LIRICS

Deliverable D4.1

Methodological aspects of semantic annotation and representation

Project reference number	e-Content-22236-LIRICS
Project acronym	LIRICS
Project full title	Linguistic Infrastructure for Interoperable Resource and Systems
Project contact point	Laurent Romary, INRIA-Loria 615, rue du jardin botanique BP101. 54602 Villers lès Nancy (France) romary@loria.fr
Project web site	http://lirics.loria.fr
EC project officer	Erwin Valentini
Document title	Methodological aspects of semantic annotation and representation
Deliverable ID	D4.1
Document type	Report
Dissemination level	Public
Contractual date of delivery	M9
Actual date of delivery	30th September 2005
Status & version	Draft
Work package, task & deliverable responsible	UTiL
Author(s) & affiliation(s)	Harry Bunt and Amanda Schiffrin, UTiL
Additional contributor(s)	Marieke van Erp
Keywords	Annotation, Semantic Representation

Document evolution

version	date	version	date
1.0	30th September 2005		
1.1	13th October 2005		
1.2	9th November 2005		

1 Introduction

At a time when semantic content annotation is proving to be at the cutting edge of rapid and accurate information extraction of all varieties, there is some urgency in setting standards for reusability and interoperability of resources for wide application and distribution. The LIRICS project therefore has as one of its aims to propose a common and well-defined set of descriptors for semantic annotation in the form of data categories in an on-line registry, in accordance with ISO standard 12620 (Terminology and other language resources – Data categories for electronic lexical resources; see Romary 2004).

The work in this part of the LIRICS project is performed in Work Package 4 in close collaboration with two related initiatives. First, the area of semantic content annotation has been recognized as important by the International Organization for Standardization (ISO), which has formed the ad hoc Task Domain Group TDG 3 devoted to this area, within Technical Committee 37, Subcommittee 4 (Language Resources Management). In addition, an independent expert group was formed within the ACL Special Interest Group on Computational Semantics (ACL/SIGSEM), the Working Group on the Representation of Multimodal Semantic Information (MMSemR). The work in WP 4 of the LIRICS project is carried out in close collaboration with the activities of this Working Group and in ISO TC 37/SC 4/TDG 3, as witnessed by a joint workshop that took place in January 2005 in Tilburg (see http://let.uvt.nl/research/ti/iso-tdg3).

So far, the activities have consisted primarily of:

1. identifying commonalities in alternative approaches to the annotation and representation of various types of semantic information; and

2. developing methodological principles and concepts for identifying and characterizing representational concepts for semantic content.

The present deliverable is concerned especially with the latter point, inspired by discussions in the MMSemR Working Group and by methodologies that have been developed in ISO activities in other domains of annotation of linguistic resources.

2 Semantic information, annotation and representation

Any effort to annotate or represent semantic information presupposes a delineation of what is meant by semantic information. We follow Bunt & Romary (2002) who define meaning, in relation to the processing of linguistic textual, spoken or multimodal utterances, as the way in which the information state of an interpreting system is meant to be affected by understanding the utterance. This definition covers both the meaning of utterances in interactive discourse and that of textual 'utterances' in documents that are meant for non-interactive consumption. It includes aspects of meaning that are sometimes regarded as 'pragmatic' rather than semantic, as they are concerned with the meanings of *utterances* rather than *sentences*, i.e. with the meaning of linguistic or multimodal expressions as used in a particular context.

The LIRICS project does not aim at contributing to the development of a standard *format* for the annotation and representation of semantic content, but at providing well-defined *descriptive concepts*. In particular, the aim is to build an on-line registry of definitions of such concepts, called '*data categories*'. These semantic data categories are abstract concepts, whose use is not restricted to any particular format or representation language.

Annotation and representation have related, though different aims. A semantic representation of a natural language utterance is assumed to be a formal description of the exact meaning of the utterance in a formal language that itself has a well-defined semantics. Annotations, in contrast, are intended to capture certain properties of the material that is annotated, depending upon the purposes for which the annotation is designed. For instance, temporal annotations of linguistic corpus material may be intended for the purpose of a particular linguistic analysis, where the interest is to find all expressions in the corpus related to time as rapidly as possible. For such a purpose, the annotations would likely take the form of tagging linguistic expressions with grammatical time- and tense-related descriptors. If, by contrast, a corpus of texts in a certain language is annotated in order to facilitate the determination of the temporal ontology that underlies the grammatical distinctions made in the language, then the markup may be expected to make use of 'onto-temporal' rather than linguistically motivated descriptors[1]. If the purpose of the annotation is to support applications like information retrieval, information extraction and question answering, then the temporal annotation should allow temporal reasoning, in particular for finding the textual elements that provide information which is relevant to a given query that refers to a particular date or period (cf. Pustejovsky et al. 2003). The markup used for temporal annotation should in this case have a well-defined semantics.

Annotations are often not conceived as expressions in a formal language, but as consisting of simple tags or lists of tags. Semantic annotation is likewise normally understood as intended to indicate certain aspects of the meaning of the utterances that are marked up, rather than representing their full meanings. Interestingly, though, the trend in computational semantics in recent years has been to focus on the construction of underspecified semantic representations, rather than fully-fledged meaning representations (see e.g. Bunt and Muskens, 2001). Underspecified semantic representations are formal but incomplete representations of meaning, which may leave a variety of aspects of the meaning of a given utterance unspecified. In fact, semantic annotations and fully-fledged semantic representations may be seen as the two extremes of a continuous scale, where underspecified semantic representations move in the direction of annotations as they become more underspecified, and where annotations move in the direction of semantic representations as they capture more information. As the exhaustive, precise formal representation of natural language meanings may be a utopia anyway, we believe that the perspective of semantic annotation systems that develop in the direction of semantic representations is a promising one, to which the LIRICS project may well be able to contribute.

3 Metamodels

3.1 Models and metamodels

Alternative approaches to the marking up of linguistic resources differ most importantly in the categories of information that they aim to capture. The choices that are made in this respect can be represented by specifying the classes of objects and relations that are covered by their markup tags. Such a characterization is called a *model.* Looking for commonalities in alternative approaches therefore implies comparing their underlying models. It is often possible to capture what alternative models have in common by moving to a more abstract level than that of each of the models, and this is where *metamodels* come in. Metamodels are well-known from software engineering, where they usually take a graphical form, like a UML diagram, and are defined as "an explicit model of the constructs and rules needed to build specific models within a domain of interest" (http://www.metamodel.com/), or more loosely simply as a model that describes a set of models. Bunt and Romary (2004) have proposed a more formal interpretation of the term metamodel by relating it to the notion of model as used in model-theoretic semantics. It is felt that this may be helpful as a methodological tool for the definition of concepts for semantic annotation and representation, and, as a consequence, for the isolation and definition of corresponding semantic data categories of importance.

In model-theoretic semantics, a model serves as the basis for providing the formal semantics of a semantic representation language. It does two things: (1) it specifies sets of entities from which denotations can be constructed for the terms of a representation language; and (2) it

[1] Thanks to Kiyong Lee for this example.

assigns specific denotations to the descriptive terms of the language. For instance, a model for the language of standard first-order predicate calculus, PL_1, has the form $M_{PL1} = <D, F>$ where D is a set of individuals and F is a function assigning values to the descriptive terms of PL_1. (To each individual constant it assigns as denotation an element of the set D of individuals; to each k-ary predicate constant it assigns a set of k-tuples of individuals; to each individual function it assigns a set of pairs of individuals.)

A metamodel, by contrast, does not assign denotations to descriptive terms, but is an abstract specification of the *sorts* of semantic entities that are assigned to the terms of the language. A metamodel does not specify any semantic entities, but only *types of semantic entities*. For example, in PL_1 we have three kinds of descriptive terms: individual constants (*id*)s, predicates (*pd*)s, and functions (*fn*)s from individuals to individuals (such as *mother_of*). A metamodel for PL_1, should thus say that: (1) individual constants (*id*)s correspond to individuals; (2) k-ary predicate descriptors (*pd*)s correspond to sets of k-tuples of individuals; and (3) descriptors of individual functions (*fn*)s correspond to sets of pairs of individuals (argument-value pairs). This can be formulated mathematically by defining a *first-order metamodel* as a pair:

$MM_{PL1} = <\text{ind}, \{(id \rightarrow \text{ind}), (pd^{(n)} \rightarrow S_n[\text{ind}]), fn \rightarrow [\text{ind} \rightarrow \text{ind}]\}>$

where the first element, ind, is the type of individuals; the second element lists the assignment of types of denotations to kinds of descriptive terms, using $[A \rightarrow B]$ to denote the type of functions from A to B, and $S_k[t]$ to denote the type of sets of *k*-tuples of elements of type *t* (with $0 \leq k$).

