Grant 1R21LM009824-01A1 from the National Library of Medicine

Realism-based versioning for biomedical ontologies

Final Report

June 21, 2011

Werner Ceusters

University at Buffalo, Department of Psychiatry
Ontology Research Group (ORG), New York State Center of Excellence in Bioinformatics & Life Sciences
701 Ellicott Street, suite B2-160
Buffalo NY 14203
USA

ceusters@buffalo.edu
office: (716) 881-8971
mobile: (716) 418-4237

Table of Contents

1 Executive Summary

1.1 Vision and research goals

The broad, long-term vision underlying our research over the past six years is one in which representational artifacts designed for use in software applications mimic the structure of reality to the best understanding of their authors. This holds for artifacts that represent generic information such as classification systems, terminologies and ontologies as well as for data repositories such as electronic health records and data warehouses. And it holds not only for what is believed to be the case today, but also for how matters have been in the past.

The research carried out under this grant aimed to advance the state of the art in version management of biomedical terminologies and ontologies. For these systems to fit in the vision, they should thus not just reflect the state of the art in biomedical science in terms of what entities exist in reality and of how they are related, but also keep track of whether the changes introduced in successive versions of it reflect (1) changes in the underlying reality, or (2) in the views of the artifacts' authors – or in associated scientific theories that the authors endorse, or (3) are corrections of editorial mistakes. By applying this view, which we call *realism-based ontology versioning* because of its foundations in a computable form of philosophical realism, biomedical ontologies and terminologies will become similar to biomedical textbooks in that they reflect the state of the art in the domain but - in contrast to textbooks - in a way which is interpretable by software algorithms.

To make this vision come true, we must have objective measures for ontology quality. Our hypothesis is that when this vision is endorsed by the authors of biomedical ontologies and when the ontology authoring environments support the application of this view by means of appropriate software, it would become possible to develop an objective measure for the quality of an ontology as it evolves over time.

Under this grant, we tested this hypothesis by trying to apply realism-based ontology versioning to the Systematized Nomenclature of Medicine - Clinical Terms (SNOMED CT), a reference terminology for which the authors are trying to provide an ontological basis.

1.2 Progress towards originally stated aims

Our plan was to test our hypothesis through achievement of the following specific aims:

Aim 1: analyze SNOMED CT's existing history mechanism to find out whether the principles of realism-based ontology versioning are able to cope with all requirements put forward by SNOMED CT. Adjust when needed.

Aim 2: develop a prototype of a realism-based ontology versioning software component that can serve as plug in for ontology authoring systems such as Protégé, ODE or SWOOP.

Aim 3: use the prototype to restructure SNOMED CT's history information in line with the principles of realism-based ontology versioning.

Aim 4: compute the quality improvement of SNOMED CT over time in order to demonstrate the usefulness of the approach and foster its acceptance in other ontologies.

All aims have been achieved completely, although we had to readjust the detailed work plan as initially conceived in terms of seven specific tasks to meet our final objectives. The reasons for this were:

1. unclarity about what SNOMED CT concepts exactly denote;
2. although the principles of realism-based ontology versioning were found to cope with SNOMED CT's requirements (aim 1), the opposite turned out not to be the case: SNOMED CT's history mechanism, and in particular its own 'reasons for change' as coded in SNOMED CT distributions do not provide enough information to allow third parties to translate these reasons for change into the various change configurations recognized by realism-based ontology versioning (aim 3);
3. SNOMED CT's absence of version management for the relationships table has the consequence that when some relationship is present in some versions and not in others, it cannot be assessed whether the absence corresponds to a real absence or an implicit presence inferable through description logic reasoning. These computations were not feasible with the technology publicly available in the course of the project.

These findings introduce some error in the quality assessment of SNOMED CT over time as we have been able to compute (aim 4), errors which can however be eliminated completely when SNOMED CT's version management policies would be adjusted according to our recommendations.

1.3 Results

The following table summarizes our results in terms of the work plan as originally conceived. Details about these results are provided in additional sections of this report and papers published as a result of this grant.

Tasks	***Aim***	***Results***
1	1	Data collection and preparation: • we obtained all SNOMED CT US distributions from January 2002 until July 2010, • we wrote Extract-Transform-Load (ETL) routines to create a relational database suitable for our analysis needs.
2	1	Statistical data analysis: • we computed lists of concepts exhibiting the highest number of changes over time. They turned all out to be candidates for post-coordination which is a useful finding itself, but not in light of our goals. • we analyzed a sample of 1,000 randomly selected concepts (n=264) and descriptions (n=736) that underwent a status change of some sort, the goal being to find underlying principles to translate automatically SNOMED CT's 'reasons for change' to our realism-based change configurations. This produced useful results but with some caveats. • we created in addition a subset composed of (1) 883 SNOMED CT concepts used within a cancer clinic for encoding synoptic pathology reports and tumor registry data and for querying a bio-specimen repository, all together covering almost 16,000 occurrences related to 10,000 unique patients, and (2) 1,415 concepts present in the transitive closure set of the former by means of 15,689 relationships. The analysis produced useful statistics to decide on the basis of the history information whether users should upgrade to a new version of SNOMED CT.
3	1	Detailed analysis of SNOMED CT's history mechanism: • we created graphing software and produced various sorts of grafts

		showing the extremely complex change history of concepts retrieved on the basis of either (1) key phrases or (2) homonymic terms. • we compared SNOMED CT's current history mechanism and our novel method based on Ontological Realism and outlined ambiguities and areas of missing information. • we created a Semantic Wiki as a prototype example of how the two mechanisms can be combined. • we improved and expanded our methodology to represent more accurately not only - as originally aimed for - what SNOMED CT authors must believe to have changed in reality or in their understanding with respect to the *last* version, but with respect to *all* previous versions.
4	2	Requirements specifications for the realism-based versioning prototype: • we described the functionalities and procedures that have to be implemented in a prototype that is able to support realism-based ontology versioning based upon the improved methodology arrived at in task 3.
5	2	Prototype development: • we implemented the functionalities and procedures identified in task 4 as Java classes in a Web service architecture. The services interface is provided through the RESTful Web Services architecture. Each RESTful web service can be invoked through an http client library, which is available in all programming languages either as built-in or third party implementations.
6	3	Applying realism-based ontology versioning to SNOMED CT: • we used the principles for history mapping developed under task 2 and 3 to generate a history view of SNOMED CT compatible with our new method.
7	4	Measuring quality improvements in SNOMED CT: • we used the view developed in task 6 to compute the believed quality improvements of SNOMED CT since its inception. • we found that under our view the quality of SNOMED CT between the January 2002 and July 2009 versions increased for concepts by 18.8%, for descriptions by 47.7% and for relationships by 178.1 % under the assumption that the July 2009 version were accurate.

1.4 List of Publications

- Ceusters W. SNOMED CT Revisions and Coded Data Repositories: When to Upgrade? (Accepted for AMIA 2011)
- Ceusters W. *SNOMED CT's RF2: is the Future Bright?* Medical Informatics Europe Conference, MIE 2011, Oslo, Norway, August 28-31, 2011 (in press)
- Ceusters W, Capolupo M, De Moor G, Devlies J, Smith B. *An Evolutionary Approach to Realism-Based Adverse Event Representations.* Methods of Information in Medicine, 2011;50(1):62-73.
- Ceusters W, Smith B. *Foundations for a realist ontology of mental disease.* Journal of Biomedical Semantics, 2010, 1:10 (9 December 2010).
- Ceusters W. *Applying Evolutionary Terminology Auditing to SNOMED CT*. In American Medical Informatics Association 2010 Annual Symposium (AMIA 2010) Proceedings, Washington DC, November 13-17, 2010:96-100.

- Ceusters W, Smith B. *A Unified Framework for Biomedical Terminologies and Ontologies*. Proceedings of the 13th World Congress on Medical and Health Informatics (Medinfo 2010), Cape Town, South Africa, 12-15 September 2010. Studies in Health Technology and Informatics 2010;160:1050-1054. (PMID: 20841844)
- Smith B, Ceusters W. Ontological Realism as a Methodology for Coordinated Evolution of Scientific Ontologies. Applied Ontology, 2010;5(3-4):139-188.

1.5 Generated resources

The following materials - other than papers - have been produced in the course of the project and will be used for further research in line with our global vision. They are also available 'as is' to other researchers upon request, those marked with '*' depending on approval of the copyright holder of SNOMED CT since they make use of the various distributions that have been released over time.

- Homonym collection: over 48,000 graphs displaying SNOMED CT's use of homonyms over the period January 2002 - July 2010 and the historical semantic neighborhood of the concepts involved in terms of the is a links and concept history attributes as well as the corresponding source files in the DOT language used by GraphViz to generate these graphs.
- SNOMED CT RT-Wiki: rendering of the various versions of SNOMED CT which allows to annotate changes with realism-based versioning information.
- Functional and technical specifications document for realism-based ontology evolution tracking.
- Referent Tracking based ontology/history tracker: generic web services software implementing realism-based evolution tracking for ontologies or terminologies.

1.6 Recommendations

In 2010, the International Health Terminology Standards Development Organization (IHTSDO) announced the future distribution of SNOMED CT under a new format called *'RF2'* of which more detail became officially available with the January 2011 version. This format, at first sight, seems to hold much promises to deal with a number of issues concerning the ontological underpinnings of SNOMED CT and the version management implemented thus far.

Our suggestions are:

1. do not make double use of the ConceptID as an identifier for the concept and an identifier for the Concept Component;
2. add to each Concept Component a field that indicates to what broad category the intended referent of that concept belongs;
3. expand the Concept Inactivation Value sub-hierarchy with concepts that reference whether a change in SNOMED CT is motivated by (1) a change in reality, (2) the SNOMED CT authors' or users' understanding of reality as reflected in the advance of the state of the art in the biomedical domain, or (3) a mistake that is strictly internal in SNOMED CT as an information artifact [1], and this along the lines described in section 9 page 53.
4. add mechanisms:
    a. to represent the provenance of a class more explicitly;
    b. to separate the time-period during which a component is believed to have been valid in SNOMED CT from the period it is believed to be (or has been) valid in reality since the latest release;

2 Introduction

The work described in this report is a logical continuation of the research initiated by the PI in the early nineties, which aims:

(1) to bring unconstrained natural language understanding up to a level that it can be used for man-machine communication and

(2) to design software that is able to make data semantically interoperable for automated decision support.

This research has primarily been focused around methods and techniques for overcoming the burdens associated with traditional paradigms for structured documentation in electronic patient records [2-7]. Central to our earlier work is the vision that, to understand natural language and structured patient data, software programs must incorporate knowledge about how the world is structured, how this structure is perceived by humans, and how humans communicate about it [8-10].

We found that ontologies, primarily those based on sound philosophical theories, are essential components for providing this sort of knowledge, and in such a way as to do justice to the difference concerning what is the case and what is known or believed to be the case [11].

The word ‘ontology’ is used for various types of artifacts created and used in different communities to represent those entities and relationships salient to a given domain. Such artifacts range from formal upper-level ontologies expressed in first order logic to the simple user-defined keyword lists used, for example, to annotate resources on the Web. In between are taxonomies and controlled vocabularies such as MeSH, often used for information indexing and retrieval, and whose organization is primarily hierarchical, as well as ontologies and vocabularies which represent also non-hierarchical relationships such as the Foundational Model of Anatomy [12-14], SNOMED-CT [15-18] and the NCI Thesaurus [19-24].

For an ontology to maintain its usefulness, it must be updated at regular intervals. In domains where new knowledge is accumulated at a high rate, updates are sometimes published on a daily basis, the most conspicuous example being the Gene Ontology [25-28]. However, when new versions of ontologies are released, the changes made at each stage are either not documented, or information is limited to which entries in the ontology appeared, disappeared, or became fused or split. Only in very rare cases is information provided about the reasons for the changes made.

2.1 Vision

The **broad, long-term vision** behind our project is one in which biomedical ontologies do not just reflect the state of the art in biomedical science in terms of what entities exist in reality and of how they are related, but that they also keep track of whether the changes introduced in successive versions of ontologies reflect (1) changes in the underlying reality, (2) in the views of ontology authors – or in associated scientific theories, or (3) are corrections of editorial mistakes. By applying this view, which we call *realism-based ontology versioning* because of its foundations in a computable form of philosophical realism, ontologies will become similar to biomedical textbooks in that they reflect the state of the art in the domain in a way which is interpretable by software algorithms.

To make this vision come true, especially in the context of ontologies that will work in complex domains such as biomedicine, we must have objective measures for ontology quality.

2.2 Hypothesis

Our hypothesis is that when this vision is endorsed by the authors of biomedical ontologies and when the ontology authoring environments support the application of this view by means of appropriate software, it would become possible to develop an objective measure for the quality of an ontology as it evolves over time.

2.3 Specific aims

We proposed to test this hypothesis by applying realism-based ontology versioning to the Systematized Nomenclature of Medicine (SNOMED) on the basis of the following **specific aims**:

- Aim 1: analyze SNOMED-CT's existing history mechanism to find out whether the principles of realism-based ontology versioning as we have outlined them in [13] are able to cope with all requirements put forward by SNOMED-CT. Adjust when needed.
- Aim 2: develop a prototype of a realism-based ontology versioning software component that can serve as plug in for ontology authoring systems such as Protégé, ODE or SWOOP.
- Aim 3: use the prototype to restructure SNOMED-CT's history information in line with the principles of realism-based ontology versioning.
- Aim 4: to compute the quality improvement of SNOMED-CT over time in order to demonstrate the usefulness of the approach and foster its acceptance in other ontologies.

2.4 Background and significance

2.4.1 Ontology versioning

An ontology is commonly defined as 'a shared and agreed upon conceptualization of a domain'. An ontology such as the UMLS Semantic Network correspondingly takes the form of a graph, whose nodes refer to concepts [29]. The combinations of nodes and edges in such a graph provide both concept descriptions and also, in the best case, concept definitions. Unfortunately, the documentation of such concept-based ontologies leaves insufficiently specified what concepts actually are, or to what, if anything, they might correspond in reality [30]. The result is that it is very hard to avoid mistakes in the development of such ontologies, and annotations made in their terms still suffer from ambiguities [15, 23, 31-35].

Of a different sort are those ontologies that are based on philosophical realism and require the nodes and edges in an ontology graph to correspond not to concepts but rather to entities in reality, for example to lesions or diseases on the side of the patient. Here the nodes in the graph refer to *universals* (such as *person*, *organ*, *liver*, *tumor*) which are instantiated by open-ended families of similar *individuals* (also called *particulars*, examples being the PI of this proposal, his liver, and so forth). The edges in the graph correspond to relationships between universals, as expressed in assertions such as: *liver* is_a *organ*, *human liver* part_of *human being*, and so on. Realism-based ontologies may then be used to annotate data about those particulars that instantiate the corresponding universals by means of assertions such as: patient #324 instance_of *person*.

Following a recently proposed terminology [36] we use the term 'portion of reality' (POR) to denote particulars, universals, and the simple and complex combinations thereof. Examples of ontologies conforming to realist principles are Basic Formal Ontology [37] (BFO) and DOLCE [38], and the same principles serve also as the basis for the Relation Ontology laid down by the Open Biomedical Ontologies consortium as part of its OBO Foundry development efforts [39-40].

2.4.2 Critical gaps in ontology versioning

Ontologies, when in use, evolve over time. Changes in ontologies are introduced to correct errors, to accommodate new information or to adjust the representation of the domain: classes may disappear, fuse with other classes, become split, and so forth. Hence there is a need for methods and means to manage the evolution of ontologies to ensure that applications using different versions of an ontology remain compatible with respect to each other and that the data annotated by means of different versions can still be compared and interpreted in the right way. Interestingly, the versioning means and methods currently used share at least two common defects, and this for both conceptualist and realist ontologies.

The first one is that current methodologies do not offer a metric that allows ontology authors or users to measure the improvements obtained in successive versions of an ontology. For *application ontologies* that are used in decision support systems for instance, an indirect metric might be obtained by testing the performance of the system in terms of recall and precision using some pre-established benchmark. But in case of *reference ontologies* whose purpose is to describe a domain, such benchmarks are not available.

The second defect is that when new versions of such ontologies are released, very little information is provided about the reasons for the changes made. As witnessed by three surveys [41-43] which review prior work on ontology versioning and evolution including research conducted by Oliver and Shahar [44], Stojanovic [45], and Noy and Musen [46], efforts in this area have focused thus far exclusively on techniques for keeping track of which entries in an ontology appeared, disappeared, became fused or split in successive versions. Because the question is not raised as to ***why*** such changes are made, crucial distinctions are missed between the different kinds of changes in an ontology, reflecting for example:

1. changes in the underlying reality (does the appearance or disappearance of an entry in a new version of an ontology relate to the appearance or disappearance of entities or of relationships among entities in reality?);
2. changes in our scientific understanding;
3. reassessments of what is considered to be relevant for inclusion in an ontology, or
4. encoding mistakes introduced during ontology curation (for example through erroneous introduction of duplicate entries reflecting lack of attention to differences in spelling).

That such differences are overlooked is no surprise in the case of concept-based ontologies: entities in reality are in these ontologies thought of as playing at best a secondary role, and so the associated reasoning machinery takes care only of *internal* consistency. But also ontologies based on realism have thus far neglected to record the reasons for change of the sorts just sketched.

2.5 Significance and relevance to health

Our research intends to address these gaps in the context of SNOMED-CT. We have SNOMED-CT chosen as our case study because it becomes more widely used as a reference terminology on an international scale, and therefore the need for quality assurance becomes ever more urgent.

We can think of the core components of SNOMED-CT as forming a graph structure, whose constituent nodes are joined together by *is_a* relations representing the fact that all instances of a given child concept are also instances of its parent concept. Concepts themselves are represented by the nodes of the graph, which in SNOMED-CT are called ‘classes’ and thus correspond with what we proposed to be called ‘*representational units*’. Classes are labeled with the concept identifier associated with the concept that the class represents. They are further associated with a variable number of elements such as their *relationships* to other classes and the *terms* – linked to the classes by means of *descriptions* – that can be used to refer to them by means of natural language. Whereas some terms can be used to refer to several classes (homonymy), there is always one term, called the '*fully specified name*' (FSN), which is unique, and consists of a regular name suffixed (in parentheses) with a reference to what SNOMED-CT calls the '*primary hierarchy*' of the class, the latter corresponding roughly to the top-level node of the including graph.

The content of SNOMED-CT evolves with each release, changes including the addition or deletion of classes, descriptions, and relationships. Changes are said to be '*driven by changes in understanding of health and disease processes; introduction of new drugs, investigations, therapies and procedures; new threats to health; as well as proposals and work provided by SNOMED partners and licensees*' [47]. A history mechanism keeps track of the changes over time. However, although the history mechanism does capture ***what*** changes have been introduced over time, it gives no reason as to ***why*** such changes were made, nor does it help us in assessing to what extent a specific release represents an improvement over its predecessors. If, for instance, a new disease class is added at a certain time, is this because (a) the disease denoted by the class did not exist earlier, or because (b) the disease has only recently been discovered? In case (a), both versions would be equally faithful to the corresponding parts of reality they were designed to represent; in case (b), the earlier version would be marked by the unjustified absence of the class that was added later.

3 Realism-based approaches to terminology and ontology

Concept-based terminologies consist of groups of terms, each such group being linked to a 'concept' that is said to define the meaning of the corresponding terms. We have argued that the inconsistent interpretations of the word 'concept' embraced by the creators and users of such terminologies have given rise to multiple distinct modeling practices, which in turn have given rise to inconsistent representations. [48-49]

Our identification of these problems – which are now acknowledged also by other experts in the field [23, 32, 50-52] – does not, however, imply that we dismiss traditional terminology resources as being without value. On the contrary, it is clear that the majority of these systems will continue to play an important role in the information-driven clinical and translational science of the future, and this for at least two reasons.

First, huge quantities of clinical and research data have already been annotated (and in some cases compiled *ab initio*) in their terms, and it cannot be expected that these data will be annotated a second time using realism-based ontologies that follow, for example, the Open Biomedical Ontology (OBO) Foundry [40] and are created *de novo*.

Second, where OBO Foundry ontologies seek to represent the entities on the side of reality, traditional terminology systems are designed to reflect the ways language is used by clinicians and others in reporting (for example) patient encounters.[9] This closeness to the needs of clinicians and healthcare institutions suggests that concept-based systems may still be in common use in the future.

The problem must be addressed, however, that the data resulting from such annotation efforts, precisely because they stay so close to the language used in specific disciplinary communities, and because they are affected by the multiple modeling paradigms associated with the orientation around 'concepts', are marked by the detrimental effects of silo formation. The widespread adoption of SNOMED CT would diminish such effects. But as long as SNOMED CT itself does not use a consistent ontological approach [32], we believe that the data expressed with its aid, too, will involve too high a degree of redundancy and of inconsistent coding [53].

SNOMED's structure does not as yet provide a consistently accessible and reliable representation of the reality on the side of the patient as this changes through time. Moreover, SNOMED in its current form will not be able to do justice in consistent fashion to the changes in our knowledge of this reality which will be brought by advances in translational science [54]. To address these problems we need a strategy to map legacy terminologies such as SNOMED CT to OBO Foundry ontologies in such a way as to ensure that both can contribute to the creation of the non-redundant common framework for data integration and exploitation that will be needed in the future. [55]

3.1 Introducing Ontological Realism

The realist orientation in biomedical terminology is based on the view that terms in terminologies are to be aligned not on '*concepts'* but rather on entities in reality [56-57]. Central to this view are three assumptions.

- The first is that biological reality exists *objectively* in itself, i.e. independent of the perceptions or beliefs of cognitive beings. Thus not only do a wide variety of entities exist in reality (human beings, stomachs, bacteria, disorders, ...), but also how these entities relate to each other (that certain stomachs are parts of human beings, that certain bacteria cause disorders in human beings, and so forth) is not a matter of agreements made by scientists but rather of objective fact.

- The second assumption is that reality, including its structure, is accessible to us and can be discovered: it is scientific research that allows human beings to find out what entities exist and what relationships obtain between them.
- The third assumption is that an important aspect of the quality of a terminology is determined by the degree to which the structure according to which the terms of the terminology are organized *mimics* the pre-existing structure of reality.

Realism-based terminology *development* was introduced into biomedical informatics some ten years ago as a means of detecting and avoiding the systematic mistakes characteristic of *concept-based* terminologies [19, 49, 58-59], mistakes which are not eliminated through the use of description logics or similar computational devices [60]. The Foundational Model of Anatomy [61] and the Gene Ontology (GO) [25] were among the early adopters of a realist methodology along these lines. The methodology acquired broader acceptance after it was used to develop the Relation Ontology [39] which was adopted as a quality requirement for inclusion of any ontology in the OBO Foundry [40].

The first ideas towards realism-based terminology *versioning* and *auditing*, in contrast to *development*, were proposed in 2006 as a means to assess how successive versions of terminologies and ontologies evolve over time [11].

3.2 Basics of Ontological Realism

Ontological Realism rests on three principal distinctions: [55]

1) between *generic* and *specific portions of reality (**POR**s)*,
2) between the various purposes that can be served by *definitions*, and
3) between three distinct levels of reality.

3.2.1 Generic versus specific portions of reality

The first distinction separates *generals* from *particulars*, or in other words it separates *generic portions of reality* (***GPR***) from *specific portions of reality* (***SPR***). While this distinction, like the remaining proposals outlined in this section, can be applied to both *continuants* (such as cells and organisms) and *occurrents* (such as lives and deaths), we shall concentrate here exclusively on the case of continuants.

Amongst the generic portions of reality are *universals* (***UNV***) and what we shall call *generic configurations* (***GCO***).

Universals are denoted by general terms such as ‘human being’, ‘president’, ‘nation’, ‘population’. Universals are *instantiated* by particulars such as President Obama, the USA, the inhabitants of Buffalo, which, respectively can be denoted by terms and phrases such as 'President Obama', 'the USA', 'the inhabitants of Buffalo' [36].

Generic configurations are configurations formed by *generic portions of reality* that stand in some relation to each other that can be represented by some statement. An example is the portion of reality represented by the statement ‘*cell membrane part_of cell*’. Here ‘*part_of*’ represents the generic *part_of* relation as described in the Relation Ontology [39]. Another example is the portion of reality represented by the sentence ‘*clinicians are human beings*’. Here the word ‘*are*’ denotes what we shall call the *subgroup* relation, which holds between *clinicians* and *human beings*.

Amongst the *specific portions of reality* (***SPR***) are, analogously, *particulars* (***PAR***) and *specific configurations* (***SCO***).

PARs are entities that exist only once and are confined in space and time. Examples are: Mary, Buffalo, and the World Health Organization (WHO). Some ***PAR***s are what linguists would describe as ‘named entities’, but the majority – a liver cell in Mary, the fracture in her leg, and so forth – are not.

Both specific and generic configurations are represented by statements. Each ***SCO*** involves at least one ***PAR*** that stands in some relation to something else, for example to another ***PAR***, as in the specific configuration represented by the statement ‘*Mary’s left leg part_of Mary*’. If Mary’s left leg is amputated, then the two ***PAR***s involved in this ***SCO*** may survive the amputation, but the ***SCO*** itself will cease to exist.

Particulars can be divided into *atomic particulars* (***APA***) and *groups* (***GRP***). An atomic particular is a ***PAR*** that constitutes a unity in the sense that it has a complete, spatially connected external boundary. Examples, again, are: Mary and Mary’s left leg. ‘*Atomic*’ is here not to be understood as implying that the entity in question is not further decomposable. If Mary’s left leg is amputated, then it may still exist, though not any more as part of Mary. Nor is it to be understood that anatomic particulars cannot themselves contain parts which are atomic (for example Mary herself contains parts which are her cells).

GRPs are entities denoted by generic terms such as ‘limb of vertebrate’, ‘limb of human being’, and even ‘limb of Mary’. Although the latter example will likely not be found in a terminology or ontology, terms of the same sort do occur, examples being ‘citizen of the United States’, ‘Nobel Prize winner’, ‘veteran of the Second World War’. Terms denoting ***GRP***s are typically formed via combination of smaller terms which themselves denote universals, particulars, or other ***GRP***s.

If Mary is a healthy human being, the entity denoted by the noun phrase ‘Mary’s limbs’ is an example of a group (***GRP***). Each of healthy Mary’s limbs is at the same time a *part* of Mary and a *member* of the corresponding ***GRP***. All members of a ***GRP*** at any given time are such as to exist at that time.

Among ***GRP***s, we distinguish further between, *bona fide groups* (***BGR***), *fiat groups* (***FGR***) and extensions (***EXT***) [62]. While these distinctions are by no means trivial, their correct understanding is important if we are to find coherent ways to manage the large families of terms (for example in SNOMED the family consisting of terms such as ‘absent leg’, ‘amputated leg’, ‘withered limb’, ‘absent bone in leg’, ‘limb amputee’, ‘amputation of lower limb’, ‘amputation of limb’), whose meanings are otherwise difficult to capture in a coherent way.

A bona fide group (***BGR***) is a group whose members are homogeneous, are causally linked together, and which is maximal in the sense that all causally linked entities of the relevant sort are members of the group. Examples are: Mary’s limbs, Mary’s cells, Mary’s molecules.

A fiat group (***FGR***) is a group that is demarcated by fiat, such as: left lungs of people currently in Buffalo, the left lungs of all the people now participating in clinical trial #77639.

At any time at which the ***BGR*** constituted by healthy Mary’s 4 limbs exists, a cognitive being may explicitly recognize the simultaneous existence of any combination of two or more of her limbs. Some of these combinations, for instance any group of 3 of her limbs, are distinct ***FGR***s, since they fall short of being maximal. The groups formed by her two arms and by her two legs, in contrast, are ***BGR***s. The relation between fiat subgroups of the bona fide group that is formed by Mary’s limbs is analogous to the relation between some proper part of Mary that is demarcated by fiat and Mary as a whole. There is a fiat boundary between healthy Mary’s left arm and the rest of Mary’s body in the region of her left shoulder.

To each continuant universal corresponds a group, called its *extension* (***EXT***), formed by all and only those particulars that are instances of that universal at any given time.

We also distinguish three major families of relations between entities in the categories just sketched:

- *<p, p>-relations*: from particular to particular (for example: ***Werner Ceusters' s brain*** being part of ***Werner Ceusters***);
- *<p, u>-relations*: from particular to universal (for example: ***Werner Ceusters*** being an instance of HUMAN BEING);
- *<u, u>-relations*: from universal to universal (for example: HUMAN BEING being a subkind of ORGANISM) [39].

3.2.2 The purposes of definitions

The second distinction recognizes three purposes which a *definition* of a representational unit may serve:

- P1: to specify the conditions that must be satisfied for a term to be an acceptable designator for a given entity in some given community. An example would be:

  chronic pain =def. *a pain that has been present for more than 3 months*

- P2: to specify what is characteristic of particulars that instantiate a certain universal, for instance: disorder =def. *a part of an organism which serves as the bearer of a disposition to pathological processes* [63]
- P3: to demarcate an ***FGR*** by specifying characteristics that certain members of its including ***BGR*** exhibit.

P1 definitions are essentially a matter of terminological decisions. The given definition excludes the use of the term 'chronic pain' for pains lasting less than 3 months. This does not mean, however, that a pain in a specific patient that has already lasted for 90 days *becomes* a chronic pain one day later. It was, in fact, a chronic pain already from the very beginning, even though this fact was unknown to any observer.

P2 and P3 definitions help in determining whether a given particular is to be classified in a given way. P2 does this at the level of universals, while P3 does it for ***GRP***s.

3.2.3 First-order entities versus representations

The third distinction concerns the *level of reality* at which the referent of some representation exists. Of importance here is the distinction between:

- L1. *first-order entities* such as patients, disorders, families,
- L2. *beliefs* in people's minds (including beliefs putatively about objects such as unicorns which do not in fact exist), and
- L3. *representations* in some publicly accessible medium, for instance a term in an ontology.

This distinction of three levels allows us to differentiate disorders and diseases from diagnoses. [63] Whereas disorders and diseases are L1-entities – they exist in first-order reality on the side of the patient – diagnoses belong to either L2 or L3 depending on whether they are formulated in, for instance, a clinician's mind, or in a paper describing a case study or in an entry in an Electronic Health Record. Diagnoses are *about* disorders and diseases.

Traditional concept-based terminologies tend to blur this distinction, a typical example being the National Cancer Institute Thesaurus (NCIT) whose authors confuse the two realms repeatedly by claiming, for instance, (1) that the NCIT is a '*principled representation of key cancer-related <u>concepts in areas such as</u> cancers, findings, drugs, therapies, …*' [24] (p31), assigning '*concepts'* thus to the realm of first-order reality that is *represented in* the NCIT, but also in the very same paper (2) that '*all metadata class and attribute names correspond to <u>concepts in NCI Thesaurus</u>*' (p31-32). Adepts of Object Oriented Analysis approaches are specifically vulnerable to these sorts of mistakes because of the over-broad use of the term '*object'* on which such approaches rest as for example in: '*An object is a representation of a real-life entity or abstraction. For example, objects in a flight reservation system might include: an airplane, an airline flight, an icon on a screen, or even a full screen with which a travel agent interacts*' [64]. This passage would have us believe that airplanes and travel agents are *representations* rather than vehicles in which we fly from place to place or persons to whom – in days of yore – we could complain when our flights were cancelled.

To avoid confusions of the sort just sketched, we will use here an extended version of the terminology proposed in [36].

3.3 Representational artifacts

On the side of a terminology, we are – or at least we should be – dealing primarily with entities that *are about* or *denote* entities or relations in first-order reality. Again in line with [36], we will use the term '*representational unit*', abbreviated as '*RU*', for any symbolic representation (code, character string, icon, …) which denotes a portion of reality. Thus in good terminologies we should find RUs for universals, for groups, for relations, and so forth.

