Topic driven access to scientific handbooks

Caracciolo, C.

CHAPTER 3

A Browsable Map for Logic and Language

In this chapter we describe the Logic and Language Links (LoLaLi) map: a structure that is inspired by the semantic approaches described in Chapter 2 (i.e., taxonomies, thesauri, semantic networks and ontologies) and whose purpose is to provide a browsable and searchable resource, to make explicit to end users, and especially non-expert end users, the internal organization of the domain. Importantly, it is also meant to provide a bridge to the *Handbook of Logic and Language* by providing links to relevant excerpts from the handbook. Thus, the LoLaLi map, as a searchable and browsable resource, is meant to support information seeking for a variety of information needs.

The chapter is organized as follows. In Section 3.1 we present the requirements we impose on a browsable map for the domain of logic and language. In Section 3.2 we describe the approach we followed to design and populate the map. In Section 3.3 we describe the pieces of information attached to the topics in it, and in Sections 3.4 the relations used. In Section 3.5 we highlight the features of the map. Section 3.6 is dedicated to a discussion of the modeling choices made, and in Section 3.7 we describe the practical aspects of editing and managing the LoLaLi map, including the editorial environment we used and the way the map is accessed through a web browser. Finally, in Section 3.8 we present the conclusions we draw from this work.

3.1 User Requirements

The intended end users of the LoLaLi map are assumed to have some, but not extensive, knowledge of the domain. They may have some technical notions but may not be aware

of in-depth details of the area. This choice has implications on the organization of the map and on the selection of its content. Also, end users are not supposed to have any background in knowledge modeling, which implies that the information to be shown to them and the way it is shown should be carefully planned. Given this picture of our intended user, the map should satisfy the following requirements:

Requirement A. Include relevant topics from the chosen domain.

Requirement B. Be informative for the audience addressed.

Requirement C. Avoid information overload.

None of these requirements can be further pinned down in a formal way because they all depend on many vague factors. All requirements refer to a common notion of end user, and Requirement A also refers to a notion of *coverage* that depends on the end user, on the domain at hand, and on the text to which we want to connect the map. Requirement C refers to a phenomenon that, although difficult to define sharply, certainly results from excessive demands made on the cognitive processes, in particular memory; it also depends on the background of the envisaged end user. Between requirements B and C there is a natural tension, because the more pieces of information are made explicit, the greater the information and cognitive load for the user. We now discuss requirements A, B and C in a bit more detail.

Requirement A. Include relevant topics from the chosen domain. A *topic* included in the LoLaLi map is any topic, notion or idea pertinent to the domain at hand that the domain experts developing the map consider *relevant* for the end user. The relevance of a topics is then decided by the authors of the map. We do not impose that *all* relevant topics be included in the map, as *total coverage* (with respect to the domain, and with respect to the *Handbook of Logic and Language*) is a goal that will only be achieved with difficulty, and over a long time, assuming that it is possible to achieve at all [Cimino, 1998]. We rather aim at incremental coverage to be achieved over time.

Requirement B. Be informative for the audience addressed. By *informativity*, we refer to the *type* of pieces of information that the map should include in order to satisfy the information needs of our envisaged end users. In this respect, we consider a map to be informative if it provides a representation of topics included and of the relations between topics, that is appropriate to the understanding of the users.

In order to be informative to our intended users, topics should be provided with some essential textual information about them, such as short definitions, together with a selection of fundamental relations to other topics in the map. These relations should be more detailed than simple ‘see also’ or ‘broader than’ relations, but they should not overload the user (see the following requirement) by applying too fine-grained distinctions that can only be grasped on the basis of a deeper knowledge of the area. Semantic

relationships have always played an important role both in the area of databases [Storey, 1993] and in the area of knowledge organization [Clarke, 2001], and considerable effort has been put into the classification of different types of relationships. The question for us is then: what are the right relationships (in type and number) for our purposes? We answer the question by considering the relations as dependent not solely on the domain, but also on the end users we address, both in terms of their background knowledge and in terms of the use we expect them to make of the map. We tried to meet these requirements when gathering the input provided by a group of domain experts (cf. Section 3.2).

Requirement C. Avoid information overload. The expression *information overload* is vague, as it may refer to a variety of circumstances in a variety of domains observed under a variety of possible perspectives. Information overload can be a condition resulting from receiving information that overwhelms one's short term memory capacity, or it can result from information that exceeds one's ability to benefit from it [Eisenberg and Small, 1993]. It can also be defined as the overwhelming feeling deriving from having too much information to deal with [Stanley and Clipsham, 1997]. Others have defined it as having more information than one can take in, or having information that goes unused because one simply lacks the time or motivation to process and understand it [Wilson, 1995]. Other views on information overload include those given in [Biggs, 1989, IEEE, 1995].

As far as we are concerned, information overload is related to both the "amount" of information provided and the "ability" to process it (themselves related to one another). In particular, this can result from a map that is too complex (in terms of relations among the topics), from a carelessly designed graphical user interface (GUI), and from confusing interaction modeling. These considerations played a role in the process of designing and populating the map (Section 3.2), and in designing the user interface for it (Sections 4.1 and 4.3). For example, the resulting set of relationships is relatively small compared to the number of relations that an experienced researcher might assign to topics in the domain. We are also aware of what a delicate issue the GUI is: as in any digital environment, it is all too easy to overload the user with information and visual hints, and the result is that the user is distracted and can have difficulties processing the information. One common form of information overload comes from being presented with too many items of information on a single screen, and it easily worsens when dynamic features are added to the interface. Other sources of overload are abrupt "jumps," forced by hypertextual organization of content, and by interfaces where the layout changes when moving from one screen to another. These types of information overload can be avoided by careful interface design, others have to be taken into account already when designing the map.

3.2 Design and Content of the Map

From an initial survey we conducted, it emerged that no reusable resources were available for our purposes. The ACM classification system [ACM, 1964] (see Section 2.4 and Figure 2.3), the closest to our needs, is too high-level and partially tangential to our domain and it only includes a BT/NT structure. Also, the terms (about 100) coming from work previously carried out within the LoLaLi project [Ragetli, 2001] were encoded as a set of Prolog facts and organized by means of BT/NT relations (see also Section 3.7). Therefore, we had to reorganize and extend the data we had.

Although the focus of our work is not on knowledge representation, the building of the LoLaLi map can be viewed as a knowledge engineering effort, where knowledge has to be elicited by domain experts, often with the constraints imposed by a set of user requirements. The issue of designing and populating structures representing domain knowledge has been widely studied, both in the area of knowledge engineering [Schreiber et al., 2000] and ontology design [Noy and McGuinness, 2001]. Given these similarities, we borrowed several techniques and insights from that community.

As mentioned earlier (Section 1.1), we wanted domain experts to provide both the design (i.e., which relations to use) and the population of the map (i.e., which topics to include in it). The implication of this is that we had to mediate between the different perspectives and approaches of the different domain experts, and between their input and the requirements we illustrated in the previous section. In Sections 3.3 and 3.4 we report on our current view of the map. This view results from an iterative process that included our analysis of the domain, preliminary interviews with subject experts (not knowledge representation experts), regular consultations with the User Centered Design group at Elsevier (the publisher of the *Handbook of Logic and Language*), and the user studies we report on in Chapter 4.

The content of the LoLaLi map comes from what was inherited from a previous phase of the LoLaLi project (re-organized according to the chosen set of relations), and from a substantial number of new contributions provided by domain experts in the course of the second phase of the project. Domain experts were asked to draw simple hierarchical representations of fragments of their choice, following the "laddering technique" [Corbridge et al., 1994] often used for knowledge acquisition. Domain experts were asked to only use the pre-defined set of relations, but in some cases they freely gave their own schematization of their area of expertise, pointing at relations that we did not include but they found important to represent (one of these cases is in fact the source of the discussion presented in Section 3.6). Finally, we provided some of the content ourselves, using the back-of-the-book index of the original paper version of the handbook as a basis. Our intervention was often required to provide a junction between fragments contributed by domain experts, or to ensure the desired level of correspondence between the map and the *Handbook of Logic and Language*. This very distributed way of populating the map shed interesting light on the issue of organizing a workflow involving different roles such as authors and editors. In the concluding chapter we reflect on this issue.

