Strategies for Referent Tracking in Electronic Health Records

Werner Ceusters (1), Barry Smith (2)

(1) European Centre for Ontological Research, Saarbrücken, Germany

(2) Institute for Formal Ontology and Medical Information Science, Saarbrücken, Germany

Department of Philosophy, University at Buffalo, NY, USA

Contact:

    Dr. Werner Ceusters

    European Centre for Ontological Research

    Universität des Saarlandes

    Postfach 151150

    D-66041 Saarbrücken

    Germany

    Email: Werner.Ceusters@ecor.uni-saarland.de

    Fax: +49 (0)681-302-64772

Abstract

The goal of referent tracking is to create an ever-growing pool of data relating to concrete entities in reality. In the context of Electronic Healthcare Records (EHRs) the relevant concrete entities are not only particular patients but also their body parts, diseases, therapies, lesions, and so forth insofar as these are relevant to their diagnosis and treatment. Within a referent tracking system, all such entities are referred to explicitly, something which cannot be achieved when familiar concept-based systems are used in what is called "clinical coding". In this paper we describe the components of a referent tracking system in an informal way and we outline the procedures that would have to be followed by healthcare personnel in using such a system. We argue that the referent tracking paradigm can be introduced with only minor – though nevertheless ontologically important – technical changes to existing EHR infrastructures, but that it requires primarily a different mindset from that which has prevailed hitherto.

1 Introduction

Electronic health records (EHRs) consist primarily of descriptions about a patient's medical condition, the treatments administered and the outcomes obtained. These descriptions are about concrete entities in reality: for example about the particular pain that the particular patient John experienced in his chest on this specific day; or about the particular pacemaker – with this specific serial number assigned to it by its manufacturer – that was implanted in John during the particular surgical procedure that started at a certain precise moment in time on a certain day and took three hours to complete.

The descriptions contained in current EHRs contain very few explicit references to such entities. This lack of explicit reference is usually a minor problem for human interpreters, but it makes an accurate understanding of EHR data nearly impossible for machines. This is because reference resolution in running text (still the most common format for descriptions in EHRs) is one of the hardest problems in natural language understanding [1]. But even those EHR systems which incorporate data in more structured formats, for example by resorting to controlled vocabularies, terminologies or even ontologies, are in no better shape in this respect. This is because the terms or codes contained in the latter are used simply as an alternative to what would otherwise have been registered by means of general terms in natural language. By picking a code from such a system and then registering that code in an EHR, one refers generically to *some* instance of the class represented by the code. It is still left at best only partially specified which particular instance is intended in concrete reality.

This has some obvious consequences. When a patient suffers from the same type of disease and exhibits the same kinds of symptoms on two successive occasions, then the descriptions of these conditions using codes from a terminology will be identical. When another patient suffers from the same type of disease and exhibits similar symptoms in his

turn, then the resulting descriptions will also be identical to those relating to the first patient.

Certainly some of the associated references to specific persons, dates and places, will be different but these are not always sufficient to assess whether the same (i.e. numerically identical) or different (but qualitatively similar) entities are being referred to. Is the closed atlas fracture referred to by using the SNOMED-CT concept code 269063003 in John's EHR at time t1 the very same fracture as that which is referred to by means of the same code at time t2? If t1 and t2 are not far apart in time, then it is probable that the answer to this question is 'yes'; if there is a large gap, then poor John probably broke his neck twice. But who will be able to tell the difference for sure? One can be certain that, under normal circumstances, a fracture is not healed within a time frame of 2 weeks, and also that it should be healed after 2 months. But there is a period in between during which assertions as to whether or not a fracture is healed cannot safely be made. And what if it is not a fracture that is being referred to, but a disease such as diabetes, or a malformation such as a webbed neck? In such cases, as is obvious (again) to human beings but not to the machine, the very same disorder or malformation may be referred to at successive times over the whole of the patient's lifetime. Of course a given record may contain descriptions about when a specific disease was considered to have been cured (for example a wound to have been healed), so that if a new entry about the same type of disease is made thereafter, then it will be possible to infer that it pertains to a new instance. Unfortunately, however, patients tend to visit their physicians primarily when they are ill, and not when they are cured, so that EHRs contain far more traces of information pertaining to the initial phases of a disease than to its successful treatment. And even where information pertaining to the termination of a disease has been registered, it will still often be very hard for software applications to understand this information properly and to make the appropriate inferences.

Similar considerations can be made regarding descriptions in the EHRs of two different patients. Note, first of all, that one cannot assume that if the same code is used in two such records, then they refer to two distinct entities. Temperatures and blood pressures, as well as disorders depend uniquely on their particular bearers, and so they necessarily constitute different things when referred to in the EHRs of different patients. But SNOMED-CT has codes for places such as "swimming pool", or relatives such as "father". Obviously, two different patients may or may not have different fathers, may or may not have visited the same swimming pool from which they then obtained (different) nasty viral warts. They may or may not have received transfusions of blood from the same donor, thereby becoming HIV-positive, and so forth. In these cases it is important, at least from an epidemiological perspective, that inferences can be reliably made as to whether it is the same or different entities in reality which are being referred to.

Only a few codes in systems such as SNOMED-CT refer always to the same particular entity. This is so for instance in the case of codes referring to countries such as *Belgium*, the *United States of America*, and so forth, or to specific protocols, social security plans or named drug regimes. But this facility, which is a strength from the point of view to be defended in this paper, is in fact to be assessed as an ontological mistake from the perspective of SNOMED-CT itself. This is because the latter system – like so many others – is not able to distinguish between terms, such as "*country*", used in a general sense, and terms, such as "Belgium", that invariably refer to some specific country.

Finally, one also cannot assume that if two different codes are used in an EHR then they refer to different entities. Possible reasons for this are manifold. It may be that the most specific or detailed code is not always used when the same entity is referred to on successive occasions. A *colon polyp*, for example, when re-examined in the course of a follow-up visit where no change has been observed, might simply be referred to as

intestinal polyp, or just *polyp*, and thus associated on successive occasions with different codes, even though the physician was fully aware that it was the same instance (the same polyp) that was being referred to.

It might also be that the polyp has become malignant, and then it will be assigned the code for *malignant neoplasm of colon*. Clearly, the entity, i.e. the polyp, underwent changes. But it is still the same entity: its identity did not change. (In a similar way, persons undergo changes, grow older, lose hair, and so forth, but still remain the same entities, preserving their identity over time.) This preservation of identity in the presence of phenotypic difference is important, for instance in matters of prevention. As an example, there was a time when it was not yet common knowledge that particular polyps may deteriorate and become cancerous. Different surgeons may indeed have observed this process in particular patients; but they were not aware that by reporting what they had observed they would be contributing valuable scientific information. Suppose, now, that a statistical study targets patients suffering from intestinal polyps, the patients being selected on the basis of the presence of the code for *intestinal polyp* in their EHRs at some time t1. From the records of these patients one can also extract all disorder codes that are registered at times t later than t1. Imagine, however, that in a statistically significant portion of the latter, the code for *malignant neoplasm of colon* has been registered. As a result, taken over all records, one may conclude that the presence of an intestinal polyp is a risk factor for the appearance of a malignant tumor at some later time. But if, as is the case under current EHR regimes, one would not know that the very same (i.e. numerically identical) polyps had turned malignant over time rather than a second polyp discovered elsewhere in the colon, then one would also not be led to postulate the removal of those initial polyps as the best prevention for malignancy.

