

## A methodology for partitioning a vocabulary hierarchy into trees<sup>1</sup>

Huanying (Helen) Gu<sup>a,\*</sup>, Yehoshua Perl<sup>a</sup>, James Geller<sup>a</sup>,  
Michael Halper<sup>b</sup>, Mansnimar Singh<sup>c</sup>

<sup>a</sup> *Department of Computer and Information Science, New Jersey Institute of Technology, Newark, NJ 07102, USA*

<sup>b</sup> *Department of Math and Computer Science, Kean University, Union, NJ 07083, USA*

<sup>c</sup> *Onyx, Inc. Ridgefield, CT 06811, USA*

Received 30 September 1997; received in revised form 20 February 1998; accepted 1 April 1998

---

### Abstract

Controlled medical vocabularies are useful in application areas such as medical information systems and decision-support systems. However, such vocabularies are large and complex, and working with them can be daunting. It is important to provide a means for orienting vocabulary designers and users to the vocabulary's contents. We describe a methodology for partitioning a vocabulary based on an IS-A hierarchy into small meaningful pieces. The methodology uses our disciplined modeling framework to refine the IS-A hierarchy according to prescribed rules in a process carried out by a user in conjunction with the computer. The partitioning of the hierarchy implies a partitioning of the vocabulary. We demonstrate the methodology with respect to a complex sample of the MED, an existing medical vocabulary. © 1999 Elsevier Science B.V. All rights reserved.

*Keywords:* Controlled medical vocabulary; Object-oriented modeling; Ontology; Semantic network; Partitioning

---

\* Corresponding author. Fax.: +1 973 5965777; e-mail: helen@homer.njit.edu

<sup>1</sup> This research was (partially) done under a cooperative agreement between the National Institute of Standards and Technology Advanced Technology Program (under the HIIT contract # 70NANB5H1011) and the Healthcare Open Systems and Trials, Inc. consortium, and the Center for Manufacturing Systems.

## 1. Introduction

Controlled medical vocabularies (vocabularies for short) play an important role in many medical enterprises that employ a large number of disparate information systems (e.g. clinical databases). Often, each such system has its own inherent ‘language’ or terminology. A number of such vocabularies have appeared in the medical field [11,34,35]. Of note is the Medical Entities Dictionary (MED) developed and in use at Columbia-Presbyterian Medical Center (CPMC) [8,9]. Controlled vocabularies have been shown to greatly facilitate the process of integrating medical information systems [10] using different terminologies. They also help to standardize common information handling tasks and reduce the overall cost of data processing.

While a controlled vocabulary offers tremendous benefits, these benefits do come at a price. A vocabulary can be quite extensive and can contain an overwhelming amount of structural and semantic complexity. For example, the MED contains over 48000 concepts, over 61000 IS-A links and over 71000 other links. (We are referring to a particular version of the MED, dated 12/96, throughout this paper.) Obviously, the job of comprehending such a vocabulary can be an extremely difficult problem in itself.

In this paper, we are concerned with providing a tool to help users comprehend vocabularies. In particular, we present a methodology to make large and complex vocabularies easier to understand. Our approach is based on the partitioning of a vocabulary into *manageably-sized, meaningful* units. The partitioning assumes the existence of a vocabulary with an IS-A hierarchy and centers around this IS-A (or concept subsumption) hierarchy.

To enhance comprehension of the MED vocabulary [14], we have mapped it into an OODB schema representation based on partitioning the vocabulary into sets of concepts with the same sets of properties. In [21], we reported on implementing the InterMED (a partial revised version of the MED) [26,33] using ONTOS, a commercial OODB system. We call the resultant OODB the Object-Oriented Healthcare Vocabulary Repository (OOHVR). Among other things, the OOHVR’s schema captures the complete structure of the vocabulary in a compact form which aids in its comprehension. However, for the much larger MED, each class in the corresponding OODB schema summarizes on average 500 concepts. A vocabulary of 500 concepts is still hard to understand. Thus, further partitioning efforts are needed to enhance comprehension.

The backbone of many controlled vocabularies is the IS-A hierarchy which relates more specialized concepts (subconcepts) to more generalized concepts (superconcepts) that subsume them. The IS-A hierarchy also serves as the basis for property inheritance. In general, the IS-A hierarchy of a controlled vocabulary will be a directed acyclic graph, permitting multiple superconcepts and multiple inheritance. Our methodology is based on the following two premises: (1) a vocabulary’s IS-A hierarchy taken alone is much more comprehensible than the entire vocabulary itself; (2) a ‘forest’ IS-A hierarchy (i.e.

a collection of trees in which every link is an IS-A link and where, by definition, no concept has more than one superconcept) is easier to comprehend than a directed acyclic graph containing the same number of concepts.

With these premises in mind, we develop a theoretical framework that reduces an entire vocabulary (typically represented as a large semantic network) into a forest hierarchy composed of small trees, each representing a logical unit whose graphical representation can fit on a computer screen. This reduction in size makes it easier for users and system designers alike to comprehend the contents of the vocabulary in a modular fashion.

Our methodology relies on an interaction between a user (presumably the vocabulary designer or administrator) and the computer. The process requires that a user refines the vocabulary's IS-A hierarchy according to some prescribed principles so that it conforms to what we call the rules of *disciplined modeling*. After the refinement, the computer can automatically reduce the vocabulary to a forest structure. We formally prove that our approach always finds a forest partition as long as the rules of disciplined modeling are adhered to. Let us note that partitioning networks (graphs) according to various criteria has been shown to be NP-complete, i.e. computationally intractable [12].

In previous work [28], we have employed a similar paradigm to reduce the complexity of large object-oriented database (OODB) subclass hierarchies. In this paper, we rework and adapt the approach to the IS-A hierarchy of an extensive, complex vocabulary. Furthermore, we present an interactive methodology for partitioning the vocabulary. To ground our discussion in a real-world application, we will focus on the MED as our test-bed vocabulary. The methodology developed herein will be applied to a complex subnetwork of the MED.

Our approach is closely related to the principle of orthogonal taxonomies' as implemented in the GALEN project [30,31]. There, a taxonomy is organized from the start by requiring that all primitive entities have only one primitive parent. In our methodology, an existing vocabulary is partitioned to achieve a similar effect.

