INTERNATIONAL STANDARD

ISO 24617-6

First edition
2016-02-15

Language resource management — Semantic annotation framework —

Part 6:
Principles of semantic annotation (SemAF Principles)

Gestion des ressources linguistiques — Cadre d'annotation sémantique —

Partie 6: Principes d'annotation sémantique (SemAF Principes)

Reference number
ISO 24617-6:2016(E)

Contents

Page

Foreword ... **iv**

1 Scope ... **1**

2 Terms and definitions ... **1**

3 Purpose and motivation ... **2**
- 3.1 Purpose ... 2
- 3.2 Motivation ... 2

4 Overview ... **3**

5 Annotation principles and requirements ... **4**
- 5.1 Principles inherited from the Linguistic Annotation Framework ... 4
- 5.2 Other general annotation principles ... 5
- 5.3 Principles specific to semantic annotation ... 5

6 The methodological basis of SemAF ... **7**
- 6.1 Steps in the design of an annotation scheme ... 7
- 6.2 Metamodels ... 8
- 6.3 Abstract syntax, concrete syntax and semantics ... 10
- 6.4 Steps forward and feedback in the design process ... 12
- 6.5 Optional elements in an annotation scheme ... 14

7 Overlaps between annotation schemes ... **15**
- 7.1 Semantic and terminological consistency ... 15
- 7.2 Spatial and temporal relations as semantic roles ... 15
- 7.3 Events ... 17
- 7.4 Discourse relations in dialogue ... 18

8 Semantic phenomena that cut across annotation schemes ... **18**
- 8.1 Ubiquitous semantic phenomena ... 18
- 8.2 Quantification ... 18
- 8.3 Quantities and measures ... 19
- 8.4 Negation, modality, factuality, and attribution ... 20
- 8.5 Modification and qualification ... 21
  - 8.5.1 Modification and quantification ... 21
  - 8.5.2 Qualification ... 22
  - 8.5.3 Other issues ... 23

Annex A (informative) **An approach to the annotation of quantification in natural language** ... **24**

Bibliography ... **28**

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the WTO principles in the Technical Barriers to Trade (TBT) see the following URL: Foreword - Supplementary information.

The committee responsible for this document is ISO/TC 37, *Terminology and other language and content resources*, Subcommittee SC 4, *Language resource management.*

ISO 24617 consists of the following parts, under the general title *Language resource management — Semantic annotation framework (SemAF)*:

— *Part 1: Time and events (SemAF-Time, ISOTimeML)*

— *Part 2: Dialogue acts (SemAF-Dacts)*

— *Part 4: Semantic roles (SemAF-SR)*

— *Part 5: Discourse structures (SemAF-DS)* [Technical Specification]

— *Part 6: Principles of semantic annotation (SemAF Principles)*

— *Part 7: Spatial information (ISOspace)*

The following parts are in preparation:

— *Part 8: Semantic relations in discourse (SemAF DR-core)*

— *Part 9: Reference (ISOref)*

Language resource management — Semantic annotation framework —

Part 6:
Principles of semantic annotation (SemAF Principles)

1 Scope

This part of ISO 24617 specifies the approach to semantic annotation characterizing the ISO Semantic annotation framework (SemAF). It outlines the SemAF strategy for developing separate annotation schemes for certain classes of semantic phenomena, aiming in the long term to combine these into a single, coherent scheme for semantic annotation with wide coverage. In particular, it sets out the notions of both an abstract and a concrete syntax for semantic annotations, mirroring the distinction between annotations and representations that is made in the ISO Linguistic Annotation Framework. It describes the role of these notions in relation to the specification of a metamodel and a semantic interpretation of annotations, with a view to defining a well-founded annotation scheme.

This part of ISO 24617 also provides guidelines for dealing with two issues regarding the annotation schemes defined in SemAF-parts: a) conceptual and terminological inconsistencies that may arise due to overlaps between annotation schemes and b) the treatment of semantic phenomena that cut across SemAF-parts, such as negation, modality and quantification. Instances of both issues are identified, and in some cases, direction is given as to how they may be tackled.

2 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

NOTE In addition, the terms ‘event’ and ‘eventuality’ are used (as synonyms) as defined in ISO 24617-1 as *something that can be said to obtain or hold true, to happen or to occur*.

2.1
primary data
electronic representation of text or communicative behaviour

EXAMPLE Digital representations of text, transcriptions of speech, gestures or multimodal dialogue.

Note 1 to entry: ISO 24612 defines primary data as the ‘electronic representation of language data’. This definition is unsatisfactory for this part of ISO 24617 as semantic annotation may relate to non-verbal or multimodal data, such as stretches of spoken dialogue with accompanying gestures and facial expressions, and even gestures and/or facial expressions without any accompanying speech.

2.2
annotation
linguistic information added to *primary data* (2.1), independent of its representation

[SOURCE: ISO 24612:2012, 2.3]

2.3
semantic annotation
annotation (2.2) which contains information about the meaning of a segment or region of *primary data* (2.1)

2.4
metamodel
schematic representation of the concepts that are used in the analysis and description of the phenomena covered in *annotations* (2.2) and of the relationships between them

3 Purpose and motivation

3.1 Purpose

The purpose of this part of ISO 24617 is to provide support for the establishment of a consistent and coherent set of international standards for semantic annotation within the Semantic Annotation Framework (SemAF). It aims to do so in three ways.

First, by making explicit which basic principles underlie the approach that has been followed in defining international standards in the SemAF parts that have been published so far (ISO 24617-1 and ISO 24617-2, ISO 24617-4 and ISO 24617-7), and in parts that are close to publication (ISO 24617-6) or in preparation (ISO 24617-8). This approach provides the Semantic Annotation Framework with methodological coherence and helps to ensure mutual consistency between existing, developing, and future SemAF parts.

Second, by identifying overlaps between SemAF parts and indicating how such overlaps may be dealt with. Examples are the occurrence of temporal and spatial relations among semantic roles and of discourse relations between dialogue acts.

Third, by identifying common issues that arise in various parts of SemAF (they are only partly covered in these parts, if they are covered at all) and, where possible, by giving directions as to how these issues may be tackled. Examples of such issues are polarity, modality, quantification, measures, qualification, veridicity, attribution and non-literal language use.

3.2 Motivation

Semantic annotation enhances primary data with information about their meaning. The state of the art in computational semantics makes it unlikely that a single existing formalism for annotating semantic information would receive wide support from researchers and developers. Moreover, semantic annotation tasks often have the limited aim of annotating certain specific semantic phenomena, such as semantic roles, discourse relations or coreference relations, rather than annotating the full meaning of stretches of primary data. A strategy was therefore adopted in ISO TC 37/SC 4 to devise the SemAF standards in different parts, with separate annotation schemes for those classes of semantic phenomenon for which the state of the art would justify the establishment of annotation standards; these schemes could be extended and combined over time, growing into a wide-coverage framework for semantic annotation.

This ‘crystal growth’ strategy has contributed significantly to the progress made in establishing standardized annotation concepts and schemes supporting the development of interoperable resources, but it also entails certain risks:

a) the annotation schemes defined in different SemAF parts are not necessarily mutually consistent, especially in the case of overlaps in scope;

b) it may not be possible to combine the schemes, defined in different parts, into a coherent single scheme with a wider coverage if they incorporate different views or employ different methodologies;

c) some semantic phenomena do not belong to the scope of any SemAF parts but cannot be disregarded entirely in some parts, and this may result in these phenomena being unsatisfactorily treated.

The methodological principles and guidelines provided in this part of ISO 24617 are designed to minimize these risks.

With regard to the issue of mutual consistency between SemAF parts, it may be noted that ISO 24617-1 for annotating time and events and ISO 24617-2 for annotating dialogue acts are concerned with sufficiently distinct kinds of semantic information to allow their definitions to be established independently. Other SemAF parts, such as those concerned with semantic roles, with relations in discourse and with spatial information show a certain amount of overlap in the information that they aim to capture, and the question therefore arises: can we ensure that the annotation schemes, defined in these parts, are mutually consistent?

Mutual consistency of SemAF parts relates to the possible *integration* of annotation schemes defined in different parts. For example, it would be desirable to use the ISO 24617-1 scheme ("ISO-TimeML") for annotating time and events in combination with the ISO 24617-4 scheme for semantic roles, thereby annotating coherently not only the events identified in the data with their temporal properties, but also the way in which these events are related to their participants. Integrating these annotations with those of spatial information, using the ISO 24617-7 scheme for spatial information, would be another plausible and desirable step, given that time and space are intertwined with concepts relating to motion and velocity. More generally, the integration of SemAF parts would greatly enhance the significance of the individual parts; in the end, SemAF's 'crystal growth' strategy of SemAF is only really useful if the annotation schemes defined in the various parts can grow into a single scheme with a wide coverage of semantic phenomena. Only then can it effectively support such applications as text-based question answering or extracting semantic information from text, and form the basis for automatically recognizing semantic phenomena by means of machine-learning techniques. Clearly, this is only possible if the annotation schemes are mutually consistent (e.g. they use the same classification of event types), and are coherent whether, for example, temporal and spatial objects are viewed as event participants or as the circumstances of an event.

With regard to the risk of unsatisfactory partial treatments of phenomena that are not among the core issues of any (current) SemAF part, it should be noted that some of these phenomena cut across several of these parts and are important for semantics-driven applications. Negation, or more generally negative polarity, and quantification are two cases in point. Given that the aim in ISO-TimeML, for instance, is to support the annotation of events, of their relation to time, and of the temporal relations among temporal objects, it is desirable to be able to deal with sentences like the following:

(1) John teaches every Monday.

(2) Mary called twice this morning.

(3) John rang home twice a day.

Sentence (1) is about a *set* of "*teach*" events, each of which is related to a different element of the set of temporal objects that are called "*Monday*", so this is a case of quantification involving two sets, a set of events and sets of days. Similarly, sentence (2) about a set of two "*call*" events, both related to the same period of time. Sentence (3) is about a set of events and their frequency of occurrence.

In order to deal with such phenomena, ISO-TimeML has certain provisions for annotating quantification, but they are not really adequate[13] and do not generalize to cases of quantification where no events are involved.

4 Overview

The ISO efforts aiming to develop standards for semantic annotation rest on certain basic principles, some of which have been laid out by Reference [14] as requirements for semantic annotation, and have been developed further in Reference [5]; others have been formulated as general principles for linguistic annotation and are part of the ISO Linguistic Annotation Framework (LAF; see Reference [18] and ISO 24623-1). The two sets of principles and requirements are considered in Clause 5.

The three kinds of risk associated with the SemAF 'crystal growth' strategy that have been identified above correspond to the following issues of consistency and completeness that arise in the design of semantic annotation schemes within the SemAF framework.

Consistency among annotation schemes:

— *methodological consistency*: the same basic approach is followed with respect to the distinction between abstract and concrete syntax and their interrelation, and with respect to their semantics;

— *conceptual consistency*: different schemes are based on compatible underlying views and ontological assumptions regarding their basic concepts, as reflected in metamodels (e.g. verbs are viewed as denoting states or events, rather than relations);

— *terminological consistency*: terms that occur in different annotation schemes have the same meaning in every scheme and the same term is used across annotation schemes to indicate the same concept.

Completeness of a set of annotation schemes: the combination of multiple annotation schemes leads to a scheme that

— covers a wide range of semantic phenomena,

— does not have significant gaps when covering the semantic phenomena that it aims to cover, and

— deals in a satisfactory way with semantic phenomena that cut across the combined schemes but which do not belong to the core phenomena that any of the combined schemes are designed to cover.

