Investigations into the use of Snomed CT to enhance an OpenMRS health information system

Ken Halland, Katarina Britz, Aurona Gerber

School of Computing, Unisa, Pretoria; and Knowledge Systems Group, Meraka Institute, CSIR, Pretoria

ABSTRACT

In this paper, we discuss the advantages of using formal medical ontologies to enhance health information systems. In particular, we consider the suitability of the medical ontology Snomed CT for enhancing a health information system developed in the OpenMRS framework. We propose ways in which a module of Snomed CT can be linked to an OpenMRS application, based on an implementation analysis.

CATEGORIES AND SUBJECT DESCRIPTORS
D.2.12 [Software]: Software Engineering - *interoperability*
I.2.4 [Computing methodologies]: Artificial Intelligence - *knowledge representation formalisms and methods*

KEYWORDS:
Ontologies, Snomed CT, OpenMRS, Health Information Systems

1 INTRODUCTION

An ontology (in the field of Computer Science and Information Systems) is a formal expression of knowledge about some domain, specifying the commonly accepted terminology and the relationships between its terms [1]. Perhaps the most well-known use of ontologies is for the so-called *semantic web*, where web pages are marked up with semantic information so that intelligent agents can mine the web for data more effectively [2]. Ontologies also have many other uses, such as for specifying terminology in the medical field.

Snomed CT is the most comprehensive and widely used ontology in health information systems. It consists of clinical terminology "with unique meanings and formal logic-based definitions organised into hierarchies" [3]. It is used extensively in the USA and UK, and is either being used or under serious consideration in numerous other countries, as well as bodies such as the EU.

OpenMRS is a "community-developed, open-source, enterprise electronic medical record system framework" [4]. The fact that it is a framework means that it provides a shell for implementers to create applications for storing medical records that meet the particular needs of a hospital or clinic. OpenMRS applications have been developed and successfully deployed for keeping records about HIV/Aids and TB patients at selected hospitals and clinics in a number of countries in Africa, including South Africa, Kenya, Rwanda, Lesotho, Zimbabwe, Mozambique, Uganda, and Tanzania.

The HISA (Health Informatics South Africa) conference in June 2008 incorporated an OpenMRS implementers meeting where developers of the OpenMRS framework and implementers of OpenMRS applications could get together and discuss issues of mutual concern. During these meetings, the need was expressed to enrich the data model, in particular the concept dictionary provided by OpenMRS, with some form of ontology. We therefore decided to investigate the possibility of combining Snomed CT and OpenMRS in some way in order to fulfil this need.

The structure of this paper is as follows: In Section 2, we describe ontologies in more detail, and illustrate how they can be used to specify terminology in the medical domain. The main uses of ontologies, namely for semantic interoperability, for reasoning and for enhancing database access, are also discussed. In Section 3, we describe Snomed CT and list its so-called upper level concepts. In Section 4, we discuss OpenMRS, in particular its use of a concept dictionary to store clinical terminology. In Section 5, we discuss how the upper level concepts of Snomed CT could be linked to an OpenMRS concept dictionary, and some of the issues involved in this, and in Section 6, we describe our experiences and findings that arose from such an attempt. Finally, in Section 7, we make some recommendations about how problems that were encountered, could be overcome, suggesting various approaches that could be followed.

Email: Ken Halland hallakj@unisa.ac.za, Katarina Britz arina.britz@meraka.org.za, Aurona Gerber aurona.gerber@meraka.org.za

2 ONTOLOGIES AND THEIR USES

One of the most quoted definitions of *ontology* is due to Gruber [5], namely "a formal, explicit specification of a shared conceptualisation". In other words, an ontology defines the terminology or vocabulary used in a domain so that people or systems can ensure that they unambiguously mean the same thing when they use those terms. The meaning(s) of the terms that constitute the terminology are captured by precisely specifying the types of terms and their relationships with one another.

A medical ontology is therefore a list of medical terminology and a specification of the relationships between the terms that constitute it. For example, here is part of a medical ontology about the disease *tuberculosis* (TB):

$$\mathsf{Tuberculosis} \sqsubseteq \mathsf{BacterialInfectiousDisease}$$
$$\mathsf{MycobacteriumTB} \sqsubseteq \mathsf{Bacterium}$$
$$\mathsf{Tuberculosis} \sqsubseteq \exists\mathsf{causedBy}.\mathsf{MycobacteriumTB}$$

These three statements express the medical knowledge that 'tuberculosis is a bacterial infectious disease', that 'mycobacterium TB is a bacterium', and that 'tuberculosis is caused by some mycobacterium TB', respectively.

In order to ensure that the meaning of each term can be precisely defined, and also to allow computer systems to work with them, an ontology is always expressed in some formal notation. Various formal notations exist for expressing ontologies, e.g. abstract description logic syntax (used in the example above) [6], web ontology language (OWL) [7] and knowledge representation system specification (KRSS) [8]. The example above would look as follows in KRSS:

```
(define-primitive-concept Tuberculosis
     BacterialInfectiousDisease)
(define-primitive-concept
     MycobacteriumTB Bacterium)
(define-concept Tuberculosis
     (some causedBy MycobacteriumTB))
```

Most (but not all) ontology formalisms are logic-based. The advantages of logic-based formalisms are that the semantics are precisely defined and well-understood (i.e. there is a long history of research into the semantics of various forms of formal logic), and they allow automated logical reasoning (i.e. many reasoning algorithms have been developed to work with these formalisms).

In general, the two main uses of ontologies are for semantic interoperability and for reasoning. There is also renewed interest in the integration of logic-based ontologies with databases. We discuss these three uses now.

