Some Requirements and Experiences in Engineering Terminological Ontologies over the WWW

Aldo Gangemi, Domenico M. Pisanelli, Gerardo Steve
Consiglio Nazionale delle Ricerche - ITBM, Viale Marx 15, Roma, Italy
{aldo,nico,geri}@color.irmkant.rm.cnr.it

Abstract

Sharing and reusing large subsets of the medical terminology is needed in various areas: knowledge-based systems, information retrieval, standardization, etc. The main obstacle to sharing and reusing medical terminologies is the lack of conceptual integration of terms. Actually the intended meaning of terms is different according to the context in which they appear and to the context of use.

Interdisciplinary research in ontology provides good evidence that use of generic ontologies specified from literature is the grounding matter for conceptual integration of terminologies.

Following our experiences in engineering a methodology for terminology integration, we suggest that the contextual dependency of terms should be overcome by means of a collaborative modelling environment, a distributed approach, an expressive language and a sound methodology.

ON9, our current medical ontology library, evolved using expressive languages like GRAIL, Ontolingua, Loom and OCML. It also took advantage from tools for the distributed negotiation of ontologies like Ontosaurus.

1. INTRODUCTION

In this paper we sketch out the motivations which led us to design a methodology for engineering terminological ontologies, and a description of the languages and tools that have been used to construct the current ON9 library of ontologies.

Sharing and reusing large subsets of the medical terminology is become a necessity in various areas: knowledge-based systems, information retrieval, standardization, etc. The main obstacle to sharing and reusing medical terminologies is the lack of conceptual integration of terms. Actually the intended meaning of terms is different according to the context in which they appear and to the context of use.

Interdisciplinary research in ontology provides good evidence that use of generic ontologies specified from the literature is the grounding matter for conceptual integration of terminologies. Detailed generic theories require rich languages and tools as well as collaborative effort to be extensively used.

Consequently, we strongly commit to groundedness of conceptualization (according to the definition given by [Har90]), expressive languages, modular architectures, and distributed tools for collaborative modelling.

Our research primarily concerns the integration and reuse of terminological ontologies in medicine. Terminological ontologies are crucial for activities such as vocabulary standardization [CEN95], natural language (lexical) processing, terminology server design [GAL92, Hum92], conceptual modelling of subdomains [Ros97], knowledge integration, sharing, and reuse [Gen1994, Fal94, Swa96, Val96], and multi-agent system development [Fal96].

Our source for building terminological ontologies are medical terminology systems. Most medical terminology systems do not have a terminological ontology, however this does not mean that terminology systems are not founded on a conceptualization, but only that their conceptualization is left to the interpretation of the experts who use the systems.

In the following we present our approach towards an explicit conceptualization of the domain ontologies of terminology systems. Our aim is to build a library of grounded terminological ontologies (of representation, generic, and domain kind).

We started working on medical language processing in 1989 and produced a schema for a machine-dictionary of medicine [Ros90], which provides a normalization of medical vocabulary by decomposing the morphological units of terms. Those efforts revealed that term normalization must be followed by a conceptual account of the normalized term [Gan92], namely decomposed terms required the explicitation of their intended meaning in order to provide a sensible conceptualization. For example, the term "viral hepatitis" can be decomposed in "vir-", "-al", "hepat-", "-itis" (the component morphemes), and we could normalize the components in "virus", "<adjective>", "liver", and "inflammation". However, we still need the interpretation of the relations among the components, and the classification of the components within a comprehensive domain terminology.

The explicitation of meaning was initially carried on in an informal way, for example by analyzing the work done in medical terminologies or so-called "coding systems". It was clear that a conceptualization for a term was context-dependent, where "context" had to be intended in a wide sense, including:

• the belief space of an agent which uses the term;
• the disciplinary domain in which the term is used;
• the region of space in which some experts use the term;
• the spatio-temporal situation in which the term is uttered;
• the historical use and evolution of the term;
• the text in which the term appears;
• the possible terminological repositories in which the term is classified.

Such context dependency (or "situatedness") convinced us to look for a broad perspective, including primarily a methodology for explicitating the conceptualization of a term only as far as its context requires, and secondarily languages and tools for implementing a conceptualization.

In the following we give: §2.: our proposal of some methodological issues to be supported in engineering terminological ontologies; §3. a practical description of the ONIONS methodology; §4. a brief overview of the WWW toolkits we have experimented with; §5. a sketch of our current ON9 ontology library, integrated from the terminological ontologies of five terminology systems.

2. METHODOLOGICAL ISSUES

The conceptualization activity — the activity providing a specification to terms — poses severe problems to modellers when concepts must be shared by different users in different contexts. Local conceptualizations are not suitable to support the tasks of making standards, writing guide-lines, reusing, integrating or sharing knowledge. To this purpose, we need:

(1) Procedures for capturing terminological knowledge: knowledge conceptualized in existing models may cover different areas, and this coverage is hardly predictable. Moreover, it is unclear how much coverage is needed in a standard model; terminological knowledge has various contextual constraints, and only the relevant ones should be conceptualized.
We need a methodology for capturing all and only the knowledge we need for a scope, and for tracing the borders among the different areas covered by different models, but not in the sense of managing conflicting knowledge in different areas or even inside the same area (this is not a terminological issue).

(2) Procedures for explicitating domain ontologies and the related generic ontologies: the intended meaning of concepts in a local conceptualization is tailored to the local needs, thus a different conceptualization might have a different intended meaning. Moreover, as far as standards or guidelines are concerned, a conceptualization has to be acceptable to an entire community, not only to a local task.
We need a methodology for conceptualizing the intended meaning of local conceptualizations under a unified, "multi-local" conceptualization. To this purpose, we require a library of generic ontologies to be constructed (see issue (6)). Also, we need procedures for reusing existing generic ontologies or formalizing other existing, but informal ones.

(3) A common, expressive language for expressing the resulting conceptualization: the language used is not neutral to the resulting conceptualization, thus different languages pose problems of translatability. A common language should have the expressivity to translate the constraints posed by other languages; it should be very expressive in any case.

(4) A viable implementation for concept classification: concepts defined in an ontology should be classified, for example according to the structural resemblance of their definitions [Mac94]. Structural resemblance is the most used strategy for concept classification within the description logics domain and poses hard problems of complexity; for example, it has been demonstrated that languages with a certain expressivity are recursively undecidable [Sch89]. On the other hand, common practice has given various arguments in support of a more liberal strategy, specially if "normal case" (instead of the "worst case") is adopted for testing tractability of a language (cf. [Spe95]).

(5) The explicitation of representation ontologies (a conceptualization of the intended meaning of the so-called "Meta-Level Categories" (MLC), like "class", "slot", "property", "relationship", etc.): an ontology uses formal languages which eventually result in additional constraints provided by the MLC of those languages. MLC are used with a different semantics in different languages and their interpretation is usually left to the intuition of the modeller. We need a semantic analysis of meta-level categories and good guidelines for applying them in the conceptualization activity.

(6) A library (modular) architecture for ontological theories: when the number of domain concepts exceeds a certain size, the maintenance of a unique domain ontology is very difficult, both from a computational and from

a conceptual viewpoint. The problem is even harder when a domain ontology is a specialization of a generic ontology library (indeed, as it should be), because generic knowledge might be specified in generic ontologies which are not compatible among them, *if they are taken as wholes*. For this reason, we need to be modular. Generic modules could be *included* as wholes in domain modules, or they could only provide some concepts to domain ontologies (they could be *used*).

(7) Guidelines for the distributed modelling of ontologies: ontological engineering requires some decisions that are somewhat arbitrary:

a) about the ontologies to be included in a library;
b) about the definitions to be included in generic ontologies;
c) about which definitions are to be specified from generic to domain ontologies.

Although a rationale for a)b)c) is supplied by a methodology which deals with issue (1) and (2), the actual decisions to be taken in the application of this methodology are preferably to be discussed among various groups or institutions; for example, among an expert of a subdomain, a knowledge engineer, a philosopher and an industrial partner, all involved in making a terminological standard for surgical device concepts.

(8) Tools for on-line availability of ontologies (browsing and editing): once we have a methodology, a library architecture and a distributed modelling activity, one should find the fittest tools for carrying out such an activity. The ideal toolbox should provide:

a) Internet-available ontology libraries;
b) remote on-line browsing and editing of ontologies and definitions of concepts, possibly with interfaces customized to the expertise level of the user;
c) an interactive tool for collaborative discussion about the libraries.

We examine issues (1)(2)(3)(5) in more detail in [Ste96, Ste97, Gan97]. A relevant effort in the direction of (3) has been done by [Gru93]. The problems in (5) have been studied by several authors [Gan96, Gru93, Gua94, Sow96]. The issues in (1) and (2) have received little attention in AI, until recent times [Usc95, Ste96, Val96]. Issue (4) is a classic subdomain of AI, taken into account by so-called description logics [Sch89, Bra91, Mac94, etc.].
In medicine, an important effort has being done in Europe by some CEN[1] committees which address issues (1)(2)(3)(5) at various degrees of depth. The CANON group [Eva94] mainly addressed (3); the MoSe pre-standard [CEN95] provides some guidelines for (1) and (2), other groups are writing standard conceptual systems in medical sub-domains dealing with the issues in (2) [Ros97]. The issue (2) has been treated in [Grü96, Ste96, Bor97], but mainly founds itself on the literature of naïve physics, linguistics and philosophy.
Issues (6), (7) and (8) are a trademark of the quite recent research in ontological engineering [Fal94, Far96, Swa96]. (7) is also investigated in the current continuation of GALEN, GALEN-IN-USE.

Some research projects have a position in most of the issues presented. We could classify such approaches to ontology modelling by means of several features related to the above issues:

features related to the conceptual tradeoff between:

• terminological coverage (the number of concepts defined, cf. issue (1)) and
• conceptual principles (the number and the size of generic theories exploited, cf. issue (2));

features related to the formal tradeoff between:

• expressivity (issue (3)) and
• tractability (issue (4)) of the language used;

other features related to:

• the presence of an explicit representation ontology (issue (5));
• the presence of a modular architecture for ontologies (issue (6));
• the distributed modeling of ontologies (issue (7));
• the on-line availability of ontologies (issue (8)).

In order to quantitatively synthesize the concern of these features for some research projects (with no claim of completeness or presision: this is only a general indication), we present our assessment in the graphs in Fig. 1,

---

[b] Comité Européen de Normation, the European standardization body, federating the National bodies.

2, and in the Tab. 1. The values of these sets of features are on a conventional scale from 0 to 1. Fig. 1 shows the assessment of features related to the conceptual tradeoff; fig. 2 shows the assessment of features related to the formal tradeoff; tab. 1 shows the "yes-no" assessment for other features.
Consider that the validity of such comparisons is relative: aims and contexts of different projects generate peculiar motivations; for example, only four of the nine systems listed have a main concern with terminological ontologies, which are a secondary aspect in the others. Anyway, some generalities can be described.
So called "bottom-up" approaches tend to privilege terminologic coverage, tractability of the language, syntactic simplicity, and to be more distributed, while so-called "top-down" approaches tend to privilege conceptual principles, expressivity of the language, metalinguistic exactness and are more modular. Some stay in the middle, getting the most from both approach types.