The rest of this paper is organized as follows. In Section 2, we describe the notions of informational thinning and partitioning with respect to vocabularies. Section 3 introduces the rules of disciplined modeling and proves that they make it possible to obtain a meaningful forest hierarchy from a directed acyclic graph. In Section 4, we describe our methodology for partitioning the vocabulary. In Section 5, we apply the methodology to a very complex portion of the MED. Section 6 contains our conclusions. A short, preliminary version of this paper appeared in [15].

## 2. Informational thinning and partitioning

In this section, we describe two approaches which are used to enhance the comprehension of large and complex vocabularies. If a vocabulary network,

containing a vast amount of objects (representing concepts), relationships and attributes, is displayed on a screen, then the user typically has difficulties comprehending and dealing with it. For such an overwhelming display of the InterMED, see [27].

According to our experience, the difficulties of understanding a vocabulary stem more from the number of relationships than from the number of concepts. We define the *complexity*  $c$  of a network or vocabulary as the ratio of the number of relationships between objects to the number of objects. As we mentioned, the MED is an example of a large, complex vocabulary. The complexity of the MED is  $c = (61000 + 71000)/48000 \approx 2.75$ . For two networks with the same number of objects, the more complex network is more difficult to comprehend. Thus, there exists a need to reduce the number of relationships in order to display a simplified comprehensible subnetwork of the vocabulary with a lower complexity. Informational thinning is used to achieve this goal.

**Definition 1:** *Informational thinning* is a technique for eliminating partial information from the display of a whole network. This is done by prioritizing various kinds of properties of the objects in the network and displaying only *kinds* of properties with high priority.  $\square$

In our graphical OODB schema editor OODINI [17], we support two levels of informational thinning. One level removes all attributes and the other removes attributes *and* non-hierarchical relationships leaving only the IS-A hierarchy displayed. The latter level of informational thinning will be used in the figures of this paper. The hierarchy of IS-A relationships is the backbone of a vocabulary, which helps users to comprehend it. The use of informational thinning (level 2) permits us to concentrate on the IS-A hierarchy.

To test our theoretical paradigm and methodology we looked for a subnetwork of the MED with a very complex hierarchy. Our reason is that for our techniques to be applicable for the whole MED vocabulary, it is necessary, although not sufficient, to be successfully applicable to such a subnetwork. We identified a subnetwork with a very complex hierarchy in the MED as follows. From the 48000 concepts in the MED, the concept **CPMC Drug: Cortisporin Ophthalmic Ointment** has the most ancestors, 39. The subnetwork with a complex hierarchy, which we call cortisporin subnetwork, includes the concept **CPMC Drug: Cortisporin Ophthalmic Ointment** and all its ancestors. It contains 821 attributes, 62 IS-A relationships and 157 other relationships. Thus, the complexity of cortisporin subnetwork is  $c = (62 + 157)/(39 + 1) \approx 5.5$ . Such a complex network with so many properties cannot be displayed on one screen.

In Fig. 1, we show the hierarchy of IS-A relationships of cortisporin subnetwork. This hierarchy has the same number of concepts as the original network but fewer relationships. The complexity of the IS-A hierarchy of cortisporin subnetwork is  $c = 62/40 \approx 1.55$ , a much lower complexity than that of cortisporin subnetwork itself. For comparison, the complexity of the IS-A hierarchy of the whole MED is  $c = 54547/42744 \approx 1.27$  which is lower than that of cortisporin subnetwork. To help

in the forthcoming analysis, we added in Fig. 1 some concepts which are not the ancestors of the concept **CPMC: Cortisporin Ophthalmic Ointment**. The added concepts are (33), (34), (39), (41), (43) and (45).

Obviously, the use of informational thinning makes it easier to understand a vocabulary. But comprehending a large and complex IS-A hierarchy may still be very difficult, although informational thinning was applied, due to multiple inheritance and the large number of objects. Considering the limitations of human comprehension capacity and the size of computer monitors (e.g. a network of less

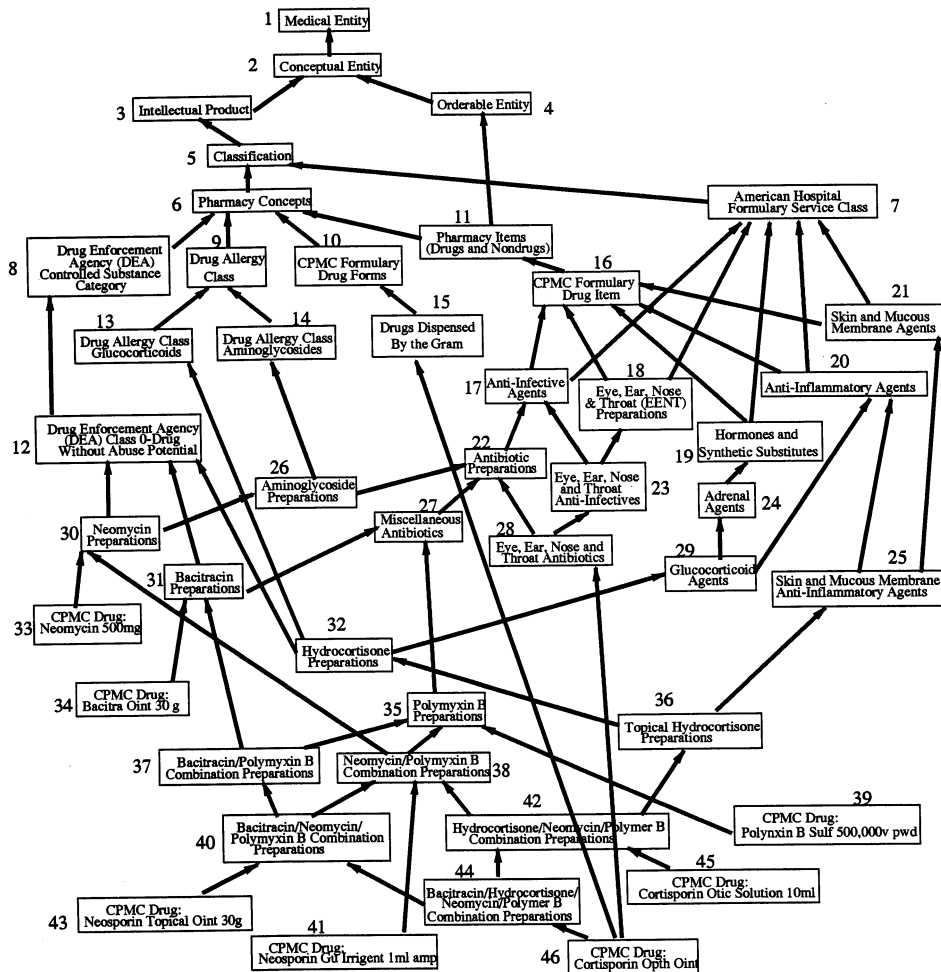


Fig. 1. Cortisporin subhierarchy of the MED with topological sort order after informational thinning.

than 20 objects can easily fit on one computer screen), we will provide a set of less complicated and smaller subhierarchies of the original IS-A hierarchy to simplify the comprehension process. To realize this target, another approach, partitioning, is introduced.