2.1 Semantic interoperability

By *interoperability* we mean the ability of computer systems to communicate with one another. Data transfer between two systems has to be accurate (and this is normally achieved by some form of network protocol), but the systems must also be sure that they understand what the data *means* in the same way. If both systems use the same ontology, we can ensure that the transfer of such knowledge and understanding is accurate [9]. This is termed *semantic interoperability.*

The ideal situation is where two systems use exactly the same ontology, but often this is not possible. There is extensive and ongoing research in the fields of *ontology integration* [10] and *ontology mapping* [11]. Ontology integration is where two different ontologies need to be merged so that the resulting ontology maintains the meanings of the terms specified in the separate ontologies. Ontology mapping is where terms in one ontology are mapped to terms in another ontology so that their meanings remain the same. Things get even more complicated when the two ontologies are specified in languages of differing expressivity. (See Section 2.2 below for a discussion of expressivity.) These are all problems in achieving semantic interoperability.

In the medical domain, there are often multiple health information systems (HISs) that need to communicate with one another. The SemanticHealth Report published by the European Commission [12] defines interoperability in the realm of HISs as

> ... the ability to ... exchange, understand and act on citizens/patients and other health-related information and knowledge among ... disparate health professionals, patients and other actors and organisations within and across health system jurisdictions in a collaborative manner.

Interoperability becomes a challenge when the HISs involved use different standards and/or data formats for storing and processing information. Just as important is the medical terminology that the two systems use, and particularly what is meant by each of the clinical terms. An important purpose of a medical ontology is therefore to achieve semantic interoperability between HISs.

2.2 Reasoning

By reasoning, we mean being able to derive some logical conclusion from knowledge. If the knowledge is expressed in statements using some formal notation, reasoning allows us to infer some additional statements that are implicit in the stated knowledge, i.e. which are not stated explictly. For example, from the three statements given at the beginning of Section 2 above, we could conclude that

$$\mathsf{Tuberculosis} \sqsubseteq \exists\mathsf{causedBy}.\mathsf{Bacterium}$$

i.e. 'tuberculosis is caused by some bacterium', even though this is not stated explicitly in our knowledge.

There are different types of reasoning tasks that can be posed to a reasoner about a set of statements. One task would be to ask whether a particular statement is true with respect to a set of statements. For example, we could ask whether the above statement is true with respect to the three statements given earlier. Another task would be to check whether a set of statements is consistent, i.e. that they do not contradict one another. Yet another task would be to check whether a given description is satifiable with respect to a set of statements, i.e. whether it is possible that there are individuals which comply with the description. For example, we could ask the reasoner whether the description $\mathsf{Tuberculosis} \sqcap \mathsf{Bacterium}$ is satisfiable with respect to the set of statements given earlier. Finally, we could ask whether a statement about an individual is true with respect to a set of statements, i.e. whether some individual complies with a given description (see Section 6.2.2 below for an example).

This final reasoning task may seem very similar to a database query, like 'give me all the individuals who are infected with tuberculosis'. However, reasoning over logic-based ontologies amounts to reasoning over all possible interpretations of the statements expressed in the ontology. This is in contrast with reasoning over a database, which amounts to reasoning over a single interpretation. (See the discussion of the open- and closed-world assumptions in Section 2.3 below.)

Almost all logic-based ontologies are based on *description logics* [6], a family of logics particularly suited to expressing ontological knowledge and reasoning about it. Each member of this family has a different measure of expressiveness, i.e. it is able to express particular nuances of knowledge. The reason why all these logics are not lumped together into one über-expressive logic is that one wouldn't be able to reason efficiently over knowledge expressed in it. Each logic in this family is limited in its expressiveness by the existence of an efficient algorithm to reason over it. In other words, there is a trade-off between the expressiveness of respective description logics and the efficiency of algorithms that can reason about knowledge expressed in them.

There are some highly expressive description logics, e.g. $\mathcal{SHOIN}(\mathcal{D})$ and $\mathcal{SROIQ}$, whose reasoning algorithms, although theoretically shown to be of intractable complexity, nevertheless in practice perform quite acceptably for small to medium-sized ontologies.

For very large ontologies, less expressive description logics which have reasoning algorithms of tractable complexity are preferred. For example, the description logic $\mathcal{EL}^+$ [13], which is the underlying logic of SNOMED CT, has limited expressiveness (e.g. it does not allow one to express negation, as in 'a bacterium is *not* a virus'). This restriction allows algorithms to reason efficiently over large ontologies expressed in $\mathcal{EL}^+$. On the other hand, the description logic *DL-Lite* [14] allows primitive negation but does not allow qualified existential quantification (as in 'tuberculosis is caused by *some* bacterium') as provided in $\mathcal{EL}^+$. This allows the implementation of reasoners which can operate efficiently over database schemas expressed as *DL-Lite* ontologies [15].

2.3 Integration with databases

Linking ontologies to databases has been an active field of research recently and several approaches have been proposed [16, 17, 18, 6]. The reader should note that in this field, the term *ontology* does not always refer to a formal, logic-based ontology, but is often used in a wider context. However, we limit our definition of *ontology* to mean a *logic-based ontology*.

There are a number of applications of logic-based ontologies in the field of databases [19, 6]. Firstly, reasoning can be used to identify problems in the conceptual data model of an existing system, or during the development of a system. If the data model is expressed as an ontology, reasoning can be used to find semantic inconsistencies such as any concepts that are unsatisfiable. An example of this is the ICOM tool for intelligent conceptual modeling [20].