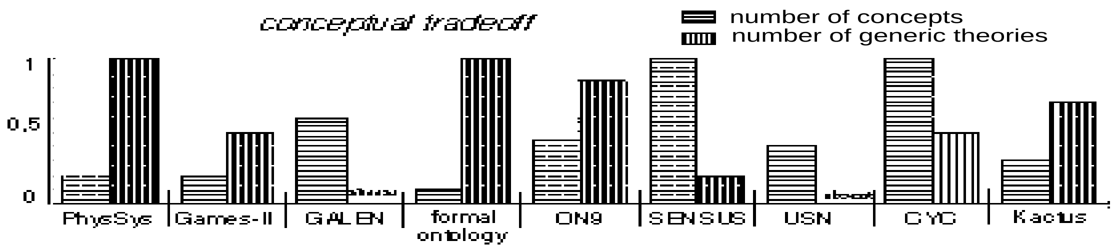

Fig.1: The conceptual tradeoff in some ontology systems. *GALEN* [GAL92], *ON9* [Gan97] and *USN* [Hum92] are specifically tied to treat medical terminological ontologies; *CYC* [Len90] and *SENSUS* [Swa96] are tied to treat a-specific terminological ontologies, specially for natural language processing; *Games-II* [Fal94], *Kactus* [Lar96] and *PhysSys* [Bor97] are mainly tied to domain knowledge-modelling ontologies (the first with application to medicine); *formal ontology* [for example, Brg96, Var95, etc.] is not a specific project, but rather a wide, interdisciplinary research program to provide solid bases to generic ontologies.

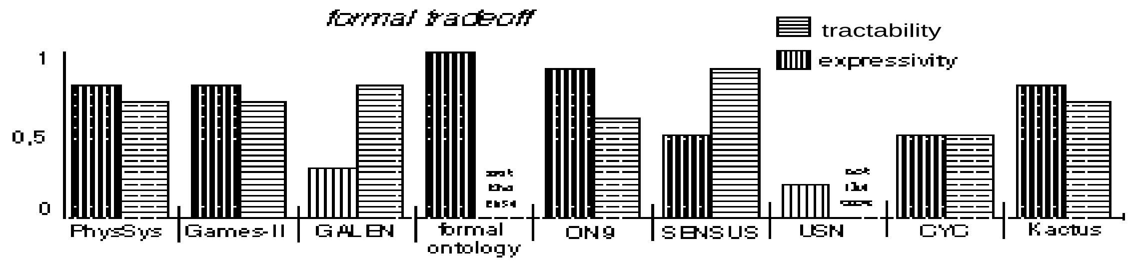

Fig.2: The formal tradeoff in some ontology systems

features | systems	PhysSys	GAMES	GALEN	formont	ON9	SENSUS	USN	CYC	Kactus
explicit representation ontol.	+	+	-	+	+	+	-	+	+
modularity	+	+	-	+	+	+	-	+	+
distributed modeling	-	+	+	-	+	+	-	-	+
on-line availability	-	+	-	-	+	+	+	-	-

Tab.1: Other methodological features of some ontology systems

3. THE ONIONS METHODOLOGY

A main concern of our research is to provide a terminological ontology to the most important terminology systems in medicine. To this purpose, we developed a methodology which addresses the above issues.
We initially defined ONIONS as a methodology to build the core model of a medical terminology server. In the context of the GALEN Project we developed our own methodology to integrate terminology systems by using the GRAIL description logic. Our approach has subsequently taken us to the design of ONIONS, with the goal of:

• building (or re-using) a library of generic ontologies by formalizing ontologies from the literature in AI, philosophy, linguistics, cognitive science;
• generating a domain ontology for each terminology system by including a subset of the generic library as the building blocks which have motivated the particular organization of the terminology system [Gan96, Ste96].

Once these processes are carried out, different domain ontologies can be integrated because they share the library of generic ontologies.

ONIONS then led to the successful integration of the most general concepts (more than five thousands) of five terminology systems [2]. A complete description is in [Gan97] and an account of the operative and philosophical requirements that motivated ONIONS design can be found in [Ste96].
We adopted Ontolingua and Loom as formalisms for representing the results of the integration of our terminology systems. Ontolingua [Gru93] — a language developed from KIF [Nec91] — allows expressivity for both frame-like and axiomatic constraints. Loom [Mac91 is a quite expressive implementation of a description logic with classification services.

The most relevant need to satisfy was to have an ontology open to revisions without giving maintenance troubles, together with a "buy-what-you-need" approach: if one only talks about anatomy, why inflating his/her ontology with all the stuff about chemicals? Such an approach also allows negotiability, i.e. if one does not agree on a certain part of a conceptualization, not the whole ontology has to be discarded.
Therefore we put an emphasis on modularization providing an architecture allowing alternatives and conflicts without loosing the reference to the generic ontologies that are included or used in the modules.

In Figure 3 we introduce in an abstract and schematic form the basics of ONIONS methodology.
The motivations why we aim at such a methodology and the related feasibility concerns are the matter of a wide discussion and are only briefly recalled here.
Here we describe the properties of a terminology system at different development states, thus it is largely independent from the issue if the phases we design are just the right ones to realize those properties.
Figure 3 is a schematic account of some the previous issues, which envisages a methodology with six phases and a set of input and output states in the building of a terminology system. Such states are described by a set of structural and ontological properties. We name a property "ontological" if it concerns the *principles* of a conceptual system.
Each ONIONS phase M_i makes a terminology system evolve from a state S_i into a state S_{i+1}. P_i and O_i are respectively the structural and ontological properties of S_i systems. Hence, such properties allow a classification of existing terminology systems according to their structural and ontological properties.
The methodology has been tested on relevant portions of ICD10, SNOMED-III, and GMN, and on the USN and GCM terminological ontologies. Other specialized corpora of medical terms have been conceptualized as well (e.g. surgical procedures [Ros97]). Currently we are extending the models to cover the entire UMLS Metathesaurus™.
Depending on the purpose of the integration, a terminology system may reach a state – e.g. S_4 – and stay there without needing a further evolution.
P and O properties do not just repeat the issues presented before, because methodological phases are designed to account mainly for issues (1)(2)(3)(5), and are motivated by the organization of existing terminology systems.

It should be emphasized that the lifecycle presented here is that of the ONIONS methodology and actual terminology systems might not follow it strictly. A system might stay in a status S_i without having passed the previous ones or it might be in a hybrid state where its structural property is P_j and its ontological property is O_k with $j \neq k$.

[2] The USN (all ~170 'semantic types' and relations, and their defined templates), SNOMED-III (~600 most general concepts), GMN (~700 most general concepts), ICD10 (~250 most general concepts), and the GCM (version 5g, all ~2000 concepts and 'attributes').

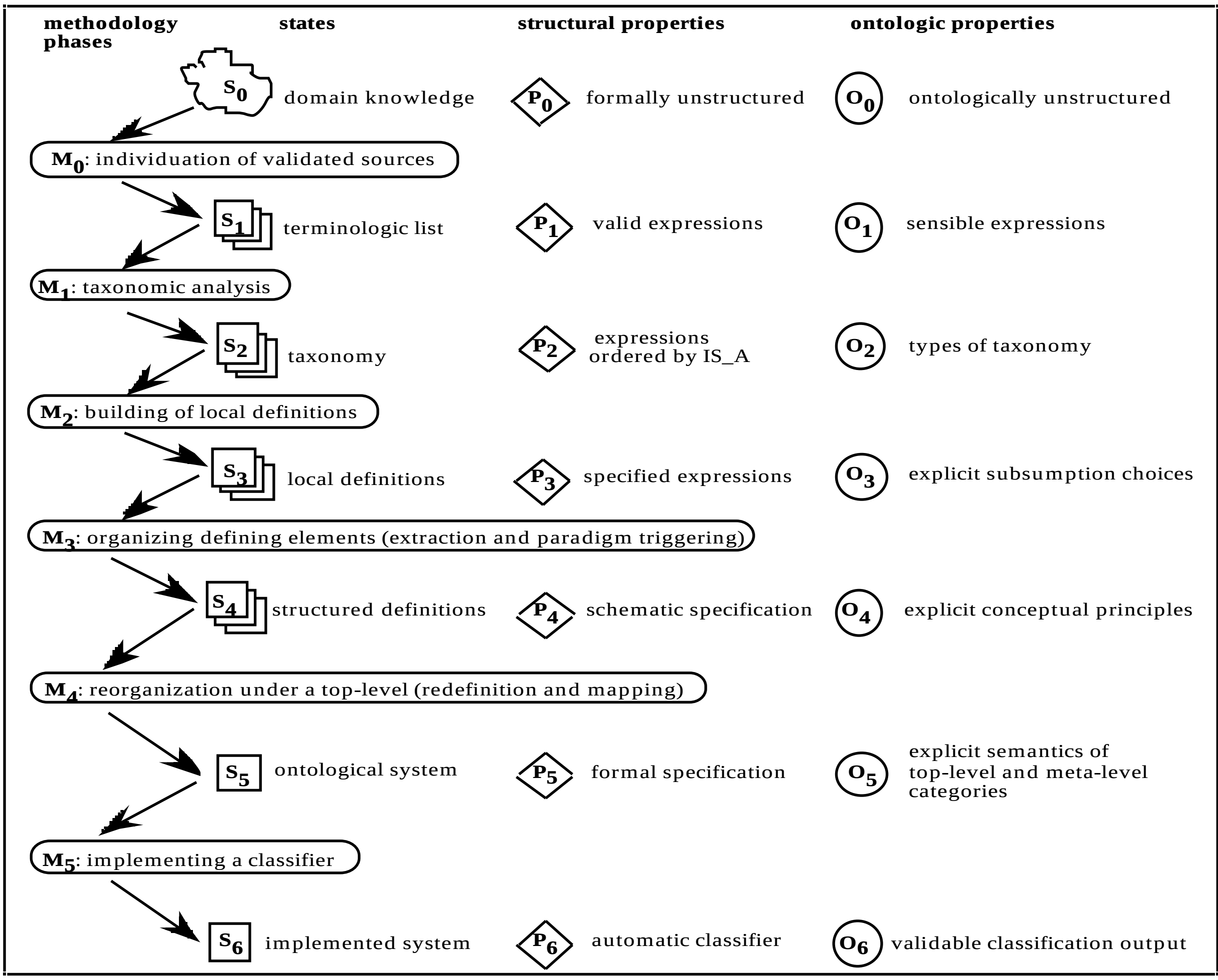

Figure 3. The methodological phases to build a terminology system which can face the new communication needs: knowledge integration, re-use, sharing. The output of any phase is a special state of one or more terminology systems, described by both structural and ontological properties. Such states are independently re-usable for a specific purpose.

S0* At state S_0 domain knowledge is formally unstructured (P_0) as well as ontologically opaque (O_0: no explicit conceptualization). Obviously, no terminology system is classified as such.