Similarly, a metamodel for second-order predicate logic, which also supports second-order predicates ('*ppd*s', i.e. predicates that apply to first-order predicates), has the following structure:

$MM_{PL2} = <\text{ind}, \{(id \rightarrow \text{ind}), (pd^{(k)} \rightarrow S_k[\text{ind}]), (ppd^{(k)} \rightarrow S_k[S_*[\text{ind}]])\}>$

for $k, n \geq 1$, where $S_*[D]$ is the set of n-tuples of elements of D for any $n \geq 1$.

3.2 Metamodel contents

Above, we have introduced an operational definition of meaning by considering the situation in which a reader or listener processes linguistic (or multimodal) input. The definition of meaning as the way in which an interpreting system's state is intended to be affected by understanding the input, may seem to have a focus on interactive language use, but does apply equally well to situations where a reader (human or computer) is processing text in a non-interactive manner. In that situation too, there is a source who has produced the text; the text was produced at a certain time and is processed at a certain time; the text was intended for a certain audience, which at the time of reading is instantiated by the reader, and there may be an additional, non-intended audience, as in the case of a children's book read by adults (or the other way around). In all cases, it usually makes sense to consider the meanings of parts of the linguistic material, down to a certain level – usually the smallest meaningful unit for semantic analysis is thought to be that of morphemes. Larger units are usually even more important to consider; in particular the units that we have called 'utterances'. A precise definition of 'utterance' is hard to give (see e.g. Larsson 1998), but in spoken interaction, an utterance is usually taken to be the part of a 'turn' that has a separate function or purpose, where a 'turn' in turn is defined as the linguistic material contributed by a single speaker and bounded by pauses larger than a certain threshold. When considering written text it is unusual to speak of an utterance, and the term 'sentence' is used instead.

This view of meaning has immediate implications for the basic ingredients that should be available for the description of semantic information. When linguistic or multimodal input occurs, there is, first of all, an event: something that happens at some point in time (or during a certain time interval). Second, it has at least two participants, who have different roles; those of sender and addressee. So we minimally want to distinguish (1) events; (2) temporal objects and relations for anchoring events in time; (3) entities participating in events; and (4) semantic roles relating events to participants. This corresponds to the metamodel depicted in Figure 1.

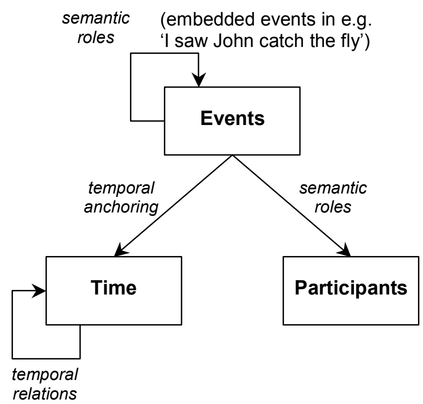

Figure 1 General metamodel of events, temporal objects, participants and semantic roles.

This metamodel contains semantic roles not only for relating events to participants, but also for relating events to other events. This is because events are needed not only for describing communicative events, but also for the semantic analysis of natural language utterances and sentences, where 'roles' correspond to semantic relations between a verb and its arguments. Such events may have other events as 'participants', as in the sentence 'I saw John catch the fly', where a see event has an embedded catch event. To avoid overlapping between the types of entities in different boxes, we intend the box 'participants' in Figure 1 to be 'participants which are themselves not events', and as a consequence we get semantic roles as relations in two places.

An example of a data category, as it may figure in the on-line registry built in the project, that fits in with this metamodel, is shown in simplified form in Figure 2.

/Addressee/
Definition: An entity that is an intended recipient of a communicative event, whose state of information is intended to be influenced by the event.
Source: (implicit) an event, whose eventType should be ***/CommunicativeAct/***.
Target: a participant (user or system).

Figure 2 Sample data category for *Addressee*

The establishment of a metamodel to support the design of an annotation scheme may be useful since it specifies the types of objects and relations that the annotation language should be able to represent. For the purposes of the LIRICS project, this means that the metamodel determines the classes of concepts (objects and relations) for which data categories should be supplied. Note that data categories in this view may relate to the entity types at the 'nodes' (the boxes) of the metamodel, as graphically represented, as well as to the 'edges' relating the nodes. (This is in contrast with some other uses of metamodels, where data categories relate only to nodes. The latter view forces the metamodel designer to conceptually reify every relation that can be relevant in annotation, which we believe makes metamodels unnecessarily unwieldy.)

While a metamodel is clearly not a method of mapping one annotation scheme onto another, it is highly relevant to such a task. Annotation schemes with underlying models that cannot be covered by one and the same metamodel cannot be fully mapped onto one another, and the metamodel will indicate where such a mapping may be possible and where the difficulties are. We will see this illustrated below in Section 4.2, when we try to combine two alternative models underlying temporal annotation schemes into a single metamodel.

4 Types of semantic information and related preliminary metamodels

The LIRICS project will cover at least the following specific areas of semantic interest: dialogue acts, temporal entities and relations, reference annotation and semantic roles. There are very clear motivations for considering these areas in particular. Firstly, they largely coincide with similar areas of interest in ISO. This is of some significance, as one of the stated goals of the project is to have proposed a common and stable set of data categories in the form defined by ISO standard 12620 (Terminology and other language resources – Data categories for electronic lexical resources), in order to enable the interoperability and reuse of multilingual language resources, digital content and language engineering software.

Secondly, each of the areas of interest under discussion have achieved a certain level of maturity in the semantics (and pragmatics) research communities; while there may be disagreements about details of representation and structure, at a grass roots level there is a perceived broader consensus about the types of information that need to be presented.

4.1 Dialogue acts

One area of language and speech technology that could benefit greatly from internationally agreed annotation standards is that of dialogue modelling. Annotated dialogue corpora are important sources of information for the design of spoken dialogue systems, for the development of interactive text-based multimodal human-computer interfaces and for the creation of systems that mediate in a useful way in human-human dialogue. Dialogue acts have become popular for annotating dialogues in order to indicate what the participants are doing (see e.g. Jurafsky & Martin 2000, Chapter 19).

The term ‘dialogue act’ is sometimes used rather informally, in the sense of ‘speech act used in dialogue’. Accordingly, a dialogue act has a certain function or purpose, corresponding to the ‘illocutionary force’ of speech act theory, and a propositional or referential content (‘propositional content’ in speech act theory). A dialogue act may also be said to have ‘locutionary’ and ‘perlocutionary’ aspects, although these are often not considered, because the locutionary act is constitutive of the dialogue act itself, while the perlocutionary act does not lend itself to systematic formalisation and identification. In order to be optimally useful for dialogue analysis purposes, however, a more precise and self-contained notion of dialogue act would be necessary, focusing on its role in assigning meanings to utterances in dialogue. In line with both the definition of meaning given above and with the ‘information-state update’ or ‘context-change’ approach to dialogue which we have taken, we may define a dialogue act as a semantic unit in the description of dialogue utterance meaning, having two main components: a *communicative function* and a *semantic content*. The semantic content is information that the source of the dialogue act is bringing to the addressee’s attention, and the communicative function specifies what the addressee should do with the semantic content, i.e., in what way the addressee should use that information to update his information state (or context model), upon understanding the utterance.

Dialogue acts are thus tied to utterances, which are parts of turns, which are parts of a dialogue. Mostly, a dialogue act is realized by a single utterance, but it may be that the dialogue act is expressed by two utterances together, for instance when a speaker is interrupted by some dialogue-external circumstance. It may also happen that a speaker is unable to complete an utterance that he intends to make in order to express a certain dialogue act, and that the addressee helps the speaker; in that case, we believe it is still best to consider only the first participant as playing the sender role, and the second as playing the addressee role. Similarly if an addressee provides backchannel acts to give positive feedback while the speaker is speaking. In sum, a dialogue act is in general related via an utterance to a turn and thus to a single participant in the sender role and another participant in the addressee role.

