Representing and sharing knowledge using SNOMED
Proceedings of the 3rd international conference on Knowledge Representation in Medicine (KR-MED 2008)
R. Cornet, K.A. Spackman (Eds)

Leveraging SNOMED CT with a General Purpose Terminology Server

R. Weida, PhD, J. Bowie, ScD, R. McClure, MD, D. Sperzel, MD
Apelon, Ridgefield, CT, USA
weida@apelon.com

General purpose terminology server software facilitates coordinated use of multiple standard medical terminologies for diverse healthcare applications. SNOMED CT is an important clinical reference terminology, whose size and scope make advanced terminology server capabilities particularly useful. Moreover, capabilities tied to SNOMED CT's special features and requirements can result in substantial further benefits. Enhancements to a general purpose terminology server have been developed to facilitate the tailored creation, validation, organization, deployment, distribution, submission and maintenance of (post-coordinated) extensions to SNOMED CT.

INTRODUCTION

Standard medical terminologies are vital to all sorts of contemporary healthcare information technology endeavors, ranging from encoding and exchanging information in electronic health record (EHR) systems to facilitating outcomes analysis and decision support. However, effective integration of terminologies into clinical applications poses substantial challenges. These applications generally require multiple terminologies since each terminology has been designed for different purposes by different healthcare constituencies, e.g., SNOMED CT for representation of clinical data; ICD-9-CM, ICD-10-CM and CPT-4 for reimbursement; LOINC for laboratory test results; and HL7 for application interfaces. Drug nomenclatures such as RxNorm and NDF-RT, device taxonomies such as UMDNS, specialty ontologies, and others are also important, as are enterprise-specific terminology enhancements. Terminologies employ different data models and they are delivered in different data formats. Finally, terminologies are constantly evolving, so they must be regularly updated in clinical and other applications. However, revision schedules and processes vary widely and are often inconsistent. Such challenges can be effectively met with a comprehensive, general purpose *terminology server,* defined as a networked software component that centralizes and integrates terminology content and reasoning to provide (complete, consistent, effective) terminology services for users and other network applications. Earlier terminology servers[1,2,3,4] did not provide the modular classification, subset, template or SNOMED-specific features described here.

Terminology servers support diverse applications. For example, they are used by informaticists to create, maintain, localize and map terminologies; by clinical applications and their users to select and record standardized data; and by software integration engines to map data elements between applications. SNOMED CT is of special interest due to its broad clinical scope, extensive detail, formal structure, and international standing.[5,6] This paper describes some ways that one general purpose terminology server has been enhanced and applied to support SNOMED CT within the context of a full complement of other healthcare terminologies.

DISTRIBUTED TERMINOLOGY SYSTEM

Apelon's Distributed Terminology System (DTS) is an open source terminology software suite whose key component is a terminology server. DTS is robust and mature, benefiting from years of production deployment in diverse healthcare industry settings. It has been used by software and content vendors, pharmaceutical companies, government agencies, universities and research institutions, healthcare delivery systems, and standards development organizations around the world.

DTS Architecture

DTS employs typical three-tier architecture, as illustrated in Figure 1. Multi-tier architectures offer many well known advantages, including the ability to support highly flexible, easily scalable, and extremely dependable deployment solutions.

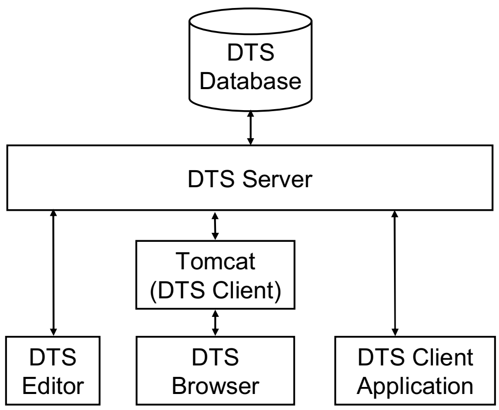


Figure 1 – DTS Architecture.