> E.g., the knowledge of a liver disease which seems to be caused by a virus rather than by a bacterium. This knowledge is usually conveyed by objective tests which for example say that the virus is a hepatitis virus type A; by observation or anamnesis, for example the observation of jaundice; and by shared domain knowledge, for example that the incubation period for hepatites caused by virus A is between 15 and 50 days.

M0>> The methodological phase M_0 consists in the individuation or building of validated terminological sources, which are in the state S_1.

S1* At state S_1 domain knowledge is represented by a list of valid expressions (P_1), which are meant to be conceptually plausible (O_1). Term lists compiled by experts and standard bodies, or extracted from free text, can be classified as such.

> E.g., the code 070.1 for *viral hepatitis type A* in the the code D-0521 for *viral hepatitis* in SNOMED-II, etc.

A term list item can be introduced as a primitive in a theory, with no defining expressions. At most, one can search for a code within a more structured terminology system, like SNOMED or ICD. For example, in Ontolingua we could create a new theory: "infectious-diseases" and introduce a term, e.g. "viral-hepatitis-A", taken from a list provided by experts, standard bodies, etc.:

```
(define-theory infectious-diseases ())                                    (1)
```

```
(define-class viral-hepatitis-a (?vh)                                          (2)
   :issues ((:see-also "in SNOMED, the code is D-0521" "in ICD9-CM the code is
    070.1")))
```

M1>> In the methodological phase M_1 one must find out the main taxonomic structure within terminological lists. This leads to an S_2 system.

S2* At state S_2 the lists should have an order induced by IS_A inclusions, namely, the lists are mono- or multi-hierarchical taxonomies (P_2). From an ontological perspective we could ask how many and which meta-classes further organize the taxonomies: in other words we assess which kind of taxonomy we have got (O_2). Most 'classic' terminology systems (ICD10, SNOMED-II, GMN, etc.) can be classified as such.

E.g., the terminology system holds that *viral hepatitis type A* has some parents such as *hepatitis*, or *condition*, or *disease*, or *viral hepatitis*, or *diagnosis,* or no parent concept at all.

In our theory we could start giving a taxonomic constraint to our term, so that the definition becomes:

```
(define-class viral-hepatitis-a (?vh)                                          (3)
  :axiom-def (subclass-of viral-hepatitis-a inflammation)
  :issues ((:see-also "in SNOMED, the code is D-0521" "in ICD9-CM the code is
    070.1")))
```

We must also introduce another define-class form for inflammation.

M2>> In the methodological phase M_2 one must individuate the reason why the IS_A links in an S_2 taxonomy exist and the reason why a term differs from its closest relatives in the taxonomy. For example, what distinguishes a "viral hepatitis" from the parent "hepatitis" is the fact that the viral hepatitis is necessarily due to a viral infection. This phase leads to S_3.

S3* At state S_3 the taxonomic list should be coupled with free text descriptions of terms (P_3); ontologically, these definitions should be the explicitation (O_3) of taxonomic constraints (the so-called *differentia specifica*). Most dictionaries and glossaries in medicine are classifiable as S_3.

E.g., the terminology system has an explicit description of *viral hepatitis-A*, for example, that it is an inflammation of the liver due to a hepatitis virus A, with an incubation of 15 to 50 days, and usually with jaundice (this partial definition has been taken from [Std95]).

In our theory we could include such a description, and our definition might become:

```
(define-class viral-hepatitis-a (?vh)                                          (4)
  "the inflammation of liver caused by virus A; it has an incubation of 15 to
   50 days and is accompanied by jaundice"
  :axiom-def (subclass-of viral-hepatitis-a inflammation)
  :issues ((:see-also "in SNOMED, the code is D-0521" "in ICD9-CM the code is
    070.1")))
```

It is a common concern in ontological engineering that an ontology should minimize its commitment level. In other words, one should be parsimonious when detailing the specification of a conceptualization.
ONIONS follows this guideline, but in agreement with the minimal intended meaning presupposed by the source where a term comes from. Current terminology systems provide hints for a very minimal conceptualization, and consequently we do not force them. In the particular case of the example given here, it originates from a medical dictionary, which is a particularly rich kind of terminology system. In practice dictionaries offer few hints to distinguish terminological knowledge from case-based knowledge. We included here a very small piece of the knowledge provided to viral hepatitis type A: that which is necessary to thoroughly distinguish it from other viral hepatites in the context of that dictionary entry (for a discussion of terminological vs. other kinds of knowledge, cf. [Ste96, Gan97]).

M3>> The methodological phase M_3 consists in the schematization of the elements in the S_3 description, in order to provide some "weak" constraints, and consists in the discovery of the conceptual principles which motivate the description. This leads to S_4 systems.

S4* At state S_4 the constraints should be "framed", which amounts to say that elements ("fillers") used in the description of the term have explicit relationships ("slots" or "roles") with the described term (the "frame") (P_4).

In logical terms, a frame gives constraints on the domain and the range of the relations applicable in a certain context of knowledge. At this state, the frame constraints may not have a formal semantics.
Few terminology systems have P_4 property: the UMLS Semantic Network is one (with an informal frame notation), the GRAIL models in the GALEN project are another example of S_4 (with a formal frame notation; the GRAIL implementation also features classifier services (P_6), see below).

E.g., the terminology system has a frame definition of *viral hepatitis type A*, for example (in the GRAIL syntax):

```
[Inflammation which                                                    (5)
  < affects-Liver
  isCausedBy-VirusA
  hasIncubation-15To50Days
  isAccompaniedBy-Jaundice >] name ViralHepatitisA
```

which says that the intersection among: the class Inflammation, the domain of the relation affects (with range:Liver), the domain of the relation isCausedBy (when range=VirusA), the domain of the relation hasIncubaton (when value=15To50Days), the domain of the relation isAccompaniedBy (when range=Jaundice), is the class ViralHepatitisA[3]. This example could also trigger a discussion on the difference between terminological and assertional level. We avoid to take into account such issues here (see below; for a discussion, see [Gan97]).
Definition (5) can be translated at first with the help of the generic-frame syntax, which is supported by Ontolingua and is compliant with most Object Oriented systems. Such a syntax constrains us to understand which are the slots referring to the class itself (":class-slots") and which slots refer to the instances of the class (":instance-slots"). In our theory "infectious-diseases" the "viral-hepatitis-A" definition would then become:

```
(define-class viral-hepatitis-a (?vh)                                    (6)
 "the inflammation of liver caused by virus A; it has an incubation of 15 to
  50 days and is accompanied by jaundice"
 :class-slots ((subclass-of viral-hepatitis-a inflammation))
 :instance-slots ((has-incubation viral-hepatitis-a 15-to-50-days)
                  (affects viral-hepatitis-a liver)
                  (caused-by viral-hepatitis-a virus-a)
                  (is-accompanied-by viral-hepatitis-a jaundice))
 :issues ((:see-also "in SNOMED, the code is D-0521" "in ICD9-CM the code is
          070.1")))
```

We must also introduce (or reuse) other define-class forms for *liver*, *virus A*, and *jaundice*, as well as define-relation forms for *affects*, *caused by*, *has incubation*, and *is accompanied by*, and a define-instance form for *15 to 50 days*. As (5)(6) show, the frame notation is at odds with complex instances like a time span (15 to 50 days). One could invent ad-hoc concepts or relations for bypassing this problem, but this usually results in ontological opacity and an inelegant naming policy.
On the other hand, even if we accept a tricky strategy on the formal part, in such a way we have fulfilled only the structural (P_4) property of S_4. Ontologically, a frame should be the explicitation of the conceptual principles (O_4). For instance, a very common conceptual principle concerns *part-whole* relationships, another the *teleology* of structures, still others the *quantities*, the *topology* of structures, the *physical* properties, and so on. In other words, we should make a call for some valuable and practicable generic ontology. In the example "viral-hepatitis-A", the differentia specifica "viral" calls for the formalization of some generic ontologies which specify axioms about organisms and causality.
At present, no terminology systems in medicine fulfil O_4. How to fulfil O_4 as well? For ONIONS-produced ontologies, we have developed a library of generic ontologies (see next state and §5.).
At this state, an operational suggestion is to make such calls by memorizing them in the :issues field of an Ontolingua definition:

```
(define-class viral-hepatitis-a (?vh)                                    (7)
 "the inflammation of liver caused by virus A; it has an incubation of 15 to
  50 days and is accompanied by jaundice"
 :class-slots ((subclass-of viral-hepatitis-a inflammation))
 :instance-slots ((has-incubation viral-hepatitis-a 15-to-50-days)
                  (affects viral-hepatitis-a liver)
                  (caused-by viral-hepatitis-a virus-a)
                  (is-accompanied-by viral-hepatitis-a jaundice))
 :issues ((:see-also "in SNOMED, the code is D-0521" "in ICD9-CM the code is 070.1")
```

[3] In other words, the "which" operator is a logical AND and the "name" operator is a logical equivalence. Indeed, GRAIL has a nice pseudo-natural-language syntax.

```
(:generic-theories "inflammation requires a multiple account within a theory
of biologic functions and a theory of biologic morphologies" "affects
requires a theory of actants and a theory of functions" "caused-by requires a
theory of causality" "the patient status is not mentioned" "anatomy is not
mentioned: at least, a part-whole theory is required")))
```

Obviously, making calls for generic theories is not a formal issue, but a guideline for an empirical investigation in the literature. Although we cannot formalize it, we can envisage the exploitation of some special on-line sites which maintain a huge library of generic ontologies for the common good, possibly with a rich documentation on the sources (informal theories, ontologies written in a non-standardized language, etc.). This is a relevant research issue.

M4>> The methodological phase M_4 consists in:

1) the construction (or the reuse, if available) of a library of generic ontologies to account for the `(:issues (:generic-theories))` requirements memorized at S_4 (for example those given in (7)); this equals to build a well-grounded *top-level* [Sow96];
2) the inclusion in the domain ontology of generic ontologies specifying relevant conceptual principles;
3) the assignment of sound meta-level categories to the classes and relations in the library.

This leads to S_5. In Fig. 3 only one system is in the state S_5 (only one square frame), in order to represent that it can integrate the previous ones. Of course, we can get different integrated systems according to the particular generic ontologies that one decides (for preference or pertinence) to include or use.

S5* At state S_5 definitions are axiomatized (P_5); ontologically, definitions have an *explicit semantics* (O_5) of both the *top-level concepts* (the concepts provided by generic ontologies) and of the *meta-level categories* (the concepts provided by a representation ontology). As explained in S_4, no classic terminology system has S_5 features.