The development of the LoLaLi map was not set up as a formal knowledge representation or knowledge engineering effort. The LoLaLi map does not come with an associated formal semantics. We arrived at a "loose" representation of the domain at hand based on both principled and pragmatic considerations. First and foremost, the information elicited from our subject experts is not of a prescriptive nature but of a descriptive aimed one capturing how topics are actually used in the domain. Second, while the formality of knowledge representation languages may be appropriate for, say, exchanging data between knowledge bases, it does not fit well with the vagueness and ambiguity inherent in people's language usage, and may even be counter-productive—where knowledge transfer involves humans, there is no vehicle like natural language [Spärck Jones, 2004]. In other words, a formal presentation of the LoLaLi map is at odds with requirements B (informativity) and C (avoid overload). Moreover, given the data-driven and bottom-up approach used for retrieval in Chapters 5 and **??**, a formalization seems superfluous; the methods we test and evaluate in those chapters are independent of the structure of the map and only take into account the topics in it.

Before going into the details of modeling issues, a few remarks on the typographical conventions used. We write in sans serif the topics included in the map and in `type writer` the name of relations in the map. Examples and topics and relations that are mentioned in a general way (not because they belong to the LoLaLi map) are mentioned in 'single quote.' All graphical representations of (fragments of) the map presented in the rest of this chapter were realized by using Graphviz [GRAPHVIZ, 2007]; the arrows that connect topics to one another point upward (from the specific to the general), so that 'A $\longrightarrow$ B' reads as 'A is some sort of B.'

3.3 Topics

A topic in the LoLaLi map is anything that is judged relevant by the authors of the map and suitable for inclusion in it (cf. Requirement A). More generally, anything about which something can be asserted can be included in the map. Topics in the LoLaLi map are represented by means of a *title* and an information definition, or *gloss*.[1] A title consists of one or more (English) terms used to express that topic. Terms in a title are related to one another by a *synonymy* relation, i.e., the lexical relation between terms that can be substituted for one another in certain, but not necessarily all, contexts.

As pointed out in Section 2.4, the relation of synonymy represents a core relationship in WordNet [Fellbaum, 1998] since the meaning of a concept is represented by means of a *synset*: all terms that can be used interchangeably in given situations (partial synonymy). The underlying idea of using synsets in WordNet is that a term can be ambiguous when taken out of context, but is disambiguated by all its synonyms in a certain context. For example the term 'wood' can signify the hard fibrous substance under the bark of trees, or the trees and other plants in a large densely wooded area,

[1] Originally, a gloss is a note made in the margin of a book, to explain the meaning of the text.

or any wind instrument other than the brass instruments. The second and third case can be distinguished by the first one by using the synonyms 'forest' and 'woodwind instrument.'

In monolingual thesauri [Clarke, 2001], the relation between synonyms or quasi-synonyms (i.e., partial synonyms) is often indicated as an equivalence relation. However, in thesauri the terms used in a synonymy relation are often not true synonyms, but rather just terms whose meanings are sufficiently close, in certain contexts, for the purpose of the database to be indexed and searched. For example, the terms 'porcelain,' 'bone china' and 'crockery' are not real synonyms, but they could serve as such in a non-specific thesaurus for general use. Similarly, a general, non-specific thesaurus could contain 'linguistics,' 'historical linguistics' and 'formal linguistics' as synonyms.

Also in the domain of logic and linguistics, there are terms that can be used interchangeably. For instance, first order logic, FOL, and predicate calculus are synonyms, and together they form a title, the same way context free grammar and CFG do. Contrary to thesauri, and similarly to WordNet, we have no notion of preferred terms.

Glosses are short pieces of text, usually of about 2–3 sentences, added to each topic in order to give the user quick insight into the topic. Also, when two topics are known in the literature by the same name (with no synonyms available to distinguish one from the other) but have different meanings, their glosses will point out the differences between them. A number added next to the title also differentiates them, as is normally done in dictionaries. For example, the LoLaLi map contains a topic logic (1) under computer science, mathematics, artificial intelligence and linguistics, and a topic logic (2) under philosophy. The gloss of logic (1) is "A system of calculus or reasoning," the gloss of logic (2) is "The branch of philosophy that analyzes inferences" (Figure 3.1). Glosses also provide the de facto definition accepted by the author for a topic in the map.

3.4 Relations

The thesaurus relations broader term/narrower term (BT/NT) are very flexible in that they can be used to link any pairs of concepts such that one is "broader" than the other. Such a flexibility is to the detriment of expressivity, as pairs of concepts whose relations are intuitively very different are linked by the same BT/NT relation. Consider for example the following two pairs taken from [FAO, 2007]: Fishing Line is NT of Fishing Gear, and Adriatic Sea is NT of Mediterranean Sea. One of the purposes of the semantic relations of subclass and part-of is to account for these differences [Clarke, 2001]. It is also often customary to distinguish between subclasses and instances, where the latter indicate individuals belonging to a class (i.e., when the class is taken as a set).

These distinctions also apply to the domain of logic and linguistics, where we identified the need to be able to express the following: that a given topic *is a* type or specification of another topic (e.g., first order logic *is a* logic), that a topic *is part of* a broader

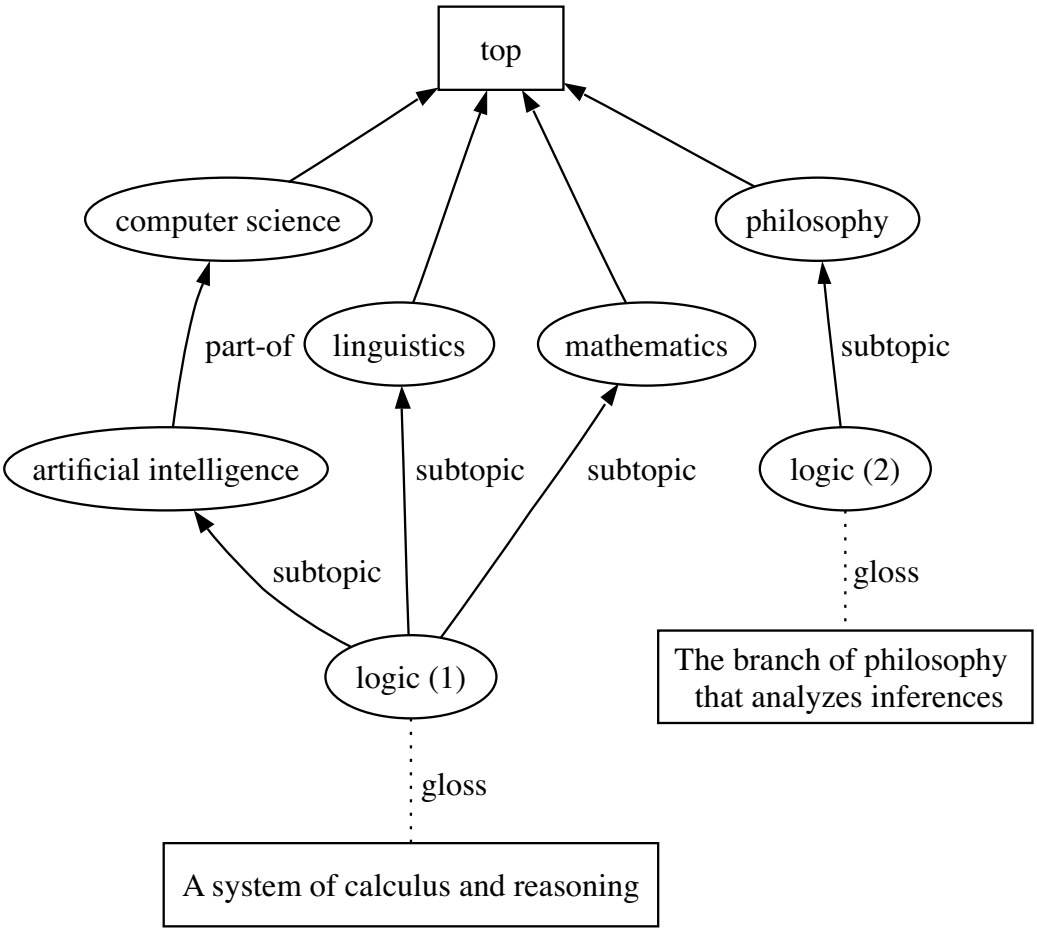


Figure 3.1: Glosses help distinguish two "senses" of the same topic.