Clause 5 describes the methodological framework for defining annotation schemes in SemAF parts, thereby ensuring methodological consistency. Clause 6 discusses conceptual and terminological consistency issues that arise due to overlaps between SemAF parts, while Clause 7 identifies issues of completeness regarding the annotation of semantic phenomena that cut across existing SemAF parts.

5 Annotation principles and requirements

5.1 Principles inherited from the Linguistic Annotation Framework

The annotation of semantic information when using SemAF inherits the principles for linguistic annotation as formulated in LAF. These principles are often of a very general nature; they include the principle that relevant segments of primary data are referred to in a uniform and TEI-compliant way, and the principle that different layers of annotation over the primary data can co-exist by using stand-off annotation and a uniform way of cross-referencing between layers.

The latter principle, which concerns the distinction of layers of annotation enabled by a stand-off representation format, is of particular relevance for SemAF because it allows different annotation layers to be used for different types of semantic information; for example, one layer could be used for the annotation of events, time and space, and another one could be used to annotate semantic roles. In principle, this allows for the use not only of layers that are not mutually consistent, but also of alternative annotations that employ different annotation schemes for the same phenomena. However, the SemAF 'crystal growth' strategy is designed to ensure that the annotation schemes for the various types of semantic information can grow into a coherent annotation scheme for a wide range of semantic phenomena, and it is therefore highly undesirable to have inconsistencies between annotation layers concerned with different SemAF parts.

Also of particular relevance for SemAF is the distinction between 'annotations' and 'representations'.[18] An *annotation* is any item of linguistic information that is added to primary data, independently of any particular representation format. A *representation* is a format into which an annotation is rendered, for example as an XML expression. ISO standards are assumed to be defined at the level of annotations, rather than representations. The fundamental distinction between annotations and representations has prompted the development of a methodology for developing semantic annotation schemes that draws a distinction between the 'abstract syntax' of annotations and the 'concrete syntax' of representations. This methodology is described in Clause 6.

5.2 Other general annotation principles

In addition to the principles that SemAF inherits from LAF, other general principles for designing annotation schemes (in particular as part of an ISO standard) are worth mentioning; most of these emerged during the development of the ISO 24617-2 standard for dialogue act annotation.

a) **Theoretical validity**: Annotation standards should consolidate existing knowledge and accordingly should be firmly rooted in theoretical studies of the annotated phenomena. Any concept that may occur in annotations according to the standard should therefore be well established in the scientific literature.

b) **Empirical validity**: Annotation standards are designed to be useful for annotating corpora of recorded empirical data; the annotation scheme defined in a standard should not therefore include theoretical constructs that are not found in such corpora, but only concepts that correspond to phenomena that are observed in empirical data.

c) **Learnability**: For an annotation scheme to be useful in the construction of annotated language resources, it should be possible both for human annotators and for automatic annotation systems to effectively learn how to apply the scheme with acceptable precision.

d) **Generalizability**: ISO standards should not be restricted in their applicability to particular languages, subject domains or applications.

e) **Extensibility**: While ISO standard annotation schemes are designed to be language-independent, domain-independent and application-independent, some applications and some languages may require specific concepts that are not relevant in other applications or languages. Annotation schemes should therefore be open, that is to say, they should allow extension with language-specific, domain-specific and application-specific concepts.

f) **Completeness**: An annotation standard is designed to provide a good coverage of the phenomena of which it is designed to enable the annotation; the set of concepts defined in an annotation standard should, in that sense, be complete.

g) **Variable granularity**: One way to achieve good coverage is to include annotation concepts of a high level of generality and which cover many specific instances. Since an annotation scheme which uses *only* very general concepts would not be optimally useful, this leads to the principle that annotation schemes should include concepts with different levels of granularity. This is also beneficial for its interoperability, as it provides more possibilities for conversion between existing annotation schemes and the standard scheme.

h) **Compatibility**: In order to enable mappings between alternative annotation schemes and thereby contribute to the interoperability of annotated resources, concepts that are commonly found in existing annotation schemes should preferably be included in an annotation standard.

5.3 Principles specific to semantic annotation

The idea behind annotating a text, which dates from long before the digital era, is to add information to a primary text in order to support its understanding. The semantic annotation of digital source texts has a similar purpose, namely to support the understanding of the text by humans, as well as by machines.

An annotation that does not add any information would therefore seem to make little sense, but the following example of the annotation of a temporal expression using TimeML seems to do just that:[39]

NOTE 1 For simplicity, the annotations of the events that are mentioned in the previous sentence is suppressed here.

(4)

```
<timeml>
 The CEO announced that he would resign as of
 <TIMEX3 tid="t1" type="date" value="2008-12-01"/>
  the first of December 2008
 </TIMEX3>
</timeml>
```

In this annotation, the subexpression `<TIMEX3 tid="t1" type="date" value="2008-12-01"/>` adds to the noun phrase "*the first of December 2008*" the information that this phrase describes: the date 2008-12-01.This does not add any information; rather, it paraphrases the noun phrase in TimeML. This could be useful if the expression in the annotation language had a well-specified semantics that could be used directly by computer programs for applications like information extraction and question answering. Unfortunately, TimeML does not have a semantics.

NOTE 2 It would be very simple to provide a semantics for the XML fragment shown here but it would be very difficult to do so for the whole of TimeML. See also 7.3.

A case where the annotation of a date as in the above example *does* add something is (5). From the utterance "*Mr Brewster called a staff meeting today*", it is impossible to know the date on which the event that is mentioned took place; in this case, the annotation, which is identical to (4), would be informative.

NOTE 3 Note that the examples of TimeML annotations shown here are "old-fashioned" in the sense that the TIMEX3 element is wrapped around the annotated string. Modern annotation methods (e.g. in ISO-TimeML) use stand-off representations.

(5) Mr Brewster called a staff meeting today.

```
<timeml>
 Mr Brewster called a staff meeting
 <TIMEX3 tid = t1 type = "date" value = "2008-12-01"/>
 today
 </TIMEX3>
</timeml>
```

The examples in (4) and (5) illustrate two different functions that semantic annotations may have: interpreting a natural language expression by recoding it in a formal annotation language with a well-defined semantics, and adding context information, in order to allow the interpretation of context-dependent expressions.

A third function that semantic annotations may have is to make explicit how certain parts of an utterance are semantically related; for example, this is the function of semantic role labelling and of indicating semantic relations between sentences in a discourse when no such relation is mentioned. Note that the first function presupposes annotations to have a well-defined semantics; the other two functions do not presuppose this, but since semantic annotations in digital corpora are typically designed to support interpretation and inference, a desideratum for all functions that a semantic annotation may have is that it has a well-defined semantics. Semantic annotations, like other linguistic annotations, may also serve the purpose of supporting linguistic research, such as identifying syntactic and semantic patterns in sentences and texts.

These considerations lead to the following two principles for semantic annotation:

— **Semantic additivity**: *semantic annotations add semantic information to source data, or re-express certain source data in a formal representation*.

— **Semantic adequacy**: *semantic annotations should have a well-defined semantics, making the annotations machine-interpretable*.

6 The methodological basis of SemAF

6.1 Steps in the design of an annotation scheme

An annotation scheme determines which information may be added to primary data and how that information is expressed. The design of an annotation scheme from scratch should begin with a conceptual analysis of the information that the annotations should capture. This analysis identifies the concepts that form the building blocks of annotations and specifies how these blocks may be used to build annotation structures. This may be broken down into two steps, the first establishing a conceptual view of the phenomena to be annotated, the second an articulation of this view in the form of a formal specification of categories of entities and relations, and of how annotation structures can be built up from elements in these categories. The first of these steps corresponds to what is known in ISO projects as the establishment of a 'metamodel', that is to say, the expression of a conceptual view of the phenomena to be annotated in the form of a (UML) diagram. The formal specification produced in the second step constitutes the abstract syntax of an annotation language.

While these two steps make explicit what information can be captured in the annotations, they do not concern the use of representation formats, such as XML strings, logical formulas, graphs or feature structures; the abstract syntax defines the specification of information in terms of set-theoretic structures, called *annotation structures*. An annotation structure is a collection of two kinds of structure, *entity structures* and *link structures*. An entity structure contains semantic information about a segment of primary data and is formally a pair $<m, s>$ consisting of a markable, which refers to a segment of primary data, and of certain semantic information. A link structure contains information about the way two segments of primary data are semantically related; for example, in semantic role annotation a link structure is a triple $<e_1, e_2, R_i>$ where e_1 is an entity structure that contains information about an event, e_2 is an entity structure that contains information about a participant in the event, and R_i is a relation denoting a semantic role.

The third step in the definition of an annotation scheme is the specification of the meaning of the structures defined by the abstract syntax, that is to say, the specification of semantics for annotation structures.

The fourth and final step is the definition of a format for representing annotation structures, such as a serialization in XML. The semantics of an expression in this format is that of the annotation structure that it represents. In sum, this approach to designing annotation schemes consists of four steps, with as many feedback loops as may be needed.

The method consisting of these four steps is called 'CASCADES': **C**onceptual analysis, **A**bstract syntax, **S**emantics, and **C**oncrete syntax for **A**nnotation language **DES**ign. Figure 1 visualizes the CASCADES method, of which the central concept, that of an abstract syntax for annotations with the specification of a semantics for annotation structures (rather than for their representations in a particular format), was introduced in Reference [5].

The CASCADES method is useful for enabling a systematic design process, in which due attention is given to the conceptual and semantic choices on which more superficial decisions, such as the choice of particular XML attributes and values, should be based. Apart from supporting the design of an annotation scheme starting from scratch, the method also provides support for improving an existing annotation scheme. This support consists not only of the distinction of four well-defined design steps, but also of procedures and guidelines for taking these steps. These procedures are outlined in 6.3. But before that, the use of metamodels, resulting from the conceptual analysis phase of designing an annotation scheme, is discussed in 6.2.

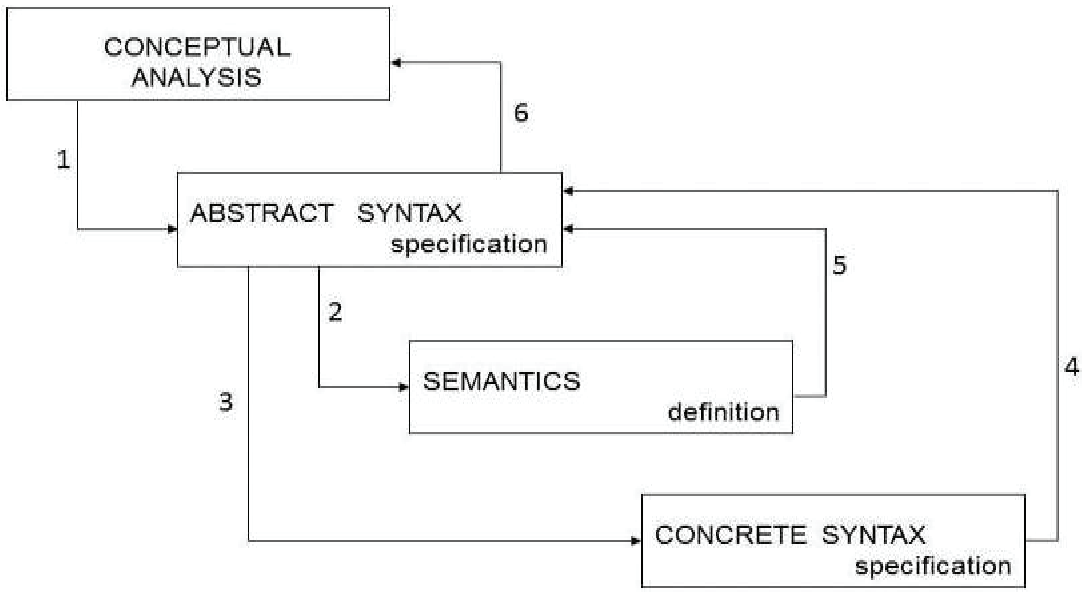

Figure 1 — Steps and feedback loops in the CASCADES method

6.2 Metamodels

A metamodel of an annotation standard is a schematic representation of the relations between the concepts that are important in the analysis and description of the phenomena covered in the annotations. Over the years, two slightly different notions of a metamodel have been used in ISO projects, namely the following:

— A: as a representation of the relations between the most important concepts that are mentioned in the document in which the standard is defined;

— B: as a representation of the relations between the concepts denoted by terms that occur in annotations, according to the standard.

