BS ISO 24617-1:2012

BSI Standards Publication

Language resource management — Semantic annotation framework (SemAF)

Part 1: Time and events (SemAF-Time, ISO-TimeML)

bsi.

...making excellence a habit.™

Document Sharing Hub

National foreword

This British Standard is the UK implementation of ISO 24617-1:2012.

The UK participation in its preparation was entrusted to Technical Committee TS/1, Terminology.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

© The British Standards Institution 2013.
Published by BSI Standards Limited 2013.

ISBN 978 0 580 64814 4

ICS 01.020

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 January 2013.

Amendments issued since publication

Date	Text affected

Documents
Document Sharing Hub

Full document details: BS ISO 24617-1:2012

INTERNATIONAL STANDARD

ISO 24617-1

First edition
2012-01-15

Language resource management — Semantic annotation framework (SemAF) —

Part 1: Time and events (SemAF-Time, ISO-TimeML)

Gestion des ressources langagières — Cadre d'annotation sémantique (SemAF) —

Partie 1: Temps et événements (SemAF-Time, ISO-TimeML)

Reference number
ISO 24617-1:2012(E)

Documents
Document Sharing Hub

© ISO 2012

Full document details: BS ISO 24617-1:2012

COPYRIGHT PROTECTED DOCUMENT

© ISO 2012

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Documents
Document Sharing Hub
© ISO 2012 – All rights reserved

Full document details: BS ISO 24617-1:2012

Contents

Page

Foreword vi

Introduction vii

1 Scope 1

2 Normative references 1

3 Terms and definitions 1

4 Overview 4

5 Motivation and requirements 4

6 Basic concepts and metamodel 5

7 Specification of ISO-TimeML 8
7.1 Overview 8
7.2 Abstract syntax 8
7.2.1 Introduction 8
7.2.2 Conceptual inventory 9
7.2.3 Syntax rules 9
7.3 Concrete XML-based syntax 10
7.3.1 TimeML vs. ISO-TimeML: Stand-off annotation and other differences 10
7.3.2 Naming conventions 12
7.3.3 Example annotations 12
7.3.4 Basic elements: <EVENT>, <TIMEX3>, and <SIGNAL> 12
7.3.5 Link elements: <TLINK>, <SLINK>, <ALINK> and <MLINK> 18
7.3.6 Other tags: <CONFIDENCE>, <CERTAINTY> and <ISO-TimeML> 22

8 Towards a semantics for ISO-TimeML 26
8.1 Overview 26
8.2 Tense and aspect in language 26
8.2.1 Tense 26
8.2.2 Aspect 26
8.3 Temporal relations 27
8.4 An interval-based semantics for ISO-TimeML 28
8.4.1 Technical preliminaries for interval temporal logic 28
8.4.2 Basic event-structure 29
8.4.3 The interpretation of <TIMEX3> 31
8.4.4 Interpretive rule summary 36
8.5 An event-based semantics for ISO-TimeML 37
8.5.1 Introduction 37
8.5.2 Defining an event-based semantics 38

Annex A (normative) **Core annotation guidelines** 41
A.1 Introduction 41
A.2 ISO-TimeML elements and their attributes 41
A.2.1 The <EVENT> element 41
A.2.2 The <TIMEX3> element 48
A.2.3 The <SIGNAL> element 55
A.3 The link elements: <TLINK>, <SLINK>, <ALINK> and <MLINK> 56
A.3.1 Overview 56
A.3.2 The <TLINK> element 56
A.3.3 The <SLINK> element 59
A.3.4 The <ALINK> element 61
A.3.5 The <MLINK> element 62

Documents
Document Sharing Hub

© ISO 2012 – All rights reserved

Full document details: BS ISO 24617-1:2012

Annex B (informative) **Completely annotated examples** 63
B.1 Complex TIMEX3 examples 63
B.2 Complex TLINK and SLINK examples 64
B.3 Causative examples 67

Annex C (informative) **Event and temporal annotations for Chinese** 68

Annex D (informative) **Annotation for Italian fragment** 74
D.1 Introduction 74
D.2 Basic references 74
D.3 ISO-TimeML elements and their attributes 74
D.3.1 How to annotate EVENTs 74
D.3.2 Event identification and tag span 75
D.3.3 What NOT to tag 78
D.3.4 Introductory note 78
D.4 The <SIGNAL> element 81
D.5 The link tags 82
D.6 Informative: Examples of tense, aspect and mood annotation in Italian 82
D.7 Sample of Italian annotation 84