**Definition 2:** *partitioning* means to divide a complex large semantic network into disjoint smaller subnetworks which comprise logical units of the original network and jointly constitute the original network. □

To partition a graph into logical units, it is necessary to comprehend it first. Thus, the logical partitioning of a network seemingly results in a vicious cycle. Our experience has been that partitioning into logical units tends to minimize the number of relationships between different units. Unfortunately, the problem of partitioning a network according to the above or similar criteria is NP-complete, that is, no efficient algorithm is known for it, and it is conjectured that no such algorithm exists [12]. A possible line of action is to combine informational thinning and partitioning. After an IS-A hierarchy is obtained by applying informational thinning to the original network, the partitioning technique is put to use by partitioning the IS-A hierarchy and then imposing this partition on the original network.

Due to multiple inheritance, the IS-A hierarchy forms a directed acyclic graph, just like our cortisporin subhierarchy shown in Fig. 1. If the IS-A hierarchy is a directed acyclic graph, its partitioning problem in general is still NP-complete. On the other hand, if it is a tree, then there exist efficient algorithms for various partitioning criteria, e.g. max–min or min–max [1,3–5,19,22,29]. In the next section, to make the partitioning possible, we will present a new technique for modeling called *disciplined modeling*. Based on the rules of disciplined modeling we develop a theoretical paradigm and methodology to identify a meaningful forest subhierarchy within the IS-A hierarchy. If the trees in the forest hierarchy are still too large, the above mentioned efficient partitioning algorithm may be applied to them, to yield smaller trees.

### 3. Theoretical paradigm using disciplined modeling

In order to identify a meaningful forest subhierarchy of an IS-A hierarchy, we shall look into the nature of the specialization IS-A relationship. In previous OODB research [13], we and others [25] have identified two different kinds of SUBCLASS relationships between object classes, called *category-of* and *role-of*. Both are specialization relationships. *Category-of* relates the specialized class to the more general class where both are seen in the same application context. *Role-of* relates the specialized class to the more general class where the two classes are in different contexts of the application.

In [28] we presented a theoretical paradigm for partitioning of an OODB hierarchy schema. However, modifying the theoretical paradigm from the class level

[28] to the instance level requires careful examination. One issue is how to interpret *category-of* and *role-of* at the instance level. Naturally, these relationships between classes imply *category-of* and *role-of* relationships between objects which are instances of the corresponding subclass and superclass. Similarly, they can be defined between objects of a semantic network as follows.

**Definition 1:** *category-of* is a specialization relationship which relates the specialized object to the more general object where both are seen in the same application context. □

**Definition 2:** *role-of* is a specialization relationship which relates the specialized object to the more general object where the two objects are in different contexts of the application. □

For example, **Aminoglycoside Preparations** is *category-of* **Antibiotic Preparations** and **Neomycin preparations** is *category-of* **Aminoglycoside Preparations**, because all of them are in the same application context ‘Anti-Infective Agents.’ On the other hand, **Neomycin preparations** is *role-of* **Drug Enforcement Agency (DEA) Class 0-Drug Without Abuse Potential** in the context of Drug Enforcement Agency (DEA) Controlled Substance Category (Figs. 6 and 7).

A second issue is that in [28] we discussed a relation ‘represents the same real-world object’ between instances of classes. However, in a semantic network-based vocabulary, objects describe general concepts rather than concrete, real-world objects. Therefore, we need to find an alternative for the relation ‘represents the same real-world object’ to be employed in the necessary proof for the vocabulary environment. The impact of this difference on the development of our theoretical paradigm has to be inspected. An adapted proof technique is presented later in this section.

The decision whether a given IS-A relationship in the hierarchy is either a *category-of* or a *role-of* depends on whether the superobject and object are in the same context or not. An intuitive understanding of the application is required to help make this decision. However, this decision is not always so easy. In spite of extensive research [6,7,16,18,23,24,32], there is still no widely accepted definition of context. Building a gigantic knowledge base in the CYC project [20] was found doomed to failure if contexts were not introduced as a structuring mechanism. Following the research of [6,7], others have assumed that a context is a first-class object used to parameterize axiom schemata [16,23]. However, no clarity about the nature of contexts themselves is gained by this approach. As a workshop on context in Natural Language Processing showed [18], researchers tend to agree that they disagree on what contexts are. Our approach is that we are not trying to define the notion of context. Rather we are making the pretheoretical (axiomatic) assumption that contexts exist in human thinking, and we are requiring the designers and users of an application to identify them explicitly.

In [28], we provided a theoretical paradigm for the existence of such assignments of classes to contexts. This assignment results in a forest subhierarchy of a directed

acyclic graph hierarchy, which supports increased comprehension of the OODB schema. The theoretical paradigm is supported by three rules of disciplined modeling which ensure that a forest subhierarchy can be identified. However, while in [28], disciplined modeling was described for a schema of classes, we modify it now for a hierarchy of objects. This modification will provide us with a theoretical paradigm for partitioning a large complex hierarchy of objects into small parts, each of which has a tree structure. For further explanations and motivations on disciplined modeling beyond the material in this section, see [28].

Before we give the rules of disciplined modeling, we define the mathematical relation *equicontext*, or ‘in the same context,’ between objects. A pair  $(a, b)$  of two objects belongs to the equicontext relation if both objects  $a$  and  $b$  belong to the same context.

**Rule 1:** The equicontext relation between objects is an equivalence relation satisfying three conditions of reflexivity, symmetry and transitivity. Thus it partitions all objects of a network into disjoint contexts.  $\square$

**Rule 1** forces the designer into explicit specification of the contexts in his hierarchy and leads him to resolve some ambiguous situations. We do not claim to have a unique way of assigning objects to contexts. As we are dealing with a problem of data modeling, there are usually different ways to model the same real-world environment. We further do not claim that contexts are naturally disjoint. To the contrary, in many applications, initial contexts may overlap. However, disciplined modeling forces the modeler to design disjoint contexts, leading to the desired partitioning.