Type-A metamodels may help nontechnical readers to have a better understanding of annotation schemes; those of type-B are a visual representation of the abstract syntax of the annotations according to the scheme and may help readers see at a glance what information the annotations may contain. (Note that a type-A metamodel may have a type-B metamodel as a proper part.) For example, Figure 2 shows the metamodel of ISO 24617-4 for semantic role labelling. This metamodel reflects the view that semantic roles are relations between eventualities and their participants; different roles correspond to different ways in which a participant is involved in an eventuality. For example, in "*Chris wrote a poem",* a poem participates in an event in the *Result* role (i.e. created by the event); in "*Chris revised the poem*", the poem is involved in the *Patient* role (i.e. affected by the event); and in "*Chris read a poem",* a poem participates in the *Theme* role (i.e. not affected by the event). The annotation of semantic roles therefore involves the use both of eventualities, most often corresponding to verbs, and of participants, typically expressed by noun phrases. The metamodel associates eventualities and participants with markables, whose source is a stretch of text in the primary data. The participants in an eventuality are typically individuals, but may also be properties, numbers, quantities, propositions, sets of individuals or embedded eventualities, as the examples in (6) illustrate.

(6) a. *He painted his house blue.*

b. *The birth of the twins increased the number of children to four.*

c. *The weight of this suitcase exceeds 20 kilos.*

d. *John believes <u>he will get the money together.</u>*

e. *<u>Marie, Jill and Chris</u> got together.*

f. Eric and Nicole went to <u>the concert</u>.

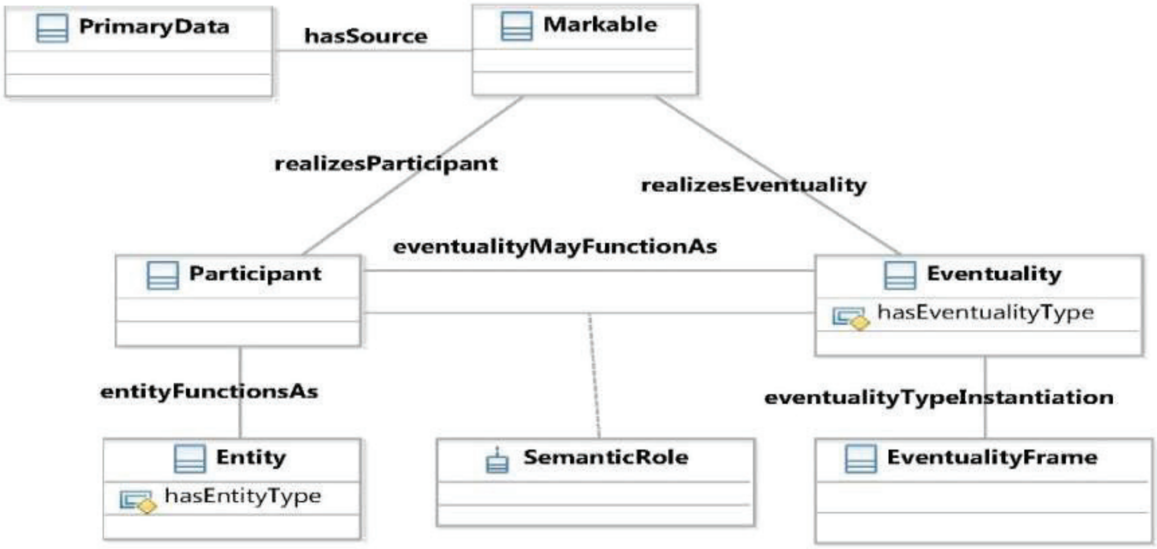

Figure 2 — Metamodel for semantic role annotation (ISO 24617-4)

The term ‘entity’ is used in ISO 24617-4 to denote anything except an eventuality; the fact that a participant in an eventuality can be an ‘entity’ of any sort as well as an (embedded) eventuality, as in (6f), is reflected in the metamodel by the “entityFunctionsAs” relation between entities and participants, and the “eventualityMayFunctionAs” relation between eventualities and participants. Many approaches to semantic role labelling (e.g. PropBank, FrameNet and VerbNet) make use of “eventuality frames” that specify the set of semantic roles expected in a type of eventuality, and take the eventuality frame as a whole into consideration when determining the choice of individual semantic roles. For that reason, eventuality frames have also been included in the metamodel.

Is this a type-A metamodel or a type-B metamodel? The following is an example of an annotation representation according to ISO 24617-4 (where m1 is a markable that refers to *“The soprano”*; m2 is a markable that refers to “*sang*”, and m3 is a markable that refers to *“an aria*”):

(7) a. The soprano sang an aria.

b.

```
<event xml:id="e1" target="#m2" eventFrame="sing.01"/>
<entity xml:id="x1" target="#m1"/>
<srLink event="#e1" participant="#x1" semRole="agent"/>
<entity xml:id="x2" target="#m3"/>
<srLink event="#e1" participant="#x1" semRole="theme"/>
```

In this annotation, eventualities are directly related to entities by means of an element like `<srLink event="#e1" participant="#x1" semRole="agent"/>`. There is no ‘participant’ object, related to an entity via an ‘entityFunctionsAs relation’, as there is in the metamodel. It follows that this metamodel is not just a visual representation of the (abstract) syntax of annotation structures, but rather a type-A model, since it has elements that do not appear in annotations. This is corroborated by the fact that eventuality frames do not occur as such in annotations; eventuality frames may be used in the *process* of semantic role labelling, but are not part of the resulting annotations.

6.3 Abstract syntax, concrete syntax and semantics

The *abstract syntax* of an annotation scheme, as mentioned above, specifies the information in annotations in terms of set-theoretical structures such as pairs and triples, like the triple $<e_1, e_2, R_i>$ which relates the two arguments e_1 and e_2 through the relation R_i. More generally, these structures are *n*-tuples of elements that are either basic concepts taken from a store of basic concepts called the 'conceptual inventory' of the abstract syntax specification, or *n*-tuples of such structures.

A *concrete syntax* specifies a representation format for annotation structures, such as the XML format illustrated in (7), where a triple like $<e_1, e_2, R_i>$ is represented by a sequence of three XML elements, of which the element `<srLink event="#e1" participant="#x1" semRole="agent"/>` represents the relation and the other two elements represent two entity structures.

A representation format for annotation structures should ideally give an exact expression of the information contained in annotation structures. A concrete syntax, defining a representation format for a given abstract syntax, is said to be *ideal* if it has the following properties:

— **completeness**: every annotation structure defined by the abstract syntax can be represented by an expression defined by the concrete syntax;

— **unambiguity**: every representation defined by the concrete syntax is the rendering of exactly one annotation structure defined by the abstract syntax.

The representation format defined by an ideal concrete syntax is called an *ideal representation format*. Due to its completeness, an ideal concrete syntax CS_i defines a function F_i from annotation structures to CS_i-representations, and due to its 'unambiguity', there is also an inverse function F_i^{-1} from CS_i-representations to annotation structures. If I_a is the interpretation function defining the semantics of the abstract syntax, then the meaning of a representation *r* in the ideal format CS_k is defined by $I_a(F_k^{-1}(r))$. It follows that, for any two ideal representation formats CS_i and CS_j, there is a meaning-preserving conversion C_{ij} defined by:

(8) $C_{ij}(r) = F_j\,[F_i^{-1}(r)]$

This mapping is meaning-preserving due to the fact that the meaning of a representation is the meaning of the annotation structure that it encodes. Since this holds for *any* ideal concrete syntax, it follows that any two ideal representation formats are semantically equivalent.

Figure 3 visualizes the relations between abstract syntax, semantics, and alternative ideal concrete-syntax specifications. It is immediately clear that a given representation *r* defined by concrete syntax CS_i can be converted into a semantically equivalent representation *r'* in the representation format CS_j by first applying the function F_i^{-1} in order to determine the annotation structure which it encodes, and applying to that annotation structure the function F_j which encodes it in the format CS_j.

NOTE 1 The conditions defining an ideal concrete syntax require every annotation structure to have at least one representation (but they do not rule out the possibility of an annotation structure having more than one representation; the encoding functions, like F_j, may assign multiple CS_i-representations to a given annotation structure). If these conditions are met, a conversion function like the one defined in (8) assigns a *set of* CS_j*-equivalents* to a given CS_i-representation.

An ideal concrete syntax can be derived systematically from an abstract syntax. The procedure for doing so depends on the kind of concrete representations that are aimed for. For example, a concrete syntax defining XML representations can be constructed using the following procedure:

NOTE 2 The description given here is simplified, for illustrative purposes. For more details, see Reference [7].

a) For each element of the conceptual inventory specify an XML name;

b) For each type of entity structure $<m, s>$, define an XML element with the following attributes and values:

— the special attribute 'xml:id', whose value is an identifier of the element;

— the special attribute ‘target’, whose value represents the markable, *m*;

— attributes whose values represent the components of *s*.

c) For each type of link structure, define an XML element with three attributes, two whose values refer to the representations of the entity structures that are linked, with the value of the third denoting the relation between them.

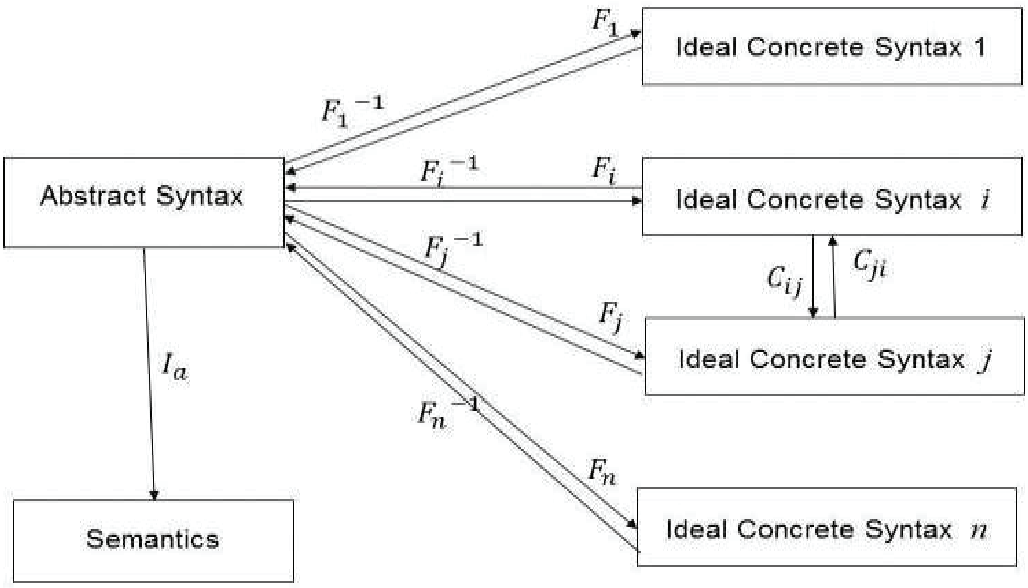

Figure 3 — Relations between abstract syntax, semantics, and concrete syntax of annotations

Proposals for semantics of semantic annotations include the Interval Temporal Logic semantics for TimeML,[35] the event-based semantics for TimeML,[10] and other attempts to formally interpret temporal annotations.[27][29] In Reference [4] and Reference [11], a semantic annotation language is presented with a formal semantics that integrates temporal information, semantic roles and coreference relations. These proposals all involve a translation of annotation representations into first-order logic, and have shown that it is very hard to define such a semantics in a satisfactory, compositional manner, where the translation of a representation would be systematically constructed from the translations of the representation components (see Reference [29]). A different approach, proposed in Reference [6], bases the semantics on Discourse Representation Theory (DRT).[24] The use of Discourse Representation Structures (DRSs) has an advantage over the use of first-order logic, with which DRSs are formally equivalent, due to the fact that these structures were designed to facilitate their incremental construction. This feature can be exploited when constructing DRSs systematically from the components of an annotation representation.