Furthermore, when using an ontology as conceptual data model, it is possible to reason over queries. In other words, it is often possible to simplify the query before it is posted to the database, or even to answer the query without doing a table lookup. Reasoning can also be used for so-called *intelligent querying*, i.e. answering queries utilising logic-based reasoning that can't be answered by standard query mechanisms [21]. (Note, by *reasoning over queries* we exclude the types of query simplifications which are possible with standard database technology.)

With regards to the coupling of ontologies to databases, several of the initial tools that were developed, imported the data from a database into the ontology as instance data (see Section 6.2.2). Examples are DataMaster, RDB2Onto and Relational.OWL [16, 22, 17]. Other tools such as DB2OWL, VisAVis, DBOM, $\mathrm{R_2O}$, D2R Map, D2RQ and OBDA retain the database separate from the ontology implementation and provide an ontology-to-database mapping mechanism to interact with the data [23, 24, 25, 26, 27, 18]. Except for OBDA, these tools support the coupling to database data using binary relations only and they mostly do not support the latest OWL 2.0 standard or state of the art reasoning technologies. For the purpose of this paper, we limit the discussion to tools that support the most recent developments in OWL reasoning, namely SHER [28] and the work related to *DL-Lite* and the OBDA toolset [29, 30, 31].

One of the most active fields of research in the area of combining formal ontologies with large databases, is in the *DL-Lite* family of description logics [32, 33]. In this research the intent is to provide access to data in a database through a mediating ontology. The ontology provides the semantic model of the data which should allow for inferring new knowledge about the data, the verification of data integrity and semantic data integration. An OBDA plugin for Protégé is available for this purpose that provides ontology editing and data mapping functionality, as well as a querying facility that allows a user to query the database through the mediating ontology [29, 30, 31]. The biggest disadvantage of the *DL-Lite* and OBDA approach is the limited expressiveness of *DL-Lite*. However, inference of subsumption queries already provide a user with functionality that is not readily available in RDBMSs with SQL queries. In addition, the fact that a query can be posed to the data source through the ontological domain model is regarded as a substantial benefit by most users. This advantage means that it should not be necessary to appoint database and SQL specialists in order to extract relevant information from the relational data sources. A domain expert should be able to extract information using domain knowledge through the ontology in a far more intuitive way.

Sher is described as a *scalable highly expressive reasoner* that provides the functionality for semantic querying of large relational datasets through Owl ontologies. Sher provides standard description logic reasoning services including consistency checking and conjunctive query answering, and supports the Owl 1.0 logic Owl-DL but excluding nominals and datatypes [34, 28]. The Sher toolset performs limited reasoning when loading an ontology and executes most of its reasoning when doing query answering. Another key feature of Sher is its ability to tolerate logical inconsistencies in the data by not terminating when inconsistencies are detected, but by pointing a user to the source of the inconsistencies.

Issues than have to be borne in mind when coupling ontologies with databases, are the *impedance mismatch* problem, as well as the *open-world vs closed-world assumption.* Poggi et al [18] summarise the impedance mismatch as the problem arising from the difference between the basic elements managed by the data source, namely the data tuples, and the elements managed by the ontology, namely concepts and instances. When this problem is not handled properly, the user will not extract the correct data. The solution is a robust mapping language that allows a user to map data source elements appropriately to the elements of the ontology, a claim made by the OBDA team. The impedance mismatch has to be managed by creating mappings that ensure the correct consequences and inferences.

In addition, when coupling ontologies with databases, a user has to keep in mind that formal ontologies support an *open-world assumption*, meaning that if a fact is not known, an answer will not be returned that contradicts this fact. In the database world however, a *closed-world assumption* holds, meaning that "if a fact is not contained in the database, the fact is assumed false" [35]. Thus, when asking the question to a database whether there is a train between Johannesburg and Pretoria tonight at 20:00, a query to a RDBMS will return *no* if there is no data record in the database for a train at 20:00. Absence of data implies a *false* assumption. However, the same question posed to an ontology will result in an *empty* or *null* answer, not *false.* If the fact is not asserted that there is no train at 20:00, a query to the ontology will not answer that there is no train based on the absence of data such as in a RDBMS. When coupling an ontology with a database and posing queries to the application through the ontology, the results will be based on an open-world assumption and only facts that are asserted and could be inferred from assertions, will be returned.

3 SNOMED CT

As stated above, Snomed CT is an industrial-scale, logic-based ontology specifying all the terminology one needs for any medical or clinical purpose. It consists of more than 300 000 terms representing concepts as well as over 1 000 000 terms representing relations between the concepts. The concepts are arranged into hierarchies, with more general concepts higher up, and more specific concepts lower down. The so called *upper level* concepts are the most general [3] and are as follows:

- *Clinical finding/disorder* Results of clinical observations, assessments or judgements, including diseases and disorders
- *Procedure/intervention* Activities performed in the provision of health care, including invasive procedures, administration of medicines, imaging, education and administrative procedures
- *Observable entity* Aspects, factors or procedures to which values can be assigned, for example blood pressure, temperature, colour of nails, etc.
- *Body structure* Normal as well as abnormal morphological/anatomical structures specifying body sites involved in diseases or procedures
- *Organism* Animals, plants and micro-organisms of significance in medicine, particularly causes of diseases and conditions
- *Substance* Active chemical constituents of drugs, food and chemical allergens, causes of adverse reactions, toxicity or poisoning, etc.
- *Pharmaceutical/biologic product* Medicines, drugs, vaccines and other pharmaceutical compounds
- *Specimen* Entities obtained (usually from a patient) for examination or analysis, often including the source from which they are obtained, the procedure used to collect them and the substance(s) of which they are comprised
- *Physical object* Natural and man-made objects such as medical devices, implants, surgical implements, life support systems and artificial organs
- *Physical force* Primarily forces that represent mechanisms of injury, such as heat, pressure, electric current, or friction
- *Event* Environmental occurrences such as floods, earthquakes and chemical spillages
- *Environment/geographical location* Medical and other environments as well as named locations such as countries, states, and regions
- *Social context* Social conditions and circumstances such as family and economic status, ethnic and religious heritage, life style, and occupations
- *Staging and scales* Assessment scales (e.g. burn degrees and intelligence scales) and tumor/cancer stages

