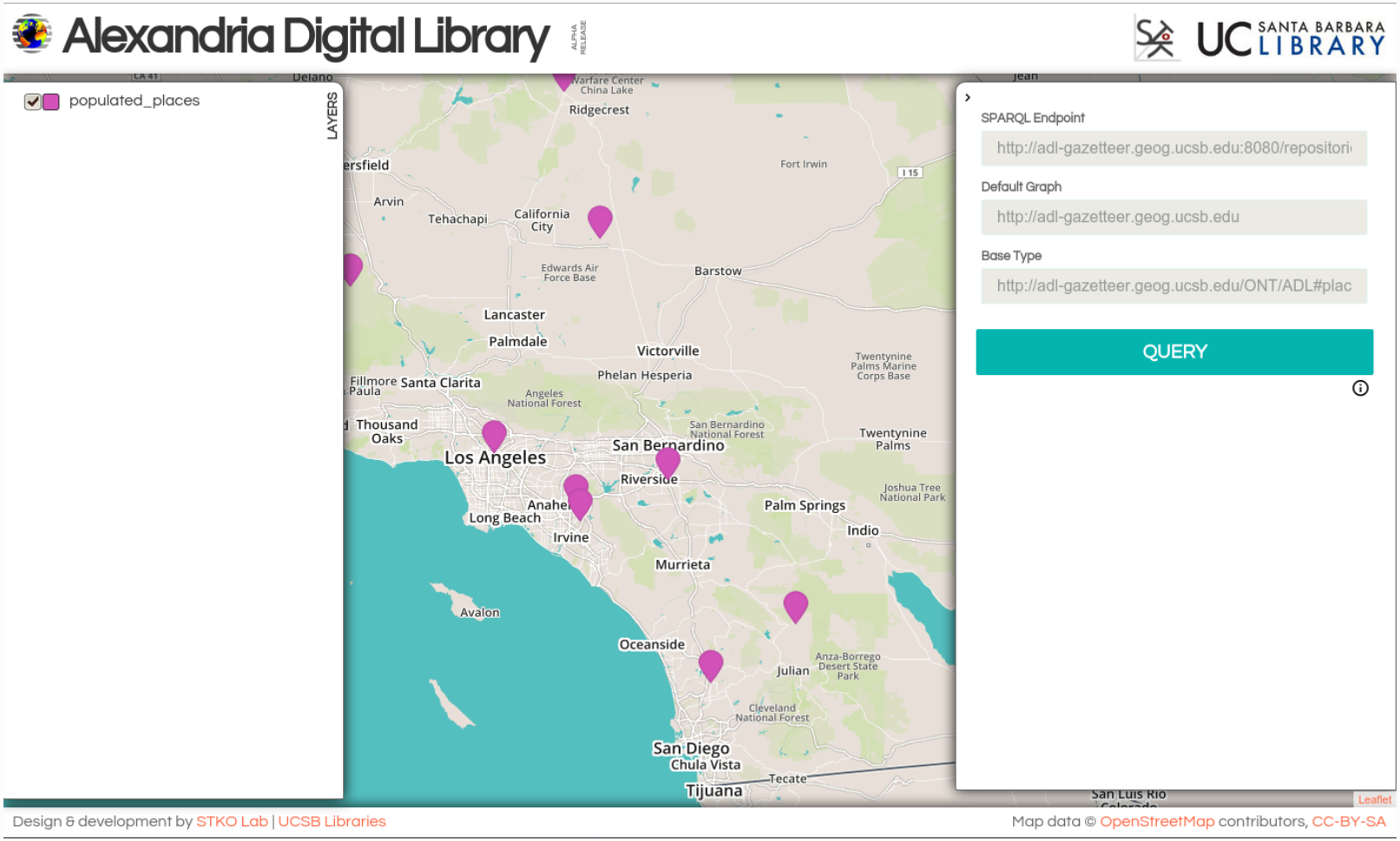
Our completely **client-based** JS explorer can be connected to any triple store

