OUTDATED ! USE GoogleDoc !

Integrate 2 RDBs (2002-2010) in order to query them together via NTDO and its epidemiology extension:

RDB1: Disease Symptoms

Exploitable fields: Sex, Age, Race, birthday, name, mothers name, **notified disease**, **place where the disease happen**, where the person live

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | DTB1 (SINAN-Morbidity) |  |  |  |  |  | DTB2 (SIM-Mortality) |  |  |
| DB field name |  | NM\_PACIENT | DT\_NASC | NM\_MAE\_PAC |  | DT\_OBITO |  |  |  |
| English Name |  | Patient Name | Birth Date | Mothers Name |  | Death Date |  |  |  |
| Value description |  |  |  |  |  |  |  |  |  |
| Used Datatype |  | String | Date | String |  | Date |  |  |  |
| Used Terminology |  |  |  |  |  |  |  |  |  |
| Mappable Terminology |  |  |  |  |  |  |  |  |  |
| Mappable Ontology |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

RDB2: Mortality

Exploitable fields: Sex, Age, Race, birthday, name, mothers name, death date, **main cause of death**, **where the death took place**, the actual **residency of the death person**

Primary key to connect RDBs: name, birthdate, parents name

Most fields are numeric (approximations for defined enumerated values/Symbol type datatypes). There is no freetext fields. The Region is a String. The disease is ICD10.

Use lexical text mining for normalization of Primary key fields, in order to ease mapping.

Scope and envisioned Application:

As the DTBs cover equal domains and same classes, this is about data quality checks rather than profiting from integrated synergy of new different information.

Ultimately we envision live feeds of uptodate DTB content. So an anonymization layer needs to be introduced.

Endusers are nurses and clinitians. They are provides with a predefined set of interesting queries in natural language. These are coupled to SPARQL queries we create.

AI:

Put stuff in svn.

Organize meeting with Zulma to get possible clinical queries of impact to the domain.

Create Architecture-diagram describing normalization, D2R and Endpoint layer.

Create DTB specification in Englisch and a Mapping Table for the paper.

DS: Start paper writing, Do a first backbone. (Googledoc?)

Fred: Where is the reasoning connected to

The GFO-inspired temporal reasoning with the casualty and death representation aims to model the background-knowledge to provide us the ontological endpoints **to create mappings between the mortality database and NTDO**. The Reasoner is an own implementation (Freddy).

In the OBML paper we described a new DL constructor (seems as an equal with a dot on the top) which we use to state that a temporal mark from a given process (mainly a pathology or a casualty) leads to death (by means of a biological death process). This symbol is used to **describe that an end temporal instance from a previous process is the same as the beginning instance of the following, to describe a temporal sequence of events.**

**The constructor allows for an equivalent of owl:same as, but on the axion level.**

Notes Zulma Meeting:

A new future project:

Freddy has phD student who creates a Quiz inspired eLearning setup: intelligent tutoring system based on NTDO & Pädagocical concepts, maybe synergies with Martin Boekers Lernziel Ontologie and the Freiburg eLearning Platform can be drawn.

DS: We should talk about the NTDO and its connection to the connected DBs. What is the intended Application ?

A high level querying system based on SPARQL and NTDO to provide integrated access to morbidity and casualty data. The gains are:

* The query uses the ontological terms much closer to natural language than the complicated field names of the DBs
* Reasoning is possible, e.g., one can ask for vector borne diseases´ notifications – such query cannot be performed over the DBs today,
* Some morbidity data can be rectified, e.g., if a person dies of a heart attack and there is a notification of Chagas disease of this individual, then his death main cause should change to Chagas instead of a heart attack.
* Track some outliers, e.g., a person in the countryside that goes down with a Schistosomosis symptom. Either we can find a previous casualty of this disease for the same individual in a place near some water basin, or we try to infer where he was contaminated (the nearer water basin).
* Geographical constraints of the diseases described in NTDO are of many uses such as the one above.

What are CQ that can only be answered via ontological semantics, but not with querying the 2 connected SQL DBs?

Using data probably no CQ, but querying only the ontology any complex DL reasoning cannot be captured by SPARQL queries.

What Semantics is to be exploited? How ?

See 1st answer.

What annotation is envisioned (Deep vs Shallow?), How are the deep annotations done ?

I guess first only shallow and after we gain experience on using the integrated solution, it is likely that we envision deep annotation.

Normalizations?

We plan to rely on already ready solutions and algorithms from the Health Ministery (researcher Antony Stevens).

CV/terminology mappings ?

How is Temporal reasoning applied ?

Rules or other procedural solution to link some disease symptoms that occur in a sequence.

In sparql we can (at max) exploit the Taxonomic generalizations found in the Ontology; the higher DL semantics stays unexploited. The is-a hierarchy can be traversed to drive more intuitive queries can be used to allow for more extensive DTB querying.

We can discover data outliers, e.g. when a person has a disease that in not normally occurring in a particular geographical location. (leveraging on different data found in the two DBs).--> CQ:

Use NTDO based temporal reasoning for separating Main cause of death from co-morbidities, i.e. discover data blurs in this direction.

DS: Are there other DBs that we can leverage on by integrating them into th existin two ? E.G. the Therapiel of disease with Pharmaceuticals ? PS: Not easy to get, not existing yet: The public sector do not provide yet this kind and level of information. The regulation to create this kind of system is still been created.

What queries do and can the state now ? What are the ones they like to state, but can’t with given single DBs ? What do they do with the query results ? Statistics ? Pipe into Tools/Apps, e.g. Diagrams for monitoring ?