**Rule 2:** Two objects which are *category-of* specializations of a superobject cannot have a common *category-of* descendant object, and one cannot be a *category-of* descendant of the other.  $\square$

According to our definition, the *category-of* relationship is used for refinement in the case where the superobject and the object are in the same context. **Rule 2** guarantees that when we refine a concept represented by an object into several subconcepts in the same context, we achieve a partition into mutually exclusive concepts.

In the next section, we discuss techniques how to specify IS-A relationships as *category-of* or *role-of* in different cases in a way which satisfies **Rule 2**. Examples are provided in Section 5.

**Rule 3:** For each context there exists one object which is the *major* (or defining) object for this context such that every object in this context is a descendant of this object.  $\square$

This means that each context has only one object which is a ‘root’ for it, i.e. there is a directed path of *category-of* relationships from each object in the context to this

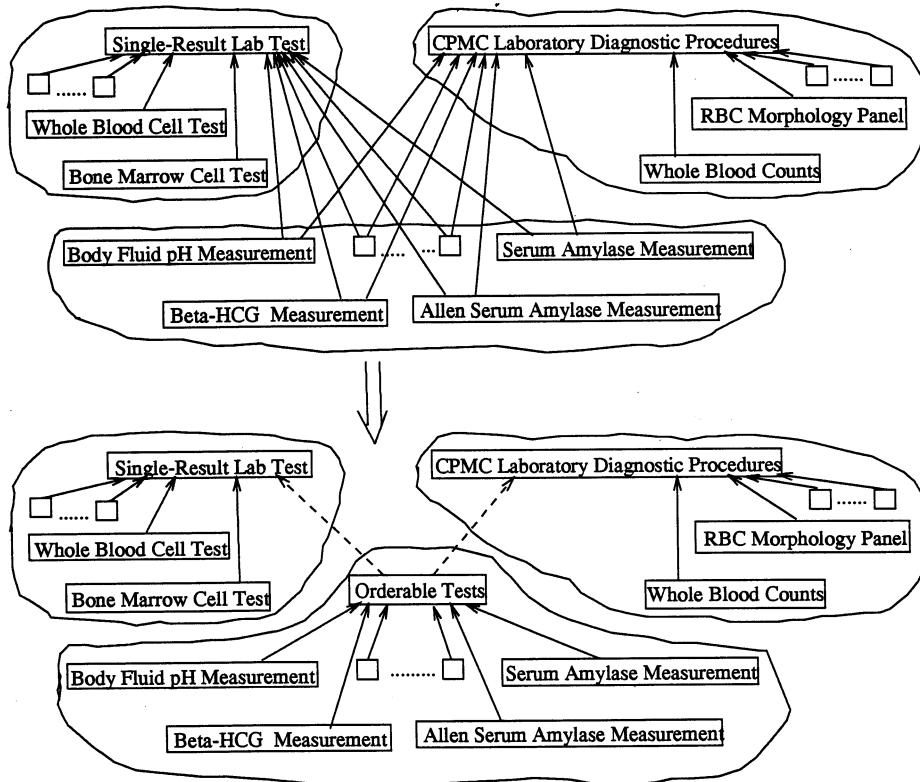


Fig. 2. Adding new object as the root of a context.

root object. Note that we use here the notion of a directed tree where all the directions are towards the root. In graph theory terms, the root is a sink.

We note that sometimes in a semantic network the designer would like to group a subnetwork which has several roots, rather than one, together into one context. In such a case, the designer can add an extra object and make these original roots children of the extra object. The new root will be named to reflect the ‘meaning’ of its context. For example, there are many terms in the MED for procedures which doctors order. Thus, these terms are grouped into one context ‘procedure.’ There are also many terms in the MED for tests grouped into a context ‘tests.’ Tests are typically ordered as components of procedures. However, in some cases, a component can be ordered by itself and such a test therefore has the properties of a procedure (e.g. an order code, a cost, etc.). Thus, all tests which cannot be ordered separately will reside in the ‘test’ context. All other tests which can be ordered separately will be grouped into another context ‘orderable test,’ because all of them have properties of tests and procedures at the same time. The context ‘orderable test’ has many roots. It has been helpful to introduce a new object, **Orderable Tests**, as the root of this context to keep track of those tests. All those tests become children of **Orderable Tests** (Fig. 2).

In a previous paper [28], we already proved the following theorem: ‘Using disciplined modeling, a *class* has at most one *category-of* superclass.’ Now, we need to prove the corresponding theorem for the object level.

**Theorem:** Using disciplined modeling, an *object* has at most one *category-of* superobject.

**Proof:** Assume to the contrary that there exists an object  $a$  which has two *category-of* superobjects  $b$  and  $c$  (see Fig. 3). According to the definition of *category-of*,  $a$  and  $b$  are in the same context. Similarly,  $a$  and  $c$  are in the same context. By the transitivity of the equicontext relation (**Rule 1**),  $b$  and  $c$  are in the same context.

By **Rule 3**, there is a major (root) object  $d$  for this context such that the objects  $b$  and  $c$  are *category-of* descendants of  $d$ . This implies that there is a sequence of *category-of* relationships from  $b$  ( $c$ ) up to  $d$ . (Note that actually  $d$  may be one of the objects  $b$  or  $c$ . This case does not cause a problem due to the second possibility in **Rule 2**. However, we avoid referring to this option in the rest of the proof to avoid complication of the presentation). If the paths of *category-of* relationships from  $b$  to  $d$  and from  $c$  to  $d$  are not disjoint (i.e. the object  $d$  is not the first object which appears in both paths), then denote now by  $d$  the first such joint object on these two paths. Let  $e$  ( $f$ ) be a subobject of the object  $d$  on a path of *category-of* relationships from  $b$  ( $c$ ) to  $d$ . Hence, object  $a$  is a *category-of* descendant of object  $e$  ( $f$ ). Thus, both the *category-of* subobjects  $e$  and  $f$  of the object  $d$  have a common *category-of* descendant object  $a$ . But by **Rule 2**, such a situation is forbidden, a contradiction.  $\square$

Due to this theorem, we can guarantee that the *category-of* hierarchy has a forest structure which contains one or more trees. This forest structure serves as backbone of the hierarchy and will be critical in the efforts to comprehend the hierarchy and partition it into manageable subhierarchies.