These proposals all make use of a translation of XML-representations into an interpreted logical formalism. By contrast, the CASCADES approach defines a semantics for abstract *annotation structures*, rather than XML-representations. Such a semantics is outlined in Reference[9] and makes use of the fact that annotation structures consist of entity structures and link structures, defined as *n*-tuples of semantic concepts, the significance of an element in an *n*-tuple being encoded by its position (rather than being named by an XML attribute); moreover, annotation structures are translated into a DRS in a compositional way by combining the translations of the component entity structures and link structures. See also Annex A, where this approach is applied to annotations of quantification.

6.4 Steps forward and feedback in the design process

While the procedures for making the CASCADES steps are helpful for defining well-founded annotation schemes, it would be unrealistic to think that annotation schemes can be designed simply through a linear sequence of steps, from conceptual analysis to the specification of a representation format. Realistic design processes require feedback loops. Reference [37] has introduced the "MATTER" cycle for developing an annotation scheme, which distinguishes six steps (see Figure 4). After the initial design of an annotation scheme in step (1), a certain amount of primary data are annotated in step (2). The resulting corpus is used to apply machine-learning and train an annotation program in step (3). In step (4), this program is run and tested on an unannotated corpus. The results are evaluated in step (5), and used to decide on revisions in the annotation scheme in step (6). The entire cycle can then be repeated for the revised scheme.

The MATTER cycle assumes that the testing and evaluation of an annotation scheme are carried out with the help of a machine-learned annotation program. A slightly more general design cycle, proposed in Reference [38] and called the 'MAMA' cycle, does not make this assumption and distinguishes only four steps: (1) Model, (2) Annotate, (3) Evaluate, and (4) Revise.

In the CASCADES method, feedback cycles can occur between each of the four design stages, as shown in Figure 1. First, the specification of an abstract syntax is a way of formalizing the conceptual analysis in the first stage of the process; this formalization may very well clarify or alter some aspects of the initial analysis. Step 6 feeds the results of the formalization back into the conceptual analysis. Second, the specification of a concrete syntax, defining a particular representation format, may motivate adaptations in the underlying abstract syntax; step 4 represents this feedback in the process. Third, since the definition of a semantics for an abstract syntax is a good way of detecting inadequacies in the latter, this may be fed back into the abstract syntax specification in step 5. And finally, the latter two feedback loops can very well be combined; if the feedback in step 4 has resulted in a revised definition of the abstract syntax, this will require revising the semantics (step 2), which may in turn be fed back again into the abstract syntax specification (step 5). This cycle <2;5> may be repeated until the abstract syntax and its semantics are satisfactory and stable, at which point, the annotations are assumed to meet the requirement of semantic adequacy. The concrete syntax should now be adapted to this abstract syntax (step 3).

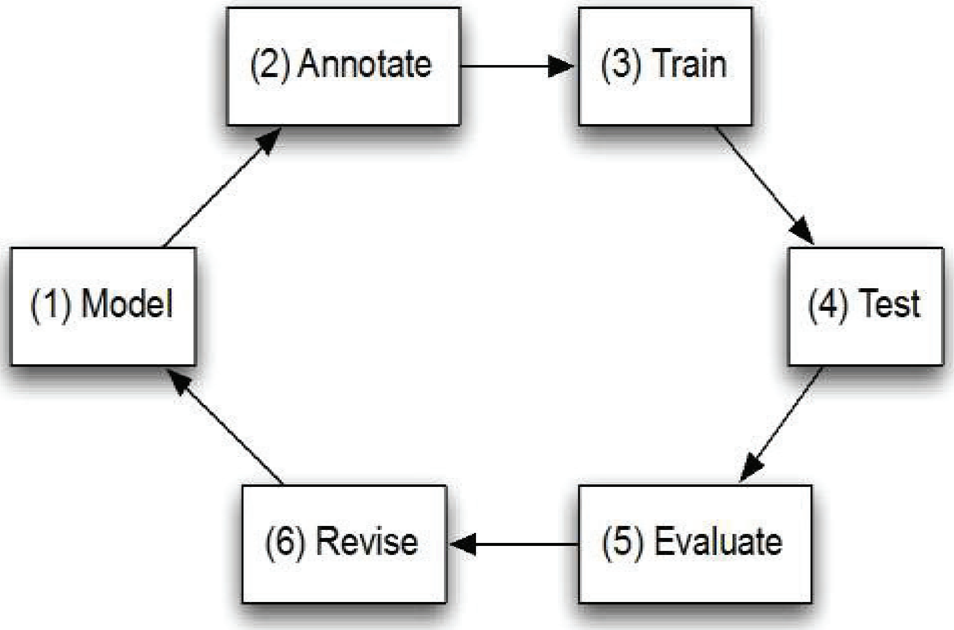

Figure 4 — The MATTER cycle of developing an annotation scheme[37]

In fact, it would not make much sense to perform the ‘outer cycle’ <4;3> if it is not combined with the ‘inner cycle’ <2;5>, resulting together in the iterative feedback loop <4;<2;5>*;3>. This feedback loop is particularly important in the application of the CASCADES method for improving an *existing* representation format, for detecting and resolving semantic deficiencies, and for turning an existing format into an annotation scheme that meets the requirements of the ISO Linguistic Annotation Framework and the requirement of semantic adequacy. In practice, the design of semantic annotations often starts from an existing representation format or annotation practice. When developing an ISO standard, an abstract syntax (with a semantics) should be constructed that fits the representations. Starting from an existing practice, the CASCADES method can be used by following the iterative feedback loop <4;<2;5>*;3>, commencing with the reconstruction of an abstract syntax.

The CASCADES method has been used in this ‘reverse engineering’ mode in the development of ISO-TimeML, starting from TimeML.[39] Ide et al. (2011) have ‘reverse-engineered’ an abstract syntax for the representation format of the Penn Discourse Treebank (PDTB)[36] with the aim of designing a GrAF representation (Ide and Suderman, 2001) for these annotations and have shown that, even without specifying a semantics for this abstract syntax, this leads to improvements in the PDTB annotations. They note that “*the exercise of creating an abstract syntax for the PDTB scheme and rendering it in a graphic form shows the structure of the annotations more clearly. The concrete syntax is much more readable than the original format, and therefore errors and inconsistencies may be more readily identified.”* Similarly, in designing a GrAF-based representation for the annotation of semantic roles in PropBank (Palmer et al., 2005), it was noted that the existing annotation scheme is ambiguous with regard to the relations between the parts of an annotation. Applying the combined ideas of abstract syntax and GrAF to a variety of existing annotation schemes, Ide and Bunt (2010) observe that “*the original PropBank encoding is close to an ideal concrete syntax, as it can be generated from the abstract syntax. However, the round trip back to the abstract syntax is not possible, because it is necessary to do some interpretation of associations among bits of annotations in order to construct the abstract syntax*”; Ide and Suderman (2007) conclude that this is a demonstration of an “*all-too-pervasive feature of many annotation schemes: reliance on human interpretation*”.

The CASCADES design steps and feedback loops integrate perfectly with the MATTER and MAMA development cycles, viewing the CASCADES steps starting with Conceptual Analysis as an implementation of the Model stage of the MATTER and MAMA cycles and the feedback loops as an implementation of the ‘Revise’ stage. To this, the MAMA cycle adds the stages of ‘Annotation’ and ‘Evaluation’ while the MATTER cycle adds the additional stages of ‘Training’ and ‘Testing’ in machine-learned annotation. An advantage of such an integration is that it clarifies the relation between the Model and Revise stages in the MATTER and MAMA cycles. Intuitively, revising an existing annotation scheme should involve some of the same activities as the Model stage; the CASCADES steps make this explicit, since the feedback loops for revising an annotation scheme are also part of the modelling stage. Figure 5 shows the integration of the MAMA and CASCADES cycles. The additional steps of the MATTER cycle (Training and Testing) could clearly be added in between the Annotation and Evaluation steps.

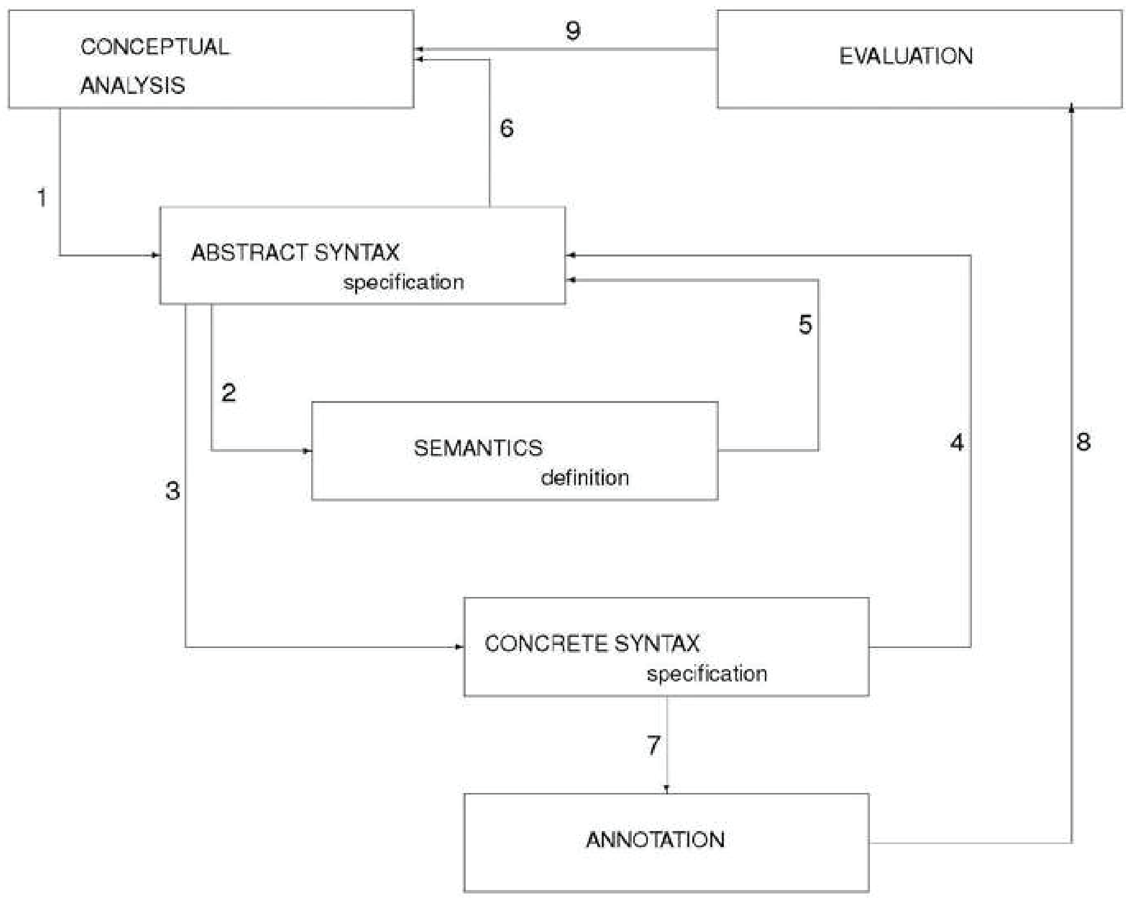

Figure 5 — Integration of the MAMA and CASCADES cycles

6.5 Optional elements in an annotation scheme

The distinction between the abstract and the concrete syntax of annotations opens up interesting possibilities for optional elements to occur in annotations and their representations in various ways.