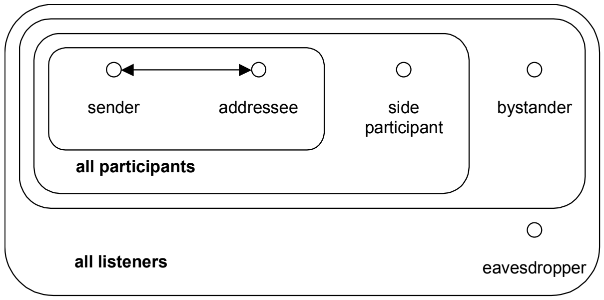

Figure 3 Participant relationship space.

In the case of multiparty dialogue, there may be multiple addressees. Also, there may be additional participants who witness the dialogue without belonging to the intended audience (but both sender and addressee(s) may be aware of their presence, and take that into account); their role might be called that of 'overhearer'. The category of overhearer corresponds to the ratified participants in a conversation, such as the side participant (who may take part in the conversation without intrusion) and the bystander, of which the sender and addressee are aware, and the non-ratified participants, the eavesdroppers, of which they are not. This relationship space is shown in Figure 3 (from Clark 1996: 14).

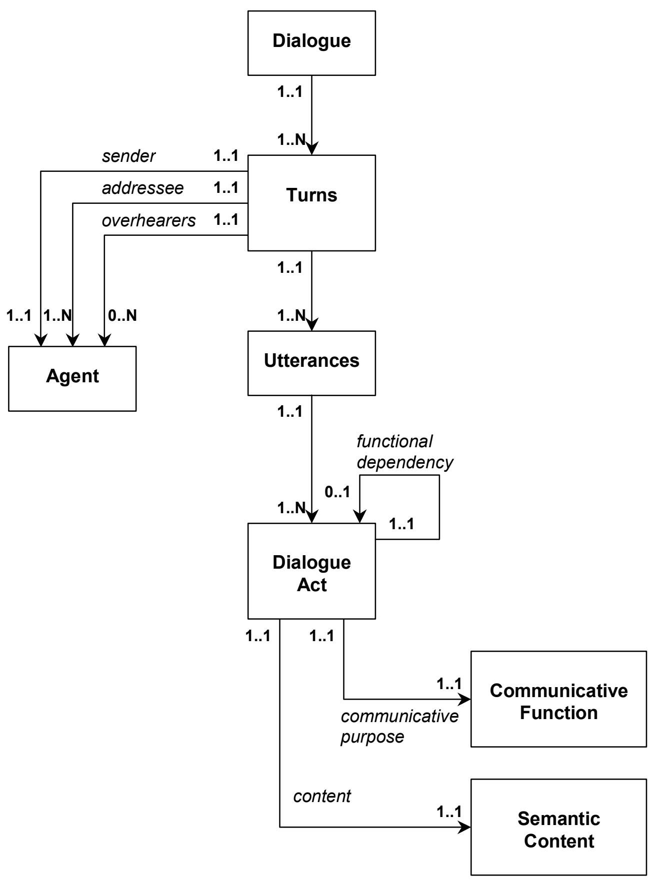

Figure 4 Preliminary metamodel for dialogue acts.

The metamodel shown in Figure 4 captures these considerations, with the additional inclusion of possible functional dependencies between dialogue acts, which encompass such things as indicating to which question a response is intended to be an answer, etc. Note also, and most importantly, that an utterance may correspond to multiple dialogue acts, due to the multidimensional nature of communication and the multifunctionality of natural language utterances (see Allwood, 2000; Bunt, 2000; Bunt & Girard, 2005).

The metamodel for dialogue acts is not anticipated to cause major controversy. The differences will come with the attempt to define a circumscribed and rigorous set of data categories built from the foundation of the metamodel here presented.

Dialogue acts are themselves a type of event, which take place anchored temporally. Time and order are important for the identification of some acts; for example, it is impossible to agree with someone unless that someone has previously expressed a commitment to the content of the agreement. This is covered at the moment by the functional dependency link in Figure 4, but this may not be enough to capture fully the temporal aspects of dialogue acts.

4.2 Temporal entities and relations

There are currently two major developing standards for the annotation of temporal entities and relations: OWL-Time (which used to be called 'DAML-Time') and TimeML.

4.2.1 OWL-Time

OWL-Time (Hobbs & Pan, 2004) is an ontology (one of 282 developed during the DARPA Agent Markup Language project) that provides and defines a structure for the logical relationships between different temporal expressions, with the aim of marking up textual elements (primarily on the web) for the rapid extraction of 'surface-level' temporal information. It was developed as "an ontology of temporal concepts, for describing the temporal content of Web pages and the temporal properties of Web services" (Hobbs & Pan, 2004:1); in other words, mainly with the idea of reasoning about temporal events in mind, not for the annotation of natural language texts. As a consequence, only the static, topological qualities of temporal entities are captured. So, for example, with the application of some kind of reasoning engine to information marked up with OWL-Time, one might be able to detect whether scheduling a meeting for 2:00pm PST for 45 minutes would clash with a teleconference set for 6:00pm EST on the same day, but one could not reason about what happened every day last month and the kinds of facts or events that held true at that time. The idea was to create temporal constructs to enable web service providers to describe the semantic time properties of their services. Thus the developers of OWL-Time were principally concerned with the representation of such information as, for example, the opening times of doctor's surgeries, the times of meetings across different time zones (as in the previous example), the times/durations of theatre performances, etc.

An attempt to visualise the basic model for OWL-Time is shown in Figure 5. The 'Event' element shown to the right of the dashed and dotted line is not in fact explicitly specified by Hobbs & Pan (2004), but is presumably the kind of 'element' that can take place in either an instant, an interval, or (if interrupted) a sequence of intervals. So, concerning the information to be extracted in the examples given earlier, the 'Event' might be equivalent to being seen by the doctor, or the meeting itself, or the play, and so on.

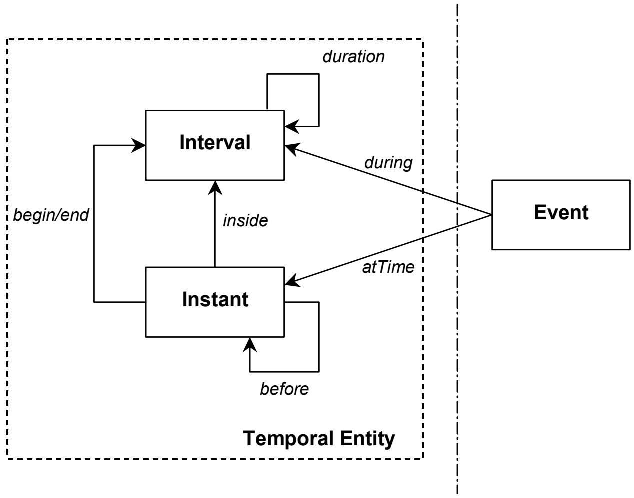

Figure 5 Graphic representation of OWL-Time ontology.

Hobbs and Pan (2004) admit that in the future, simply marking up items in this way will be insufficient for retrieving all relevant content expressed in natural language form from the web (which is the main intended application of these developing technologies).

OWL-Time does not provide an event ontology; Hobbs & Pan (2004) explicitly state that they believe that any event ontology should be kept separate from a time ontology. They make this their reason for not providing descriptions of event elements themselves. However, in Pan & Hobbs (2004), there is an attempt to link a temporal Instant with an Instant Event (a Punctual Event in the phraseology of TimeML), and a temporal Interval with an Interval Event (an Extended Event in TimeML) – this is shown in Figure 6.