E.g., an S_5 account of the previous definition for viral--hepatitis-A is:

```
(define-class viral-hepatitis-a (?vh)                                        (8)
  "the inflammation process of liver caused by virus A; it has an incubation
   of 15 to 50 days and is accompanied by jaundice"
  :def (and (inflammation-process ?vh)
            (exists ?vir
                    (and (has-a-cause ?vh ?vir) (virus-a ?vir)))
            (exists ?liv
                    (and (is-embodied-in ?vh ?liv)
                         (and (liver ?liv)
                              (exists ?pat
                                      (and (part ?liv ?pat) (*patient ?pat))))))
            (exists ?inc
                    (and (is-constitutive-phase-of ?inc ?vh)
                         (and (incubation ?inc )
                              (= (temporal-value ?inc) ?n)
                              (>= ?n 15) (=< ?n 50))))
            (forall (?jau ?pat)
                    (=> (and (jaundice ?jau) (*patient ?pat) (is-embodied-in ?jau ?pat))
                        (occurs-in ?jau ?vh))))
  :issues ((:see-also "in SNOMED-II the code is D-0521" "in ICD9-CM the code is
                    070.1")))
```

S_5 definitions are usually more detailed than the lower level ones. This is caused by the reference to generic ontologies, which constrain the modeller to explicitate what is usually "collapsed" in local definitions. A typical case is here the passage from the relation "has-incubation" in (7) to the complex quantified statement in (8). Local definitions do not need to "say it all". But when different local definitions must be integrated, some collapsed parts have to be expanded. In fact, definition (8) differs from (7) in several aspects, because it gives an answer to the calls specified in the :issues of (7) and deals with the forms non expressible in the frame syntax; in particular, we needed:

(a) concepts which are subsumed by other concepts already defined in a generic ontology;
(b) an ontologically sound representation of the complex instance in (7);
(c) the specification that only some of the instances occurring as the second argument of the relation is-accompanied-by when the first instance is a viral-hepatitis-A, are instances of jaundice.

(d) some quantified expressions to talk of a generic patient whose liver is infected and shows some symptoms after an incubation period: this cannot be represented in simple frame style. One should link patient with inflammation, liver, jaundice, incubation and virus-A;

(a) and (b) are solved by specializing the appropriate concepts from the required generic ontologies:

- "inflammation" can have at least three concomitant sense components, all normally presupposed by experts; in fact it refers to both a process (the "inflammation-process"), a morphology (the "inflammation" quality of a region of liver), and an object (the "inflammated-region" of liver). Since viral-hepatitis-A is usually focalized as a process (or the assessment of a process, namely a diagnosis), the first sense has been selected;
- "affects" and "caused-by" require an ontology of "actants", namely of the generic relationships betwen objects and processes. From ON9 theory: actants (http://saussure.irmkant.rm.cnr.it/HOME/ON9/actants/index.html), we used the relations: "is-embodied-in" for affects and "has-a-cause" for caused-by. The axiomatizations in theory: actants mainly rely on results obtained in cognitive science, linguistics and narratology investigations [Pri82, Fil71, Mil76, etc.];
- "has-incubation" requires an analysis of the notion of incubation; we have concluded that incubation is a kind of temporal context (a time-span) which encompasses the beginning, asymptomatic phases of infectious diseases; consequently we treated "incubation" as a subclass of time-span (see (11)), and specified the temporal value of incubation for viral-hepatitis-A. A specific range of a temporal value needs some operators to be specified: we have taken >= and =< from the theory: kif-numbers, built-in in Ontolingua; in LOOM the keyword :through (see (10)) is equivalent;
- "is-accompanied-by" also requires a time-oriented approach and consequently we used the "occurs-in" relation (see (12)) from theory: time;
- and in similar way for the other concepts.

(c) and (d) are solved by conjoining the three universally quantified implication expressions (within the :def keyword) corresponding to the three slot-value-type expressions (within the :axiom-def keyword), with two more expressions, existentially quantified, which account for the incubation period and the jaundice sign. Classifiers usually do the conjoining job by allowing an encapsulation of "AND" constructs and an 'aliasing' mechanism. For example, in GRAIL we would represent a piece of (8) as:

```
[InflammationProcess which                                    (9)
      < affects-[Liver which <isPartOf-Patient>]
        isCausedBy-Virus>] name ViralHepatitis
```

GRAIL lacks a precise correspondence to First-Order Predicate Calculus (FOPL) expressions in (8). For example, distinguishing among FOPL quantifiers is not easy when modelling in GRAIL, which on the contrary provides some "qualifiers" for "sanctioning" the intensional validity of an expression. For example, the qualifier "grammatically" is suggested for use with very general concept definitions (not assured to be satisfiable for all interpretations), while the "sensibly" qualifier is suggested for use with domain specific concept definitions (presumably satisfiable for most domain interpretations). In GRAIL these qualifiers are 'hierarchical', say a "sensibly" concept definition needs a previous "grammatically" definition of a parent concept. This is a nice property for domain knowledge evaluation, and it deserves to be studied in order to provide a precise semantics to it, or at least to design sound guidelines for using it.
In fact, we remarked that a two-level hierarchy is sometimes tricky to manage. Moreover, these are intensional problems, partly related to the qualitative assessment of an ontology by the experts rather than to the issues listed in §2. In this paper, we do only describe ways to capture and to integrate terminological knowledge which is already assessed; moreover, we suggest tools and guidelines for negotiating ontologies collaboratively. Introducing formal a-priori subdivisions on the assessment process is another, although interesting, story.
In Loom, we can extend the classifier translation to the entire (8) by using just the Tbox language:

```
(defconcept viral-hepatitis-a :context infectious-diseases :is-primitive          (10)
 (:and inflammation-process
  (:some has-a-cause virus-a)
  (:some is-embodied-in (:and liver (:some part *patient)))
  (:some has-constitutive-phase (:and incubation
   (:the temporal-value (:and day (:through 15 50)))))
  (:all has-occurrence-of (:and jaundice (:some is-embodied-in *patient))))
 :annotations
 ((documentation
   "the inflammation process of liver caused by virus A; it has an incubation
   of 15 to 50 days and is accompanied by jaundice")))
```

In the Loom language, :is-primitive means that viral-hepatitis-A is a primitive concept (it has not a complete definition), slots and types are introduced by means of :and, :all and :the keywords. Full classification for predicate calculus is not available in Loom, but it is being implemented in PowerLoom [Mac94].

Thus, Loom provides efficient syntax and semantics for managing a consistent subset of FOPL with its classifier.
Moreover, we need to define "incubation":

```
(define-class incubation (?i)                                                  (11)
  "the kind of temporal context for the initial phases of infectious diseases,
   producing no evident medical signs or symptoms"
  :def (and (time-span ?i)
            (exists ?id
               (and (infectious-disease ?id)
                      (exists ?ph
                              (and (has-constitutive-phase-of ?id ?ph)
                           (is-started-by ?id ?ph)
                           (is-context-of ?i ?ph)))
                         (exists ?pat
                             (and (*patient ?pat)
                                      (is-embodied-in ?id ?pat)
                                      (not (exists ?ms
                                               (and (or (evident-medical-sign ?ms)
                                                        (symptom ?ms))
                                                        (forall ?ph
                                                            (=> (embodies ?pat ?ms)
                                                                (occurs-in
                                                                      ?ms ?ph)))))))))))
```

On its turn, this definition cannot be managed by not very expressive languages because of its negated existential expression including a disjunction (for a discussion of such problems, [Spe95]).
Further complication is carried out for temporal constraints of processes (like diseases are), which typically induce a "non-monotonic" situation-change; for example, in the medical domain a situation change can trigger a "condition" which was not previously present/absent/the same. In fact, the relation "occurs-in" used in (11) requires a meta-statement for situation-change (or a temporal description logic, cf. [Art95]):

```
   (define-relation occurs-in (?te1 ?te2)                                      (12)
     "the relation of occurrence between temporal entities. For any temporal
      entities p and q such that occurs-in(p, q) holds, there can be a part of
      p in which q does not occur. Temporal entities include processes, contexts
      like situations and time spans, and signs representing an underlying process"
     :def (and (or (process ?te1)
                   (*sign ?te1)
                   (context ?te1))
               (or (process ?te2)
                   (*sign ?te2)
                   (context ?te2))
               (exists (?s1 ?s2)
                   (and (situation ?s1) (situation ?s2)
                        (is-context-of ?s1 ?te1)
                        (is-context-of ?s1 ?te2)
                        (is-context-of ?s2 ?te2)
                        (ist ?s1 "(occurs-in ?te1 ?te2)")
                        (ist ?s2 "(not (occurs-in ?te1 ?te2)"))))
     :issues ((:see-also "the definition of 'during' in theory: time")))
```

Anyway, any tricks can be found for representing most concepts in most languages; the problem is maintaining clarity, elegance, and easy negotiation and cooperation in the modelling activity.

Metaontology. As far as metaontology is concerned, predicates in previous definitions (given their specification in the ON9 library, see §5.), distribute defining elements among different meta-level categories, which we have defined in a representation ontology called "metaontology" (see the model in [Ste96, Gan97] or our WWW site at: http://saussure.irmkant.rm.cnr.it/HOME/ON9/metaontology/index.html). For example:

• the concepts "part", "embodied-in", "is-a-cause-of", etc., are assigned to the category !relation, which has a semantics similar to the one given for "binary-relation" in the frame-ontology, but has been constrained to range over the instances of the "structural-concepts" of an ontology (those categorized by !type, !category, or !reified-property categories);
• the concepts "liver", "inflammation", "virus-A", etc., are assigned to the category !type, which is for the "rigid" classes of an ontology, namely an instance i of a !type class cannot occur in a domain of situations in which a situation encompasses i and another does not. Moreover such instances are countable (they can be structurally dishomogeneous);

• the concept *patient is assigned to the category !role, whose classes are "non-rigid" and typically "reify" a unary relation or a domain or range of a binary relation. !role classes also grasp the common sense notion of "role", which is constrained by some actantial notions (see above in this paragraph).

In our opinion, having a rich metaontology is not an overcommitment as far as the ontology languages do not try to define special formal properties at the object-level. For example, there can be some nominalistic conflict between our use of meta-level-categories at real meta-level and the formal use of similar names for particular entities in the Loom language ("property", "concept", "relation"). Since the use of MLC at Loom object-level is unavoidable, some confusion may arise.
The problem is easily solved if all concepts have a similar object-level environment and then receive a special meta-level assignment. In support of this view, the forthcoming PowerLoom, based on the PC-Classifier [Mac94], treats all concepts as n-ary set-theoretic relations, as well as KIF does [Nec91].

At this point, we should also revise the definition of the theory: infectious-diseases for including the relevant theories (the following included theories are a subset of the inclusion lattice of ON9, presented in §5.):

```
(define-theory infectious-diseases (diseases micro-organisms))                    (13)
```

The relation of inclusion between theories is transitive. The semantics (and pragmatics) of the inclusion is different according to the language used. For example, in Ontolingua the modularity of the library does not prevent one: a) to *use* in an ontology O concepts from an O' which is not *included* in O; b) to make different or even conflictual definitions of the same concept, except in the same ontology.

Another language which in our experience has resulted both worthly and friendly is OCML [Mot95] (Operational Conceptual Modelling Language), which operationalizes (namely, assumes an operational semantics for) the most relevant constructs of Ontolingua, plus other special features (see tables 2 and 3 for a summary of its features).

