Orbis Semantic Annotations Research project

AGFA HealthCare

Orbis Semantic Annotations

- Problem space: heterogeneous data sources
- Semantic Web Technology
- Demo Basic Annotations
- Architecture
- Demo reasoner

Heterogeneous Data Sources

- Orbis: Electronic Patient Record of Agfa HealthCare
  - Broad coverage of healthcare domain
  - Consisting of serveral modules
    - Each with their specific domain objects and data structures
- Long history of legacy data
  - Acquisition of multiple EPR's in different countries
  - Availibility of advanced form generator
    - Allowing hospitals to build their own clinical record.
- Multiple data models
  - Relational data model – 'Static data model'
  - E.A.V data model – (Entity Attribute Value) 'Generic data model'
  - Mixed

Heterogeneous Data Sources

- How to use those data coming from heterogeneous data sources for
  - Re-use
  - Exchange of data (import, export)
  - Exploring
    - Aggregate and analyze relevant clinical data
  - Clinical decision support
- Architecture allowing uniform and controlled (read-only) access to patient data.
  - By abstracting from physical data structures
- Providing a logic view of the physical data structures used by the single modules

Semantic Web Technology

- How can computers grab the semantics or meaning of data?
  - Approach one: make them so intelligent that they will be able to process the information about the world in its full complexity. E.g. Understand human language.
  - Approach two: Simplify the description of the world to a level that even stupid computers will be able to act ‘intelligently’ on it.

    => Semantic technology

- Semantic Web
  - Vision of WWW to have (part of) its data in this simplified form rather than plain human language
  - Evolving from Web of documents to Web of meaningfull data.

Semantic Web Technology

- Basic unit of knowledge = Triple
  - Fact expressed as a <Subject Predicate Object> triple
    - ‘Tarzan Loves Jane’
    - ‘Myocardial_infarction’ ‘is_treated_by’ ‘Perfusion_therapy’.
  - Statement with subject, verb and object
  - Something with a meaning **connecting with a meaning to** something else with a meaning
  - Intelligent
    - From Latin ‘Intelligere’ - inter + legare (to tie, bind, unite)

Semantic Web Technology

- As explicit meaning as possible:
  - Trying to be Unambigious
  - Using URI: pointing to individual elements in the world
    - ‘http://www.ihtsdo.org/owlname#Myocardial_infarction’
      ‘http://www.agfa.com/w3c/2009/Therapy#is_treated_by’
    - ‘http://www.ihtsdo.org/owlname#Perfusion_therapy’.
  - **RDF** Resource Description Framework
    - Express data in a formal way (triples of URI’s..)
    - Collection of statements, each with subject, verb and object.
- Ultimately forming a **graph** of knowledge

Semantic Web Technology

- As explicit meaning as possible:
- **Ontologies** to fight ambiguity of concepts
  - Formalisation of a domain of discourse, enabling knowledge sharing
  - **RDF schema**
    - **This language permits declaration of a ontology**
    - rdfs:class, rdfs:subClass, rdfs:domain, rdfs:range
  - Rdfs:class = the class of all classes
    - Snomed concept
  - dfs:subclass = linking a class to its superclass
    - Snomed IS A hierarchy
  - rdfs:domain, rdfs:range = linking a property to its domain or range class
    - Currently only expressed as free text in snomed CT guides