While in a well-ordered terminology RUs can be classified on the basis of what they *denote*, it is for some terminologies hard to fathom whether their authors consider the RUs to denote entities in first-order reality, or entities ('*concepts'* as they would have it) inside the terminology itself [30, 48], or even whether they denote anything at all.

RUs can also be classified on the basis of their *form*, for instance as *codes* (e.g. '*GO:0048869*'), *terms* (e.g. '*cellular developmental process*'), or *expressions* (e.g. '*GO:0042995 : cell projection ---[i] GO:0019861 : flagellum*', which denotes the portion of reality consisting of the universal FLAGELLUM, the universal CELL PROJECTION, and the sub_kind relation that holds between them).

By convention, we will use the term '*term-RU*' for representational units in a terminology that have the form of a term. This allows us then to express, for example, that the term '*cellular developmental process*' is a term-RU in some terminology, or, in line with one of the objectives of terminology as a discipline [65], that the term '*developmental process*' would not be an adequate term-RU in a terminology because it does not express adequately that exclusively *cellular* developmental processes are denoted by it.

3.4 Parallels between terminologies and reality

The third item on the realist agenda in terminology development is the requirement that the structure of a terminology should mimic the structure of the ***POR*** that is covered by the terminology. Granular Partition Theory (GPT) provides a formal account of what it means for a structure to mimic (or not) another structure and it does this along various dimensions such as *correctness*, *transparency*, *fullness*, *completeness*, and so forth [66]. GPT allows for instance a terminology that represents whales as fish to be recognized as incorrect, where a terminology that classifies whales as animals but not as mammals, while not incorrect, still to be what GPT calls '*locally non-transparent*'. GPT does however not provide a means to quantify such differences, nor does it deal with issues such as whether it matters, for the purposes for which the terminology has been designed, whether whales are mammals, or what the reasons are for given sorts of mismatch. This is especially relevant in domains where our scientific understanding of reality is advancing rapidly and so that terminologies seeking to keep pace with these advances need to be updated frequently.

3.5 Bridging the gap between concept-based terminologies and realism-based ontologies

Concept-based terminology authors are motivated in their approach through criteria such as those put forward by Cimino [67], or by exploiting the power of description-logics and natural language understanding based algorithms (for recent reviews of the domain and some additional proposals, see for instance [68-69]). At first, the well-known criteria of non-vagueness (each term in a terminology should have at least one meaning) and non-ambiguity (each term should have no more than one meaning), seem to be very reasonable. When applied literally, however, they do not do justice to the fact that synonyms and homonyms are abundantly used in natural language [70]. Therefore, a common strategy is to replace in the criteria '*term'* by '*concept'*, where a '*concept'* stands for the meaning that all terms attached to it share. But, as argued by Smith [56], this does not eliminate the possibility that terms are included that rest on ontologically false beliefs, rather than denoting entities in first-order reality, which leads him to believe – and we with him – that RUs in terminologies should in every case denote universals or defined classes [36]. Interestingly, Cimino, in defense of his desiderata [50], agrees that '*the notion of terminologies that are limited to well-behaved universals, each one clearly understood because of its extension in reality, is appealing*', and suggests '*a path that acknowledges the importance of representing reality, as best we can know it, but accepts the need for concepts to help us, among other things, reason under uncertainty*'. He considers this a '*realistic path*' – rather than a '*realism-based*' one – and argues that in this path '*terminologies contain terms that refer to universals and to concepts, along with various names and unique identifiers for these. Sometimes, a single term will refer to an entity that has both universal and conceptual characteristics*'. But what then with the original criteria of non-vagueness and non-ambiguity? And is this then not mixing epistemology with ontology in a way that leads to problems of the sort outlined by Bodenreider *et.al.* when they concluded *'… that epistemology-loaded terms are pervasive in biomedical vocabularies, that the "classes" they name often do not comply with sound classification principles, and that they are therefore likely to cause problems in the evolution and alignment of terminologies and associated ontologies*' [71] ?

When a terminology has been selected as one that needs to be mapped to a realism-based ontology, each of its representational units should be inspected to identify, in terms of corresponding representations in such ontologies, what sorts of ***POR***s it is able to denote. A problem is that terms from concept-based terminologies often denote multiple distinct sorts of ***POR***s, for example because of asserted subtype relationships, as in SNOMED CT, whose concept 'Finger structure' subsumes the concepts 'entire finger' (a ***UNV*** under a realist framework) and 'all fingers' (a ***GRP***) (though SNOMED does not specify whether the latter means: 'all fingers in the world', 'all fingers of a given patient', 'all fingers on a given hand'). To address this problem, we introduce an intermediary layer made up of *classes* (***CLA***), understood as arbitrary totalities of elements which are either (i) defined through some descriptor referring to ***POR***s of any of the sorts described thus far (for example: 'the disorders in all the patients treated by Dr. McX'), or (ii) totalities whose elements are themselves so defined, or (iii) combinations of (i) and (ii). Classes under (i) thus carve out ***POR***s in ways which go far beyond ***GRP***s as defined in the foregoing. Classes under (ii) and (iii) allow simultaneous reference to entities associated together in ways which have no counterpart ***POR***, for example when we wish to assert heritability relations between Mary and certain of her ancestors who died many years before she was born.

Where groups have *members*, classes have *elements.* A *Defined Class* (***DCL***) is a class all of whose elements are specified by some class description. In the simplest case, this will be of the form '*ξ which stands in R to λ*', where '*ξ*' names some universal, for example 'person born in Belgium', which defines what we shall call a *Specifically Defined Class* (***SDC***), or 'patient who has tuberculosis', which defines a *Generically Defined Class* (***GDC***), each of whose elements enjoys the same relation (*exemplifies*) with instances of the single universal: tuberculosis. In more complex cases the definition will be of a logically more complex form, such as '*ξ* **has duration** *which stands in R to λ*', for example in the ***GDC*** *chronic pain*, where *ξ* is the universal: pain, *R* is the relation *longer_than* and *λ* is the temporal interval: 90 days. Many of the terminological definitions distinguished under P1 above will define terms which refer to ***GDC***s in the outlined sense.

For each ***GDC*** and for each ***SDC*** there is some universal from whose extension all its elements are drawn. An *Ad Hoc Class* (***AHC***), in contrast, is a ***CLA*** formed through combinations of ***GDC***s and ***SDC***s which is such that there is no such overarching universal. An example is, again, the SNOMED CT concept 'finger structure', since among the entities that can be denoted by this term are both ***GRP***s and ***APA***s

Among ***AHC***s, too, we can distinguish both *Generic* (***GAC***) and *Specific Ad Hoc Classes* (***SAC***). An example of a ***SAC*** is the class whose elements are the clinical signs exhibited by some specific patient with tuberculosis [63]. An equivalent ***GAC*** would be the class whose elements are the clinical signs exhibited by all tuberculosis patients assigned to the control group of a given clinical trial.

3.6 Future work

It has been stated that 'Terminologies should not be developed by reference to a system of preferred terms, rather they should be developed in such a way that their individual nodes and [the] relations amongst these nodes are modeled on an underlying formal ontology, where the linguistic content of these nodes will be filled in based on a system of terms and synonyms (from many different languages) that is associated with each node based on the intended ontological interpretation of that node'. [70] Few, if any, existing biomedical terminologies exhibit these characteristics. The framework we propose is designed to promote progress in this respect, with the goal, not of developing an underlying formal ontology for these terminologies themselves, but rather of achieving appropriate mappings to OBO Foundry ontologies. The approach provides a tool for terminologists to detect ambiguities and conflations in the conceptual structures they have designed and to determine the correct handling of terms proposed as synonyms; it also forces developers of realism-based ontologies to be more precise about what exactly the representational units in their artifacts denote. Certainly there is a long way to go. We acknowledge that the proposed approach is not easy to apply because of the subtle distinctions it requires, distinctions which are perhaps not easy to understand especially for adepts of the concept-based approach. We believe, however, that the approach promises significant benefits, both practical and theoretical, in the long run.

4 Realism-based versioning of ontologies and terminologies

4.1 Rationale

High quality ontologies are – or at least should be – the analogue of scientific textbooks in that they contain what is believed to be the case in a scientific domain, excluding what is not known or judged irrelevant. Appropriately developed ontologies have the advantage over textbooks that their content can be understood by software agents. Like textbooks change when the state of the art (SoA) in the domain covered by them changes, so should ontologies change accordingly. Changes in ontologies are required either (1) because of changes in reality (new diseases arise – AIDS, SARS, … - bacteria become resistant, new drugs are manufactured) or (2) because scientists come to discover what is already the case for some time but was unknown or judged irrelevant thus far (biomarkers, disease pathways) or (3) realize that earlier assumptions were wrong. The degree to which an ontology corresponds to the SoA as well as to the degree to which changes in successive versions correspond to changes in the SoA are therefore important markers to measure quality objectively. [56]

Most biomedical ontologies developed thus far are released either without any versioning information at all, or with information limited to what has changed in comparison with the previous version. It is thus left unspecified in new versions why alterations have been introduced, i.e. whether there are corresponding changes in reality or in the ontology authors' understanding or representation of reality. This hampers the re-interpretation of data annotated by means of earlier versions. Furthermore, some ontologies provide documentation to the effect that a specific class was added in a certain version at a certain time, but fail to tell us since when in history there are believed to have been instances of that class. This poses problems to annotate patient data that have been collected prior to the new release because for some classes it might be such that the corresponding entities in reality did not yet exist (long time) before the inclusion of the class, while for others, that might be the case.

Indeed, efforts in the domain of ontology versioning and evolution have focused thus far on techniques for keeping track of which entries in an ontology appeared, disappeared, became fused or split in successive versions. Because the question is not raised as to *why* such changes are made, crucial distinctions are missed between the different kinds of changes in an ontology, reflecting for example:

1. changes in the underlying reality (does the appearance or disappearance of an entry in a new version of an ontology relate to the appearance or disappearance of entities or of relationships among entities in reality?);
2. changes in our scientific understanding;
3. reassessments of what is considered to be relevant for inclusion in an ontology, or:
4. encoding mistakes introduced during ontology curation (for example through erroneous introduction of duplicate entries reflecting lack of attention to differences in spelling).

That such differences are overlooked is no surprise in the case of concept-based ontologies, where, because entities in reality are thought of as playing at best a secondary role, the associated reasoning machinery takes care only of *internal* consistency. An example is the CONCORDIA model for managing divergence in concept-based terminologies. [44] It consists of a well-elaborated change model that is able to capture 27 different sorts of changes such as adding or merging concepts, or adding and deleting terms, but provides no facilities to log motivations for these changes in the way we intend to propose here.

Typical for the concept-based approach is its 'inward'-orientation: the rules or criteria designed to help authors make better terminologies have no other basis than the rules themselves; there is no external benchmark. As a consequence, it is very hard to use these rules in any other way than for the purpose of counting. This is witnessed by the vocabulary criteria defended in [72] and applied to the Gene Ontology as reported in [73]: of the 99 criteria deemed important, GO was found to meet 78 criteria totally, 5 partially, and 2 not at all. Furthermore, 13 criteria were found not to be applicable and 1 was not assessed. But how, we ask, do these findings correlate with quality?

As another example, Hartung and colleagues '*consider the evolution in the relative share of leaf (vs. inner) nodes, the number of relationships, the distribution of is-a, part-of and other relationships, as well as in the concept node degrees and number of paths*' [74], but they also give no further indications as to how these metrics as applied by them to the Gene Ontology and other life science terminologies, relate to quality. They recognize in their conclusion, however, opportunities for future work, more specifically that their '*analysis framework can be extended by additional types of change*' and that '*algorithms to generate annotation and ontology mappings can be extended or refined to improve their stability w.r.t. ontology evolution, e.g., by taking obsolete concepts and versioning explicitly into account*'. This is indeed the strategy that we proposed in [11] and have implemented here.

An alternative to concept-orientation is offered by approaches based on ontological realism [57]. Although realism-based auditing has proven successful to detect flaws in concept-based terminologies [19, 22, 49, 58, 60, 75-81], it has prior to our project not been applied systematically to each representational unit in a terminology.

4.2 Basic principles of realism-based versioning

Different ontology authors maintain different positions concerning the correspondence between their representations and reality. Authors of realism-based ontologies maintain that their ontologies are intended to mirror reality; authors of concept-based ontologies maintain that their ontologies are intended to mirror cognitive representations on the part of domain experts. Our metric is based on the realist view, which means that it seeks to use objective reality as benchmark of correctness. This means in turn that it assumes that it is possible for humans to gain access to this reality, for example through the methods of evidence-based medicine. Since human cognition is fallible, both our cognitive representations and the representational artifacts based thereon may contain mistakes. But such mistakes can also be corrected, and it is above all this fact which makes possible a metric along the lines proposed.

In line with the theory of granular partitions, [66] we see complex representations as being composed in modular fashion of sub-representations built out of *representational units* that are assumed to correspond to **POR**s.

Some characteristics of the units in a representation created for clinical or research purposes are:

1. each such unit is assumed by the creators of the representation to be veridical, i.e. to conform to some relevant **POR** as conceived on the best current scientific understanding (which may, of course, rest on errors);
2. several units may correspond to the same **POR** by presenting different though still veridical views or perspectives, for instance at different levels of granularity (one thing may be described both as being brown and as reflecting light of a certain wavelength, or one event as an event of buying and of selling);
3. what is to be represented by the units in a representation depends on the purposes which the representation is designed to serve.

We shall concentrate in what follows on representational artifacts such as ontologies and terminologies, in which the representational units are terms from some natural or formal language, which are assumed to refer to universals or defined classes.

4.3 The relevance and veridicality of expressions

Because ontologies, as thus conceived, (1) are artifacts created for some purpose (e.g. to serve as controlled vocabulary, or to provide domain knowledge to a software application), and because (2) they are intended to mirror reality, and because (3) reasoning with ontologies requires efficiency from a computational point of view, we argue that an optimal ontology should constitute a representation of *all and only those portions of reality that are relevant for its purpose*. Clearly, things may go wrong on the way to achieving this optimal representation. First, ontology developers may be in error as to what is the case in their target domain, leading to *assertion errors*. Second, they may be in error as to what is objectively relevant to a given purpose, leading to *relevance errors*. Third, they may not successfully encode their underlying cognitive representations, so that particular representational units fail to point to the intended **POR**s because of *encoding errors*.

An ideal ontology, now, would be marked by none of these three types of errors. Each term in such an ontology would designate (1) a single **POR**, which is (2) relevant to the purposes of the ontology and such that (3) the authors of the ontology intended to use this term to designate this **POR** Moreover, (4) there would be no **POR**s objectively relevant to these purposes that are not referred to in the ontology.

4.4 Match/mismatch configurations

As shown in **Table 1**, there are 17 possible configurations of match or mismatch – 2 more than in our original proposal [11] – which are divided into two groups, labelled '*P*' and '*A*', denoting respectively the presence or absence of a RU. Each group can further be subdivided into two smaller groups on the basis of whether the presence or absence of a RU in a terminology is justified ('*P*+' and '*A*+') or unjustified ('*P*-' and '*A*-').

The configurations reflect the different kinds of mismatch between what the terminology authors *believe* to exist or to be relevant, on the one hand, and matters of *objective* existence and *objective* relevance-to-purpose on the other. The encoding of a belief can be either correct (R+) or incorrect, either (a) because the encoding does not refer (¬R) or (b) because it does refer, but to a **POR** other than the one which was intended (R-). The two configurations not considered in our original proposal [11] both involve a RU that denotes an intended and objectively existing **POR** that, however, is already denoted by another RU in the terminology (R++).

As an example, configuration P-1 would hold for a RU stating that 'whales are fish': the putative **POR** does not exist – hence the 'N' in column (2) of **Table 1** – and therefore objective relevance does not apply, as indicated by the '-' in column (3). The authors of the terminology do however *believe* that whales are fish and consider it to be relevant; therefore this configuration is marked by the presence of 'Y' in both columns (4) and (5). Finally, they use the representational machinery offered by the terminology correctly such that the RU is the intended representation – note the 'Y' in column (6) – but this in absence of a corresponding **POR**, as indicated by '¬R' in column (7).

Of the 17 configurations, only 3 are desirable: P+1, which consists in the justified presence of a RU that correctly refers to a relevant **POR**; and A+1 and A+2, which consist in the justified exclusion of a RU, either because there is no **POR** to be referred to, or because this **POR** is not relevant to the terminology's purpose. A-3 and A-4 are borderline cases, in which errors made by terminology authors are without deleterious effect, either because something that is erroneously assumed to exist is deemed irrelevant, or because something that is truly irrelevant is overlooked. There are 10 different kinds of 'P' configurations of which, interestingly, only P+1 and P-6 refer correctly to a corresponding **POR**: the former reflects our ideal case for presences; the latter is marred by the incorrect inclusion of a RU which lacks relevance. P-9 and P-10 also denote an existing and intended **POR**, but the mistake here is that the terminology authors are not aware of their departure from the principle that for each entity in first-order reality there should be maximally one RU of a specific form.

Table 1: Typology of expressions included in and excluded from an ontology in light of relevance and relation to external reality

Configuration	Reality		Representation				Magnitude of error
			Authors' Belief		Encoding		
	Objective Existence	Objective Relevance	In existence	In relevance	Intended encoding	Type of reference	
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
P+1	Y	Y	Y	Y	Y	R+	0
A+1	N	-	N	-	-	-	0
A+2	Y	N	Y	N	-	-	0
P-1	N	-	Y	Y	Y	¬R	3
P-2	N	-	Y	Y	N	¬R	4
P-3	N	-	Y	Y	N	R-	5
P-4	Y	Y	Y	Y	N	¬R	1
P-5	Y	Y	Y	Y	N	R-	2
P-6	Y	N	Y	Y	Y	R+	1
P-7	Y	N	Y	Y	N	¬R	2
P-8	Y	N	Y	Y	N	R-	3
P-9	Y	Y	Y	Y	Y	R++	1
P-10	Y	N	Y	Y	Y	R++	2
A-1	Y	Y	Y	N	-	-	1
A-2	Y	Y	N	-	-	-	1
A-3	N	-	Y	N	-	-	1
A-4	Y	N	N	-	-	-	1

The last column of Table 1 shows the magnitude of the error committed when a RU reflecting a given type of configuration is included in or left out of a terminology as measured against its corresponding ideal configuration. Because these ideal configurations are P+1, A+1, and A+2, and because for any other configuration the '*corresponding'* ideal configuration is the one which has the same values in columns (2) and (3), the number of mistakes committed in P-4, P-5, P-9, A-1 and A-2 need to be measured against P+1. Similarly A+1 is the ideal configuration for P-1, P-2, P-3 and A-3, and A+2 for all the others. The magnitude of an error is calculated by counting the number of differences that a specific configuration exhibits with respect to its ideal configuration in each of the columns (4) to (7) of Table 1, with the additional rule that a non-intended encoding which denotes an existing and thus non-intended **POR** – the presence of 'R-' in column (7) – counts double. This is because we judge that users of a terminology will be less likely to use RUs which denote nothing than RUs that denote non-intended **POR**s: probably far more users will notice that an RU of the type 'whales are leprechauns' is a mistake – and thus never use that RU in some annotation – than there would be users that would notice the mistake in an RU of the type 'whales are fish'.

4.5 Quantification of structural mismatches regarding whole terminologies

Theoretically, it would now be an easy exercise to assess the quality of a terminology as a whole: we would have to (1) inspect each RU in the terminology to determine what match/mismatch configuration it exhibits, and (2) examine its coverage domain to see what relevant RUs are missing. Because the magnitude of a mistake in an undesirable configuration is maximally 5, we would give each best case configuration encountered a score of 5, while each deviation there from would receive the difference between 5 and the corresponding penalty for the corresponding sort of deviant case. The total score would be the ratio of the sum of the scores obtained for each present RU, over the sum of five times the number of RUs present and 4 times the number of RUs missing. The latter is because all missing RUs have an error magnitude of 1, and 5-1=4. The general formula is:

$$\frac{\sum_{i=1}^{n} RU_i^p (5 - e_i)}{5n + 4m} \qquad (1)$$

in which RU^p stands for any RU present in the terminology, e_i for the magnitude of the error (if any) for the corresponding RU^p, n for the number of *RUs* present in the terminology and m for the number of RUs unjustifiably absent.

The score itself can be viewed as a variation to the well-known recall and precision metric, but combined in but one metric and adjusted for the magnitude of the errors committed.

Table 2: Scoring the quality of terminologies using reality as benchmark

	Reality	**Terminology 1**		**Terminology 2**		**Terminology 3**	
	Config.	**Config.**	**Error**	**Config.**	**Error**	**Config.**	**Error**
(1)	**(2)**	**(3)**	**(4)**	**(5)**	**(6)**	**(7)**	**(8)**
Animal	P+1	P+1	0	P+1	0	P+1	0
Fish	P+1	P+1	0	P+1	0	P+1	0
Whale	P+1	P+1	0	P+1	0	P+1	0
mammal	P+1	P+1	0	P+1	0	P+1	0
fish are animals	P+1	P+1	0	P+1	0	P+1	0
mammals are animals	P+1	P+1	0	P+1	0	P+1	0
whales are fish	A+1	P-1	3	A+1	0	A+1	0
whales are animals	P+1	P+1	0	P+1	0	P+1	0
whales are mammals	P+1	A-2	1	A-2	1	P+1	0
SCORE	8*5/ ((8*5)+(0*4)) = 1.00	((7*5)+(1*2))/ ((8*5)+(1*4)) =0.84		7*5/ ((7*5)+(1*4)) =0.90		8*5/ ((8*5)+(0*4)) =1.00	

Table 2 gives an example of how this metric should be applied. Imagine three terminologies that provide a vocabulary for describing whales. All three terminologies have RUs for WHALE, FISH, ANIMAL and MAMMAL, but they differ in whether whales are asserted to be (1) fish (Terminology 1 - T1), (2) animals without further precisification (Terminology 2 - T2), or (3) mammals (Terminology 3 - T3). In reality, of course, whales are mammals. We further assume, for the sake of the example, that the terminology authors did not make encoding mistakes: if there is a mistake in the terminology, then it is because their scientific understanding of reality is

erroneous, not because they encoded a known fact erroneously. We also assume that all **POR**s in the domain are relevant to the purposes for which the terminologies are built. When we then compare the three terminologies against the benchmark of reality, the latter being expressed in column (2) of **Table 2**, we see that T1 has one erroneous RU, which is an example of a mistake of type P-1, and one unjustified absence of type A-2; T2 exhibits the same unjustified absence, but in contrast to T1 it does not include an erroneous RU; T3, finally, mimics the structure of reality completely. For each RU in each terminology, the corresponding error magnitudes, if any, are shown in columns (4), (6) and (8). Applying the formula described above, this gives a quality score for T1 of 0.84, for T2 of 0.90 and for T3 of 1.00.

Note that we took the justified absence of type A+1 (whales are fish) into account *only* because there is a RU (in T1) that posits the opposite. It is of course *not* a presupposition of our proposal that one should include all putative RUs which do not denote a corresponding **POR** – e.g. that animals are fish, that animals are whales, that fish are mammals, that unicorns are leprechauns, and so forth – in any such assessment. Importantly, not doing so does not affect the magnitude of the overall score. This can be seen in relation to T2 and T3 whose quality scores are not influenced by the fact that they do not contain an erroneous RU to the effect that whales are fish. This is one of the desirable mathematical properties that our metric exhibits, of which the complete characterization, however, falls beyond the scope of this paper.

Note also that this procedure reflects what might initially appear to be an unacceptable idealization, because determining the type of configuration an (included or excluded) RU is involved in depends upon two factors – objective relevance-to-purpose, and relation to objective reality – whose assessment is something which could be correctly carried out only by someone able to adopt the perspective of a god-like observer. Less idealistically, this god-like observer might be replaced by another terminology that is used as gold standard [82], and we adopt here a generalization of this latter approach by using successive versions of a terminology as the gold standard relative to its predecessors.

4.6 Quality assessment of terminologies over successive versions

The minimal requirement for releasing a terminology as expressed in terms of the realist paradigm (though independent of whether or not authors of a given terminology endorse a realist view) is that its authors should assume in good faith that all its constituent expressions are of the P+1 type (requirement R1). A stronger requirement would be that the authors advance the terminology as complete, i.e. as containing RUs designating *all* **POR**s deemed relevant to its purpose (requirement R2). Successive versions of a terminology should approximate ever more closely to this latter ideal. To exploit the paradigm completely, one could even argue that it should be part of the standard terminology authoring process to document any changes made in successive versions by means of the typology described in **Table 1** [11]. This requires terminology authors to register whether or not the changes they introduced in a new version of the terminology are dictated by changes in (1) the underlying reality (requirement R3), (2) objective relevance of an included expression to the purposes of the ontology (requirement R4), (3) the ontology authors' understanding of each of these (requirement R5), and also by (4) the correction of encoding errors (requirement R6).

To see how the heuristic of using a new version of a terminology functions as surrogate for a god-like observer in relation to its predecessors, consider again the whale/fish example of **Table 2**. This time, however, we will consider T1, T2 and T3 to be versions of the same terminology, T3 being newer than T2, and T2 being newer than T1. The results of this interpretation are summarized in **Table 3**; with **Table 4** showing how the individual quality scores are calculated.

Table 3: Scoring the quality of terminologies using new versions

	Time t1		Time t2				Time t3					
	T1		T1		T2		T1		T2		T3	
	C.	E.	C.	E.	C.	E.	C.	E.	C.	E.	C.	E.
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)
Animal	P+1	0	P+1	0	P+1	0	P+1	0	P+1	0	P+1	0
Fish	P+1	0	P+1	0	P+1	0	P+1	0	P+1	0	P+1	0
Whale	P+1	0	P+1	0	P+1	0	P+1	0	P+1	0	P+1	0
Mammal	P+1	0	P+1	0	P+1	0	P+1	0	P+1	0	P+1	0
fish are animals	P+1	0	P+1	0	P+1	0	P+1	0	P+1	0	P+1	0
mammals are animals	P+1	0	P+1	0	P+1	0	P+1	0	P+1	0	P+1	0
whales are fish	P+1	0	P-1	3	A+1	0	P-1	3	A+1	0	A+1	0
whales are animals	P+1	0	P+1	0	P+1	0	P+1	0	P+1	0	P+1	0
whales are mammals	-	-	-	-	-	-	A-2	1	A-2	1	P+1	0
SCORE	1.00		0.93		1.00		0.84		0.90		1.00	

Table 4: Calculation of quality scores for terminology versions at different times

Terminology	Time of assessment	Formula for quality score	Quality Score
T1	**t1**	(8*5)/(8*5)	1.00
	t2	((7*5)+(1*2))/(8*5)	0.93
	t3	((7*5)+(1*2))/((8*5)+(1*4))	0.84
T2	**t2**	(7*5)/(7*5)	1.00
	t3	(7*5)/((7*5)+(1*4))	0.90
T3	**t3**	(8*5)/((8*5)+(0*4))	1.00

Table 5: Views on the quality of a terminology through successive versions

	Time		
Terminology version	**t1**	**t2**	**t3**
T1	1.00	0.93	0.84
T2	-	1.00	0.90
T3	-	-	1.00

When the first version of the terminology (T1) is released, the authors assume in good faith that their work is correct, i.e. that all RUs denote the desired **POR**s, and that all and only relevant RUs are present. They might believe that some RUs are missing, but of course, they have no clue which ones, otherwise they would have been included. Therefore, version T1 at time t_1 was assumed to be 'state of the art' and therefore of quality 1.00, the maximal attainable score. At time t_2, however, the authors discover that whales are not fish and they make the corresponding RU 'obsolete'. Note that obsoleting a RU by giving the reason for the change, is preferable to just removing it: if, indeed, the only change introduced between T2 and T1 would be the deletion of the RU that whales are fish, external auditors might wonder whether (1) the deletion is an omission brought about by an encoding error, in which case the RU which was believed to be of type P+1 at t_1 has to be believed to be of type P-2 at t_2, or (2) a deletion based on a conscious decision either (2a) that whales are still to be considered to be fish, but that the RU is not relevant for the purposes for which the terminology is being built, hence consisting in an A-3 type of mistake, or (2b) that the right sort of discovery was made and thus the original RU was of type P-1. Because the latter is the case, the quality score of T1 at t_1 can be recalculated according to the state of the art reached at t_2 using Eq. (1).

A similar analysis can be carried out at t_3, but now applied to both T1 and T2; in general, each new version of a terminology allows us to assess the quality of all previous versions of the terminology in light of the state of the art reached when the new version is released (see **Table 5**).

5 An exploratory analysis of SNOMED CT's history mechanism

SNOMED CT is a clinical reference terminology for annotating patient data designed to enable electronic clinical decision support, disease screening and enhanced patient safety. [83] It was first issued in 2002 following the merger of SNOMED-RT and Clinical Terms Version 3.

5.1 Concept-orientation in SNOMED CT

SNOMED CT is structured around a taxonomy of what is called ‘concepts’. [84] Concepts are further associated with a variable number of elements such as their relationships to other concepts and the terms – linked to the concepts by means of descriptions. Whereas the descriptions provide the vocabulary to talk about the concepts (or what might be instances thereof when the vocabulary is used to annotate patient data), the concepts and relationships are supposed to be a representation of what exists, and is relevant for certain purposes in biomedicine [11].

Until the January 2010 version, SNOMED CT’s authors defined a concept as ‘*a clinical idea to which a unique ConceptId has been assigned*’ thereby further specifying that ‘*each Concept is represented by a row in the Concepts Table*’ [12].

In 2010, in line with earlier critiques about the ambiguities concept-based systems in general suffer from [13], the glossary of the Technical Reference Guide marks the word ‘Concept’ as ‘*an ambiguous term. Depending on the context, it may refer to: a clinical idea to which a unique ConceptId has been assigned; the ConceptId itself, which is the key of the Concepts Table (in this case it is less ambiguous to use the term “concept code”); the real-world referent(s) of the ConceptId, that is, the class of entities in reality which the ConceptId represents (in this case it is less ambiguous to use the term “meaning” or “code meaning”)*’ [14].

However, merely pointing this out, however true it might be, does not yet solve the problem. For one could still read in the same document, for example, that a SNOMED CT term is ‘*a text string that represents the Concept*’. So what is it then that is represented by a term: (1) the clinical idea, (2) less likely, but nevertheless in line with the expressed ambiguity – the ConceptId, or (3) the real-world referent(s)? The same question must then be asked for the several hundred occurrences of the word ‘concept’ throughout the SNOMED CT documentation. In some cases, readers can infer from the context which meaning is intended, but in most cases, only the SNOMED CT authors can provide the answer by rewriting the entire documentation.

Unfortunately, as inspection reveals, it is very hard for readers and even for SNOMED CT authors, to disambiguate on the basis of the minimal context provided in sentences in which the word ‘concept’ appears between concept as clinical idea and concept as meaning, i.e. as real-world referent. This is not only because clinical ideas are real-world entities themselves – although of a different nature than, for example, persons, viruses and surgical procedures, and some being such that they are about other real-world entities while others are about nothing at all [8] – but also because SNOMED CT authors have not yet made it clear what sorts of real-world entities their concepts represent: denoting real-world entities unambiguously requires ontological commitment and it has been shown that SNOMED CT is incoherent in this respect [7].