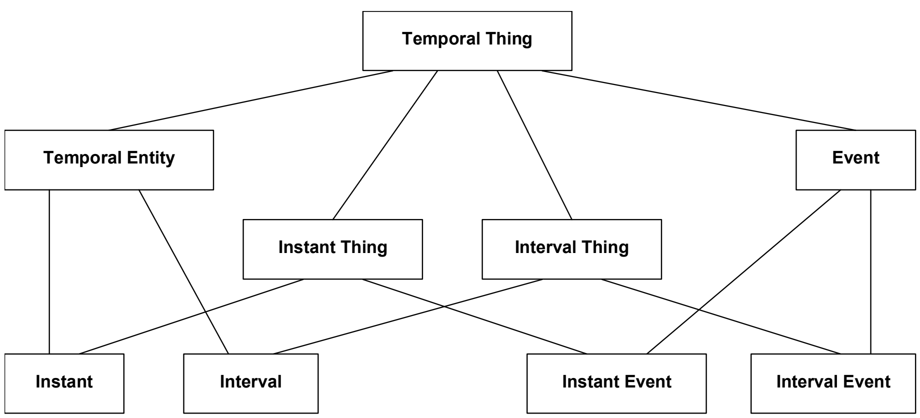

Figure 6 Subclass hierarchy of temporal concepts (from Pan & Hobbs, 2004: 2).

This would seem to indicate that the developers of OWL-Time do indeed link events inextricably to temporal entities, even including both as children of a generic 'Temporal Thing' (in Figure 6). How these two types of ontologies might be combined (i.e. by merging OWL-Time with TimeML) is explored in Hobbs & Pustejovsky (2003).

One conceptual problem with OWL-Time is its treatment of durations, which are viewed as intervals. For example, a duration of 1 week is modelled in OWL-Time as the concatenation of seven intervals of 1 day. This is conceptually confusing, however, since concatenation is an operation that applies to intervals, not to their lengths. A duration is the *length* of an interval, measured with the help of a certain unit; the relation between a duration of 1 week and one of 7 days is not one of concatenation but one of equivalence, due to the possible conversion from weeks to days. More generally, a duration is an amount of time, and just like amounts in other dimensions (amount of weight, volume, velocity, etc.), it is formally an equivalence class defined through the conversions between units of the same dimension (see Bunt, 1985, Chapter 6, for general discussion, and see Figure 8 for an attempt to amalgamate this concept in an overarching metamodel for time).

To sum up, there are a number of reasons why OWL-Time on its own would not be a suitable base for the development of a standard for the linguistic annotation of temporal elements:

- OWL-Time was not designed with the full annotation of natural language in mind.
- It does not deal with deictic time, temporal aggregates (although see Pan & Hobbs (2005) and Pan (2005) for how this is being incorporated), or vagueness.
- There is no satisfactory treatment of durations.
- It has not been fully tested yet using a real-world application domain with a temporal reasoner.
- OWL-Time does not indicate the role of tense in the extraction of information from the web in any way, which would seem of crucial importance when trying to reason about whether an event has already occurred or whether it will occur at some point in the future.

Some of these criticisms and shortcomings of OWL-Time are addressed by the temporal annotation scheme proposed by the developers of TimeML.

4.2.2 TimeML

TimeML provides a rather extensive scheme for temporal annotation of natural language texts. The annotation scheme not only marks up temporal expressions, but also provides links describing relations between these and the events being described.

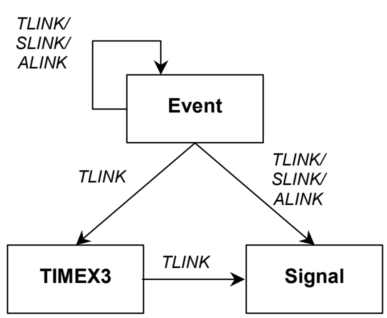

Figure 7 Graphic representation of TimeML annotation scheme.

Although TimeML does represent the temporal structures and relationships in natural language in a more complete way than the OWL-Time ontology, there is little indication of how the information marked up might be extracted and reasoned with. The TimeML tags essentially do not have a semantics, which would be needed for temporal reasoning. Hobbs & Pustejovsky (2003) suggest that the OWL-Time ontology can be used to give a semantics to TimeML annotations, and illustrate this with a number of examples. However, for certain cases that they do not discuss, it seems problematic to give a semantics to TimeML markups in OWL-Time terms. This is for instance the case for tense information, which Hobbs & Pustejovsky mention as temporarily left out of consideration. While the two schemes are roughly complementary, there are some further difficulties to be faced when attempting to combine the two, as we shall see below.

Generally speaking, TimeML's strengths are where OWL-Time has weaknesses, and vice versa. Some of the limitations of TimeML are as follows:

- TimeML tags in general do not have a semantics. This means that TimeML annotations are not a reliable basis for temporal reasoning (although it may support some shallow forms of reasoning).
- TimeML has some redundancy in allowing the temporal relations between events to be indicated both directly (using TLINK relations between events) and indirectly (via TLINK relations to their temporal anchoring).
- TimeML has no satisfactory treatment of durations. For instance, a duration of 30 minutes (when mentioned as such in a text) is annotated as <TIMEX3 (...) type="DURATION" value="PT30M" (...)>30 minutes</TIMEX3>. This makes it next to impossible to reason that this duration was half an hour.
- TimeML has no provision for time zones and daylight saving conventions, which OWL-Time has. This makes it difficult to produce annotations that denote exact times.
- The 'SIGNAL' tag in TimeML seems a kind of wastebasket, as it is used to annotate a wide variety of semantically very different entities, such as temporal prepositions, temporal modifiers ("twice", "every"), negatives ("not", "never"), modals ("might", "may"), and still others.

4.2.3 Combining metamodels for temporal information, time ontologies, and events

From these two emerging contenders for the semantic representation of time in text, we have extrapolated a preliminary metamodel of the kinds of entities and relations that would be necessary in a generic model of temporal information (shown in Figure 8).

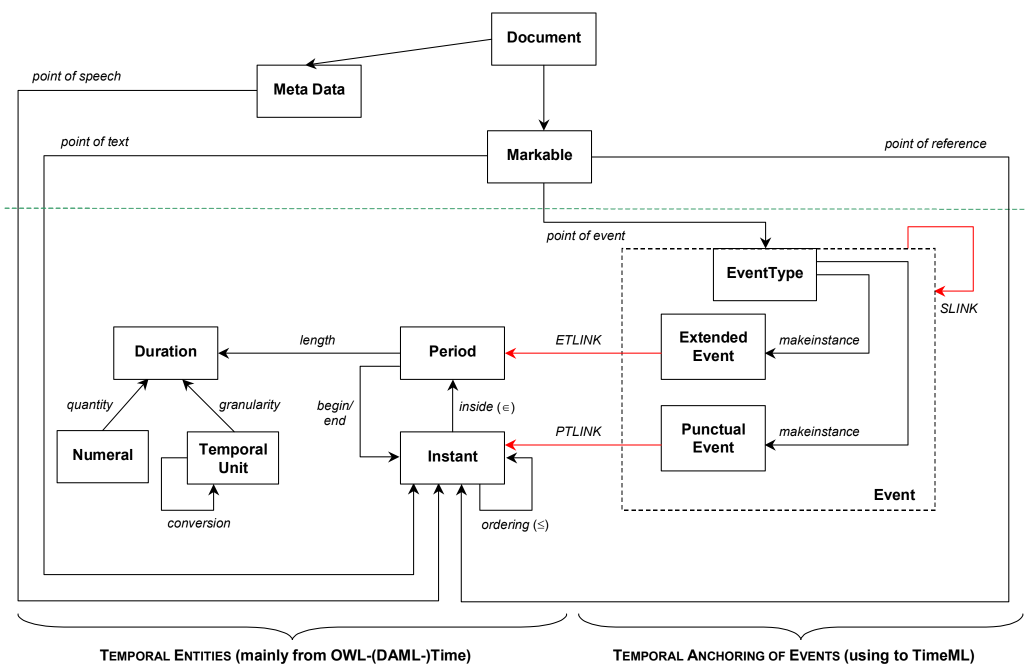

Figure 8 Preliminary metamodel for temporal information, inspired by the OWL-(DAML-)Time and TimeML schemes.

One important respect in which the underlying models of TimeML and OWL-Time are not compatible is in how they view interrupted events as being anchored in time. OWL-Time allows an extended event to be interrupted, and thus to occur during a sequence of temporal intervals; TimeML, by contrast, considers the various parts of an interrupted event to be separate events, which are called ‘event instances’, and which are related by being instance of the same ‘event’ through the application of the function ‘makeinstance’. In the metamodel of Figure 8, we have opted for following the TimeML model, but we think this should in fact be open for discussion. For clarity, we have in the metamodel of Figure 8 used ‘event’ instead of ‘event instance’, and ‘EventType’ instead of TimeML’s ’event’.