Semantic Web Technology

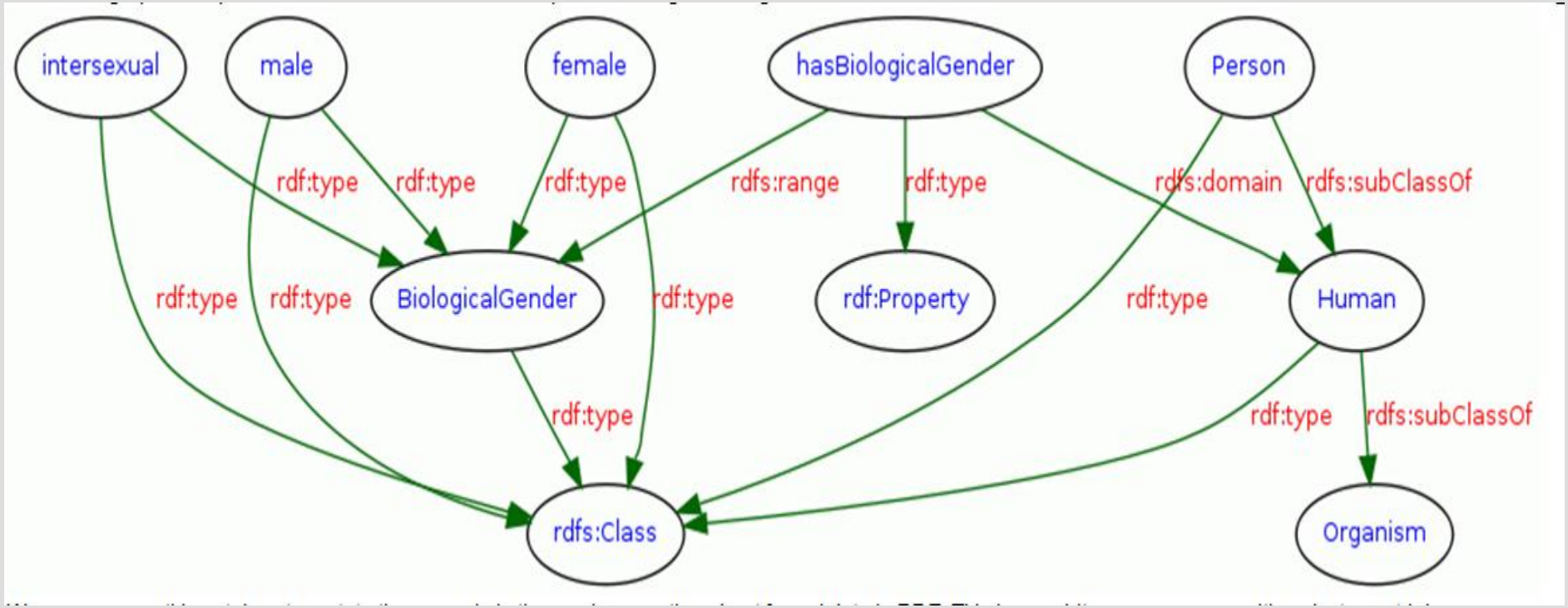
```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.
@prefix organism: <http://eulersharp.sourceforge.net/2003/03swap/organism#>.
@prefix human: <http://eulersharp.sourceforge.net/2003/03swap/human#>.

# Classes:
human:Human a rdfs:Class.
human:Human rdfs:subClassOf organism:Organism.
human:Person
    a rdfs:Class;
    rdfs:subClassOf human:Human.
human:BiologicalGender a rdfs:Class.
human:male
    a rdfs:Class;
    a human:BiologicalGender.
human:female
    a rdfs:Class;
    a human:BiologicalGender.
human:intersexual
    a rdfs:Class;
    a human:BiologicalGender.

# Peoperties:
human:hasBiologicalGender
    a rdf:Property;
    rdfs:domain human:Human;
    rdfs:range human:BiologicalGender.
```

Semantic Web Technology

RdfGraph



Semantic Web Technology

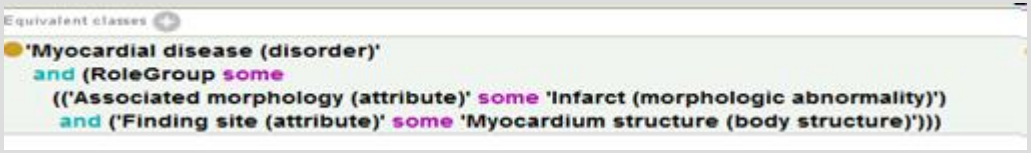
- Ontologies to fight ambiguity of concepts:
  - OWL
    - Extension of RDFS
    - permits declaring ontologies in a more expressive way than RDFS
  - Examples of OWL elements:
    - owl:equivalentClass
      - Snomed Fully defined!
    - owl:intersectionOf
      - Logical AND
    - owl:Restriction
      - Defines an unnamed class
      - This class defined by owl:onproperty and owl someValuesFrom(/allValues From) combination
    - owl:onProperty
    - Owl:someValuesFrom
      - All individuals that are related by the onProperty to other indivuals by at least …