Annex E (informative) **Temporal annotation of predicates in Korean** 89
E.1 Introduction 89
E.2 Basic references 89
E.3 Morpholoy of Korean predicates 89
E.4 Temporal structure: informative 91
E.5 Temporal annotation of non-Latin texts 92
E.6 Tense 93
E.6.1 Tense markers 93
E.6.2 Annotation guidelines for the attribute @tense 95
E.6.3 Contextual interpretation of tense 96
E.7 Aspect 114
E.7.1 Aspect markers 114
E.7.2 Annotation of aspect markers 115
E.7.3 Interpretation of aspectual features 116
E.7.4 Interpretation conditions of aspect 118
E.8 Modality 118
E.8.1 Conjectural modal markers 118
E.8.2 Annotation of modality CONJECTURAL 119
E.8.3 Interpretation of modality CONJECTURAL 120
E.9 Mood 120
E.9.1 Mood markers 120
E.9.2 Annotation of mood RETROSPECTIVE 121
E.9.3 Interpretation of RETROSPECTIVE mood 122
E.10 Specific values for <EVENT> attributes in Korean 122
E.11 Summary 122

Annex F (informative) **Past and current activities on temporal and event annotation** 124
F.1 Introductory remarks 124
F.2 Annotating temporal expressions 124
F.3 Annotating events 125
F.4 Annotating relations between times and events 127
F.4.1 Ways of capturing time-event relational information 127
F.4.2 Subordinating and aspectual relations 129

Annex G (informative) **Tools and templates** 130
G.1 Overview 130
G.2 Annotation tools and templates 130
G.2.1 Overview 130
G.2.2 The ALEMBIC workbench 131
G.2.3 The CALLISTO toolkit 131
G.2.4 The TANGO temporal relation editor 131
G.3 Analytic tools 132

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub

Full document details: BS ISO 24617-1:2012

G.3.1 Overview ... 132
G.3.2 The TARSQI toolkit ... 132
G.3.3 The IBM TimeML annotator ... 133
G.3.4 The Amsterdam temporal component extractor ... 133
G.3.5 The Time Calculus analyser ... 133

Annex H (normative) **Specification** ... 134
H.1 Requirement ... 134
H.2 Attribute classes ... 134
H.2.1 att.anchored ... 134
H.2.2 att.annotate ... 135
H.2.3 att.id ... 135
H.2.4 att.lang ... 135
H.2.5 att.linguistic ... 136
H.2.6 att.pointing ... 138
H.2.7 att.typed ... 138
H.3 Elements ... 139
H.3.1 <ALINK> ... 139
H.3.2 <CONFIDENCE> ... 139
H.3.3 <EVENT> ... 140
H.3.4 <MLINK> ... 141
H.3.5 <SIGNAL> ... 141
H.3.6 <SLINK> ... 142
H.3.7 <TIMEX3> ... 143
H.3.8 <TLINK> ... 145
**H.3.9 ** ... 146
H.3.10 <isoTimeML> ... 146
H.3.11 <s> ... 146
H.3.12 <w> ... 146

Bibliography ... 147

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

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.

ISO 24617-1 was prepared by Technical Committee 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, ISO-TimeML)*

— *Part 2: Dialogue acts*

The following parts are under preparation:

— *Part 4: Semantic roles (SemAF-SRL)*

— *Part 5: Discourse structure (SemAF-DS)*

The following parts are planned:

— *Part 3: Named entities (SemAF-NE)*

— *Part 6: Principles of semantic annotation*

— *Part 7: Spatial information (ISO-Space)*

— *Part 8: Relations in Discourse (SemAF-DRel)*

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

Introduction

This part of ISO 24617 results from the agreement between the TimeML Working Group and the ISO Working Group, ISO/TC 37/SC 4/WG 2, *Language resource management – Semantic annotation*, that a joint activity should take place to accommodate the two existing documents for annotating temporal information, *TimeML 1.2.1* and *TimeML Annotation Guidelines*, into ISO international standards. This work should lead to the achievement of two objectives:

— modification of the two documents in conformance to the ISO International Standards;

— verification of the annotation guidelines for a wide coverage of multilingual resources.

It should be noted that this part of ISO 24617 provides normative guidelines not just for temporal information, but also for information content in various types of events in English as well as other languages.

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

Documents
Document Sharing Hub

INTERNATIONAL STANDARD ISO 24617-1:2012(E)

Language resource management — Semantic annotation framework (SemAF) —

Part 1:
Time and events (SemAF-Time, ISO-TimeML)

1 Scope

Temporal information in natural language texts is an increasingly important component to the understanding of those texts. This part of ISO 24617, *SemAF-Time*, specifies a formalized XML-based markup language called *ISO-TimeML,* with a systematic way to extract and represent temporal information, as well as to facilitate the exchange of temporal information, both between operational language processing systems and between different temporal representation schemes. The use of guidelines for temporal annotation has been fully attested with examples from the TimeBank corpus, a collection of 183 documents that have been annotated by TimeML before the current version of *ISO-TimeML* was formulated.