A third reason why different general codes from a coding system may not automatically be taken to refer to different particular instances turns on the fact that a code may not suffice to describe a given instance appropriately. If, for example, one wants to use SNOMED-CT (v0301) to code a *closed pedicular fracture of the fifth cervical vertebra* then a single code for this is not available; to give a faithful description one must use instead both the codes for '*fracture of pedicle of cervical vertebra*' and '*closed fracture of fifth cervical vertebra*'. If, however, these codes are not entered in the EHR in such a way that it is clear that they refer to the same entity, then their presence might be taken incorrectly to refer to two different fractures.

Similar mistakes may arise also in virtue of the fact the same class is represented twice by means of two different codes in a single coding system (e.g. in SNOMED-CT v0301, 41191003: '*open fracture of head of femur*' and 208539002: '*open fracture head, femur*') [2].

As an intermediate conclusion we can therefore state that, even where coding systems and terminologies (and, as we will argue later, the majority of those systems called 'ontologies') provide rich vocabularies to describe in general terms entities that exist in reality, they have as yet been associated with no mechanism to express *what those descriptions are about*, i.e. what entities in reality they refer to. This is not, be it noted, a criticism of existing coding systems. The latter were not, after all, designed to have such a mechanism in place. But it is our claim that such a mechanism is indispensable at the interface where coding systems meet the clinical record if we are to gain maximal advantage from coding efforts and from so-called formal representations and descriptions in EHR systems.

2 Referent tracking

In [3], *referent tracking* was introduced as a paradigm under which it will become possible to refer explicitly to all of the concrete individual entities relevant to the accurate description of each patient's condition, therapies, and outcomes through the assignment of unique identifiers. Such an identifier is called a *IUI*, for *Instance Unique Identifier*. This means that not only does the patient receive a IUI, but so also does the particular fracture he is suffering from, the particular bone that is fractured, and even, if the clinician finds this important, the particular pain the patient is experiencing in a certain time period or the particular document in which the pain is first recorded. Note that IUIs refer to the real entities themselves out there in reality, and not to data about these entities. They are thus distinct for example from the Life Science Identifiers (LSIDs) [4], which are data that refer to data. IUIs, being data referring to real entities, are the means whereby the constellations of particular entities in reality that are relevant to clinical care can become represented in an EHR in the same direct way in which the corresponding classes are already represented by means of clinical coding systems.

Thus IUIs are also not *the entities themselves*. This might seem obvious, but use-mention confusions of this kind in which an entity in reality and its digital representation are confounded together, are abundantly present in the literature on knowledge representation in general and on concept-based terminology systems in particular [5].

In the context of this paper we will use expressions of the form 'IUI-*uvwx…*' , where *u*, *v*, ... will be substituted by numerical digits, to denote the identifiers themselves. We will then be able to discuss, for example, the length of a IUI, or the font in which it is written. Expressions of the form '*#X*', in contrast, where '*X*' stands proxy for a IUI, will denote the individual entity in the real world. This will allow us to write for instance that IUI-3006 *refers* to the particular patient John, while #IUI-3006 *is* that particular patient. Of course,

these conventions are for expository purposes only, and they will not apply to the IUIs that will be used in the actual EHR systems of the future.

The referent tracking paradigm distinguishes between *IUI assignment*, which is possible only in relation to entities that exist or have existed in the past, and *IUI reservation*, which is a provision made for entities, such as an X-ray ordered for tomorrow, that are expected to come into existence in the future. The order itself can have a IUI assigned already today, but for the resultant image, one can only *reserve* a IUI at the time of ordering.

Note, too, that IUI assignment or reservation does not by itself entail any assertion as to the class (or, since we take a position grounded in realism as a philosophical theory, the *universal* [6]) of which the particular entity in question is an instance. Thus we might assign a IUI to a syndrome on behalf of a given patient before we have any clear idea what sort of syndrome it is with which we are dealing. This facility, too, has no analogue in code-based EHR systems as currently constituted.

In this paper we explore ways in which the referent tracking paradigm can be implemented in the healthcare environment. Our hypothesis is that, once the right infrastructure is in place, the burden on clinicians and nurses (or on whomever the task of registering patient data is assigned) will be not significantly greater than under existing strategies for data entry – but that the direct benefits, in terms of semantic interoperability of computer systems, and the indirect benefits in terms of patient management, epidemiology and disease control, cost containment and the advance of science in the domain of biomedicine, can be enormous.

3   Overall architecture of a referent tracking system

The purpose of a referent tracking system (RTS) is, as its name suggests, to keep track of *referents*. Referents are entities that exist in reality, i.e. in the real world that surrounds us.

Most referents are *particulars*, examples being: a copy of the journal in which this paper is published, its authors, those of its readers who find its ideas appealing, as well as those readers who doubt the existence of physical reality (and hence their own existence). Other referents are *universals*, examples being *journal*, *paper*, *person* and so forth. According to the philosophy of realism, universals are as real as particulars (where, from the perspective of nominalism, the existence of universals is denied). Referent tracking deals primarily with the tracking of particulars (so that even nominalists may be able to take advantage of its potentialities), by collecting information about particulars in an RTS. We assume without further consideration that the tracking of universals is (or will be) taken care of in adequate ontologies. In this paper, we focus on the tracking of particular referents in the context of maintaining EHRs, but the paradigm is clearly applicable in other contexts as well.

An RTS will contain information about particulars, and the users who enter this information will be required to use IUIs in order to assure explicit reference to the particulars about which the information is provided. Thus the information that is currently captured in the EHR by means of sentences such as: "this patient has a left elbow fracture", would in the future be conveyed by means of descriptions such as "#IUI-5089 is located in #IUI-7120", together with associated information to the effect that "IUI-7120" refers to the patient under scrutiny, and "IUI-5089" to a particular fracture in patient #IUI-7120 (and not to some similar left elbow fracture from which he suffered earlier). The RTS must correspondingly contain information relating particulars to universals, such as "#IUI-5089 is a fracture" (where 'fracture' might be replaced by a unique identifier pointing to the representation of the universal *fracture* in an ontology). Of course, EHR systems that endorse the referent tracking paradigm should have mechanisms to capture such information in an easy and intuitive way, including mechanisms to translate generic statements into the intended concrete form, which may itself be operative primarily only behind the scenes, so that the IUIs themselves remain invisible to the human user. One could indeed imagine that natural

language processing software will one day be in a position to replace in a reliable fashion the generic terms in a sentence with corresponding IUIs for the particulars denoted by them, with manual support in flagged problematic cases. This is what users already expect from EHR systems in which data are entered by resorting to general codes or terms from coding systems.

At least the following requirements have to be addressed if the paradigm of referent tracking is to be brought into existence:

- a mechanism for generating IUIs that are guaranteed to be unique strings;

- a procedure for deciding what particulars should receive IUIs;

- protocols for determining whether or not a particular has already been assigned a IUI (except for some exceptional configurations that are beyond the scope of this paper, each particular should receive maximally one IUI);

- practices governing the use of IUIs in the EHR (issues concerning the syntax and semantics of statements containing IUIs);

- methods for determining the truth values of propositions that are expressed through descriptions in which IUIs are used;

- methods for correcting errors in the assignment of IUIs, and for investigating the results of assigning alternative IUIs to problematic cases;

- methods for taking account of changes in the reality to which IUIs get assigned, for example when particulars merge or split.