The DTS client tier (below the DTS Server in Figure 1), provides both Java and .Net APIs for developing custom terminology applications. DTS comes with packaged client applications such as an extensible desktop (fat client) terminology editor, the DTS Editor. There is also a web-based (thin client) terminology browser, the DTS Browser, which requires an Internet browser and an intermediary Apache Tomcat (or equivalent) web server. The middle tier of DTS consists of the DTS Server, a terminology-focused application server which supports highly concurrent, authenticated access to terminology services via the APIs. It features numerous performance optimizations, logging, tracing, remote monitoring, etc. The APIs support browsing, navigation, search, query, editing, localization, mapping, subsetting and other common terminology operations. A relational database comprises the third – or data – tier of DTS, shown at the top of Figure 1. In addition, DTS supplies various utilities for software and content management, including content subscription updates. Readers interested in DTS features outside the scope of this paper are referred to the DTS White Paper[7].

DTS Namespaces

DTS employs a unified content model for uniform access to diverse terminologies, including ones based on Description Logic (DL) such as SNOMED CT, the NCI Thesaurus and NDF-RT, as well as non-DL terminologies like CPT, ICD, and LOINC. A subscription service is available for all major medical terminologies (plus cross-terminology mappings) formatted for easy loading into DTS, ensuring that the latest versions of the terminologies are always available. A DTS *namespace* is the unit of management for content delivery (and access control). Thus, each standard terminology resides in a separate namespace so it can be independently updated and versioned. A *mapping* between (elements of) a pair of terminologies, e.g., from CPT to SNOMED CT, is also typically delivered in its own separate namespace. DTS also supports an unlimited number of *local namespaces* enabling users to create and maintain user- or organization-specific terminology data. These local terminologies are also housed in distinct namespaces, as are the local extensions to standard terminologies described below.

DESCRIPTION LOGIC

Description Logic (DL) is a well known field of study within the area of knowledge representation.[8] DL is a type of formal logic focused on creating definitions of concepts and reasoning about them effectively. Thus, DL is well suited for expressing precise descriptions of medical concepts, including anatomy, diseases, drugs, procedures, and so on. DL enables clear and unambiguous *formal definition* of a concept's meaning, primarily in terms of its relationships with other concepts. A given concept (e.g., representing a class of drugs) can be described succinctly by naming the concepts it specializes (more general classes of drugs) and introducing distinguishing characteristics (e.g., relationships to its ingredients). The logical consistency of an entire set of concepts, such as those comprising a medical terminology, is automatically tested and enforced. Moreover, logical consequences that are implicit in the given descriptions are automatically made explicit.

A particular DL provides a language for describing concepts and a repertoire of logical inferences for reasoning about them. SNOMED CT uses the Ontylog DL[9], which is also used for the US Veterans Health Administration's NDF-RT (National Drug File – Reference Terminology) and the National Cancer Institute's NCI Thesaurus, all standards of the US Government's Consolidated Health Informatics (CHI) Initiative[10]. Ontylog syntax and semantics have been published in connection with the NCI Thesaurus.[11] Among the most powerful aspects of DL are its facilities for reasoning about relationships among concepts and thus automatically managing a logically consistent taxonomy (i.e., generalization hierarchy or "is-a" hierarchy) of concepts.

The DL *classification* operation automatically organizes concepts into a taxonomy based on their logical descriptions. Software that implements classification is called a *classifier*. As a simplified expository example, a set of concepts { *A, B, C, D, E, F, G, H, I, J* } might be classified into the taxonomy shown in the top portion of Figure 2, where *A* is a generalization of *B*, *C* and *D*; *B* is a generalization of *E*, *F* and *G*, etc. We will use this taxonomy in subsequent examples. Extant classifiers generally create an explicit representation of a taxonomy, including explicit information corresponding to each of the lines shown between pairs of linked concepts. The Apelon classifier generates a very high performance, in-memory "classification graph" which includes all information necessary to continue classifying additional concepts in the future.