In a given annotation task, it may be relevant to take account of information that does not form part of the focus of the annotation scheme, but which can be useful for performing the task. The addition of syntactic information to semantic annotations is an example of this. In co-reference annotation, in order to identify chains of co-referential expressions, it is useful to identify the noun phrases that are potential antecedents of referential pronouns according to their grammatical number (singular or plural) and their grammatical gender (or their natural gender, depending on which of these is relevant for pronominal reference in the language under consideration). It is therefore useful to annotate the grammatical number and gender of noun phrases and pronouns. This may now be supported by an annotation scheme that includes the representation of gender and number in the concrete syntax but does not include number and gender information in the abstract syntax, and therefore does not deal with the semantics of number of gender annotations.

Another form of optionality is that the concrete syntax defines default values for certain attributes. For example, an attribute ‘polarity’, with possible values “positive” and “negative”, can be assumed to have the value ‘positive’ by default. Unlike the previous form of optionality, optional elements of this kind correspond to elements in the abstract syntax and do have a semantic interpretation.

A third kind of optionality occurs when semantic information may take more or less elaborate forms. Two examples of this are a) the occurrence of a ‘qualifier’ in dialogue act annotation to indicate certainty, conditionality or sentiment associated with a communicative function and b) the annotation of attribution and argument type for discourse relations. The latter example is briefly considered here; for the qualification of communicative functions, see 8.5.2.

In a preliminary study for developing an ISO standard for the annotation of discourse relations, where the CASCADES method is used to re-engineer the annotation scheme of the PDTB (see Reference [12]), the entity structures that annotate the arguments of discourse relations are defined as follows in the abstract syntax: *"An Argument Entity Structure, corresponding to an argument of a discourse relation, is a pair <m,s> consisting of a markable m and the semantic information s, which is either vacuous (i.e. the entity structure only identifies the markable corresponding to an argument of a discourse relation), or contains information about the attribution (atr) of the argument and/or s specifies the type (aot) of the argument. Formally, s is one of the following structures: s =<>; s =<atr>; s =<aot>; or s=<atr,aot>"*. This specification is a way of saying that the semantic information may include certain components, but does not have to. The optionality of attributions and argument types is thus formalized by allowing annotation structures with and without these elements. This form of optionality may be useful when dealing with information components that are not always applicable or which in some situations are irrelevant.

It is therefore possible to distinguish three types of optional element in annotation representations:

(1) those that are semantically insignificant, and can freely be included or left out of representations;

(2) those that have a default interpretation when not included in a representation;

(3) those that make representations more informative if included.

Elements that are not of one of these three types cannot be omitted from a representation without making it incomplete or 'underspecified', which may be acceptable for annotation tasks where the annotator does not always have sufficient information to make a complete annotation. This might lead to an ambiguous situation for optional elements of types (2) and (3); for such elements an "underspecified" value is strongly recommended so that it is possible to distinguish a situation in which an annotator is unable to assign a value from one in which the annotator intends the default value to be assigned [type (2) case] or where the element is considered to be inapplicable or irrelevant [type (3) case].

7 Overlaps between annotation schemes

7.1 Semantic and terminological consistency

The mutual consistency of two annotation schemes is in danger if the same or very closely related notions play a role in both schemes. This may give rise to a *semantic* inconsistency in the sense that the same concept is treated differently in the semantics of the two schemes. *Terminological* inconsistency arises if the two schemes use different terms for the same concept. If the same term is used in the two schemes with different meanings, there is both a semantic and a terminological inconsistency. This Clause identifies a number of cases where different parts of SemAF introduce overlapping annotation schemes, primarily in order to provide an agenda of issues regarding mutual consistency that need to be addressed in future SemAF parts or in future versions of existing parts.

7.2 Spatial and temporal relations as semantic roles

One of the overlaps between SemAF parts concerns the annotation of spatial and temporal relations. The annotation schemes of ISO-TimeML and ISOspace include the annotation of relations between events and their place and time of occurrence, as well as temporal relations between temporal entities and spatial relations between spatial entities. The annotation scheme of ISO 24617-4 ('SemAF-SR') for annotating semantic roles viewed as relations between events and their participants includes spatial and temporal participants. The relations defined in SemAF-SR for annotating semantic roles concerned with time and place also occur in ISOspace and ISO-TimeML.

SemAF-SR defines the following eight semantic roles of a spatial or temporal character:

a) location;

b) initial-location;

c) final-location;

d) path;

e) distance;

f) duration;

g) initial-time;

h) final-time.

These concepts also occur in ISOspace and in ISO-TimeML, sometimes using the same terminology, and this gives rise to both semantic and terminological inconsistencies. For example, ISOspace defines the concept 'path' as shown below (quoted from ISO 24617-7:2014, Clause 3) as a spatial entity like a road or a river, which can be used to get from one location to another.

3.13
path
series of **locations** (3.7)
NOTE 1 A spatial object **path** (3.13) is a location where the focus is on the potential for traversal, or which functions as a boundary. This includes common nouns like "*road*", "*coastline*" and "*river*" and proper names like "*Route 66*" and "*Kangamangus Highway*". Some nouns such as *valley* can be ambiguous: it can be understood as a **path** (3.13) in "*we walked down the valley*", or as a **place** (3.14) in *"we live in the valley".*

At this point, ISOspace is semantically inconsistent with SemAF-SR, which defines the term 'path' as dependent on the occurrence of an event as follows:

/path/	
Definition	Intermediate location or trajectory between two locations, or in a designated space, where an event occurs.
– Source	Adapted from Sowa [2000]
Example	'(The baby [agent e1]) crawled e1 (across the floor [path e1])'

The example sentence in this data category would be annotated as follows (in the XML representation used in the ISO 24617-4:2014, Annex C; markable m1 refers to "*The baby";* m2 to "*crawled*", and m3 to *"across the floor"*):

(9) The baby crawled across the floor.

```
<event xml:id="e1" target="#m2" eventFrame="crawl.01"/>
<entity xml:id="x1" target="#m1"/>
<srLink event="#e1" participant="#x1" semRole="agent"/>
<entity xml:id="x2" target="#m3"/>
<srLink event="#e1" participant="#x2" semRole="path"/>
```

So whereas 'path' is a spatial object in ISOspace, it is a relational notion in SemAF-SR.

ISOspace also defines the related concept 'event-path', 'which is the dynamic notion of a trajectory followed in a motion, like the trajectory that a frisbee follows through the air when it is thrown. This notion is defined in ISO 24617-7:2014 as follows.

3.3

event-path

path or trajectory followed by a spatial object coincident with a **motion-event** (3.9)

This notion is in essence the same as the semantic role 'Path' in SemAF-SR; it can be seen as a case of terminological inconsistency with different terms used for the same concept. On the other hand, there is a difference between the way ISOspace views an event-path and the way SemAF-SR views a *Path* role, since the latter is as a relation whereas the ISOspace notion is as an object; this is brought out by comparing the annotation representation (9) with the ISOspace example (10), taken from ISO 24617-7 (where the markable m1 refers to "*We camped*", m2 to "*three miles*", and m3 to "*the river*"):

(10) We camped [three miles$_{me2}$] from the [river$_{p1}$] [Ø$_{pl2}$]

```
<path xml:id="p1" markable="#m3"/>
<place xml:id="pl2" markable="□"/>
<measure xml:id="me2" markable="#m2" value="3" unit="miles"/>
<mlink xml:id="ml1" figure="#pl2" ground="#p1" trigger="#me2"
   relType="distance" val="#me2"/>
```

In (10), 'path' occurs as an object, whereas in (9) it occurs as a relation.

A similar difference can be found concerning the concept 'distance', which in ISOspace is a relation between (a) two spatial entities and (b) an extent of space [see (10)], whereas SemAF-SR uses Distance to indicate a semantic role, as illustrated in (11):

(11) missiles [pivot s1] (capable s1) of (travelling e1) (more than 300 km [distance e1])

Another case, which will be discussed in 8.2, is that of the semantic role *Amount* in SemAF-SR compared with the concept 'measure' as defined in ISOspace.

A general question that arises concerning the overlapping sets of concepts defined in ISOspace, ISO-TimeML and SemAF-SR is whether all the distinctions among spatial and temporal relations that are made in ISOspace and ISO-TimeML should be reflected in distinctions between semantic roles in SemAF-SR. For example, ISOspace uses the attribute ' " ', with possible values "true", "false" and "uncertain, in order to distinguish between cases like "*John arrived in Boston*", where John reached his destination, from "*John left for Boston*", where we do not know if he did. SemAF-SR has no provisions for making this distinction.

7.3 Events

In ISO-TimeML, which includes a typology of events (in a broad sense, including states and processes), events take centre-stage. Events are equally important in ISOspace, which inherits from ISO-TimeML the concept of an event plus everything that is said about events in ISO-TimeML annotations. Events are also of crucial importance in SemAF-SR, where semantic roles are viewed as relations between events and their participants, but SemAF-SR does not assume any particular typology of events. ISOspace, while explicitly inheriting the event concept from ISO-TimeML, makes a basic distinction between motion events and non-motion events that cuts through the ISO-TimeML typology. Whether this leads to consistency problems is not immediately clear.

The distinction between motion events and non-motion events seems to be relevant for semantic role assignment since only motion verbs have spatial entities in roles like '*Initial Location*', '*Path*' and '*Final Location'*. Motion verbs used in a negative sentence such as "*John did not leave home*" seem to require a different spatial role ('*Location'*?) for characterizing the relation between "*leave*" and "*home*"; it is not available in SemAF-SR. The same is true for "*John stayed at home*".

7.4 Discourse relations in dialogue

The semantic relations between sentences that lend coherence to a text also occur in dialogue. This is well known and has been the subject of a number of studies in the literature (see Reference [45], Reference [34] and Reference [28]); however, the overwhelming majority of studies of discourse relations have focused on text, rather than on interactive discourse.

The ISO 24617-2 annotation scheme for dialogue act annotation includes the concept of a 'rhetorical relation'; this is designed to be identical to that of semantic discourse relation as used in ISO 24617-8 ("SemAF DR-core"), but leaves open which specific set of relational concepts may be used (i.e. in XML representations of dialogue act annotations, the possible values of the attribute 'rhetoRel' are not specified). This is a good example of how the annotation schemes of different SemAF parts can be combined: once the annotation scheme for SemAF DR-core has been established, the set of discourse relations that it proposes can be taken as the possible values of the 'rhetoRel' attribute in dialogue act annotations.

Both spoken dialogue and written dialogue (e.g. chats on the internet) constitute particular forms of discourse in which utterances may also be related by relations other than the discourse relations that have been studied for written text. The ISO dialogue act annotation scheme distinguishes two other types of relation: a) feedback relations, in which a dialogue act provides or elicits feedback about the success of the interaction (such feedback acts are related to the preceding utterance(s) about whose processing they provide/elicit feedback) and b) dialogue acts that, due to their communicative function, are dependent for their semantic content on a preceding dialogue act (e.g. an answer being dependent on a question, an accept apology on an apology, and a reject offer on an offer). This relation is called a '*functional dependence relation*' while the relation between a feedback and its 'intecedent' is called a *'feedback dependence relation'*. These relations are not present in any existing annotation scheme for discourse relations, presumably because of their focus on written discourse. The ISO annotation scheme for discourse relations should inherit these from ISO 24617-2, not only because such relations are needed for the annotation of interactive discourse, but also since a written text such as a novel or a film script may well contain dialogues.