topic (e.g., pragmatics *is part of* linguistics) and that a topic *is an instance* of an already defined topic (e.g., K4 *is an instance* of modal logics). Therefore, we included the following three relations, explained in Sections 3.4.1, 3.4.2 and 3.4.3, respectively:

1. `subtopic` (ST),
2. `part-of` (PO),
3. `instance` (IN).

In addition to these, we selected a small number of relations meant to account for important relations typical of the domain at hand: that a topic expresses the feature, or the "nuts and bolts," of the domain (for example, completeness and variable to logic); that a topic is a theorem (or lemma and the like) that holds in a given area; that a topic is a computational tools used or developed in a given area, e.g., for instance, a theorem

prover for logical systems and a parser for a natural language; and that a topic provides a historical perspective on a given topic, e.g., the treatment of quantifiers in aristotelian logic with respect to the current theory of quantifiers. These considerations lead us to consider the following domain-specific relations:

1. `features and internal machinery` (FI),
2. `mathematical result` (MR),
3. `computational tool` (CT), and
4. `historical view` (HV).

The domain-dependent relations are presented in Section 3.4.4. Figure 3.2 depicts a small fragment of the map.

3.4.1 Many Flavors of ISA

The notion that something *is a* type or specification of something else, is a fundamental notion for (at least) human cognition, linguistics, and mathematics. When called *subclass* [Brachman, 1983, Cruse, 1986, 2002], it is the most widely used hierarchical relation in the literature, but it is also widely known as *ISA* (i.e., is a), *AKO* (i.e., a kind of), *subsumption*, and *hyponymy* (inverse: *hypernymy*). As noted elsewhere [Brachman, 1983, Cruse, 2002], and as the abundance of different names for it should suggest, the apparent simplicity of the term subclass hides a gamut of slightly different semantics.

A common understanding of the subclass relation implies a reference to objects in the *real* world. Objects are then grouped into classes, which can be further specified to define subclasses (or subsets, subcategories). For example, a 'daisy' can be intensionally defined as "A small flower with white petals and a yellow center, which often grows in grass."[2] As a class, extensively defined, it would include all daisies in the world. Since all daisies are 'flowers,' it follows that the class 'daisies' is a subclass of the class 'flower.' Although intuitively appealing, this understanding is more problematic than it may seem at first: because it is not always possible to identify an object in the physical world to which to refer and, most importantly, because the grouping depends on some definition of the object to classify, i.e., it depends on the selection of suitable features describing an object.

The application of a strict set-theoretical view is not suitable to cover the (large) domain considered in this work, namely the area at the interface between logic and linguistics, although specific subareas do admit a formal description based on set theory. For instance, a system of logic can be taken as the set of formulas formed using

[2]This definition is taken from the Cambridge Dictionary, obviously not a dictionary for botanists. In the same dictionary we find the following definition for 'flower:' "The part of a plant which is often brightly colored with a pleasant smell, or the type of plant that produces these."

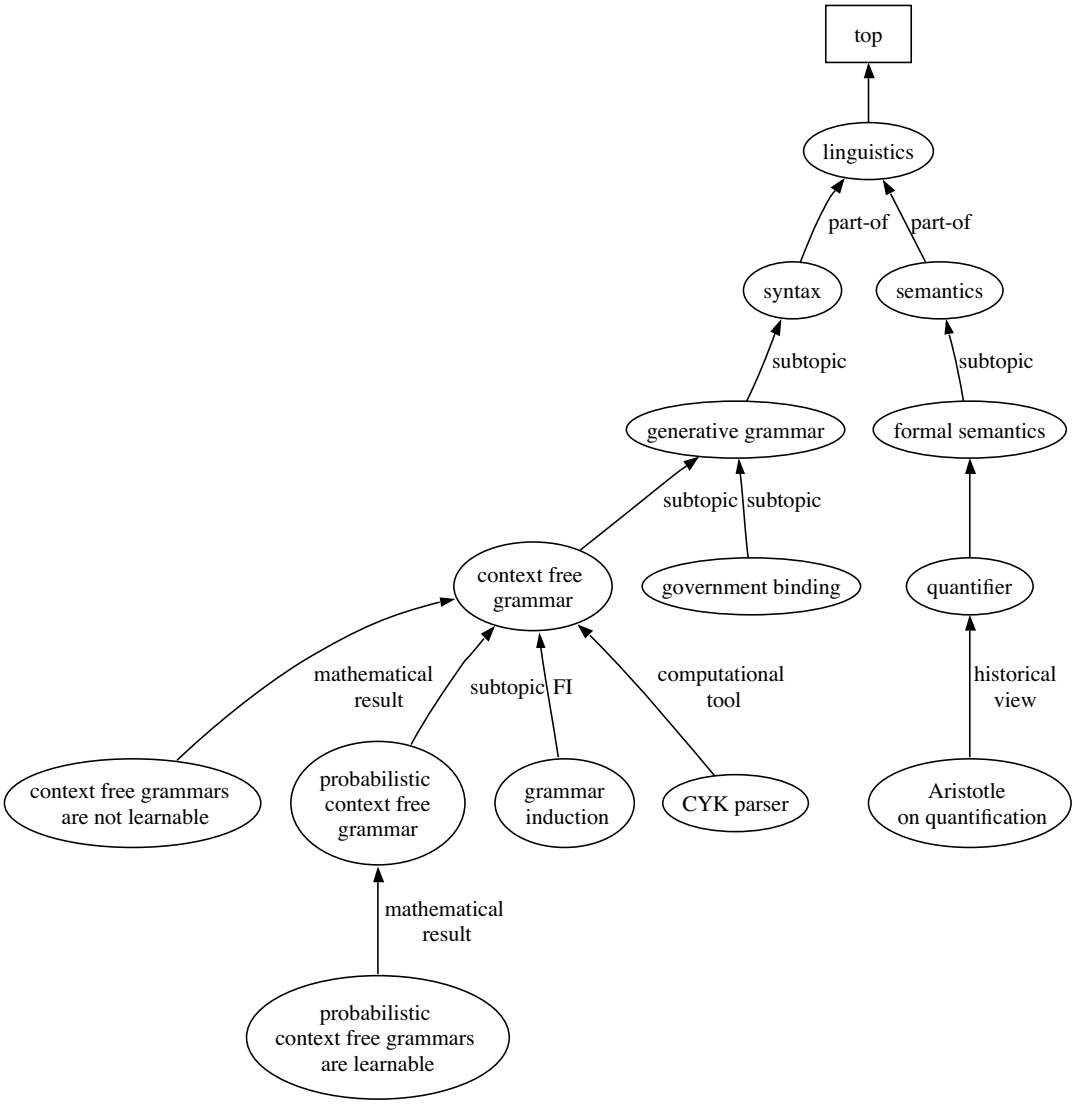


Figure 3.2: A fragment of the LoLaLi map.

a given alphabet and a given set of rules: the larger the alphabet, the more expressive power of the logic. A logic, A, can then be included in another logic, B (consisting of more formulas than A) and in this sense A is a subclass of B; on the other hand, since B utilizes a larger alphabet, it can be considered a "specification" of A, and often presented to non-expert readers "after" A for historical reasons. So for example, in a set-theoretical view, propositional calculus would be a subclass of first-order logic because propositional calculus has a smaller alphabet than first order logic, and all propositional formulas are also first order logic formulas.