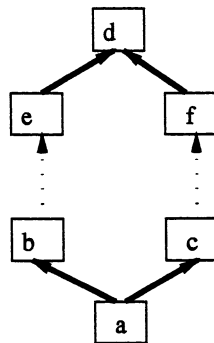


Fig. 3. The hierarchies demonstrating our proof.

#### 4. A methodology for context partitioning of a hierarchy

We have described a conceptual framework which guarantees that for the price of following the rules of disciplined modeling, there can be found a forest-structure subhierarchy of the given directed acyclic graph hierarchy. This forest structure serves as a skeleton supporting the comprehension of the hierarchy. Furthermore, the trees of the forest represent contexts which are logical subhierarchies concentrating on a specific subject area, further supporting the comprehension of the original hierarchy.

In this section, we will describe a methodology to transform an existing hierarchy which was not designed according to the rules of disciplined modeling. By a methodology we mean a process that involves human-machine cooperation. The human domain expert is called upon to make some judgment decisions based on his understanding of the application, while the computer supports the human by providing results of algorithmic procedures for tasks which do not involve complex intuitive decisions but might require many computational steps. By domain expert judgment we refer to:

1. Identifying disjoint contexts in the hierarchy, which correspond to subtrees of *category-of* relationships in the forest structure obtained.
2. Defining some IS-A relationships as *role-of* and others as *category-of*, so that the rules of disciplined modeling are followed.

In the following description of the methodology, we will specify which parts are performed by a computer and which are performed by a human expert. We will differentiate between three kinds of *role-of* relationships. They are regular *role-of*, *role-of/intersection* and *role-of/category-of*. However, for partitioning purposes, they are all just *role-of*. The result of our methodology is a refinement of the IS-A hierarchy. Every IS-A link becomes either a *category-of* or a *role-of*. For the purpose of partitioning, the *category-of* links will form a forest.

**Step 1:** Topological sort (Computer).

Arrange the hierarchy in topological sort order.

**Step 2:** Identify roots of contexts. (Human)

Scan the hierarchy top-down according to the order from **Step 1**. In this scanning, identify objects which should serve as defining objects (roots) for contexts. The choice should be made by the meaning and importance of the object in the application compared to its superobjects' meaning. These chosen objects start new contexts rather than refining the contexts of their superobjects.

After these objects are identified, they are *role-of* their superobjects. This kind of *role-of* relationship is a regular *role-of*, where the relationship models a switch of context, that is, the relationship goes from an object in one context to an object in another context.

**Step 3:** Multiple superobjects (Computer).

List all objects with multiple superobjects in bottom-up order. In the discussion following **Step 4** and at the end of this section, we will explain why we are using bottom-up processing at this point.

**Step 4:** Identify primary parent. (Human)

For each of the objects identified in **Step 3**, the expert needs to identify at most one superobject which is in the same context as the object. The relationship to this superobject will be defined as a *category-of* relationship while all other superobjects should belong to different contexts from the ‘chosen’ superobject, and the relationships to them are defined as *role-of*.

From our experience, for most of the objects with multiple superobjects an expert can easily determine which of the superobjects is the defining one, i.e. which should be in the same context and have a *category-of* relationship directed to it. There is a minority of cases where the decision about a major or definitional superobject of a given object is not easy. In such cases, we try to distinguish which of the several superobjects, if any, should have a *category-of* relationship pointing to it, based on the partial context information we have already accumulated in our bottom-up processing.

We distinguish several cases.

**Case 1:** One of the superobjects is definitional while the others are functional. For example, drugs can be classified by the chemicals that they contain (definitional) and by their therapeutic uses (functional). Then we look at the context to which the object and its descendants belong. (This is the reason for the bottom-up processing). We try to determine whether the nature of the *category-of* relationships is functional or definitional. If it is definitional, we will prefer the definitional superobject. If it is functional, then we will prefer the functional superobject (or if there are several functional superobjects, we will prefer the one which fits the function appearing in the context of the object). If the object is the only object in its context, we will choose the definitional superobject. In this case, one superobject is chosen as primary superobject. The object is *category-of* this primary superobject and *role-of* the other superobjects. This kind of *role-of* relationship is a regular *role-of* since a switch of context from superobject to object has occurred.

**Case 2:** All superobjects are definitional with the same importance or indistinguishable importance as each of them contributes to the definition of the object in an equal or indistinguishable way. In such a situation, the object with multiple superobjects could belong to the context of any of its superobjects. However, by the **Rule 1** it cannot belong to more than one context. Also, we have no reason to prefer one over the other. Each choice of context will disassociate the object from the other contexts. This conflict is resolved by requiring that such an object start a new context which represents the concept obtained as intersection of the concepts of all its superobjects. Thus, this object is *role-of* all its superobjects. We call this type of *role-of* ‘*role-of/intersection*’ represented as r/i in the figures. By this term, we emphasize that this is not an actual case of a switch of context but an artificial case

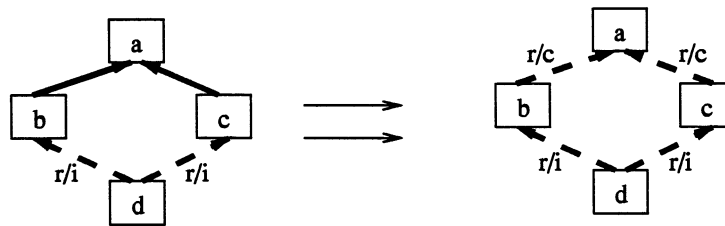


Fig. 4. The diamond structure.

due to the requirement of the theorem to forbid two *category-of* superobjects. Without the theorem, we could probably leave the intersection concept in the context of its superobjects if all belong to one context.

**Case 3:** The concept of the object is a combination of the concepts of the multiple superobjects in different contexts, but one of them contributes more to the meaning than the others. Then the *category-of* relationship should point to the preferred superobject, as those two should belong to the same context, while the other relationships should be *role-of* relationships.

**Step 5:** Identify diamond structures (Computer).

Scan the hierarchy according to the topological order bottom-up to find all the objects with more than one superobject. For each such object  $a$  and for each pair of superobjects  $s_1$  and  $s_2$  of  $a$ , find a lowest common ancestor  $b$  of both  $s_1$  and  $s_2$ . For each pair of such objects  $a$  and  $b$ , output the diamond or extended diamond structure (represented by  $\langle a, b \rangle$ ) containing  $a, b$  and all the objects which are descendants of  $b$  and ancestors of  $a$ . The object  $a$  is called the source of  $\langle a, b \rangle$ , and the object  $b$  is called the sink of  $\langle a, b \rangle$ .