Semantic Web Technology

```
:SCT_22298006 rdf:type owl:Class ;

              rdfs:label "Myocardial infarction (disorder)" ;

              owl:equivalentClass [ rdf:type owl:Class ;
                                    owl:intersectionOf ( :SCT_57809008
                                                         [ rdf:type owl:Restriction ;
                                                           owl:onProperty :RoleGroup ;
                                                           owl:someValuesFrom [ rdf:type owl:Class ;
                                                                                owl:intersectionOf ( [ rdf:type owl:Restriction ;
                                                                                                       owl:onProperty :SCT_116676008 ;
                                                                                                       owl:someValuesFrom :SCT_55641003
                                                                                                     ]
                                                                                                     [ rdf:type owl:Restriction ;
                                                                                                       owl:onProperty :SCT_363698007 ;
                                                                                                       owl:someValuesFrom :SCT_74281007
                                                                                                     ]
                                                                                                   )
                                                                              ]
                                                         ]
                                                       )
                                  ] .
```



Semantic Web Technology

- Inferencing by reasoning engine
  - Deriving new knowledge out of facts and rules
  - Making implicit knowledge explicit
- A rule is a statement with a logical implication
  - Consist of a premise and a conclusion
    - which is implicated if the condition in the premise is fulfilled
- **RDFS, OWL properties entail new knowledge!**
  - Example:

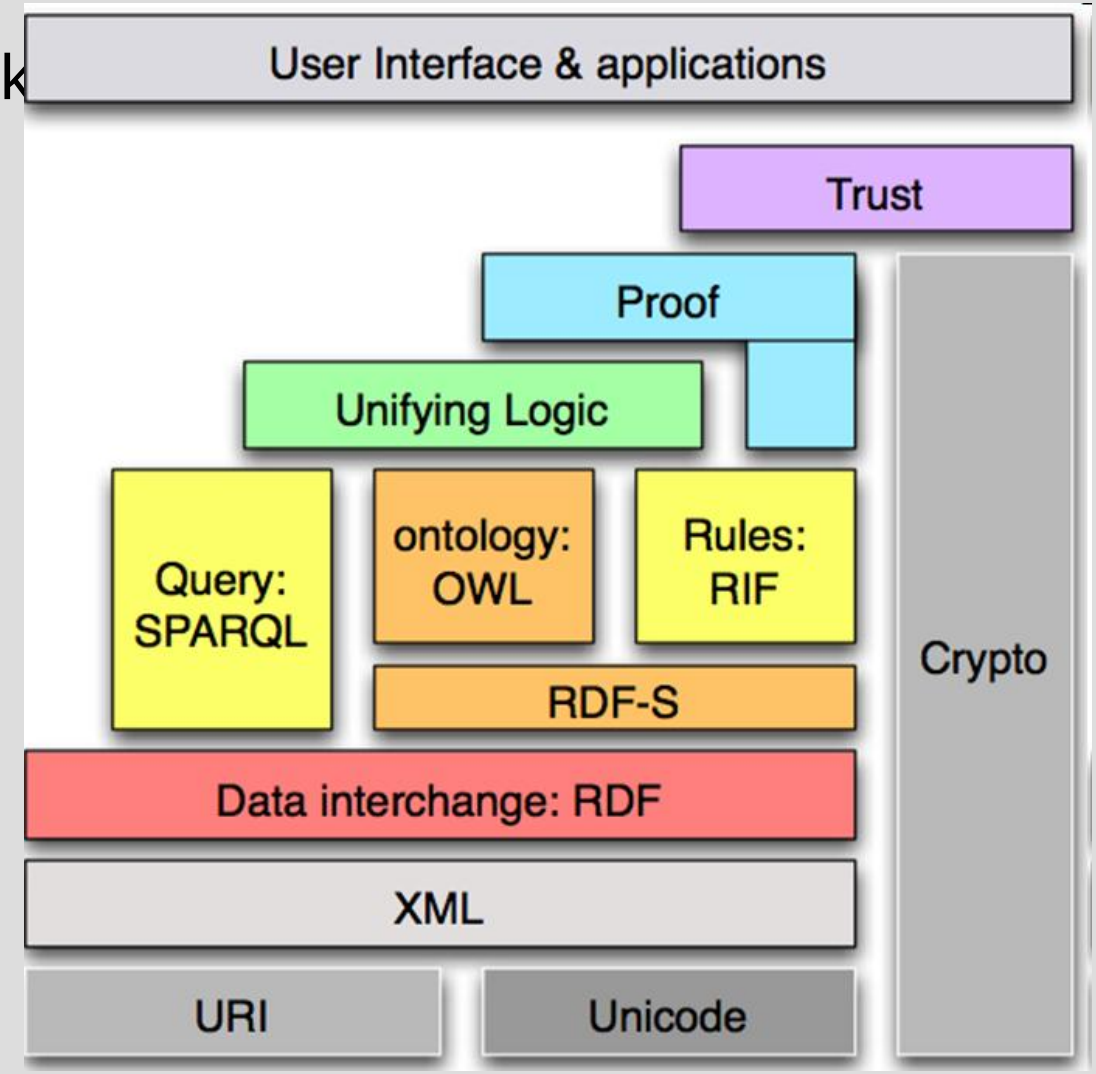
    {?X a ?Y. ?Y rdfs:subClassOf ?Z} => {?X a ?Z}.