As a result of classification, each concept in the taxonomy is guaranteed to be more specific than its parents and all other ancestors (directly or indirectly connected concepts above), as well as more general than its children and all other descendants (directly or indirectly connected concepts below). Therefore, concepts are always found in predictable locations. That makes it easier to envision relationships among concepts and to recognize unintended results. Wellorganized taxonomies allow medical knowledge (e.g., advice, rules, warnings, arbitrary codes, etc.) to be associated with concepts at the most appropriate level in the taxonomy (neither too general nor too specific) and appropriately inherited by (implicitly associated with) descendant concepts.

A *terminology* is a collection of presumably related concepts. In DTS, a namespace is a set of concepts that are managed as a group. Thus, one can classify the set of concepts comprising a namespace into a taxonomy. Ordinarily, an entire terminology is contained – and thereby managed – in one namespace, e.g., all the concepts shown in the top portion of Figure 2 might comprise a single namespace. (For authoring purposes, some DLs allow terminologies to be composed by "importing" (the concepts of) one terminology into another, but the entire result is still classified monolithically.)

MODULAR EXTENSION

DTS terminology extension features are motivated largely by the existence of SNOMED CT and the desire of users to adapt it in diverse ways. SNOMED CT contains hundreds of thousands of concepts. New versions of SNOMED have been released twice yearly. Many different users (persons or organizations) may wish to extend SNOMED by adding their own concepts. The SNOMED data model provides for this possibility. Indeed a single user may be interested in extending SNOMED several different ways. However, it is important to clearly distinguish the authoritatively published core of SNOMED from any extensions thereof. Furthermore, it is important to classify terminology extensions, including post-coordinated expressions, as rapidly as possible. Traditional classifiers organize an entire set of concepts into a taxonomy by "starting from scratch" and classifying (processing) each and every concept in turn.

Modular Classification

DTS uniquely facilitates multiple independent extensions of a concept taxonomy based on DL. Separate classification operations determine how one or more distinct sets of additional concepts, each comprising an extension, fit in with the original taxonomy while leaving the original taxonomy intact and without copying it. Classification results are recorded so that the original taxonomy as well as every extension thereof can be independently browsed, searched, queried and retrieved on demand. As a result, DL taxonomies such as SNOMED CT can be extended easily and accurately, using the same language as the original, in multiple independent ways, to meet local and/or specialized needs in a timely manner. We call this process *modular classification.* Thus, DTS introduces effective means for working with multiple independent extensions of an existing taxonomy while preserving the integrity of the original. Indeed, DTS uses the same classification software used in the creation of SNOMED CT.

We will refer to an existing, self-contained namespace, e.g., a namespace containing SNOMED CT, as a *base namespace*. Concepts therein are referred to as *base concepts*. Then, an *extension namespace* contains one or more additional concepts to be classified, viewed, and otherwise used *as if* they were also part of the base namespace, but without altering and without copying the base namespace. Concepts within an extension namespace are referred to as *extension concepts*.

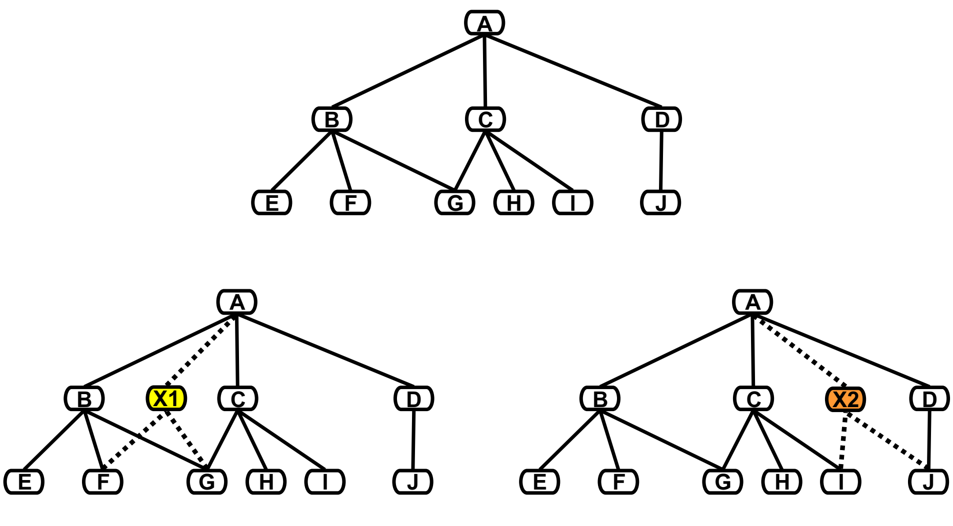


Figure 2 - Base Namespace Taxonomy (top) with Multiple Independent Extended Taxonomies (bottom).