An RTS can be set up in isolation, for instance within a single general practitioner's surgery or within the context of a hospital. The referent tracking paradigm will however serve its purpose optimally only when it is used in a distributed, collaborative environment. One and the same patient is often cared for by a variety of healthcare providers, many of them

working in different settings, and each of these settings uses its own information system. These systems contain different data, but the majority of these data provide information about the same particulars. Under the current state of affairs, it is very hard, if not impossible, to query these data in such a way that, for a given particular, all information available can be retrieved. With the right sort of distributed RTS, such retrieval becomes a trivial matter.

The system we have in mind should offer at least three services.

The first is to generate unique identifiers to be used as IUIs.

The second we shall refer to as the *IUI-repository* (although technically it does not need to be implemented as a single monolithic entity). This is the most crucial service to be provided by an RTS and consists in keeping track of the identifiers assigned to already existing entities or reserved for entities that are expected to come into existence in the future. It will do this in such a way that each IUI represents exactly one particular, and that no particular is referred to by more than one IUI. These two requirements are not easy to fulfil, since both depend on the ability and willingness of users to provide accurate information. This, however, is not different in principle from the problem facing any other type of information system whose users are called upon to provide information of a non-trivial and occasionally sensitive sort.

The third service, here called the *referent-tracking database* (RTDB), should provide access to all the information that has been entered in given EHRs about the particulars referred to in the IUI-repository for those users authorized to access the information in question. Where the IUI repository is an inventory of what concrete entities have been stated to exist, and, consequently, what IDs to use if one wants to refer to them, the RTDB is an inventory (or index) of descriptions reflecting assertions made concerning the features and interrelations of these entities and the ways they change in the course of time. The

RTDB, too, does not need to be set up as a single central database (in which case it would be some sort of data warehouse); rather, it should rely on the LSID (or some similar) paradigm, which means that EHRs that participate in a given referent tracking initiative would inform the associated RTDB automatically about the availability of information related to given particulars. The RTDB itself can then serve as an *index,* pointing to this data, rather than as a container for this data itself.

The primary role of the RTDB is thus to keep track of the features of given particulars and of their relationships to other particulars as they change through time, AND of the assertions that have been made thereof. It has an important role also in helping users to determine whether particulars they encounter for the first time have been registered already in the IUI-repository, or whether a new IUI must be assigned for use in new descriptions. For sure, this places some additional burden on the person that has to enter information; but time perceived as being lost at this stage will be recovered when searching for information thereafter.

4   The generation of IUIs

Generating strings that are guaranteed to be unique is not a major problem. Several schemas are already in use, such as Microsoft's Globally Unique Identifier paradigm (GUID), which implements UUIDs (Universally Unique IDs) as defined by the Open Software Foundation in the specification of its Distributed Computing Environment [7]. The advantage of GUID paradigm is that unique identifiers can be generated easily on any machine with a network card, without the need to resort to a central authority to guarantee uniqueness.

UUIDs have recently been standardized through ISO/IEC 9834-8:2004, which specifies format and generation rules that enable users to produce 128-bit identifiers every 100 nanoseconds which are either guaranteed to be or have a high probability of being globally

unique [8]. The standard also specifies the procedures for the operation of a Web-based Registration Authority for UUIDs. Although some older versions of UUID generating algorithms may produce IDs that contain meaningful information (such as the MAC address of the machine used to generate the ID), recent versions no longer exhibit this behaviour.

UUIDs have hitherto been used only for unique identification of software components such as the pop-up windows generated in the course of a program's execution; but there is no reason why they should not be used also to identify particulars in the real world outside the machine. In the specific case of health related particulars, ethical, safety and security considerations might require *certification* of their uniqueness. To that end, the medical community may want to install an authority that not only registers IUIs, but also certifies the uniqueness of the strings to be used within a given IUI-repository and also guarantees that the assignments claimed to have been made by given authors were indeed made by those authors. This can be compared to the services offered by trusted third parties in private key management for asymmetrical encryption purposes [9]. Moreover, central registration in some form will in any case be necessary if we are to fulfil our requirement explained in the next section, to the effect that no particular be assigned more than one IUI.

Note that the IUIs themselves do not carry any information as to what particular they refer to. They are simply "meaningless" strings. As explained above, information about what the IUIs actually stand for is to be found in the RTDB.

5   To assign or not to assign

IUIs should be *assigned* exclusively to entities that exist, or have existed in the past. Although each particular is, by definition, a unique entity, particulars are standardly not such as to have a uniquely identifying label already attached. Newborns in the USA are

assigned Social Security numbers only as a result of an application procedure, and IUI assignment in general is typically an act carried out by the first cognitive agent who feels the need to acknowledge the existence of a particular it has observed (or has information about that is analogous in its reliability to that which is gained through observation). In the healthcare environment the assigning entity will typically be a person (clinician, nurse, patient); but it might also be a device, as for example when radiographic films are manufactured in such a way that each film is automatically tagged with a IUI; analysis software that operates on digital images might automatically assign IUIs to specific configurations found therein, such as fracture lines or coin lesions.

From a logical perspective, each act of IUI assignment rests on a complex belief (a presupposition) on the part of a cognitive agent involved to the effect that the following three propositions are true:

1. this particular (here before me now) exists;

2. this particular has not yet been the object of a IUI assignment;

3. the string that functions as IUI for this particular has not been used thus far for any other entity.

The agent in question acquires this believe on the basis of prior consideration of the particular on question and his knowledge of the workings of the pertinent referent tracking system in light of the fulfilment of the criteria described hereafter.

5.1   Criteria for IUI assignment

The first criterion which needs to be fulfilled before a IUI can be assigned is: "*Does what I want to assign a IUI to exist?*". Only if the answer to this question is 'yes' is assignment allowed. Where the particulars in question are a patient's body parts or objective signs such as skin lesions, checking existence is trivial. Things or events for which we have good reason for believing that they will exist or occur in the future such as the subjects of orders,

cannot be assigned a IUI because they do not, as yet, exist. However, it is acceptable to *reserve* a IUI on their behalf. Such reservation then entails certain problems of its own which will not however be dealt with in this paper. Other cases – such as a patient's subjective symptoms – will be more problematic still. Certainly when a patient complains about a headache, then his making this complaint is a particular utterance event to which a IUI can unproblematically be assigned. But that event is of course distinct from the particular which is the headache itself.

As stated already above, consideration of the exact nature or type of a particular do not play a role at this stage of checking its existence. That we do not know all that there is to know about a given particular is an epistemological issue, which has no consequences for the ontological status or nature of the particular itself. (If we are realists about the future, so that the existence of objects in the future is analogous to that of objects in the past and present, then the difference between assigning and reserving a IUI may be a special case of this epistemological issue.) A patient (#IUI-001) might be stung by a particular bee from a particular swarm consisting of bees of a sort to which he is allergic. A faithful registration of this event requires a IUI for that particular bee although it might be very hard to identify that bee. This can however be achieved by assigning (e.g.) IUI-2345 to the swarm, and then assigning IUI-567 to *that bee (whichever one it is) which is a member of #IUI-2345 and which stung #IUI-001 at time t1*. The same strategy may be used to refer to one or a group of severely inflamed acne pustules on a patient's face, or to the particular pain attack (from a series) that finally made the patient decide to consult a physician.