    {?P a owl:TransitiveProperty. ?S ?P ?X. ?X ?P ?O.} => {?S ?P ?O}.

    http://eulersharp.sourceforge.net/2003/03swap/eye-owl2.html
- Snomed Classifier = reasoner

Semantic Web Technology

Semantic Web Stack



Semantic Annotations

- How did we leverage the technology provided by Semantic Web?
- First step: Annotation
  - Providing additional meta data for Orbis structures so that this data can be retrieved and processed by semantic layer
  - Basically an Orbis data element is tagged with to a concept of a external terminology.
    - Snomed CT
  - An ontology , close to the structures found in patient record
    - Static data
    - Generic data: generated on the fly

Semantic Annotations

- Second step: Retreiving data
  - Using SPARQL and SPARQL endpoint

- SPARQL
  - ***S****PARQL* ***P****rotocol* ***a****nd* ***R****DF* ***Q****uery* ***L****anguage*
  - Query language for RDF , similar to SQL
  - Triple facts are extracted from Orbis database into a RDF graph
    - Resulting in single logic view of the physical data structures

- **SPARQL Endpoints** allow querying existing data with SPARQL
  - Using standard HTTP protocol
  - For Orbis, a SPARQL endpoint has been defined

SPARQL Basic Query

Data:

```
@prefix
    http://example.org#> .
:John :age 25.
:Bill :age 30.
:Mary :age 24.
:Jane :age 26.
:John :loves :Mary.
:Bill :loves :Jane.
```

Query:

```
@prefix:  http://example.org#.

SELECT ?girl ?age
WHERE
{ ?man :loves ?girl .
?girl :age ?age }
```

Result:

girl	age
:Mary	24
:Jane	26

Annotating in Orbis

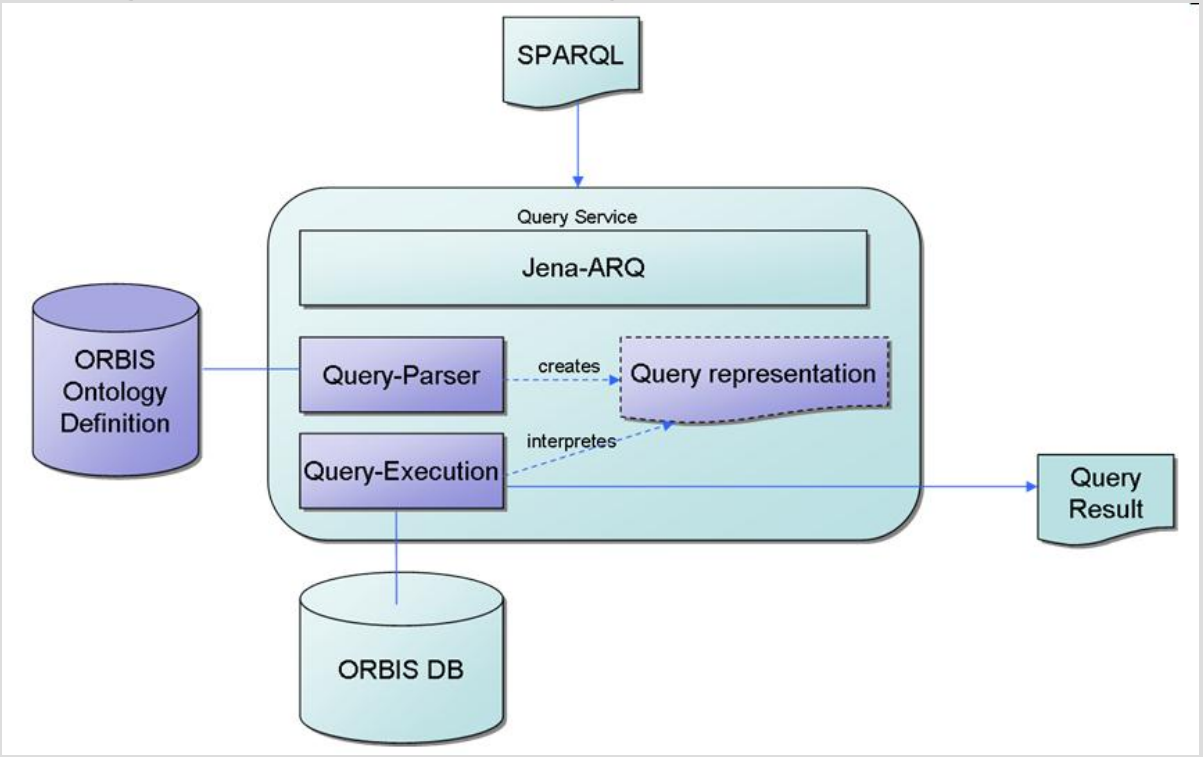
- Tagging data with explicit meaning by linking to a concepts of a clinical terminology (SNOMED CT)
- By filling in data in Orbis patient file, the concept of the terminology transforms from some theoretical representation of a clinical notion to an actual instance within the patient record.
- In other words, by doing so we are asserting facts, creating behind the scene triples.
  - patientX has_weight weightX
  - weightX has_value 83.
  - weightX is_measured_in Kg.
  - weightX has_timestamp 2010-09-09
  - ...
- Which we can retrieve by SPARQL
  - Where ever this weight has been filt in.