Ontologies as Interfaces



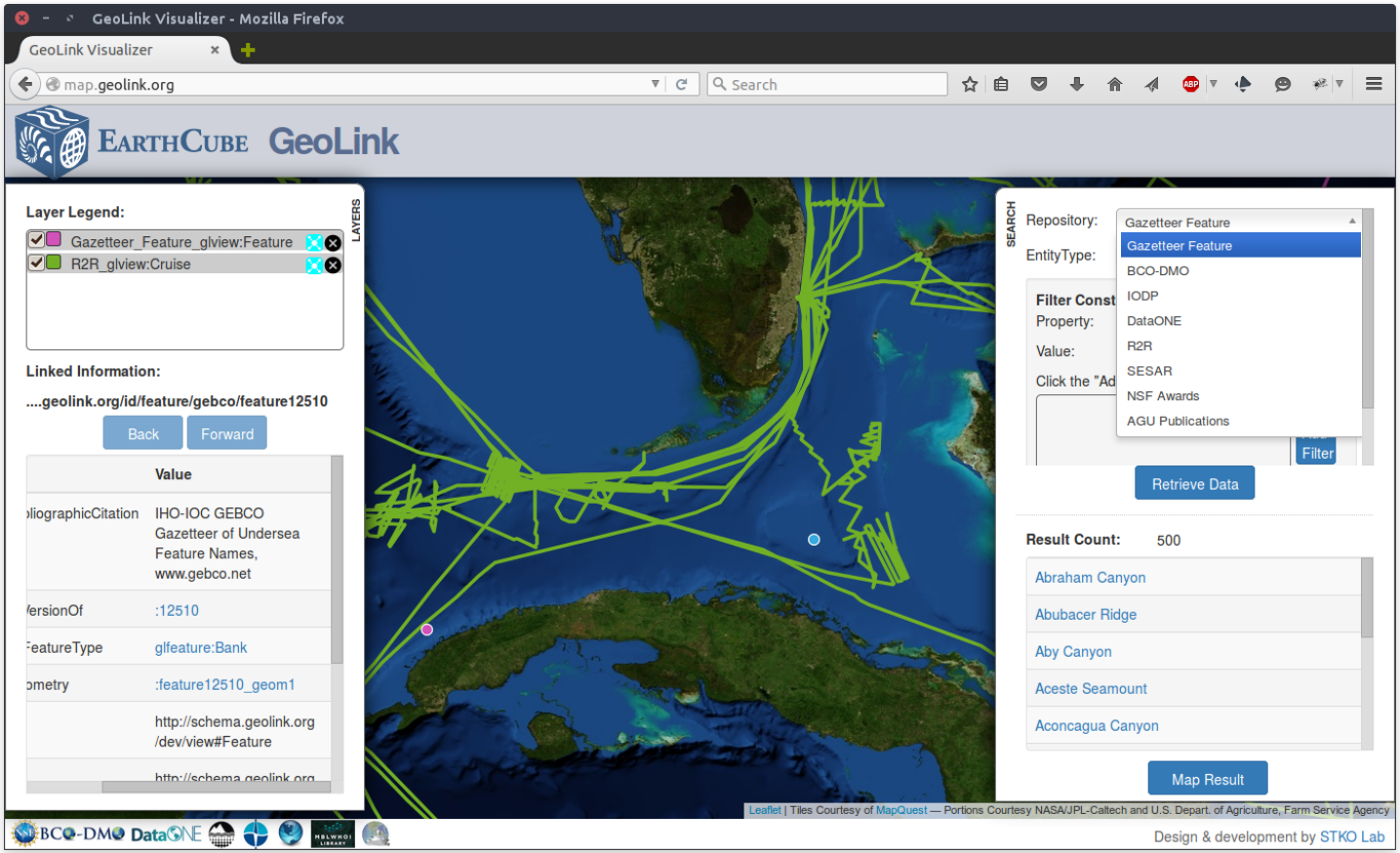
Allows users to explore Linked Data; constructs filters (e.g., >) based on **probing**

Ontologies as Interfaces



How does the interface know **which data can be mapped** and how to do this?

Ontologies as Interfaces



Our explorer can even combine data from multiple sources as **layers** (here cruises from R2R in green and gazetteer features in pink)

What about Time and Age?

Eratosthenes age

About 796,000 results (0.31 seconds)

about 2288 years (BC 276)
Eratosthenes, age
Hide details

Eratosthenes - Biography of ... - about.com
Eratosthenes was born around **276 B.C.E.** at a Greek ...

Eratosthenes - windows2universe.org
Eratosthenes was born in **276 B.C.** ...