Some authors, like Cruse [1986, 2002], distinguish a generic notion of subclass (then called simple hyponymy) from a more precise relation of taxonomy. A simple hyponymy corresponds to the sentence 'An X is a Y,' as in 'A white table is a table,'

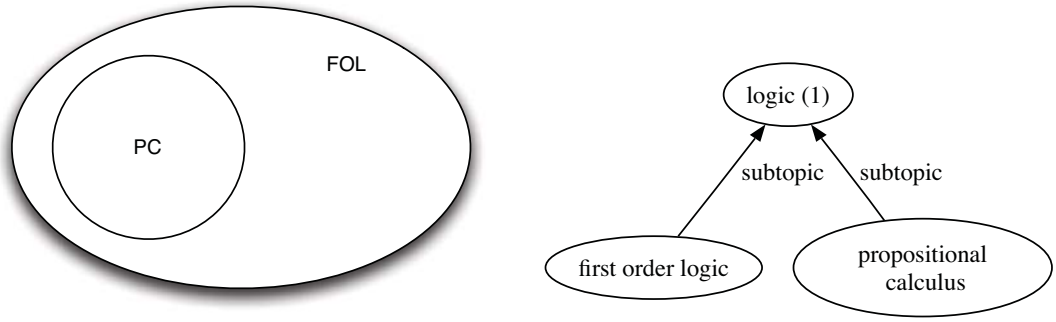
while a proper taxonym corresponds to the sentence 'An X is a kind of/type of Y,' as in 'A bedside table is a table' (cf. Section 2.4). It turns out that the distinction between the two relationships is not always clear, since 'A brown bear is a bear' can easily be interpreted as a taxonomic distinction. In fact, the notion of simple hyponymy seems to belong more to an investigation of natural language semantics than to a discourse on ontological relationships.

Several principles and heuristics have been proposed for "correctly" assigning the subclass relation to pairs of concepts. Here we only mention the most recent and influential. Cruse [1994] suggests that the decision about classifying subclasses should be made on the basis of a coherent *perspective* of the hypernym. The idea is that when organizing a class into subclasses (or taxonomies) we have to select one or more features and use them to identify subclasses. To use his example, 'stallion' is a bad taxonomy for 'horse' because it does not adopt the same perspective, while 'ash blond' is a good taxonomy for 'blond,' because they both adopt the same perspective. Later on, trying to further clarify the notion of perspective, Cruse [2002] proposes to choose categories that are (a) internally cohesive, (b) externally distinctive and (c) maximally informative. Storey [1993] observes that the subclass relation is often confused with various types of part-of relations, because they all imply membership of an individual in a larger set (cf. next section). In practice this distinction leads to an implicit heuristic for deciding when a subclass relation should be assigned.

Others [Guarino and Welty, 2000, 2002a,b] have made attempts to give a general, well-defined notion of the subclass relation by focusing on the ontological nature of the arguments involved in it, more than on the relation itself, and using the philosophical notions of identity, unity and essence. Using these criteria as landmarks, they analyzed several ontologies and found many inconsistencies. Unfortunately the complexity and abstraction of their method has hampered a broad adoption in real life applications.

The considerations expressed above, together with Requirement C (Section 3.1) and observations made during the user studies reported on in Chapter 4, made us opt for the notion of subclass that corresponds to the taxonomic relation of 'is a type of'. Then, in order to avoid confusion with a very loaded term, we prefer to talk about *subtopic*. For example, formal semantics `is a subtopic of` semantics, or modal operator `is a subtopic of` operator. So, in the view we adopt, first order logic and propositional calculus are both subtopics of logic as they are both formal systems, but with different properties. Figure 3.3 presents a set-theoretical view of them (a) and the organization used in LoLaLi (b). Similarly, in our view hybrid logics are subtopics of modal logics, despite the fact that syntactically hybrid logics are extensions of modal logics (i.e., they include more operators than just modal operators).

Summarizing, given the requirements imposed in Section 3.1 and the breadth of the domain at hand, we opted for an intuitive notion of subclass, that we call `subtopic`.



(a) (b)

Figure 3.3: (a) First order logic (FOL) and propositional calculus (PC) viewed as sets of formulas. (b) First order logic and propositional calculus as subtopic of logic.

3.4.2 Part-of

The relation between the parts of a whole is called *part-of*, *part-whole* or *meronymy* (inverse: holonymy) [Artale et al., 1996, Cruse, 1986, Simons, 1987]. While the decomposition of classes into subclasses leads to taxonomies, the decomposition into parts leads to partonomies [Tversky, 1990].

While taxonomies have been studied extensively in many domains, this has not happened for partonomies, although from a cognitive point of view, there is evidence suggesting that children can process meronymy relations earlier than taxonomic relations [Inhelder and Piaget, 1964]. The development of a theory of parthood at the beginning of the last century aimed at defining a single theory that, unlike set theory, is founded only on concrete entities, but there is no unanimous agreement on its semantics. The basic problem of mereology as a field of study is that there appear to be various, often inconsistent, semantics associated with the term 'part-of.' In fact, the same expression 'to be part of' is used to describe arguably different situations, as in: 'A leg is part of the body,' 'A boat is part of a fleet,' and 'A window is part of a car.' Consider the following example taken from [Salustri, 1998].

1. A piston is a part of an engine.
2. An engine is a part of an automobile.
3. An automobile is a part of a fleet.

Each statement, on its own, is perfectly reasonable. Furthermore, from statement 1 and 2 we can reasonably deduce that a piston is a part of an automobile, but from all three statements, can we reasonably deduce that a piston is a part of a fleet?

Some authors, like Srzednicki et al. [1984] and Gruber [1992] consider the parthood relation to be single, universal, and transitive, meaning that all distinctions about types of parts are really conceptualizations and are not rooted in reality; they use first-order logic to introduce sufficient predicates to distinguish between kinds of things.

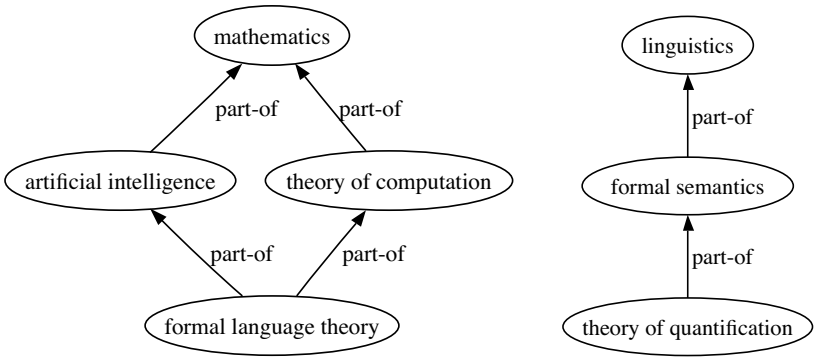


Figure 3.4: Two examples of chains of part-of relations in the LoLaLi map.

Artale et al. [1996] believe that a proper mereology must handle the transitivity problem directly by admitting distinctions between different part-of relations. In that approach, different part-of relations are explicitly defined to handle different conceptualizations (e.g., assembly/component versus space/region), and transitivity is not preserved across them.