Another point that deserves attention when trying to combine the OWL-Time and TimeML models regards the relations between temporally related events. TimeML distinguishes three kinds of relations: SLINKs (subordination relations), ALINKs (aspectual relations) and TLINKs (temporal relations). Two events are for instance temporally related in cases like “He smiled while he looked at the picture”, relating a smile event and a look-at event. Allowing such direct temporal relations between events creates a certain amount of redundancy in the model, since alternatively it would be possible to link the smile event to a temporal interval that is included in the interval during which the look-at event occurred. Similarly this is so for aspectual links. Once an integrated model of events and temporal entities is in place, as the result of merging the underlying models of OWL-Time and TimeML, it seems best to represent all temporal relations between events through their anchoring in time; this is reflected in the metamodel of Figure 8. Only subordination relations between events remain, as they are fundamentally not just temporal relations, but rather semantic relations from which temporal relations may be inferred. As a consequence, some of the temporal relations that in TimeML would be covered by TLINKs or ALINKs are covered in this metamodel through the ETLINK (‘Extended Temporal Link’) and PTLINK (‘Punctual Temporal Link’) relations that anchor events in time.

Besides dealing with these aspects of relating TimeML and OWL-Time, we have also amended OWL-Time to take care of its confusing treatment of durations, and added durations as lengths of intervals, defined by the combination of a numeral and a unit (which is related to other units through a table of conversion). We have also renamed the ‘interval’ as a ‘period’ to avoid terminological confusion with musical and mathematical intervals.

Note that in Figure 8 there are a number of links labelled ‘point of x’: point of speech, point of event, point of reference and point of text. These are necessary in order to be able to interpret tense information in natural language. The first three (speech, event and reference) are those that are traditionally used to place an event in time (see Reichenbach 1947; Lyons 1968). The last (text) is rather subtler to pin down. It is the point of the text of the speech. So for example, when a person is telling a story, it is not enough to know the point of the speech itself (the document creation time), but the point at which the speech in the story is taking place. In fact, knowing the point of text is necessary for the temporal placing of any reported speech, e.g., “Last Friday the Home Secretary said he would be taking the matter further”. In this example, the point of speech is the time at which this sentence is uttered or written (needed to anchor the phrase ‘last Friday’), the point of text and the point of reference is ‘last Friday’, and the point of event is at some time in the future from the point of reference.

Finally, in order to encompass the problem of anchoring an instant to a concrete point in time, it may be necessary to identify a particular instant by means of a timestamp. This pinpoints any instant by relating it to the successively nested periods in which it is found and is conventionally used to define a Calendar Clock Time within a Calendar Date; these together comprise the Timestamp, as shown in Figure 9.

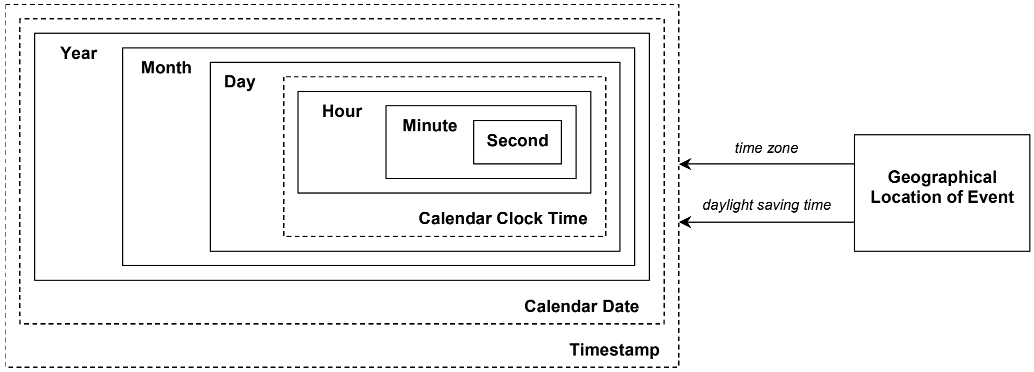

Figure 9 The composition of a Timestamp, which gives the temporal 'address' of an Instant (composed of the relative position of the Instant within a predefined Period or Periods).

The OWL-Time scheme provides for the definition of a timestamp in a similar way, although the time zone information is included as part of the timestamp. However, this does not really capture the true status of the time zone information. In the model we propose shown in Figure 9, the Timestamp always denotes local time, which is itself relative to the Geographical (spatial) Location of the Event Instant. So, the timestamp is always interpretable with respect to the location of the event, but this itself is not a part of the *temporal* model of information rather than of the *spatial*. The same observation applies to the adjustment of the timestamp for the purposes of daylight saving. The timestamp will inherit this information from knowledge of the geographic location of the event, and comparisons between different timestamps will require the ability to convert a timestamp of one location into the timestamp of another.

How all these points will be taken into account in the development of data categories in the future work of the LIRICS project is a matter for further discussion.

4.3 Reference annotation

The annotation of coreference relations (also known as reference and anaphora annotation), consists basically in (a) the identification of the textual elements that may be considered as referring expressions in a given discourse, and (b) the kind of relations that link these markables as far as their reference is concerned. This has already been the subject of considerable practical and theoretical research in order to establish basic principles and to unify different approaches for coherent and consistent annotation (see for example, amongst others, Hirschman & Chinchor 1997, Davies and Poesio 2000, van Deemter & Kibble 2000, Vieira et al. 2003, Salmon-Alt & Romary 2004a and Poesio 2004, amongst others). The maturity of the field of coreference annotation should certainly allow for the proposal of more concrete standards.

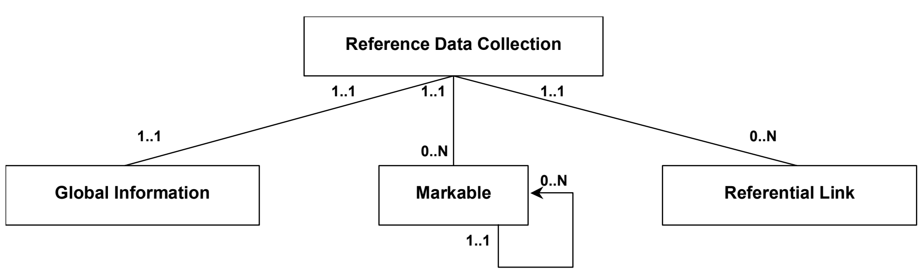

Figure 10 Previous coreference 'metamodel'.

Figure 10 shows an earlier version (from Salmon-Alt & Romary 2004b) of the suggested metamodel for coreference. The Global Information in this metamodel is for capturing metadata associated with the resource element containing the annotated material. A problem with this representation is that it does not really abstract away enough from the annotation of coreference, with each of the boxes labelled Markable and Referential Link intended to be coded as separate tags. This however does not give a good idea of the dependency relations between these concepts, which is what is supposed to be shown at the metamodel level, without direct recourse to the actual entities used in annotation. We have therefore amended the metamodel to reflect this consideration in Figure 11.

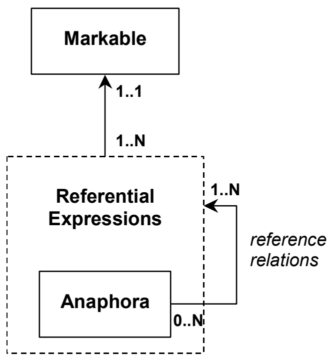

Figure 11 Proposed coreference metamodel.

What Figure 11 shows, is that the source text contains referential expressions (which are ‘referred to’ by the concept labelled Referential Expressions), and that these referential expressions themselves may in turn have relations defined between them; it is these relations that identify the coreference between different referring expressions.

In the design of data categories for coreference to follow, it will be the types of referential expression and the types of relations that hold between them, and how they may be instantiated that will be of concern to the LIRICS project to define and exemplify in the definition of data categories. These types of referential expression and relation may also have considerable overlap with other levels of linguistic description, both within the level of semantics itself as well as with other levels such as morpho-syntax and syntax. How such overlaps might be taken into consideration and be taken advantage of is the subject for future discussion.

4.4 Semantic roles

There have been intensive efforts to produce syntactically annotated text (now very much a mature area of linguistic annotation); however syntactic annotation by itself is not enough to represent the meaning of the text. For example (taken from Palmer et al., 2005: 71):

(1) John broke the window.

(2) The window broke.