Is this accurate? Yes - No

Eratosthenes - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/**Eratosthenes**
For the ancient Athenian statesman of the fifth century BC, see **Eratosthenes** .... 285 BC) since he is unlikely to have studied under him at the young **age** of 14.
↳ Sieve of Eratosthenes - Eratosthenes (crater) - Eratosthenes Seamount

Eratosthenes - Biography of **Eratosthenes**
geography.about.com/od/historyofgeography/a/**eratosthenes**.htm
Eratosthenes was born around **276 B.C.E.** at a Greek colony in Cyrene, Libya. ... In old **age**, **Eratosthenes** became blind and died of self-induced starvation in ...

Eratosthenes

en.wikipedia.org

Eratosthenes of Cyrene was a Greek mathematician, geographer, poet, athlete, astronomer, and music theorist. He was the first person to use the word "geography" and invented the discipline of geography as we understand it.
Wikipedia

Born: BC 276, Cyrene

Died: BC 194, Alexandria

Parents: Aglaus

People also search for


Ptolemy


Aristarc...


Archimedes


Euclid


Hipparchus

Report a problem

Google's Knowledge Graph in 2012

What about Time and Age?

Eratosthenes age

Web Images Maps Shopping More Search tools

About 2,160,000 results (0.19 seconds)

82 (276 BC–194 BC)

Eratosthenes, Age at death

Feedback / More info

Eratosthenes - Biography of **Eratosthenes** - Geography - About.com
geography.about.com › Education › Geography › History of Geography
In old **age**, **Eratosthenes** became blind and died of self-induced starvation in either 192 or 196 B.C.E. He had thus lived to be about 80 to 84 years old.

Eratosthenes - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/**Eratosthenes**
For the ancient Athenian statesman of the fifth century BC, see **Eratosthenes** .... 285 BC) since he is unlikely to have studied under him at the young **age** of 14.
Sieve of Eratosthenes - Eratosthenes (crater) - History of geodesy

Eratosthenes - Simple English Wikipedia, the free encyclopedia
simple.wikipedia.org/wiki/**Eratosthenes**
Eratosthenes was a friend of Archimedes, who also lived and worked in Alexandria. Archimedes was the greatest mathematician and inventor of the **age**, ...

Eratosthenes of Cyrene (Greek scientist) -- Encyclopedia Britannica

Eratosthenes

Mathematician

Eratosthenes of Cyrene was a Greek mathematician, geographer, poet, astronomer, and music theorist. He was the first person to use the word "geography" in Greek and he invented the discipline of geography as we understand it. Wikipedia

Born: 276 BC, Cyrene, Shahhat, Libya
Died: 194 BC, Alexandria, Egypt
Education: Platonic Academy
Parents: Aglaus