	Ontolingua 4.0	OCML 1.0
Structuring principles	Ontologies ("theories"), functions, relations, classes, instances.	Ontologies, tasks (and inferences), roles, functions, relations, rules, classes, instances.
Semantics	Set-theoretic; the same as KIF [Nec91]. There is a KIF parser, an ontology analyzer, but no inheritance system. Strict subsumption hierarchies among atomic concepts are shown and ontology reports are generated.	Operational; there are three interpreters: prolog-based theorem prover, function interpreter, and control interpreter. The function interpreter and the theorem prover are integrated - i.e. functions can be called from within proofs and proofs can be invoked from functions. Inheritance system is also integrated with the prolog-based theorem prover.
Formality	The semantics of Ontolingua constructs is given in terms of the corresponding KIF constructs, plus the frame-ontology, which is given a semantics by specializing KIF constructs.	There is no formal semantics for control language. The semantics of functions, relations, rules, classes, and instances is given in terms of the corresponding Ontolingua constructs.
Reusability	There is a library of reusable components: representation ontologies (currently a set of KIF ontologies (for relations, sets, numbers, lists, etc.); the frame-ontology and slot-constraint-sugar ontology which support second-order style of modelling of frame and O/O); domain ontologies.	There is a library of reusable components. These are divided into five categories: basic, tasks, method, domain, application. Basic is the base ontology (lists, numbers, strings, etc.). Domain are domain models. Methods are problem-solving methods. Tasks are generic tasks (e.g. parametric design). Applications combine task / methods / domain mapping and application-specific knowledge.
Support	Libraries, translations to knowledge representation systems, htmlification of models and documentation.	Libraries.
	LOOM 3.0	GRAIL
Structuring principles	Ontologies ("contexts"), roles, functions, relations, rules, classes, instances.	Relations, classes.

Semantics	Set-theoretic. Separate semantics for terminological (Tbox) and assertional (Abox) components. There are various services: deductive reasoning through classification-based and production-based inferences; procedural programming through pattern-directed methods.	Partly set-theoretic, partly based on intensional considerations about the dependency between generality and expertise level of knowledge. A description classifier is implemented which computes subsumption relations between descriptions. The non-set-theoretic part is implemented as constraints on the possibility of making descriptions.
Formality	The formal semantics can be done in the KRSS standard for description logics [Pat93].	The set-theoretic part has a Tbox description logic formal semantics (e.g., as KRSS standard may provide). The non-set-theoretic part has no formal semantics (some "qualitative" mapping is suggested in Tab. 3).
Reusability	LOOM contexts are reusable components.	No reusable components, at least according to present standards in library specification and maintenance. In fact, a translator could help the reuse of GRAIL models.
Support	Libraries.	Independent models.

Table 2: Some features of the (implemented) languages used in our experience with ONIONS.

Beyond features in tab. 2, OCML has a more compact syntax than Ontolingua. More extensions are foreseen, for example a translator to Loom.

M5>> The methodological phase M5 consists in the implementation of a domain ontology in a system which allows automatic classification. Obviously, the generated classification should pass a validation control. This leads to S6.

S6* At state S_6 a domain ontology is implemented in an automatic classifier (P_6. Some terminology systems are currently implemented with an automatic classifier, for example the GALEN Core Model. Currently, our method is to export a library of ontologies from the Ontolingua form into Loom and to make Loom classify the library. For example, see (10).
For reference, we give a semantical comparison among a bunch of the operators of the languages used in this paper (Table 2).

<table>
<tr><th>ONTOLINGUA</th><th>OCML</th><th>LOOM</th><th>GRAIL</th><th>SET-THEORETIC SEMANTICS</th></tr>
<tr><td>(and (B ?a) (C ?a))</td><td>(and (B ?a) (C ?a))</td><td>:and B C</td><td>addSuper (B C)</td><td>$B^I \cap C^I$</td></tr>
<tr><td>(or (B ?a) (C ?a))</td><td>(or (B ?a) (C ?a))</td><td>:or B C</td><td></td><td>B^I C^I</td></tr>
<tr><td>(not (B ?a))</td><td>(not (B ?a))</td><td>:not B</td><td></td><td>$\Delta^I \setminus C^I$</td></tr>
<tr><td>(slot-value-type [A] R B)</td><td>((R :type B))</td><td>:all R B</td><td>sensibly R B</td><td>{i Δ^I| j.(i,j) R^I j B^I}</td></tr>
<tr><td>(value-type ?a R B)</td><td>((R :type B))</td><td>:all R B</td><td>sensibly R B</td><td>{i Δ^I| j.(i,j) R^I j B^I}</td></tr>
<tr><td>(has-slot-value-of-type [A] R B)</td><td>((R :type B
:min-cardinality 1))</td><td>:some R B</td><td>necessarily R B</td><td>{i Δ^I| j.(i,j) R^I j B^I}</td></tr>
<tr><td>(has-value-of-type ?a R B)</td><td>(exists ?b
(R ?a ?b))</td><td>:some R B</td><td>necessarily R B</td><td>{i Δ^I| j.(i,j) R^I j B^I}</td></tr>
<tr><td>(has-single-slot-value-of-type [A] R B)</td><td>((R :type B
:max-cardinality 1))</td><td>:the R B</td><td></td><td>{i Δ^I| !j.(i,j) R^I j B^I}</td></tr>
<tr><td>(has-one-of-type ?a R B)</td><td>((R :type B
:max-cardinality 1))</td><td>:the R B</td><td></td><td>{i Δ^I| !j.(i,j) R^I j B^I}</td></tr>
<tr><td>(define-class A (?a)
:def (forall ?c
(=> (R ?a ?c)
(C ?c))))</td><td>(def-class A (?a)
:constraint (forall ?c
(=> (R ?a ?c)
(C ?c))))</td><td>(defconcept A
:is-primitive
(:all R C)</td><td>A sensibly R B</td><td>AN^I {i Δ^I| j.(i,j) R^I j B^I}</td></tr>
<tr><td>(define-class A (?a)
:iff-def (and (B ?a)
(forall ?c
(=> (R ?a ?c)
(C ?c)))))</td><td>(def-class A (?a)
:iff-def (and (B ?a)
(forall ?c
(=> (R ?a ?c)
(C ?c)))))</td><td>(defconcept A
:is
(:and B
(:all R
C)))</td><td>B which R C name A</td><td>$AN^I = B^I \cap$ {i Δ^I| j.(i,j) R^I j C^I}</td></tr>
<tr><td>(define-class A (?a)
:def (exists ?c
(and (C ?c)
(R ?a ?c))))</td><td>(def-class A (?a)
:constraint (exists ?c
(and (C ?c)
(R ?a ?c))))</td><td>(defconcept A
:is-primitive
(:some R C)</td><td>A necessarily R B</td><td>AN^I {i Δ^I| j.(i,j) R^I j B^I}</td></tr>
</table>

Table 2: Some examples of concept (first nine rows) and statement (last three rows) construction in some languages experimented with ONIONS. On the right column an equivalent set-theoretic semantics is shown.

A remark on GRAIL: <grammatically R B> is a special case of <sensibly R B> used for "maximal" definitions, which are supposedly given for very general concepts. The subsumed concepts of these are then to be specified by means of sensibly (eg, <grammatically R B> and <sensibly R B1>).
The $\cdot^{I}$ in the set-theoretic semantics column means the (extensional) domain of interpretation.

4. AN OVERVIEW OF THE TOOLS

In §2. at issues (7)(8) we had proposed some requirements; in particular, we claimed that modeling terminological ontologies needs a toolbox for distributed collaboration.
In §3. we have shown the complexity of term conceptualization activity: several decision have to be taken on terminology system analysis, formal choices, theories to include, literature to scan, translations to perform, etc. Those decisions can be validated only by collaborative effort of interdisciplinary experts.
For this reason, we formulated four required functions:

(a) ontology libraries available on the Internet;
(b) on-line remote accessing of libraries for editing, saving, and exporting ontologies and concept definitions;
(c) interfaces to libraries which are customized to the expertise level of the user;
(d) an interactive tool for collaborative discussion about the libraries: where different modellers could experiment and face each other about the effects of ontological choices on terminology integration, as well as about the constraints posed by terminology integration on ontological choices.

Several tools have these functions. During the development of ON9 and its former versions (ON6-8 [Ste96]), we experimented with some of them.

Function **(a)** is supported by Ontolingua 4.0 (for main features, see Tab.2), a Common Lisp implementation of Ontolingua released in 1994, but now no more in distribution. We appreciate its high expressive power, which allows both first and second order logic expressions, as well as frame-like expressions. We still use it to write the primary sources of our ontologies. Ontolingua 4.0 can translate ontologies in Loom, KIF, Generic Frame Protocol (GFP), and other languages, and creates nice html directories containing hypertextual versions of: the source files of ontologies, the ontology reports and the individual concept definitions in GFP.
The main drawback of Ontolingua is the lack of inferential capabilities. In fact, due to the high expressive power, one may lose the control of the consistency among concepts and among ontologies: sometimes we have experienced this when translating from Ontolingua to Loom. On the other hand, even such a drawback can be an advantage if one is not interested, at least in a first phase, in spending a lot of time in revising theory inclusions and definition allocations.

Function **(b)** (together with (a)) is currently supported by various tools: the Stanford Ontolingua Server [http://www.ksl.stanford.edu; some European mirrors should be active soon. Also Far96], Ontosaurus [Swa96], KSSn [Gai94], etc.
A centralized Server is the current policy of Ontolingua developers. The Server allows the on-line remote accessing of libraries for editing, saving, and exporting ontologies, all with a nice interface, but it provides less predicate calculus construct types than Ontolingua 4.0; on the other hand, the developers have enhanced the frame-like constructs.
Ontosaurus (or "Loom-HTTP") is an ontology server implemented using CL-HTTP [Mal94], a Common Lisp Web server, the Loom knowledge representation system [Mac91] (for main features, see Tab.2), and Lisp code that interfaces Loom to CL-HTTP. Ontosaurus incorporates Loom, thus takes advantage of Loom's reasoning capabilities, specially for concept classification. On the other hand, having an operational KR system constrains to maintain coherence and thus makes multiple simulataneous edits to the knowledge base (a part of our function (d)) difficult, as explicitly recognized by developers. Ontosaurus includes translators to Ontolingua, KIF, KRSS [Pat93] and C++ (with obvious limitations in translatability), among others.
We are currently using Ontosaurus to perform function (b): it offers many semantic services for conceptualization activity; also, it is quite portable and thus sharable with collaborating centers. Since our primary sources are written in Ontolingua, we translated them in Loom. The original translator from Ontolingua 4.0 is helpful, but substantial hand revision must be performed for some constructs. Examples of Ontosaurus are in Figs. 4 through 6: the definition of viral-hepatitis-a (10) is shown as in Fig. 4: the upper frame contains the main commands for browsing, loading, editing and saving Loom contexts; the left frame is a "reference" frame where one can put some useful file; the right frame is the actual working frame. In Fig. 5 a different view of the same concept is shown which provides taxonomical information on the left and the applicable relations on the right. Fig. 6 shows a piece of the editing environment.