In the LoLaLi map, the relation `part-of` (PO) is applied to abstract entities and its meaning is taken in an abstract sense, to indicate the main *parts* into which a domain is articulated. For example, syntax, semantics and pragmatics are all parts of linguistics. We also consider this relation to be transitive. We believe that, for navigational purposes this is most natural/appropriate. Figure 3.4 shows some examples of chains of `part-of` relations currently included in the LoLaLi map.

3.4.3 Instance

As previously hinted, the relations `is an instance of` (IN) is taken in many semantic structures to mark the belonging of a physical object to a class. In the LoLaLi map, this relation is used to relate specific examples to a more general group. Specifically, this happens often in the case of logical systems, when a specific set of axioms in a given language is studied *per se*. This is a common situation, e.g., in modal logic, where logical systems such as K4, S4, S3, KD4 are all `instances` of modal logic.

3.4.4 Domain Specific Relations

In this section we present the relations introduced in order to capture the relationships between topics specific to the specific domain we deal with.

Features and Internal Machinery

The relation `features and internal machinery` (FI) is a domain-dependent relation, introduced in order to point out to our end users what the fundamental notions

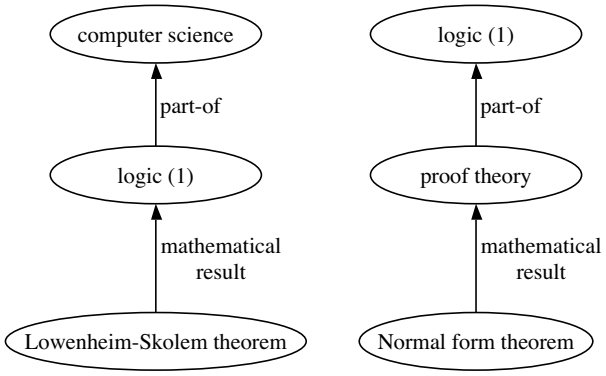


Figure 3.5: Two examples of the relations `mathematical result` (MR) in the LoLaLi map.

in the subject are, i.e., the nuts and bolts of the domain. Also notions used to classify topics in a domain are covered by this relation. Examples of these relations are: variable and constant for logic (1), and part of speech in linguistics. Completeness and soundness are both `FI` of a logic (1), while arity is a `FI` of operator, since it indicates how many arguments an operator takes.

Mathematical Result

The relation `mathematical result` (MR) links a mathematical statement to the topic representing the area to which it is relevant. Theorems, corollaries (i.e., propositions inferred immediately from a proven proposition with little or no additional proofs), and lemmas (i.e., an auxiliary proposition used in the demonstration of another proposition) are all `mathematical statements` of their supertopic. Figure 3.5 depicts two examples taken from the LoLaLi map.

Computational Tool

The relation `computational tool` (CT) is another domain-dependent subtopic relation. This relation has been introduced to emphasize the role of computational systems like theorem provers in logic or parsers in computational linguistics. For example, SPASS [Informatik, 2006], an automated theorem prover for first-order logic with equality, is a `computational tool` for first order logic, a CYK parser is a `computational tool` for context free grammars (Figure 3.2), and HyloRes [Areces et al., 2001, Areces and Heguiabehere, 2002] is a `computational tool` for hybrid logic.

Historical View

The relation `historical view` (HV) has been introduced to account for the historical evolution of topics. For example the topics Aristotle on quantification and Frege on quantification are historical views of the more general topic quantification. Given

the coverage of the handbook, this relation turned out to be the least used in the LoLaLi map.

3.4.5 Non-Hierarchical Relations

The relations `related` and `antonomy` have been introduced to make explicit the connections between topics to which no hierarchical relation is applicable. They are non-hierarchical in the sense that there is no implication that one of the topics involved in the relation is more "general" than the other; these relations are symmetric.

Related

The relation `related` is a non-hierarchical binary relation, mainly introduced for ease of navigation among topics. It connects concepts that are for some reason judged as similar or connected, such as quantification and quantifier, or reference and referent. In order to provide more information about the nature of each pair of topics, whenever possible a comment is added to explain to the reader what kind of connection links the two concepts. We assume that the `related` relation does not hold between pairs of concepts already connected by any hierarchical relation, but siblings under the same parent can be linked as `related`.

Antonymy

Pairs of topics that are opposite to one other are linked by the `antonymy` relation. This relation used to be widely used in dictionaries. Though not that frequently used anymore in modern dictionaries, the antonym relation is an important relation in WordNet [Fellbaum, 1998]. For example, the following two word senses in WordNet are antonyms of each other:

- black, sense 6, ((board games) the darker pieces)
- white, sense 9, ((board games) the lighter pieces)

In the LoLaLi map the `antonymy` relation only applies to pairs of topics that are siblings under the same parent and hold the same relation to the parent (usually the `feature and internal machinery` relation). Examples are completeness and incompleteness under logic (1).

3.5 The LoLaLi Map: Features

In this section we present the figures concerning the current status of the LoLaLi map and describe its structural features.

The LoLaLi map is organized into four main branches: computer science, mathematics, philosophy and linguistics. These topics are gathered together under an empty

Total # of topics	547
Number of subtopic relations	629
Number of part-of relations	42
Number of historical view relations	6
Number of instance relations	6
Number of mathematical results relations	11
Number of computational tool relations	2
Number of FI relations	7
Maximal depth	9
Average outdegree	1.1
Maximal outdegree	32

Table 3.1: Details about the LoLaLi map.

topic called TOP, which serves as the root of the map. Under those four first level topics all other topics find a place. The nature of the relations between each of these four topics and TOP is not specified, since TOP is only used for ease of computation and for technical reasons. At the time of writing, the LoLaLi map consists of over 500 topics, mainly under the branch logic (1). Table 3.1 presents key figures concerning the map.

Mathematically, the LoLaLi map is a graph, where nodes are topics and arcs are relations. When restricted to hierarchical relations, the map is a connected, acyclic and directed graph.

Connected Graph

It is always possible to find a path from any topic in the graph to the top node. This means that disconnected topics or isolated subgraphs are not allowed. This feature is also used as an intuitive constraint in the process of building the map: all new topics must be attached to existing ones. Thus, at no stage of the creation process do we have isolated subgraphs. This feature is important also from the end user perspective, as it ensures that we have, at any point of time, a map that is always completely browsable (i.e., both top-down and bottom-up).

Oriented and Acyclic

Since edges in the graphs are directed (the edges have arrows), the graph is oriented. The graph may not contain cycles of hierarchical relations (i.e., `subtopic`, `parthood`, `instance`, `mathematical result`, `FI`, `computational tool`), meaning that no concept can be an ancestor of itself. Obviously, the non-hierarchical relations `related` and `antonymy` are not affected by this constraint, because these relations do not imply a direction between the topic they connect.

It can be argued that cycles of relations do not represent a flaw, since many dictionaries do have cycles in their definitions, but in our view this would be a source of problems and confusion for our map, especially for its navigation, because it would lead to counterintuitive situations of more general topics being placed underneath more specific ones.

Multiple Parenthood

By using a structure with multiple parenthood we stress the fact that a topic can be relevant, and therefore connected, to more than one area. In this sense, the multiple parenthood structure also allows one to use multiple classifications of the same object: for example a modal operator is a particular kind of operator (i.e., a subtopic of it), but also part of a modal language (Figure 3.6).

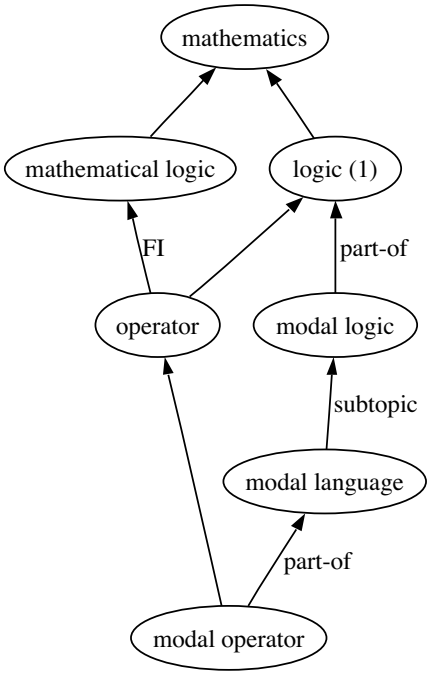


Figure 3.6: Multiple parents as multiple classification: the case of operator.