Annotating in Orbis

- Demo:
  - Annotating in Nursing form Body weight
  - Annotation in Lab Ordering Body weight
    - lab systems ask for additional clinical observations for some lab tests.
  - Re-use of data coming from different data sources
    - Pre-filling the Lab observation with values from Nursing.

Architecture

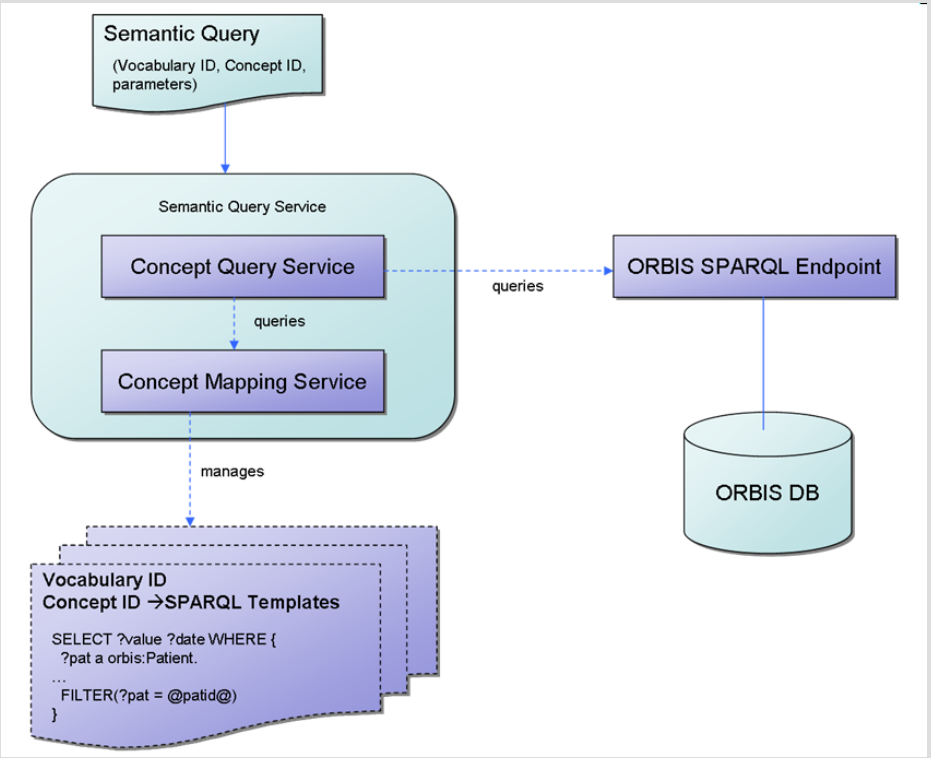
- SPARQL engine as foundation layer



- An ORBIS Ontology defines the object model SPARQL queries can operate on.
  - DDO: Data definition ontology
  - Ontology is close to the structures found in patient record