The second criterion is fulfilled when it is established that the particular to which one wants to assign a IUI is not the same as some particular whose existence has already been ascertained (whether or not it has already received a IUI). The classical example used in the philosophy of language is the planet Venus [10]. For a long time, people believed in the

existence of two distinct particulars, one visible in the firmament in the morning, the other in the evening. In reality, however, it was the very same planet that was being perceived on both sets of occasions. The same situation may be encountered in healthcare, for instance when, in the course of a patient's medical history, a disease is assumed to be the cause of certain manifestations which then disappear, a second disease is a few years later assumed to be the cause of certain other manifestations which also disappear, until it is finally established that one and the same (i.e. numerically identical) underlying disease, for example multiple sclerosis, had been causing all these manifestations from the very beginning. Another example is that of several different patients who have been bitten by the same dog on a series of successive occasions.

From these examples, it should be clear that the referent tracking paradigm has to include a facility for dealing with mistakes. (This, too, will not be dealt with further here, but some initial remarks can be found in [3].)

The third criterion, usually closely related to the previous one, concerns whether or not the particular whose uniqueness has been determined has already been assigned a IUI, since our paradigm insists – drawing on familiar arguments which were used to justify the introduction of unique patient identifiers [11] – on at most one IUI per particular. Otherwise, information in different (or even in the same) EHRs might not be interpretable as pertaining to the same entity. This is not to say that other, temporary IDs might not be produced for pragmatic reasons, for example when the referent tracking system is off-line, or when data has to be entered under extremely urgent circumstances and one does not have the time to perform an adequate search to reliably establish that all criteria for IUI assignment have been met. The resultant IDs are not IUIs, however, and steps should be taken to ensure that they can always be replaced by IUIs in course of time.

The fourth criterion concerns whether a given particular entity is, from the point of view of clinical care, sufficiently salient to justify the assignment of its own IUI. Here we can mention a clear distinction between the entities that in everyday life are considered sufficiently important to receive their own unique ID (usually by means of giving them a *proper name*, rather than a number), for instance children, pets, yachts, and those that qualify for assignment in the context of healthcare. It is usually *continuants*, i.e. entities that preserve their identity through time (and thus are wholly present at every time during the course of their existence, even though they may gain or lose parts from one time to the next), that are named, and not processes or events. In healthcare, however, it may be equally important that particular processes such as giving injections, removing or transplanting organs, reducing fractures, and so forth should be uniquely identified. For it might well be that a specific act performed at time t1 leads to consequences different from those yielded by a second, exactly similar act at time t2, or that the two leads to same consequences, but with different medico-legal effects.

In principle, we defend the thesis that all entities to which one would standardly refer either individually or generically under the current rules and practices of clinical record-keeping should receive a IUI under the referent tracking paradigm. Since these rules and practices differ from one institution or physician to the next, IUI assignment will not everywhere be implemented in the same way or to the same extent. For EHR systems that are primarily built around notes in natural language and offer only minimal facilities for structured reporting, we would accept a less demanding policy. However, we advocate the principle that *if* a particular has already been assigned a IUI by somebody else, then even in the sparse documents produced under a regime of the latter sort, references to this particular should be associated with pointers to these IUIs so that the recorded data (or pointers to them) can be submitted to the RTDB. In this way, the future advances in natural language processing technology which we believe will be facilitated by the referent tracking

paradigm will become directly and immediately applicable also to the data contained in the given records.

5.2 Publishing IUI assignments

When, after due consideration, a particular has been identified as requiring a IUI, then a thus far unused alphanumeric string is generated by the unique-ID generator and an act of *assignment* is carried out (analogous to an act of baptism), which creates a IUI out of the generated string by attaching it to the particular in question [12]. Three factors can be distinguished as structural elements involved in such an assignment act:

1. generating the relevant alphanumeric string;

2. attaching it to the relevant object;

3. publishing/announcing this attachment.

The resulting IUIs will, together with certain further types of associated information, constitute the *IUI-repository*. The units deposited in this repository can be represented as ordered quadruples of the form:

$$A_i = \; < IUI_p, \; IUI_a, \; t_{ap}, \; c >$$

where IUI_p is the IUI of the particular in question, IUI_a is the IUI of the author of the assignment act, t_{ap} is a time-stamp indicating when the assignment was made, and c is an optional description of the particular in free text, or a pointer to an address where such a description may be found. In light of the need to resolve mistakes in IUI assignments, each such quadruple will need to be complemented with meta-data recording by whom and at what time they were made accessible in the system. These meta-data are ordered triples of the form:

$$D_i = <IUI_d, \; A_i, \; t_d>$$

where IUI_d is the IUI of the entity registering the IUI in the system, A_i is the information-unit in question, and t_d is a reference to the time the registration was carried out (which is also the time from which IUI_p can be used in descriptions concerning the particular in question).

Note that neither t_{ap} nor t_d should be taken to carry information about when the particular referred to by IUI_p started to exist nor about its continued existence. It can however be inferred that IUI_p did not start to exist at a time later than t_{ap}.

5.3 Management of assignments

Both the A_i and D_i should be stored in the IUI-repository in such a way that they can be accessed by software applications. The repository itself might be one centrally maintained database. Since its contents are data, however, they may also be addressed by means of the LSID paradigm. (Although 'LSID' is an abbreviation for *Life Science Identifiers*, the LSIDs are in fact more properly conceived not as identifiers but as addresses: they inform a software program where it can find data, rather than for purposes of identification of particulars.) It should be a requirement for all systems that are part of a referent tracking environment that they register all the A and D tuples that are created by their users in the IUI-repository. This is a necessary (but not a sufficient) condition for ensuring that no particular is given two different IUIs.

Ideally, the A_i tuple should be entered into the system as soon as an assignment act has taken place, and the corresponding D_i tuple as soon as possible thereafter (normally just a short time later). Clearly, measures should be implemented to prevent content being deliberately entered that is based on false information. We suggest that any computer system contributing to the referent tracking architecture should require users to log in in such a way as to ensure that the user's IUI is itself used to log the data he enters. This

brings also the advantage that when an A_i tuple is entered in which IUI_a is the same as the IUI of the user, then the corresponding D_i tuple can be automatically generated.

Additional rules for IUI assignment may be implemented also. One might require, for example, that the patient authorises those who are allowed to make IUI assignments for particulars that concern him (which may or may not correspond to the persons authorised to manage his EHR). This is similar to the '*need to know*' policy installed in those institutions where authorisation to use a hospital information system does not entail authorisation to access *all* its information. This requirement would also limit the (from the point of view of the patient) uncontrolled accumulation of information about his health. Another rule might state that when clinicians do not enter patient data directly into the system but use some form of transcription, then they may authorise specific transcribers to register IUI assignments on their behalf, and so forth.

The IUI-repository is there to allow queries of different sorts from different sorts of users, either for purposes of adding new A and D tuples, or in order to retrieve tuples already stored. It is not allowed to delete already existing content: neither the existence of a particular nor the act of IUI assignment (itself a particular in its own right) can be undone. Thus when errors in IUI-assignment are corrected, then the original assignment information will be preserved, albeit in a form which ensures that those who use the information are aware of its erroneous nature.