Multiple parenthood structures are typical of many multilingual thesauri, although in semantic networks they are often called "tangled hierarchies" [Fahlman, 1979]. WordNet is an example of such a tangled hierarchy.

Different Degrees of Kinship

In a graph there can be more than one path connecting a pair of nodes. In the LoLaLi map we also allow topics to have a hierarchical relation with both a topic and

one of its direct descendants: we say that the two topics have more than one degree of kinship. Figure 3.7 presents such a case, where the topic logic (1) can be reached in two steps from top, either passing through mathematics, or through linguistics, or through computer science. The same topic logic (1) can also be reached in three steps from top, by passing through computer science and artificial intelligence. In other words, the topic computer science is both parent and grandparent of logic (1).

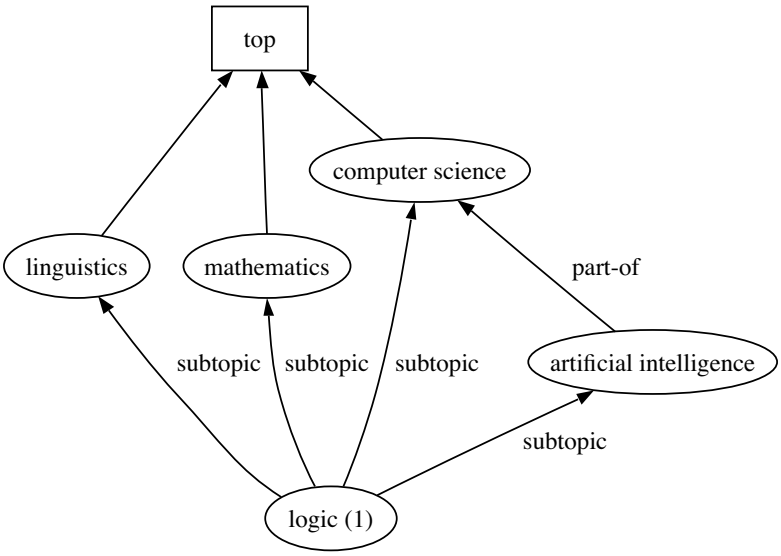


Figure 3.7: Logic (1) can be reached from top by more than one path. Different degrees of kinship: logic (1) is connected to both computer science and artificial intelligence.

3.6 Discussion

When dealing with an area as broad as the interface between logics and linguistics, the difficulties in selecting relations that are appropriate to the domain, in type and in number, and that can be informative to the end user without overwhelming her. In this section we report on what we learned from our implementation of the LoLaLi map.

3.6.1 Dealing with more Relations

Since the *Handbook* treats the domain at the interface of logic and linguistics, it includes topics that encompass inter-connected areas and that have contributed to the discovery of more connections among them. Following the style adopted in the *Handbook*, we call these subjects *frameworks*. An example of a framework is 'categorial grammar,' in which concepts from linguistics, computer science, and mathematics play an important role. Table 3.2 lists these three areas with the relevant subtopics for 'categorial grammar.' For a more pictorial view, consider Figure 3.8, where solid and dashed

Categorial Grammar relates to:

Linguistics	*Computer Science*	*Logic*
quantification	complexity theory	proof theory
pronoun		lambda calculus
coordination		category theory
polarity		modal logic
		type theory

Table 3.2: A summary of connections between the categorial grammar (a framework) and other areas.

lines are used to represent connections among some of the above-mentioned areas that were previously known and unknown, respectively. Some connections among these fields had already been established before the beginning of categorial grammar as an independent area of study (solid lines). For example, connections between complexity theory, proof theory and modal logic (in logic), and connections between lambda calculus, type theory and quantification (in linguistics). Other relations were created by categorial grammar itself (dashed lines): between proof theory and quantification, between quantification and category theory, between category theory and pronouns, coordination, and polarity items (the latter three being linguistic phenomena). In order to accommodate these connections, the LoLaLi map should use additional specialized relationships, but in so doing we would only talk to advanced readers, experts in the field or researchers. This choice would lead us to violate both Requirement B and Requirement C in Section 3.1 (about informativeness and information overload, respectively). Therefore we decided to introduce a topic categorial grammar connected to the relevant topics (i.e., linguistic phenomena, quantification, lambda calculus and so on) by means of the non-hierarchical relation `related`. A comment is included to explain the type of relatedness.

Another issue arose when treating topics for which more than one classification schema was possible. Consider, for example, the notion of operators in logic. Operators can be classified by arity (an operator can be unary, binary and so on) or by type (it can be a modal operator, a truth functional operator and so on). By adopting both perspectives, we can model different taxonomies of operators at the same time. Note that this issue is connected to the one of *perspective* we mentioned when discussing the subclass relation in Section 3.4. Instead of adopting a single "perspective" to distinguish subtopics of a topic, we list all possible taxonomies. This way we are able to give as complete an account as possible of the various subtopics of certain topics.

3.6.2 More on Subtopics

Intuitively, if a topic has a property, its subtopics should also have that property. For example, if 'apple' is a subtopic (rather, subclass) of 'fruit,' and the property 'has seeds' holds of 'fruit,' then it also holds of 'apple.' In the case of the LoLaLi map, the decision to adopt what we called an intuitive view of the subtopic relation (as opposed