Relying on ‘meaning’ unfortunately doesn’t help much. According to SNOMED CT’s glossary definition for ‘concept’ discussed above, the meaning of a concept(Id) would correspond to what Frege referred to as the ‘Bedeutung’ (‘reference’, ‘extension’) of a term [15]. However, in the User Guide, it is specified that ‘*a “concept” is a clinical meaning identified by a unique numeric identifier (ConceptId) that never changes. The concepts are formally defined in terms of their relationships with other concepts. These logical definitions give explicit meaning which a computer can process and query on*’ [16]. Here, the word ‘meaning’ corresponds rather to

Frege's 'Sinn' ('sense', 'intension') [15]. And finally, in the SNOMED-CT Editorial Guide, a document that became part of the official documentation only since the latest release (although parts of it existed earlier in the form of drafts for comments), SNOMED CT is described as a 'terminological resource' which '*consists of codes representing meanings expressed as terms, with interrelationships between the codes to provide enhanced representation of the meanings*' [17]. As a result, the reader is not only left with the question what sort of meaning is discussed each time the word 'meaning' is used – the Editorial Guide is indeed more about 'meanings' than 'concepts' – but also what actually is represented in SNOMED-CT: (1) clinical ideas – in people's minds or concretized in writings, software programs and presentations, respectively called L2 and L3-entities in [8], (2) a broader group of real-world referents that includes not only tangible entities such as patients and knives but also the processes in which the latter participate and the forces they undergo, or (3) 'meanings'.

5.2 SNOMED CT's history mechanism

The content of SNOMED CT evolves with each release. Types of changes involving the core components include the addition or deletion of concepts, descriptions, and relationships. These changes are said to be '*driven by changes in understanding of health and disease processes; introduction of new drugs, investigations, therapies and procedures; new threats to health; as well as proposals and work provided by SNOMED partners and licensees*'.[85]

A history mechanism keeps track of the changes over time thereby adhering to the well known requirements for terminology management proposed by Cimino [67] on the basis of the following requirements: (1) graceful evolution rather than radical change, (2) the concept represented by the class does not change, (3) classes may become inactive but are never deleted, (4) concept identifiers are persistent over time and are never reused, (5) the link between a term and a class is persistent, so that if a term is no longer appropriate to a given class, then it is inactivated, and (6) recognition of redundancies.

As shown in **Table 6**, this history mechanism captures *what* changes have been introduced over time, and partly *why* such changes were made. The table indicates that the number of changes is very large. They result in a pool of 'useful' (i.e. active and non-limited) classes comprising 75% of the whole terminology.

Table 6: Distribution of SNOMED CT concepts' status in release January 2007

		January 2007	
ST	**Concept Status**	**N**	**%**
0	active in current use	281,693	75.37%
6	active with limited clinical value (classification concept or an administrative definition)	27,200	7.28%
1	inactive: 'retired' without a specified reason	6,832	1.83%
10	inactive because moved elsewhere	1,091	0.29%
2	inactive: withdrawn because duplication	40,018	10.71%
3	inactive because no longer recognized as a valid clinical concept (outdated)	1,199	0.32%
4	inactive because inherently ambiguous.	14,694	3.93%
5	inactive because found to contain a mistake	1,004	0.27%
	TOTAL	373,731	100%

Another quantitative view on the number of concept changes in SNOMED CT is provided in **Table 7** and **Table 8**.

Table 7: number of concept changes in SNOMED CT from Release Jan 2002 to July 2009

ST	Existing concept made …	N
0	active: in current use	2,010
1	inactive: 'retired' without a specified reason	1,993
2	inactive: withdrawn because duplication	9,711
3	inactive because no longer recognized as a valid clinical concept (outdated)	1,348
4	inactive because inherently ambiguous.	5,829
5	inactive because found to contain a mistake	1,204
6	active with limited clinical value (classification concept or an administrative definition)	4,461
10	inactive because moved elsewhere	14,406
11	pending move	
	TOTAL	40,962

Table 8: Changes related to SNOMED concepts from version 2002-07-31 to version 2007-01-03 as listed in the Component History Table.

CT	ST	020731	030131	030731	040131	040731	050131	050731	060131	060731	070131	Total
0	0	7456	7765	8067	4266	4578	2588	1699	2411	2112	3029	43971
0	1	0	238	0	0	0	0	0	0	0	0	238
0	2	9	3382	28	37	103	6	3	0	0	0	3568
0	3	0	0	2	0	0	0	0	0	0	0	2
0	4	0	12	1	59	0	1	0	0	0	0	73
0	5	0	4	1	0	0	0	0	0	0	0	5
0	6	4	23	13	112	8	42	16	0	0	0	218
1	0	27	282	68	226	821	222	15	48	18	29	1756
1	1	1140	39	19	24	22	14	1	8	7	54	1328
1	10	16	885	0	0	6	50	28	87	20	0	1092
1	2	1327	1684	821	989	1262	462	233	392	322	757	8249
1	3	4	8	319	393	14	58	13	29	298	65	1201
1	4	1116	730	696	533	222	320	170	218	369	477	4851
1	5	21	290	373	66	30	79	46	32	58	56	1051
1	6	3	53	368	3866	31	5	14	4	2	24	4370
2	0	11766	7919	2175	1942	3069	903	656	8706	5785	1126	44047
2	10	0	2	0	0	0	0	0	0	0	0	2
2	2	4	4	1	0	0	12	6	4	0	0	31
2	6	1090	135	36	202	205	29	10	5960	1691	15	9373
Activated		20346	16177	10727	10614	8712	3789	2410	17129	9608	4223	103735
Inactivated		3637	7278	2261	2101	1659	1002	500	770	1074	1409	21691
Added		7469	11424	8112	4474	4689	2637	1718	2411	2112	3029	48075
Changed		16514	12031	4876	8241	5682	2154	1192	15488	8570	2603	77351
Total		23983	23455	12988	12715	10371	4791	2910	17899	10682	5632	125426

Legend: CT= Change Type (0=added, 1=status change, 2=minor change); ST= status type

Table 8 shows that over the 10 revisions 38% of the changes concerning classes were additions (48,075, CT="0") as compared to 62% (77,351, CT="1" or "2") which were class modifications, of which over 17% were cases of class retirement (21,691). The percentage of additions versus modifications varies widely from one release to another.

One would legitimately be surprised to find that in release 2003-01-31, 238 classes were *added* (not *changed*) with the status 'retired', i.e. inactivated without any specified reason, 3382 *added* as ‘duplicate’, 12 as ‘ambiguous’, and even 4 as ‘erroneous’ (**Table 8**). We found in total 5402 of such classes over the entire SNOMED CT history. An example is the class with FSN ‘*Green peppercorn RAST test (procedure)*’ added as ‘duplicate’ in 2004-07-31 and declared ‘same as’ ‘*Piper nigrum (unripe seed) specific IgE antibody measurement (procedure)*’ which was added in 2003-01-31 and has among its associated terms ‘*Green peppercorn RAST test*’. The reason for these strange additions turns out to be that, during the first few releases of SNOMED CT, the UK was continuing to make updates to the 4-byte and 5-byte versions of the Read codes. New additions to these necessarily involved 4-byte Read code identifiers, and different 5-byte Read code identifiers. These were added to the SNOMED CT tables to maintain 100% inclusion of all the Read codes ever issued. As a result, some new rows in the Concepts table were (intentionally) duplicates, or they were ambiguous, from the start.

5.3 Various ways of counting changes

As indicated by **Table 9**, the same concept or description can undergo several modifications over time. Some modifications introduced at a certain time, may become invalidated at later time.

Table 9: Distribution of the number of concepts and descriptions according to the number of modifications they underwent over time (status January 2007).

Modifications	Concept		Descriptions	
N	N	%	N	%
1	230,738	61.74%	869,736	83.40%
2	120,913	32.35%	158,488	15.20%
3	19,972	5.34%	12,871	1.23%
4	2,030	0.54%	1,728	0.17%
5	74	0.02%	43	0.00%
6	3	0.00%	3	0.00%
7	0	0.00%	2	0.00%
8	1	0.00%	0	0.00%
Total	373,731	100%	1,042,871	100%

When also the number of relationships in which a concept functions as source concept is taken into account, the number of changes per concepts is overwhelming, the top 100 for the versions Jan 2002 to July 2009 being displayed in **Table 10**. The sorts of changes accounted for include:

- changes at the level of the concept (C) concerning its status (active, ambiguous, …), fully specified name, and being primitive;
- changes at the level of the concept's descriptions (D) including status of the description, capitalization, type (FSN, synonym, …), and language,
- changes at the level of the concept's relationships (R) including adding and deleting, refinability, definitional aspect and participation in a role group.

Table 10: Top 100 of concepts having undergone changes between the January 2002 and July 2009 versions of SNOMED CT

Oldest-FSN - CONCEPTID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	C	R	D	ToT	ToC
Retrograde catheter ureteropyelography (procedure) - 176086008	19	21	19	11	8	9	33	32	36	32	27	4	0	2	8	0	1	258	2	261	242
Pereyra procedure including anterior colporrhaphy (procedure) - 40587007	20	14	22	23	22	22	30	21	20	20	26	2	0	4	10	0	1	252	3	256	236
Epidural injection of neurolytic solution, caudal (procedure) - 58418003	17	22	11	12	16	18	26	10	10	10	37	10	12	34	7	0	2	247	3	252	235
Infusion of intra-arterial thrombolytic agent with percutaneous transluminal coronary angioplasty, multiple vessels (procedure) - 80762004	26	10	14	5	2	19	40	26	29	28	29	0	7	5	20	0	1	257	2	260	234
Infusion of intra-arterial thrombolytic agent with percutaneous transluminal coronary angioplasty, single vessel (procedure) - 91338001	26	10	14	5	2	19	40	26	29	28	29	0	7	5	20	0	1	257	2	260	234
Reduction of closed carpometacarpal fracture dislocation of thumb with manipulation and skeletal fixation (procedure) - 46989008	17	13	11	10	10	15	25	20	20	20	28	3	3	19	33	1	4	238	6	248	231
Percutaneous transluminal injection of therapeutic substance into coronary artery NEC (procedure) - 175072001	17	12	6	0	0	7	38	36	40	36	28	0	7	5	8	0	1	237	2	240	223
Open reduction of fracture of femur with internal fixation (procedure) - 57470004	20	17	8	5	4	8	37	34	30	30	33	16	0	0	0	0	2	238	2	242	222
Revision to open reduction of fracture dislocation and external fixation (procedure) - 179692000	30	23	7	2	0	4	13	14	12	12	31	34	22	22	20	0	2	242	2	246	216
Ocular-mucous membrane syndrome (disorder) - 95331002	27	33	15	20	14	15	26	20	21	21	30	0	0	1	0	0	1	237	5	243	216
Extracorporeal shockwave lithotripsy of the gallbladder and bile duct (procedure) - 14372000	14	13	2	5	0	23	31	28	30	28	29	10	16	0	0	0	1	226	2	229	215
Open reduction of fracture of carpals and metacarpals with internal fixation (procedure) - 68432004	20	19	12	11	8	12	29	18	18	18	39	30	0	0	0	0	1	231	2	234	214
Revision to open reduction spinal fracture and external fixation (procedure) - 178726008	23	12	11	3	0	3	22	22	22	22	38	32	25	0	0	0	2	231	2	235	212
Open reduction of fracture of tibia and fibula with internal fixation (procedure) - 74011006	19	15	6	5	4	8	26	22	22	22	40	39	0	0	0	0	1	225	2	228	209
Renewal of prosthetic collar around male bladder neck (procedure) - 303601004	14	9	13	4	4	14	29	24	24	25	30	4	5	0	0	23	1	219	2	222	208
Revision to open reduction of fracture and internal fixation with cerclage wiring (procedure) - 179082009	20	22	7	7	4	7	32	26	26	26	34	14	0	0	0	0	2	221	2	225	205
Primary arthroscopic reduction and fixation of fracture (procedure) - 179188006	13	13	11	7	6	8	30	28	28	22	25	25	0	0	0	0	1	213	2	216	203
Push-bang operation for ureteric calculus (procedure) - 236192006	25	14	26	22	4	16	36	20	20	20	21	0	0	0	0	0	1	221	2	224	199
Primary open reduction of fracture dislocation and external fixation (procedure) - 179617005	28	27	5	2	0	3	9	12	12	12	26	32	9	19	26	0	1	219	2	222	194
Hydromyelocele with hydrocephalus (disorder) - 253115005	9	22	8	4	4	11	29	21	20	20	35	3	0	16	0	0	2	198	2	202	193
Open reduction of fracture of phalanges of hand with internal fixation (procedure) - 43050006	18	15	8	3	2	4	23	24	24	24	34	28	0	2	0	0	2	205	2	209	191
Cystourethroscopy with ureteroscopy and pyeloscopy (procedure) - 112900002	16	16	22	11	10	16	25	12	22	24	32	0	0	0	0	0	2	199	5	206	190
Revision to open reduction and internal fixation of proximal femoral fracture with screw/nail device alone (procedure) - 179065007	22	19	8	3	2	5	20	16	23	22	28	26	18	0	0	0	2	206	4	212	190
Revision to open reduction and internal fixation of proximal femoral fracture with screw/nail and intramedullary device (procedure) - 179066008	22	19	8	3	2	5	20	16	23	22	28	26	18	0	0	0	1	209	2	212	190
Tolosa-Hunt syndrome (disorder) - 95794005	26	32	15	19	8	13	18	18	17	16	29	0	0	0	1	3	1	211	3	215	189
Ligation of fallopian tubes with division by endoscopy (procedure) - 88259002	19	17	16	8	8	15	22	16	16	16	24	4	2	15	9	0	2	202	3	207	188
Subdural hemorrhage following injury with open intracranial wound, with no loss of consciousness (disorder) - 209958006	19	20	10	12	8	13	17	9	17	20	29	16	6	2	0	7	1	201	3	205	186
Subdural hemorrhage following injury with open intracranial wound, with less than 1 hour loss of consciousness (disorder) - 209959003	19	20	10	12	8	13	17	9	17	20	29	16	6	2	0	7	1	201	3	205	186
Subdural hemorrhage following injury with open intracranial wound, with 1-24 hours loss of consciousness (disorder) - 209960008	19	20	10	12	8	13	17	9	17	20	29	16	6	2	0	7	1	201	3	205	186
Subdural hemorrhage following injury with open intracranial wound, with more than 24 hours loss of consciousness and return to pre-existing conscious level (disorder) - 209961007	19	20	10	12	8	13	17	9	17	20	29	16	6	2	0	7	1	201	3	205	186
Subdural hemorrhage following injury with open intracranial wound, with more than 24 hours loss of consciousness without return to pre-existing conscious level (disorder) - 209963005	19	20	10	12	8	13	17	9	17	20	29	16	6	2	0	7	1	201	3	205	186
Percutaneous thrombolysis of common iliac artery (procedure) - 233336006	20	16	9	0	0	8	26	26	29	28	22	0	8	5	8	0	2	200	3	205	185
Primary open reduction of fracture of patella and fixation with tension band wiring (procedure) - 310865004	23	20	5	3	2	6	24	16	16	16	39	37	0	0	1	0	1	205	2	208	185
Sacral epidural neurolysis (procedure) - 231392003	16	20	12	12	16	19	28	4	4	4	27	1	12	21	4	0	2	196	2	200	184
Cervical epidural neurolysis (procedure) - 231390006	16	20	12	12	19	17	27	4	4	4	27	0	12	21	4	0	2	195	2	199	183
Thoracic epidural neurolysis (procedure) - 231391005	16	20	12	12	19	17	27	4	4	4	27	0	12	19	4	0	2	193	2	197	181
Subarachnoid hemorrhage following injury without open intracranial wound AND with concussion (disorder) - 4332009	18	19	13	16	12	14	20	18	20	18	26	2	3	0	0	0	1	194	4	199	181
Endoscopic laser destruction of lesion of trachea using rigid bronchoscope (procedure) - 173084001	15	15	9	3	1	7	32	26	26	26	25	0	0	10	0	0	1	192	2	195	180

Percutaneous thrombolysis of coronary artery (procedure) - 232731009	18	13	13	3	0	12	34	20	21	20	24	0	7	5	8	0	2	194	2	198	180
Reduction of fracture of fibula with internal fixation (procedure) - 12077004	19	14	4	1	0	5	27	22	22	22	31	29	0	0	0	1	2	193	2	197	178
Cystoscopy and bulb catheter ureteropyelography (procedure) - 236197000	15	23	18	8	6	7	34	12	14	12	23	4	0	16	0	0	1	187	4	192	177
Percutaneous thrombolysis of internal iliac artery (procedure) - 240936000	20	17	9	0	0	8	16	25	27	28	27	0	7	5	8	0	2	193	2	197	177
Open reduction of fracture of arm with internal fixation (procedure) - 81334001	19	18	5	3	2	6	21	16	16	16	28	27	0	0	18	0	3	188	4	195	176
Endoscopic fiberoptic laser destruction of lesion below trachea (procedure) - 173153000	14	10	9	4	3	5	29	26	26	26	24	3	0	10	0	0	2	184	3	189	175
Percutaneous thrombolysis of profunda femoris (procedure) - 233340002	19	12	7	0	0	4	27	26	29	26	22	1	8	5	8	0	2	189	3	194	175
Percutaneous thrombolysis of popliteal artery (procedure) - 233341003	19	12	7	0	0	4	27	26	29	26	22	1	8	5	8	0	2	189	3	194	175
Endoscopic retrograde cholangiopancreatography with biopsy (procedure) - 6157006	23	19	19	4	4	6	20	16	17	16	29	17	8	0	0	0	1	195	2	198	175
Fiberoptic endoscopic removal of foreign body from trachea (procedure) - 173142006	13	14	14	11	0	14	33	18	18	18	22	4	0	8	0	0	1	183	3	187	174
Revision to closed reduction of fracture dislocation and external fixation (procedure) - 179685005	28	24	7	6	4	9	19	8	8	8	23	1	29	0	27	0	2	197	2	201	173
Extradural hemorrhage following injury without open intracranial wound AND with brief loss of consciousness (less than one hour) (disorder) - 40549004	16	16	9	10	6	11	17	16	18	18	26	4	18	4	0	0	1	183	5	189	173
Reduction of closed shoulder dislocation with fracture of greater tuberosity with manipulation (procedure) - 81337008	17	11	3	2	2	10	20	14	14	14	30	3	6	25	15	3	3	180	6	189	172
Subarachnoid hemorrhage following injury without open intracranial wound AND with moderate loss of consciousness (1-24 hours) (disorder) - 111673001	17	18	11	12	8	11	17	18	21	20	28	2	1	4	0	0	1	182	5	188	171
Cystourethroscopy with insertion of radioactive substance (procedure) - 14826006	18	11	18	6	3	8	29	20	19	16	26	0	0	1	14	0	2	185	2	189	171
Cystoscopic upward disimpaction of ureteric calculus (procedure) - 236191004	15	11	13	8	2	7	27	24	24	24	21	0	0	4	0	5	1	182	2	185	170
Open reduction of open mandibular fracture with interdental fixation (procedure) - 73915007	16	12	11	2	2	7	23	20	20	20	30	21	0	2	0	0	1	183	2	186	170
Primary posterior decompression lumbar spine and fusion (procedure) - 178603007	17	11	7	10	10	15	18	12	12	12	22	22	18	0	0	0	1	183	2	186	169
Ureteroscopic replacement of ureteric stent (procedure) - 236188004	16	10	14	4	4	13	29	22	22	22	19	0	0	2	8	0	1	182	2	185	169
Small incision phakoemulsification cataract and insertion of intraocular lens (procedure) - 313999004	11	15	8	2	9	11	16	12	12	12	19	10	9	22	11	0	3	160	16	179	168
Revision to open reduction of fracture dislocation and skeletal traction (procedure) - 179655000	23	13	6	0	0	6	11	12	8	8	30	30	24	0	19	0	2	186	2	190	167
Primary open reduction of fracture of neck of femur and open fixation using dynamic hip screw (procedure) - 239285007	26	21	10	3	2	6	32	16	12	12	27	26	0	0	0	0	1	189	3	193	167
Open reduction of fracture of tarsals and metatarsals with internal fixation (procedure) - 359932000	19	16	15	4	4	6	18	12	12	12	34	34	0	0	0	0	2	182	2	186	167
Primary open reduction spinal fracture and external fixation (procedure) - 178719008	23	14	11	3	0	2	22	22	22	22	23	21	4	0	0	0	1	186	2	189	166
Revision to open reduction of fracture and internal fixation with screw(s) (procedure) - 179083004	19	14	9	5	4	4	20	12	20	18	23	16	21	0	0	0	3	180	2	185	166
Reduction of fracture of leg with internal fixation (procedure) - 62864006	17	15	6	3	2	5	23	16	16	16	26	30	8	0	0	0	1	180	2	183	166
Subarachnoid hemorrhage following injury without open intracranial wound AND with brief loss of consciousness (less than one hour) (disorder) - 87020000	17	18	11	12	8	11	17	18	20	20	28	2	1	0	0	0	1	177	5	183	166
Bronchoscopic destruction of lesion - laser (procedure) - 232598003	11	8	8	4	2	6	26	27	26	26	20	0	2	10	0	0	1	173	2	176	165
Local anesthetic cervical epidural block (procedure) - 302336001	26	16	10	11	12	14	25	12	15	12	17	10	2	3	6	0	1	183	7	191	165
Extradural hemorrhage following injury without open intracranial wound AND with concussion (disorder) - 77498000	17	17	9	10	6	10	18	16	16	16	23	4	12	8	0	0	1	177	4	182	165
Incision and drainage of submaxillary abscess, extraoral (procedure) - 16697005	14	13	5	4	4	6	11	17	10	10	21	9	19	8	27	0	2	174	2	178	164
Extradural hemorrhage following injury with open intracranial wound, with no loss of consciousness (disorder) - 209978003	18	19	7	10	6	16	19	17	16	16	23	4	3	8	0	0	1	178	3	182	164
Reduction of fracture of tibia with internal fixation (procedure) - 21635002	18	15	5	3	2	5	22	16	16	16	26	29	9	0	0	0	2	176	4	182	164
Extradural hemorrhage following injury without open intracranial wound AND with prolonged loss of consciousness (more than 24 hours) AND return to pre-existing conscious level (disorder) - 84792001	16	16	9	10	6	11	17	16	16	16	23	4	16	4	0	0	1	174	5	180	164
Subdural hemorrhage following injury with open intracranial wound AND moderate loss of consciousness (1-24 hours) (disorder) - 90165008	15	15	8	9	4	8	17	18	21	20	28	2	9	5	0	0	1	173	5	179	164
Extradural hemorrhage following injury without open intracranial wound AND with prolonged loss of consciousness (more than 24 hours) without return to pre-existing conscious level (disorder) - 16907002	16	16	9	10	6	11	17	16	16	16	23	4	15	4	0	0	1	173	5	179	163
Fiberoptic endoscopic aspiration of trachea (procedure) - 173141004	14	21	18	5	2	7	30	14	14	14	25	5	0	8	0	0	3	171	3	177	163
Subarachnoid hemorrhage following injury without open intracranial wound (disorder) - 28048009	16	18	11	12	8	10	18	19	19	18	24	3	2	1	0	0	1	175	3	179	163
Thromboendarterectomy with graft of subclavian artery by neck incision (procedure) - 69489008	15	15	2	2	0	9	18	8	8	8	23	11	24	6	29	0	1	175	2	178	163
Rigid esophagoscopy and injection sclerotherapy of varices (procedure) - 173660005	21	13	15	5	4	6	24	18	18	18	24	1	0	16	0	0	3	175	5	183	162
Diagnostic endoscopic examination of sigmoid colon and	19	16	13	9	2	2	18	22	22	22	23	4	0	0	0	8	1	177	2	180	161

biopsy of lesion of sigmoid colon using rigid sigmoidoscope (procedure) - 174221009																					
Percutaneous transluminal coronary thrombolysis using streptokinase (procedure) - 175071008	19	9	9	4	4	15	30	16	17	16	21	0	7	5	8	0	2	175	3	180	161
Primary closed reduction and internal fixation of proximal femoral fracture with screw/nail device alone (procedure) - 179103000	16	14	2	2	0	3	16	12	12	12	30	34	24	0	0	0	2	171	4	177	161
Primary open reduction of fracture dislocation and functional bracing (procedure) - 179614003	15	16	2	0	0	3	4	6	4	4	25	34	8	31	24	0	1	173	2	176	161
Subdural hemorrhage following injury with open intracranial wound AND concussion (disorder) - 43262000	16	16	10	13	8	13	18	12	15	12	21	2	9	12	0	0	1	172	4	177	161
Pereyra operation for paraurethral suspension (procedure) - 82502009	11	8	12	15	14	14	19	14	14	14	21	2	0	4	10	0	1	168	3	172	161
Incision and drainage of submaxillary abscess, intraoral (procedure) - 15068003	14	13	5	4	4	6	11	22	16	16	22	9	22	10	0	0	2	170	2	174	160
Revision to closed reduction spinal fracture and external fixation (procedure) - 178712004	21	14	5	5	2	4	20	8	8	8	30	30	25	0	0	1	2	177	2	181	160
Revision to closed reduction and internal fixation of proximal femoral fracture with screw/nail device alone (procedure) - 179106008	18	10	3	1	0	3	17	12	12	12	28	29	17	0	0	16	1	175	2	178	160
Cerclage wiring of fracture (procedure) - 257948005	22	17	6	5	2	6	27	20	20	20	22	7	0	8	0	0	3	176	3	182	160
Subarachnoid hemorrhage following injury without open intracranial wound AND with no loss of consciousness (disorder) - 35672006	17	18	11	12	8	11	17	18	18	18	24	2	3	0	0	0	1	171	5	177	160
Subarachnoid hemorrhage following injury without open intracranial wound AND with loss of consciousness (disorder) - 69458009	19	19	11	12	8	11	17	18	18	18	25	2	1	0	0	0	1	170	8	179	160
Extradural hemorrhage following injury without open intracranial wound AND with loss of consciousness (disorder) - 90178008	16	16	9	10	6	11	18	17	17	16	23	5	12	0	0	0	1	171	4	176	160
Transurethral resection of female bladder neck (procedure) - 176246007	15	15	14	5	4	8	28	18	21	18	20	2	0	6	0	0	1	171	2	174	159
Primary open reduction of fracture and internal fixation with tension band wiring (procedure) - 179034003	21	15	5	3	2	2	19	16	16	16	33	32	0	0	0	0	1	177	2	180	159
Revision to closed reduction of fracture dislocation and skin traction (procedure) - 179682008	20	17	9	2	2	5	16	4	4	4	26	10	19	18	23	0	5	168	6	179	159
Subarachnoid hemorrhage following injury without open intracranial wound AND with prolonged loss of consciousness (more than 24 hours) AND without return to pre-existing conscious level (disorder) - 20276007	17	18	11	12	8	11	17	18	18	18	25	2	1	0	0	0	1	170	5	176	159
Congenital fistulae between uterus and digestive and urinary tracts (disorder) - 204847000	14	19	6	8	9	13	18	12	12	12	17	0	22	4	0	7	2	169	2	173	159
Subarachnoid hemorrhage following injury without open intracranial wound AND with prolonged loss of consciousness (more than 24 hours) AND return to pre-existing conscious level (disorder) - 88747000	17	18	11	12	8	11	17	18	18	18	25	2	1	0	0	0	1	170	5	176	159
Bronchoscopy with destruction of tumor by laser surgery (procedure) - 18049000	14	8	6	1	1	6	25	22	22	22	26	3	0	16	0	0	2	167	3	172	158
Epidural neurolysis (procedure) - 231389002	18	18	14	8	18	16	21	0	0	0	24	2	12	21	4	0	2	170	4	176	158
Closed reduction of fracture of tibia and fibula with internal fixation (procedure) - 79670000	20	14	4	4	2	4	22	18	18	18	25	28	0	0	0	0	3	172	2	177	157

5.4 Impact of changes on concepts and relationships

Changes affecting individual concepts can be quite dramatic over time. As an example, Figure 1 shows the impact of taxonomy changes to the concept '*Cell phenotyping performed (situation)*' for all versions since January 2002 until July 2010. The arrows represent the relationships as they are found in the relationships table and are labeled - right or above the arrow - with the preferred name of the relationship and the period(s) during which it was stated to hold; a relationship labeled, for instance, '0501-1007', became thus introduced in the January 2005 version and has been present all the time since then. The arrows are further color coded for quick visualization of their history: red, green and blue mean, respectively, that a relationship is found prior to the previous version, in the previous version, and in the last version.

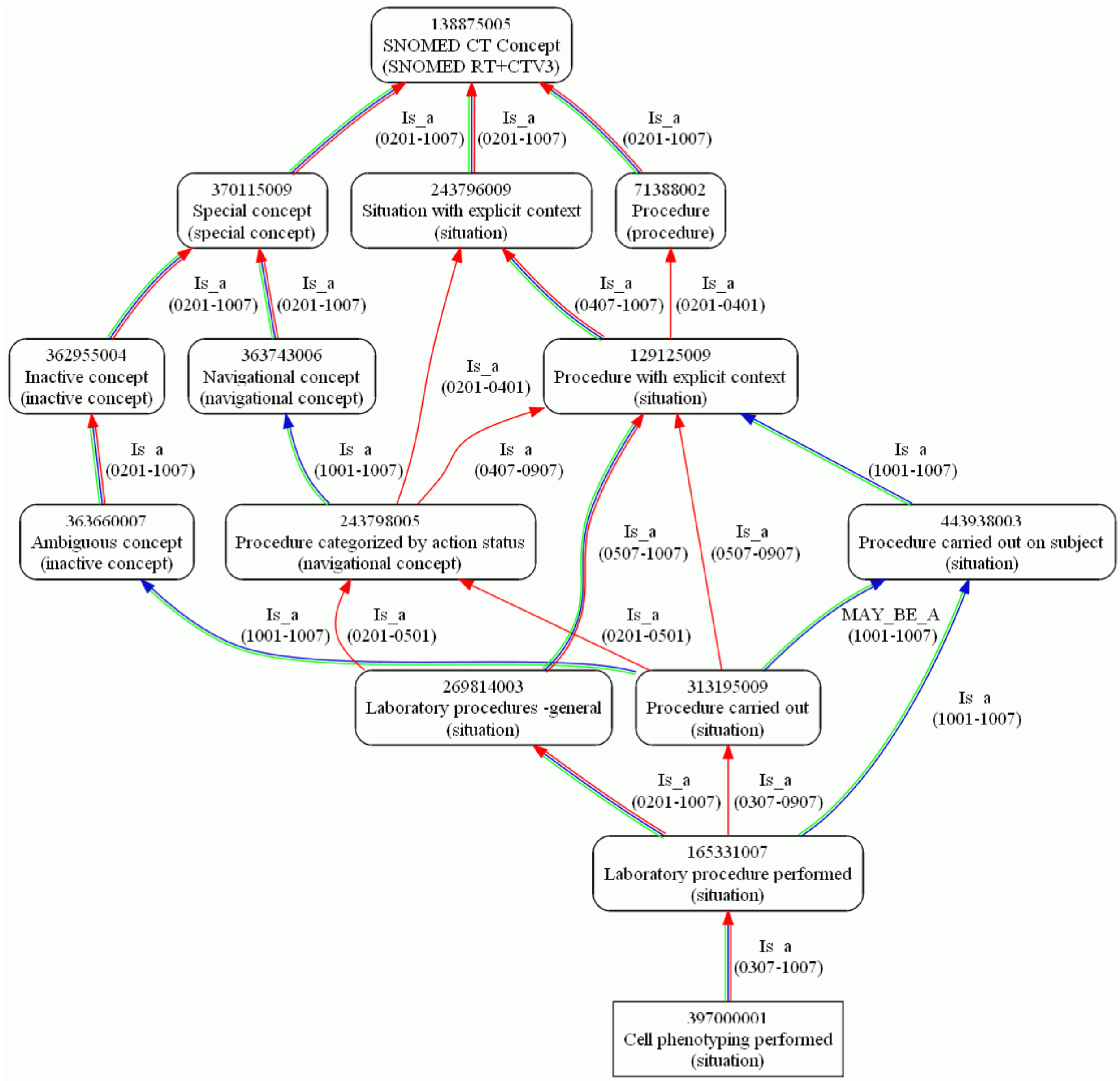

Figure 1: Impact of SNOMED CT revisions on the classification of SNOMED CT concept 397000001 with Fully Specified Name '*Cell phenotyping performed (situation)*'.