6   Using IUI's in EHR statements

Once a IUI is registered in the referent tracking environment by means of A and D tuples, it can be used in descriptions of relevant facts or hypotheses about a patient's medical condition, his treatment, risk factors, and so forth. Descriptions may be directly about the particular itself, but also about other particulars that stand in some relation to it. Thus, if

IUI-924 refers to patient #IUI-0067's temperature, some statement may assert that at time t1, #IUI-924 had the value 37.6° Celsius. Additional information in such a statement may inform us about who performed the measurement (#IUI-3456), using what instrument (#IUI-4109), under what room temperature, and so forth. Although in this example the statement made is not directly about #IUI-3456 or #IUI-4109, still it tells us indirectly as much about these particulars as it does about the patient's temperature at that specific time, and hence might provide useful management information: a device that is used a number of times might require maintenance; or one could detect statistical differences (such as calibration errors) in the measurement data obtained using similar devices or by specific persons.

What should be included in descriptions of any given particular is not something that is dictated by the referent tracking paradigm. Users may follow the recommendations for good clinical registration issued by relevant bodies, openEHR archetypes being just one example [13]. As with *A* content itself, and in line with recommendations pertaining to EHR standards, associated descriptions should also be registered as having a precise author and tagged with data stating who made the information available and at what time.

The format in which information can be entered in a specific EHR system depends on the facilities the EHR system offers. As already seen, many systems do not allow a *formal* notation for statements, but expect data to be entered rather by means of natural language. In the following paragraphs we first discuss some strategies for using the referent tracking paradigm in text-based systems before moving on to discuss ways in which data may be entered in a more formal manner. For both types of systems, we assume that the user is authorised to have access to the IUI-repository and that he is able to view information about given particulars that is available through the RTDB.

6.1 IUIs in text-based EHR systems

The problem of how to deal with references to particulars in text-based EHR systems is not significantly different from the problem of how to deal in such systems with codes from concept-based terminologies or classifications such as ICD-9 or SNOMED-CT.

In the worst case, a system foresees nothing at all in the way of coding data. Typically, such systems allow you to type in free text in fields labelled *complaints*, *symptoms*, *diagnosis*, and so forth. All the user can do in order to enter codes – or IUIs under our paradigm – is to type them into the same data entry field as he types the free text. For example, he might just write them at the end of each natural language statement, using some syntactic convention to separate them from the text itself. A suitable interface would then need to be able to inform the RTDB that the sentence in question contains information about the particulars referred to by the IUIs listed.

For example in:

Open left elbow fracture was reduced with pins in 1984 (IUI-5089, IUI-1002, IUI-4900),

IUI-5089 might refer to the fracture, IUI-1002 to the reduction and IUI-4900 to the pins, though this fact cannot be derived from the statement itself.

Theoretically, it would be possible to use a more elaborate syntax, such as that used in Cassandra-tagging along the lines proposed in [14]. The mentioned example would then be written (using indentation for better display) as:

((open elbow fracture)IUI-5089

{ (reduced)IUI-1002

{{[with] (pins)IUI-4900}

{{[in] (1984)}

}

)

Note that the individual phrases in the example above – '*open elbow fracture*', '*reduced*', … – do not give a uniquely identifying description of the particulars that are referred to by the IUIs that follow these phrases syntactically. For #IUI-5089 this is obvious: no elbow fracture is "just" an elbow fracture. It must be of a very precise elbow, namely either the left or the right elbow of some particular patient (the patient whose case is being described). But it might be that #IUI-5089 is already described elsewhere in more detail (detail that might be found by searching the RTDB), or that this detail is not known (for example if the registration is entered in the course of a patient anamnesis and the patient does not remember whether the fracture was on the left or on the right). Even if the phrase would have read '*open left elbow fracture*', then it still would not uniquely identify the fracture. An identifying description of the fracture would be obtainable only by resorting to IUI-1002 as well, and this only if #IUI-1002 is that very precise reduction event in which #IUI-5089 exclusively partook as the fracture was being reduced. It could be argued that uniquely identifying descriptions would go some way towards making IUIs redundant; that this is not the case, however, follows from the fact that their might be different descriptions that each identify a particular uniquely thereby giving various sorts of information about that particular, whereby each portion of information is not derivable from the others.

In this sense, the semantics of the registrations following the Cassandra syntax is different from that suggested by the original Cassandra tagging, a language which was proposed to

relate phrases in sentences to concepts in concept-based systems. The same example as above, but using SNOMED-CT instead of IUIs, might look like this:

$$(\qquad (open\ elbow\ fracture)302232001$$

$$\{\ (reduced)122469009$$

$$\{[with]\ (pins)77444004\}$$

$$\{[in]\ (1984)\}$$

$$\}$$

$$)$$

where *302232001* is the SNOMED-CT concept code for the concept with the fully specified name '*elbow fracture – open (disorder)*', *122469009* the code for '*reduction procedure (procedure)*' and *77444004* the code for '*bone pin, device (physical object)*'.

Of course, one cannot expect a clinician or nurse to enter patient data by means of a Cassandra-like syntax. Renderings of the types proposed above should rather be the outcome of subjecting free text statements to automatic natural-language analysis [15, 16]. An intermediate solution would be to allow the text editor used for entering natural language sentences in the EHR to incorporate hyperlinks through which the user could enter IUIs associated with the linked phrases. With a user-interface of this type, it would thereafter be possible to right-click on the hyperlinked phrases in order to launch a query to the RTDB that would then return all information related to the particular under scrutiny.

6.2   IUIs in formally structured statements

EHR systems incorporating record architectures such as those proposed by GEHR [17], OpenEHR [18] or CEN ENV 13606 [19], would be almost ideally suited to the referent

tracking paradigm. Although none of these architectures currently take particulars properly into account – they are all biased towards the concept-based paradigm – the modifications that would need to be made are minor. In the case of CEN ENV 13606, for example, it would require an additional *compound data type* to be defined in order to make it formally clear that the content of a particular *data item*, i.e. one of the architectural components defined by that standard, is a IUI, rather than a proposition. The modification we propose would then allow instance data to be exchanged between EHR systems that endorse the CEN ENV 13606 standard in such a way that formal reasoning about these data, including reasoning applied to data drawn from different systems, would become more reliable.

Particularly interesting from the point of view of the information they provide are descriptions stating who or what the particular under scrutiny actually *is*, for it is these which are most relevant for determining whether a particular for which IUI assignment is considered is already registered in the IUI-repository. A clue might already be given by the optional *c* terms in *A* tuples already entered for other particulars, which may take the form of descriptions such as "Werner Ceusters' nose", or "#IUI-0945's nose". Such formats however are rather obstacles for accurate interpretation by software programs. The strongest statements would be those that would enable an interpreter to point to the particular in question without the possibility of error. If all particulars would carry their IUIs with them, as it were indelibly attached, then the (universally applicable) statement "*#IUI-xyz is that particular which carries IUI-xyz*" would be all that would be needed to identify particulars unambiguously. Interestingly, hard- and software implementations exploiting RFID (Radio Frequency Identification) can be viewed as applying a slightly modified version of such statements, namely "*#IUI-xyz is the particular that produces IUI-xyz when probed by an appropriate of sensor*" [20, 21].

While statements of this kind provide identity criteria for particulars, they are not informative with respect to which universals the particulars instantiate, what *kind* of particulars they are. For this an ontology is required, which means a representation of whatever is the pertinent domain of reality which

> (1) reflects the universals instantiated by the particulars and the relations in that domain in such a way that there obtains a systematic correlation between reality and the representation itself,

> (2) is intelligible to a domain expert, and

> (3) is formalised in a way that allows it to support automatic information processing.