NOTE Throughout this document, *SemAF-Time* refers to the ISO 24617-1, while *ISO-TimeML* refers to the annotation language specified in this document.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

NOTE The first reference shows how dates and times are represented and the second provides a format for the standoff representation of *ISO-TimeML* annotation presented here.

ISO 8601:2004, *Data elements and interchange formats — Information interchange — Representation of dates and times*

ISO 24612:2011, *Language resource management — Linguistic annotation framework* (*LAF*)

3 Terms and definitions

For the purposes of this document, the terms and definitions in ISO 8601:2004 and the following apply.

NOTE The terms and definitions provided below are provided to clarify the terminology relating to the metamodel, specification, and semantics of *ISO-TimeML*. Terminology derived from XML and other formal languages as well as from general temporal logics is not defined here.

3.1
ALINK
linking tag that represents a phase relation between an aspectual verb (or morpheme) and a predicate denoting an **event** (3.5)

3.2
annotation
process of adding information to segments of language data or that information itself

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

3.3
beginning
instant (3.6) at which a **temporal interval** (3.17) begins

NOTE Adapted from Hobbs and Pan (2004).

3.4
end
instant (3.6) at which a **temporal interval** (3.17) ends

NOTE Adapted from Hobbs and Pan (2004).

3.5
event
eventuality
something that can be said to obtain or hold true, to happen or to occur

NOTE The term "event" is used here with a very broad notion of event, which includes all kinds of actions, states, processes, etc. It is not to be confused with the more narrow notion of event as something that happens at a certain point in time (such as the clock striking 2, or waking up) or during a short period of time (such as laughing).

3.6
instant
point in time with no interior points

NOTE Time is often viewed as a straight line from minus infinity to plus infinity. In this view, time is formed by an infinite sequence of points. An instant can also be seen as an infinitesimally small interval. Cf. OWL-Time Ontology for "instant": http://www.w3.org/TR/owl-time/.

3.7
markable
entity in general, or segment of a text in particular, that is subject to an annotation (3.2)

3.8
MLINK
linking tag that represents the measurement of the duration of an **event** (3.5) or the measurement of the length of a (possibly discontinuous) time span

3.9
point of event
instant (3.6) at which the **event** (3.5) mentioned in a given utterance occurs

NOTE Next to a point of speech, a point of event also needs to be defined in order to interpret tense. For example, in "Arthur smiled", the temporal location of the point of event can be defined as being prior to the point of speech.

3.10
point of reference
instant (3.6) of temporal perspective on the **event** (3.5) in a given utterance

NOTE 1 "Arthur will have gone by tomorrow", where the point of speech is now, the point of event is some time in the future, but before the point of reference referred to by "tomorrow".

NOTE 2 To locate certain tenses in time, a third anchor point is also required, defined as the point of reference.

3.11
point of speech
time unit (3.17) at which a given utterance occurs

NOTE 1 The notion of point of speech is needed in order to interpret tense. This requires the use of anchor points in time, of which the point of speech is one (point of text, see 3.12, is another one). For example, in "Arthur smiled", the point of speech is the time that the utterance is made.

NOTE 2 For a document as a whole, this may be considered to be the same as the document creation time.

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

3.12
point of text
instant (3.6) at which reported speech is anchored

NOTE It is the point of time considered in the text of the speech. So for example, when a person is telling a story, it is not enough to know the point of the speech itself (the document creation time), but the point at which the speech in the story is taking place.

3.13
representation
format in which an **annotation** (3.2) is rendered, for instance in XML, independent of its content

3.14
SLINK
linking tag that represents a subordinating relation between two **events** (3.5)

3.15
temporal interval
period
uninterrupted stretch of time, with internal point structure.

NOTE 1 Adapted from WordNet.

NOTE 2 Time is often viewed as a straight line from minus infinity to plus infinity. A temporal interval is a part of that line without any holes, containing all the points between its beginning and its end.

NOTE 3 In mathematics, an important issue is whether an interval includes its beginning and its end (is “closed”) or not (is “open” or “half-open”). In natural language descriptions of intervals this may also be relevant, as when describing an interval in terms of a number of days, but not with the same granularity as in mathematics. Cf. OWL-Time Ontology for “interval”: http://www.w3.org/TR/owl-time/.