The modular classifier operates on DL elements of SNOMED extension concepts defined in extension namespaces. These concepts are linked by SNOMED relationships to other concepts in the base namespace and/or the same extension namespace. DTS extension namespaces can also contain other local information about core SNOMED concepts. Examples include additional local synonyms; local associations connecting them to or from other concepts, e.g., to represent mappings from a local terminology; and local properties (attribute value pairs, e.g., to indicate that a procedure is performed locally, or that a certain person last edited the concept). In all cases, extensions to SNOMED CT could become problematic if a base SNOMED concept is later retired. Reports detailing any such connections are available, thus allowing for remediation.

As an example, the fictitious Podunk Hospital may wish to extend the *SNOMED CT* base namespace with a *Podunk Hospital* extension namespace. That extension namespace may include an extension concept for a disorder, *Familial vertigo*, with definitional relationships to several base concepts in *SNOMED CT***.** In general, an extension concept can be defined in terms of its relationships to base concept(s) and/or fellow extension concept(s). The user's definition of *Familial vertigo* is shown on the right in Figure 3. This definition was created interactively within the DTS Editor, drawing from concepts and relationships (roles) in the standard SNOMED CT Namespace. Following modular classification, the position of *Familial Vertigo* with respect to one branch of the SNOMED taxonomy is shown on the left. Of note, the classifier has inferred the position of *Familial Vertigo* directly under a concept *Labyrinthine disorder* not mentioned explicitly in its definition. The DTS Editor italicizes extension concepts in the context of a base namespace for emphasis.

In the interest of clarity and brevity (SNOMED CT has hundreds of thousands of concepts), the upper portion of Figure 2 shows a much simpler sample taxonomy for a base namespace. Beneath that are two independent extensions, one where the taxonomy is extended with a namespace consisting of the concept *X1*, and another where the taxonomy is extended with a namespace consisting of the concept *X2*. Notice that an extended taxonomy effectively contains the entire set of concepts from the base namespace augmented with additional concept(s) from the extension namespace. The dashed lines are intended to suggest that while the relationships of the extension concepts to the base taxonomy have been determined, they are not (destructively) spliced into the original base taxonomy (shown with solid lines). While these simple illustrations show only one concept per extension, an extension can of course contain an arbitrary number of concepts. We have used the DTS modular classifier with an extension namespace that (experimentally) extends SNOMED with LOINC laboratory concepts, and another extension namespace containing the US Drug Extension[12], each containing well over 15,000 concepts.

A base namespace may have multiple extensions which depend on it; extensions are mutually independent. Multiple independent namespaces extending SNOMED CT might have a variety of custodians and purposes, including a person (for learning and testing), a project (for research and development), an organization (for specific institutional needs), a specialty society (for terminology related to their practice area), national