Syntax alone would tell us that ‘the window’ is the direct object of the verb in (1) and the subject in (2), but fails to represent that in both these cases, it is the thing that is broken, i.e. that even though it plays a different syntactic role, it plays the same semantic one. Recently there has been an increasing trend towards the annotation of the semantic role played by constituents of text to improve the recall and precision of searches and for rapid information extraction.

A promising approach to this representational problem is that based on frame semantics (Fillmore 1976) and the classes of verb defined by Levin (1993). In terms of annotation, there have been two major projects to annotate semantic roles utilising frame structures (primarily of verbs): FrameNet and PropBank. Both are built around definitions in VerbNet and WordNet, but are designed with different kinds of priorities in mind. FrameNet starts from the text itself with no predetermined structure, whereas PropBank uses the pre-parsed data from the Penn Treebank. This has the side effect that in PropBank, every node in the parse tree is labelled with a semantic role of some sort, whereas in FrameNet, this is not the case. FrameNet is more concerned with the semantic roles of the constituents of a frame within a situation, and with how those frames are related to each other hierarchically. For the developers of PropBank, the main concern is the predicate-argument structures of syntactic constituents. PropBank's framesets are hinged on the verb; whereas the roles are fixed in FrameNet, in PropBank these are determined with respect to the verb (see Figure 12 for an example).

PropBank		**FrameNet**
buy	*sell*	*COMMERCE*
Arg0: buyer	Arg0: seller	Buyer
Arg1: thing bought	Arg1: thing sold	Seller
Arg2: seller	Arg2: buyer	Payment
Arg3: price paid	Arg3: price paid	Goods
Arg4: benefactive	Arg4: benefactive	Rate/Unit

Figure 12 Comparison of example frames in PropBank and FrameNet (from Palmer et al. 2005: 88)

We can see from Figure 12 that there is a good correspondence between the types of roles defined for these related frames; it is mainly the determination of the order and structure that differs.

This is a very brief overview of the two currently most important and influential schemes in existence for the annotation of semantic frames and roles. At this stage, when we are principally considering the methodological aspects of approaching the problem of semantic role annotation, we need not go into further detail about the choice of semantic role labels that would be required for the provision of a comprehensive set of data categories for semantic roles. We note that, although FrameNet and PropBank approach semantic roles from different perspectives, these differences are largely cosmetic and there is a great overlap between these schemes. This is borne out by the fact that we can define a metamodel fairly simply that will encompass both (see Figure 1).

Future work in the area of semantic roles within the LIRICS project will focus upon deciding what would be an appropriate set of role data categories by looking carefully at the roles defined in the FrameNet and PropBank schemes to see where there is a significant overlap and where a difference. Decisions will have to be made concerning the granularity of the data categories; for example, whether to concentrate on the types of roles (FrameNet) or the part that the roles play with respect to the verb (PropBank). It would not be impossible to develop some form of hierarchical structure that might encompass both of these considerations. An attempt to demonstrate this is shown in Figure 13.

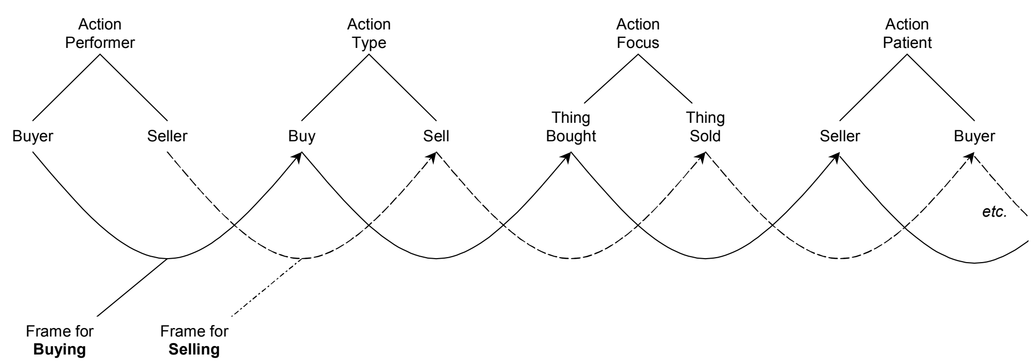

Figure 13 An abstracted model for the verb frames of *buying* and *selling*.

Whether it is possible to abstract so far away from the event type has yet to be determined. Perhaps the structure shown in Figure 13 will only fit the events of buying and selling because they are in effect the same transaction taken from differing points of view (the buyer's and the seller's). This is essentially what is done in the FrameNet scheme by treating 'buying' and 'selling' as instances of the frame of commerce, which has set role types. FrameNet however has nothing at all to say about the relationship between the semantic roles defined and the type of commerce (buying or selling), nor about how this will affect the order of these roles within the text.

The above example highlights a problem with the frame representation of semantics, namely that of granularity and of inheritance of characteristics. There is little point in defining frames that uniquely correspond with every verb and noun, because this would be expensive to search. We would ideally like to be able to infer certain patterns from the frames that are defined. So, in short, while a great deal of the work in the definition of semantic roles and descriptions has been carried out using frames as a basic structure, the status of frames themselves from a semantic point of view is not yet clear and is a subject for further discussion.

Just as we discussed in Section 4.3 for reference annotation, in the design of the data categories for semantic roles to follow, it will be the *types* of roles and the *types* of relations that hold between them that will be of most importance to choose and define. Again, how these are then instantiated and utilised in a specific annotation scheme is not of primary concern to the LIRICS project. These types of semantic roles and relations may also have considerable overlap with other levels of linguistic description, both within the level of semantics itself (temporal information for instance) as well as with other levels, especially that of syntax. How such overlaps might be taken into consideration and taken advantage of is the subject for future discussion.

There is also hierarchical and inheritance of information that is not made explicit in the metamodel of semantic roles: namely that of how events are related to other events.

This is an issue not only for semantic roles, but also for other types of semantic content information. For example, when defining a dialogue act, each dialogue act has exactly one communicative function, but its specificity will inherit much from its more general type. This information is not reflected in the metamodel.

4.5 Other potential areas of interest

There are several other areas of semantic information that are of potential interest to consider. First, an obviously important area for a wide range of applications is that of word sense disambiguation. However, the annotation of a text with indications of intended senses for ambiguous words (and multiword expressions) essentially comes down to associating occurrences of words with pointers into lexical resources, which as such does not seem to warrant the investment of research effort in the LIRICS project. What might be relevant for the LIRICS project is the annotation of textual elements with *constraints* on applicable word senses that might be derivable from context. On a related note, it may also be interesting to investigate how semantic ambiguity, which is not necessarily resolvable simply by reference to the context, could be represented by linking the source to more than one 'frame' of semantic content; but again this is not directly relevant to the LIRICS project. These are issues for future discussion.

Second, the central role that events play in semantics is general, and also in the metamodels considered above, would suggest that it is useful to consider events not just from a temporal perspective, as we have done in the metamodel discussed in Section 4.2. From a temporal point of view, events fall into two categories: those that are seen as occurring instantaneously and those that occur over an extended period of time. Other, more articulate ontologies of event types make finer distinctions, identifying for instance *processes* and *achievements.* Such distinctions are semantically relevant because they support different conclusions that may be drawn from the assertion that an event of such a type has occurred. The integration of a finer ontology of states and events into the metamodels considered above seems relatively straightforward. Also, events occur not only at a specific time but also at a certain place, so it might seem relevant to also consider spatial information. However, natural languages do not have a spatial equivalent of the rich tense systems that are found, and in general seem to pay considerably less attention to marking the place of an event than to marking the time. We therefore do not give a high priority to the annotation of spatial information. Perhaps an exception should be made for geographical information; as noted above, detailed specification of the time of an event often depends on geographical information, since that determines time zones, daylight savings conventions, etc.

Third, there is a sort of textual analogue to dialogue acts formed by discourse relations. Rhetorical Structure Theory (RST, Mann & Thompson 1988) is the de facto standard in this area, but there are numerous variations and alternatives of the RST proposal. Relevant work for Japanese by Ichikawa (1990) should also be mentioned here. This is possibly another area where the LIRICS project may contribute.