A similar view is provided by **Table 11** which shows transitive closure set computations for the concept '*Surgical margins involved by tumor*'. For this concept, all concepts - referred to as *target concepts* - within the transitive closure set of the *Is a* relation and all hierarchical relations - *Was A*, *Replaced By*, *Same As*, *May Be*, *Moved To*, and *Moved From* - were computed for each SNOMED CT version from January 2002 to July 2010, together with their concept status and path length towards the source concept. Computing the transitive closure set involves traversing the target of each of the relationships included in the Relationships Table of each version to look for and follow further relationships until all paths through the hierarchy reach the root concept (closure). When a target concept could be reached by traversing more than one path, the shortest path length from source concept to target concept was preserved.

Table 11: Transitive closure sets for the source concept '44228008: *Surgical margins involved by tumor* (finding)'

Rel-ID	Version																		Rel-Type	Target Concept
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18		
H-18608	6	6	7	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	Is a	SNOMED CT Concept (SNOMED RT+CTV3)
H-23694	5	5	6																Is a	Finding (finding)
H-18607	4	4	5																Is a	Finding by method (finding)
H-12792	3	3	4																Is a	Test finding (navigational concept)
H-12789	3	3	3																Is a	Laboratory test finding (navigational concept)
H-07371	2	2	2																Is a	Sample finding (finding)
H-07373	2	2																	Is a	Morphologic finding (finding)
220039029	1	1																	Is a	Clinical sample finding (finding)
H-07370			3																Is a	Histopathology finding (finding)
H-07368			2																Is a	General pathology (finding)
H-07369			2																Is a	Laboratory finding present (navigational concept)
2030386023			1																Is a	Pathology examination findings present (finding)
2030387025			1																Is a	Surgical margin finding (finding)
H-18182				8	8	8	8	8	8	8	8	8	8	8	7	6	6	6	IsA	SNOMED CT Concept (SNOMED RT+CTV3)
H-27662				7															IsA	Finding (finding)
H-23376				6	6	6	6	6	6	6	6	6	6	6					IsA	Finding by method (finding)
H-18183				5	5	5	5	5	5	5	5	5	5	5	5				IsA	Test finding (navigational concept)
H-07379				4	4	4	4	4	4	4	4	4	4	4	4	3	3	3	IsA	Histopathology finding (finding)
H-12412				4	4	4	4	4	4	4	4	4	4	4	4				IsA	Laboratory test finding (navigational concept)
H-07380				4	4	4	4	4	4	4	4	4	4	4	4				IsA	Sample finding (finding)
H-12418				4	4	4	4	4	4	4	4	4	4	4	4				IsA	Morphologic finding (finding)
H-07378				3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	IsA	General pathology (finding)
H-12793				3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	Is a	Special concept (special concept)
H-07377				3	3	3	3	3	3	3	3	3	3	3	3				IsA	Clinical sample finding (finding)
H-07381				3	3	3	3	3	3	3	3	3	3	3	3				IsA	Laboratory finding present (navigational concept)
H-07374				2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	Is a	Inactive concept (inactive concept)
H-07382				2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	IsA	Pathology examination findings present (finding)
H-07384				2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	IsA	Surgical margin finding (finding)
2228147020				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	Is a	Duplicate concept (inactive concept)
2295897028				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	SAME AS	Surgical margin involved by tumor (finding)
H-18186					7	7	7	7	7	7	7	7	7	7	6	5	5	5	IsA	Clinical finding (finding)
H-12417																4	4	4	IsA	Evaluation finding (finding)

Legend. ***Rel-ID***: relationship ID, either original component ID from SNOMED CT or generated during computation (preceded by 'H-'); ***Version***: digits represent the minimal path length, blank when the relationship is not present in a version; ***Rel-Type***: either an original relationship type from SNOMED CT (here 'Is a' when path length equals '1' and 'SAME AS') or a computed one following the transitivity principles outlined in Table 2; ***Target Concept***: the Fully Specified Name of a SNOMED-CT concept in the transitive closure set of the source concept.

Table 12 displays the rules used to compute the composite relationships during the transitive closure computation of this concept.

Table 12: Transitivity rules for relationships with distinct signature.

C1→C2	C2→C3	C1→C3
Is a	Is a	Is a
Is a	IsA	IsA
IsA	Is a	IsA
IsA	IsA	IsA
SAME AS	Is a	IsA
SAME AS	IsA	IsA
Is a	SAME AS	IsA
IsA	SAME AS	IsA

5.5 Inspecting occurrences of homonymy

To obtain another view on the change history of SNOMED CT from a different perspective, we generated graphs for all concepts in the July 2010 version that have descriptions participating in a relation of homonymy at any point in time during their history, the total being 48,332. The hierarchy of these concepts was constructed up to the highest level in the taxonomy for which historical attributes are available. An example is provided in **Figure 2**.

5.6 Motivations for versioning

There are at least three use cases that justify the existence of SNOMED CT's history mechanism.

One is the support that it can give to users who want to update data annotated with clinical codes from a previous version to conform to codes from the latest release. The Historical Relationships Table allows this to be done automatically for the relationships 'same as' and 'replaced by', while it can generate triggers for classes that have been found to be ambiguous.

A second use case is internal quality control: awareness of past mistakes may prevent SNOMED CT authors from making similar mistakes in the future.

The third use case relates to SNOMED CT's ambition of being a '*reference terminology for clinical data*', defined as '*a set of concepts and relationships that provides a common reference point for comparison and aggregation of data about the entire health care process, recorded by multiple different individuals, systems, or institutions*'. [86] But here, as will be demonstrated in the next section, the current mechanism falls short in many respects.

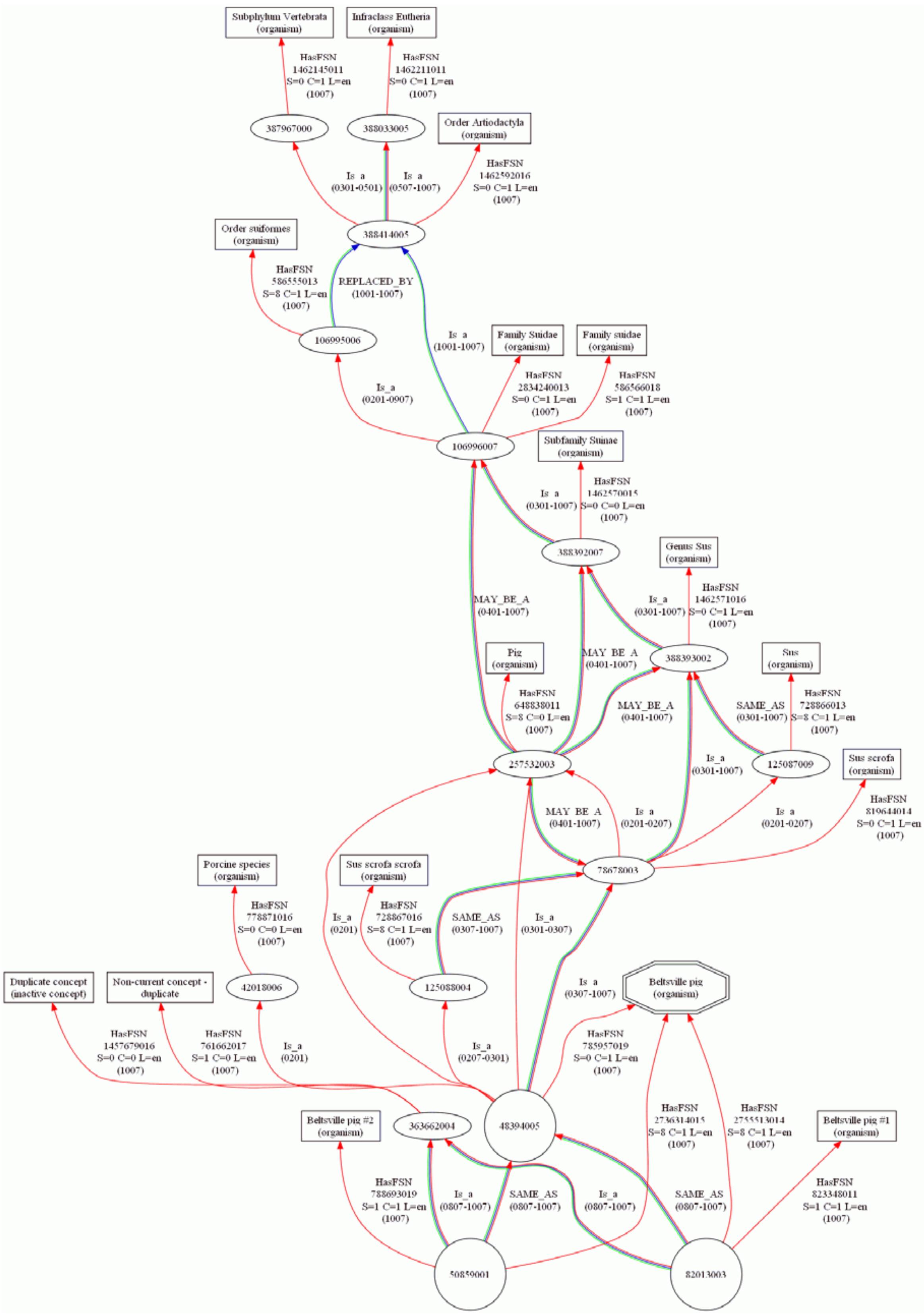

Figure 2: History of concepts involved in the homonymy relation of descriptions with the term 'Beltsville pig'

6 Would SNOMED CT benefit from realism-based versioning ?

Clearly, the history mechanism implemented in SNOMED CT provides insight into how the system has evolved over time, giving information primarily about the sorts of actions its authors undertook in the course of time. It is an interesting resource into which a great deal of effort has been invested, but to the best of our knowledge it has not thus far been acknowledged in the literature or been the subject of research.

However, in its current form this mechanism does not do justice to needs of SNOMED CT as a reference terminology of international scope. To serve as such a common reference point, a terminology should faithfully capture the state of the art in the domain which it is intended to serve. Several studies have shown that (static) releases of SNOMED CT perform well in terms of *coverage* of the biomedical domain, [17, 87] but there has also been criticism of the way in which (like other terminology systems) it runs together (1) what is the case in the domain, (2) what clinicians believe and (3) what clinicians communicate [15, 71]. We have argued that these distinctions should be made more explicit. It is now clear that such criticisms can also be applied to SNOMED CT's history mechanism. There is sufficient evidence to show that changes in successive versions of SNOMED CT were often driven neither by changes believed to have occurred in the corresponding part of reality nor by changes in our scientific understanding of that part of reality to which the given SNOMED CT classes are supposed to refer. In cases where release changes do reflect a change that is external to SNOMED CT, we are left uninformed about whether the change was in reality, or in our understanding thereof.

6.1 A few case studies

6.1.1 Ehrlichia risticii

So we found that the class *Ehrlichia risticii* was removed from release 2007-01-31 because it was deemed to be 'outdated', which means: '*withdrawn from current use because it is no longer recognized as a valid clinical concept*'.[47] This leaves open the question whether it is only *now* that the concept is deemed to be no longer valid, perhaps because the corresponding species died out, or whether the concept in question was *never* valid, because a corresponding species never existed at all but that it is only now that science has come to this insight. The additional information that we find for this case in the Historical Relationships Table does not add more clarity. We learn that the outdated class was 'replaced by' the newly added class '*Neorickettsia risticii'*, but, as for all classes added, no reason is given. Is this a new species that evolved from the former one? Is it a species that has long existed already, but has just been discovered? It is only on the basis of external information not distributed as part of the new release that we learn about a recommendation issued in 2001 to reclassify the genus of the organism because the previous classification of its genus was flawed.

6.1.2 'Saquinavir (free base) 200mg capsule (product)'

An extreme case is class 324847008 with the FSN '*Saquinavir (free base) 200mg capsule (product)*' which between the January 2002 and January 2007 versions underwent 8 modifications at the level of the concept. Although being a true outlier, it is an interesting case to demonstrate the various types of changes introduced.

This concept, referred to in what follows as C1, was introduced in the first version of SNOMED CT – it was not present in either SNOMED-RT or CTV3 –, and was initially associated with the FSN '*Saquinavir (free base) 200mg capsule (substance)*' and the preferred term'*Saquinavir (free base) 200mg capsule*'. This original FSN was 'retired without any specified reason' in release 2002-07-31 and replaced by '*Saquinavir (free base) 200mg capsule (product)*'. The same type of modification was applied to all other substances. This explains the high number of changes in release 2002-07-31, in which there were a number of minor concept changes without concept retirement (11,766, see **Table 8** page 30).

In 2003-01-31, C1 was declared inactivated because of duplication with another concept and as a consequence, its associated terms were annotated as being descriptions for a retired concept. The duplicating concept was '*Saquinavir mesylate 200mg capsule (product)*' (C2) which had been earlier added in release 2002-07-31. At that time, it was also noticed that a third concept (C3), with the FSN '*Saquinavir (as mesylate) 200mg capsule (product)*', had been already included in the first version of SNOMED CT (again without a prior appearance in either SNOMED-RT or CTV3). This concept, too, was rendered inactive as a duplicate of C2.

With release 2004-01-31, the SNOMED CT authors changed their minds. They re-activated C3 while deactivating C2, thereby still declaring that both are representations of the same concept by means of the SAME AS relation in the Historical Relationships Table. At the same time they reactivated C1, thereby deeming it to be no longer a duplicate of C2. From then on, C1 started to lead a life of its own. It became deactivated once again in 2004-07-31, being considered a duplicate of C3. It was reactivated in 2005-01-31, and deactivated (for the third time) in 2005-07-31, thereby again being declared a duplicate of C3, whose FSN in 2004-07-31 was changed to '*Saquinavir mesylate 200mg capsule (product)*', surprisingly (or not?) the very same FSN which was assigned to C2, which had been retired in 2004-01-31.

In 2007-01-31, a new concept, with conceptId 422836001, (C4) was added to SNOMED CT, and was given the FSN '*Saquinavir mesylate 200mg capsule (product)*'! C2, still deactivated because of duplication, was then 'replaced' by C4 with the motivation that it (C2) contains an error, while C3 was also deactivated and 'replaced' by C4 for the same reason. At the same time, the concept (C5) with FSN '*Saquinavir 200mg capsule (product)*' which was incorporated in the first version of SNOMED CT as an active concept – although with a status of having 'limited' clinical value – through the integration of CTV3, was inactivated for being 'ambiguous', and accordingly further annotated in the Historical Relationship Table as being 'may be a' C4 and 'may be a' C1.

Activating and deactivating the C1 class ('*Saquinavir (free base) 200mg capsule (product)*') had nothing to do with the appearance or disappearance of that product from the market. Both saquinavir mesylate (Invirase) and saquinavir (Fortovase) were already approved by the FDA (on December 6, 1995, and November 7, 1997, respectively) as antiretroviral protease inhibitors that act by blocking a protein that HIV needs to replicate itself. This activation and deactivation had nothing to do, either, with any change in our scientific understanding of these protease inhibitors. Users of SNOMED are left to attempt to infer from insufficient information what the motivation might have been not only for the changes mentioned but for a wide variety of other sorts of changes, including all additions of classes to SNOMED CT.

6.2 Evolution of SNOMED CT's concept model

SNOMED CT's documentation and its Concept Model as reflected in the Linkage Attributes were studied for all releases from January 2002 to July 2010. To assess the evolution of the Concept Model, we generated from the relationship tables included in each version a graph representing the relationships actually used in linking conceptIDs from one hierarchy to conceptIDs from the same or another hierarchy, thereby keeping track in each version of the number of times a specific relation, e.g. '*USING DEVICE*' was used in relation to the status, e.g. *'current'*, *'ambiguous'*, etc., between specific hierarchies.

As an example, the relationship *'Computerized tomography guided biopsy of brain (procedure) → METHOD → Biopsy – action (qualifier value)*' in version V would increment the occurrence count of the 5-tuple *'procedure – (0) → METHOD → qualifier value – (0)*' for version V where '0' indicates the status *'current'*.

We found astonishing results such as

- '*substance → SAME AS → procedure*'
- *'event → MAY BE → navigational concept*',
- '*person → MOVED TO → namespace concept*'
- '*physical object → IS A → inactive concept*'.

Clearly statements of the latter sort do not have the same kind of meaning as '*procedure → METHOD → physical object*' [9]. The former are statements about the concepts as representational units in SNOMED CT itself (i.e. meta-language statements), while the latter is a statement about the referents of these concepts (an object-language statement). The problem arises because SNOMED CT does not assign, in contrast to entries in the Description and Relationships Table, a separate component ID to an entry in the Concept Table.

7 Realism-based versioning applied to SNOMED CT

The main hypothesis motivating our research is that an objective metric for ontology quality can be developed when ontology authors systematically document the reason for which they added, deleted, split or merged classes in a new version of an ontology. The only requirement would be that these reasons are expressed in terms of what changes occurred in that part of reality that the ontology is intended to cover, or in the authors interpretation or representation thereof. We selected SNOMED CT as a test case because of the availability of history information of some sort.

7.1 Methods

To perform our analysis, we used the versions from SNOMED CT released between January 2002 and July 2009. Applying this methodology to a terminology the size of SNOMED CT - i.e. retrospectively provide manually for each concept what sort of change under the realism-based perspective - would be an impossible task. But here, for demonstration purposes, we assume naively that with each release, its authors assume in good faith that all its constituent expressions are of the correct type: active concepts should be of type ‘P+1’ while inactive ones either ‘A+1’ or ‘A+2’ (see **Table 1** p.23). The further assumption that the authors advance with each release the terminology as complete, i.e. as containing RUs designating *all* **POR**s deemed relevant to SNOMED CT’s purpose, does likely not hold but adopting it allows us to use a new version as a benchmark for all previous ones, while still remaining faithful to the realist agenda.

To avoid individual inspection of each term and concept, we applied a number of principles to project a change made in each version onto an error – if any at all – in all previous versions. First, if a newly introduced RU was never inactivated, there had to be an unjustified absence in each version prior to the addition, and a justified presence starting with the version in which the addition was introduced. Second, if an RU was found to have been made inactive and this action was never undone, there was a justified absence both prior to the introduction of the corresponding RU and after it was inactivated (including the version in which the RU was made inactive), and an unjustified presence in each version that contained the RU. If a RU, made inactive previously, was found to be re-introduced, then there must have been an unjustified absence prior to the addition, a justified presence after the addition until the RU was inactivated, again an unjustified absence after the latter change, and finally a justified presence from the point of re-introduction onwards.

Table 13: mapping SNOMED CT 'reasons for change' to realism-based error-types

CT	Existing concept made …	Error Type
0	active: in current use	A-1
1	inactive: ‘retired’ without a specified reason	P-1
2	inactive: withdrawn because duplication	P-9
3	inactive because no longer recognized as a valid clinical concept (outdated)	P-1
4	inactive because inherently ambiguous.	P-4
5	inactive because found to contain a mistake	P-1
6	active with limited clinical value (classification concept or an administrative definition)	A-1
10	inactive because moved elsewhere	P-6
11	pending move	P-6

Legend: CT: concept status as defined in SNOMED CT; Error Type: corresponding error in previous version according to the typology described in Table 1, p23.

For those cases in which SNOMED CT provides a reason for the change, a mapping was established as outlined in **Table 13** for changes in concept status. A similar mapping was performed for changes in the status of descriptions. Inactivation of descriptions because of inactivation of the corresponding concept was considered to reflect a justifiable absence and was thus not counted as an error. Adding or removing relationships were taken into account as well, but not changes in the refinability status or their inclusion or withdrawal from a role group.

7.2 Results

The changes that SNOMED CT underwent in its core components during the period studied are enormous: 8,361,989, of which 583,292 at the level of concepts and 1,528,653 concerning descriptions (including all introductions in the first release).

Table 14: Realism-based quality evolution of SNOMED CT

VERSION	RUTYPE	T0201	T0207	T0301	T0307	T0401	T0407	T0501	T0507	T0601	T0607	T0701	T0707	T0801	T0807	T0901	T0907
T0201	Concepts	100.0%	97.5%	95.5%	93.4%	92.2%	91.1%	90.4%	90.0%	89.5%	88.9%	88.2%	87.6%	87.1%	86.0%	84.7%	84.2%
	Descriptions	100.0%	94.0%	90.3%	88.1%	86.6%	85.1%	84.4%	83.8%	81.7%	80.3%	79.4%	78.3%	77.8%	71.0%	70.4%	70.1%
	Relationships	100.0%	60.6%	52.4%	50.0%	48.9%	45.9%	43.6%	42.9%	41.8%	41.5%	41.8%	41.3%	40.8%	37.1%	36.3%	36.0%
	TOTAL	100.0%	75.7%	69.1%	66.7%	65.5%	62.9%	61.1%	60.5%	59.1%	58.5%	58.8%	58.1%	57.6%	53.1%	52.4%	52.1%
T0207	Concepts		100.0%	97.6%	95.2%	94.0%	92.8%	92.1%	91.7%	91.1%	90.5%	89.7%	89.2%	88.6%	87.5%	86.1%	85.6%
	Descriptions		100.0%	95.6%	93.1%	91.4%	89.8%	89.0%	88.2%	86.0%	84.5%	83.6%	82.4%	81.8%	74.9%	74.3%	73.9%
	Relationships		100.0%	77.1%	72.9%	71.2%	65.9%	62.1%	61.1%	59.4%	58.9%	55.9%	55.1%	54.3%	49.3%	48.2%	47.7%
	TOTAL		100.0%	85.3%	81.8%	80.2%	76.3%	73.7%	72.8%	71.0%	70.2%	68.2%	67.4%	66.7%	61.4%	60.4%	60.0%
T0301	Concepts			100.0%	97.6%	96.2%	94.9%	94.2%	93.8%	93.2%	92.6%	91.8%	91.2%	90.6%	89.4%	88.0%	87.5%
	Descriptions			100.0%	97.3%	95.4%	93.7%	92.9%	92.1%	89.7%	88.1%	87.2%	85.8%	85.2%	77.7%	77.0%	76.7%
	Relationships			100.0%	80.3%	78.1%	71.4%	66.4%	65.2%	63.0%	62.4%	58.9%	58.0%	57.0%	51.6%	50.4%	49.8%
	TOTAL			100.0%	87.7%	85.6%	80.9%	77.5%	76.5%	74.3%	73.5%	71.1%	70.1%	69.3%	63.6%	62.5%	62.0%
T0307	Concepts				100.0%	98.6%	97.3%	96.5%	96.1%	95.4%	94.8%	94.0%	93.4%	92.8%	91.5%	90.1%	89.6%
	Descriptions				100.0%	98.1%	96.3%	95.4%	94.5%	92.1%	90.5%	89.5%	88.1%	87.5%	79.9%	79.2%	78.8%
	Relationships				100.0%	95.5%	85.9%	78.8%	77.1%	74.2%	73.5%	62.1%	61.1%	60.0%	53.9%	52.6%	51.8%
	TOTAL				100.0%	96.7%	90.6%	86.1%	84.7%	82.2%	81.2%	74.0%	73.0%	72.1%	65.8%	64.7%	64.1%
T0401	Concepts					100.0%	98.7%	97.9%	97.4%	96.7%	96.1%	95.3%	94.6%	94.0%	92.7%	91.3%	90.8%
	Descriptions					100.0%	98.2%	97.2%	96.3%	93.8%	92.1%	91.1%	89.7%	89.1%	81.4%	80.7%	80.3%
	Relationships					100.0%	88.7%	80.7%	79.0%	75.9%	75.1%	63.4%	62.4%	61.2%	54.8%	53.5%	52.6%
	TOTAL					100.0%	92.9%	87.9%	86.5%	83.9%	82.8%	75.4%	74.3%	73.4%	66.9%	65.8%	65.1%
T0407	Concepts						100.0%	99.2%	98.7%	98.0%	97.4%	96.5%	95.9%	95.2%	93.9%	92.5%	91.9%
	Descriptions						100.0%	99.1%	98.1%	95.2%	93.5%	92.5%	91.1%	90.4%	82.6%	81.9%	81.5%
	Relationships						100.0%	87.4%	85.1%	80.9%	80.0%	67.1%	65.8%	64.4%	56.9%	55.5%	54.4%
	TOTAL						100.0%	92.5%	90.8%	87.4%	86.2%	78.2%	77.0%	75.9%	68.7%	67.5%	66.7%
T0501	Concepts							100.0%	99.5%	98.8%	98.2%	97.3%	96.6%	96.0%	94.7%	93.2%	92.7%
	Descriptions							100.0%	99.0%	96.1%	94.4%	93.3%	91.9%	91.2%	83.4%	82.7%	82.3%
	Relationships							100.0%	93.9%	88.4%	87.3%	72.8%	71.1%	69.4%	60.6%	59.0%	57.7%
	TOTAL							100.0%	96.2%	92.1%	90.8%	82.0%	80.5%	79.2%	71.3%	70.0%	69.1%
T0507	Concepts								100.0%	99.3%	98.6%	97.8%	97.1%	96.5%	95.2%	93.7%	93.1%
	Descriptions								100.0%	97.1%	95.3%	94.3%	92.8%	92.1%	84.2%	83.5%	83.1%
	Relationships								100.0%	91.2%	90.0%	74.8%	73.0%	71.2%	61.8%	60.2%	58.7%
	TOTAL								100.0%	94.1%	92.7%	83.6%	82.0%	80.6%	72.4%	71.0%	69.9%
T0601	Concepts									100.0%	99.3%	98.4%	97.8%	97.1%	95.8%	94.3%	93.8%
	Descriptions									100.0%	98.1%	97.0%	95.4%	94.7%	86.6%	85.9%	85.5%
	Relationships									100.0%	95.4%	78.5%	76.4%	74.4%	64.4%	62.3%	60.7%
	TOTAL									100.0%	96.8%	86.6%	84.9%	83.4%	74.7%	73.1%	72.0%
T0607	Concepts										100.0%	99.1%	98.4%	97.8%	96.4%	94.9%	94.4%
	Descriptions										100.0%	98.9%	97.3%	96.5%	88.3%	87.5%	87.1%
	Relationships										100.0%	79.9%	77.7%	75.5%	65.3%	63.1%	61.4%
	TOTAL										100.0%	88.1%	86.3%	84.7%	75.8%	74.1%	73.0%
T0701	Concepts											100.0%	99.3%	98.7%	97.3%	95.8%	95.2%
	Descriptions											100.0%	98.4%	97.6%	89.3%	88.6%	88.1%
	Relationships											100.0%	95.6%	91.7%	76.2%	73.1%	70.8%
	TOTAL											100.0%	97.0%	94.6%	83.1%	80.9%	79.4%
T0707	Concepts												100.0%	99.3%	98.0%	96.4%	95.9%
	Descriptions												100.0%	99.2%	90.9%	90.1%	89.6%
	Relationships												100.0%	95.3%	78.5%	75.2%	72.8%
	TOTAL												100.0%	97.2%	85.0%	82.7%	81.1%
T0801	Concepts													100.0%	98.6%	97.1%	96.5%
	Descriptions													100.0%	91.6%	90.8%	90.3%
	Relationships													100.0%	81.1%	77.5%	74.8%
	TOTAL													100.0%	86.7%	84.3%	82.6%
T0807	Concepts														100.0%	98.4%	97.9%
	Descriptions														100.0%	99.1%	98.6%
	Relationships														100.0%	93.4%	88.3%
	TOTAL														100.0%	96.0%	93.0%
T0901	Concepts															100.0%	99.4%
	Descriptions															100.0%	99.5%
	Relationships															100.0%	93.7%
	TOTAL															100.0%	96.4%
T0907	Concepts																100.0%
	Descriptions																100.0%
	Relationships																100.0%
	TOTAL																100.0%

Table 14 demonstrates how the metrics just described can be used to obtain two distinct, yet closely related views. Read horizontally, the table shows for all versions how the quality of a specific version deteriorates in light of the state of the art represented in a more recent version. Read vertically, it shows how much of the state of the art in a more recent version was already accounted for in a previous version. As can be inferred from the formula of our metric and the principles for quantifying the error involved in mismatches, each version considers itself to be perfect as witnessed by the series of “100%” along the diagonal of the matrix.

Figure 3 and **Figure 4** depict these two views graphically over all versions analyzed. The trend lines marked with triangles, squares and circles correspond to changes in the concepts, descriptions and relationships respectively. The trend lines without markers depict the overall changes.

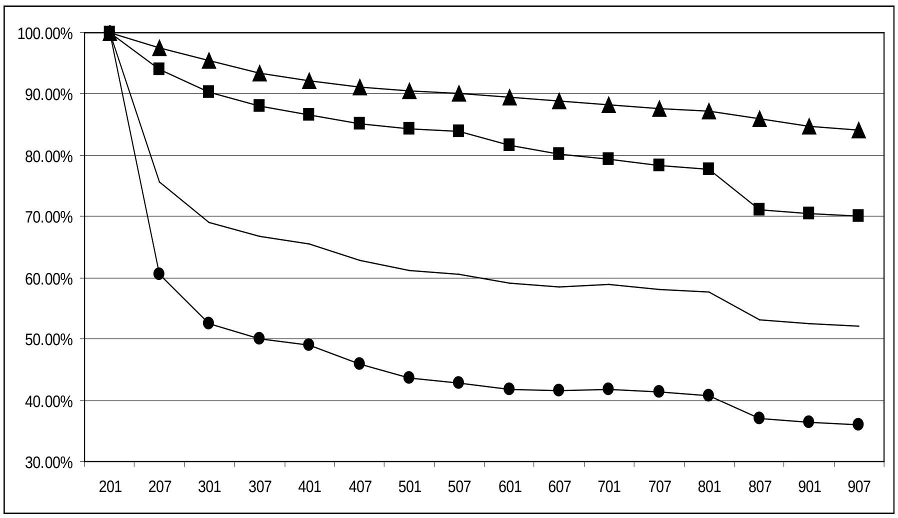

Figure 3: Evolutionary view on the Jan 2002 release of SNOMED CT since its inception (‘201’) until the July 2009 release (‘907’).

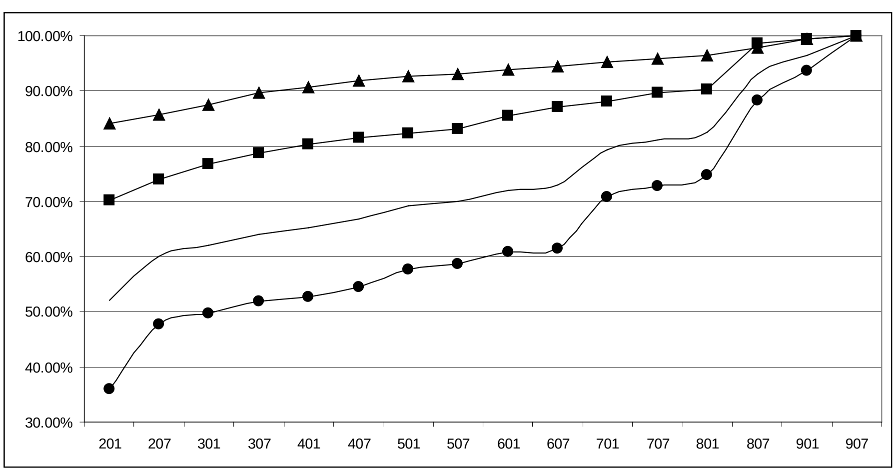

Figure 4: Evolutionary view on the relative increase in quality of SNOMED CT since its inception (‘201’) until the July 2009 release (‘907’).