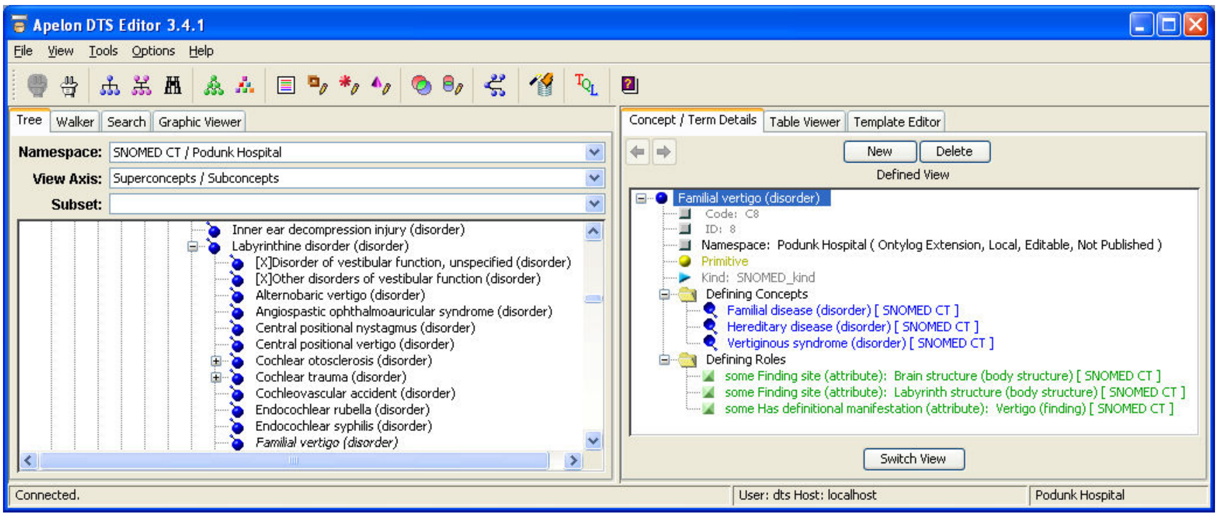
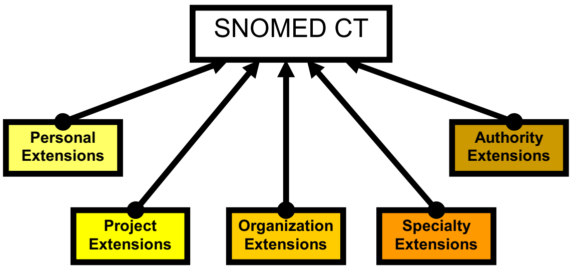


Figure 3 – DTS Editor with Extension Concept (right) and Extended Taxonomy (left).

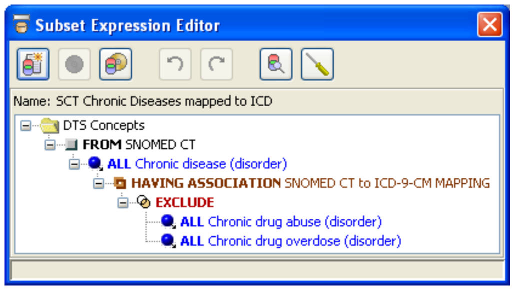
authorities, or even the creators of the base namespace themselves (e.g., to preview possible future enhancements to the base):



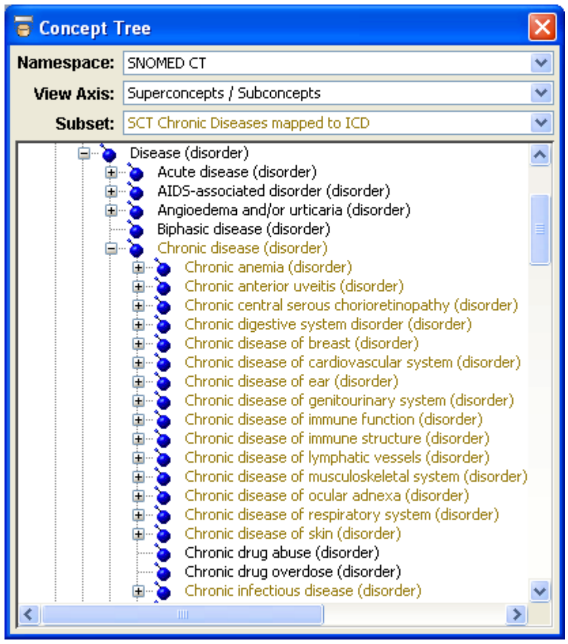
So far, we have focused on authoring sets of concepts covering a unified extension of interest. However, it is important to note that modular classification is equally adept at "on the fly" post-coordination of new concepts in accord with the SNOMED model, e.g., to help populate EHRs at run-time using the DTS API. Logical equivalence (hence redundancy) with a base concept or another extension concept is always detected and reported by the modular classifier.