The coding systems in common use, which we shall refer to in what follows by means of the term "*concept-based systems*", do not meet these requirements. Systems that do conform to this definition are BFO [22], the OBO Relation Ontology [23], and the FMA [24]. Ontologies of this kind contain relationships between universals that are formulated in such a way that they can be used to describe also relationships between the corresponding particulars.

6.2.1 Relationships between particulars

The OBO Relation Ontology distinguishes (1) relations that obtain between particulars, (2) relations that obtain between universals, and (3) relations that obtain between particulars and universals. In this paper, we will use **bold** type to indicate relations of types (1) and (2), and *italic* to pick out relations of type (3). The former can be used to formalise descriptions in an EHR system which assert relationships between precisely those particulars that are relevant to the given patient, rather than between the corresponding general classes, and so can be much more narrowly targeted: #IUI-1921 (the first author's left testicle) is not just the left testicle of some instance of *human being*, but of the very precise particular #IUI-0945. Moreover, relationships such as parthood do indeed have distinct properties at the

particular and at the class levels [25]. Thus from the statement "#IUI-1921 **part_of** #IUI-0945 at time *t1*", one can infer that "#IUI-0945 **has_part** #IUI-1921 at time *t1*", while a similar conclusion does not hold at the class level: "left testicle *part_of* human being" (we here leave aside the fact that also other mammals have testicles) does not entail that "human being *has_part* left testicle" under the usual interpretation given to such a proposition, namely that for all human beings *h* there exists some left testicle *t* such that *h* **has_part** *t*. (There are humans who do not have a left testicle, most of them being female.)

Note that for relations that obtain between continuants such as #IUI-0945 and #IUI-1921 time may not be neglected. It might indeed be that at a time later than *t1*, the **has_part** relation between #IUI-0945 and #IUI-1921 no longer holds. Such considerations, too, do not arise at the level of relations between universals.

Descriptions which express relationships amongst particulars we will refer to as *PtoP* – **p**articular to **p**articular – descriptions. Here again we can distinguish a number of structural elements which are present in every case.

1. an authorized user observes one or more objects which have already been assigned IUIs in the RTS in hand,

2. the user recognizes or apprehends that these objects stand in a certain relation, which is represented in some ontology *o*

3. the user asserts that this relation obtains and publishes this assertion by entering corresponding data into the RTDB.

This data will then take the form of an ordered sextuple

$$R_i = <IUI_a, t_a, r, o, P, t_r>$$

where

- *IUI_a* is the IUI of the author asserting that the relationship referred to by *r* holds between the particulars referred to by the IUIs listed in *P*,

- *t_a* is a time-stamp indicating when the assertion was made,

- *r* is the designation in *o* of the relationship obtaining between the particulars referred to in *P,*

- *o* is the ID of the ontology from which *r* is taken,

- *P* is an ordered list of IUIs referring to the particulars between which *r* obtains, and

- *t_r* is a time-stamp representing the time at which the relationship was observed to obtain.

P contains as many IUIs as are required by the arity of the relation *r*. In most cases, *P* will be an ordered pair, which is such that *r* obtains between the particular represented by the first IUI and the one referred to by the second IUI. As with *A* tuples, *R* tuples must also be accompanied by corresponding *D* tuples capturing when the sextuple became available to the referent tracking system.

From the example used earlier we could then derive the following tuples:

<IUI-6231, 18/04/2005, **has-participant***, RO, <IUI-1002, IUI-5089>, 1984>*

<IUI-6231, 18/04/2005, **has-participant***, RO, <IUI-1002, IUI-4900>, 1984>*

where #IUI-6231 is the author of the statement, and the relationship **has-participant** is taken from the OBO Relation Ontology. Note that, for the sake of the example, resorting to *RO* only would result in information loss because that ontology has currently only two relationships to describe ways entities may partake in events, which is clearly insufficient. Using the resources of that ontology, it is not possible to distinguish between the role

played by #IUI-5089 (the fracture mentioned before) which is that of *theme* and the role of #IUI-4900 (the pins used) which is that of *instrument*, in relation to #IUI-1002 (the treatment) [26].

As in the case of *A* and *D* tuples introduced above, the format in which the *R* tuples discussed here are presented is that of an abstract syntax; it is not anticipated that such tuples should be entered in this form by end-users.

6.2.2   Relationships between particulars and classes

The second type of information that can be provided about a particular is what class within the context of an ontology it is an instance of at time *t1*. Here also time is relevant, since a particular, through its evolution, may cease to instantiate one class and start to instantiate another: #IUI-0945 changed from *foetus* to *newborn*, and from *child* to *adult*.

Descriptions of this type (which we will refer to as *PtoCL* entries - **p**articular to **cl**ass) can be represented by ordered tuples of the form:

$$CL_i = <IUI_a, t_a, inst, o, IUI_p, cl, t_r>$$

where

- IUI_a is the IUI of the author asserting that IUI_p *inst cl*,

- t_a is a time-stamp indicating when the assertion was made,

- *inst* is the designation in *o* of the relationship of instantiation,

- *o* is the ID of the ontology from which *inst* and *cl* are taken,

- IUI_p is the IUI referring to the particular whose *inst* relationship with *cl* is asserted,

- *cl* is the designation of the class in *o* with which IUI_p enjoys the *inst* relationship, and,

- t_r is a time-stamp representing the time at which the relationship was observed to obtain.

Note that it is necessary to specify from which ontology *inst* and *cl* are taken, and precisely which *inst* relationship if an ontology contains several variants [27]. Such specifications will not only ensure that the corresponding definitions can be accessed automatically, but also facilitate reasoning across ontologies that are interoperable with the ontology specified.

7   IUIs in relation to concept-based systems

In section 6.2 we required the designations of relationships and classes used in statements describing properties of particulars to be taken from ontologies rather than from concept-based systems. There are many reasons for this, including:

- the relationships between particulars and the corresponding concepts from such systems are often obscure [5],

- the relationships themselves are inadequately defined [23],

- there is an inconsistent reading of statements with respect to existential or universal quantification [28],

- ontology and epistemology are mixed together in inappropriate ways [29].

We do not blame the authors of such systems for including mistakes: everything that is man-made must be expected to contain mistakes, and realist ontologies, too, will not be error-free. What we question, rather, is the unprincipled way in which such systems have been put together [30, 31]. The good news, on the other hand, is that, as we will explain below, some of these systems may be transformed into sound ontologies of the sort which will be free of at least many of the given types of errors when they are used in an RTS.

7.1 How concept-based systems can help in referent tracking

Even in their current form, concept-based systems can play a useful role in the context of the referent tracking paradigm, since the latter in and of itself brings many of the advantages to be gained by the enforcement of sound ontological principles.

At the level of the IUI-repository, links to the most appropriate concepts in a concept-based system stored in the c argument of an A-tuple would enable the terms of the system to be used in searches, for example searches designed to establish whether a particular has already been assigned a IUI.