**Step 6:** Diamond cutting (Human).

Each diamond or extended diamond structure must contain objects from more than one context in order to fulfill **Rule 2** of disciplined modeling. After executing the first five steps we discussed above, all diamond structures already satisfy **Rule 2**. But there is one case where we must artificially change the *category-of* relationships to *role-of* relationships, to resolve a contradiction.

In this case, which we call *contradictory diamond case*, the source  $d$  of the diamond structure  $\langle d, a \rangle$  is a *role-of/intersection* of its superobjects. All other objects in the diamond structure belong to one context (Fig. 4). Since the source  $d$  is the intersection of two superobjects  $b$  and  $c$ , they cannot both belong to the same context of their superobject  $a$ . Otherwise, because the intersection of a context with itself will result in the original context, the intersection must belong to this common context. Thus, the objects  $b$  and  $c$  are also defined as separate contexts. The *category-of* relationships are changed, due to **Rule 2**, to *role-of*. However, we want to maintain the distinction between this *role-of* and the two other kinds. Therefore, we denote this kind of *role-of* as '*role-of/category-of*.' It is represented by  $r/c$  in the figures. This concludes the six steps of our methodology.

Note that the methodology used both top-down processing and bottom-up processing. The determination of the context of objects is performed top-down, as every context root itself has a top-down nature, since the context of the root concept defines the context of its descendants. When scanning the hierarchy top-down, an expert can identify where an object defines a new context rather than continuing a context of one of its superobjects which has been processed already.

On the other hand, when determining bottom-up to which context an object belongs, choosing from among its superobjects, it is important to know the descendants of the object which belong to the same context. This knowledge will help to determine which of the contexts of the superobjects fits best to the already constructed context.

## 5. Applying the methodology to a complex hierarchy

In order to test the effectiveness of our methodology, we applied it to the previously mentioned cortisporin subnetwork of the MED. First, informational thinning was used to obtain a directed acyclic graph hierarchy out of this subnetwork (Fig. 1). Then we used the methodology introduced in the previous section to partition the hierarchy into trees. Each tree produced by the partitioning is a logical unit in the forest hierarchy. The root object of a tree defines the unifying context for the objects in that tree.

**Step 1**, topological sort, is applied to the hierarchy of Fig. 1. Since there are degrees of freedom in applying topological sort to a directed acyclic graph, the order we used is from left to right and breadth first search [2]. The object numbering from 1 to 46 in Fig. 1 reflects this order.

In **Step 2**, following the topological sort ordering, the domain expert scans the hierarchy top-down to find all objects which define new contexts. All IS-A relationships from these objects to their superobjects are defined as *role-of*.

Note that by specifying IS-A relationships as *category-of* or *role-of*, the designer is making modeling decisions, which may differ from one designer to another, influencing the emerging contexts. Modeling decisions made by a pharmacist will differ from those made by a surgeon. Thus, each of them can create his own local partitioning of the vocabulary which represents his view of the vocabulary. In our consideration in this section, we try to take the inclusive approach of a vocabulary administrator (VA).

Because the concept **Medical Entity** (1) is the unique root for all other concepts in the MED, it starts a new context and it does not have any *category-of* or *role-of* superobjects.

Following topological sort order, we can see the concept **Conceptual Entity** (2), which is a straightforward specialization of its superobject **Medical Entity** (1), in the same context as (1). Thus, concept (2) is *category-of* its superobject (1). The concepts **Intellectual Product** (3), **Orderable Entity** (4), **Classification** (5), **Pharmacy**

**Concepts (6)**, and **American Hospital Formulary Service Class (7)** are all straightforward specializations of their superobjects. In our intuitive judgment, they are also similar in nature to their superobjects, and thus they are in the same context as their superobjects. Thus, all of them are *category-of* their superobjects.

The concepts **Drug Enforcement Agency (DEA) Controlled Substance Category (8)**, **Drug Allergy Class (9)** and **CPMC Formulary Drug Forms (10)** have only one superobject **Pharmacy Concepts (6)** which is a broad term that refers to various ways of grouping drug concepts. But concept (8) defines drug concepts that are controlled by the DEA, while concept (9) is a group of drug concepts that have allergic or antiallergic effects, and concept (10) refers to the dispensation form (tablet, injectable, etc.) of the drug. Thus, all of these objects represent drug classifications according to various new dimensions and are considered to be defining objects for new contexts. All of them are regular *role-of* children of their superobjects.

The concept **Pharmacy Items (11)** has two superobjects. One is **Pharmacy Concepts (6)** which was analyzed above; the other is **Orderable Entity (4)** which describes a heterogeneous group of concepts that can be ordered and may be pharmacy or non-pharmacy concepts. Thus, the concept (11) is defined as a root object for a new context, as it is a classification according to a new dimension. It is a regular *role-of* its two superobjects.

The concept **Anti-Infective Agents (17)** has two superobjects which are **CPMC Formulary Drug Item (16)** and **American Hospital Formulary Service Class (7)**. Both superobjects define formulas of various pharmacological preparations and contribute their own formulations (one from CPMC and the other from American Hospital Formulary) to the concept. The concept (17) is neither in the same context as (16) nor (7). It is *role-of* both superobjects and starts a new context. The concepts **Eye, Ear, Nose and Throat Preparation (18)**, **Hormones and Synthetic Substitutes (19)**, **Anti-Inflammatory Agents (20)** and **Skin and Mucous Membrane Agents (21)** all require the same analysis as (17). All are *role-of* their superobjects and are root objects for their contexts.

Consider the concept **Glucocorticoid Agents (29)**; it has two superobjects, **Adrenal Agents (24)** and **Anti-Inflammatory Agents (20)**. Glucocorticoid Agents are secreted by adrenal glands and therefore the superobject (24) indicates the physiological source for the Glucocorticoid group of agents. Another child of the concept (24) describes Mineralocorticoids (Aldosterone) (not shown in the figure) which are functionally distinct from Glucocorticoids. (20) describes a heterogeneous set of concepts that includes steroidal anti-inflammatory drugs like Glucocorticoids and non-steroidal anti-inflammatory agents like Aspirin, Ibuprofen, Indomethacin and Phenylbutazone Preparations. Therefore, (29) starts a new context and is *role-of* its two superobjects.