These upper level concepts each represent entire hierarchies of further, more specific concepts. Concepts from one hierarchy are linked to concepts in other hierarchies by means of relations. For example, the relation hasCausativeAgent relates some subconcept of Disease in the ClinicalFinding hierarchy to some concept in the Organism or Substance hierarchies.

Part of SNOMED CT is provided in the Appendix.

4 OPENMRS CONCEPT DICTIONARY

The OpenMRS data model comprises numerous tables for storing all sorts of data; primarily health records of patients. A selection of these tables are used to define the so-called *concept dictionary* of an application. This lists all the possible medical concepts that can occur in the application. These concepts are grouped into classes, and together they can be considered as a 'flat ontology'. The guidelines provided to OpenMRS implementers for populating the concept dictionary recommend the following classes [4]:

- *Test* Laboratory tests or physical examination maneuvers
- *Procedure* Actions performed in the diagnosis or treatment of conditions
- *Drug* Medications, prescriptions and over-the-counter dispensing
- *Diagnosis* Medical conclusions
- *Finding* Observations or results of tests or examinations
- *Anatomy* Body parts
- *Question* Queries to which there are open-ended or coded responses
- *LabSet* Groupings of tests or procedures
- *MedSet* Groupings of medications
- *ConvSet* Groupings of questions (e.g. vital signs)
- *Symptom* Signs or indications of possible conclusions
- *Specimen* Samples of tissue or fluid
- *Program* Plans or sets of plans consisting of tests or procedures to be followed
- *Workflow* Processes described/prescribed by the organisation
- *State* Descriptions of patients' status
- *Misc* Unclassifiable concepts

Some of these classes are represented by their own tables (e.g. drugs) which are related to concepts in the dictionary by standard database relations. However, apart from this and the simple *is-a* relation provided by the abovementioned classes, the OpenMRS data model does not allow the definition of hierarchies of concepts or of relations between concepts as a proper ontology would.

5 COMBINING OPENMRS & SNOMED CT

The similarities between the upper level concepts of SNOMED CT and the concept classes in an OpenMRS concept dictionary suggest the possibility of a mapping. Upper level concepts of SNOMED CT missing from the recommended classes in an OpenMRS concept dictionary include Observable entity, Organism and Substance, whereas the concepts Diagnosis, Question and Symptom are missing the other way around.

As stated above, the concept classes in the concept dictionary of an OpenMRS application are only recommended in the guidelines; there is nothing to stop one from populating the concept dictionary with concepts from SNOMED CT. This could address the mismatch in one direction, but not the other way around.

Another issue is that SNOMED CT is a large and cumbersome ontology, and since OpenMRS applications generally only store information about medical interventions of limited scope, it makes sense to only link a part of SNOMED CT to an OpenMRS application. In particular, we decided to extract a module from SNOMED CT dealing specifically with TB, and link it to the concept dictionary of a simple OpenMRS application dealing only with TB patients. (By a *module*, we mean a sub-ontology that only uses a subset of the terminology of the main ontology, but that preserves the meaning of the terminology [36].) The smaller scale of this problem would also make it easier to evaluate the process.

We foresaw that some adaptation of the extracted module would be needed in order to match the concepts in the OpenMRS dictionary and/or to address local issues of multi-drug resistent and extreme drug resistant (MDR and XDR) TB.

6 FINDINGS

6.1 Experiences of extracting a module from SNOMED CT

We considered two approaches for extracting the module: (i) to use the ProSÉ plugin [37] for the Protégé ontology editor [38], and (ii) to use the module extraction facility provided by the CEL reasoner [39].

The version of SNOMED CT we had access to was in KRSS format. There are a number of different versions of KRSS syntax used by different ontology softwares. For example, Protégé can convert files from a particular KRSS format to (its native) OWL format, and the CEL reasoner accepts ontologies in KRSS format of a different syntax. Some syntax massaging of the version of SNOMED CT that we were in possession of was required to make it readable by these programs.

Two other problems that we experienced with Protégé were that the reasoners that could be used with it at that stage only supported description logics like $\mathcal{SROIQ}$ and $\mathcal{SHOIN}$ which are far more expressive than $\mathcal{EL}^+$, the underlying description logic in which SNOMED CT is defined. (The CEL reasoner can now be used with Protégé, see [36].) There were also memory problems of loading SNOMED CT into Protégé, since it is such a massive ontology.

We had more success with the CEL reasoner which was specifically designed to work with ontologies defined in $\mathcal{EL}^+$ and expressed in KRSS format, and could also be used to extract modules. The module we extracted from SNOMED CT about TB using the CEL reasoner is provided in the Appendix.

The problem with this module is that it only contains the superconcepts of Tuberculosis, not the subconcepts. To be usable as an ontology for linking to the concept dictionary of an OpenMRS application, this ontology would have to be expanded to include many of the relevant subconcepts of Tuberculosis, for example, ActiveTuberculosis, ChronicTuberculosis, DrugResistantTuberculosis etc.