7.3 Discussion

Under the assumptions entertained, the figures seem to indicate that with respect to concepts, only small quality improvements are introduced with each new version, i.e. roughly 2% with an overall quality improvement of about 16% since 2002. This need not be a negative finding for two reasons: (1) the proposed metric becomes less sensitive when the size of the terminology increases, and (2) it might very well be that SNOMED CT 'got it right' from the very beginning, since, after all, its real foundations were created almost 50 years ago. However, as suggested earlier, also a lack of resources to make necessary changes can be responsible. Changes in the descriptions exhibit larger improvements: 30% over the past 8 years. The biggest gains seem to be obtained in the relationships. However, several reflections need to be made.

For concepts, our analysis principles used thus far treat all new introductions as being unjustifiably missing in earlier versions. This is adequate for most types of concepts, except for pharmaceutical products – new products come on the market constantly – and certain information artifacts such as newly constructed rating scales or named guidelines and protocols: when such entities come into existence after the release of a SNOMED version, then absence of corresponding RUs in that and earlier versions is, of course, justifiable. Mistaking a justifiably absent concept for an unjustifiably present one for reasons of non-existence (P-1) makes a difference in error rate of 0 versus -3. The move of SNOMED CT to migrate brand-named products to extensions eliminates this problem, although the presence of brand-named products in versions before migration occurred needs to be judged as an unjustifiable presence for relevancy reasons (P-6, error: -1).

A second concern is the mapping between SNOMED CT's documented reasons for status changes and our reality-based interpretation. The main problem here is that the SNOMED documentation does not contain enough information on what precisely motivated its authors to introduce changes of a certain type, this on top of the fact that the status labels are rather ambiguous. Only status 'duplicate' can directly be translated into our P-9 configuration. For status changes 1, 3, 4 and 5 (**Table 13**) matters are less clear.

Our mapping is the best estimate that we could make on the basis of an analysis of a sample of 1000 randomly selected concepts (n=264) and descriptions (n=736) that underwent a status change of some sort, the goal of the analysis being to find some underlying principles. It turned out that all concepts with the status 'outdated' in our sample involved organisms, the change probably being introduced because of re-classification in the biology domain. We found them replaced by other concepts that nevertheless carry the original preferred name of the outdated concept as a synonym. The majority of concepts stated to be inactivated for reasons of 'ambiguity' do in our opinion not look ambiguous at all, as further witnessed by the fact that some of them have been replaced by a concept with an identical name, in addition to a more specific one. An example is 'breech extraction (procedure)' that became replaced by 'breech extraction (procedure)' and 'total breech extraction (procedure)'. If this line of thinking is to be taken seriously, then each concept which has 'children' is ambiguous. We assume that the main reason for this state of affairs is the correction of inadequate original assignments of synonyms such as 'partial X' and 'total X' for just 'X'. We did not find any principle underlying the assignment of 'inactive, reason not specified' and 'erroneous'. For the latter case, we spotted a few typographic mistakes, an issue which has little to do with whether or not there are corresponding entities in reality. For type 1 inactivation, we spotted, for example, occurrences where an earlier inactivation for reason of duplication was changed into an inactivation for unspecified reason (e.g. 'biological test (procedure)').

For sure, the assumptions described in the methods section are not valid from one version to another and the statistics obtained need to be assessed in that light. Lack of resources might for instance prevent changes to be introduced although the authors know it has to be done at some point. Having a better insight in the concrete reasons for change, would give a more accurate application of our proposed metric. This is certainly the case for the relationships, although here further work can be done: the disappearance of a relationship in a newer version might not be a real disappearance since the relationship might still be inferred from the graph structure underlying SNOMED CT. Figuring this out, however, requires a lot of computer effort and time over the entire taxonomy, but has been performed on a subset of concepts used in pathology reports (section 11).

7.4 Conclusion

Our method tries to answer two questions: (1) how much is a new version of a terminology better than any previous version and (2) to what degree do terminology changes reflect evolutions in the underlying domain or the terminology authors' understanding thereof. The answer to the first question, in the context of SNOMED CT, seems to be: not much, at least not for the concepts. This is in contrast to our findings on the application of our method to the Gene Ontology for which the same assumptions were used [26].

The answer to the second question is less straightforward. Close inspection of the documented motivations for status changes and new additions which are said to be '*driven by changes in understanding of health and disease processes; introduction of new drugs, investigations, therapies and procedures; new threats to health;…*' [88] reveals that the majority of them have little to do with changes in the domain or altered understanding thereof, but rather with the idiosyncrasies of SNOMED CT's representational framework: the distinction between 'concepts' and 'terms' is far less absolute than one would expect.

Our recommendation is that the SNOMED CT authors provide for future versions greater insight into the underlying reasons for changes they introduce and that they do this in a way that supports computation. Above all, we hope that our findings lead to further introspection on the appropriateness of the concept-based approach [56] for a resource as famous as SNOMED CT, or that, at least, more attention is given to the lack of ontological commitment. [32]

8 SNOMED CT versioning Wiki - 'RT-Wiki'

The RT-Wiki application is build on Mediawiki, a free open source wiki application, thereby borrowing some source and rendering ideas from Lexwiki[1], which is a collaborative concept authoring system, and BioMedGT. The goal of the system is to combine SNOMED CT's own history mechanism with realism-based ontology versioning in a browser environment.

8.1 User Interface

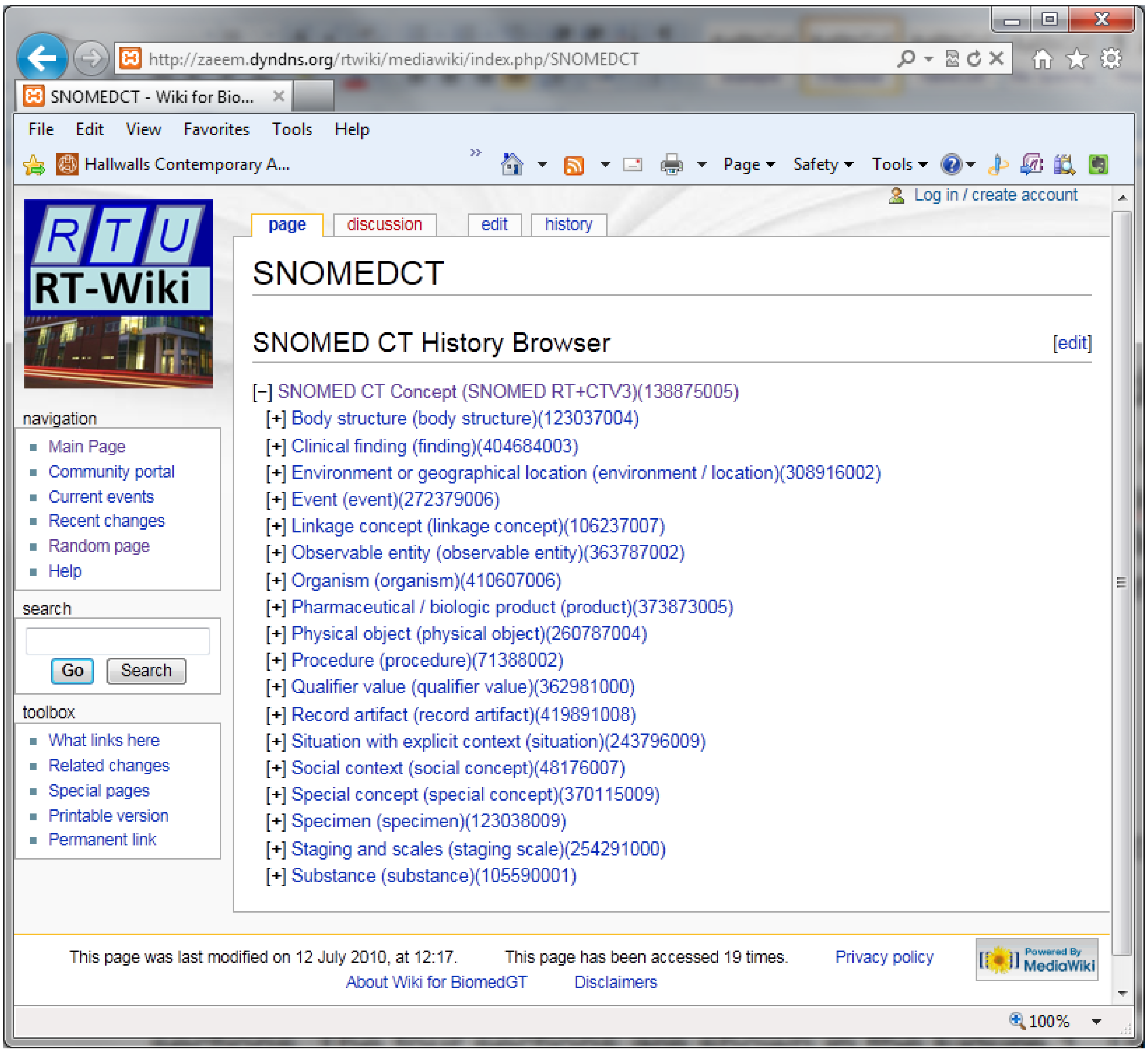

Figure 5: RT-Wiki home page

RT-Wiki contains a separate page for each SNOMED-CT concept, for example as shown in Figure 6.

[1] https://cabig-kc.nci.nih.gov/Vocab/KC/index.php/LexWiki

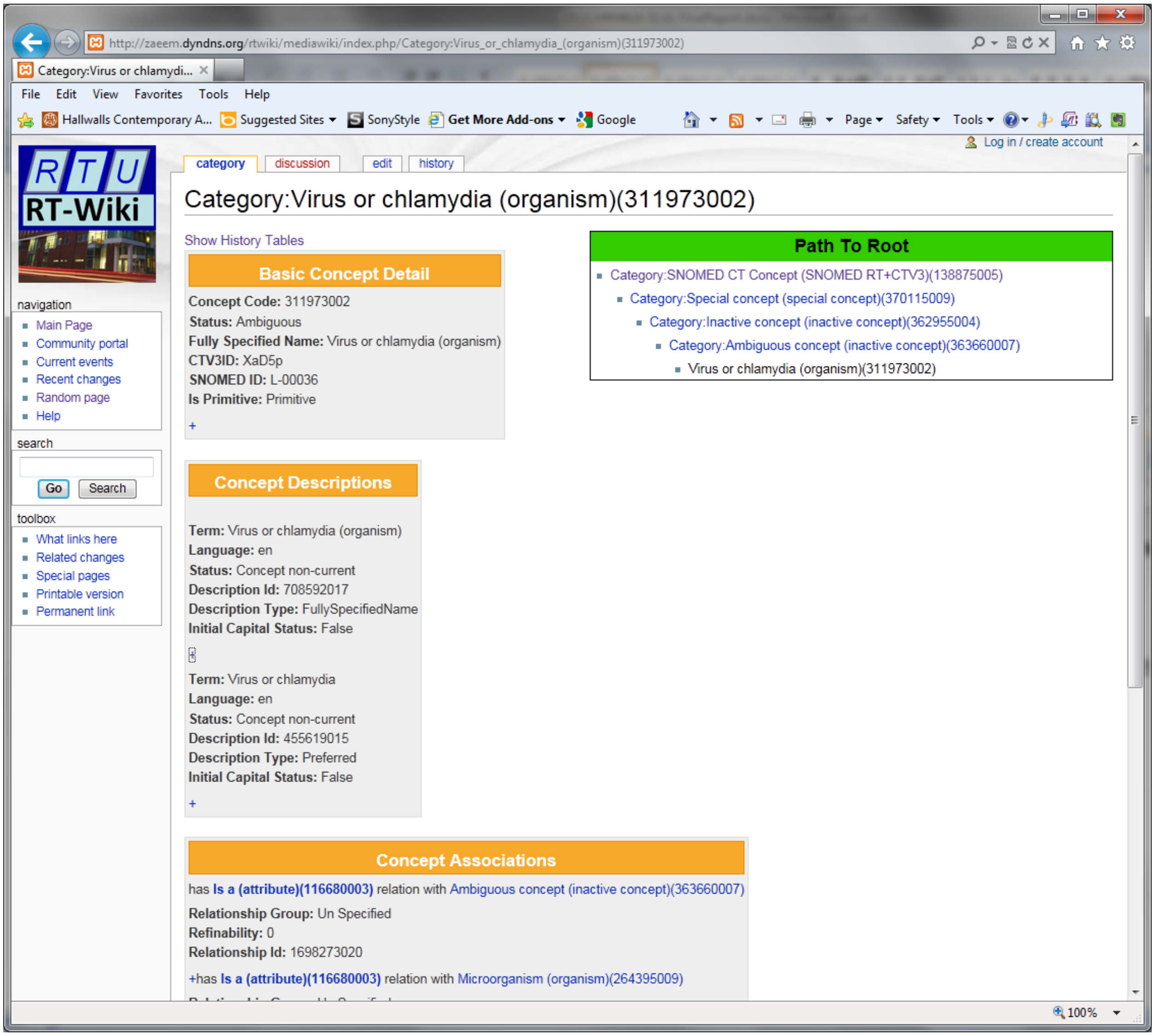

Figure 6: A Wiki page for a SNOMED-CT concept

The concept page shows for a given concept the available information in the concept table belonging to the latest release of SNOMED CT that is loaded in the system and similarly for the descriptions and relationships. Clicking the 'show history tab' displays all changes that occurred to the concept, its descriptions and its association with other concepts. Figure 7 shows the realism-based versioning interpretation of the change that occurred to SNOMED CT concept 'Virus or chlamydia (organism)' with concept ID 311973002 when it was judged as being 'ambiguous'. Clicking the 'edit' button allows the user other interpretations to be made (Figure 8).

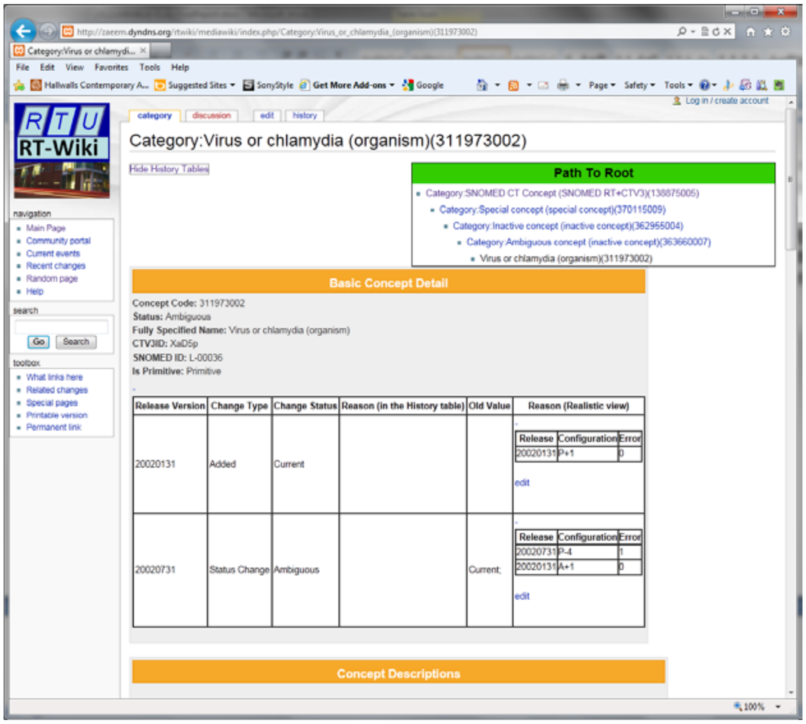

Figure 7: Change History of the concept 'Virus or chlamydia (organism)'

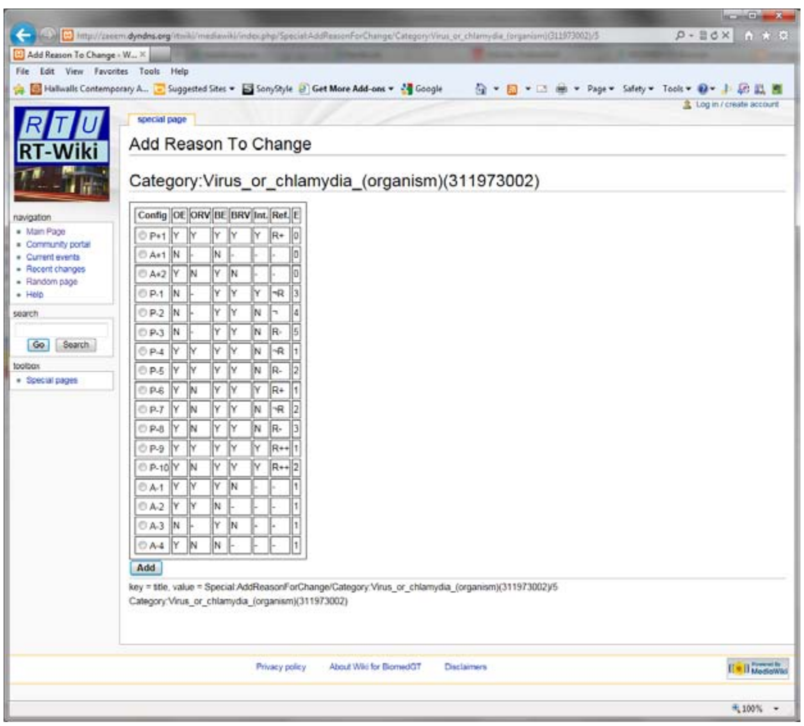

Figure 8: Interface for manually changing realism-based versioning information

8.2 Implementation

The concept wiki pages are encoded through templates[2] which offer a standard way to include content in several wiki pages). The following RT-Wiki_Concept template encodes the concept detail rendered in the "Basic Concept Detail" heading as shown in **Figure 6**.

{{RT-Wiki_Concept

|conceptid=*311973002*

|statusdesc=*Ambiguous*

|fullyspecifiedname=*Virus or chlamydia (organism)*

|ctv3id=*XaD5p*

|snomedid=*L-00036*

|isprimitive=*1*}}

Mediawiki maintains a separate page for each template in which the standard text is maintained which is included with the encoding of the template. The following text (in standard wiki syntax[3]) is the contents of the RT-Wiki_Concept template page that generates for a specific concept the code just described.

*'''Concept Code: '''{{#if:{{{conceptid}}}|{{{conceptid}}}|NOT SPECIFIED}}
*

*'''Status:''' {{#if:{{{statusdesc}}}|{{{statusdesc}}}|NOT SPECIFIED }}
*

'''Fully Specified Name:''' {{#if:{{{fullyspecifiedname}}}| {{{fullyspecifiedname}}}|NOT SPECIFIED }}

*{{#if:{{{ctv3id}}}|
 '''CTV3ID:''' {{{ctv3id}}}| }}*

*{{#if:{{{snomedid}}}|
 '''SNOMED ID:''' {{{snomedid}}}| }}*

*{{#if:{{{isprimitive}}}|
 '''Is Primitive:''' {{#ifeq:{{{isprimitive}}}|1|Primitive| {{#ifeq: {{{isprimitive}}}|0|Fully defined|}} }} }}*

In this project, we have developed templates to capture all required content as exemplified by the following instantiations of each template:

- ***{{RT-Wiki_Basic Data Header}}:*** encodes the header for the lexical section of the entry and includes preferred names, definitions, code, description and so forth, providing thus all what is rendered under the 'Lexical' heading.
- ***{{RT-Wiki_Concept Code|1=64572001|statuscode=0|statusdesc=Current***}}: renders the Concept code and status of a concept.
- ***{{RT-Wiki_Preferred Name|1=SMD0 Disease (disorder)}}:*** renders the preferred name of a term.
- ***{{RT-Wiki_Component History Header }}:*** used to display the change history of the components (concepts, descriptions and relationships) in SNOMED CT. It marks the

[2] http://www.mediawiki.org/wiki/Help:Templates

[3] http://www.mediawiki.org/wiki/Help:Editing

start of this history of a particular component. During rendering it produces history table field names.

- ***{{RT-Wiki_Concept History Data Record Header}}:*** marks the start of a new row in a history table and inserts an empty row.
- ***{{RT-Wiki_Hst Release Version|1=19940101}}:*** renders the release version from the component history table.
- ***{{RT-Wiki_Hst Component ID|ID=64572001}}:*** renders component id.
- ***{{RT-Wiki_Hst Change Type|code=0|desc=Added}}:*** renders the CHANGETYPE field of the component history table.
- ***{{RT-Wiki_Hst Status|code=0|desc=Current}}:*** renders the STATUS field of the component history table.
- ***{{RT-Wiki_Hst Reason|reason=}}:*** renders the REASON field of the component history table.
- ***{{RT-Wiki_Realistic Reason Header}}:*** marks the start of the comments to be entered by authors who want to write more clearly why a change in the corresponding term/relation of the history table row was effected,
- ***{{RT-Wiki_Realistic Reason Trailer|ID=33}}:*** marks the end of the corresponding header template. The ID parameter is used by programming scripts.
- ***{{RT-Wiki_Concept History Data Record Trailer***}}: marks the end of the row in a history table.
- ***{{RT-Wiki_Component History Trailer}}:*** marks the end of the history table.
- ***{{RT-Wiki_Synonym|1=Disease (disorder)|lang=en|ID=803980011|statuscode=0| statusdesc= Current}}:*** *renders* display names from *sct_descriptions* table.
- ***{{RT-Wiki_Basic Data Trailer}}:*** marks the end of the corresponding header template.
- ***{{RT-Wiki_Association Header}}:*** marks the start of the terms relationships.
- ***{{RT-Wiki_Association|1=SMD0 Course (attribute)(260908002)|2=SMD0 Courses (qualifier value)(288524001)|type=Association|qual=some|view=defined|defining=true}}:*** renders a relationship between terms.
- ***{{RT-Wiki_Association Trailer}}:*** *m*arks the end of its corresponding header table.

Transcription of the SNOMED history data into the RT-Wiki was achieved using the 5-step algorithm depicted in Figure 9.

1. Data Import native SNOMED CT: this process takes the SNOMED CT core distribution files (tab separated text files) taken from different versions of SNOMED CT as input, and imports them into a relational database.
2. Find Changes: this process constructs the concept change history according to the realism-based versioning principles. The input of this process is the SNOMED relational database created through the data import process.
3. Import DBase into MySQL: this step imports the “Changes DBase” database produced through “Find Changes” process into the MySQL database.
4. Wiki Pages Generator: this process generates a concept-page (a wiki page) for each SNOMED CT concept. It generates templates for all sections of a concept-page.
5. Upload Pages: before running this process MediaWiki is to be installed and running on an apache server (an http web server). This process is an http client, which uploads the concept-pages to the Mediawiki that stores these pages in the Mediawiki MySQL database.

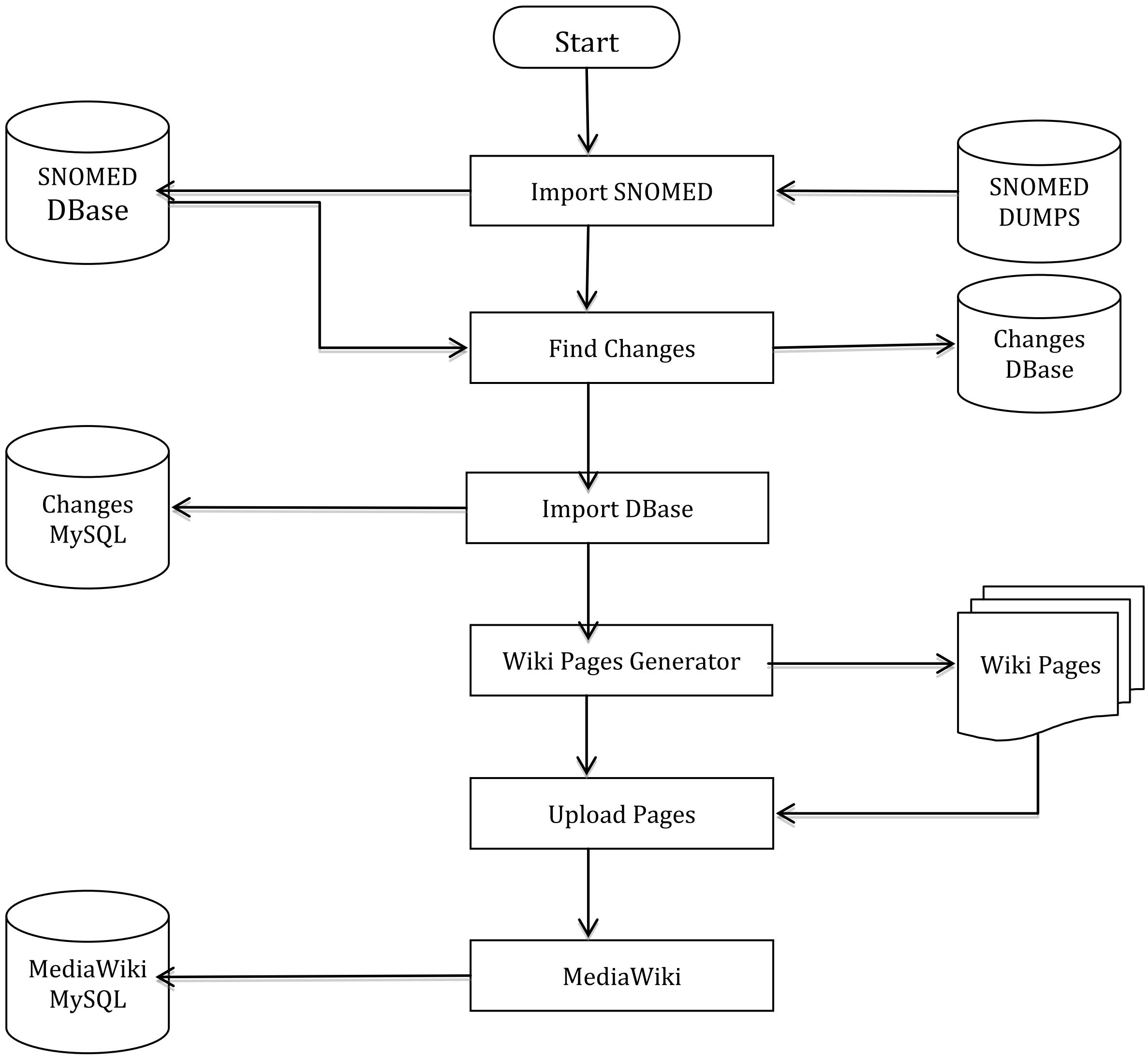

Figure 9: Data Flow Diagram of Wiki page generation process

9 RT-based Ontology/Terminology (O/T) History Tracker (HT): Functional and Technical Specifications

9.1 Server-side HT activation

When installing an HT-tracker on a server, this specific tracker, as well as each service described further for this tracker has to be assigned an Instance Unique Identifier (IUI). Create all referent tracking tuples as required in RT (A-tuples, D-tuples, …). The IUIs together with the names for the services must be made public for use by client-side applications that want to interact with this specific HT-tracker. The names are shown (*italicized and between parentheses*) in the headings below which describe services. Two distinct HT instances will use the same names for the same services, but of course distinct IUIs.

The name of the service will be used to identify the requested service when the client-side requests a service. This input will not be repeated further in the service descriptions.

9.2 General behavior for responding to a service request

We here describe the intended behavior of a specific HT-tracker that has been installed and made his services public. We do not describe here a service that tracks HT-trackers.

Each service request is identified by the name for the service as is made public to client applications.

Each request, except the one for registering an HT-user, must contain the IUI for the O/T-side user identifier which is generated and returned when an HT-user registration service is requested.

HT-side activities ***for each service request*** (exceptions apply as described) in addition to the specific activities described further for each service:

- o Check, except for the service for registering an HT-user, whether a supplied user-id IUI (not the O/T-side user-id) is provided. If not, return a "requesting user IUI missing" error code;
- o Check, except for the services for registering an O/T, an O/T-version, or the services which a user is allowed to request for an O/T, whether a supplied IUI for an O/T-version is provided. If not, return a "O/T-version IUI missing" error code;
- o Check, except for the service for registering an HT-user, by means of the provided user-id IUI:
  - o whether such user is registered. If not, return a "user unknown" error code;
  - o if such user is registered, check in the <u>*permitted services table*</u> whether the user is allowed to request this particular service. When that is not the case, return a "user not authorized" error code. There can be more than one record in this table indicating the permission for a user to activate a service, but only the presence of one such record needs to be checked.

Check whether the other inputs for the specific service are provided. If not, return the applicable error-code. (The inputs required for each specific service as well as corresponding error-codes are listed in the description of the service below)

Generate an IUI for the service-activation and store the following information in an RT-compatible <u>*service-activation table*</u>:

- o the IUI for the requested service
- o the IUI for the service-activation
- o the IUI for the O/T-side user identifier that made the service request
- o date/time of the request

- o code indicating success or error-code(s) as just specified or described further in the specific services
- o the http-header information that was supplied by the client application to activate the service

When no errors, return the IUI for the service-activation plus the requested information as described in each specific service.

NOTE: these actions are not repeated in the service descriptions provided further but need to be done.

9.3 Service to obtain info about a service-activation

(*GetServiceActivationInfo*)

Input:

- o IUI associated with the requester
- o IUI of the service-activation

HT-side activities in addition to applicable general behavior:

- o check whether an IUI for the service activation is provided. If not, return and register a "service-activation IUI missing" error-code,
- o find in the ***service-activation table*** the entries containing this service-activation IUI. If the service-activation IUI is not found in that table, return and register a "incorrect service-activation IUI" error-code,
- o if found, return to the requester the following information:
  - o the IUI for the requested service
  - o the name of the requested service
  - o the IUI for the O/T-side user identifier that made the service request
  - o date/time of the request for which info was requested (thus not of this request)
  - o code indicating success or error-code(s) as just specified or described further in the specific services
  - o the IUI of this service-activation

9.4 Service for registering an HT-user

(RegisterUser)

General principle: each user registers himself to the HT

Input: O/T-side user identifier

HT-side activities in addition to applicable general behavior:

- o Check whether an O/T-side user identifier is provided, and if not, return and register a "O/T-side user-id missing" error code;
- o Check whether an identical O/T-side identifier already exists. If so, return a "user-id already registered" error code;
- o If no errors, generate an IUI for the O/T-side user identifier (this IUI does indeed identify the O/T-side user identifier and not the O/T-side user because the client-side might use multiple identifiers for the same user and the HT has no way to check this);
- o Add an entry to the ***user identifier registration table*** with the following information:
  - o The O/T-side identifier
  - o The IUI for the O/T-side user identifier
  - o IUI of this service-activation
- o Return to the service requester:

    - the IUI of the user just registered
    - the IUI of this service-activation

9.5 Service for registering an O/T

(RegisterOT)

General principle: each user can register O/Ts by providing a distinct O/T-side identifier for each O/T. O/T-side O/T identifiers need to be globally unique within a specific HT tracker.

Input:

- the IUI related to the O/T-side user identifier
- an optional O/T-side identifier for the O/T

HT-side activities in addition to applicable general behavior:

- If an O/T-side identifier for the O/T is submitted, check whether it has already been registered. If so, return and register a "O/T identifier already registered" error code;
- Generate an IUI for the newly registered O/T;
- Add an entry to the ***O/T registration table*** with the following information:
    - The O/T-side identifier (empty if none given)
    - The IUI for the O/T
    - IUI of the service-activation
- Return to the service requester:
    - the IUI of the O/T
    - the IUI of the service activation

9.6 Service for registering the services a user is allowed to access for a specific O/T

(SetPermissions)

General principle: the one who registered an O/T is allowed to set the permissions for any user he wishes, and for any service, including the permission to grant permissions.