Let us check the concept **Polymyxin B Preparations (35)** which has one superobject **Miscellaneous Antibiotics (27)**. The superobject (27) describes a heterogeneous group of antibiotics that belong to chemical families that do not fall into the major antibiotic families like **Penicillins**, **Cephalosporins**, **Aminoglycosides**, etc. Some of the subobjects of (27) are **Vancomycin** (a glycopeptide), **Bacitracin Preparations** (a



In order to improve the clarity of the presentation and to eliminate complicated medical terms, we will sometimes use only the topological sort numbers to represent these medical terms in the balance of this section.

**Step 3** of our methodology is to find all the objects which have more than one superobject in bottom-up order (reversing the order of the topological sort in Fig. 1). These objects are (46), (44), (42), (40), (38), (37), (36), (32), (31), (30), (29), (28), (26), (25), (23), (21), (20), (19), (18), (17), and (11).

Because the number of primary superobjects for each object with multiple parents is at most one, the domain expert needs to identify at most one primary parent for each object listed above, in **Step 4** of the methodology. For example, the concept **CPMC Drug: Cortisporin Ophthalmic Ointment** (46) has three superobjects, **Bacitracin/Hydrocortisone/Neomycin/Polymyxin B Combination Preparations** (44), **Drug Dispensed by Gram** (15) and **Eye, Ear, Nose and Throat Antibiotics** (28). The superobject (44) defines the chemicals that form the **Cortisporin Ophthalmic Ointment**. They uniquely define the structural components of the ointment, and therefore by **Case 1** of **Step 4**, (44) is the primary superobject of (46). The superobject (15) specifies the mode of dispensation and the superobject (28) specifies the site and action, and therefore both do not define the context of the concept. Thus, according to **Case 1** of our methodology, (46) is *category-of* (44), *role-of* (15) and *role-of* (28).

Let us check another object which has more than one superobject. The concept **Bacitracin/Hydrocortisone/Neomycin/Polymyxin B Combination Preparations** (44) has two superobjects, **Bacitracin/Neomycin/Polymyxin B Combination Preparations** (40) and **Hydrocortisone/Neomycin/Polymyxin B Combination Preparations** (42). Both superobjects contribute two chemicals common to both concepts (Neomycin and Polymyxin B) to the concept. In addition, (40) contributes Bacitracin and (42) contributes Hydrocortisone. All these chemicals together define the concept (44). According to **Case 2** of **Step 4**, it is not possible to identify the primary superobject. Hence it is *role-of* its superobjects. As we defined before, this kind of *role-of* is *role-of/intersection*. In Fig. 6, we marked this *role-of* as ‘r/i’ to distinguish it from a regular *role-of*. A similar analysis can be applied to all concepts that are roots of drug combinations like (40), (42), (38) and (37).

Another example is the concept **Bacitracin Preparations** (31) which has two superobjects, **Miscellaneous Antibiotics** (27) and **Drug Enforcement Agency (DEA) Class 0-Drug Without Abuse potential** (12). We already analyzed the superobject (27) in **Step 2**. (31), which is a polypeptide, is a child of (27). It is in the same context with its superobject (27). The other superobject (12) simply indicates a classification for the DEA according to a drug’s abuse potential and is not a definitional superobject for the concept. Thus, (31) is *category-of* (27) and *role-of* (12).

After **Step 4** is completed, none of the objects with multiple superobjects in Fig. 6 has more than one primary parent. That means that each object is *category-of* at most one superobject.

Now we need to identify the diamonds or extended diamonds structures in bottom-up order. As discussed above, we use a pair  $\langle A, B \rangle$  to denote a diamond structure with A as source and B as sink. The (extended) diamond structures in Fig.



Now, let us examine  $\langle (37), (27) \rangle$ . As a result of **Step 2** and **Step 4**, both concepts (31) and (35) would be *category-of* concept (27). The source concept (37) is *role-of/intersection* of two superobjects (31) and (35) according to the result of **Step 4**. This diamond structure is a contradictory diamond case as described in **Step 6**. Thus, at least one of the superobjects (31) and (35) must be made *role-of/category-of* the superobject (27). Since both concepts (31) and (35) are in the same configuration that we encountered before, we cannot choose only one to be *role-of/category-of* their superobject. Thus, both of them now are *role-of/category-of* their superobject. In Fig. 6, the *role-of/category-of* relationship is represented as r/c. This is the only diamond structure in Fig. 1, for which the contradictory diamond case of **Step 6** holds. In this way, we represent the knowledge that both concepts (31) and (35) were separated from their parent's context just to fulfill the requirements of **Rule 2**. But for other purposes, they and their *category-of* descendants may be considered part of the context to which the concept (27) belongs.

After all IS-A relationships in Fig. 1 have been changed to *category-of* or *role-of*, the forest subhierarchy of the original subnetwork is obtained by removing all the *role-of* relationships. Fig. 7 shows all contexts as trees in the forest. The relationship between objects of different contexts (trees) is *role-of*.

The hierarchy in Fig. 1 is partitioned into 18 contexts, many of which are very small and seem to be too detailed. But note that this is not a typical subnetwork of the MED. By choosing a subnetwork with a very complex hierarchy we ended up with a network with many interrelated subjects. Furthermore, even the contexts shown in Fig. 7 are not complete since some terms which belong to these contexts are not shown as they are not ancestors of (46). To demonstrate this, we added in Fig. 7 some of those extra concepts A, B and C representing **CPMC Drug: Polysporin Ophthalmic Ointment 3.5 Gm**, **CPMC Drug: Polysporin Topical Ointment 30 Gm**, and **CPMC Drug: UD Polysporin Ointment**.

We applied our methodology to the InterMED (an offshoot of the MED) containing about 3000 concepts. It was partitioned into 545 contexts, 394 of them consisting of single concepts due to the InterMED's incompleteness (i.e. if more concepts of the MED would be added to the InterMED, then some of these singleton concepts would get descendants and turn into actual contexts). Thus, the InterMED is practically partitioned into 151 actual contexts with an average size of 16. This partition of the InterMED achieves our original goal of partitioning the vocabulary into screen-sized, logical units reasonably comprehensible to a user.

## 6. Conclusions

Vocabularies promise to be important tools for many medical information processing tasks. They can help overcome differences in terminology between different databases and information systems and different categories of users. Unfortunately, the job of understanding and maintaining the vocabulary itself is difficult and time-consuming. A graphical representation can help in the process of understanding most vocabularies. However, if the vocabulary is very large, the



pairs of contexts. This presents a major improvement over studying ‘the part of the vocabulary that is just now displayed in the window.’