Also tuples similar in form to *PtoCL* tuples, but in which *cl*, i.e. the reference to a class from an ontology, is replaced by a reference to a concept from a concept-based system, would be useful for searching. Of course, the relationship to be used cannot be some variant of '*instance of*' since the standard definitions in use for '*concept*' (such as '*unit of knowledge*' [32] or '*unit of thought*' [33]) disallow most particulars from being declared as instances of concepts. (Instances of concepts would be, perhaps, contents of a knowledge base under the first definition, or ideas in people's minds under the second.) But #IUI-1921 (the first author's left testicle) will never be an instance of a concept, however the latter notion might be defined or whatever might happen to that particular in the future. Hence we prefer to place concept-based systems back in close relation to the task for which they were originally designed, namely to assist specialists in a given domain in obtaining a better grasp of the variety of terms and contexts in use in that domain for purposes of communication in a particular language such as English or French. Thus we would use the concept-*codes* that can be found in concept-based systems (turning aside from the concepts that they are intended to represent) as place-holders for the corresponding natural-language expressions for purposes of vocabulary regimentation. We would therefore be interested, not in any special entities called 'concepts', but rather in codes, in *synonyms,* and in

preferred terms. What we shall refer to as *PtoCO* tuples (**p**article to **co**ncept code) will then have the form

$$Co_i = <IUI_a, t_a, cbs, IUI_p, co, t_r>$$

where

- IUI_a is the IUI of the author asserting that terms associated to *co* may be used to describe *p*,

- t_a is a time-stamp indicating when the assertion was made,

- *cbs* is the ID of the concept-based system from which *co* is taken,

- IUI_p is the IUI referring to the particular which the author associates with *co*,

- *co* is the concept-code in the concept-system referred to by *cbs* which the author associates with IUI_p, and

- t_r is a time-stamp representing the time at which the author considers the association appropriate.

Such tuples are to be interpreted as providing a facility equivalent to a simple index of terms in a work of scientific literature. The "annotation" of an entry in a database by means of a term from, for example, the Gene Ontology [34], is a typical example. All that the information in such a tuple tells us is that, within the linguistic and scientific community in which the concept-system referred to by *cbs* is used, it is acceptable to use the terms associated with *co* to refer to the particular.

As an example, the tuple

<IUI-0945, 18/04/2005, SNOMED-CT v0301, IUI-1921, 367720001, forever>

tells us that the first author of this paper on April 18, 2005 asserted that his left testicle within the linguistic and scientific community in which SNOMED-CT is accepted for use may always be denoted by the phrase "*that left testis*", since the term "*left testis*" is in SNOMED-CT v0301 recognised as an adequate term for concept code *367720001*. (And also, however odd this may sound, that "*that entire left testis*" is acceptable too, since "*entire left testis*" is an alternative term associated in SNOMED-CT with the same concept-code.) Furthermore, by taking advantage of the structure of (properly designed) concept-based systems, in which terms more generic than a given term (its ancestor terms in an *is_a* tree) are also acceptable, we can refer to #IUI-1921 in addition by using the phrases "*that testicle*", "*that male gonad*", "*that testis*", "*that genital structure*", …, "*that physical anatomical entity*", and so on – though not (in spite of the fact that we would then be still progressing upwards in the SNOMED-CT *is_a* hierarchy) "*that SNOMED-CT concept*". We consider this last *is_a* relationship to be a mistake in the structure of SNOMED-CT.

For the reasons given already above, the relationships that are used in concept-based systems to associate concepts with each other are too imprecisely defined to be usable in describing relationships that obtain between corresponding particulars. They have some value, rather, in providing guidance on how to browse through the systems in question to find terms that can be used to denote related particulars in ways acceptable to given user communities.

7.2   How referent tracking can help concept-based systems maintenance

We believe that referent tracking, when properly used, can solve one of the most substantial problems in the domain of concept-based systems, namely how to map them amongst each other. Indeed, if referent tracking would be applied in a sufficiently large community, mappings between different terminologies would from a certain point in time be generated as automatic by-products of the referent tracking effort. Systematic referent tracking would

also solve the problem of how to reuse data that have been coded by means of older versions of specific systems, and also to help uncover mistakes in such systems, in the application of such systems in given institutions, and so forth.

To see how this works in more detail, imagine that patient #IUI-001 consults #IUI-201, a physician working in hospital #IUI-211, and that, in order to obtain a second opinion, the same patient thereafter consults #IUI-3900, a surgeon in the clinic #IUI-0098. The EHR-system in #IUI-211 is not designed to work with formal ontologies, but it nonetheless has facilities to code data in detail using SNOMED-CT and to represent the data by means of *PtoCO* tuples. It is also connected to the same RTS to which the EHR-system of #IUI-0098 is connected. The latter, however, uses MEDCIN, a concept-based system of a quite different stamp [35]. Clearly, the entities described by both physicians will be the very same entities, so that many of their descriptions will contain the same IUIs, but different codes from two different concept systems will be used in their descriptions. If both physicians have done their jobs properly (or the coders or transcribers registering data on their behalf have done so), then their combined effort results in a mapping of a small portion of MEDCIN to a small portion of SNOMED-CT and vice versa. When many health facilities, some of them using the same, some using different coding systems, but servicing overlapping patient populations, are all connected to the same RTS, there would in due course be a massive pool of *PtoCO* tuples covering identical particulars. Such a pool of data could be mined automatically, not only with respect to the particulars that are described and about which relevant inferences can be made (as well as concerning the universals they are instances of), but also in order to uncover certain problems related to the concept systems employed in the creation of the data, problems connected with the various ways in which interdependencies are maintained (for example in the creation of mappings between a concept-based system designed for billing purposes and another designed for clinical coding) their adequacy with respect to the task they are intended to perform, and so

forth. In fact, standard statistical techniques could be used to identify codes that are likely to be misused (or not used at all, and hence obsolete), and to identify clinicians who do not understand the intended meaning of certain codes (due to a lack of training or because the concept system from which the codes are taken is poorly documented), as well as many other shortcomings in the process of documenting patient cases.

8 Conclusion

We have sketched, in very broad outline, how the referent tracking paradigm might be implemented in the healthcare environment, particularly in relation to clinical record-keeping. The key idea is for the first time to do full justice to the *what it is on the side of the patient* that is being documented in an EHR. From a functional perspective, the effort required from staff involved in the documentation process is not significantly greater, at least if they are already accustomed to working with concept-based systems. Rather than using a concept-based system for retrieving *codes* that are applicable generically to the patient's condition, these systems should be used to retrieve the *IUIs* referring specifically to that concrete condition itself (just as we use proper names, and not general terms like "human being", to refer to individual patients). In that process, one might discover that a code used previously is no longer appropriate (for one of the many reasons discussed above), which will necessitate the addition of one or more additional statements and the flagging of existing statements to accommodate the new situation. This effort serves the direct purpose of providing better patient descriptions, and has the indirect consequence of leading to mappings between and to quality improvements within concept systems, and so forth.

We foresee a time when, in addition to or in replacement of concept-based systems, principles-based ontologies will be used, including relations defined therein that are

appropriate for describing relations that obtain between particulars. From that point on, there will become available a much richer description of real-world phenomena, of a type that will, we believe, be capable of being used for automatic decision support in a variety of still only barely imaginable ways. Again, the effort required to document relations between particulars by using ontologies of this sort is not greater than that which is involved when using the unprincipled and at best poorly defined relationships that are found in the standard concept-based systems which are currently in use.

9 Acknowledgments

10 References

[1]   Popescu-Belis A and Lalanne D. Reference Resolution over a Restricted Domain: References to Documents. ACL 2004 Workshop on Reference Resolution and its Applications, Barcelona, Spain, July 2004, pp. 71-78.