8 Semantic phenomena that cut across annotation schemes

8.1 Ubiquitous semantic phenomena

This Clause identifies a number of ubiquitous semantic phenomena that cut across the various SemAF-parts. The main purpose of this Clause is to provide an agenda for continued and future work within the framework of SemAF. For some of these phenomena, suggestions are provided for how they may be dealt with.

8.2 Quantification

It was noted in 3.2 that quantification cannot be disregarded when time and events are annotated and that ISO-TimeML therefore has some provisions, albeit very limited, for annotating quantification over time. The situation is very similar for the annotation of spatial information, as the following sentences illustrate.

(12) a. There's a computer <u>on every desk.</u>

b. We walked <u>through all the valleys.</u>

c. The paper must be <u>somewhere on my desk</u>.

Quantification phenomena arise whenever a predicate is applied not to a single individual, as in "*John played the piano*", but to one or more sets of individuals, as in "*Three men moved both pianos*". Quantification has been studied extensively, but not so much in relation to events, times and places. Nonetheless, in principle, *any* relation between two sets of entities is quantified, as are the relations between events and temporal entities, for instance by means of temporal quantifiers such as "*always*", "*sometimes*" and "*every Monday*". For this reason, ISO-TimeML has some provisions for annotating time-related quantification. The attribute 'quant' has been introduced for this purpose as one of the attributes of temporal entities. For example, a temporal quantifier like *"daily"* is represented as follows, where "P1D" stands for "period of one day":

```
<TIMEX3 xml:id="t5" target="#token" type="SET" value="P1D"
 quant="EVERY"/>
```

However, temporal quantification cannot be analysed in a satisfactory manner by means of attributes of temporal entities, since quantification phenomena are not properties of the entities participating in a predication, but are aspects of *relations*, as the following examples illustrate.

(13) a. These boxes are heavy.

b. All the students read two papers by Bertrand Russell.

c. Three men moved both pianos.

d. Everybody will die.

In (13a), a predicate is applied to a set of boxes and the question arises whether it applies to the individual members of that set ('distributive' reading) or to the set as a whole ('collective' reading). In (13b), a set of students and a set of papers are related by read-events and here, the well-known issue of relative scope arises: did all the students read the same two papers? In (13c), sets of men and pianos are related by move-events, and here the issues of 'distribution' (how is the predicate 'distributed' over the elements of a reference domain?) and relative scope arise. In (13d), a set of die-events and a set of people are related and, in a not-so-obvious way, the issue of scope arises: does the sentence mean that everyone will die in the same apocalyptic event?

Reference [7] and Reference [8] proposed an analysis of quantification in terms of feature structures which can be the basis for annotating quantification in such a way that components of annotation structures by and large correspond to the linguistic expression of quantification. This is convenient for annotators and supports a semantic interpretation that can be combined with a compositional semantics of noun phrases, which itself is useful since many of the features of quantifications are expressed syntactically within noun phrases. This approach is outlined in Annex A.

8.3 Quantities and measures

Duration, length, volume, weight, price and many other ways of measuring quantities (or 'amounts') of something are linguistically expressed by means of a unit of measurement plus a numerical indication, such as "*one and a half hours*", "*90 minutes*", "*just over two kilos*". From a semantic point of view, a measure is an equivalence class formed by pairs <*n,u*> where *n* is a numerical predicate and *u* is a unit. [2] Given the relations between the units in a particular system of units, any of the equivalent pairs can serve as a representative of the equivalence class. For instance, <*1.5, hour*> represents the same amount of time as <*90,minute*>; they belong to the same equivalence class since 1 h = 60 min.

Units can be complex, like 'kilowatt-hour' (unit of electrical energy) or 'meter/second' (unit of velocity). Formally, a unit is either a basic unit or a triple <*u1, u2, Q*> where *Q* = × (multiplication) or *Q* = / (division) and *u1* and *u2* are (possibly complex) units. This allows for complex units such as *meter/(second × second)* (meter per square second) for measuring acceleration, and *euro per (meter × meter)* (euro per square meter) for measuring the price of land.

The abstract syntax of annotations for quantities can be defined by introducing pairs <*n,u*>, where *'u'* is either an elementary unit or a triple, as indicated above. A corresponding XML-based concrete syntax uses an element 'amount' with attribute, value pairs for the numerical part and the unit part, as in (14) (where markable m1 refers to "*three miles*").

(14) a. three miles

b. `<amount xml:id="am1" target="#m1" num="3" unit="mile"/>`

ISOspace includes amounts (called 'measures') of space for measuring distances; likewise, ISO-TimeML includes amounts of time for measuring durations. In both cases, only elementary units are considered; the above abstract syntax for complex units, which generalizes this simple notion of amount, would correspond to the introduction of unit elements in the concrete syntax, as in (15) (where the markable m1 in (15b) refers to *"sixty miles per hour*" and m2 to "*miles per hour*").

(15) a. sixty miles per hour

b.
```
<amount xml:id="am1" target="#m1" num="60" unit="#u1"/>

<unit xml:id="u1" target="#m2" unit1="mile" unit2="hour"
 operation="division"/>
```

Amount expressions involving comparisons, like *"more than five miles*" in *"We walked more than five miles"*, may be treated as involving an existential quantification over locations, as in: "We walked a distance greater than five miles", that is to say, there is an amount of space greater than five miles, and that is the distance that we walked (where m1 refers to "*We",* m2 to *"walked",* m3 to *"more than five miles",* and m4 to *"five miles*"):

(16) a. We walked more than five milesß

b.

```
<event xml:id="e1" target="#m2" pred="walk"/>
  <entity xml:id="x1" target="#m1"/>
  <srLxink event="#e1" participant="#x1" roleType="agent"/>
  <amount xml:id="d1" target="#m3"/>
  <amount xml:id="d2" target="#m4" num="5" unit="mile"/>
  <relation arg1="#d1" arg2="#d2" relType="greaterThan"/>
  <srLink event="#e1" participant="#d1" roleType="distance"/>
```

8.4 Negation, modality, factuality, and attribution

Negation, modality, factuality and attribution are different, but related, aspects of the factual content of an utterance or a text. The following example comes from the Penn Discourse Treebank:[36]

(17) *"The public is buying the market when in reality there is plenty of grain to be shipped,*" said Bill Biederman, Allendale Inc. director.

Even though Biedermann says *"in reality...",* it would be incorrect to conclude from this text that there is plenty of grain to be shipped. It is essential to take account of the source to which a statement is attributed; if, for example, the *Wall Street Journal* were to report (rather than quote somebody as saying) that there was plenty of grain to be shipped, it would perhaps be more justified to draw that conclusion.

It is also clear that negations have a strong influence on which information can be extracted from a text. ISO-TimeML makes use of an attribute 'polarity', with possible values "positive" and "negative", as one of the attributes of an event. However, positive and negative are just two extremes or a scale of possibilities, as the following examples illustrate:

(18) a. There's plenty of grain to be shipped.

b. There's not nearly enough grain to be shipped.

c. There's probably enough grain to be shipped.

d. There's probably not enough grain to be shipped.

e. There may not be enough grain to be shipped.

f. According to Bill Biederman, there's surely enough grain to be shipped.

Sentences (18c) to (18f) make claims of varying strength about whether there is enough grain to be shipped, with a strength which lies in between the extremes indicated in (18a) and (18b). Modalities as expressed by "*probably*", "*maybe*" and "*surely*", as well as the attribution of the claim to a certain source, all have an influence on the possibilities of extracting information from a text. Expressions of modality, including degrees of certainty and possibility, have been studied, e.g. in Reference [25] and Reference [26]. The factuality of statements about events has been studied in Reference [42] and annotated in the FactBank.[43]

Negations and modalities are important in relation not only to events, but also in relation to other types of entity (e.g. an amount of time as in *"I'll wait <u>no more than 10 minutes</u>"*), amounts in other dimensions, numerical quantities *("not less than five")* and properties such as *"hardly surprising"* and *"probably made in China"*. See also Reference [32] and Reference [33] for work on the annotation of negation, modality and attribution.

Epistemic modalities as expressed by "*possibly*", "*probably*" and "*maybe*" are also relevant to take into account in dialogue act annotation, as the following example illustrates:

(19) 1. A: Would you like to have some coffee?

2.a B: Maybe; how much time do we have?

2.b B: Not sure I want any.

In this case, the epistemic qualifiers "*maybe*" and "*not sure*" do not relate to the truth of a statement, but to modalities of accepting an offer. This is taken up in 8.5.2.

8.5 Modification and qualification

8.5.1 Modification and quantification

The modification of nominal expressions, for example, by adjectives, prepositional phrases or relative clauses, gives rise to many of the same issues as the expression of quantification; in particular, issues of scope and distribution arise in much the same way. In the following example, the text should be visualized next to a box of bell peppers.

(20) Bell peppers for fifty pesos

This expression is ambiguous as to whether the preposition phrase *"for fifty pesos"* is used to indicate that the bell peppers in the box cost 50 pesos apiece (individual reading) or that the whole content of the box costs 50 pesos (collective reading). Similarly, the sentence in (21) below can express that each of the boxes in question is heavy (individual reading) or that they are heavy together (collective reading).

(21) Let me help you with those heavy boxes.

More generally, adjectives and prepositional phrases, when used as modifiers, can be viewed as one-place predicates. The application of such a predicate to a set of arguments gives rise to quantificational issues, as noted in 8.2.

The ambiguity of (20) and (21) is not due to an ambiguity in the predicate, but to the way the predicate is applied to its arguments. This suggests an approach to the annotation of modification similar to that of the predicative use of an adjective, as in "These books are heavy", with a 'restrictive modifier' formed by being the Theme of a state of being heavy. Such a structure allows the distribution of the modification to be a property of the Theme. In a semantic role link in an XML representation, such an annotation could look like this:

(22) a. heavy boxes

b.

```
    <entity id="x1" target="#m2" signature="set"/>
    <property id="p1" target="#m1 />
    <modLink id="m1" head="#x1" modifier="#p1" relType="restrictiveModifier"
distribution="individual"/>
```

Modification by means of relative clauses gives rise to all the issues that are known to arise in quantifications; this can be seen by transforming a quantified sentence into a modified noun phrase, as illustrated in the sentence pairs (23) and (24):

(23) a. That crane moved thirty big pipes.

b. Thirty big pipes moved by that crane.

(24) a. Two students read the six papers.

b. The six papers read by two students.

Sentence (23a) is ambiguous in the distributive aspect of the quantification in *"thirty big pipes"*, that is to say, did the crane move the pipes one by one (distributive reading) or all in one go (collective reading), or in smaller groups ('subcollective' reading, see Reference [2])? Sentence (23b) has the same ambiguity: does the modification "*moved by the crane*" apply collectively to the set of 30 pipes, or to individual pipes, or to subcollections of pipes? While less conspicuous, the ambiguity is there, since the sentence might be intended to refer to that particular set of 30 pipes of which the members were moved one by one, rather than to another set of 30 pipes, which were moved in one go. Similarly, sentence (24b) has the same ambiguity as the quantifying sentence (24a) with respect to the scopes of the quantifications over students and papers: did each of the students read six papers or were the same six papers read by each of the two students?.

In view of the analogy between modification and quantification, the development of an approach to the annotation of modification integrated with that of quantification is recommended (see Annex A).

8.5.2 Qualification

The notion of a 'qualifier' has been introduced in ISO 24617-2 for dialogue act annotation in order to make distinctions between dialogue act types that are subtler than would be possible by simply using the set of 57 communicative functions defined in the annotation scheme. Although this set is fairly comprehensive and richer than that of other dialogue act annotation schemes, it is not sufficient for dealing in a satisfactory way with subtle differences like those illustrated in the following examples of responding to an offer.