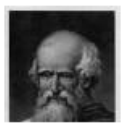
People also search for


Ptolemy


Hipparchus


Aristarchus of Samos

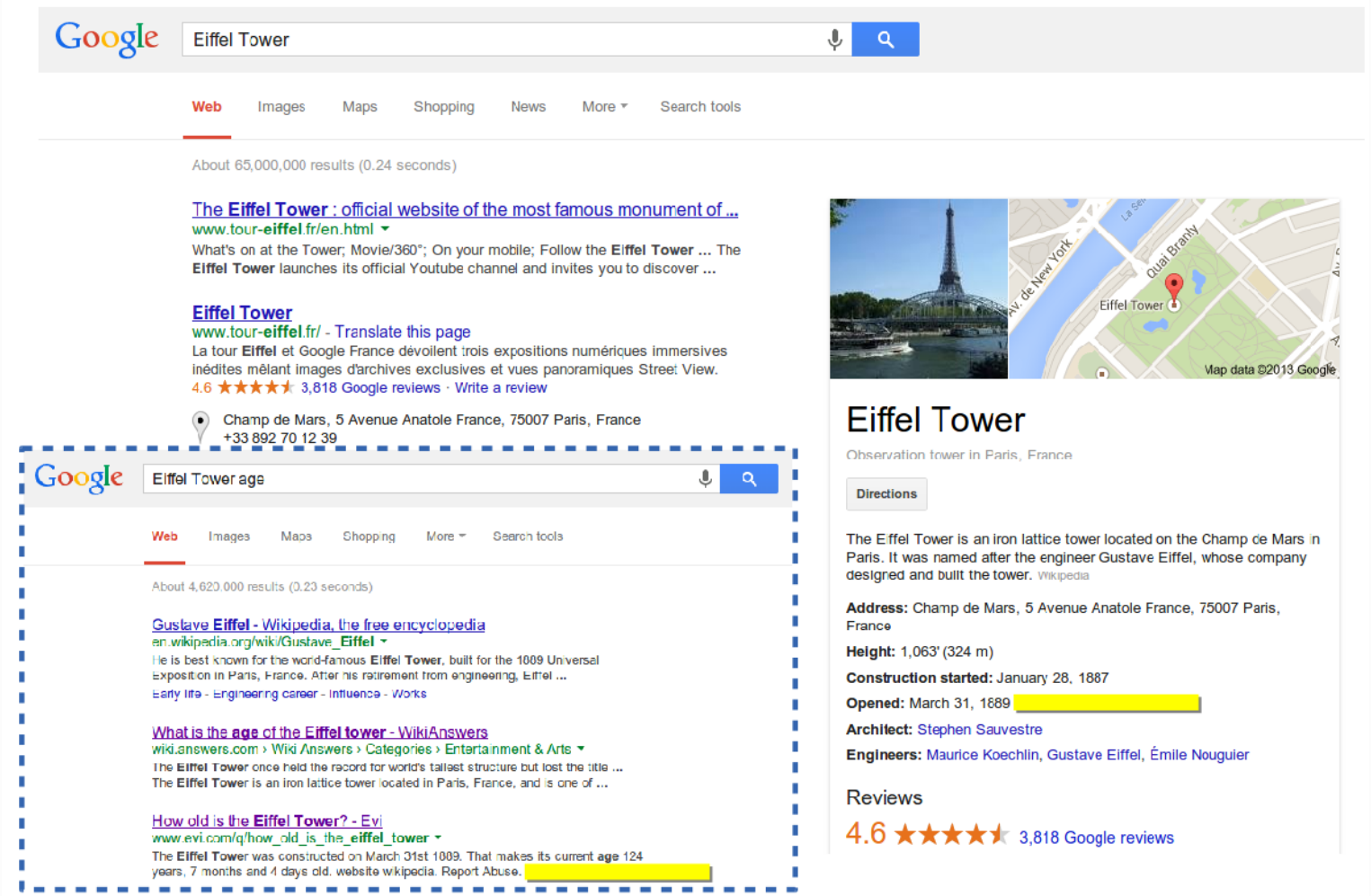