SUBSETS

Considering the large size and broad scope of SNOMED CT and other contemporary medical terminologies, it can be extremely helpful to work with smaller, more focused subsets of terminologies when populating pick lists in EHR systems or fields in HL7 messages (HL7 value sets), constraining searches to pertinent concepts for data matching and analysis, etc. Subsets of interest can themselves be large and therefore challenging to maintain when the underlying terminologies are revised, e.g., concepts that are members of the subset may be retired and new concepts that should become members may be introduced. Enumerating each element of a large subset is tedious, opaque and often highly inefficient. Therefore, DTS takes a constructive approach to subset specification: a concise *subset expression* compositionally defines an arbitrary subset by specifying member concepts according to their names, synonyms, other properties, and relationships. Subset expressions can specify inclusion or exclusion of identified concepts and/or all of their descendants in the (base or extended) taxonomy. Moreover, subset expressions can be arbitrarily nested to include sub-taxonomies, exclude portions thereof, etc. Subset expressions can use various concept attributes, even those that refer to other namespaces, e.g., we can specify all SNOMED chronic diseases mapped to ICD-9-CM but (strictly for illustration) excluding chronic drug abuse and chronic drug overdose:



Visualization of subsets greatly aids review and revision. The DTS Editor (and likewise the web-based browser) can highlight subset members in the larger context of the entire SNOMED taxonomy; note the subset member concepts highlighted in gold:

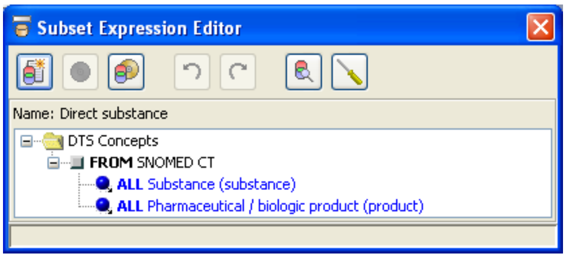


The DTS Editor can also render and browse the hierarchical structure of the subset members alone, just as if all non-members were spliced out of the original taxonomy (not shown for brevity). Of course, DTS can also enumerate and export subsets, test for subset membership, search within subsets, etc. All of these features are available in the DTS Editor GUI application and also via the DTS APIs for runtime application integration.

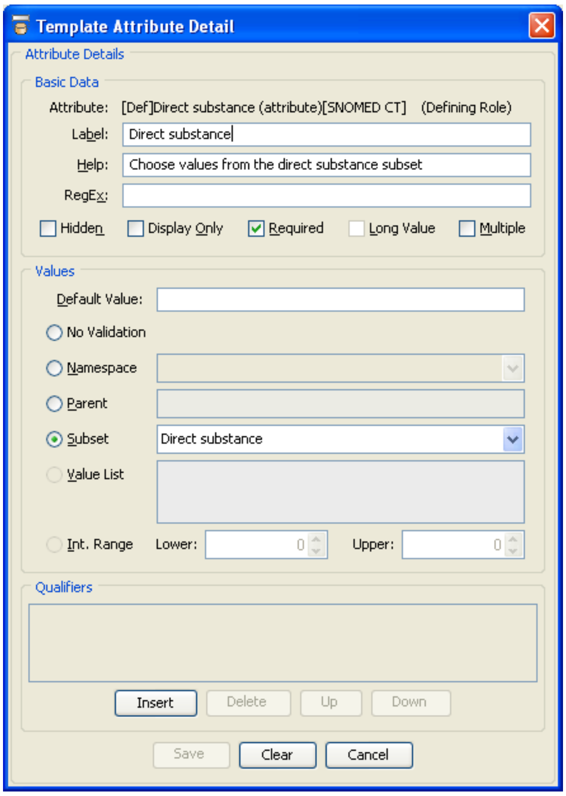
TEMPLATES

Since DTS is a general purpose system for arbitrary terminologies, the DTS Editor enables unconstrained editing using generic terminology constructs. However, the SNOMED model carefully constrains concept definitions. Particular types of concepts (within a particular SNOMED hierarchy) are to be defined using particular SNOMED relationships to target concepts chosen from particular portions of SNOMED. The DTS *Template Builder* (a DTS Editor "plug-in") has been developed to specify templates for context-dependent editing in compliance with such a model. Due to space restrictions, the following example is necessarily very abbreviated and simplified but conveys the gist.