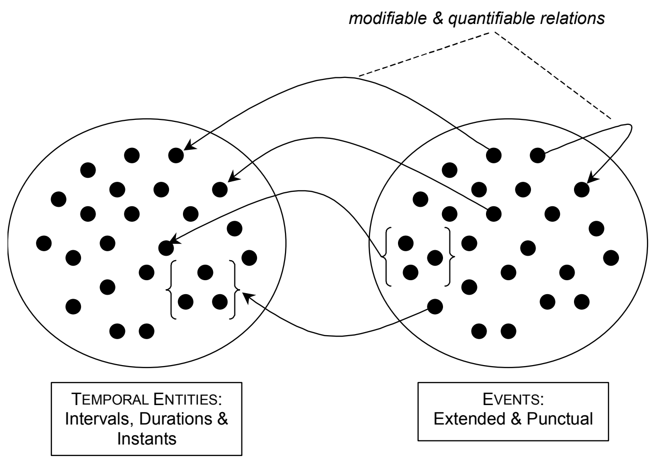

Figure 14 Relationships between Events and Temporal Entities.

Fourth, there is the area of structural semantic relations such as quantification, modification, scope, etc. It was noted above that both OWL-Time and TimeML have difficulties in dealing with temporal quantifiers such as ‘always’ and ‘sometimes’, in the sense that OWL-Time does not contain devices for annotating such expressions, while TimeML annotates them in a way that has no well-defined semantics. Also, quantification (as well as the related notions of modification and scope) is hard to capture in a metamodel, since quantification arises when a predicate, applicable to individual objects of a given type, is combined with a set of those objects, or when a relation that applies between individual objects is used in combination with sets of these objects. This means that every relation in a metamodel may be quantified (as in “I always eat breakfast at eight”). It is therefore impossible to identify quantification with a particular point or part in a metamodel. Figure 14 illustrates this for temporal annotation. The same goes for modification: every relation can in principle be modified (and negated), as in “I nearly had an accident this morning” or “I did not have breakfast today”. We leave it for future discussion whether the LIRICS project should spend some effort on these issues.

4.6 Putting it all together

While the approach that is followed in the LIRICS project, as well as in the related ISO activity, is to start from specific areas of semantic information and work from there (the ‘growing crystal’ approach), the four areas considered above in some detail are conceptually related, and this can be brought out in connections between their metamodels, leading to an all-encompassing metamodel, as shown in Figure 15, which may be a useful basis for a metamodel for semantic information more generally. This metamodel in turn may itself form part of a greater, overarching metamodel, and might not give the complete picture, but is intended to show how different areas of semantic annotation interact and intersect with each other.

First, in Section 3.2 it was noted that the definition of meaning that is adopted in the project goes together with a view on linguistic material as representing a ‘discourse’, which may be a single-author text or a dialogue between two or more participants. Where a dialogue consists of turns, a monologue text consists of units often called ‘discourse segments’, a term which is sometimes applied to dialogue as well (see Grosz & Sidner 1986, for example). Segments again consist of smaller parts, which in the case of interactive discourse are called ‘utterances’, and in the case of monologue discourse ‘sentences’. These are typically interesting units to consider for semantic annotation, and we will therefore call them by the neutral term ‘markable’ (while allowing smaller units to be ‘markable’ as well). These considerations allow the top part of the metamodel for dialogue acts to be generalized to non-interactive discourses, as shown by the upper part of Figure 10. In the case of a monologue text, all the discourse segments have the same source, addressee(s) and potential other audience, so one might want to link that information to a higher level, but this does not invalidate the metamodel as such.

At the level of markables (utterances/textual sentences) we may note that these in general contain referring expressions, and coreference relations are relations between such expressions, as indicated in the integrated metamodel.

If the discourse under consideration is interactive, then the markables can be annotated as expressing a number of dialogue acts, so there we can integrate the rest of the metamodel for dialogue act annotation, including communicative functions and semantic contents. If the discourse is non-interactive, we directly associate a semantic content with a markable. We generally follow the widely accepted Davidsonian approach to semantics, which analyses sentence meanings in terms of states and events, allowing us to integrate the metamodel of Figure 1 relating events, participants and time with the rest of the metamodel, thereby giving a place to semantic roles. This is indicated in the lower left corner of Figure 1, which should be understood as incorporating the entire metamodel for temporal annotation. Note that relations among events and states like cause/effect, which are usually considered to be the basis of discourse relations (see Hasida, 2005) may also get a place in this scheme. We believe that a general metamodel should also allow for cases where the meaning of a sentence/utterance should not be viewed in terms of events, but rather in terms of timeless facts, such as “Grass is green”, “Lions are carnivores”, “Two plus two is four”, etc.; this is indicated in the lower right corner of the metamodel.

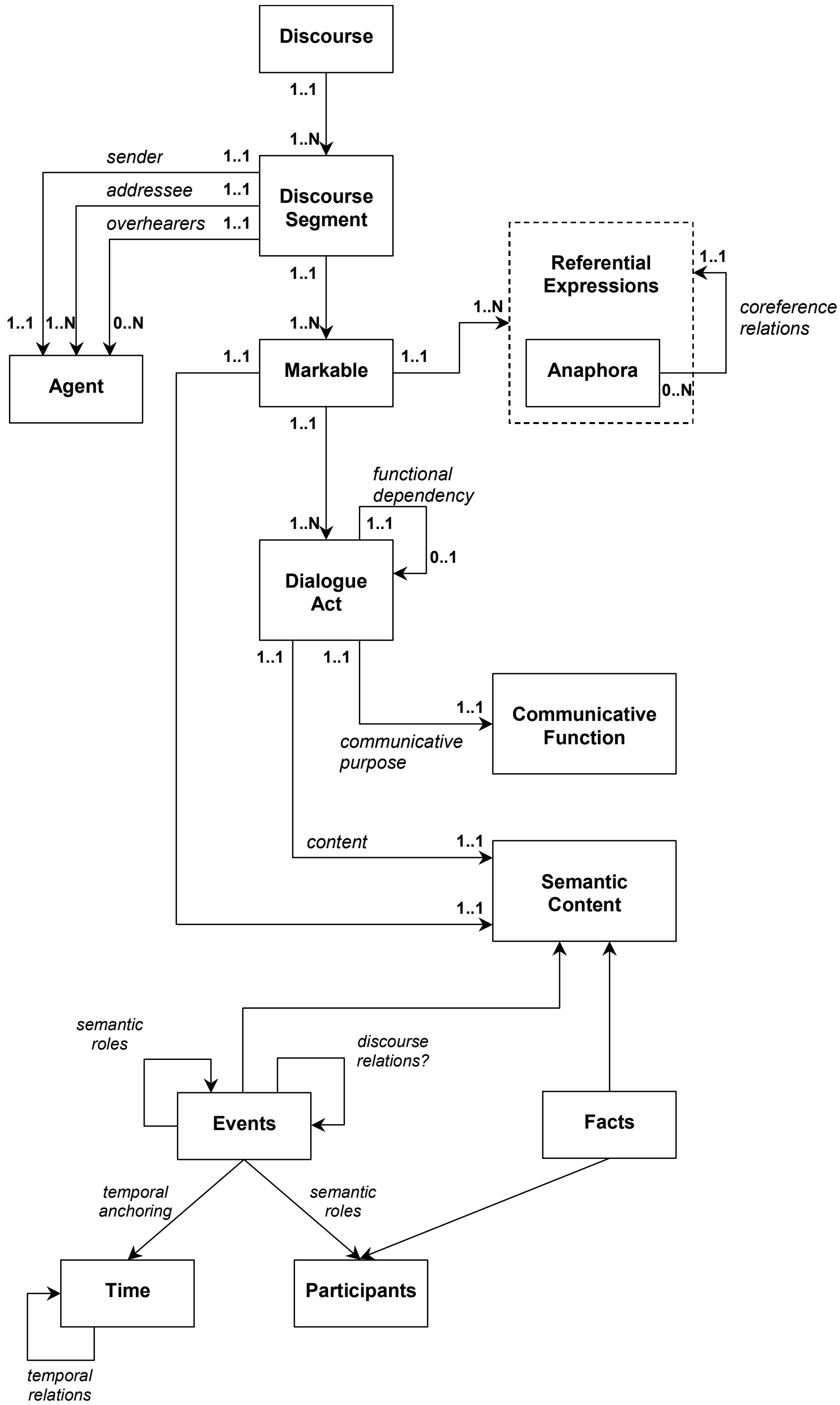

Figure 15 Showing how the different metamodels might connect.