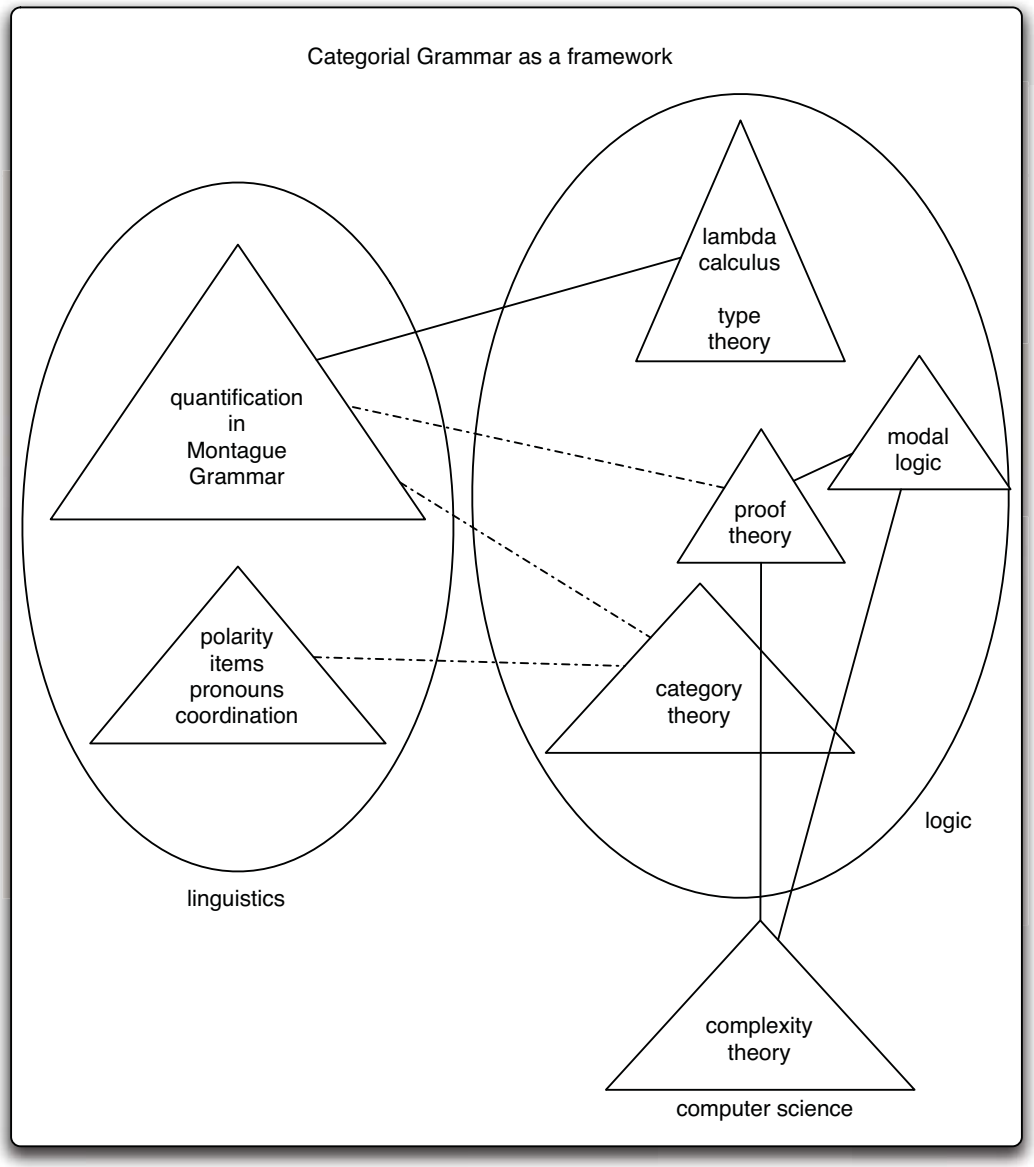


Figure 3.8: Categorial grammar as a framework. Dashed lines stand for relations created by categorial grammar. Solid lines indicate previously known relations. Ellipses group together topics in the same area.

to the set theoretical one) has interesting consequences in this respect. Consider the above-mentioned theorem prover HyloRes, which in the LoLaLi map is a computational tool for Hybrid logic (Figure 3.9). Had we adopted a set-theoretical perspective, modal logic would actually be a subtopic of hybrid logic, because hybrid logics add hybrid operators to modal logics (i.e., the set of hybrid formulas is larger than the set of modal formulas). As a consequence, HyloRes would also be a computational tool for modal logic, which is actually true, but arguably not very informative (hybrid operators

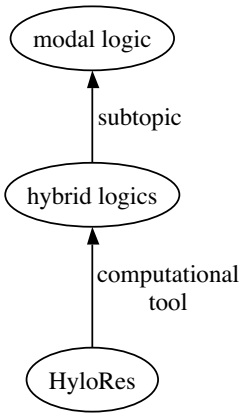


Figure 3.9: Hybrid logics in LoLaLi.

should be ignored). In this sense, our notion of subtopic relation matches well with the purpose of the LoLaLi map.

As another example, let us consider SPASS, a theorem prover for first order logic, and the example previously given about the modeling of first order logic and propositional calculus (Section 3.4.1 and Figure 3.3). In the LoLaLi map, both First order logic and propositional calculus are subtopics of *logic*, and SPASS is *only* a subtopic of FOL, which suits our purposes well. If we had taken the set-theoretical perspective (Figure 3.3 (a)) we would have "learned" that SPASS is also a theorem prover for PC which is, again, true but not very informative.

Let us consider the fragment of the map depicted in Figure 3.10 and the relation `mathematical result`. Probabilistic context free grammars is a subtopic of context free grammars, and the map reports the following mathematical results: "probabilistic context-free grammars are learnable," while "context-free grammars are not learnable." Both theorems refer to the concept grammar induction (the possibility of learning a grammar), but in opposite ways. What normally happens in the case of subclass relations is that the amount of specification (or properties) increases when going down the line of subclass. As we have just mentioned, this also happens in our examples, as by adding information (the constraint for the grammar to be probabilistic) there is a "new" result that does not hold for the supertopic context free grammar. The difference between this situation and other taxonomic structures is that, when looking "down" from supertopic to subtopic (from CFG to PCFG) we cannot assume an inheritance of properties. On the other hand, when looking up, from subtopic to supertopic, we have the same specificity of information (theorem on learnability) but of different type/content (whereas in usual taxonomies, supertopics are less specified than subtopics): we have specific results (mathematically proven) also at the level of supertopic and not only at the level of subtopics. If inheritance of theorems were allowed, we would mistakenly conclude that probabilistic context free grammars, just as context free grammars, are not learnable. Once again, given the specificity of the domain at hand, it turned out

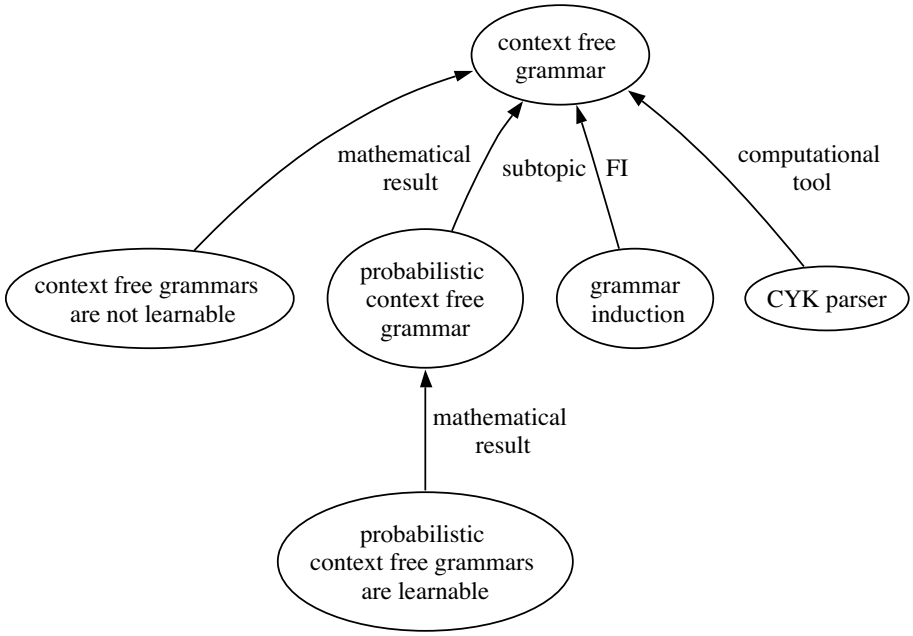


Figure 3.10: Two mathematical results for context free grammars.

to be a good choice to avoid the adoption of a set-theoretical notion of subclass and avoid the notion of inheritance altogether. These anecdotal observations support our modeling decision to adopt an intuitive interpretation of the subtopic relation.

3.7 Editing and Managing the LoLaLi Map

In this section we present and discuss the solutions we adopted to edit and maintain the map, in terms of the software used for editing, the choices made to model the LoLaLi map in RDFS, and how we made the map accessible through the web.

3.7.1 A Bit of History: First Attempts

The first version of the LoLaLi map, consisting of about 100 topics, was encoded as a collection of Prolog facts connected by BT/NT relations. The visualization of these topics was done by means of a Java applet (for more details about this version, see [Ragetli, 2001]) that resulted in something that was difficult to read and not scalable.

A second version of the LoLaLi map was encoded in XML [XML, 1998], and provided with a Document Type Definition (DTD). Each topic was represented as an XML document, and all pieces of information attached to each concept, i.e., gloss, description, references to the parent and child topics, metadata about the author of the

node and the last update, and references to related and antonym topics were allocated to distinct elements. The entire hierarchical structure was stored in a relational table. An HTML version of the map was periodically generated and published online.