We did not proceed with extracting a more comprehensive TB module due to some problems that had become apparent during the process. These are discussed below.

6.2 Pros and cons of SNOMED CT

As mentioned above, SNOMED CT has numerous strengths that make it the medical ontology of choice for this enterprise. It is an international standard, and for this reason it is good for semantic interoperability. SNOMED CT is also a logic-based ontology and is therefore eminently suitable for reasoning. A number of efficient reasoners have been developed for processing and reasoning over SNOMED CT.

SNOMED CT has two major disadvantages, however, namely its design legacy and its lack of support for instance data.

6.2.1 Design legacy

SNOMED CT has undergone numerous reincarnations in its development. Since the expressiveness of the underlying logic was restricted by the availability of reasoners (during the early stages of the development of SNOMED CT) that could operate effectively over the ontology, restrictions were placed on what knowledge could be expressed. A particular problem was to express certain part-whole relations, particularly for describing parts of the anatomy. For example, the finger is part of the hand and the hand is part of the arm. From this we would like to be able to infer that the finger is part of the arm without having to explicitly state it.

Such part-whole reasoning requires transitive relations (i.e. from $R(a, b)$ and $R(b, c)$ infer $R(a, c)$) which were not available in the reasoner being used. A clever trick called *SEP (Structure, Entire, Part) triplets* was introduced by [40] to allow transitive relations to be expressed in the ontology without implementing them in the reasoner. Here is an example of SEP triplets being used to express the transitivity of the part-of relation of fingers, hands and arms:

$\mathsf{Arm} \sqsubseteq \mathsf{ArmS}$
$\mathsf{ArmP} \sqsubseteq \mathsf{ArmS} \sqcap \exists \mathsf{partOf.Arm}$
$\mathsf{HandS} \sqsubseteq \mathsf{ArmP}$
$\mathsf{Hand} \sqsubseteq \mathsf{HandS}$
$\mathsf{HandP} \sqsubseteq \mathsf{HandS} \sqcap \exists \mathsf{partOf.Hand}$
$\mathsf{FingerS} \sqsubseteq \mathsf{HandP}$
$\mathsf{Finger} \sqsubseteq \mathsf{FingerS}$
$\mathsf{FingerP} \sqsubseteq \mathsf{FingerS} \sqcap \exists \mathsf{partOf.Finger}$

As shown here, this requires the introduction of two additional (S and P) concepts for each concept which needs to participate in the partOf relation.

After the 'SEP-triplification' of SNOMED CT, Suntisrivaraporn et al [41] developed a reasoner that could work with transitive relations and showed that it could do so without any additional complexity (i.e. without the algorithm requiring any appreciably additional time or space). By specifying that the part-of relation is transitive (with a statement like (transitive partOf)), the above statements could be expressed simply as follows:

$\mathsf{Finger} \sqsubseteq \exists \mathsf{partOf.Hand}$
$\mathsf{Hand} \sqsubseteq \exists \mathsf{partOf.Arm}$

From this, $\mathsf{Finger} \sqsubseteq \exists \mathsf{partOf.Arm}$ could be inferred.

Despite this breakthrough, the damage had been done. Unfortunately researchers have been unable to automate the expunging of SEP triplets in a safe way (without affecting the relationships between other terms) – it has to be done manually. As it stands now, SNOMED CT is still riddled with redundant SEP triplets.

Although this problem is not evident in the module which we extracted (as given in the Appendix), any more extensive module that would (need to) be extracted that refers to any body structure (for example, the lungs or the alveoli) would involve SEP triplets. In fact, just for the concept Lung, SNOMED CT currently has the concepts EntireLung, LungPart and LungStructure for this purpose. Transitivity of relations for the TB module would be necessary to be able to infer, for example, that the alveoli are part of the lungs, and that infection of the alveoli would imply infection of the lungs.

Considerable reworking of the more extensive module would be necessary to get rid of SEP triplets.

6.2.2 Instance data

SNOMED CT is a list of clinical terms which refer to types of diseases, parts of the body, drugs, etc. in general terms. It is not designed or intended to express knowledge about specific patients, specific measurements or specific interventions performed at specific times.

In ontologies based on some description logic, instance data is stored in the form of *assertional* statements about individuals. For example,

$\mathsf{Patient(P123)}$
$\mathsf{infectedWith(P123, DrugResistantTuberculosis)}$

To be able to infer that P123 is a TB patient, there would need to be *terminological* statements like

$\mathsf{TBPatient} \sqsubseteq \mathsf{Patient} \sqcap \exists \mathsf{infectedWith.Tuberculosis}$
$\mathsf{DrugResistantTuberculosis} \sqsubseteq \mathsf{Tuberculosis}$

SNOMED CT only contains terminological statements to define medical terminology. It has no assertional statements, and no terminological statements that define or use the concepts needed for such assertional statements.

Another important aspect of patient data is the necessity to express negation, for example when a particular condition is ruled out by the results of a test.

As stated in Section 2.2, negation is not expressible in $\mathcal{EL}^+$, the underlying DL of SNOMED CT.

This 'disadvantage' of SNOMED CT would only be an issue if one wanted to use the ontology to integrate with the rest of an OpenMRS database (as explained in Section 2.3), i.e. to allow reasoning over instance data in the form of patient records.

7 RECOMMENDATIONS

We are convinced of the advantages of enhancing OpenMRS with some (richer) ontology. As argued above, some adaption of SNOMED CT or at least of a module of it would be needed for this purpose.