As we have mentioned at the start of this section, this is only a first attempt to unify the individual 'crystal' metamodels for the different areas of semantic content representation to see if it would indeed be possible to 'grow' or 'knit' them into one. It is included here as a springboard for further discussion within the research community and is not intended as a formally fixed or finished metamodel.

5 Conclusion

In this document we have put forward a number of fundamental, methodological considerations for the development of semantic data categories within the scope of the LIRICS project. We have stressed the importance of establishing metamodels as a first step to the design of annotation schemes and the specification of their elements in the form of data categories, and we have discussed potential metamodels for dialogue act annotation, temporal annotation, reference annotation, and semantic role annotation, and how they might fit together in a generic metamodel for semantic annotation. We also considered some potentially relevant other areas of semantic information. All this is intended primarily to spark further discussion with and among experts in the various areas, rather than to suggest definitive metamodels or annotation schemes at this point.

When trying to design an annotation scheme there are a variety of issues and problems to consider, which do not just hold for the annotation of semantic content, but are general to any particular area in linguistics. These issues and problems should be taken into account in future discussions of the development of data categories.

First, in the development of a specific annotation scheme, there is always a choice to be made between two opposing and mutually contradictory standpoints: whether to go for a simple, shallow approach, which is less expressive or representative of the information to be modelled, but easier to annotate consistently and to process; or for a complex, in-depth approach which allows for more detailed expression and representation but is more difficult to annotate consistently and more 'expensive' to process.

Second, the complexity of annotation has an enormous effect on the ability to produce consistent and reliable markup of texts, both for human- and machine-annotation. Human, hand-annotated texts will always be prone to mistakes. Especially where it concerns natural language, with its inherent vagueness and ambiguity, a small weakness in specification will lead to multiple potential interpretations and therefore to multiple potential diverging applications of the annotation. Corollary to this, as most attempts to tag text automatically use human annotated texts to train the programs, complexity will also have consequences upon the efficacy of such taggers.

Finally, developers of different annotation schemes within the same field, or even from within different but related fields, will often come to the task with different aims and objectives in mind, as well as from different theoretical standpoints. Different backgrounds will focus on different aspects, often conflating two or more distinct areas for annotation into one. In other words, there is a problem not only in deciding the granularity of an annotation scheme, but also what exactly it should encompass. Often there is inevitably an overlap between schemes, e.g. between time and event ontologies, which are arguably more or less inseparable when dealing with the annotation of temporal information. It is especially for dealing with issues of this kind that the establishment of a metamodel can be of great help.

Bibliography

Allwood, J. (2000). ‘An activity-based approach to pragmatics’. In H. Bunt & W. Black (eds.) *Abduction, Belief and Context in Dialogue*. Benjamins, Amsterdam: 47-80.

Bunt, H. (1985). *Mass terms and model-theoretic semantics*. Cambridge University Press, Cambridge, UK.

Bunt, H. (2000). ‘Dialogue Pragmatics and Context Specification’. In H. Bunt & W. Black (eds.) *Abduction, Belief and Context in Dialogue*. Benjamins, Amsterdam: 81-150.

Bunt, H. & Y. Girard (2005). ‘Designing an open, multimodal dialogue act taxonomy’. In C. Gardent & B Gaiffe (eds.), *Proceedings of the Ninth Workshop on the Semantics and Pragmatics of Dialogue* (*DIALOR’05)*, Nancy, June 2005.

Bunt, H. & R. Muskens (2001). *Computing Meaning, Volume 2*. Dordrecht: Kluwer Academic Publishers.

Bunt, H. & L. Romary (2002). ‘Towards multimodal content representation’. *Proceedings LREC 2002 Workshop*.

Bunt, H. & L. Romary (2004). ‘Standardization in multimodal content representation: Some methodological issues’. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC 2004)*, 2219-2222.

Clark, H. H. (1996). *Using language.* Cambridge: Cambridge University Press.

Davies, S. & M. Poesio (2000). ‘Coreference’. In *MATE Dialogue Annotation Guidelines-Deliverable D2.1*, January 2000, 126-182.

Fillmore, C. J. (1976). ‘Frame semantics and the nature of language’. In *Annals of the New York Academy of Sciences: Conference on the Origin and Development of Language and Speech*, **280**: 20-32.

Grosz, B.J., & Sidner, C.L. (1986). ‘Attention, intentions, and the structure of discourse’. *Computational Linguistics*, **12** (3): 175-204.

Hasida, K. (2005). ‘Discourse relations’. Presented at the *Joint ISO-SIGSEM Workshop on the representation of Multimodal Semantic Information*, Tilburg, January 10-11, 2005. See http://let.uvt.nl/research/ti/iso-tdg3/.

Hirschman, L. & N. Chinchor (1997). ‘MUC-7 coreference task definition’. In *MUC-7 Proceedings*.

Hobbs, J. R. & F. Pan (2004). ‘An ontology of time for the semantic web’. *ACM Transactions on Asian Language Information Processing (TALIP)*: 66-85

Hobbs, J.R.,& J. Pustejovsky (2003). ‘Annotating and Reasoning about Time and Events’. In *Proceedings of AAAI Spring Symposium on Logical Formalizations of Commonsense Reasoning,* Stanford, California, March 2003.

Ichikawa, T. (1990). Bunshoron-gaisetsu. Kyóiku-Shuppan, Tokyo.

Jurafsky, D. & J. H. Martin (2000). *Speech and Language Processing.* Prentice-Hall, New York.

Larsson, S. (1998) ‘Coding schemes for dialog moves’. Unpublished paper, available at http://www.ling.gu.se/sl

Levin, B. (1993). *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press, Chicago.

Lyons, J. (1968). *Introduction to Theoretical Linguistics*. Cambridge University Press, Cambridge.

Mann, W. and S. Thompson (1988). 'Rhetorical Structure Theory: Toward a functional theory of text organization'. *Text* **8**(3): 243-281.

Palmer, M., D. Gildea & P. Kingsbury (2005). 'The Proposition Bank: An Annotated Corpus of Semantic Roles'. Computational Linguistics, **31**(1): 71-106.

Pan, F (2005). 'A Temporal Aggregates Ontology in OWL for the Semantic Web'. In *Proceedings of the AAAI Fall Symposium on Agents and the Semantic Web*, Arlington, Virginia, 2005.

Pan, F. & J. R. Hobbs (2004). 'Time in OWL-S'. In *Proceedings of the AAAI Spring Symposium on Semantic Web Services*, Stanford University, CA: 29-36.

Pan, F. & J. R. Hobbs (2005). 'Temporal aggregates in OWL-Time'. In *Proceedings of the 18th International Florida Artificial Intelligence Research Society Conference (FLAIRS-2005)*, Clearwater Beach, Florida: 560-565, AAAI Press.

Poesio, M. (2004). 'The MATE/GNOME Proposals for Anaphoric Annotation, Revisited'. In *Proceedings of the 5th SIGdial Workshop*.

Pustejovsky, J., J. Castaño, R. Ingria, R. Saurí, R. Gaizauskas, A. Setzer & G. Katz (2003). 'TimeML: Robust specification of event and temporal expressions in text'. In *Proceedings of the Fifth International Conference on Computational Semantics*, Tilburg.

Reichenbach, H. (1947). *Elements of symbolic logic*. Macmillan.

Romary, L. (ed.) (2004). 'Specification of data categories and management of a data category for language resources'. ISO document ISO/TC 37/SC 3 N 488. KORTERM, Taejon (Korea).

Salmon-Alt, S. & L. Romary (2004a). 'RAF – towards a reference annotation framework'. In *Proceedings of LREC 2004*.

Salmon-Alt, S. & L. Romary (2004b). 'Data Categories for a Normalized Reference Annotation Scheme'. In *5th International Conference on Discourse Anaphora and Anaphor Resolution* - DAARC'2004, São Miguel (Azores), Portugal. 2004.

van Deemter, K. & R. Kibble (2000). 'On Coreferring: Coreference in MUC and related annotation schemes'. *Computational Linguistics* 26(4).

Vieira R., S. Salmon-Alt S., C. Gasperin, E. Schang, G. Othero (2003). 'Coreference and anaphoric relations of demonstrative noun phrases in a multilingual corpus'. In *Anaphora Processing*. Branco, A. et al. (eds.), John Benjamins Publishing Company.