The XML encoding of the map had major drawbacks, especially for the management of references to parent, child, and related and antonym topics. Moreover, the lack of suitable tools for managing the XML repository hampered the gathering of contribution to the map from authors. We explored the possibility of using XLink [XML, 2001] (an XML based language able to manage possibly multidirectional links between XML documents or parts thereof), but we were confronted with the lack of mature software tools. As it turned out, it was more convenient to store the map structure in a relational table instead of using an XML document. Finally, search within the collection of XML documents was not sufficiently supported (only string search was possible; at the time it was not possible, for example, to perform a structured search, i.e., within elements), nor was visualization of the map (only possible by means of the HTML interface, to be generated after every update). In practice, very little support for the editorial process was available.

In order to cope with these limitations, the map was converted into RDFS format, the vocabulary description language for RDF, because of its emerging status as an accepted standard and the large amount of software already available for it. In particular, the adoption of RDFS allowed us to take advantage of Protégé, an ontology editor able to produce RDFS output, and of off-the-shelf tools (i.e., Sesame, see Section 3.7.3) to enable access to the map through the web. For more details about the conversion from XML into RDFS, and about the use of Sesame as back-end, see [van Schie, 2003]). In the following subsection we present the main features of Protégé and discuss our modeling decisions.

3.7.2 Protégé and RDFS Modeling

Protégé [Gennari et al., 2003, Noy et al., 2000, 2001] is an ontology editor developed at the University of Stanford (precisely, at the Stanford Medical Informatics group within the Stanford University School of Medicine.)

Protégé is an ontology editor with an easy-to-use graphical interface that allows one to enter *classes* by means of forms. It support the visualization of class hierarchies (in an indented-tree like fashion), properties and instances. Also, its architecture is well suited for the inclusion of plug-ins, many of which have been made available by a large community of users. Protégé allows one to save and export data in various formats, including RDFS.[3] Concepts in a domain are modeled as *classes*, which can have *properties*, while actual data is modeled as *instances* of class(es).

The version of Protégé we used[4] allowed the modeler to impose constraints, such

[3] As with RDFS, Protégé's internal format also uses the notions of classes, class properties and instances of classes; for a detailed discussion of the relationships between RDFS concepts and native Protégé concepts see [Noy et al., 2001].

[4] Protégé 1.8.

as cardinality constraints on the number of required properties and the definition of inverse properties, that could however not be exported into RDFS. For this reason we decided not to use them at all. Also, the version of Protégé that we used supported one namespace only, it was stand-alone, did not manage multiple users working on the same project, and it did not implement user rights restrictions. This fact forced a very centralized work flow, with one person controlling all the data.[5]

Each concept in the LoLaLi map is modeled as an RDFS class endowed with a number of properties to accommodate glosses, topic descriptions and link(s) to the *Handbook* (these pieces of information are meant to be shown to the end user), and comments, creation/modification timestamps, and data about the author of each topics (these pieces of information are mainly for editorial purposes). A Protégé *metaclass* (ConceptClass) is used to serve as a template for creating new classes. Since the subtopic relation used in the LoLaLi map does not exactly coincide with the strict set-theoretical notion of subclass, we opted for modeling all relations, indistinctly, as properties.

When the data was first converted into RDFS, all relations were rendered as a property called 'Unspecified subtopic' whose refinement (according to the relations presented in Section 3.4) was carried out in parallel with the conversion and implementation of the GUI. As we will see in Chapter 4, the user studies were run in this phase and they actually contributed to the final definition of the set of relations.

By modeling all relations as properties (including the relation subtopic) we miss the opportunity of using the built in semantics of subclass in Protégé. We argue that for our purposes the loss is minor. First of all, the lack of inheritance of slots is not a problem, because this feature was not used in the modeling of the map. Also, for editing purposes we make use of a metaclass as a template to create new classes. A possibly useful application of the Protégé built-in subclass relation is that we could in principle check for cycles of subclass relations and provide an aid to the maintenance of the map. We lose this possibility because we cannot specify such a constraint at the level of class slots. In our experience this was not a problem, but it could be a problem if the map grew considerably or if more people were allowed to modify the repository. As for the maintenance of the map, Sesame proved to be a useful tool, as we defined a set of queries (for example, to check for cycles, and extract all subtopics of given topics) that was regularly run against the repository.

3.7.3 Accessing the Map through a Web Browser

In order to make the LoLaLi concepts browsable through a web browser, we needed a middleware system to interface the RDFS repository with the web. One way to do it is to parse the RDFS repository and convert it into static HTML pages. Of course, this solution implies that any change in the RDFS repository requires that the entire repository

[5] Many of the limitations of Protégé 1.8 have been overcome by the latest versions of it. The latest version of Protégé does support multiple users, although it still does not support rights management, and can be extended by a plug-in that allows one to edit ontologies in OWL [Knublauch et al., 2004].

be re-converted. Another solution is to use an RDFS repository manager connected to the user interface: for this purpose, we adopted Sesame [SESAME, 2005], an open source RDFS-based storage and querying facility. Sesame uploads the RDFS repository, parses it, and enables querying it, by means of query languages such as the RDF Query Language, RQL [Karvounarakis et al., 2002] and SeRQL [SeRQL, 2005].

At the time the LoLaLi map was implemented, the RQL language was the more stable and developed language. RQL is similar to the well-known relational query language SQL. We created fixed queries corresponding to the navigational actions performed on screen. Each click in the interface (see Chapter 4 for details about the interface) triggers an RQL request to Sesame. The solution RDFS + Sesame + RQL proved to be a suitable solution for us (although with some overload by the client, as RQL returns data in XML format, to be parsed on the client side, in tabular form, therefore with duplications). At the time of writing the RQL language is still supported but no longer actively developed. Currently, SeRQL is becoming the default language used to query a Sesame repository.

Summarizing, Protégé was used for editing the map (after conversion in RDFS format) and Sesame was used to access data in the repository and pass it to the user interface. Sesame cannot be used as a search engine for the end user (nor is it advisable to use it to support the everyday maintenance work on the map), because of the complex query language and the format in which results are shown, Instead, we provided the map with a search engine tailored to it, that allows any text to be typed in by the user and searches against every piece of information available in the map, but with meaningful assumptions on common searches and informative results (for details about the search engine see [van Hage, 2004]).

3.8 Conclusions

In this chapter we described and discussed the modeling of the LoLaLi map, covering the area of logic and linguistics. We presented the requirements we imposed on the map and the approach used to design and populate it. We also discussed its features and presented its implementation.

The relations used within the map were especially chosen to meet two of the user requirements we imposed: that the map be informative for the end user we address, and that it avoid information overload on the user. The resulting set of relations, selected on the basis of a mixture of empirical, pragmatic, and principled considerations, directly addresses our primary concern, namely the definition of a map oriented to human browsing and navigation. Anecdotal observations on the domain confirm that the adopted solution is appropriate to our aims, while the user studies on which we report in next chapter will shed some lights on the interaction of the users with the map.

The LoLaLi map is not endowed with a formal semantics, for the twofold reason that it is neither necessary for the type of users we aim at, nor for the type of connection to the text that we are investigating and reporting on in Chapters 5 and **??**. The draw-

back of this decision is that the possibility of consistency checking mechanisms are not exploited, but given the scope of our work, and the envisaged usage of the map, this is not a major problem. Instead, what turned out to be a source of difficulty during our work was the process of populating the map because we did not distinguish between different roles for the people involved in it and consequently could not benefit from an organized editorial flow. In Chapter 7 we reflect on this and related issues.

Finally, we remark that in the short time since we first started working on the encoding of the LoLaLi map in semantically oriented languages the field has greatly progressed, in terms of new and richer semantically-oriented languages (e.g., OWL), improved editing tools for ontologies, more sophisticated languages to query RDFS repositories (e.g., SeRQL). We are dealing with rapidly evolving technologies, where languages, tools and facilities are proposed, accepted as standard, and then given up in rapid succession.

In the next chapter we move our attention to the end users of the LoLaLi map: we present the graphical interface dedicated to them and report on the user studies we performed.