(25) 1. A: Would you like to have some coffee?

2.a. B: Only if you have it ready.

b. B: Maybe; how much time do we have?

c. B: Maybe later

d. B: Coffee, wonderful!

e. B: Coffee? At midnight??

These examples show the conditional acceptance of an offer (2a); an uncertain acceptance (2b); an uncertain rejection (2c); acceptance with pleasure (2d); and rejection with surprise (2e). In all these cases, it would be unsatisfactory to simply characterize the dialogue act as either acceptance of rejection of an offer. In order to take such modalities into account, which can occur with every dialogue act that has a responsive character, Petukhova and Bunt (2010) have proposed the use of 'qualifiers' for certainty, conditionality and sentiment. These are optional elements in the abstract syntax of dialogue act annotations; this means that they do not have to be used in dialogue act markup, but if they are, they have a semantic interpretation.

Note that qualifiers in natural dialogue are often expressed non-verbally, by means of facial expressions or gestures, or multimodally by a facial expression and/or a gesture in combination with a verbal expression or a nonverbal sound (e.g. a laugh, a cough or a sigh).

Qualifiers may be an interesting addition in other SemAF-parts as well, such as in the annotation of semantic roles. Similar to the situation with dialogue act types, the use of a fixed, limited number of semantic roles is bound to run into problems in the analysis of a sentence where a participant occurs for which none of the semantic roles is fully satisfactory. For example, the *Agent* role is defined in SemAF-SR as the involvement of a participant who acts "intentionally or consciously", so when a sentence like (23) is annotated, the question arises whether Peter dropped his plate intentionally.

(26) Peter dropped his plate on the kitchen floor.

If Peter dropped his plate intentionally, for instance, in order to express anger, or disgust at its content, it would be informative to make that explicit in the annotation, but the annotation scheme has no provisions for that. The introduction of a qualifier to express intentionality would be useful here.

For the annotation of texts with discourse relations, similar considerations apply: the annotation scheme will have a limited number of relations and these are bound to be inadequate in some cases. For example, the discourse connective "*but*" expresses some kind of opposition. When explicitly qualified, as in "*but unexpectedly*", a qualifier could be useful to indicate that in this case, the opposition is between what is described in the subsequent text and what was expected to happen. Similar to the annotation of communicative functions, discourse relations expressed by "*but perhaps*" or "*but fortunately*" could benefit from the use of certainty and sentiment qualifiers, in order to annotate not just that there is an opposition but also that the speaker felt uncertain about the opposition or felt good about what happened, contrary to expectation. In such a way, more subtle distinctions can be annotated than when simply using the inventory of basic discourse relations.

8.5.3 Other issues

The set of issues discussed in the preceding subclauses is clearly not exhaustive. Other semantic phenomena that cut across SemAF parts include the following.

— **Ellipsis**: The resolution of ellipsis is, for example, relevant when identifying the participants in an eventuality and therefore for the correct assignment of semantic roles.

— **Coreference**: Interpretation of personal pronouns is, for example, relevant for the correct assignment of semantic role annotations to the arguments of a verb. Reference [30] contains the basis for a proposal to initiate a new SemAF-part for developing an ISO standard for the annotation of coreference.

— Forms of **non-literal language use**, such as metonymy, irony, sarcasm and metaphor.

In order for SemAF to develop into a wide-coverage framework for semantic annotation, especially the annotation of ellipsis and coreference are of primary importance.

Annex A
(informative)

An approach to the annotation of quantification in natural language

Quantification in natural language is the name given to a number of semantic phenomena that emerge when a predicate is applied to one or more sets or arguments. Consider, for example, the sentence (A1) and some of the questions that may be asked (and answered) about its meaning.

(A1) Three men moved both pianos.

a. How many men were involved? (Answer: Three.)

b. How many pianos were involved? (Answer: Two.)

c. Did the same men move both pianos? (Answer: Yes.)

d. Did the men act collectively or individually? (Answer: Collectively, probably.)

e. Were the pianos acted upon collectively or individually? (Answer: Individually, probably.)

The answers to questions like (A1a) and (A1b) quantify the *degree of involvement* of the sets of men and pianos that are related by the 'move' predicate. These two sets are denoted by the *restriction* part of the quantifying expressions: 'men' is the restriction in 'three men', and 'pianos' is the restriction part in 'both pianos'. These sets are called the *reference domains* of the quantifications; the quantifiers "three" and "both" count the number of elements of the reference domains involved in the predication. Question (A1c) concerns the relative *scopes* of the quantifications over the sets of men and pianos. Questions (A1d) and (A1e) are about the way in which the elements of the two reference domains are involved in the predication: individually or collectively (or in groups). This aspect, called the *distribution* of a quantified predication, determines the type of argument to which the predicate is applied; in the case of example (A1), sets of men and individual pianos. The sets of possible arguments are called *predication domains*. In the case of individual distribution, the predication domain is identical to the reference domain; in the case of collective distribution, the predication domain consists of sets of elements from the reference domain.

Proportional quantifiers like "*all*", "*some*" and "*most*" specify the *fraction of the reference domain* involved in a predication; numerical or amount quantifiers like "*three*", "*more than five*" and "*2 litres of*" may also have this function, but may alternatively also be used to indicate the *size of a reference domain*, like "*twelve*" in (A2).

(A2) The twelve students in this room all speak two languages.

Proportional and absolute quantifiers can also be used to indicate the *number/amount of a predication domain per element of another predication domain*, like *five* in (A3).

(A3) Each of the dogs had five sausages.

The examples in (A1) to (A3) illustrate some of the most important aspects of quantification.[2]

(A3) Aspects of quantification

a) quantifier's restriction, describing the reference domain of the predication;

b) distribution, defining the predication domain;

c) size of the reference domain;

d) involvement of the reference domain (in absolute or relative terms);

e) relative scoping of the quantifications associated with predicate arguments;

f) scoping of argument quantifications relative to quantified events (corresponding to predicates);

g) size of groups of elements from a reference domain;

h) number of elements of a reference domain involved per element of a predication domain.

In a neo-Davidsonian event-based approach to meaning, the combination of a verb with its arguments may be seen to be introducing a set of events (in a broad sense, including states and processes) of the type indicated by the verb and with a number of properties concerning the way in which the participants of these events are involved. The application of this analysis to sentence (A1) produces a description in terms of three sets and the relations between them: a set of move-events, a set of men participating in these events, and two pianos that participate in the events. The way these three sets are related can be analysed by means of binary relations between events and participants that characterize the way a participant is involved in an event, relations called ‘semantic roles’, whose annotation is the subject of ISO 24617-4 (SemAF-SR; see also Reference [48]).

Using this approach, quantification as it occurs in sentence (A1) is analysed not as the result of the ‘move’ predicate applied to two reference domains, but as the result of predicates corresponding to the *Agent* and *Theme* semantic roles that relate respectively a set of ‘move’ events to a set of men and to a set of pianos. Following suggestions in References [13] and [31], the semantic information carried by the quantifications should then be represented as properties of link structures that correspond to semantic roles, as defined in SemAF-SR. This is illustrated in (A5).

(A5) Annotation of (A1), represented in XML:

[v1] <event xml:id=”e1” target=”#m2” pred=”move” signature=”set”/>

[p1] <participant xml:id=”x1” target=”#m1” refDomain=”man” signature=”set” involvement=”3”/>

[p2] <participant xml:id=”x2” target=”#m3” refDomain=”piano” signature=”set” definiteness=”def” involvement=”2”/>

[L1] <srLink event=”#e1” participant=”#x1” semRole=”agent” distribution=”collective”/>

[L2] <srLink event=”#e1” participant=”#x2” semRole=”theme” distribution=”individual”/>

The feature `signature=”set”` in (A5) expresses that the sentence is about sets of events and participants; labels have been attached to parts of the XML representation in order to facilitate the description of their semantics (see below).

Reference [7] and Reference [8] show that attributes can be defined for events and their participants in such a way that a wide range of forms of quantification can be correctly annotated; these include collective, distributive and group quantifications; scoped, unscoped and partially-scoped quantification; and cumulative quantification (when two quantifications have equal scope). The following three examples, (A6) to (A8), illustrate the wide coverage of the approach.

(A6) EXAMPLE 1 Wide-scope interpretation of event over its arguments

Everybody will die.

[v1] <event xml:id=”e1” target=”#m1” pred=”die” signature=”set” cardinality=”1”/>

[p1] <participant xml:id=”x1” target=”#m2” refDomain=”human” signature=”set” involvement=”all”/>

[L1] <srLink event=”#e1” participant=”x1” semRole=”theme” eventScope=”wide”/>

(A7) EXAMPLE 2: Relative scoping of verb arguments: interpretation of wide-scope “two papers”.

All the students read two papers

[v1] <event xml:id=”e1” target=”#m1” pred=”read” signature=”set”/>

[p1] <participant xml:id=”x1” target=”#m2” definiteness=”def” refDom=”student” signature=”set” involvement=”all”/>

[p2] <participant xml:id=”x2” target=”#m3” refDomain=”paper” signature=”set” cardinality=”2” involvement=”all”/>

[L1] <srLink event=”#e1” participant=”x1” semRole=”agent” distribution=”individual”/>

[L2] <srLink event=”#e1” participant=”x2” semRole=”theme” distribution=”individual”/>

[s1] <participantScopes outScoping=”#x2 #x1”/>

Not only scoped quantifications have to be considered, but also unscoped, partially scoped and equally scoped quantifications. Partially scoped and unscoped cases, where there is no or incomplete information about relative scoping, are easily annotated by not specifying values for the ‘outScoping’ attribute in the ‘participantScopes’ element, and interpreted with underspecified DRSs (using the techniques described in Reference [49] and Reference [50]). Equally scoped quantifications, as they occur in cumulative quantification[51] and in group quantification,[2] can be annotated using the attribute ‘eqScope’, as shown in (A8).

(A8) EXAMPLE 3 Cumulative reading of the following sentence (from Reference [50]). On this reading, a total of three breweries supplied in total of 15 inns.

Three breweries supplied 15 inns.

[v1] <event xml:id=”e1” target=”#m2” refDomain=”supply” signature=”set”/>

[p1] <participant xml:id=”x1” target=”#m1” refDomain=”brewery” signature=”set” involvement=”3”/>

[p2] <participant xml:id=”x2” target=”#m3” refDomain=”inn” signature=”set” involvement=”15”/>

[L1] <srLink event=”#e1” participant=”#x1” semRole=”agent” distribution=”individual”/>

[L2] <srLink event=”#e1” participant=”#x2” semRole=”theme” distribution=”individual”/>

[s1] <participantScopes eqScope=”#x1 #x2”/>

Of the information types listed in (A3), those numbered 1, 2, 4, 5, 6, and 8 can be represented by the attributes and values as shown in the above examples. For dealing with information of the types 3 and 7 the attribute ‘cardinality’ is defined for <event> (e.g. for “say twice”) and <participant> elements, and the values of another attribute, ‘groupCard’ can be used to indicate group sizes in subcollective quantifications. This provides the expressive power to represent a wide range of quantification phenomena.

In the approach outlined here, some of the properties of a quantification are annotated as parts of semantic role links, where traditionally, the semantic representation of quantification is considered to be part of noun phrase semantics. This is particularly true of distribution. Having ‘distribution’ as an attribute of <participant> elements, corresponding to noun phrases, would run into problems for a sentence such as (A9), where the subject noun phrase (“*The men*”) should be interpreted individually for the drinking, but collectively for moving the piano.

(A9) The men had a beer before they moved the piano.