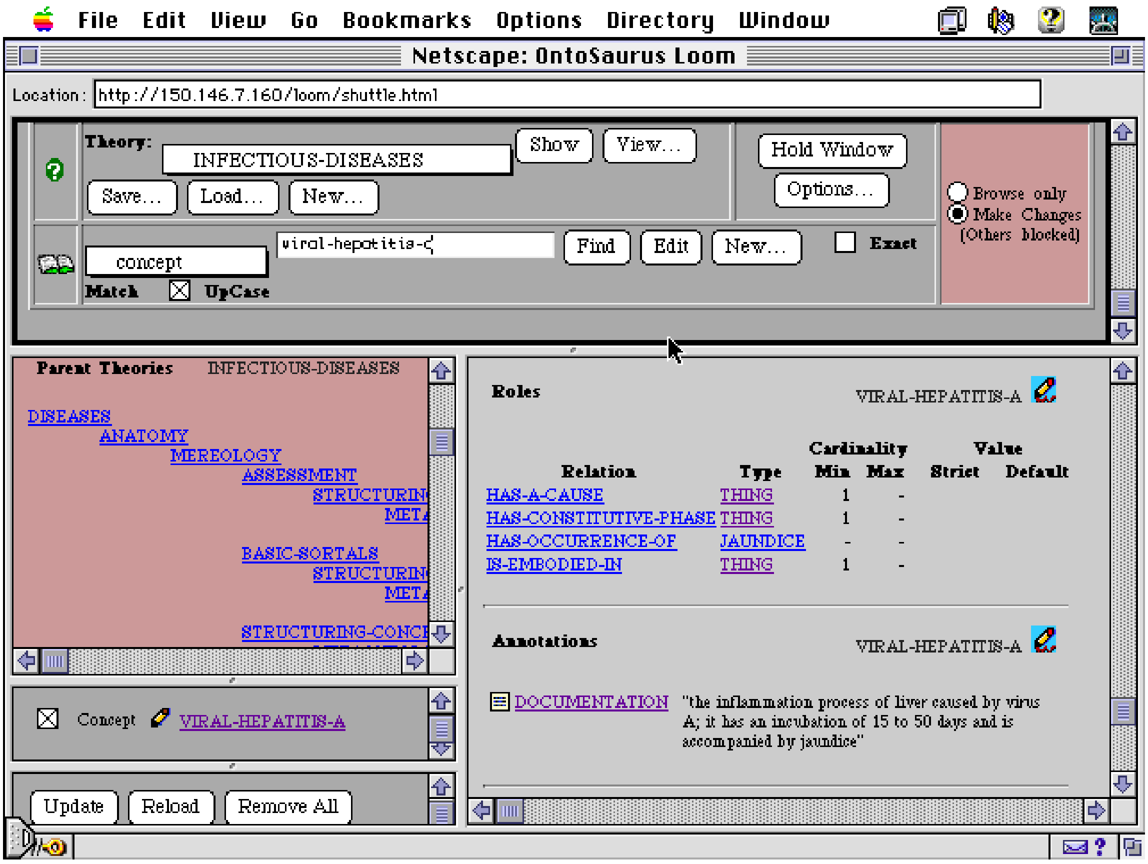

Figure 4: The Loom definition and the related context library for "viral-hepatitis-A" through Ontosaurus

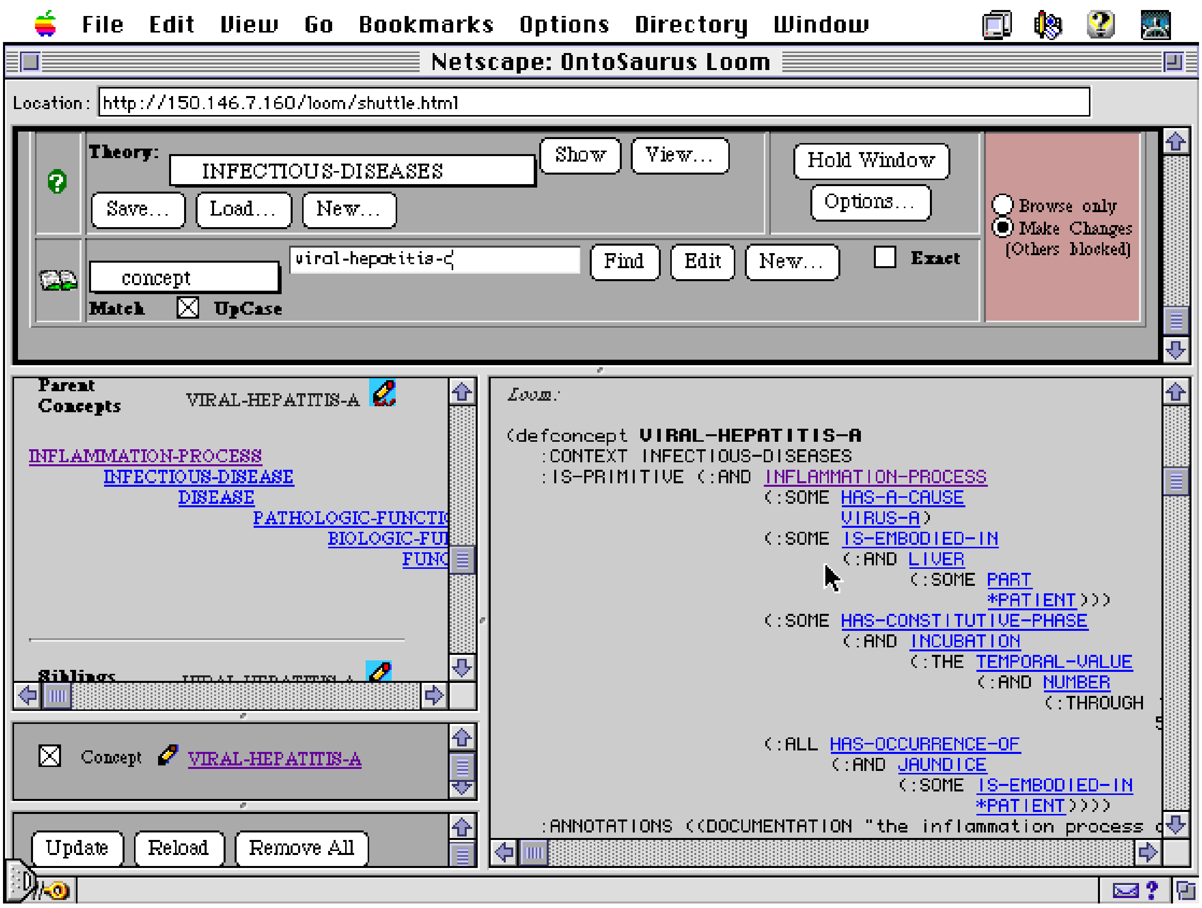

Figure 5: The Loom taxonomy and the related roles for "viral-hepatitis-A" through Ontosaurus

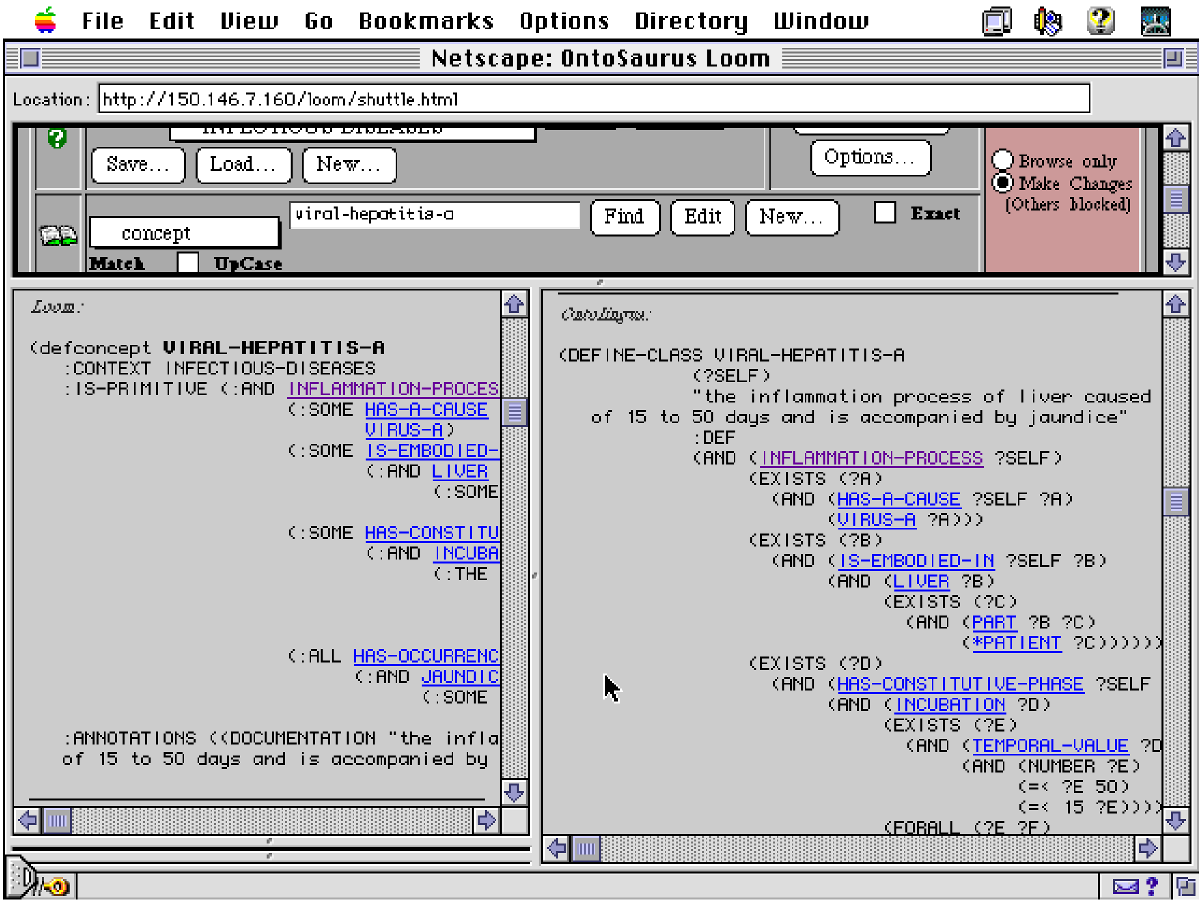

Figure 6: The Loom editing environment for "viral-hepatitis-A" through Ontosaurus

Function **(c)** is partly handled by the existing tools: all their interfaces help accessing, retrieving, editing ontologies. GFP is also a very intuitive format for frame definitions. On the other hand, physicians are not interested in understanding the logical nuances of the languages presented; in the current project GALEN-IN-USE, a special intermediate tool has been created to let medical experts make their models: even a friendly syntax as the GRAIL's resulted slightly awkward.
An intermediate tool seems to require the following notational features: no special logical operators nor quantifiers; lookup-table format or concept map format [cf. Gai94]; minimal or no constraints on meta-level categories.
For example, the UMLS Semantic Network browser is good at performing function (c). At first, one can wonder how such a tool could be interfaced to an AI application with a clear semantics. In our opinion, given ONIONS guidelines, a semantic-less or semantic-poor interface is better than nothing: ontological engineers would have to fill the semantic gaps of an intermediate tool by following those guidelines. We are also investigating how much semantic transparency can be contracted to reach a compromise.