Strabo


Archimedes

Feedback / More info

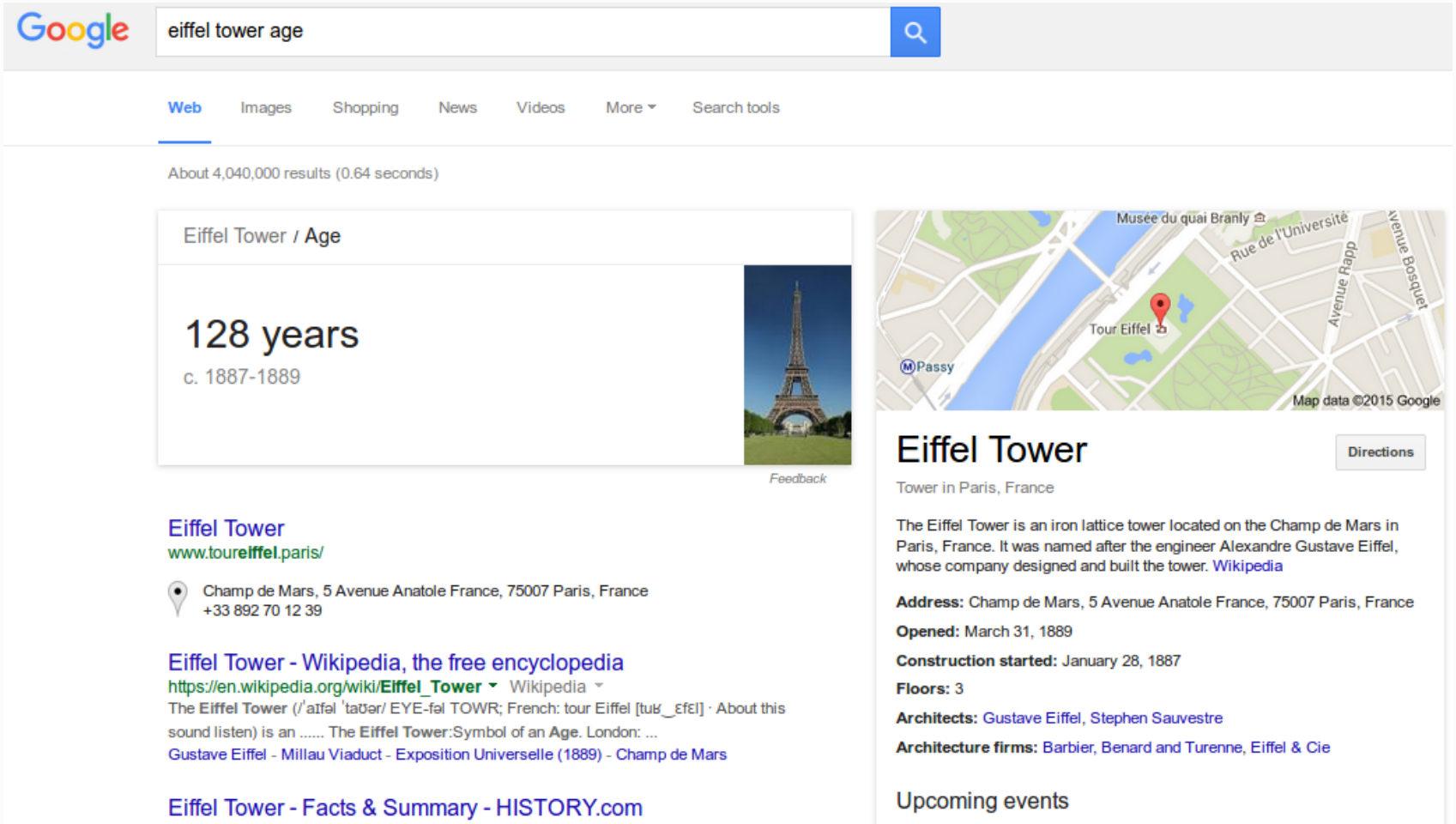
Google's Knowledge Graph in 2013

What about Time and Age?