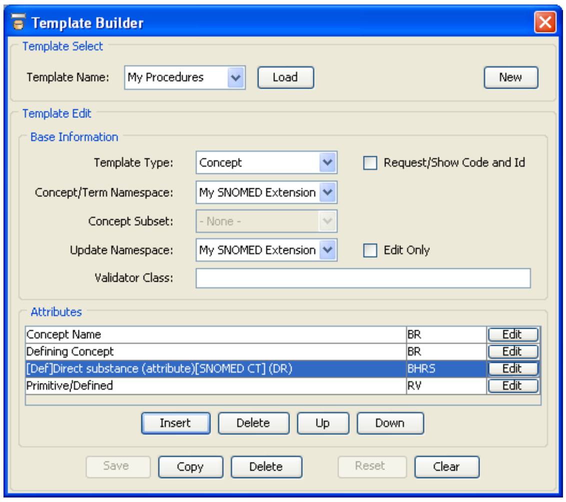
Suppose we need to extend SNOMED with more procedures. The *SNOMED CT Users Guide* specifies that the value of a *Direct substance* relationship (when present) on a *Procedure* concept should be a *Substance* or a *Pharmaceutical/biological product*. Thus, we create a *Direct substance* subset:



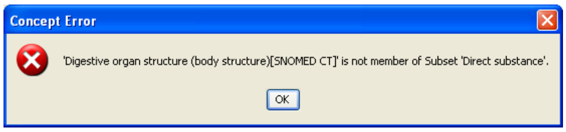
As we create a template for our procedures, we can require a value for the *Direct substance* relationship and require that it be restricted to members of our *Direct substance* subset:



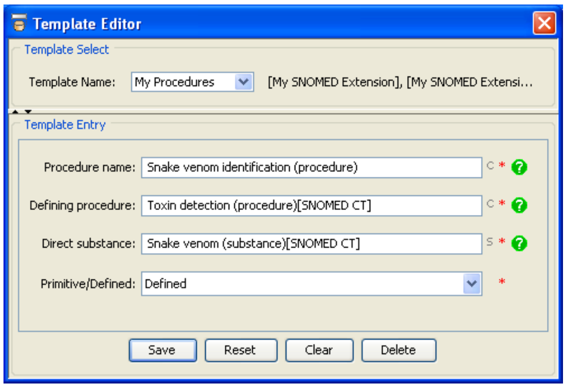
The *Direct substance* relationship is one attribute of an overall template for *My Procedures* (which are concepts in the *My SNOMED Extension* namespace):



The DTS *Template Editor* enables creation and modification of concepts according to such templates. It reports an error if we attempt to use a concept that is not a member of the specified subset as the value for a *Direct substance* relationship:



The Template Editor accepts a member of the subset, as in this definition of *Snake venom identification*:



Notice the template-specific labels: *Procedure name, Defining procedure* and *Direct substance*. Absent any intervening extension concepts, the modular classifier will place this *Snake venom identification* extension concept directly under the *Toxin detection (procedure)* concept from the *SNOMED CT* base namespace.