Architecture

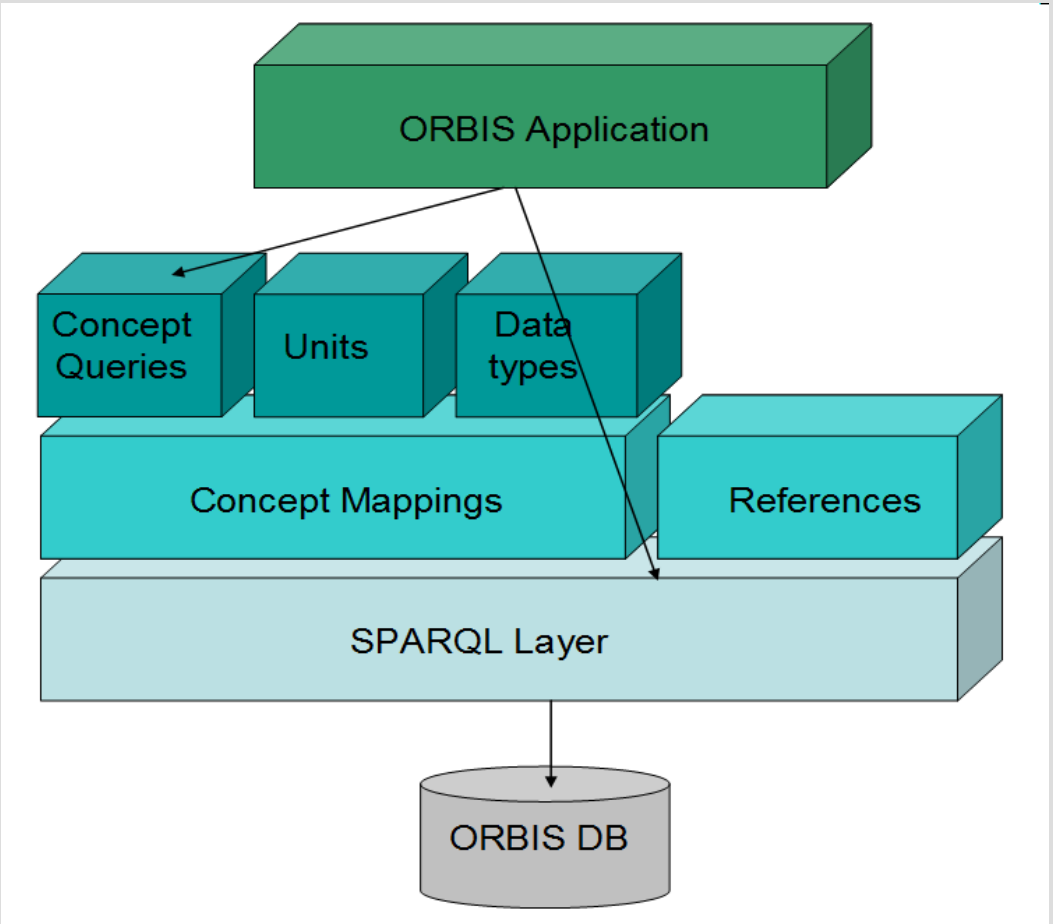
- Semantic Query Service
  - Works on top of ORBIS SPARQL endpoint
  - Maps Snomed Concepts to elements of ORBIS ontology



- Concept Mapping Service
  - Maps Snomed concepts to one or more Sparql queries.
  - SPARQL using ‘basic’ ORBIS ontology
- Concept Query Service
  - Retrieves data by executing SPARQL queries on the endpoint.