Function **(d)** (as well as (a) and (b)) is currently handled by at least one tool: the WW Lab Server [Zdr97], which uses OCML [Mot95] as its knowledge tool component. It allows real-time interaction for collaborative modeling discussions by integrating texts, images, ontologies, and hand drawn sketches in a page (a "Tadzebao") made up of several "notepads": each notepad is in its turn composed of various pages, and each page can contain a text, an image, or an ontology; this matter can be hand drawn with sketches. Everything then appears on each client side page of the experts which participate in the discussion. Ontologies are properly developed through an ontology editor/browser. WW Lab Server has three layers: an Internet infrastructure including a customized Web server based on the LispWeb Common Lisp HTTP server [Riv96], a knowledge tool (OCML) for ontology construction, and a domain layer where case libraries and problem solving methods libraries are stored.
WW Lab Server seems promising because t is compliant with most of our requirements. Moreover, OCML has the same expressive power of Ontolingua 4.0, thus we could concentrate all functions in one toolbox, then a (forthcoming) OCML to Loom translator would map ontologies to the semantic services of a classifier.

5. THE ON9 ONTOLOGY LIBRARY

ON9 (available at http://saussure.irmkant.rm.cnr.it/HOME/ON9/index.html) is a library of ontologies designed by means of the ONIONS methodology.

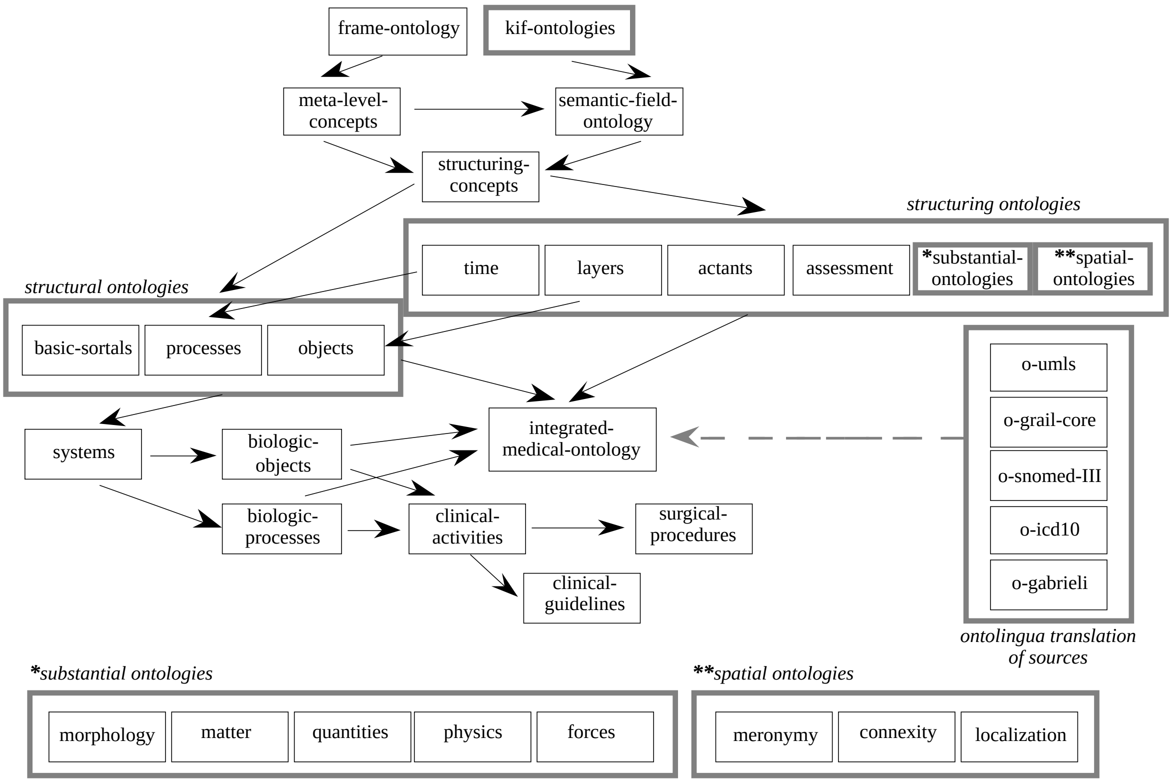

Figure 7. A significant subset of the inclusion lattice of the ON9 library of ontologies. Ontologies are represented by black circles. Thick grey frames or circles are sets of ontologies (some explictly show the elements). The semantics of black arrows is *included-in* (applied differently by Ontolingua or Loom, see text). The dashed grey arrow means *integrated-in* .

Figure 7 shows an inclusion lattice of some ON9 ontologies: the representation ontologies provided by default in Ontolingua are "frame-ontology" and the set of kif-ontologies. We defined the ontologies: "structuring-concepts", "meta-level-concepts" and "semantic-field-ontology", to link the representation ontologies with the generic ontology library. The sets of "structural ontologies" and of "structuring ontologies" contain generic ontologies. Generic ontologies are variously included in domain ontologies. In particular, integrated-medical-ontology includes all the generic ontologies, which have been used to integrate the terminological ontologies of the five terminology systems.
The current ON9 ontology library consists of five identifiable sets of models:

1) the intermediate byproducts of the ONIONS integration of the top-levels of the five terminology systems: conceptual primitives (from phase M1), taxonomical inclusions (from phase M2), and formal Local Definitions (LD) (from phase M3). For example, the LDs in USN include taxonomic constraints and some constraints ("templates") on domain and range of relations, stated within class definitions.
The following is the formalization of the LD of "organism" in the theory "o-umls" (the Ontolingua translation of USN):

```
(define-class Organism (?x)                                                    (14)
  "Generally, a living individual, including all plants and animals"
  :class-slots ((subclass-of Physical-Object))
  :instance-slots ((affected-by Organism Acquired-Abnormality)
                     (affected-by Organism Biologic-Function)
                     (affected-by Organism Congenital-Abnormality)
                     (has-part Organism Anatomical-Structure)
                     (has-process Organism Biologic-Function)
                     (has-property Organism Organism-Attribute)
```

```
                       (interacts-with Organism Organism)
                       (interacts-with Organism Organism))
  :issues ((:generic-theories "has-part requires a part-whole ontology" "has-process
             and is-affected-by require an ontology of actants")))
```

Other terminology systems are poorer, for example, the SNOMED-III similar concept is "living organisms", which is given as a primitive.

2) a library of Generic Ontologies (GO) to be used in the integration process (Fig. 7). This work has been carried out with a minimalistic strategy: only some parts of some theories which are useful for the integration process are "bought". For example, given the need of buying some theory of parts and wholes, we chose a subset from the so-called Calculus of Individuals from the philosophical literature [Leo40] and some specific notions of part from the cognitive science literature [Ger96], formalizing a theory: "meronymy". The following is an Ontolingua definition of "overlaps" from the Calculus of Individuals; it uses some second-order predicates for properties of relations and some first-order axioms of equivalence (here stated under the keyword :iff-def):

```
(in-theory 'meronymy)                                                                 (15)
(define-relation common-part-with (?x ?y)
 "this is the minimal definition for 'overlapping' in classical extensional
  mereology, and should be compliant with both Leonard-Goodman calculus of
  individuals and Tarski's axioms"
 :axiom-def (and (reflexive-relation common-part-with)
                 (symmetric-relation common-part-with)
                 (alias common-part-with overlaps))
 :iff-def (exists ?z
             (and  (part ?z ?x)
                  (part ?z ?y))))
```

3) the Integrated Medical Ontology (IMO), including some ontologies from GO and some Domain Ontologies (DO). For example, a corresponding definition to (14) is specified in the theory: "biologic objects" as follows:

```
(in-theory 'biologic-objects)                                                         (16)
(define-class organism (?org)
  "the type concept for living objects in the biologic layer (cf. M Blois
   [$M-I&M] and N Hartmann [$]P-GF)"
  :axiom-def (and (!type organism)
                  (subclass-of organism biologic-object)
                  (value-type organism has-component abnormal-body-part)
                  (value-type organism embodies abnormal-function)
                  (value-type organism embodies pathologic-function))
  :def (and (exists ?phy
                    (and (embodies ?org ?phy)
                         (physiologic-function ?phy)))
            (exists ?bp
                    (and (part ?bp ?org)
                         (body-part ?bp))) ))
```

Formula (16) makes use of a dedicated second-order predicate (we defined it in theory: meta-level-ontology), which assigns a meta-level category, in this specification expressed by !type; of some second-order axioms in the way of (16), and of some first-order axioms — stated under the keyword :constraints — which specify more complex constraints (see also §3.).

4) the mappings between each LD and the IMO. For example, having both (14) and (16), (16) is modified by adding a constraint as follows:

```
(in-theory 'o-umls)                                                                   (18)
(define-class organism (?x)
     etc. etc. {see (3)}
  :constraints (integrated-in organism organism biologic-objects))
```

which states that "organism" in the theory: o-umls is integrated in "organism" in the theory: biologic-objects, which is a module in the ON9 library. "integrated-in" is a ternary relation. Obviously, all the concepts and relationships appearing in the o-umls definition have an integration mapping in some ON9 module.

5) some specialized domain ontologies: surgical procedures, clinical activities, infectious diseases, clinical guidelines, etc., using a subset of modules from IMO (Fig. 7).

6. CONCLUSIONS

From the ONIONS experience of developing terminological ontologies in the last years, we can claim that:

a) from the viewpoint of conceptual integration of terminologies, the ontologies produced through ONIONS may support:

• formal upgrading of terminology systems: term classification and definitions are now available in a common, expressive formal language;
• conceptual explicitness of terminology systems: (local) term definitions are now available, even though the source does not include them explicitly;
• conceptual upgrading of terminology systems: term classification and definitions are translated such that they can be included in an ontology library which has a subset constituted of motivated generic ontologies.

b) from the viewpoint of reuse and maintenance, the ontologies produced through ONIONS may support:

• a motivated generic ontology library, developed from the integration of authoritative generic and domain sources;
• specialized domain ontologies which use some subset of ontologies from the ontology library;
• a refinement of the ontology library through the integration of other generic and domain sources: an integrated medical ontology.

We proposed an overview of ontology languages and we exposed why we consider rich expressivity as a prerequisite. Our experience suggests that representing a terminological ontology requires complex formal specifications involving full first-order sentences, some second-order sentences about situation and contextual change, pervasive existential quantification, definition of meta-level categories of the representation language, etc. We have found that Ontolingua, OCML, and Loom are well-suited to this purposes.

c) from the viewpoint of cooperative ontology modeling and validation on the WWW, use and integration of ON9 should be negotiated or customized by:

• a user who accepts a set of ontological definitions (available within a formal theory);
• a user who assesses as inadequates such sets of ontological definitions and cooperates in order to extend the set of ontological definitions or in order to integrate it with an other source;
• a user who rejects a set of ontological definitions and cooperates in order to define other definitions which are sounder to him.