An important decision would be whether one wanted to allow reasoning over the clinical terms (i.e. the medical terminology) stored in the concept dictionary alone, or over the patient records (i.e. the instance data) stored in the rest of the database as well. We envisage three possible options:

Use an ontology that only defines and allows reasoning over the clinical terms. This is the simplest option, since one wouldn't have to worry about modelling the instance data. It would simply require cleaning up a module extracted from SNOMED CT and perhaps adding concepts needed for the missing (recommended) classes. If the SNOMED CT module were left unaltered (i.e. with its SEP triplets and other redundant concepts), another advantage would be its compatibility with the current state of SNOMED CT, allowing interoperability with other systems that use SNOMED CT.

Use two ontologies: one for reasoning over the clinical terms, and another for reasoning over the patient records. The first ontology could be the one developed for the first option above. The second ontology could be created in a different (more or alternatively expressive) description logic. For example, if the underlying DL of the second ontology were *DL-Lite*, one could reason over the patient records by means of the technology discussed in Section 2.3.

One disadvantage would be that interoperability could only be achieved with other systems that use compatible ontologies.

Another disadvantage would be if one wanted to integrate the two ontologies for performing reasoning which involved both the clinical terms and instance data, e.g. to perform the type of reasoning about patients illustrated in Section 6.2.2. Some form of ontology integration or ontology mapping would have to be employed, with the added complexity of having to deal with formalisms of different expressivity (see Section 2.1).

Create a new, combined ontology for reasoning over both clinical terms and patient records. The main advantage of doing things together would be that the ontology could be used for reasoning over both clinical terms and patient records. In other words, the problem of ontology integration or mapping of second option would be avoided.

A disadvantage would once again be that interoperability could only be achieved with other systems that use a compatible ontology.

One could consider developing the entire ontology in something like *DL-Lite*, but this would prevent much of the type of reasoning over clinical terms that is possible in SNOMED CT. Some work has been done to allow instance data with SNOMED CT, and to reason over it. The SHER reasoner is designed to reason over large ABoxes (i.e. large collections of assertional statements) [42], and has been used to reason over patient data together with SNOMED CT [43]. Alternatively, one could consider using one of the highly expressive DLs like $\mathcal{SROIQ}$. Since one would be dealing with a much smaller ontology than the entire SNOMED CT, acceptable response times should be obtained.

8 CONCLUSION

In this paper we have documented an attempt to enrich an OpenMRS application with the SNOMED CT medical ontology. The main reason for this enterprise was to allow reasoning over the health information stored in such a system, that is not possible with the database technology currently used by the OpenMRS framework. Although we did not complete the planned implementation, we gained a number of insights into the process that will be useful for anyone attempting to do something similar. In summary, we contend that SNOMED CT in its unaltered form is not suitable for linking to an OpenMRS application. A module extracted from SNOMED CT would be more suitable, and this would further need to be refined and adapted to suit the concept dictionary of the particular OpenMRS application. Various strategies are possible, as outlined in Section 7. These primarily depend on whether one would want to be able to reason over patient records in addition to the clinical terms.

ACKNOWLEDGEMENTS

This research was funded by the joint SA-Italy collaboration agreement entitled *Technologies for conceptual modeling and intelligent query formulation* [44]. Thanks also to Tommie Meyer of the Knowledge Systems Group in the Meraka Institute of the CSIR for ideas, infrastructure and support.

APPENDIX

As stated in Section 6.1, the version of SNOMED CT we were in possession of was in KRSS format. Furthermore, all the concepts were expressed as codes called *concept IDs* rather than their names or descriptions. For example, the concept ID for `Tuberculosis` is `56717001`. To extract a module about TB, we loaded SNOMED CT into the CEL reasoner and issued the command `(extract-c-module 56717001)`, which produced output that looked as follows:

```
(define-primitive-role 47429007)
```

```
(define-primitive-role 246075003 :parent 47429007)
(define-primitive-concept 138875005 top1)
(define-primitive-concept 410607006 138875005)
(define-primitive-concept 404684003 138875005)
etc ...
```