Architecture

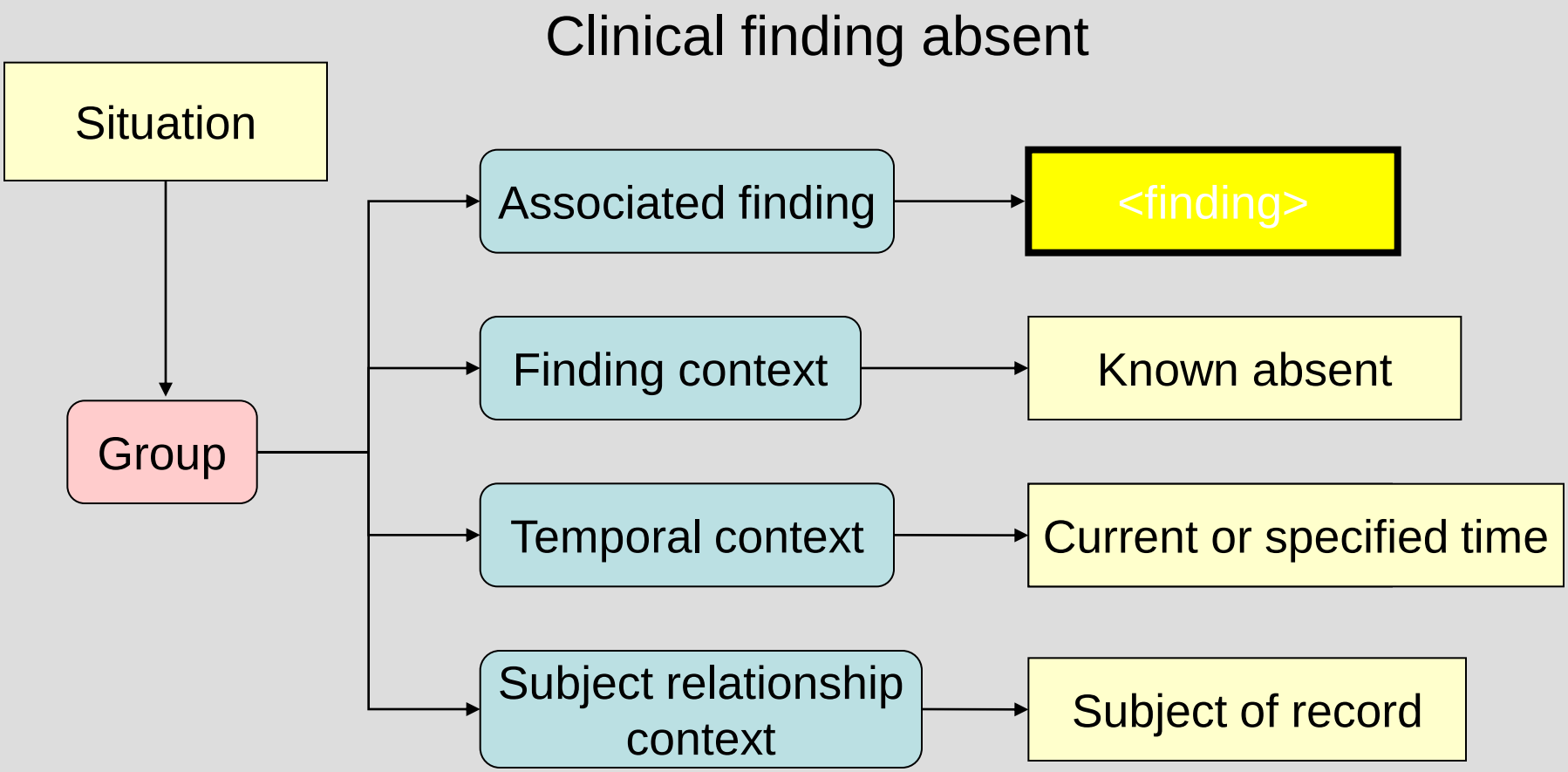
- Layered approach



Demo Reasoner

- How to handle
  - Yes – No – Don't know
  - History of
  - Family history of
- Embed in special construct "Clinical situation with explicit context"
  - Facilates recording in patient file
  - Additional information can be specified
    - to denote temporal position (in the past, now,…),
    - explicit presence or absence of this concept
    - modalities such as risk, planning state
    - subject to whom the artifact applies to (patient himself, family,…).
  - All concepts not defined in this 'situation with explicit context' and used in a clinical record are assumed to have weak defaults
    - meaning applying to the current patient, being present and current.

Demo reasoner



Demo Reasoner

- SPARQL query to retrieve data
- Mapping file Snomed
- Rule to define situation with explicit context

- On selecting ‘Cardiogenic shock’ in form, I can retrieve RDF graph stating this patient has clinical situation