The semantic adequacy of the proposed annotation format can be shown by defining a systematic translation of annotations into discourse representation structures (DRSs). XML elements describing sets of events or participants are translated to a DRS that introduces a higher-order (i.e. set-valued) discourse referent (the presentation is somewhat simplified here; see Reference [9] for the use of pairs consisting of a markable and a discourse referent, where the markables ensure that only the intended referent variables are unified upon DRS-merging), and DRS conditions that translate the other features. DRSs that interpret linking elements introduce both discourse referents for the linked sets of elements and conditions that further characterize the link; for example, for the annotation of sentence (A1) the DRS-interpretation is constructed as follows:

(10) v1’ = <{E1}, {<{e}, {e ∈ E1}>⇒<{}, {move(e)}>}>

p1’ = <{X}, {card(X)=3,<{x}, {x ∈ X}>⇒<{}, {man(x)}>}>

p2’ = <{Y}, {card(Y)=2,<{y}, {y ∈ Y}>⇒<{}, {piano(y)}>}>

L1’ = <{E,X}, {<{e}. {e ∈ E}>⇒<{}, {agent(e,X)}>}>

L2’ = <{E,Y}, {<{y}. {y ∈ Y}>⇒<{e’}, {e’ ∈ E, theme(e’,y)}>}>

The merge of these DRSs yields the correct end result:

(11) (p1’ ∪ L1’) ∪ (p2’ ∪ L2’) ∪ v1’=<{E,X,Y}, {card)(X)=3, card(Y)=2,

<{x}, {x ∈ X}>⇒<{}, {man(x)}>,

<{e}, {e ∈ E}>⇒<{}, {move(e), agent(e,X)}>,

<{y}, {y ∈ Y}>⇒<{e’}, { piano(y), e’ ∈ E, theme(e’,y)}>}>

Bibliography

[1] Asher N. Reference to abstract objects in discourse. Kluwer Academic, Dordrecht, 1993

[2] Bunt H. Mass terms and model-theoretic semantics. Cambridge University Press, 1985

[3] Bunt H. Quantification and modification represented as feature structures. In *Proceedings 6th International Workshop on Computational Semantics (IWCS-6),*Tilburg, 2005, pp. 54-65

[4] Bunt H. The Semantics of Semantic Annotation. In *Proceedings 21st Pacific Asia Conference on Language, Information and Computation (PACLIC-21)*, Seoul, 2007, pp. 13-28

[5] Bunt H. A methodology for designing semantic annotation languages. In *Proceedings 2nd International Conference on Global Interoperability for Language Resources (ICGL 2010),* Hong Kong, 2010, p. 29-46

[6] Bunt H. Abstract syntax and semantics in semantic annotation, applied to time and events. In: Recent Trends in Language and Knowledge Processing, (Lee E., & Yoon A. eds.). Hankukmunhwasa, Seoul, 2011, pp. 157–204

[7] Bunt H. A methodology for designing semantic annotations. TiCC Technical Report TR 2013-001. Tilburg Center for Cognition and Communication, Tilburg University, 2013a

[8] Bunt H. The annotation of quantification and its interpretation. In *Proceedings 10th International Workshop on Computational Semantics (IWCS-10)*, Potsdam, 2013b, pp. 307-313

[9] Bunt H. Annotations that Effectively Contribute to Semantic Interpretation. In: Computing Meaning, (Bunt H., Bos J., Pulman S. eds.). Springer, Dordrecht, **Vol. 4**, 2014, pp. 49–70

[10] Bunt H., & Overbeeke C. 2008a) Towards formal interpretation of semantic annotation. In *Proceedings 6th International Conference on Language Resources and Systems (LREC 2008)*, Marrakech

[11] Bunt H., & Overbeeke C. 2008b) An extensible, compositional semantics of temporal annotation. In *Proceedings 2nd Lingusitic Annotation Workshop (LAW II). Marrakech, pp. 26-35*

[12] Bunt H., Prasad R., Joshi A. First steps toward an ISO standard for the annotation of discourse relations. In *Proceedings 7th Joint ACL-ISO Workshop on Interoperable Semantic Annotation (ISA-7),* Istanbul. ELRA, Paris, 2012, pp. 60-69

[13] Bunt H., & Pustejovsky J. Annotating temporal and event quantification. In *Proceedings 5th Joint ACL-ISO Workshop on Interoperable Semantic Annotation (ISA-5),* City University of Hong Kong, 2010, pp. 15-22

[14] Bunt H., & Romary L. Towards multimodal content representation. In *Proceedings LREC 2002 Workshop on International Standards of Terminology and Language Resources,* Las Palmas, ELRA, Paris, 2002, pp. 54-60

[15] Carletta J., Ashby S., Bourban S., Flynn M., Guillemot M., Hain T. The AMI Meeting Corpus: A pre-announcement. In S. RENGALS and S. BENGIO (eds.) MLMI'05: Proceedings Workshop on Machine Learning for Multimodal Interaction. LNCS 3869. Springer, Berlin, 2006, pp. 28-39

[16] Hovy E., & Maier E. Parsimonious or profligate: how many and which discourse structure relations? ISI research report. Information Sciences Institute, University of Southern California, Marina del Rey, 1992

[17] Ide N., & Bunt H. 2010) Anatomy of Annotation Schemes: Mapping to GrAF. In *Proceedings 4th Linguistic Annotation Workshop (LAW IV)*, Uppsala, pp. 115–124

[18] Ide N., & Romary L. International Standard for a Linguistic Annotation Framework. Natural Language Engineering, 2004, **10**, 221–225

[19] ISO 24617-1:2012, *Language resource management — Semantic annotation framework (SemAF) — Part 1: Time and events (SemAF-Time, ISO-TimeML)*

[20] ISO 24617-2:2012, *Language resource management — Semantic annotation framework (SemAF) — Part 2: Dialogue acts*

[21] ISO 24617-4:2014, *Language resource management — Semantic annotation framework (SemAF) — Part 4: Semantic roles (SemAF-SR)*

[22] ISO 24617-7:2014, *Language resource management — Semantic annotation framework — Part 7: Spatial information (ISOspace)*

[23] ISO 24610-8:—[1)], *Language resource management — Semantic annotation framework (SemAF) — Part 8: Semantic relations in discourse (ISO DR-core)*

[24] Kamp H., & Reyle U. From Discourse to Logic. Springer, Berlin, 1993

[25] Karttunen L. Implicative verbs. Language. 1971, **47**, 340–358

[26] Karttunen L. Simple and phrasal implicatives. In: SEM 2012. Association for Computational Linguistics, Montreal, 2012, pp. 124–31.

[27] Katz G. Towards a denotational semantics for TimeML. In: Annotation, extraction, and reasoning about time and events, (Schilder F., Katz G., Pustejovsky J. eds.). Springer, Dordrecht, 2007, pp. 88-106

[28] Lascarides A., & Asher N. Segmented Discourse Representation Theory: Dynamic Semantics with Discoure Structure. In: Computing Meaning, (Bunt H., & Muskens R. eds.). Springer, Dordrecht, **Vol. 3**, 2007, pp. 87–124.

[29] Lee K. Formal semantics for interpreting temporal annotation. In Unity and Diversity of Languages, P. VAN STENKERBURG, ed. Benjamins, Amsterdam, 2008, pp. 97-108

[30] Lee K. Semantic Annotation of Anaphoric Links in Language. In *Proceedings 10th Joint ACL – ISO Workshop on Interoperable Semantic Annotation (ISA-10)*, Reykjavik. ELRA, Paris, 2014, pp. 29-38

[31] Lee K., & Bunt H. Counting time and events. In *Proceedings 8th Joint ACL – ISO Workshop on Interoperable Semantic Annotation (ISA-8, "ISA in Pisa"),* Pisa, 2012, pp. 26-33

[32] Morante R., & Daelemans W. Annotating Modality and Negation for a Machine Reading Evaluation. In *CLEF 2011 Labs and Workshop, Notebook Papers,* 2011

[33] Pareti S. 2012) The independent encoding of attribution relations. In *Proceedings 8th Joint ACL – ISO Workshop on Interoperable Semantic Annotation (ISA-8, "ISA in Pisa")*, Pisa, pp. 48-55

[34] Petukhova V., Prévot L., Bunt H. 2011) Discourse relations in dialogue. In *Proceedings 6th Joint ACL – ISO Workshop on Interoperable Semantic Annotation (ISA-6)*, Oxford, pp. 18-27

[35] Pratt-Hartmann I. 2007) From TimeML to Interval Temporal Logic. In *Proceedings 7th International Workshop on Computational Semantics (IWCS-7*), Tilburg, pp. 166-180

[36] Prasad R., Dinesh N., Lee A., Miltsakaki E., Robaldo L., Joshi A. 2008) The Penn Discourse TreeBank 2.0. In *Proceedings 6th International Conference on Language Resources and Systems (LREC 2008)*, Marrakech

[37] Pustejovsky J., & Stubbs A. Natural Language Annotation for Machine Learning. O'Reilly, Sebastopol, California, 2012

1) To be published.

[38] Pustejovsky J., & Moszkowicz J. 2012) The Role of Model Testing in Standards Development: The Case of ISO-Space. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC 2012)*, Istanbul

[39] Pustejovsjy J., Castano J., Ingria R., Gauzauskas R., Katz G., Sauri R. 2003) TimeML: Robust specification of event and temporal expressions in text. In *Proceedings 5th International Workshop on Computational Semantics (IWCS-5),* Tilburg, pp. 336-353

[40] Reyle U. Dealing with ambiguities by underspecification. Journal of Semantics. 1993, **10** (2) pp. 123–179

[41] Romary L., Budin G., Bunt H., Calzolari N., Declerck T., Francopoulo G. 2007) LIRICS Project Final Public Report. Deliverable D7.7 of eContent project LIRICS. Available from http://lirics.loria.fr

[42] Sauri R. 2008) A Factuality Profiler for Eventualities in Text. Ph.D. Thesis, Brandeis University

[43] Sauri R., & Pustejovsky J. FactBank: a corpus annotated with event factuality. Language, Resources and Evaluation, 2009, **43** (3), 227–268

[44] TEI. 2010) *TEI P5: Guidelines for Electronic Text Encoding and Interchange*. Text Encoding Initiative Consortium. Available at http://www.tei-c.org/Guidelines/P5/

[45] Tonelli S., Riccardi G., Prasad R., Joshi A. 2010) Annotation of Discourse Relations for Conversational Spoken Dialogs. In *Proceedings 7th International Conference on Language Resources and Systems (LREC 2010)*, Genoa

[46] ISO 12620, *Terminology and other language and content resources — Specification of data categories and management of a Data Category Registry for language resources*

[47] ISO 24612, *Language resource management — Linguistic annotation framework (LAF)*

[48] Bunt H., & Palmer M. Conceptual and Representational Choices in Defining an ISO standard for Semantic Role Annotation. In Proceedings Ninth Joint ISO - ACL SIGSEM Workshop on Interoperable Semantic Annotation (ISA-9), Potsdam, March 19-20, 2013, pp. 41-50.

[49] Reyle U. Dealing with Ambiguities by Underspecification. Journal of Semantics. 1993, **10** (2), 123–179

[50] Reyle U. Monotonic Disambiguation and Plural Pronoun Resolution. In Hans Kamp, ed.,Ellipsis, Tense and Questions, DYANA-2 Deliverable R2.2.B, ILLC/Department of Philosophy, University of Amsterdam, 1994, pp. 333- 364.

[51] Scha R. Distributive, Collective and Cumulative Quantification. In: Formal Methods in the Study of Language, (Groenendijk J., & Stokhof M. eds.). Mathematical Center, Amsterdam, 1981

ICS 01.020

Price based on 30 pages