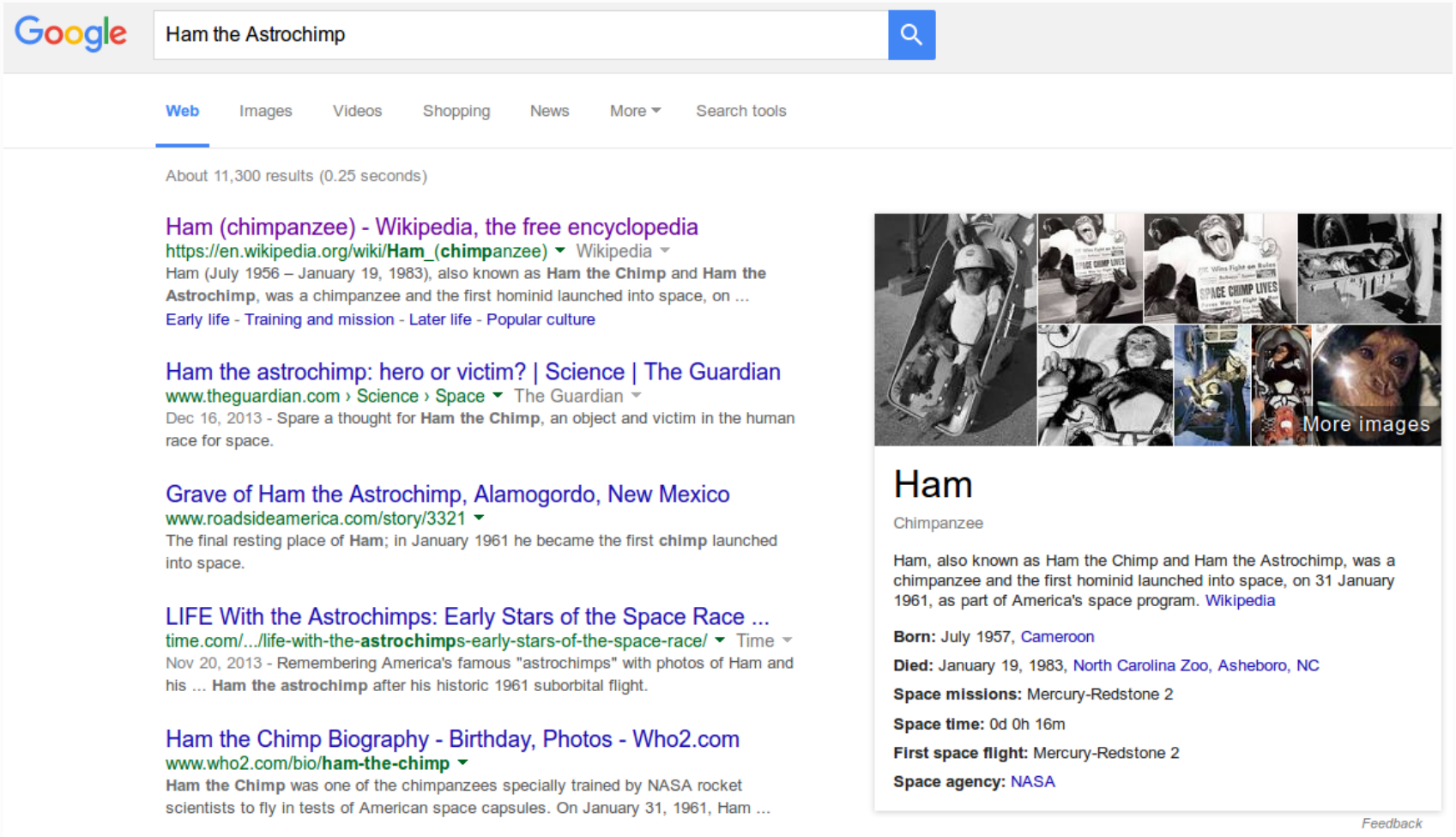
Google's Knowledge Graph in 2014

What about Time and Age?



