Making the Case for Ontology

Problems, Actions, Results

John F. Sowa

19 April 2011

Ontology Summit

PAR Methodology

Problem:

How can we show the benefits of ontology?

Proposed action:

Analyze what IT developers are currently doing.

Desired result:

Use ontology to improve interoperability, reliability, ease of development, and ease of use.

Ontology should become ubiquitous – a commodity.

Cyc at the Cleveland Clinic

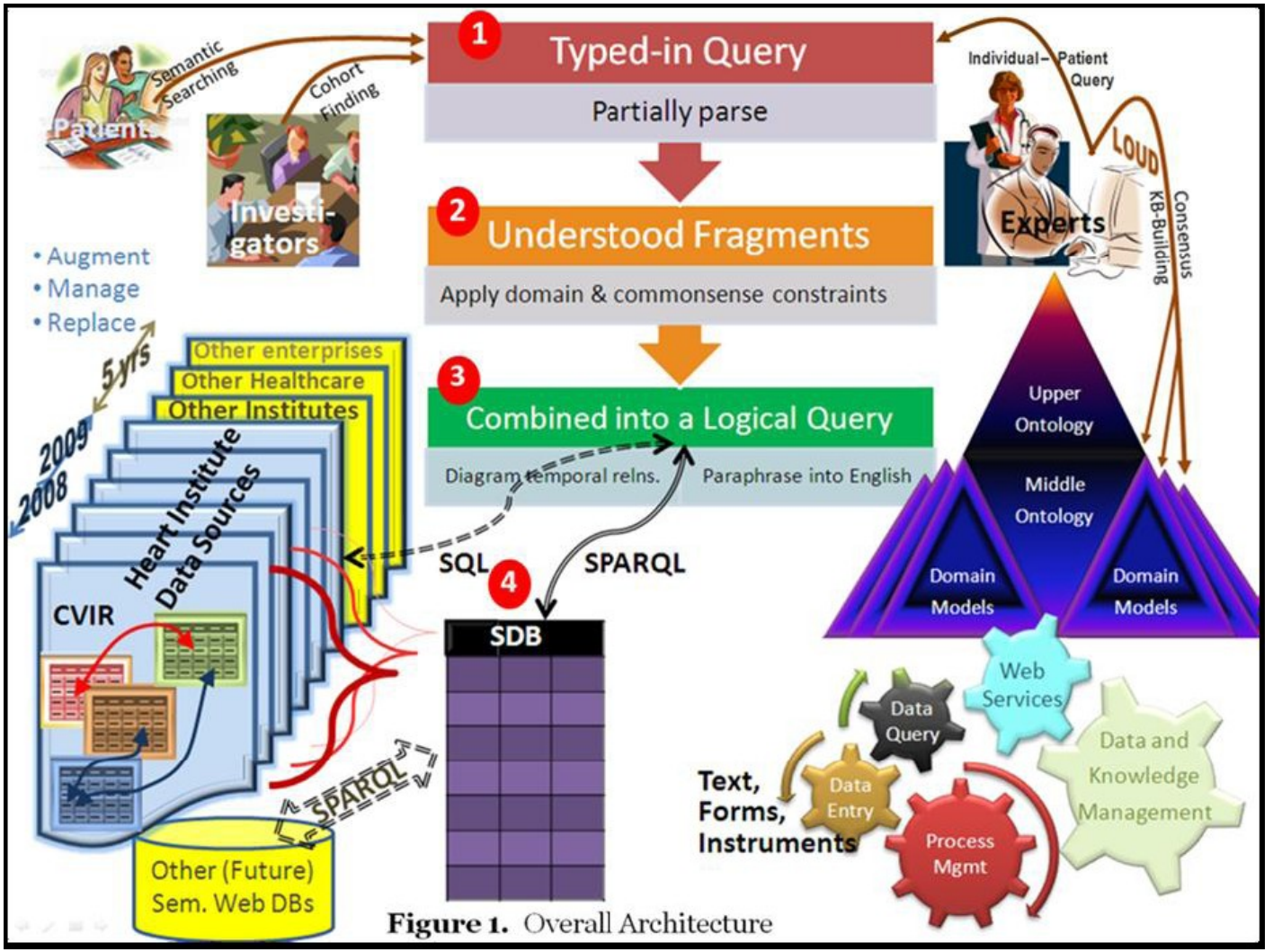

Figure 1. Overall Architecture

Source: http://www.cyc.com/technology/whitepapers_dir/Harnessing_Cyc_to_Answer_Clincal_Researchers_ad_hoc_Queries.pdf

Cleveland Clinic

Problem:

- Major differences in APIs, tools, and methodologies.
- Steep learning curve for IT personnel who try to use Cyc.
- Comment by Terry Longstreth at a DB symposium in 1980:

  "Any one of those tools, by itself, is a tremendous aid to productivity. But any two of them together will kill you."

Proposed action:

- Develop better methods for implementing, integrating, using and supporting all components.

Desired result:

- Simplify and unify the interfaces for new and old components.
- Provide a single, unified, semantic view of everything.

Challenge for Ontologists

Problem:

- Every software system has an ontology, implicit or explicit.
- That ontology is based on the terminology of the enterprise.
- Inevitable mismatch with any independently developed ontology, no matter how well designed and organized.

Required action:

- Develop methodologies and technologies for relating and integrating multiple terminologies and ontologies of any kind.

Desired result:

- Interoperable software – both new systems and legacy systems.
- Simpler, more natural interfaces for users and developers.

Meeting the Challenge

Problem:

- Trillions of dollars of legacy software.
- Most of the world's knowledge is in natural language.
- Most current semantic tools cannot process either one.

Required action:

- Extend semantic technology to support everything.

Desired result:

- Common tools that are integrated with all technologies.
- Automated extraction of ontologies from NL texts.
- Automated extraction of ontologies from legacy software.
- Automated integration of all ontologies from any source.