Input:

- the IUI of the O/T-side user that registered the O/T (this functions here too as the requesting user-ID IUI), called the *granter*;
- the IUI of the O/T
- the IUI of the O/T-side user for whom permitted services are to be registered (may be the same as the *granter*), called the *grantee*;
- a list of the names of the permitted services

HT-side activities in addition to applicable general behavior:

- check whether an IUI for a O/T is supplied. If not return and register a "O/T-IUI missing" error code;
- check whether the supplied O/T IUI is indeed the IUI of a registered O/T. If not, return and register a "O/T not registered" error code;
- check whether the granter IUI corresponds to the IUI for the user that registered the O/T. If not, return and register a "service not permitted for user" error-code;
- If no service names are provided, return and register a "service list missing" error code;
- check for each service name whether it exists on the HT. If not, return and register an "unknown service" error-code;
- if no errors, add for each service an entry to the ***permitted services table*** with the following information (except for those services for which permission was already

granted earlier by the same granter: simply ignore them and add only the new service permissions):
  - o IUI related to the granter,
  - o IUI of the O/T,
  - o IUI of the grantee,
  - o name of the allowed service
  - o IUI of this service activation

Note: users may be given permissions more than once and by different granters. Each time, an additional record is added to the table.

9.7 Service for retracting services a user is allowed to access for a specific O/T (*RetractPermissions*)

General principle: a user can only retract permissions for which he was the granter. If he gave granter rights to other users, he can retract such right, but not the permissions that have been granted by that granter before the retraction. If that is undesirable, the user who registered an O/T should not grant permission to other users to grant permissions.

Input:

- the IUI of the *granter* (this functions here too as the requesting user-ID IUI),
- the IUI of the O/T,
- the IUI of the *grantee* for whom permitted services are to be retracted (may be the same as the granter IUI),
- a list of the names of the services for which permissions have to be retracted.

HT-side activities in addition to applicable general behavior:

  - o check whether an IUI for a O/T is supplied. If not return and register a "O/T-IUI missing" error code;
  - o check whether the supplied O/T IUI is indeed the IUI of a registered O/T. If not, return and register a "O/T not registered" error code;
  - o If no service names are provided, return and register a "service list missing" error code;
  - o check for each name in the services list whether it exists on the HT. If not, return and register an "unknown service" error-code;
  - o check for each service in the list whether the ***permitted services table*** contains an entry for this granter, grantee, O/T and service.
  - o If not, return and register a "no permissions found" error code.
  - o If there are permissions, add for each service and for each record about permissions given by the granter to the grantee (a granter might have given permissions at different times) an entry to the ***retracted permissions table*** with the following information:
    - ▪ IUI related to the granter,
    - ▪ IUI of the O/T,
    - ▪ IUI of the grantee,
    - ▪ name of the service for which permission is retracted
    - ▪ IUI of ***this*** service activation
    - ▪ IUI of the service activation during which the permission was set (thus the service activation IUI registered in the ***retracted permissions table*** corresponding to this retraction).

  Then, delete the corresponding entry from the ***permitted services table***.

9.8 Services for requesting the IUI of an already registered O/T

9.8.1 By providing the IUI of the service-activation that created the registration of the O/T

(*GetIUIofOTByServiceActivation*)

Input:

- o IUI related to the requester
- o IUI of the service-activation that created the registration of the O/T

HT-side activities in addition to applicable general behavior:

- o check whether an IUI for the service activation is provided. If not, return and register a "service-activation IUI missing" error-code,
- o find in the <u>*O/T registration table*</u> the IUI for the O/T using the service-activation IUI. If the service-activation IUI is not found in that table, return and register a "incorrect service-activation IUI" error-code,
- o If service-activation IUI is found, return to the service requester
  - o the IUI of the O/T
  - o when available, the O/T-side identifier for the O/T
  - o IUI of this service activation.

9.8.2 By providing the O/T-side identifier for the O/T

(*GetIUIofOTByLocalIdentifier*)

Input:

- o IUI related to the requester
- o O/T-side identifier for the O/T

HT-side activities in addition to applicable general behavior:

- o check whether an O/T-side identifier for the O/T is provided. If not, return and register an "O/T-side O/T identifier missing" error code;
- o find in the <u>*O/T registration table*</u> the IUI for the O/T using the O/T-side identifier for the O/T. If the O/T-identifier is not in that table, return and register a "O/T not registered" error code;
- o If service-activation IUI is found, return to the service requester
  - o the IUI of the O/T
  - o IUI of this service activation.

9.9 Service for registering an O/T-version

(*RegisterOTversion*)

Input:

- o IUI related to the requester
- o the IUI for the ontology to which the version to be registered applies
- o an optional O/T-side identifier for the O/T-version
- o the date/time of the official release of this version (this date/time may be different from the date/time of this service request)

HT-side activities in addition to applicable general behavior:

- o check whether an IUI for a O/T is supplied. If not return and register a "O/T-IUI missing" error code;

- check in the *O/T registration table* whether the supplied O/T IUI is indeed the IUI of a registered O/T. If not, return and register a "O/T not registered" error code;
- check whether a date/time is supplied. If not return and register a "Date/time missing" error code;
- Generate an IUI for the newly registered O/T-version
- Add an entry to the *O/T-version registration table* with the following information:
    - The O/T-side version identifier (empty if none given)
    - The IUI for the O/T-version
    - The IUI for the O/T of which this is a version
    - the date/time of the official release of this version
    - IUI of this service activation
- Return to the service requester:
    - the IUI for the O/T-version
    - IUI of this service activation.

9.10 Services for requesting the already registered O/T-versions of an O/T

9.10.1 By providing the IUI of the O/T

(*GetOTversionsByIUI*)

Input:

- IUI related to the requester
- IUI of the O/T

HT-side activities in addition to applicable general behavior:

- check whether an IUI for the O/T is provided. If not, return and register a "O/T IUI missing" error code,
- check in the *O/T registration table* whether this IUI is indeed for a registered O/T. If not, return and register a "O/T not registered" error code.
- find in the *O/T-version registration table* all entries for this O/T and return to the requester a list of 4-tuples, one for each entry, containing:
    - The O/T-side version identifier (empty if none given)
    - The IUI for the O/T of which this is a version
    - the date/time of the official release of this version
    - IUI of the service-activation

9.10.2 By providing the O/T-side identifier for the O/T

(*GetOTversionsByLocalIdentifier*)

Input:

IUI related to the requester
O/T-side identifier for the O/T

HT-side activities in addition to applicable general behavior:

- check whether an O/T-side identifier for the O/T is provided. If not, return and register an "O/T-side O/T identifier missing" error code;
- find in the *O/T registration table* the IUI for the O/T using the O/T-side identifier for the O/T. If the O/T-identifier is not in that table, return and register a "O/T not registered" error code;
- find using the IUI for the O/T in the *O/T-version registration table* all entries for this O/T and return to the requester a list of 4-tuples, one for each entry, containing:

- The O/T-side version identifier (empty if none given)
- The IUI for the O/T of which this is a version
- the date/time of the official release of this version
- IUI of the service-activation

9.11 Service for registering the referencing of an intended extension to an O/T (*RegisterConcept*)

General principles:

The HT distinguishes explicitly between intended extensions and references. Distinct references may have the same intended extension. A reference cannot have distinct intended extensions. References can be terms or identifiers.
The purpose of this service is in the first place to register an intended extension. It can optionally be used to assign at the same time an identifier what we will call a "*local concept-identifier*". "Optional" here means that no such local concept-identifier need to be submitted, but not that there is no such identifier on the client (O/T) -side. It is also not assumed that what the requester submits as a local concept-identifier is indeed the local identifier used on the client side. If that is not the case, it is the responsibility of the client side to keep track of the relation between real client side identifiers and submitted identifiers. One reason for keeping these identifiers distinct might be not to disclose the identity of the O/T being tracked by means of a publicly accessible HT.
This service should be accessed prior to registering a local concept-identifier which references the intended extension in a specific O/T-version. The HT will check whether this is the case and if not, return an error.

Input:

- IUI of the service requester
- an optional local concept-identifier (a local concept identifier is only required when concept-identifiers need to be registered to O/T-versions)
- the date/time the local concept-identifier was created on the O/T-side (this date/time may be different from the date/time of this service request)
- the IUI of the O/T to which the local concept-identifier is added,
- a formatted "existence string" reflecting the justified belief of the service requester whether the O/T authors believed at the time of the addition whether the intended extension of the local concept-identifier did exist prior to the addition and whether it exists (or existed) at the time of the addition. This string must be one of these four: "YY", "YN", "NY", "NN". Note that distinct requesters may have distinct beliefs about the existence of the intended extension.
- a formatted "relevancy string" reflecting the justified belief of the service requester whether the O/T authors believed at the time of the addition whether reference to the intended extension of the local concept-identifier by means of this local concept-identifier would have been relevant prior to the addition and similarly at the time of the addition. Note that distinct requesters may have distinct beliefs about the relevancy of referencing the intended extension. Possible forms of the string, in relation to the string previously described are:

  YY → YY, YN, NY, NN
  YN → Y-, N-
  NY → -Y, -N
  NN → --

HT-side activities in addition to applicable general behavior:

- check whether an IUI for the O/T is provided. If not, return and register a "O/T IUI missing" error code,
- check in the *O/T registration table* whether this IUI is indeed for a registered O/T. If not, return and register a "O/T not registered" error code.
- check whether a date/time is supplied. If not return and register a "Date/time missing" error code;
- check whether the existence string has one of the four allowed forms; if not return and register an "ill-formed existence string" error code;
- if a local concept-identifier is provided, check in the *concept registration table* (see below) whether this local concept-identifier has not already been registered for this O/T (it is allowed however that the same local concept-identifier is registered to a different O/T). If it was already registered, return and register an " local concept-identifier already registered" error code.
- check whether the relevancy string has one of the nine allowed forms; if not return and register an "ill-formed relevancy string" error code;
- check whether the provided relevancy string is allowed for the provided existence string; if not, return and register an "incompatible relevancy string" error code;
- if no errors, generate an IUI representing the intended extension of the local concept-identifier and:
    - return to the requester:
        - the IUI related to this local concept-identifier, this IUI thus representing the intended extension of the local concept-identifier,
        - the IUI of this service activation
    - and store the following information in the *concept registration table*:
        - the IUI related to this local concept-identifier, this IUI thus representing the intended extension of the local concept-identifier,
        - the O/T-side local concept-identifier (empty if none given)
        - the date/time the local concept-identifier was created on the O/T-side
        - the IUI of the O/T to which the local concept-identifier is added
        - the existence string (note that this string can become long after several updates, do not set a maximum length)
        - a formatted "existence error string", motivating the beliefs expressed in the existence string. This string must be of a specific form, depending on the existence string:
            - if the existence string is YY → NN,
            - if the existence string is YN → NR,
            - if the existence string is NY → NR,
            - if the existence string is NN → NN
        - the relevancy string (note that this string can become long after several updates, do not set a maximum length)
        - a formatted "relevancy error string", motivating the beliefs expressed in the relevancy string. This string must be of a specific form, depending on the relevancy string:
            - if the relevancy string is YY → NN,
            - if the relevancy string is YN → NR,
            - if the relevancy string is NY → NR,
            - if the relevancy string is NN → NN.
            - if the relevancy string is N- → NR.
            - if the relevancy string is Y- → NR.
            - if the relevancy string is -Y → NR.

    - if the relevancy string is -N → NR.
    - if the relevancy string is -- → NN
  - a formatted "coding error string" which is "NN"
  - the IUI of the service requester (this is redundant as it can be found through the service activation IUI, but it makes things easier for updating the table)
  - the IUI of the service activation

9.12 Service for registering changes in relation to an intended extension referenced in an O/T

(*UpdateConcept*)

General principle: this service should be accessed only when the intended extension has already been registered to the O/T (with or without the registration of a local concept-identifier). The HT will check whether this is the case and if not, return an error.

Input:

- IUI of the service requester
- IUI of the intended extension as assigned by the ***RegisterConcept*** service
- the date/time for which the update is applicable (this is *not* the date/time this service is called)
- the IUI of the O/T in relation to which the intended extension is updated
- an optional local concept-identifier used to reference the intended extension in the O/T,
- a formatted "existence string" reflecting changes in the justified belief of the service requester about whether the O/T authors changed their beliefs about the existence or non-existence of the intended extension during the various time periods whose boundaries are the date/times at which registrations to the O/T or to the O/T's versions, as well as updates are made concerning this intended extension.
- a formatted "relevancy string", reflecting the justified belief of the service requester about the O/T authors' beliefs about the relevancy or non-relevancy to reference the intended extension during the various time periods whose boundaries are the date/times at which registrations to the O/T or to the O/T's versions, as well as updates are made concerning this intended extension.
- a formatted "existence error string", reflecting the sorts of errors that are believed to have been committed - if any at all - that required the existence string to be updated.
- a formatted "relevancy error string", reflecting the sorts of errors that are believed to have been committed - if any at all - that required the relevancy string to be updated.
- a formatted "coding error string", reflecting the sorts of errors that are believed to have been committed in the existence string - if any at all - that required the existence string to be updated.

HT-side activities in addition to applicable general behavior:

- check whether an IUI for the intended extension is provided. If not, return and register a "intended extension IUI missing" error code,
- check in the *concept registration table* whether this IUI is indeed for a registered intended extension. If not, return and register a "intended extension not registered" error code.
- if a local concept-identifier is submitted, check in the *concept registration table* whether this local concept-identifier has already been registered. If so, the IUI for the intended

extension specified in that entry should be identical to the IUI submitted for the intended extension in this service request. If that is not the case, return and register a "local concept-identifier already registered with different intended extension" error code;

- check whether a date/time is supplied. If not return and register a "Date/time missing" error code;
- check in the *concept registration table* whether the date/time supplied is later than the latest date/time stored for this intended extension *by the service requester*. If that is not the case, return and register a "incorrect historical sequence: later date already provided" error code,
- check whether an IUI for the O/T is provided. If not, return and register a "O/T IUI missing" error code,
- check in the *O/T registration table* whether this IUI is indeed for a registered O/T. If not, return and register a "O/T not registered" error code.
- find in the *concept registration table* the existence string and relevancy string registered for the latest date for this intended extension *by the service requester* (thus the date prior to the date/time submitted in this request),
- check whether the existence string has the required length: it must be one character more than the length of the existence string registered for the most prior date *by the service requester*. If the string is too short, return and register an "ill-formed existence string: too short" error code; if too long, return and register an "ill-formed existence string: too long" error code;
- check whether the existence string is of the right form:
  - it may only contain "Y" and "N" characters, otherwise, return and register an "ill-formed existence string: invalid characters" error code;
  - if there are "Y"-characters, they must all appear in one contiguous block (thus allowed are "YYYYNNN", "NNNNYYY" and "NNYYYN", but not "YYNNYY", "NYYNYNN", etc.; otherwise, return and register an "ill-formed existence string: impossible existence history" error code
- check whether the relevancy string has the same length as the existence string. If not, return and register an "ill-formed relevancy string: too long" or "ill-formed relevancy string: too short" error code, whatever is the case;
- check whether the provided relevancy string is allowed for the provided existence string: a compatible relevancy string is one in which each "Y" in the existence string is substituted by either "N" or "Y" in the relevancy string and each "N" by a "-"; if not, return and register an "incompatible relevancy string" error code;
- check whether an existence error string is provided, if not, return an "existence error string missing" error code;
- check whether the existence error string has the required length: it must be of the same length as the length of the existence string. If the string is too short, return and register an "ill-formed existence error string: too short" error code; if too long, return and register an "ill-formed existence error string: too long" error code;
- check whether the existence error string is of the right form: it may only contain the characters "R", "U", "C", "N" for respectively change in reality, change in understanding, correction of internal error and no error committed; if not, return and register an "incompatible existence error string" error code;
- check whether a relevancy error string is provided, if not, return an "relevancy error string missing" error code;
- check whether the relevancy error string has the required length: it must be of the same length as the length of the relevancy string. If the string is too short, return and register an "ill-formed relevancy error string: too short" error code; if too long, return and register an "ill-formed relevancy error string: too long" error code;

- o check whether the relevancy error string is of the right form: it may only contain the characters "R", "U", "C", "N" for respectively change in reality, change in understanding, correction of internal error and no error committed; if not, return and register an "incompatible relevancy error string" error code;
- o check whether a coding error string is provided, if not, return an "coding error string missing" error code;
- o check whether the coding error string has the required length: it must be of the same length as the length of the existence string. If the string is too short, return and register an "ill-formed coding error string: too short" error code; if too long, return and register an "ill-formed coding error string: too long" error code;
- o check whether the coding error string is of the right form and if not, return and register an "incompatible coding error string" error code; it may only contain the characters "N", "W", "M", "D" for respectively:
    - o N(one): intended encoding,
    - o W(rong): the local identifier was not supposed to denote this intended extension,
    - o M(issing): the local identifier does not denote any intended extension
    - o D(ouble): the intended extension is identical to another intended extension

if no errors:

- o return to the requester:
    - the IUI of this service activation
- o and store the following information in the ***concept registration table***:
    - the IUI of the intended extension as provided in the service request,
    - the date/time provided in the service request (this date/time is *not* and may be different from the date/time of this service request),
    - the IUI of the O/T in relation to which the concept-identifier is updated,
    - if provided, the local concept identifier for the intended extension (note that it is allowed by means of distinct requests to this service to provide distinct local identifiers for the same IUI. This is because the IUI represents the intended extension of the local concept-identifier and distinct local concept identifiers might have the same intended extension. It is, in contrast, not allowed for the same local O/T-side concept-identifiers to be associated with more than one intended extension and thus more than one intended extension IUI. The latter is prevented by the constraints implemented for the involved services.)
    - the submitted existence string
    - the submitted relevancy string
    - the submitted existence error string
    - the submitted relevancy error string
    - the submitted coding error string
    - the IUI of the service requester
    - the IUI of the service activation

9.13 Service for registering the addition of a local concept-identifier to an O/T-version

(*RegisterConceptToVersion*)

General principles:

- o this service should be accessed only after the local concept-identifier has been registered to an O/T.

- registering to a specific version requires also that this local concept-identifier has never been registered to a later O/T version.
- registering can be done only once,
- finally, one can only register to an O/T-version if the release date of that version is later than the date of the latest update for the intended extension of the local concept-identifier.
- the HT will check whether all this is the case and if not, return an error.

Input:

- IUI of the service requester
- the local concept-identifier to be registered in the O/T-version
- the IUI of the intended extension related to the local concept-identifier as assigned by the ***RegisterConcept*** service (note that this IUI could be retrieved also on the basis of the local concept-identifier alone; we insist however that it be submitted to prevent mistakes on the side of the requester who might believe that the local concept-identifier is related to a distinct intended extension)
- the date/time the local concept-identifier was registered to the O/T-version
- the IUI of the O/T-version to which the local concept-identifier is registered
- an optional formatted "existence string" reflecting changes in the justified belief of the service requester whether the O/T authors changed their beliefs about the existence or non-existence of the intended extension of the concept during the various time periods whose boundaries are the date/times at which registrations are made concerning this concept. This input may be omitted if the requester believes that no existence changes occurred.
- an optional (in case no existence string has been provided and no belief revision concerning the relevancy has happened) or mandatory (in all other cases) formatted "relevancy string", reflecting the justified belief of the service requester about the O/T authors' beliefs about the relevancy or non-relevancy of the concept during the various time periods whose boundaries are the date/times at which registrations are made concerning this concept.
- an optional formatted "existence error string", reflecting the sorts of errors that are believed to have been committed - if any at all - that required the existence string to be updated.
- an optional formatted "relevancy error string", reflecting the sorts of errors that are believed to have been committed - if any at all - that required the relevancy string to be updated.
- an optional formatted "coding error string", reflecting the sorts of errors that are believed to have been committed in the existence string - if any at all - that required the existence string to be updated.

HT-side activities in addition to applicable general behavior:[4]

- check whether a local concept-identifier is provided. If not, return and register a "Local concept-identifier missing" error code,
- check in the *concept registration table* whether the local concept-identifier has already been registered. If not, return and register a "local concept-identifier not registered" error code.
- check whether an IUI for the intended extension is provided. If not, return and register a "intended extension IUI missing" error code,

[4] More error checks are possible concerning the compatibility between error strings, but these checks are not described thus far

- check in the *concept registration table* whether this IUI is indeed for a registered intended extension. If not, return and register a "intended extension not registered" error code.
- check in the *concept registration table* whether the submitted IUI for the intended extension corresponds indeed with the local concept-identifier. If not, return and register a "local concept-identifier not registered for this intended extension" error code.
- check in the *O/T-version content table* (see below) whether the local concept-identifier has already been registered to this O/T-version; if that is the case, return and register a "local concept-identifier already registered to this O/T-version" error code;
- check whether a date/time is supplied. If not return and register a "Date/time missing" error code;
- compare the date/time supplied with the most recent date/time stored for this intended extension (thus not just only for the local concept-identifier) in the *concept registration table*. If the supplied date/time is earlier than the most recent date/time, return and register a "incorrect historical sequence: later date already provided " error code,
- check whether an IUI for the O/T-version is provided. If not, return and register a "O/T IUI missing" error code,
- check in the *O/T-version registration table* whether this IUI is indeed for a registered O/T-version. If not, return and register an "O/T-version not registered" error code.
- check in the *O/T-version registration table* whether the date/time supplied to this service is equal or earlier than the date/time stored for the official release of the version supplied (for future reference, this date is called the "*release date/time*"). If that is not the case, return and register a "incorrect historical sequence: version is already released" error code,
- find in the *concept registration table* the existence string, the relevancy string, the existence error string, and the relevancy error string registered for the latest date for this intended extension (thus the date prior to the date/time submitted in this request); for future reference, we will call these strings the "*latest existence string*" , "*latest relevancy string*", "*latest existence error string*" and "*latest relevancy error string*" respectively;
- check for the following possibilities:
    - if a relevancy string has been provided but no existence string, return and register a "existence string missing" error code,
    - if a relevancy string has been provided but no relevancy error string, return and register a "relevancy error string missing" error code,
    - if an existence string has been provided but no relevancy string, return and register a "relevancy string missing" error code,
    - if an existence string has been provided but no existence error string, return and register a "existence error string missing" error code,
    - if an existence string has been provided but no coding error string, return and register a "coding error string missing" error code,
    - if both existence string and relevancy string are *not* supplied to this service:
        - *create* a *new* existence string by copying the latest existence string and adding the *last* character of the latest existence string *twice* to the end of the new existence string. Thus if the latest existence string was "NNYN", the new existence string becomes "NNYNNN";
        - *create* a *new* relevancy string by copying the latest relevancy string and adding the *last* character of the latest relevancy string *twice* to the end of the new relevancy string. Thus if the latest relevancy string was "--Y-", the new relevancy string becomes "--Y---";
        - *create* a *new* existence error string by copying the latest existence error string and adding the characters "NN" to the end of it;

    - *create* a *new* relevancy error string by copying the latest relevancy error string and adding the characters "NN" to the end of it;
    - *create* a *new* coding error string by copying the latest coding error string and adding the characters "NN" to the end of it;
  - if both existence string and relevancy string are supplied to this service:
    - check whether the existence string has the required length: it must be *one* character more than the length of the existence string registered for the most prior date. If the string is too short, return and register an "ill-formed existence string: too short" error code; if too long, return and register an "ill-formed existence string: too long" error code;
    - check whether the existence string is of the right form:
      - it may only contain "Y" and "N" characters, otherwise, return and register an "ill-formed existence string: invalid characters" error code;
      - if there are "Y"-characters, they must all appear in one contiguous block (thus allowed are "YYYYNNN", "NNNNYYY" and "NNYYYN", but not "YYNNYY", "NYYNYNN", etc.; otherwise, return and register an "ill-formed existence string: impossible existence history" error code
    - check whether the relevancy string has the same length as the existence string. If not, return and register an "ill-formed relevancy string: too long" or "ill-formed relevancy string: too short" error code, whatever is the case;
    - check whether the provided relevancy string is allowed for the provided existence string: a compatible relevancy string is one in which each "Y" in the existence string is substituted by either "N" or "Y" in the relevancy string and each "N" by a "-"; if not, return and register an "incompatible relevancy string" error code;
    - check whether the submitted existence error string has the required length: it must be of the same length as the length of the existence string. If the string is too short, return and register an "ill-formed existence error string: too short" error code; if too long, return and register an "ill-formed existence error string: too long" error code;
    - check whether the submitted existence error string is of the right form: it may only contain the characters "R", "U", "C", "N" for respectively change in reality, change in understanding, correction of internal error and no error committed; if not, return and register an "incompatible existence error string" error code;
    - check whether the submitted coding error string has the required length: it must have the same length as the existence string. If the string is too short, return and register an "ill-formed coding error string: too short" error code; if too long, return and register an "ill-formed coding error string: too long" error code;
    - check whether the submitted coding error string is of the right form: it may only contain the characters "N", "W", "M", "D"; if not, return and register an "incompatible coding error string" error code;
    - check whether the submitted relevancy error string has the required length: it must be of the same length as the length of the relevancy string. If the string is too short, return and register an "ill-formed relevancy error string: too short" error code; if too long, return and register an "ill-formed relevancy error string: too long" error code;

    - check whether the submitted relevancy error string is of the right form: it may only contain the characters "R", "U", "C", "N" for respectively change in reality, change in understanding, correction of internal error and no error committed; if not, return and register an "incompatible relevancy error string" error code;
    - if no errors found in relevancy, existence and corresponding error strings:
        - *create* a *new* existence string by copying the submitted existence string and adding the *last* character of this string *once* to the end of it. Thus if the submitted existence string was "NNYN", the new existence string becomes "NNYNN";
        - *create* a *new* relevancy string by copying the submitted relevancy string and adding the *last* character of this string *once* to the end of it. Thus if the submitted relevancy string was "--Y-", the new relevancy string becomes "--Y--";
        - *create* a *new* existence error string by copying the submitted existence error string and adding the *last* character of this string *once* to the end of it. Thus if the submitted existence error string was "RRUC", the new existence error string becomes " RRUCC";
        - *create* a *new* coding error string by copying the submitted coding error string and adding the *last* character of this string *once* to the end of it. Thus if the submitted existence error string was "NNND", the new existence error string becomes "NNNDD";
        - *create* a *new* relevancy error string by copying the submitted relevancy error string and adding the *last* character of this string *once* to the end of it. Thus if the submitted relevancy error string was "RRUC", the new relevancy error string becomes " RRUCC";

if no errors in all of the above:

- return to the requester:
    - the IUI of this service activation
- store the following information in the *concept registration table*:
    - the IUI of the intended extension as provided in the service request,
    - the date/time provided in the service request (this date/time is *not* and may be different from the date/time of this service request),
    - the local concept identifier for the intended extension,
    - the IUI of the O/T-version to which the local concept-identifier is added,
    - the *new* existence string *but without the last character* (note that this truncated new existence string should be equal to the submitted existence string if an existence string was submitted)
    - the *new* relevancy string *but without the last character* (note that this truncated new relevancy string should be equal to the submitted relevancy string if a relevancy string was submitted)
    - the *new* existence error string *but without the last character* (note that this truncated new existence error string should be equal to the submitted existence error string if an existence error string was submitted)
    - the *new* relevancy error string *but without the last character* (note that this truncated new relevancy error string should be equal to the submitted relevancy error string if a relevancy error string was submitted)
    - the *new* coding error string *but without the last character* (note that this truncated new coding error string should be equal to the submitted coding error string if a coding error string was submitted)

        - the IUI of the service activation
    - only if the *release date/time* (as found in the *O/T-version registration table*) is different from the submitted date/time, store *also* (thus in a second record) the following information in the *concept registration table*:
        - the IUI of the intended extension as provided in the service request,
        - the *release date/time* (thus *not* the submitted date/time)
        - the local concept identifier for the intended extension
        - the IUI of the O/T-version to which the local concept-identifier is added,
        - the complete *new* existence string
        - the complete *new* relevancy string
        - the complete *new* existence error string
        - the complete *new* relevancy error string
        - the complete *new* coding error string
        - the IUI of the service activation
    - store in the *O/T-version content table*:
        - the IUI of the intended extension as provided in the service request,
        - the local concept identifier for the intended extension
        - the IUI of the O/T-version to which the local concept-identifier is added,
        - the IUI of the service activation

9.14 Service for calculating the change history of an O/T

(ComputeChange)

Input:

- IUI of the service requester
- IUI of the O/T for which the change history must be computed
- optionally, the IUI of the O/T-version according to which view the quality of the previous versions of the O/T must be computed, from now called on 'comparison version',
- an "error-rate string" which specifies the magnitude of the error committed for each possible combination of change

HT-side activities in addition to applicable general behavior:

- check whether an IUI for the O/T is provided. If not, return and register a "O/T IUI missing" error code,
- check whether an IUI for a comparison version is provided. If not, find in the *O/T-version registration table* the IUI of the latest version that has been registered for this O/T and use further this version as the comparison version,
- check whether there are local concept-identifiers registered to this O/T, if not, return and register a "O/T is empty" error code,
- check whether an "error-rate string" is provided. If not, return and register a "error rate string missing" error code,
- check whether the "error-rate string" is of the right form:
    - It must consist of 1 or more blocks each of which has 8 characters with only certain characters allowed per position:
        - position 1 (belief in existence at t1): either "N" or "Y"
        - position 2 (belief in existence at t2): either "N" or "Y"
        - position 3 (sort of error committed): "R", "U", "C", or "N"
        - position 4 (belief in relevance at t1): "-", "N" or "Y"
        - position 5 (belief in relevance at t2): "-", "N" or "Y"

    - position 6 (sort of error committed): "R", "U", "C", or "N"
    - position 7 (sort of encoding error): "N", "W", "M", "D"
    - position 8 (error magnitude): one of the digits "0" to "9"
  - the total length of the error rate string is thus 8, 16, 24, … characters depending on the number of blocks.
  - there may not be blocks that have the same first seven characters and a different 8th character,
  - If the error rate string does not follow these rules, return and register a "error rate string ill-formed" error code.