DISTRIBUTION AND SUBMISSION

There are several ways to transfer terminology content into, out of and between DTS instances. Apelon distributes full and incremental versions of many standard (and custom) terminologies using a compact data format which closely corresponds to the DTS database schema and can therefore be loaded very efficiently. DTS enables users to distribute their own DTS terminology content in the same format. In addition, DTS includes graphical tools – the *import wizard* and the *export wizard* – to easily move ad hoc terminology content in and out of DTS using delimited text and XML formats. However, SNOMED CT has its own release format, consisting of a set of related files, tailored to the SNOMED data model, which specifically support SNOMED extensions. A SNOMED CT Identifier (SCTID) uniquely identifies all concepts, descriptions and relationships in SNOMED CT. Those who wish to extend SNOMED CT can request their own, exclusively assigned range of SNOMED CT identifiers. To facilitate creation and distribution of SNOMED extensions using DTS, we have implemented new DTS capabilities in collaboration with a national terminology authority and with a leading academic medical center. These capabilities include generation of SCTIDs for all elements of a SNOMED Extension namespace in DTS, as well as import and export of extension namespaces in SNOMED release format. Thus, SNOMED extensions can be readily shared with collaborators, and as appropriate, could be submitted for possible inclusion in the SNOMED core. The fact that these extensions have already been successfully classified together with the SNOMED core should expedite review and possible acceptance.

CONCLUSION

Apelon DTS, now available via open source licensing, has proven to be a popular tool for enterprise terminology asset management, featuring comprehensive capabilities for working with multiple standard and local terminologies, both individually and in concert (e.g., via mappings) using a unified suite of software components. Recognizing the importance of SNOMED CT, we have added significant functionality to meet SNOMED's unique requirements and benefit from its unique capabilities.

Acknowledgments

We gratefully acknowledge the contributions of past and present Apelon colleagues to the ideas described in this paper and the DTS system. We deeply appreciate review and feedback on new SNOMED capabilities in DTS from Dr. James Campbell at the University of Nebraska Medical Center and from Australia's National E-Health Transition Authority.

References

1. Rector AL, Solomon WD, Nowlan WA, Rush TW, Zanstra PE, Claassen WM. A terminology server for medical language and medical information systems. Meth Inform Med. 1995, 34(1-2) p. 147-57.
2. Mays E, Weida R, Dionne R, Laker M, White B, Liang C, and Oles, FJ. Scalable and expressive medical terminologies. AMIA Annual Fall Symposium, 1996. p. 259-263.
3. Burgun A, Patrick D, Bodenreider O, Botti G, Delamarre D, Poulinquen B, Oberlin P, Leveque JM, Lukacs B, Kohler F, Fieschi M, LeBeux P. A web terminology server using UMLS for the description of medical procedures. J Am Med Inform Assoc. 1997; 4:356–363.
4. Chute CG, Elkin PL, Sherertz DD, Tuttle MS. Desiderata for a clinical terminology server. Proc AMIA Symp1999: 42-6.
5. Spackman KA, Campbell KE, and Cote RA. SNOMED RT: A reference terminology for health care. Proceedings of the AMIA Annual Fall Symposium, 1997. p. 640–644.
6. IHTSDO: SNOMED CT®. [Online]. 2007 [cited 2008 Jan 15]; Available from: http://www.ihtsdo.org/our-standards/snomed-ct/
7. Distributed Terminology System. [Online]. 2006 [cited 2008 Jan 15]; Available from: URL: http://www.apelon.com/products/white papers/DTS White Paper V34.pdf
8. Baader F, Calvanese D, McGuinness DL, Nardi D, and Patel-Schneider PF, editors. The description logic handbook: theory, implementation, and applications. Cambridge (U.K.): Cambridge University Press; 2003.
9. Spackman KA, Dionne R, Mays E, Weis J. Role grouping as an extension to the description logic of Ontylog motivated by concept modeling in SNOMED. AMIA Annual Symposium, 2002. p. 712-716.
10. Presidential Initiatives. [Online]. 2007 [cited 2008 Jan 15]; Available from: URL: http://www.hhs.gov/healthit/chiinitiative.html
11. Hartel FW, de Coronado S, Dionne R, Fragoso G, Golbeck J. Modeling a description logic vocabulary for cancer research. J Biomed Inform. 2005 Apr. p. 114-29.
12. CAP SNOMED Terminology Solutions. Pharmacy. [Online]. 2007 [cited 2008 Jan 15]; Available from: URL:http://www.cap.org/apps/docs/snomed/documents/pharmacy_nov07.pdf