We used the *concepts table* of SNOMED CT to translate the concept IDs into their equivalent concept names/descriptions:

```
(define-primitive-role associated_with)
(define-primitive-role has_causative_agent :parent associated_with)
(define-primitive-concept Snomed_CT_concept top1)
(define-primitive-concept Organism Snomed_CT_concept)
(define-primitive-concept Clinical_finding Snomed_CT_concept)
(define-primitive-concept Microorganism Organism)
(define-primitive-concept Infectious_agent Organism)
(define-primitive-concept Disease Clinical_finding)
(define-primitive-concept Kingdom_Procaryote Microorganism)
(define-primitive-concept Domain_Bacteria
     (and Infectious_agent Kingdom_Procaryote))
(define-primitive-concept Bacterium
     (and Infectious_agent Domain_Bacteria Microorganism))
(define-primitive-concept Form-bacillus Bacterium)
(define-primitive-concept Gram-positive_bacterium Bacterium)
(define-primitive-concept Gram-positive_bacillus
     (and Gram-positive_bacterium Form-bacillus))
(define-primitive-concept Gram-positive_branched_filamentous_hyphae Gram-positive_bacillus)
(define-primitive-concept Aerobic_bacteria Domain_Bacteria)
(define-primitive-concept Acid-fast_bacillus
     (and Aerobic_bacteria Form-bacillus))
(define-primitive-concept Phylum_Actinobacteria
     (and Gram-positive_bacillus Domain_Bacteria))
(define-primitive-concept Class_Actinobacteria Phylum_Actinobacteria)
(define-primitive-concept Subclass_Actinobacteridae Class_Actinobacteria)
(define-primitive-concept Order_Actinomycetales Subclass_Actinobacteridae)
(define-primitive-concept Suborder_Corynebacterineae Order_Actinomycetales)
(define-primitive-concept Family_Mycobacteriaceae
     (and Acid-fast_bacillus Suborder_Corynebacterineae))
(define-primitive-concept Aerobic_Actinomycetes Actinomycetes)
(define-primitive-concept Actinomycetes Gram-positive_branched_filamentous_hyphae)
(define-primitive-concept Mycobacterium
     (and Family_Mycobacteriaceae Aerobic_Actinomycetes))
(define-primitive-concept Non-photochromogenic_mycobacteria Mycobacterium)
(define-primitive-concept Slow_growing_mycobacteria Mycobacterium)
(define-primitive-concept Mycobacterium_tuberculosis_complex
     (and Slow_growing_mycobacteria Non-photochromogenic_mycobacteria))
(define-concept Infectious_disease
     (and Disease
          (some roleGroup
               (and (some
                    has_causative_agent
                    Infectious_agent)))))
(define-concept Mycobacteriosis
     (and Bacterial_infectious_disease
          (some roleGroup
               (and (some
                    has_causative_agent
                    Mycobacterium)))))
(define-concept Bacterial_infectious_disease
     (and Infectious_disease
          (some roleGroup
               (and (some
                    has_causative_agent
                    Domain_Bacteria)))))
(define-concept Disease_due_to_Gram-positive_bacteria
     (and Bacterial_infectious_disease
          (some roleGroup
               (and (some
                    has_causative_agent
                    Gram-positive_bacterium)))))
(define-concept Tuberculosis
     (and Mycobacteriosis
          (some roleGroup
               (and (some
                    has_causative_agent
                    Mycobacterium_tuberculosis_complex)))))
```

REFERENCES

[1] T. R. Gruber. "Ontology". In L. Liu and M. T. Özsu (editors), *Encyclopedia of Database Systems*. Springer-Verlag, 2009. URL http://tomgruber.org/writing/ontology-definition-2007.htm.

[2] T. Berners-Lee, W. Hall, J. A. Hendler, K. O'Hara, N. Shadbolt and D. J. Weitzner. "A framework for web science". *Foundations and Trends in Web Science*, vol. 1, no. 1, 2006.

[3] Snomed CT, 2010. URL http://www.ihtsdo.org/snomed-ct/.

[4] OpenMRS, 2010. URL http://www.openmrs.org/.

[5] T. R. Gruber. "A translation approach to portable ontology specifications". *Knowlegde Acquisition*, vol. 5, no. 2, pp. 199–220, 1993.

[6] F. Baader, D. Calvanese, D. McGuinness, D. Nardi and P. Patel-Schneider (editors). *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, Cambridge, UK, 2003.

[7] M. Horridge and P. Patel-Schneider. "Manchester Syntax for OWL 1.1". In *Proceedings of OWL: Experiences and Directions, 4th international workshop (OWLED 2008 DC)*. 2008. URL http://www.webont.org/owled/2008dc/papers/owled2008dc_paper_11.pdf.

[8] P. Patel-Schneider and B. Swartout. "Description-Logic Knowledge Representation System Specification from the KRSS Group". Available on internet, 1993. URL http://www-db.research.bell-labs.com/user/pfps/papers/krss-spec.ps.

[9] L. Obrst. "Ontologies for semantically interoperable systems". In *Proceedings of the twelfth international Conference on Information and Knowledge Management (CIKM '03)*. ACM, 2003.

[10] D. Calvanese, G. De Giacomo and M. Lenzerini. "A Framework for Ontology Integration". In I. Cruz, S. Decker, J. Euzenat and D. McGuinness (editors), *The Emerging Semantic Web: Selected Papers from the First Semantic Web Working Symposium*. 2002.

[11] Y. Kalfoglou and M. Schorlemmer. "Ontology mapping: the state of the art". *Knowl. Eng. Rev.*, vol. 18, no. 1, 2003.

[12] V. N. Stroetman, D. Kalra, P. Lewalle, A. Rector, J. M. Rodrigues, K. A. Stroetman, G. Surjan, B. Ustun, M. Virtanen and P. E. Zanstra. *Semantic Interoperability for Better Health and Safer Healthcare*. European Commission, 2009.

[13] F. Baader, C. Lutz and B. Suntisrivaraporn. "Is tractable reasoning in extensions of the description logic EL useful in practice?" In *In Proceedings of the 2005 International Workshop on Methods for Modalities (M4M-05)*. 2005.

[14] D. Calvanese, G. De Giacomo, M. Lenzerini, R. Rosati and G. Vetere. "DL-Lite: Practical Reasoning for Rich DLs". In *Proceedings of the 2004 International Workshop in Description Logics (DL 2004)*, vol. 104. 2004.

[15] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, A. Poggi, M. Rodriguez-Muro and R. Rosati. "Ontologies and Databases: The DL-Lite Approach". In S. Tessaris and E. Franconi (editors), *Semantic Technologies for Informations Systems - 5th Int. Reasoning Web Summer School (RW 2009)*, vol. 5689 of *Lecture Notes in Computer Science*, pp. 255–356. Springer, 2009.

[16] C. Nyulas, M. O'Connor and S. Tu. "DataMaster a Plug-in for Importing Schemas and Data from Relational Databases into Protégé". In *Proceedings of 10th International Protégé Conference*. 2007.