If there are no errors in the above,

  - find in the *O/T-version registration table* all O/T-versions registered up to and including the comparison version; use the release dates in this table to exclude versions with a release date later than the one of the comparison version; store the adequate versions and their release dates in a list called 'version list';
  - find in the *concept registration table* all the intended extensions that are registered for this O/T and put them in a list (no doubles allowed) which is from now on called 'concept-list',
  - find in the *concept registration table* for each intended extension present in the concept-list *all* updates that have been applied to this intended extension before or on the release date of the comparison version using the date/time provided as input to the *UpdateConcept* service request (this date/time is *not* and may be different from the date/time of this service request) stored in this table and create for ***each*** concept X from the concept list a list called 'update list of X' which holds the date/time as well as the IUI of the service request instance in relation to the applicable updates (thus not including the updates that have been effected after the release date of the comparison version),
  - find in the *O/T-version content table* for each intended extension present in the concept-list *all* O/T-versions to which the intended extension is registered, and create for each concept X from the concept list a list called 'version list of concept X' which holds the IUIs for the service request instance stored in that table thereby eliminating the versions which are later than the comparison version,
  - collect in a list called 'date/time list' all the date/times (no doubles allowed) stored in all update lists of X and in the version list, and sort that list from earliest date/time to latest date/time,
  - build a table with (dt+7) columns where dt is the number of date/times in the date/time list, and 2c+3 rows, c being the number of intended extensions in the concept list,
  - fill the first row of the table, starting in the ***third*** column, with the date/times, sorted from earliest to latest, of the date/time list;
  - use the version list to find which date/times correspond to release dates of an O/T version and store the IUIs of corresponding O/T-versions in the second row of the table under the corresponding data/times (note that not all date/times correspond to release dates since intended extensions may be modified by means of the *UpdateConcept* service several times in between versions);
  - fill the first column of the table, starting in the third row, with the IUIs of the intended extensions as found in the concept list, skipping one row each time;
  - put the string "Total" in the first cell of the last row;
  - put in the last 5 cells of the first row the terms 'E-string", "EE-string", "R-string", "RE-string", "CE-string" respectively;

The table - except for the row and column indices - should look like this:

	C1	C2	C3	C4	C5	C6	…	C7	C8	C9	C10	C11	C12
R1			D-1	D-2	D-3	D-4	…	D-dt	E-string	EE-string	R-string	RE-string	CE-string
R2			IUIv1		IUIv2	IUIv3	…	IUIv-comp					
R3	IUIc1												
R4													
R5	IUIc2												
R6													
	…		…	...	...	...	…	…	…	…	…	…	…
R7	IUIcc												
R8													
R9	Total												

- use the ***concept registration table*** to retrieve for each IUIcx the existence string, existence error string, relevancy string, relevancy error and coding error string for the ***last*** date/time for which such strings are available (if the last version contains the first occurrence of an intended extension, there will be no error strings) and put these strings in the last 5 columns respectively in the rows where the IUI for the intended extension is written in the first column;
- because the *RegisterConcept* service requires a belief about the existence and relevance of an intended extension to be expressed prior to ***and*** since the date of adding the local concept-identifier on the O/T-side, the E-string, R-string etc. of any intended extension X are each 1 character longer than there are dates in the update-list of Concept X; furthermore, not all intended extensions had to be updated on all dates; therefore, to keep track of which characters in the various strings relate to which date/time period, do the following in the rows which have in the first column an IUI for an intended extension:
    - use the earlier created "update-lists for concept X" (where X here refers to the intended extension denoted by the IUI) (or the *concept registration table*) sorted by the date from earliest to latest, to find out which date/times listed in the first row of the table apply to that intended extension. Enter in the cell for the earliest date/time and for the intended extension being processed, the number '1'; enter in the cell of the same row and the column for the second date/time found the number '2', and so forth. The last number entered for an intended extension (thus for the latest date/time found for that intended extension) should correspond to the length of the E-string minus 1;
    - enter on the row of each intended extension in the cells of all the columns which precede the column in which there is the number 1 and starting with column 2, the number '0'.

At this stage, the table may look like this (we ignored the columns and rows in which there is "…"):

	C1	C2	C3	C4	C5	C6	…	C7	C8	C9	C10	C11	C12
R1			D-1	D-2	D-3	D-4	…	D-dt	E-string	EE-string	R-string	RE-string	CE-string
R2			IUIv1		IUIv2	IUIv3	…	IUIv-comp					
R3	IUIc1	0	0	0	1	2			YYN	NUR	YY-	NUR	NNN
R4													
R5	IUIc2	0	0	0	0	0		1	YY	NN	YY	NN	NN
R6													
	…		…	…	…	…	…	…	…	…	…	…	…
R7	IUIcc	0	1		2				YYY	NNN	YYY	NNN	NNN
R8													
R9	Total												

- The cells of the table - marked in red background - in the rows just beneath the rows in which there is an IUI in the first column should now be filled with computed error values on the basis of a number of algorithms depending on the characteristic of the cell; the bottom row cells (in blue) must later contain the total error related to the corresponding version, as computed by another algorithm;

  - For all Error-cells that satisfy these two conditions:
    - are one row lower than a cell in which there is a number, and
    - are not below the number 0

    (we use for example here R4C5):

  - retrieve from the *concept registration table* the existence string and relevancy string of the intended extension whose IUI is in the first column of the row above the error cell being computed (for the example: IUIc1) and this for the n-th date/time that is registered for that intended extension, where n is the number in the cell above the error cell being processed (here 1)
  - construct a string which is the concatenation of:
    - the (n+1)th character of the retrieved existence string, where n is still the digit in the cell just above the error cell (for the example: n=1, thus the 2nd character must be taken here)
    - the (n+1)th character of the E-string in the row above the error cell (for the example: the 2nd character of "YYN" in R3C8 = "Y")
    - the (n+1)th character of the EE-string in the row above the error cell (for the example: U)
    - the (n+1)th character of the retrieved relevancy string,

   - the (n+1)th character of the R-string in the row above the error cell (for the example: Y)
   - the (n+1)th character of the RE-string in the row above the error cell (for the example: R)
   - the (n+1)th character of the CE-string in the row above the error cell (for the example: N)
- compare this string with the blocks provided in the error-rate string given as input to this service;
- if there is a block of which the first 7 characters is equal to the concatenated string, write the digit which is the 8th character of that block as error value in the error-cell being computed; if there is no such block: write 0 in the error-cell. For the sake of example. if such a block were found and the 8th character of that block were "4", the table would now look like this:

	C1	C2	C3	C4	C5	C6	…	C7	C8	C9	C10	C11	C12
R1			D-1	D-2	D-3	D-4	…	D-dt	E-string	EE-string	R-string	RE-string	CE-string
R2			IUIv1		IUIv2	IUIv3	…	IUIv-comp					
R3	IUIc1	0	0	0	1	2			YYN	NUR	YY-	NUR	NNN
R4					4								
R5	IUIc2	0	0	0	0	0		1	YY	NN	YY	NN	NN
R6													
	…		…	…	…	…	…	…	…	…	…	…	…
R7	IUIcc	0	1		2				YYY	NNN	YYY	NNN	NNN
R8													
R9	Total												

- For all error-cells below the number '0',
   - find for the corresponding intended extension the E-string and R-string for the ***first*** registered date/time,
   - concatenate a string similarly as just explained but using the ***first*** character instead of the (n+1)th,
   - write the corresponding error-value in the cell
- Find now in the table for each intended extension the error cells for date/times for which no update is provided and that come before ***and*** after (thus ***in between***) updates for that extension (the only example in the table is R8C4). Write in such cells the value of the error-cell that precedes it (for the example, that would be the value which would be in R8C3).
- Finally, find now in the table for each intended extension the error cells for date/times for which no update is provided and that come after the ***last*** registered update. In the example table, that are the cells R4C7, R8C6

and R8C7. If the last character of the E-string of that row is "Y" and the last character of the R-string = "Y", search in the submitted error rate string the block "NYU-YUN" and write the corresponding error-value (the 8th character of that block) in each of these error cells for that row.

At this time, all error-cells should have received a value. The last step is to compute the totals for each version which is to be done in the following way (for each column corresponding to a version):

- o find in the submitted error rate string the highest assigned error-value (the maximum is 9, but it may not have been assigned)
- o replace the values in the error cells by subtracting the current value of that cell from the highest assigned error-value (if the original value in the cell would be 2, and the highest assigned value in error-rate string, is 7, the new value would be 5);
- o make the total of all error-values in the column and write the result in the bottom cell of the corresponding column.

If no errors in all of the above, return to the requester:

- o the IUI of this service activation
- o a list containing for each version:
  - ▪ the IUI of the version
  - ▪ the date/time of the release of the version
  - ▪ the total error-value
- o a pointer (URI) where the entire table is discussed above can be retrieved.

10 Implementation of the history tracker

The History Tracker (HT) application provides centralized services for ontology authoring applications as described in section 9. Ontology authoring applications should be able to call the services of the HT from any programming environment, e.g. php, C/C++ and Java.

The services are implemented as Java classes, where a separate java class implements each service. The services interface is provided through the Web Services RESTful architecture based on the REpresentational State Transfer (REST) design idiom that embraces a stateless client-server architecture in which the web services are viewed as resources and can be identified by their URLs. A RESTful service can be invoked through an http client library, which is available in all programming languages either as built-in or third party implementations.

The general schema of a web service offer by the HT is:

> *http://<host domain name>:<port>/ontversioning/?format=<xml/html>servicename=<provide a name>&<service-parameters as name=value pair>*

Text with angular brackets <> serves as a placeholder for an actual value. The following is an example of an HTTP URL for registering a user in HT

> *http://localhost:8080/ontversioning/?format=xml&service-name=register-user&ot_user_identifier=Werner*

10.1 List of Services and their parameters

Service Name	***Parameters***
get-service-activation-info	service_activation_iui, ot_user_iui
register-user	ot_user_identifier
register-ot	ot_identifier, ot_user_iui
set-permissions	ot_user_grantee_iui, ot_user_iui
retract-permissions	ot_user_grantee_iui, ot_user_iui
get-iui-of-ot-by-service-activation	service_activation_iui, ot_user_iui
get-iui-of-ot-by-local-identifier	ot_identifier, ot_user_iui
register-ot-version	ot_version_identifier, ot_user_iui, date_time
get-ot-versions-by-iui	ot_user_iui
get-ot-versions-by-local-identifier	ot_identifier, ot_user_iui
register-concept	existence_string_error, relevancy_string, existence_string, local_concept_identifier, ot_user_iui, relevancy_string_error, date_time
update-concept	intended_extension_iui, existence_string, date_time, local_concept_identifier, relevancy_string_error, existence_string_error, ot_user_iui, relevancy_string
register-concept-to-version	relevancy_string, intended_extension_iui, existence_string, local_concept_identifier, ot_user_iui, ot_version_iui, date_time
compute-change	ot_user_iui, error_rate_string, ot_iui, ot_version_iui

10.2 System architecture.

The application is implemented based on the client server architecture mechanism. The client side represents the applications that call the services of the HT Server application. The application embeds a jetty web server, and it runs in the jetty (an http web server) environment.

All client requests are received by an HtServlet (an Http Servlet) component. An Http servlet is part of the java enterprise technologies, which handles all web requests and send responses (generate a web page) to users. A request is sent to the server as an URL, e.g.

> *http://localhost:8080/ontversioning/?format=xml&service-name=register-user&ot_user_identifier=Werner,*

and the Jetty server invokes the HtServlet to the handle request. The HtServlet executes the service whose name is provided in the request, and if the service is not found then the servlet returns an error. The service performs its activities and returns the results in the format (xml or html), which is requested in the URL. After performing all of its activities, the service saves data into the HT DB database through a hibernate layer[5] i.e. a java library which maps java objects to a relational database system. We have build java objects for all tables mentioned in the services' functional and technical specifications and mapped those objects to a MySQL database through the hibernate architecture.

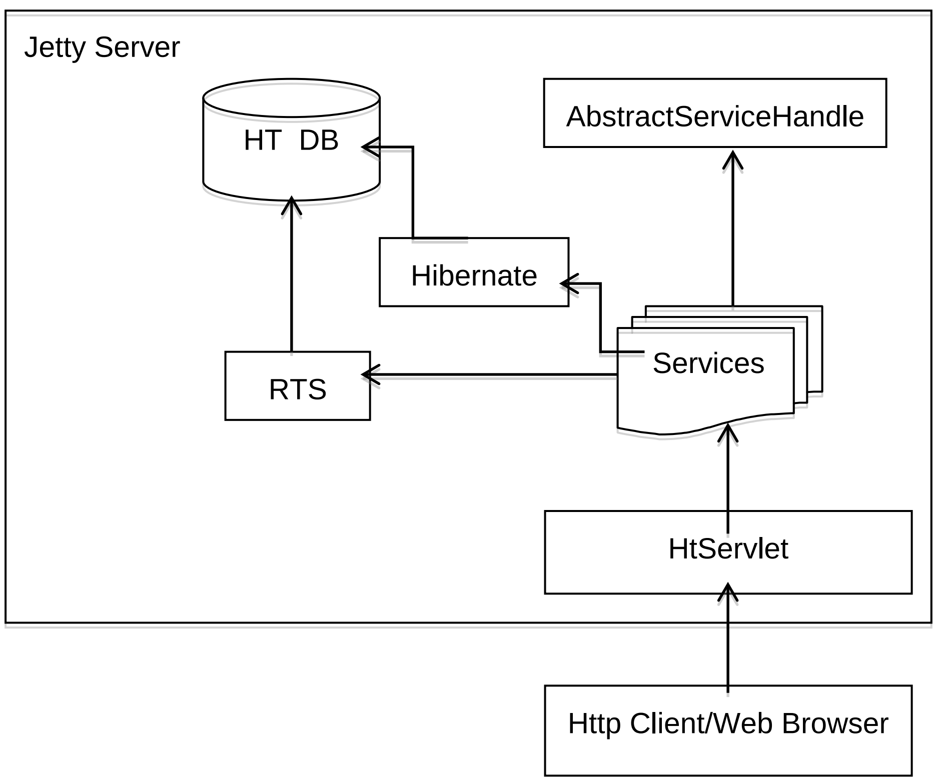

Figure 10: System Architecture

[5] http://www.hibernate.org/

10.3 Software Installation -- Server Side Setup

MySQL server is required to be running before running the HT server. We have currently tested the system on the 5.1.44 version of MySQL.

The directory structure of the application is:

> ***ontversioning****: the root directory of the project.*
>
> > ***etc****: contains the configuration files of the jetty*
> >
> > ***conf****: contains the configuration files of the HT server*
> >
> > ***lib****: contains all the third party components used by the HT server*
> >
> > > *logs: contains log file generated by the HT server.*
> >
> > ***webcontents****: contains html and image files*
> >
> > ***src****: contains the java source of the project*
> >
> > ***ontversioning****.**jar**: the binary/compiled file of the project.*
> >
> > ***build****.**xml**: is the ant build file[6] which compiles the project.*

The system requires configuration through properties files[7] (a property file contains key=value pair properties) for database connection. The database connection configurations are for

- **Host:** Domain name (localhost or www.referent-tracking.com) or IP address of the machine running the MySQL database server
- **User Name:** The database user-name
- **Password:** Password of the database user-name

and are set in the *conf/rts.properties file:*

- rts.db.host=localhost
- rts.db.user=root
- rts.db.password=password

The *conf/hibernate.cfg.xml* file contains the properties for the hibernate layer of the application.

- <property name=*"hibernate.connection.password"*></property>
- <property name=*"hibernate.connection.url"*>jdbc:mysql://localhost/ontv</property>
- <property name=*"hibernate.connection.username"*>root</property>

The conf/ont-versioning.properties file contains the properties for the HT application which helps to initialize the database:

- org.rtu.ontology.versioning.username=root
- org.rtu.ontology.versioning.password=
- org.rtu.ontology.versioning.host=localhost

[6] http://ant.apache.org/

[7] http://commons.apache.org/configuration/howto_properties.html

11 Can realism-based versioning help decision making on upgrade policies?

Changes in SNOMED CT have been quite dramatic over time. This raises several questions concerning the impact these changes have on data collections which are coded in terms of - usually a small subset of - SNOMED CT concepts. Sensible questions are, for instance,

- when it is worthwhile to use a new version since revisions might be outside the scope of the data collection,
- whether analyses performed using an earlier version are rendered meaningless because of the inactivation of concepts in later versions,
- whether a new version contains more or less knowledge or is a mere reformulation of the same amount of knowledge.

Building further on the insight obtained thus far, we tried to find out whether answers to such questions can be found, and what would be possible strategies to find reliable answers in operational environments in which research on such issues is not part of the core activities.

11.1 Background

SNOMED CT has thus far primarily been researched in terms of (1) the coverage that it provides to support coding in specific domains, [89-90] (2) the reliability and validity of such coding efforts, [91-92] and (3) its ontological coherence and consistency [32, 58, 93], and, thanks to this grant, (4) how SNOMED CT evolves over time and how to translate such changes into measures indicating (1) how much a new version of a terminology is better than any previous version and (2) to what degree terminology changes reflect evolutions in the underlying domain or the terminology authors' understanding thereof.[1]

To our best knowledge, only Wade and Rosenbloom have thus far addressed the impact of SNOMED CT's evolution on operational applications, with the conclusion that '*While the efforts of each subsequent SNOMED CT version aim for continual improvement, changes made to its core structure and post-coordination guidelines make it more difficult to migrate proprietary data to this reference standard*'.[94] That this issue thus far has not received the attention that it deserves can be explained by the rather limited number of actual implementations, a situation that probably will change dramatically in the near future.[16]

11.2 Methods

The study presented here focused on a subset of 883 SNOMED CT concepts - from here on referred to as *source concepts* - used within a cancer clinic for encoding synoptic pathology reports and tumor registry data and for querying a biospecimen repository, all together covering almost 16,000 occurrences related to 10,000 unique patients.

For each source concept, all concepts - from here on referred to as *target concepts* - within the transitive closure set of the *Is a* relation and all hierarchical relations - *Was A*, *Replaced By*, *Same As*, *May Be*, *Moved To*, and *Moved From* - were computed for each SNOMED CT version from January 2002 to July 2010, together with their concept status and path length towards the source concept. Computing the transitive closure set involved traversing the target of each of the relationships included in the Relationships Table of each version to look for and follow further relationships until all paths through the hierarchy reach the root concept (closure). When a target concept could be reached by traversing more than one path, the shortest path length from source concept to target concept was preserved. **Table 11** (page 36) shows these computations for the source concept '44228008: *Surgical margins involved by tumor* (finding)'. **Table 12** (page 37) displays the rules used to compute the composite relationships during the transitive closure computation of this concept.

Then, again for each version, the *genericity* of each target concept was computed, where *genericity* was defined as the number of times a target concept is on a path from all source concepts to the top. The maximum value for genericity, under this definition, was 883, i.e. the number of source concepts. These values were then used to compute for each source concept SC its *information content* defined as the sum of the values obtained by dividing the genericity of each target concept TC on a path from SC to the top by the respective path lengths from SC to TC. **Table 15**, as an example, shows the results for the concept '*pN1b: Metastasis in internal mammary lymph nodes with microscopic disease detected by sentinel lymph node dissection but not clinically apparent (breast) (finding)*'.

Table 15: Example of the calculation of the information content of a source concept.

Target Concepts	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15	v16	v17	v18
SNOMED CT Concept (SNOMED RT+CTV3)	0	58	56	103	139	104	104	104	93	93	93	93	93	93	93	93	93	93
Staging and scales (staging scale)	0	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Tumor staging (tumor staging)	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Cancer staging (tumor staging)	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Tumor-node-metastasis (TNM) tumor staging system (tumor staging)	0	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Generic tumor staging descriptor (tumor staging)	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Tumor-node-metastasis (TNM) classification of malignant tumor after operation (observable entity)	0	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N category (observable entity)	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
pN1 category (finding)	0	3	5	10	11	11	11	11	11	11	11	11	11	11	11	11	12	12
pN1: Metastasis in 1 to 3 axillary lymph nodes, and/or in internal mammary nodes with microscopic disease detected by sentinel lymph node dissection but not clinically apparent (breast) (finding)	0	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
Finding (finding)	0	0	19	57	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Clinical history and observation findings (finding)	0	0	16	54	0	54	54	54	46	46	46	46	46	46	46	46	46	46
Clinical finding (finding)	0	0	19	64	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Tumor finding (finding)	0	0	22	80	81	81	81	81	65	65	65	65	65	65	65	65	65	65
Node category finding (finding)	0	0	5	10	11	11	11	11	11	11	11	11	11	11	11	11	11	11
Tumor stage finding (finding)	0	0	18	0	0	0	0	0	56	56	56	56	56	56	56	56	56	56
Tumor-node-metastasis (TNM) tumor staging finding (finding)	0	0	22	71	73	73	73	73	73	73	73	73	73	73	73	73	73	73
pN category finding (finding)	0	0	6	14	15	15	15	15	15	15	15	15	15	15	15	15	15	15
N1 category (finding)	0	0	3	7	8	8	8	0	0	0	0	0	0	0	0	0	0	0
Breast TNM finding (finding)	0	0	10	10	12	12	12	12	12	12	12	12	12	12	12	12	12	12
Clinical finding (finding)	0	0	0	0	88	63	63	63	55	55	55	55	55	55	55	55	55	55
Finding of lesion (finding)	0	0	0	0	0	65	65	65	54	54	54	54	54	54	54	54	54	54
pN1b category (finding)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4
Total Information content	**0**	**123**	**203**	**484**	**440**	**500**	**500**	**492**	**493**	**493**	**493**	**493**	**493**	**493**	**493**	**493**	**498**	**498**

The sum of the information contents of all source concepts within a version would then yield the information content of that entire version.

A *first hypothesis* is that the evolution of the information content of the versions over time can be used as an indicator to decide whether to upgrade to a new version.

Intermediate inspection of these results suggested that the procedure thus far described could be used to detect possible mistakes. The grey shaded cells in Table 3 indeed show that in some versions target concepts for the source concept disappeared from the transitive closure set while reappearing in later versions. It was also discovered that when target concepts permanently disappeared from the transitive closure set, this not always could be explained by the retirement of the target concept within the corresponding version. Although this does not mean that there is a mistake - it might rather be the correction of a mistake - it was decided to register this and similar phenomena as a *suspicious event*. Each source concept / target concept pair was therefore additionally marked as being the seat (or not) of such an event and for each version tallies for such events were computed for all such events over all previous versions until another change was effected. Thus if a change was marked in some version as being a suspicious event, it stayed marked as such until in some later version - if any at all - another change occurred that did not meet the requirements for being suspicious anymore.

A *second hypothesis* is that evolution of these tallies over time, the *suspicious event perseverance*, yields a second indicator for migrating to a new version of SNOMED CT.

11.3 Results

The 883 source concepts studied were by means of 15,689 relationships linked to 1,415 target concepts which is only a small fraction of the total number of concepts in SNOMED CT. Of the 15,689 relationships, 28.73% were found to be suspicious.

Figure 11 and **Figure 12** show respectively the evolution of the information content and of the notable event perseverance of the source concepts over time. The biggest increase in information content occurs over the first few versions, with the exception of version v5 (January 2004). Also version v17 (January 2010) shows a minor (barely notable on the chart) increase: from 384960.7605 to 385449.781.

Significant changes in the suspicious event perseverance are those which constitute a downwards trend break, thus a reduction in the perseverance. This is here the case for versions v6, v7, and v14.

Both indicators together thus suggest that applications using the set of source concepts studied do not benefit from upgrades to SNOMED versions issued between July 2005 and January 2008, nor from both 2009 versions, nor July 2010.

11.4 Discussion

The information content of representational units in ontologies is usually studied from the perspective of semantic similarity.[95] In [96] it is discussed how it can be used as well for quality control in ontology development. Hogan and Slee used Shannon's information entropy which is somehow related to information content as defined here to suggest the use of SNOMED CT instead of ICD-9-CM for coding diagnoses.[97] Measures for information content have thus far not been used to assess whether it is worthwhile to upgrade from one version of an ontology to another.

An increase in information content from one version to another as defined here can be brought about by several sorts of changes: introduction of new source or target concepts in the ontology and the creation or elimination of relationships between intermediary representational units

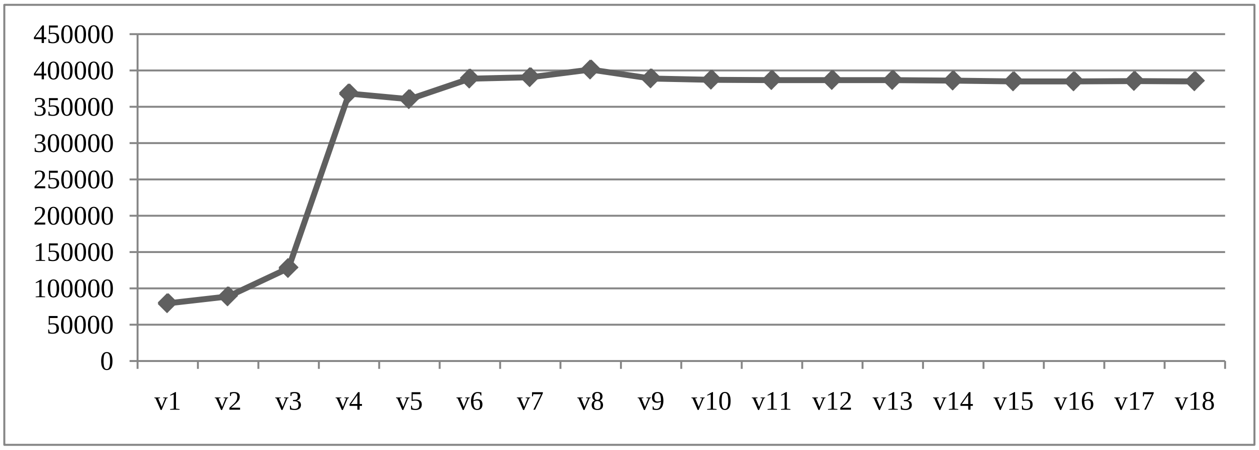

Figure 11: Evolution of the information content of all source concepts over all versions.

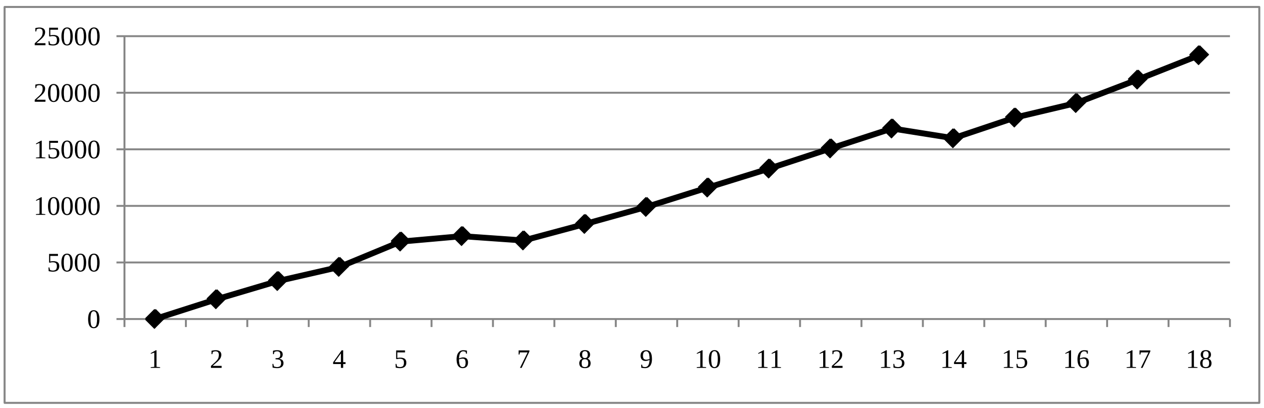

Figure 12: Evolution of suspicious event perseverance of all source concept/target concept pairs.

Table 16: Evolution of suspicious events

	N	Binary %
Unmarked	15,689	
Stay unmarked	11,182	71.27%
Become suspicious	4,507	28.73%
Stay suspicious	1,812	40.20%
Become unmarked	2,695	59.80%
Stay unmarked	2,296	85.19%
Become suspicious	399	14.81%
Stay suspicious	332	83.21%
Become unmarked	67	16.79%
Stay unmarked	66	98.51%
Become suspicious	1	1.49%
Stay suspicious	0	0.00%
Become unmarked	1	100.00%

along the transitive closure paths of the source concepts. It is the former which in the set of source concepts studied here is responsible for the substantial information content increase in the first seven versions. Situations like this will occur in the first place when data repositories make use of codes drawn from classification systems that are not yet completely mapped to a reference terminology such as SNOMED.

The notion of suspicious event as defined here is a novelty that might not only be useful for the purpose of decision making with respect to the application of a new version, but also for quality assurance purposes concerning the further development of SNOMED CT. Specifically the temporal disappearance of a target concept from a transitive closure set raises questions about the adequacy of the quality assurance principles used in the SNOMED CT authoring environment. We found 500 occurrences of such temporal disappearances which in light of the 15,689 individual relationships might seem only a small fraction, but is quite large as these erroneous deletions originate from only 833 source concepts, and are targeted towards only 1415 target concepts.

A first limitation, perhaps, of the methodology is the computational power required for the calculations. Transitive closure computations are quite computer resource consuming specifically for very large sets of source concepts. At the other hand, there is no manual intervention needed to calculate the indicators.

A second limitation is the uncertainty, for all cases, about whether a suspicious event is indicative for a mistake. Here we see a use case for the SNOMED CT developers to treat relationship components in the same way as the other components, i.e. by explicitly retiring relationships with corresponding error codes.

Although the results presented here seem promising, the adequacy of the suggestions made by the two indicators remains further to be investigated. The problem is however: how? What would be the gold standard against which to measure whether an application performs better using a new version of an ontology as compared to an earlier version, and this without changing anything to the application itself? This is of course only possible if the application is designed in such a way that, for instance, the mechanisms for query formulation are implemented generically enough such that users can formulate queries with a new version that weren't possible to formulate with an earlier version. This, in the case presented here, is part of future endeavors.

12 Conclusion

Our research was focused around the following specific aims:

Aim 1: analyze SNOMED CT's existing history mechanism to find out whether the principles of realism-based ontology versioning are able to cope with all requirements put forward by SNOMED CT. Adjust when needed.
Aim 2: develop a prototype of a realism-based ontology versioning software component that can serve as plug in for ontology authoring systems such as Protégé, ODE or SWOOP.
Aim 3: use the prototype to restructure SNOMED CT's history information in line with the principles of realism-based ontology versioning.
Aim 4: compute the quality improvement of SNOMED CT over time in order to demonstrate the usefulness of the approach and foster its acceptance in other ontologies.

All aims have been achieved completely, although to meet our final objectives we had to readjust the detailed work plan as initially conceived in terms of seven specific tasks (**Table 17**).

Table 17: Completion of tasks in relation to specific aims

Tasks	***Aim***	***Results***
1	1	Data collection and preparation: • we obtained all SNOMED CT US distributions from January 2002 until July 2010, • we wrote Extract-Transform-Load (ETL) routines to create a relational database suitable for our analysis needs.
2	1	Statistical data analysis: • we computed lists of concepts exhibiting the highest number of changes over time. They turned all out to be candidates for post-coordination which is a useful finding itself, but not in light of our goals. • we analyzed a sample of 1,000 randomly selected concepts (n=264) and descriptions (n=736) that underwent a status change of some sort, the goal being to find underlying principles to translate automatically SNOMED CT's 'reasons for change' to our realism-based change configurations. This produced useful results but with some caveats. • we created in addition a subset composed of (1) 883 SNOMED CT concepts used within a cancer clinic for encoding synoptic pathology reports and tumor registry data and for querying a bio-specimen repository, all together covering almost 16,000 occurrences related to 10,000 unique patients, and (2) 1,415 concepts present in the transitive closure set of the former by means of 15,689 relationships. The analysis produced useful statistics to decide on the basis of the history information whether users should upgrade to a new version of SNOMED CT.
3	1	Detailed analysis of SNOMED CT's history mechanism: • we created graphing software and produced various sorts of grafts showing the extremely complex change history of concepts retrieved on the basis of either (1) key phrases or (2) homonymic terms. • we compared SNOMED CT's current history mechanism and our novel method based on Ontological Realism and outlined ambiguities and areas of missing information. • we created a Semantic Wiki as a prototype example of how the two mechanisms can be combined.