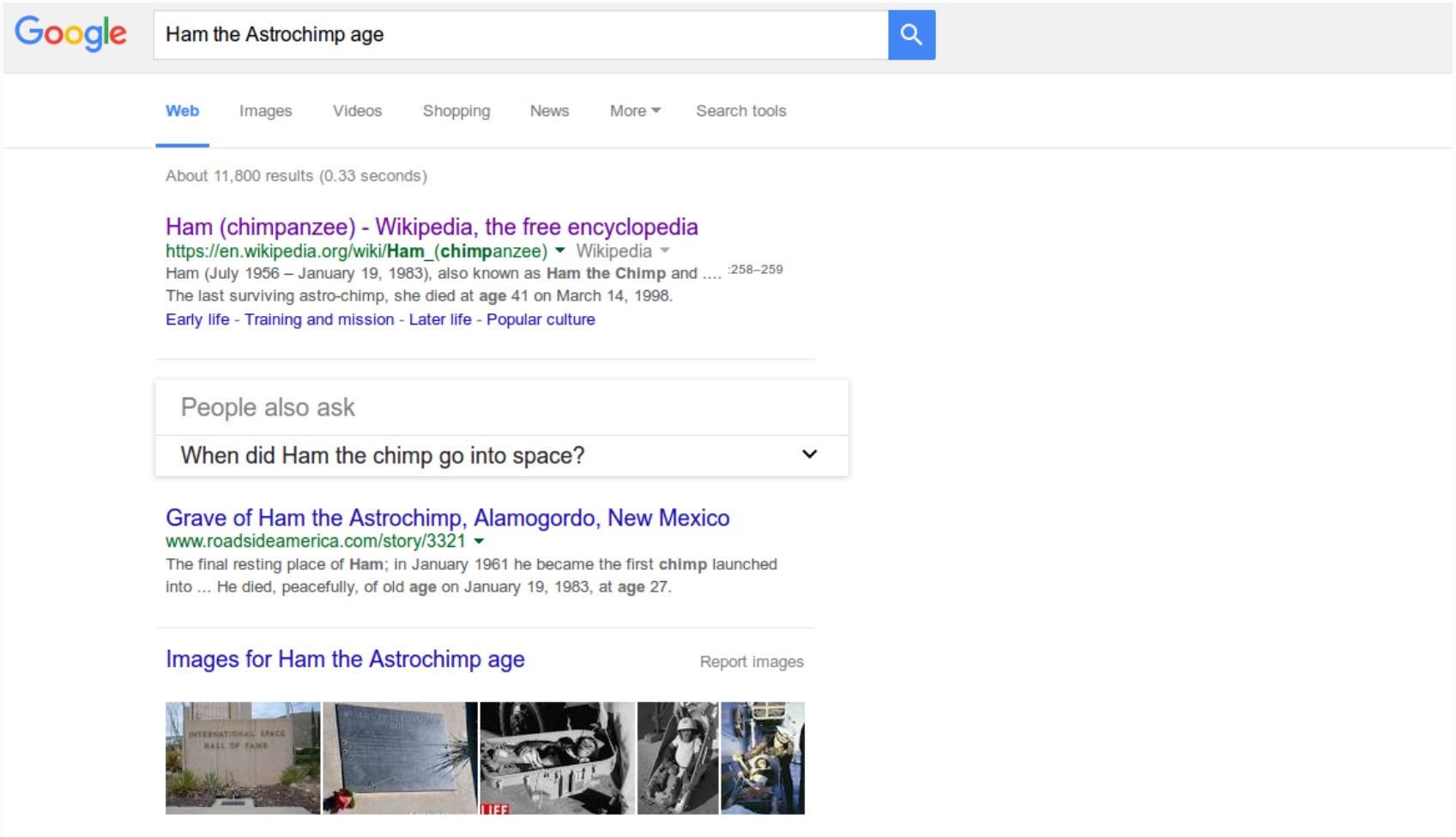
Google's Knowledge Graph in 2015

What about Time and Age?



Google's Knowledge Graph in 2015

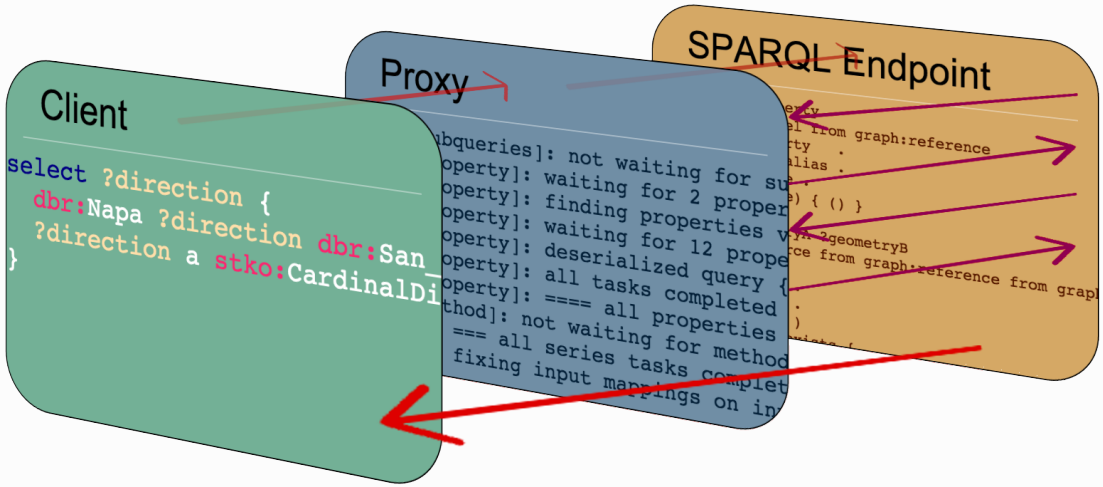
What about Time and Age?



Will you please define a rule how age is computed... (like a Time Interface ☺)

Storing versus Computing

- We cannot possibly **materialize** (store) all interesting (geo-)properties such as *cardinal directions* between places (≈ 452 billion), *nearby* triples, *elevations*, *topological* relations, population *density*, and so forth.
- However, we could try to **compute** properties on-demand (if we could figure out a way of doing so without changing the SW layer cake).
- The VOLT framework acts as a **transparent proxy** on top of any existing SPARQL 1.1 endpoint and can **compute results** and their **provenance** graphs **on-demand**.
- How to decide which properties to store and which to compute? → **patterns!**



Detailed slides at: http://www.slideshare.net/BlakeRegalia/volt-eswc-2016

Ontology Design Patterns Book