We also proposed an overview of toolboxes for ontology construction and we exposed why we consider collaborative modeling capabilities an even stronger prerequisite. We currently use Ontosaurus to fit our needs, and we plan to use WW Lab Server to test a real-time interactive modeling collaboration.
Although ON9 is still being tested by experts, there is no doubt that acceptance, rejection and extension are fundamental phases in the process of ontology validation, extension and update.
The necessity of extensive off-line human intervention in the search, choice, and formalization of generic ontologies can be seen as unavoidable bottlenecks in ONIONS ontology modelling. An appealing alternative is to adopt a systemic approach in the generic library, which is widely shared and formally available. As a matter of fact, our analysis evidentiates that system theory, widely used in engineering domains (the usual configuration of component-state-event-process), does not fit the medical domain. The basic principles motivating the conceptualization of terminology in medical domains refer also to other theories, such as those provided (mostly in informal ways), by linguistics, philosophy, and cognitive science.

Acknowledgments

This paper has benefited from the useful comments and suggestions of Enrico Motta and Fabrizio Giacomelli. Our research is partly supported by the Italian National Research Council Special Project ONTOINT (ONTOlogical tools for Information iNTegration).

REFERENCES

[Art95] Artale A, Franconi E. Hierarchical Plans in a Description Logic of Time and Action. in Borgida A, M Lenzerini, et al. (eds): *Proc. of International Workshop on Description Logics*, Rome (1995).
[Brg96] Borgo S, Guarino N, Masolo C. Stratified Ontologies: The Case of Physical Objects. in Vet (ed.) *Proceedings of ECAI96 Workshop: Ontological Engineering* (1996).
[Bor96] Borst P, Akkermans H, Top J. Engineering Ontologies. *International Journal of Human-Computer Studies*, 46 (1997).

[Bra91] Brachman R, McGuinness DL, Patel-Schneider PF, et al. Living with Classic. in JF Sowa (ed.): Principles of Semantic Networks, San Mateo, CA, M Kaufmann (1991).

[CEN95] CEN prENV 12264:1995. Medical Informatics - Categorial structure of systems of concepts - Model for representation of semantics (Document For Formal Vote). Brussels: CEN (1995).

[Coh96] Cohn AG, Randell DA, Cui Z. Taxonomies of Logically Defined Qualitative Spatial Relations. *International Journal of Human-Computer Studies*, 43 (1996).

[Cot94] Coté RA, Rothwell DJ, Brochu L (eds). SNOMED International, 3rd ed., 4 vols. Northfield, Ill: College of American Pathologists (1994).

[Doy91] Doyle J, Patil R. Two Theses of Knowledge Representation: Language Restrictions, Taxonomic Classifications, and the Utility of Representation Services. *Artificial Intelligence*, 49 (1991).

[Eva94] Evans DA, Cimino JJ, Huff SM, Bell DS for the CANON Group. Toward a Medical-Concept Representation Language. *Journal of the American Medical Informatics Association* 1:207-17 (1994).

[Fal94] Falasconi S, Stefanelli M. A Library of Medical Ontologies. in Workshop on Comparison of Implemented Ontologies, ECAI 94 (1994).

[Fal96] Falasconi S, Lanzola G, Stefanelli M. Using Ontologies in Multi-Agent Systems. in B Gaines, M Musen (eds), *Proceedings of KAW96* (1996).

[Far96] Farquhar A, Fikes R, Rice J. The Ontolingua Server: a Tool for Collaborative Ontology Construction. in *Proceedings of KAW96 (Knowledge Acquisition Workshop)* (1996).

[Fil71] Fillmore CJ. Types of Lexical Information. DD Steinberg, LA Jakobovits (eds): Semantics: an Interdisciplinary Reader in Philosophy, Linguistics and Psychology, Cambridge UP (1971).

[Gab89] Gabrieli E. A New Electronic Medical Nomenclature. *Journal of Medical Systems*; 3 (1989).

[Gai94] Gaines BR. Class Library Implementation of an Open Architecture Knowledge Support System. *International Journal of Human-Computer Studies*, 41 (1994).

[GAL92] GALEN Project. Documentation available from the main contractor Rector AL, Medical Informatics Group, Dept. [] Computer Science, Univ. Manchester, Manchester M13 9 PL, UK, (1992-4).

[Gan92] Gangemi A, Galanti M, Galeazzi E, Rossi Mori A. Compositional Semantics for Medical Records. in R Scherrer, S Mandil (eds.): *Proc. of 7th MedInfo92*, Génève (1992).

[Gan96] Gangemi A, Steve G, Giacomelli F. ONIONS: An Ontological Methodology for Taxonomic Knowledge Integration. in Vet (ed.) *Proceedings of ECAI96 Workshop: Ontological Engineering* (1996).

[Gan97] Gangemi A, Steve G, Pisanelli DM, Giacomelli F. Ontological Integration of Terminologies with ONIONS. in PJ Charrel, H Kangassalo (eds.) Proc. of European-Japanese Seminar on Information Modeling and Knowledge Bases, to appear (1997).

[Gth89] Genesererh MR, Nilsson NJ. Logical Foundations of Artificial Intelligence. Los Altos, CA, M Kaufmann (1989).

[Gen94] Gennari JH, Tu SW, Rothenfluh TE, Musen M. Mapping Domains to Methods in Support of Reuse. *International Journal of Human-Computer Studies*, 41, 399-424 (1994).

[Ger96] Gerstl P, Pribbenow S. Midwinters, Endgames, and Body Parts: A Classification of Part-Whole Relations. *International Journal of Human-Computer Studies*, 43 (1996).

[Gru93] Gruber T. A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*; 5:188-220 (1993).

[Grü96] Grüninger M. Designing and Evaluating Generic Ontologies. in *ECAI96 Wks Ontological Eng* (1996).

[Gua94] Guarino, N., Carrara, M., and Giaretta, P. An Ontology of Meta-Level Categories. In J Doyle, E Sandewall and P Torasso (eds.), *Principles of Knowledge Representation and Reasoning: Proceedings of KR94*. Kaufmann (1994).

[Har90] Harnad S. The Symbol Grounding Problem. *Physica D*, 42 (1990).

[Hum92] Humphreys BL, Lindberg DA. The Unified Medical Language System Project. in Lun KC et al. (eds): *Proc. of MEDINFO 92*, Amsterdam: Elsevier Science Publishers (1992).

[Lar96] Laresgoiti I, Anjewierden A, Bernaras A, et al. Ontologies as Vehicles for Reuse: a mini-experiment. in B Gaines, M Musen (eds), *Proceedings of KAW96* (1996).

[Len90] Lenat D, Guha R. Building Large Knowledge-Based Systems. Addison-Wesley, Reading, MA (1990).

[Leo40] Leonard HS, Goodman N. The Calculus of Individuals and its Uses. *Journal of Symbolic Logic*, 5 (1940).

[Mac91] Mac Gregor RM. The Evolving Technology of Classification-based Knowledge Representation Systems. in JF Sowa (ed.): Principles of Semantic Networks, San Mateo, CA, M Kaufmann (1991).

[Mac94] Mac Gregor RM. A Description Classifier for the Predicate Calculus. in *Proceedings of the Twelfth National Conference on Artificial Intelligence, (AAAI 94)* (1994).

[Mal94] Mallery JC. A Common LISP Hypermedia Server. in *Proceedings of WWW94* (1994).

[Mar95] Martil R, Turner T, Terpstra P. Knowledge Reuse in Technical Domains: The KACTUS Project. in *Proc. of The Impact of Ontologies on Reuse, Interoperability and Distributed Processing*, Unicom Seminar London (1995).

[McG93] McGuire JG, Kuokka DR, Weber JC et al. SHADE: Technology for Knowledge-Based Collaborative Engineering. *Journal of Concurrent Engineering*, 1 (1993).
[Mil76] Miller GA, Johnson-Laird PN. Language and Perception. Cambridge UP (1976).
[Mot95] Motta E. KBS Modeling in OCML. Modeling Languages for KBS, VU Amsterdam (1995).
[Nec91] Neches R et al. Enabling Technology for Knowledge Sharing. *AI Magazine*; fall (1991).
[Pat93] Patel-Schneider PF, Swartout B. Draft of the Description Logic Specification from the KRSS group of the DARPA Knowledge Sharing Effort (1993).
[Pri82] Prince G. Narratology. De Gruyter, Berlin (1982).
[Rec94] Rector A, Gangemi A, Glowinski A, et al.. The GALEN CORE Model Schemata for Anatomy: Towards a Re-Usable Application-Independent Model of Medical Concepts. in *Proceedings of MIE94* (1994).
[Riv96] RivaA, Ramoni M. LispWeb: a Specialized HTTP Server for Distributed AI Applications. *Computer Networks and ISDN Systems*, 28 (1996).
[Ros90 Rossi Mori A, Thornton A, Gangemi A. An Entity-relationship Model for a European Machine-Dictionary of Medicine, in RA Miller (ed.): *Proc. of SCAMC90*, Washington, IEEE Press, (1990).
[Ros97] Rossi Mori A, Gangemi A, Steve G, et al. An Ontological Analysis of Surgical Deeds. in *Proc. of Artificial Intelligence in Europe, AIME97*, 1997.
[Sch89] Schmidt-Schauss M. Subsumption in KL-ONE is Undecidable. in Proc. of the IEEE wks. on principles of KBS, Denver (1989).
[Sow96] Sowa JF. Top-Level Ontological Categories. *International Journal of Human-Computer Studies*, 43 (1996).
[Spe95] Speel PH. Selecting Description Logics for Real Applications. in A Borgida, M Lenzerini, et al. (eds): *Proc. of International Workshop on Description Logics*, Rome (1995).
[Std95] Stedman's Medical Dictionary, 26th Edition. Williams and Wilkins, Baltimore (1995).
[Ste96] Steve G, Gangemi A. ONIONS Methodology and the Ontological Commitment of Medical Ontology ON8.5. in B Gaines, M Musen (eds), *Proceedings of KAW96* (1996).
[Ste97] Steve G, Gangemi A. Some Theses on Ontological Engineering in the Context of ONIONS Methodology. in A Farquhar (ed.) *AAAI97 Spring Symposium on Ontological Engineering* (1997).
[Swa96] Swartout B, Patil R, Knight K, Russ T. Toward Distributed Use of Large-Scale Ontologies. in B Gaines, M Musen (eds), *Proceedings of KAW96 (Knowledge Acquisition Workshop)* (1996).
[Tat96] Tate A. Towards a Plan Ontology. *Journal of the Italian AI Association* (1996).
[Usc95] Uschold M, King M. Towards a Methodology for Building Ontologies. *IJCAI95 Workshop on Basic Ontological Issues* (1995).
[Val96] Valente A, Breuker J. Towards Principled Core Ontologies. in B Gaines, M Musen (eds), *Proceedings of KAW96* (1996).
[Van97] Van Heijst G, Schreiber ATh, Wielinga BG. Using Explicit Ontologies in KBS Development. *Int. Journal of Human-Computer Studies*, 1997.
[Var95] Varzi A, Casati R. Holes and other Superficialities. MIT Press (1995).
[WHO94] WHO. International Classification of Diseases 10th revision. Geneva: WHO (1994).
[Zdr97] Zdrahal Z, Domingue J. The World Wide Design lab: an Environment for Distributed Collaborative Design. Int. Conference on Engineering Design, Tampere (1997).