		• we improved and expanded our methodology to represent more accurately not only - as originally aimed for - what SNOMED CT authors must believe to have changed in reality or in their understanding with respect to the *last* version, but with respect to *all* previous versions.
4	2	Requirements specifications for the realism-based versioning prototype: • we described the functionalities and procedures that have to be implemented in a prototype that is able to support realism-based ontology versioning based upon the improved methodology arrived at in task 3.
5	2	Prototype development: • we implemented the functionalities and procedures identified in task 4 as Java classes in a Web service architecture. The services interface is provided through the RESTful Web Services architecture. Each RESTful web service can be invoked through an http client library, which is available in all programming languages either as built-in or third party implementations.
6	3	Applying realism-based ontology versioning to SNOMED CT: • we used the principles for history mapping developed under task 2 and 3 to generate a history view of SNOMED CT compatible with our new method.
7	4	Measuring quality improvements in SNOMED CT: • we used the view developed in task 6 to compute the believed quality improvements of SNOMED CT since its inception. • we found that under our view the quality of SNOMED CT between the January 2002 and July 2009 versions increased for concepts by 18.8%, for descriptions by 47.7% and for relationships by 178.1% under the assumption that the July 2009 version were accurate.

12.1 Problems encountered

12.1.1 Unclarity and inconsistency about what SNOMED CT concepts exactly denote

The meaning of a SNOMED concept is claimed to be determined by the Fully Specified Name (FSN) and for a few hundred cases by an additional textual definition; thus by reading the FSN, SNOMED CT users are supposed to grasp exactly what is meant. The logic definitions, in contrast, are an attempt to formulate a representation of the named concept in a way that enables effective processing and inference but should not be taken as definitions for that meaning.

This approach, however, is not satisfactory as in many cases the available information around a concept seems to contradict this view. In its January 2009 version, for example, SNOMED CT associates the concept '*Fractured nasal bones (disorder)*' with the following synonyms:

- o 'Fractured nasal bones' (S1),
- o 'Broken nose' (S2),
- o 'Fractured nose' (S3),
- o 'Fracture of nose' (S4),
- o 'Fracture of nasal complex' (S5), and
- o 'Fracture of nasal bones' (S6).

One consequence of the multiple interpretations that are given to the term 'concept' both inside [32] and outside [48] of SNOMED CT is that it is difficult to understand precisely how this 'association' is to be understood. In practice, what it means is that SNOMED is here acknowledging the different ways language users capture nasal bone fracture-related information when entering patient data into a record, and providing an aid to translating the corresponding bodies of data into SNOMED form. As realist ontology (and common sense) would suggest, however, it can be assumed that when a study nurse enters the term ‘fractured nasal bones’ into a patient record, then what he means thereby is not a *nose of a certain (fractured) sort* but rather a certain *group of bones.* If, accordingly, we are to devise a strategy for translating the resultant SNOMED data into the realism-based framework, then our mapping will need to take account of the mentioned ‘associations’ in a more careful way than is possible when all the mentioned synonyms are treated *en bloc*. It is for this reason that we introduce the machinery of ***CLA***s and ***GRP***s as explained in section 3.2.1 and further. This machinery is designed to make apparent the unarticulated complexity of SNOMED’s synonymy relation by allowing each synonym to be treated separately in a way which at the same time allows formulation of the needed mappings to the corresponding OBO Foundry terms.

Human bones and noses are represented in the FMA Anatomy Ontology [13] by means of representational units denoting the universals *bone* and *nose* respectively. Fractures, in contrast, would be included in an ontology of disorders [63]. To realize our proposed strategy, now, scholars developing a mapping from SNOMED CT to OBO Foundry ontologies would have to decide, in collaboration with the SNOMED authors, what precisely the synonymous terms (S1–6) mentioned in our list above should properly be understood as denoting. In the framework here proposed, for example, S2 and S3 would both denote a ***GDC*** that is a subgroup of the extension of the universal *nose*. S1 would denote, according to context, either a ***GRP*** which has *nasal bones* as members or a ***GDC*** denoted by the plural term ‘bones of the nose’.

Another advantage of our strategy is that it helps us to understand the structure of the *is a* hierarchy in SNOMED CT. 44 concepts in SNOMED CT are described as being *is a* parents of *Fractured nasal bones (disorder)*. Where all of the synonyms referred to above denote first-order entities on the side of the patient, this is not the case for all 44 of the parent concepts listed. ‘*Disorder by body site (disorder)*’, for example, reveals itself upon inspection to denote not a disorder at all, but rather certain representational units of SNOMED CT itself, which are organized according to the body sites where the corresponding disorders occur. Another problematic case is ‘*Finding by site (finding)*’: fractured nasal bones cannot, in our terms, be a (type) of finding, since something can only be found – and hence give rise to a finding – if it pre-exists, and is thus independent of, the corresponding act of observing. On our strategy, in fact, finding data would be mapped, not to bones directly, but rather to the corresponding datable observations.

12.1.2 Underspecification of SNOMED CT's 'reasons for change'

Although the principles of realism-based ontology versioning were found to cope with SNOMED CT's requirements (aim 1), the opposite turned out not to be the case: SNOMED CT's history mechanism, and in particular its own 'reasons for change' as coded in SNOMED CT distributions do not provide enough information to allow third parties to translate these reasons for change into the various change configurations recognized by realism-based ontology versioning.

12.1.3 Absence of history information for relationships

SNOMED CT's absence of version management for the relationships table has the consequence that when some relationship is present in some versions and not in others, it cannot be assessed whether the absence corresponds to a real absence or an implicit presence inferable through description logic reasoning. These computations were not feasible with the technology publicly available in the course of the project.

12.2 Recommendations

In 2010, the International Health Terminology Standards Development Organization (IHTSDO) announced the future distribution of SNOMED CT under a new format called *'RF2'* [98] of which more detail became officially available with the January 2011 version [99-101]. The RF2 format is claimed to offer greater flexibility and more explicit and comprehensive version control than RF1 with new features for configuration management thereby accommodating evolving requirements without a need for further fundamental change in the foreseeable future [101]. One such feature is that RF2, through the introduction of a new hierarchy called the '*SNOMED CT Model Component*' [99] which includes the existing Concept Model, allows SNOMED CT to be described in terms of its own structure thereby reducing, so it is hoped, the burden and costs incurred by content developers, implementers and release centers while at the same time improving product functionality and quality.

The current documentation of RF2 is marked by a focus on making language- and realm extensions as well as mappings towards other terminologies more manageable. It introduces in addition a number of merely cosmetic changes to the existing history mechanism. But at first sight, it seems also to hold much promises to deal with a number of issues concerning the ontological underpinnings of SNOMED CT that have been reported upon in the literature such as, for example, the underspecification of reasons for change [54], the (in)adequacy of SNOMED's intensional and extensional definitions [102], its still incoherent ontological commitment [32], and the ambiguities and conflations in its conceptual structures and in its treatment of terms proposed as 'synonyms' [55].

Unfortunately, the documentation of RF2 is not yet explanatory enough and lacks clearly worked out examples to assess for each issue identified whether it can be resolved by merely introducing new Model Component entries and associated data types or whether other measures are required as well.

Our first – and by far not exhaustive – proposal is therefore formulated in terms of the following *four recommendations* which experts in RF2 can then implement more adequately in the new format they have designed:

5. do not make double use of the ConceptID as an identifier for the concept and an identifier for the Concept Component;
6. add to each Concept Component a field that indicates to what broad category the intended referent of that concept belongs;
7. expand the Concept Inactivation Value sub-hierarchy with concepts that reference whether a change in SNOMED CT is motivated by (1) a change in reality, (2) the SNOMED CT authors' or users' understanding of reality as reflected in the advance of the state of the art in the biomedical domain, or (3) a mistake that is strictly internal in SNOMED CT as an information artifact [1], and this along the lines described in section 9 page 53.
8. add mechanisms:
    a. to represent the provenance of a class more explicitly;
    b. to separate the time-period during which a component is believed to have been valid in SNOMED CT from the period it is believed to be (or has been) valid in reality since the latest release;

These recommendations, despite being very modest, address several issues sufficiently.

The first solves the object-/meta-language confusion which gives rise to nonsensical statements such as '*person → MOVED TO → namespace concept*'.

The second solves the problem of what sort of entity in each individual case is referenced by a conceptId. Potential values for the proposed field can be based not only on the L1/L2/L3 distinction [**Error! Reference source not found.**] – roughly: first-order entities that are not about anything (e.g. person, scalpel) / beliefs, desires, intentions whether about something (e.g. a diagnosis) or about nothing (e.g. some psychotic beliefs) / and information artifacts such as staging scales, guidelines, and, indeed, SNOMED CT itself – but also on whether a universal or defined class is referenced [57], and potentially even on the putative ‘possibilia’ and ‘non-existing entities’ [6] endorsed by terminology and ontology developers who do not wish to be hampered by the complexity of Ontological Realism [103]. By doing so, SNOMED CT can even maintain a philosophically rather neutral position even though a clear shift towards OBO Foundry compatibility is observable.

And finally, the rather ad hoc motivation for inactivating concepts is catered for by our third and fourth recommendation.

13 Acknowledgements

I wish to thank the National Library of Medicine for the financial support provided to conduct the research described in this report.

My gratitude goes also to my collaborators in this research, specifically:

Shahid Manzoor, who has a Master in Computer Science from the National College of Computer Science in Lahore, Pakistan, and obtained a second MSc Computer Science at Saarland University, Germany. Prior to coming to Buffalo in October 2006, he was a full time researcher at the Institute of Formal Ontology and Medical Information Science (IFOMIS), Saarland University, Germany. He helped me in data preparation and analysis (tasks 1, 2 and 3), writing the software requirement specifications (task 4) and carried out the two prototype implementations (task 5).

Ron Rudnicki, who has a Master of Arts in Philosophy from SUNY at Buffalo and a Bachelor of Arts in Mathematics from Canisius College. Rudnicki joined the Ontology Research Group from Gartner where he acted as a Software Engineer developing a Business Intelligence application that supports Gartner's IT Benchmarking service. He assisted in the analyses in task 2 and task 3, and helped me to assess the benefits of the proposed metric in task 7.

Kent A. Spackman, Chief Terminologist of the International Health Terminology Standards Development Organisation (IHTSDO) and Fellow of the College of American Pathologists. He served as scientific director for SNOMED International, on the College's Informatics Committee and has been its representative to the ANSI Health Informatics Standards Board and Health Level Seven (HL7). His involvement in the project was to deliver and prepare the SNOMED-related data required for the project and to provide clarification for puzzles we would be confronted with when carrying out the annotation (task 6).

14 References

1. Ceusters, W., *Applying Evolutionary Terminology Auditing to SNOMED CT.*, in *American Medical Informatics Association 2010 Annual Symposium (AMIA 2010) Proceedings*. 2010: Washington DC. p. 96-100.
2. Ceusters, W., Smith, B., *Tracking Referents in Electronic Health Records*, in *Connecting Medical Informatics and Bio-Informatics: Proceedings of MIE 2005 The XIXth International Congress of the European Federation for Medical Informatics*, G.A. Engelbrecht R., Lovis, C., Editor. 2005, IOS Press: Amsterdam. p. 71-76.
3. Ceusters, W., Steurs, F., Zanstra, P., Van Der Haring, E., Rogers, J., *From a Time Standard for Medical Informatics to a Controlled Language for Health.* International Journal of Medical Informatics, 1998. **48**(1-3): p. 85-101.
4. Ceusters, W., Smith, B., *Strategies for referent tracking in electronic health records.* Journal Biomedical Informatics, 2006. **39**(3): p. 362-378.
5. Ceusters, W., P. Elkin, and B. Smith, *Referent Tracking: The Problem of Negative Findings*, in *Studies in Health Technology and Informatics. Ubiquity: Technologies for Better Health in Aging Societies - Proceedings of MIE2006*, A. Hasman, et al., Editors. 2006, IOS Press: Amsterdam. p. 741-746.
6. Ceusters, W., P. Elkin, and B. Smith, *Negative Findings in Electronic Health Records and Biomedical Ontologies: A Realist Approach.* International Journal of Medical Informatics, 2007. **76**: p. 326-333.
7. Ceusters, W., et al., *Managed convergence towards high quality electronic healthcare records in Europe : the PROREC initiative*. 1996, Medical Records Institute: Newton, MA. p. 127-136.
8. Buekens, F., W. Ceusters, and G.D. Moor, *The explanatory role of events in causal and temporal reasoning in medicine.* Methods of Information in Medicine, 1993. **32**: p. 274-278.
9. Ceusters, W., *Formal terminology management for language-based knowledge systems: resistance is futile*, in *Trends in Special Language and Language Technology*, R. Temmerman and M. Lutjeharms, Editors. 2001, Uitgeverij De Boeck: Antwerpen. p. 135-153.
10. Smith, B., Ceusters, W.,. *An Ontology-Based Methodology for the Migration of Biomedical Terminologies to Electronic Health Records*. in *AMIA 2005*. 2005. Washington, DC.
11. Ceusters, W. and B. Smith, *A Realism-Based Approach to the Evolution of Biomedical Ontologies*, in *Biomedical and Health Informatics: Proceedings of the 2006 AMIA Annual Symposium*. 2006, American Medical Informatics Association: Washington DC. p. 121-125.
12. Rosse, C., Mejino, J. , *A reference ontology for biomedical informatics: the Foundational Model of Anatomy. .* J Biomed Inform. , 2003. **36**(6): p. 478-500.
13. Rosse, C. and M.J. Jr, *The Foundational Model of Anatomy Ontology*, in *Anatomy Ontologies for Bioinformatics: Principles and Practice*, A. Burger, D. Davidson, and R. Baldock, Editors. 2007, Springer: London. p. 59-117.
14. Rosse, C. and J.L.V. Mejino, *A reference ontology for bioinformatics: The Foundational Model of Anatomy.* Journal of Biomedical Informatics, 2003. **36**: p. 478-500.
15. Ceusters, W., Smith, B., Kumar A., Dhaen C. *Ontology-Based Error Detection in SNOMED-CT®*. in *Proceedings of Medinfo*. 2004.
16. Elhanan, G., Y. Perl, and J. Geller, *A Survey of Direct Users and Uses of SNOMED CT: 2010 Status.*, in *AMIA Annu Symp Proc*. 2010 p. 207-11.
17. Elkin, P., et al., *Evaluation of the content coverage of SNOMED CT: ability of SNOMED clinical terms to represent clinical problem lists.* Mayo Clin Proc, 2006. **81**(6): p. 741-748.

18. Rector, A., *Terminologies, Ontologies, & SNOMED. What are they for? What would Quality Assurance mean?*, in *First European Conference on SNOMED CT*. 2006: Copenhagen.
19. Ceusters, W., B. Smith, and L. Goldberg, *A terminological and ontological analysis of the NCI Thesaurus.* Methods of Information in Medicine, 2005. **44**: p. 498-507.
20. Coronado, S.d., et al., *The NCI Thesaurus quality assurance life cycle.* Journal of Biomedical Informatics, 2009. **42**(3): p. 530-539.
21. de Coronado, S., Haber, M., Sioutos, N., Tuttle, M., Wright, L. *NCI Thesaurus: Using Science-Based Terminology to Integrate Cancer Research Results*. in *Medinfo*. 2004: IOS Press.
22. Kumar, A. and B. Smith, *Oncology ontology in the NCI Thesaurus*, in *Artificial Intelligence in Medicine Europe (Lecture Notes in Computer Science 3581)*. 2005. p. 213-220.
23. Schulz, S., et al., *The Pitfalls of Thesaurus Ontologization - the Case of the NCI Thesaurus*, in *AMIA Annual Symposium Proceedings*. 2010, AMIA: Washington D.C. p. 727-731.
24. Sioutos, N., et al., *NCI Thesaurus: A semantic model integrating cancer-related clinical and molecular information.* Journal of Biomedical Informatics 40, 2007. **40**(1): p. 30-43.
25. Ashburner, M., et al., *Gene ontology: tool for the unification of biology.* Nature Genetics, 2000. **25**(1): p. 25-29.
26. Ceusters, W., *Applying Evolutionary Terminology Auditing to the Gene Ontology.* Journal of Biomedical Informatics; Special Issue of the Journal of Biomedical Informatics on Auditing of Terminologies, 2009. **42**(3): p. 518-529.
27. Gene Ontology Consortium, *The Gene Ontology (GO) project in 2006.* Nucleic Acids Res., 2006. **34**: p. D322–D326.
28. Gene Ontology Consortium, *Creating the Gene Ontology Resource: Design and Implementation.* Genome Research, 2001. **11**(8): p. 1425-1433.
29. McCray, A., *An upper-level ontology for the biomedical domain.* Comp Funct Genom, 2003. **4**: p. 80-84.
30. Smith, B., *Beyond concepts: ontology as reality representation*, in *Proceedings of the third international conference on formal ontology in information systems (FOIS 2004)*. 2004, IOS Press: Amsterdam. p. 73-84.
31. Bodenreider, O., Smith, B., Kumar, A., Burgun, A. *Investigating subsumption in DL-based terminologies: A case study in SNOMED CT*. in *Proceedings of the First International Workshop on Formal Biomedical Knowledge Representation (KR-MED 2004)*. 2004.
32. Schulz, S. and R. Cornet, *SNOMED CT's Ontological Commitment*, in *ICBO: International Conference on Biomedical Ontology*, B. Smith, Editor. 2009, National Center for Ontological Research: Buffalo NY. p. 55-58.
33. Ceusters, W., Smith, B. , *A Terminological and Ontological Analysis of the NCI Thesaurus. .* Methods of Information in Medicine 2005. **44**: p. 498-507.
34. Ceusters, W., Smith, B., *Ontology and Medical Terminology: Why Description Logics are not enough.* Proceedings of the Conference Towards an Electronic Patient Record (TEPR 2003), 2003.
35. Ceusters, W., Smith, B., Kumar A., Dhaen C., *Mistakes in Medical Ontologies: Where Do They Come From and How Can They Be Detected?*, in *Ontologies in Medicine. Proceedings of the Workshop on Medical Ontologies, Rome October 2003*, D.M. Pisanelli, Editor. 2004, IOS Press.
36. Smith, B., et al., *Towards a Reference Terminology for Ontology Research and Development in the Biomedical Domain*, in *KR-MED 2006, Biomedical Ontology in Action.* 2006: Baltimore MD, USA

37. IFOMIS. *Basic Formal Ontology*. 2009; Available from: http://www.ifomis.uni-saarland.de/bfo/.
38. Gangemi, A., et al., *Sweetening Ontologies with DOLCE* in *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management. Ontologies and the Semantic Web* 2002, Springer-Verlag: London, UK. p. 166-181.
39. Smith, B., et al., *Relations in biomedical ontologies.* Genome Biology, 2005. **6**(5): p. R46.
40. Smith, B., et al., *The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration.* Nature Biotechnology, 2007. **25**: p. 1251-1255.
41. Haase, P. and Y. Sure, *State-of-the-art on ontology evolution. SEKT deliverable D3.1.1.b*. 2004.
42. Tagger, B., *A literature review for the problem of biological data versioning. http://www.cs.ucl.ac.uk/staff/btagger/LitReview.pdf*. July 2005.
43. Yildiz, B., *Ontology Evolution and Versioning; The state of the art. , Asgaard-TR-2006-3*. October 2006.
44. Oliver, D. and Y. Shahar, *Change management of shared and local versions of health-care terminologies.* Methods Inf Med, 2000. **39**: p. 278-90.
45. Stojanovic, L., N. Stojanovic, and S. Handschuh, *Evolution of Metadata in Ontology-based Knowledge Management Systems*, in *Proceedings of Experience Management 2002*. 2003: Berlin.
46. Noy, N.F. and M.A. Musen, *PromptDiff: A Fixed-point Algorithm for Comparing Ontology Versions.*, in *Proc. of the 18th Nat'l Conf. on Artificial Intelligence (AAAI2002)*. 2000, AAAI Press: Edmonton, Alberta. p. 744-750.
47. College of American Pathologists, *SNOMED CT® Technical Reference Guide - January 2007 Release*. 2007.
48. Smith, B., W. Ceusters, and R. Temmerman, *Wüsteria*, in *Connecting Medical Informatics and Bio-Informatics. Medical Informatics Europe 2005*, R. Engelbrecht, et al., Editors. 2005, IOS Press: Amsterdam. p. 647-652.
49. Ceusters, W., et al., *Mistakes in medical ontologies: Where do they come from and how can they be detected?*, in *Ontologies in Medicine. Studies in Health Technology and Informatics*, D.M. Pisanelli, Editor. 2004, IOS Press: Amsterdam, The Netherlands. p. 145-164.
50. Cimino, J.J., *In Defense of the desiderata.* Journal of Biomedical Informatics, 2006. **39**(3): p. 299-306.
51. Solbrig, H.R. and C.G. Chute, *Concepts, Modeling and Confusion*, in *ICBO: International Conference on Biomedical Ontology*, B. Smith, Editor. 2009, National Center for Ontological Research: Buffalo NY. p. 123-126.
52. Rector, A., S. Brandt, and T. Schneider, *Getting the Foot out of the Pelvis: Modelling Problems affecting Use of SNOMED-CT Hierarchies in Practical Applications.* Journal of the Amercian Medical Informatics Association, 2011. **18**(4): p. 432-40.
53. Hogan, W.R., *What's in a 'is a' Link?*, in *ICBO: International Conference On Biomedical Ontology*, B. Smith, Editor. 2009, National Center for Ontological Research: Buffalo NY. p. 174.
54. Ceusters, W., K.A. Spackman, and B. Smith. *Would SNOMED CT benefit from Realism-Based Ontology Evolution?* in *American Medical Informatics Association 2007 Annual Symposium Proceedings, Biomedical and Health Informatics: From Foundations to Applications to Policy*. 2007. Chicago IL: American Medical Informatics Association.
55. Ceusters, W. and B. Smith, *A Unified Framework for Biomedical Terminologies and Ontologies*, in *Proceedings of the 13th World Congress on Medical and Health Informatics (Medinfo 2010), Cape Town, South Africa, 12-15 September 2010*, C. Safran, H. Marin, and S. Reti, Editors. 2010, IOS Press: Amsterdam. p. 1050-1054.

56. Smith, B., *From Concepts to Clinical Reality: An Essay on the Benchmarking of Biomedical Terminologies.* Journal of Biomedical Informatics, 2006. **39**(3): p. 288-298.
57. Smith, B. and W. Ceusters, *Ontological Realism as a Methodology for Coordinated Evolution of Scientific Ontologies.* Applied Ontology, 2010. **5**(3-4): p. 139-188.
58. Ceusters, W., et al., *Ontology-based error detection in SNOMED-CT®.* in *MEDINFO 2004*, M. Fieschi, E. Coiera, and Y.-C.J. Li, Editors. 2004, IOS Press: Amsterdam, The Netherlands. p. 482-486.
59. Johansson, I., *Bioinformatics and Biological Reality.* Journal of Biomedical Informatics, 2006. **39**(3): p. 274-287.
60. Ceusters, W. and B. Smith, *Ontology and Medical Terminology: why Descriptions Logics are not enough.*, in *Towards an Electronic Patient Record (TEPR 2003)*. 2003: San Antonio.
61. Rosse, C., et al., *A strategy for improving and integrating biomedical ontologies*, in *Biomedical and Health Informatics: From Foundations to Applications to Policy; Proceedings of the 2005 AMIA Annual Symposium*, C.P. Friedman, J. Ash, and P. Tarczy-Hornoch, Editors. 2005, American Medical Informatics Association: Washington DC. p. 639-43.
62. Smith, B., *Fiat Objects.* Topoi, 2001. **20**(2): p. 131-148.
63. Scheuermann, R.H., W. Ceusters, and B. Smith, *Toward an Ontological Treatment of Disease and Diagnosis*, in *Proceedings of the 2009 AMIA Summit on Translational Bioinformatics, San Francisco, California, March 15-17, 2009*. 2009, American Medical Informatics Association. p. 116-120.
64. Bray, M. *Object-Oriented Analysis*. Software Technology Roadmap 1997 January 11, 2007 [cited 2008 May 4]; Available from: http://www.sei.cmu.edu/str/descriptions/ooanalysis.html.
65. Sager, J.C., *A Practical Course in Terminology Processing*. 1990, Amsterdam: John Benjamins Publishing Company. 252.
66. Bittner, T. and B. Smith, *A Theory of Granular Partitions*, in *Foundations of Geographic Information Science*, M. Duckham, M.F. Goodchild, and M.F. Worboy, Editors. 2003, Taylor & Francis Books: London. p. 117-151.
67. Cimino, J.J., *Desiderata for controlled medical vocabularies in the twenty-first century.* Methods of Information in Medicine, 1998. **37**(4-5): p. 394-403.
68. Min, H., et al., *Auditing as Part of the Terminology Design Life Cycle.* Journal of the American Medical Informatics Association, 2006. **13**: p. 676-690.
69. Rosenbloom, S.T., et al., *A Model for Evaluating Interface Terminologies.* Journal of the American Medical Informatics Association, 2008. **15**: p. 65-76.
70. Baud, R., et al., *Reconciliation of Ontology and Terminology to cope with Linguistics*, in *Proceedings of MEDINFO 2007, Brisbane, Australia, August 2007*, K. Kuhn, J. Warren, and T. Leong, Editors. 2007, Ios Press: Amsterdam. p. 796-801.
71. Bodenreider, O., B. Smith, and A. Burgun, *The ontology-epistemology divide: A case study in medical terminology*, in *Proceedings of the Third International Conference on Formal Ontology in Information Systems (FOIS 2004)*, A.C. Varzi and L. Vieu, Editors. 2004, IOS Press: Amsterdam. p. 185-195.
72. Hayamizu, T., et al., *Evaluation of Vocabulary Review Criteria: Final Report to V/CDE WS*. 2006.
73. Hayamizu, T., et al., *Evaluation of the Gene Ontology (GO) using the proposed V/CDE WS Vocabulary Review Criteria*. 2007.
74. Hartung, M., T. Kirsten, and E. Rahm, *Analyzing the Evolution of Life Science Ontologies and Mappings*, in *Leipzig Bioinformatics Working Paper*. 2008, Interdisciplinary Centre for Bioinformatics: Leipzig.

75. Smith, B., et al., *On carcinomas and other pathological entities.* Comparative and Functional Genomics, 2005. **6**(7-8): p. 379 - 387.
76. Smith, B., J. Williams, and S. Schulze-Kremer, *The Ontology of the Gene Ontology*, in *AMIA Annual Symposium Proceedings*. 2003. p. 609-613.
77. Kumar, A. and B. Smith, *The Unified Medical Language System and the Gene Ontology*, in *KI2003: Advances in Artificial Intelligence (Lecture Notes in Artificial Intelligence 2821)*. 2003. p. 135-148.
78. Smith, B., J. Köhler, and A. Kumar, *On the application of formal principles to life science data: A case study in the Gene Ontology*, in *Data Integration in the Life Sciences: First International Workshop, DILS 2004, Leipzig, Germany, March 25-26, 2004, proceedings (Lecture Notes in Bioinformatics , Vol. 2994)*, E. Rahm, Editor. 2004, Springer: Heidelberg. p. 79-94.
79. Smith, B. and A. Kumar, *On controlled vocabularies in bioinformatics: A case study in the Gene Ontology.* BioSilico: Drug Discovery Today, 2004. **2**: p. 246-252.
80. Simon, J., J.M. Fielding, and B. Smith, *Using Philosophy to Improve the Coherence and Interoperability of Applications Ontologies*, in *First Workshop on Philosophy and Informatics; DFKI, Cologne, Germany.*, B. Büchel, B. Klein, and T. Roth-Berghofer, Editors. 2004. p. 65 - 72.
81. Köhler, J., et al., *Quality control for terms and definitions in ontologies and taxonomies.* BMC Bioinformatics, 2006. **7**: p. 212-220.
82. Ceusters, W., *Towards A Realism-Based Metric for Quality Assurance in Ontology Matching*, in *Formal Ontology in Information Systems*, B. Bennett and C. Fellbaum, Editors. 2006, IOS Press: Amsterdam. p. 321-332.
83. Donnelly K, *SNOMED CT: The Advanced Terminology and Coding System for eHealth*, in *Studies in Health Technology and Informatics - Medical and Care Compunetics 3. Vol 121*, Bos L, et al., Editors. 2006, IOS Press: Amsterdam. p. 279 - 290.
84. International Health Terminology Standards Development Organisation, *SNOMED Clinical Terms® User Guide - January 2010 International Release (US English)*. 2010.
85. The International Health Terminology Standards Development Organisation, *SNOMED CT® Technical Reference Guide – July 2009 International Release*. 2009.
86. Spackman, K.A., K.E. Campbell, and R.A. Côté, *SNOMED RT: A reference terminology for health care*, in *The Emergence of Internetable Health Care: Systems that Really Work. Proceedings of the 1997 AMIA Annual Symposium*, D.R. Masys, Editor. 1997, Hanley & Belfus Inc: Philadelphia. p. 640-644.
87. Wasserman, H., Wang, J. , *An applied evaluation of SNOMED CT as a clinical vocabulary for the computerized diagnosis and problem list. .* AMIA Annu Symp Proc. , 2003: p. 699-703.
88. International Health Terminology Standards Development Organisation, *SNOMED CT® Technical Reference Guide - January 2010 International Release (US English)*. 2010.
89. Sampalli, T., et al., *An evaluation of SNOMED CT® in the domain of complex chronic conditions.* International Journal of Integrated Care, 2010. **10**: p. e038.
90. Park, H., et al., *Evaluation of the content coverage of SNOMED-CT to represent ICNP Version 1 catalogues.* Stud Health Technol Inform., 2009. **146**: p. 303-307.
91. J.E. Andrews, R.L. Richesson, and J. Krischer, *Variation of SNOMED CT coding of clinical research concepts among coding experts.* Journal of the American Medical Informatics Association, 2007. **14**(4): p. 497-506.
92. M.F. Chiang, et al., *Reliability of SNOMED-CT coding by three physicians using two terminology browsers*, in *AMIA 2006 Symposium Proceedings*. 2006. p. 131-135.
93. Wei, D., et al., *Auditing SNOMED Relationships Using a Converse Abstraction Network*, in *AMIA Annu Symp Proc. 2009*. 2009. p. 685-689.

94. Wade, G. and T. Rosenbloom, *The impact of SNOMED CT revisions on a mapped interface terminology: Terminology development and implementation issues.* Journal of Biomedical Informatics, 2009. **42**(3): p. 490–493.
95. Sánchez, D., M. Batet, and D. Iserna, *Ontology-based information content computation* Knowledge-Based Systems, 2011. **24**(2): p. 297-303.
96. Van Buggenhout, C. and W. Ceusters, *A novel view on information content of concepts in a large ontology and a view on the structure and the quality of the ontology.* International Journal of Medical Informatics, 2005. **74**(2-4): p. 125-132.
97. Hogan, W. and V. Slee, *Measuring the Information Gain of Diagnosis vs. Diagnosis Category Coding.*, in *AMIA Annu Symp Proc. 2010* 2010. p. 306-10.
98. International Health Terminology Standards Development Organisation, *SNOMED Clinical Terms® Technology Preview Guide - January 2010 International Release (US English)*. 2010.
99. International Health Terminology Standards Development Organisation, *SNOMED CT® Release Format 2.0 Reference Set Specifications - Version 1.0a (January 2011 International Release)*. 2011.
100. International Health Terminology Standards Development Organisation, *SNOMED Clinical Terms® Release Format 2.0 Data Structures Specification - Version 1.0a (January 2011 International Release)*. 2011.
101. International Health Terminology Standards Development Organisation, *SNOMED CT® Release Format 2.0 Guide for Updating from RF1 to RF2 - Version1.0a (January 2011 International Release)*. 2011.
102. Mougin, F., O. Bodenreider, and A. Burgun, *Looking for Anemia (and Other Disorders) in SNOMED CT: Comparison of Three Approaches and Practical Implications.* AMIA Annual Symposium Proceedings, 2010: p. 527-531.
103. Lord, P. and R. Stevens, *Adding a Little Reality to Building Ontologies for Biology.* Plos ONE, 2010. **5**(9): p. e12258.