[2]   Ceusters W, Smith B, Kumar A, Dhaen C. Ontology-based error detection in SNOMED-CT. Proceedings of MEDINFO 2004; 482-6.

[3]   Ceusters W, Smith B. Tracking Referents in Electronic Healthcare Records. Submitted to MIE 2005.

[4]   Clark T, Martin S, Liefeld T. Globally distributed object identification for biological knowledgebases. Brief Bioinform. 2004 Mar;5(1):59-70.

[5]   Smith B. Beyond Concepts, or: Ontology as Reality Representation. In: Achille Varzi and Laure Vieu (eds.), Formal Ontology and Information Systems. Proceedings of the Third International Conference (FOIS 2004), Amsterdam: IOS Press, 2004, 73.84

[6]   Neuhaus F, Grenon P, Smith B. A Formal Theory of Substances, Qualities, and Universals", in Formal Ontology in Information Systems edited by A. C. Varzi and L. Vieu, IOS, 2004, 49-59.

[7]   Willliams S, Kindel C. The Component Object Model: A Technical Overview. October 1994. http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dncomg/html/msdn_comppr.asp. Last visited: April 12, 2005.

[8]   ISO/IEC FDIS 9834-8:2004. Information technology -- Open Systems Interconnection -- Procedures for the operation of OSI Registration Authorities: Generation and registration of Universally Unique Identifiers (UUIDs) and their use as ASN.1 Object Identifier components.

[9]   Bellare M and Rogaway P, The exact security of digital signatures - How to sign with RSA and Rabin. Proceedings of Eurocrypt'96, LNCS vol. 1070, Springer-Verlag, 1996, pp. 399-416.

[10]   Frege G. On sense and reference. In Black, M. and Geach, P. T., editors, Translations from the Philosophical Writings of Gottlob Frege. 1984.

[11]   Netter WJ. Curing the unique health identifier: a reconciliation of new technology and privacy rights. Jurimetrics. 2003 Winter;43(2):165-86.

[12]   Smith B. On the Cognition of States of Affairs, in Mulligan K (ed.) Speech Act and Sachverhalt: Reinach and the Foundations of Realist Phenomenology, Dordrecht/Boston/Lancaster: Nijhoff, 1987, 189-225. (http://ontology.buffalo.edu/smith/articles/cogsvh/cogsvh.html)

[13] Dogac, A., Laleci, G., Kabak, Y., Unal, S., Beale, T., Heard, S., Elkin, P., Najmi, F., Mattocks, C., Webber, D., "Exploiting ebXML Registry Semantic Constructs for Handling Archetype Metadata in Healthcare Informatics" ,accepted for publication, International Journal of Metadata, Semantics and Ontologies.

[14] Ceusters W, Waagmeester A, De Moor G. Syntactic-semantic tagging conventions for a medical treebank: the CASSANDRA approach. In van der Lei J, Beckers WPA, Ceusters W, van Overbeeke JJ (eds.): Proceedings MIC'97, Veldhoven, The Netherlands, 183-193, 1997.

[15] Ceusters W, Deville G. A Mixed Syntactic-Semantic Grammar for the Analysis of Neurosurgical Procedure Reports: the Multi-TALE Experience. In Sevens C, De Moor G (eds), MIC'96 Proceedings, 59 - 68, 1996.

[16] Ceusters W, Rogers J, Consorti F, Rossi-Mori A. Syntactic-semantic tagging as a mediator between linguistic representations and formal models: an exercise in linking SNOMED to GALEN. Artificial Intelligence in Medicine 1999; 15: 5-23.

[17] Griffith SM, Kalra D, Lloyd DS, Ingram D. A portable communicative architecture for electronic healthcare records: the Good European Healthcare Record project (Aim project A2014). Medinfo. 1995;8 Pt 1:223-6.

[18] Beale T, Heard S, Kalra D, Lloyd D. The openEHR EHR Information Model, version 4.5, December 10, 2004. http://www.openehr.org/repositories/spec-dev/latest/publishing/index.html. Last visited April 18, 2005.

[19] ENV 13606-1 1999 Health informatics - Electronic healthcare record communication - Part 1: Extended architecture.

[20] Davis S. Tagging along. RFID helps hospitals track assets and people. Health Facil Manage. 2004 Dec;17(12):20-4.

[21] Locke M. UC Considers Using Barcodes for Cadavers. San Francisco Chronicles, February 5, 2005. http://www.sfgate.com/cgi-bin/article.cgi?file=/news/archive/2005/02/04/national/a110118S59.DTL. Last visited: April 19, 2005.

[22] Grenon P. Spatio-temporality in Basic Formal Ontology: SNAP and SPAN, Upper-Level Ontology, and Framework for Formalization, IFOMIS Technical Report, 05, 2003. (http://www.uni-leipzig.de/~pgrenon/Downloads/grenon-tr1-part1.pdf. Last visited: April 13, 2005)

[23] Smith B, Ceusters W, Klagges B, Kohler J, Kumar A, Lomax J, Mungall CJ, Neuhaus F, Rector A, Rosse C (2004) Relations in Biomedical Ontologies. Genome Biology, 2005 (accepted) (http://ontology.buffalo.edu/bio/OBORelations.pdf Last visited: April 13, 2005)

[24] Rosse C, Mejino JL Jr. A reference ontology for biomedical informatics: the Foundational Model of Anatomy. J Biomed Inform. 2003 Dec;36(6):478-500.

[25] Donnelly M, Bittner T, Rosse C. Formal Theory for Spatial Representation and Reasoning in Biomedical Ontologies. Submitted to Artificial Intelligence in Medicine.

[26] Schneider L, Cunningham J. Ontological Foundations of Natural Language Communication in Multi-Agent Systems , in Vasile Palade, Robert J.Howlett, Lakhmi Jain (Eds) Knowledge Based Intelligent Information and Engineering Systems, Springer LNAI-2773, 2003 pp 1403-1410

[27] Gruber TR. (1993). A Translation Approach to Portable Ontology Specifications. Knowledge Acquisition, 5:199-220.

[28] Fielding JM, Simon J and Smith B. "Formal Ontology for Biomedical Knowledge Systems Integration", Proceedings of EuroMISE, Prague, April 12-15, 2004.

[29] Bodenreider O, Smith B, Burgun A. The ontology-epistemology divide: A case study in medical terminology. In: Varzi AC, Vieu L, editors. Proceedings of the Third International Conference on Formal Ontology in Information Systems (FOIS 2004): IOS Press; 2004. p. 185-195.

[30] Ceusters W, Smith B. A Terminological and Ontological Analysis of the NCI Thesaurus. Methods of Indormation in Medicine 2005 (accepted).

[31] Ceusters W, Smith B, Kumar A, Dhaen C. Ontology-based error detection in SNOMED-CT. Proceedings of MEDINFO 2004; 482-6.

[32] ISO-1087-1:2000. Vocabulary of terminology.

[33] ISO-1087:1990. Vocabulary of terminology.

[34] Martucci D, Masseroli M, Pinciroli F. Gene ontology application to genomic functional annotation, statistical analysis and knowledge mining. Stud Health Technol Inform. 2004;102:108-31.

[35] Medicomp Systems I: MEDCIN. Chantilly, Va: Medicomp Systems, 1998.