[17] M. Laclavik. "RDB2Onto: Relational Database Data to Ontology Individual Mapping. In: Tools for Acquisition, Organisation and Presenting of Information and Knowledge. P.Navrat et al". pp. 86–89. 2006.

[18] A. Poggi, D. Lembo, D. Calvanese, G. De Giacomo, M. Lenzerini and R. Rosati. "Linking Ontologies to Data". *Journal on Data Semantics*, pp. 133–173, 2008.

[19] D. Calvanese, M. Lenzerini and D. Nardi. *Description logics for conceptual data modeling*. Kluwer Academic Publishers, 1998.

[20] ICom, 2010. URL http://www.inf.unibz.it/∼franconi/icom/.

[21] C. Necib and J. Freytag. "Using ontologies for database query reformulation". In *ADBIS (Local Proceedings)*. 2004.

[22] C. P. de Laborda and S. Conrad. "Relational.OWL: a data and schema representation format based on OWL". In *APCCM '05: Proceedings of the 2nd Asia-Pacific conference on Conceptual modelling*, pp. 89–96. Australian Computer Society, Inc., Darlinghurst, Australia, Australia, 2005.

[23] N. Cullot, R. Ghawi and K. Yetongnon. "DB2OWL: A Tool for Automatic Database-to-Ontology Mapping". In *Proceedings of the 15th Italian Symposium on Advanced Database Systems*. 2007.

[24] N. Konstantinou, D.-E. Spanos, M. Chalas, E. Solidakis and N. Mitrou. "VisAVis: An Approach to an Intermediate Layer between Ontologies and Relational Database Contents". In *Proceedings of the International Workshop on Web Information Systems Modeling*. 2006.

[25] J. Barrasa, . Corcho and A. Gmez-Prez. "R2O: An extensible and semantically based database-to-ontology mapping language". In *Proceedings of the Workshop on Semantic Web and Databases*, pp. 1069–1070. Edinburgh, Scotland, 2004.

[26] C. Bizer. "D2R MAP A Database to RDF Mapping Language". In *12th International World Wide Web Conference, Budapest*. 2003.

[27] C. Bizer and A. Seaborne. "D2RQ: Treating Non-RDF Databases as Virtual RDF Graphs". In *Proceedings of the 3rd International Semantic Web Conference (ISWC2004)*. 2004.

[28] J. Dolby, A. Fokoue, A. Kalyanpur, E. Schonberg and K. Srinivas. "Scalable highly expressive reasoner (SHER)". *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 7, no. 4, pp. 357–361, 2009.

[29] OBDA, 2010. URL http://obda.inf.unibz.it/protege-plugin/.

[30] M. Rodriguez and D. Calvanese. “Towards an Open Framework for Ontology Based Data Access with Protégé and DIG 1.1”. In *OWLED*. 2008.

[31] M. Rodriguez-Muro, L. Lubyte and D. Calvanese. “Realizing Ontology Based Data Access: A Plug-in for Protégé”. In *Proc. of the ICDE Workshop on Information Integration Methods, Architectures, and Systems (IIMAS 2008)*, pp. 286–289. IEEE Computer Society Press, 2008.

[32] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, A. Poggi and R. Rosati. “Linking data to ontologies: The description logic DL-LiteA”. In *Proceedings of OWLED 2006*. 2006.

[33] A. Artale, D. Calvanese, R. Kontchakov and M. Zakharyaschev. “The DL-Lite Family and Relations”. *Journal of Artificial Intelligence Research*, vol. 36, pp. 1–69, 2009.

[34] Sher, 2010. URL
`http://www.alphaworks.ibm.com/tech/sher`.

[35] I. Horrocks, B. Parsia, P. Patel-Schneider and J. Hendler. “Semantic Web Architecture: Stack or Two Towers?” *Lecture Notes in Computer Science: Principles and Practice of Semantic Web Reasoning: Third International Workshop*, vol. 3703 / 2005, 2005. ISSN 0302-9743. doi:10.1007/11552222.

[36] J. Mendez and B. Suntisrivaraporn. “Reintroducing CEL as an OWL 2 EL Reasoner”. In *Proceedings of the 22nd International Workshop on Description Logics (DL2009)*. 2009.

[37] ProSÉ, 2010. URL `http://protege.stanford.edu/`.

[38] Protégé, 2010. URL
`http://protege.stanford.edu/`.

[39] Cel, 2010. URL
`http://lat.inf.tu-dresden.de/systems/cel/`.

[40] S. Schulz, M. Romacker and U. Hahn. “Part-whole reasoning in medical ontologies revisited: Introducing SEP-triplets into classification-based description logics”. In C. G. Chute (editor), *Proceedings of the 1998 AMIA Annual Fall Symposium*. 1998.

[41] B. Suntisrivaraporn, F. Baader, S. Schulz and K. Spackman. “Replacing SEP-triplets in Snomed CT using Tractable Description Logic Operators”. In *Proceedings of the 11th conference on Artificial Intelligence in Medicine (AIME ’07)*. 2007.

[42] K. Srinivas. “OWL reasoning in the Real World: Searching for Godot”. In *Proceedings of the 22nd International Workshop on Description Logics (DL2009)*. 2009.

[43] C. Patel, J. J. Cimino, J. Dolby, A. Fokoue, A. Kalyanpur, A. Kershenbaum, L. Ma, E. Schonberg and K. Srinivas. “Matching Patient Records to Clinical Trials Using Ontologies”. In *Proceedings of ISWC/ASWC 2007*. 2007.

[44] T. Meyer. “Progress report, South Africa/Italy Collaboration, UID-65152”, 2008.