Two experiments with the methodology were presented. The first one used a very complex subnetwork of the MED vocabulary, which poses a challenge due to its complexity. The second one used the InterMED, a medium sized vocabulary. Both experiments demonstrated the effectiveness of the methodology.

To date, we have only anecdotal evidence that the partitioned vocabulary is easier to use than the original source vocabulary. We are planning a human-factors evaluation of the results of our methodology using students in the Biomedical Informatics program at the University of Medicine and Dentistry of New Jersey. We expect that such a study will show that students with access to our partitioned vocabulary will solve a given problem faster and more accurately than students in a control group.

### Acknowledgements

We thank Jim Cimino for his important feedback on earlier drafts of this paper.

### References

- [1] Agasi E, Becker R, Perl Y. A shifting algorithm for constrained min–max partition on trees. *Discrete Appl Math* 1993;45:1–28.
- [2] Aho AV, Hopcroft JE, Ullman JD. *Data Structures and Algorithms*. Reading, MA: Addison-Wesley, 1983.
- [3] Becker R, Perl Y. Shifting algorithms for tree partitioning with general weighting functions. *J Algorithms* 1983;4:101–20.
- [4] Becker R, Perl Y. The shifting algorithm technique for the partitioning of trees. *Discrete Appl Math* 1995;62:15–34.
- [5] Becker R, Perl Y, Schach S. A shifting algorithm for min–max tree-partitioning. *J ACM* 1982;29:56–67.
- [6] Buvac S, Fikes R. A declarative formalization of knowledge translation. In *CIKM '95*, Baltimore, MD, 1995:340–347.
- [7] Buvac S, Mason IM. Propositional logic of context. In *Proceedings of the 11th National Conference on Artificial Intelligence (AAAI-93)*, Washington, DC, 1993:412–419.
- [8] Cimino J, Barnett G. Automated translation between medical terminologies using semantic definitions. *MD Comput* 1990;7:104–9.
- [9] Cimino J, Clayton P, Hripcsak G, Johnson S. Knowledge-based approaches to the maintenance of a large controlled medical terminology. *JAMIA* 1994;1(1):35–50.
- [10] Cimino J, Hripcsak G, Johnson S, Clayton P. Designing an introspective, controlled medical vocabulary. In: Kingsland LC, editor. *Proceedings of the Thirteenth Annual SCAMC*, Washington, DC. IEEE Computer Society Press, 1989:513–518.
- [11] College of American Pathologists, Skokie, IL. *Systematized Nomenclature of Medicine*. Second edition, 1982.
- [12] Gary M, Johnson D. *Computers and Intractability*. New York: Freeman, 1979.
- [13] Geller J, Perl Y, Neuhold E. Structure and semantics in OODB class specifications. *SIGMOD Record* 1991;20(4):40–3.

- [14] Gu H, Cimino J, Halper M, Geller J, Perl Y. Utilizing OODB schema modeling for vocabulary management. In: Proc. '96 AMIA Annual Fall Symposium, Washington, DC. 1996:274–278.
- [15] Gu H, Perl Y, Geller J, Halper M, Cimino J, Singh M. Partitioning a vocabulary's IS-A hierarchy into trees. In: Proc. '97 AMIA Annual Fall Symposium, Nashville, TN, 1997:630–634.
- [16] Guha RV. Contexts: a Formalization and Some Applications, 1991.
- [17] Halper M, Geller J, Perl Y, Neuhold E J. A graphical schema representation for object-oriented database. In: Cooper R, editor. Workshop on Interfaces in Database Systems (IDS-92). London: Springer, 1993:282–307.
- [18] Iwanska L. Context in natural language processing. In: Working Notes of Workshop W1.9, IJCAI. Montreal, Canada, 1995.
- [19] Kundu S, Misra J. A linear tree-partitioning algorithm. *SIAM J Comput* 1977;6:131–4.
- [20] Lenat D. CYC: A large-scale investment in knowledge infrastructure. *CACM* 1995;38(11):33–8.
- [21] Liu L, Halper M, Gu H, Geller J, Perl Y. Modeling a vocabulary in an object-oriented database. In *CIKM '96*, Rockville, Maryland, 1996:179–188.
- [22] Lucertini M, Perl Y, Simeone B. Most uniform path partitioning and its use in image processing. *Discrete Appl Math* 1993;42:227–56.
- [23] McCarthy J. Notes on formalizing context. In 13th International Joint Conference on Artificial Intelligence, Chambery, France, 1993:555–560.
- [24] Miller G A. Wordnet: A lexical database for English. *Commun ACM* 1995;38(11):39–41.
- [25] Neuhold EJ, Schrefl M. Dynamic derivation of personalized views. In *VLDB 88*, Long Beach, CA, 1988:183–194.
- [26] Oliver D, Shortliffe E. Collaborative model development for vocabulary and guidelines. In Proc. '96 AMIA Annual Fall Symposium, Washington,DC, 1996:826.
- [27] Perl Y, Geller J. Using object-oriented databases to make medical vocabularies comprehensible. *NJIT Res* 1997;5.
- [28] Perl Y, Geller J, Gu H. Identifying a forest hierarchy in an OODB specialization hierarchy satisfying disciplined modeling. In Proc. COOPIS '96, Brussels, Belgium, 1996:182–195.
- [29] Perl Y, Schach S. Max–min tree-partitioning. *J ACM* 1981;28:5–15.
- [30] Rector A. Coordinating taxonomies: Key to re-usable concept representations. In: Barahona P, Stefanelli M, Wyatt J, editors. *Artificial Intelligence in Medicine*. Berlin: Springer, 1995:17–28.
- [31] Rector A, Bechhofer S, Goble C, Horrocks I, Nowlan W, Solomon W. The GRAIL concept modelling language for medical terminology. *Artif Intell Med* 1997;9:139–71.
- [32] Shoham Y. *Varieties of Context Artificial Intelligence and Mathematical Theories of Computation*. London: Academic Press, 1991.
- [33] Shortliffe E, Barnett G, Cimino J, Greenes R, Huff S, Patel V. Collaborative medical informatics research using the internet and the World-Wide-Web. In Proc. '96 AMIA Annual Fall Symposium, Washington,DC, 1996:125–129.
- [34] United States National Center for Health Statistic, Washington, DC. *International Classification of Diseases, Ninth Revision, with Clinical Modifications*, 1980.
- [35] US Department of Health and Human Services, NIH, National Library of Medicine. *Unified Medical Language System*, 1996.