3.16
temporal ordering relation
relation that determines how objects are ordered in time

EXAMPLE precedence, simultaneity.

NOTE There is a limited number of ways to order objects which are collectively called ordering relations.

3.17
temporal unit
element in a **time amount** (3.18) that quantifies the length of a **temporal interval** (3.15) or a set of **temporal intervals** (3.15)

NOTE 1 Adapted from Bunt (1985).

NOTE 2 In measurement systems, various units are defined for different purposes. Small units such as seconds and minutes are defined to measure small temporal intervals; as one may want to avoid working with big numbers, for larger temporal intervals, units such as week, year, decade, and century are defined.

NOTE 3 The amount of a temporal unit is called a measure.

3.18
time amount
quantity of time, measured by **temporal units** (3.17) over **temporal intervals** (3.15)

NOTE 1 Adapted from Bunt (1985).

NOTE 2 A time amount is a measure of time that can be expressed in terms of a number of temporal units, such as “half an hour” or “30 minutes”.

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

3.19
tense
way that languages express the time at which an **event** (3.5) described by a sentence occurs

NOTE This is characterized as a property of a verb form. Noun forms will not be said to exhibit tense but rather temporal markers.

3.20
TLINK
linking tag that represents a temporal relation between two temporal entities: namely, between two **events** (3.5), two temporal expressions, or between a temporal expression and an event

NOTE 1 Adapted from Pustejovsky *et al*. (2004).

NOTE 2 Some ordering relations cannot be expressed by an ordering relation between two events because a signal, like a temporal preposition, complicates the ordering or there is an ordering relation between a temporal signal and an event.

4 Overview

An understanding of temporal information is needed to better understand natural language texts in general. Previous work in time stamping is a step in the right direction, but to fully appreciate the complexity of a text with respect to time, the ability to order events and temporal expressions is needed. This part of ISO 24617 defines *ISO-TimeML*, a markup language for time and events, which has been specifically designed for this task.

ISO-TimeML annotates all expressions having temporal import, broadly categorized as temporal expressions and eventualities (situations, events, states, and activities). Temporal expressions and events participate in temporal relationships (e.g. “before”, “simultaneous”), subordinating relationships (e.g. “intensional”, “factive”), and aspectual relationships (e.g. “initiates”, “continues”). *ISO-TimeML* provides an additional expressive capability of capturing and representing the complexities of these relationships.

TimeML, the precursor of *ISO-TimeML*, is already in use in a number of applications focusing on analysis (manual and automatic) of news articles. The TimeBank corpus contains approximately 185 such documents and has been validated against the most recent version of TimeML. The resulting output of a TimeML annotated document is in XML, which allows for general XML validation methods to be used. In addition to supporting interoperability among different temporal representation schemes, TimeML has been shown adequate to support a mapping from the temporal information in a text to its formal representation in a Web Ontology Language such as OWL-Time.

Unlike prior event annotation schemes, *ISO-TimeML*'s somewhat unique definition of an event does not limit the standard's applicability to specific natural language genres. An *ISO-TimeML* event is simply something that can be related to another event or temporal expression using an *ISO-TimeML* relationship — thus an *ISO-TimeML*-compliant representation can be adapted (derived) from the full standard specification, appropriate to different genres, styles, domains, and applications. Future work will involve applying the standard in such different contexts, and formulating guidelines and principles for appropriate use of *ISO-TimeML* in a variety of language engineering environments.

5 Motivation and requirements

The identification of temporal and event expressions in natural language text is a critical component of any robust information retrieval or language understanding system, and recently this has become an area of intense research in computational linguistics and Artificial Intelligence. The importance of temporal awareness to question answering systems has become more obvious as current systems strive to move beyond keyword and simple named-entity extraction. Named-entity recognition has moved the fields of information retrieval and information exploitation closer to access by content, by allowing some identification of names, locations and products in texts. One of the major problems that has not been solved is the recognition of events and their temporal anchorings in text. Events are naturally anchored in time within a narrative. Without a robust ability to identify and extract events and their temporal anchoring from a text, the real aboutness of the text can be missed. Moreover, since entities and their properties change over time, a database of assertions about entities will be incomplete or incorrect if it does not capture how these properties are temporally updated. To this end, event recognition drives basic inferences from text.

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

As it happens, however, much of the temporal information in an article or narrative is left implicit in the text. The exact temporal designation of events is rarely explicit and many temporal expressions are vague at best. A crucial first step in the automatic extraction of information from such texts, for use in applications such as automatic question answering or summarization, is the capacity to identify what events are being described and to make explicit when these events occurred.

Another important point is that, although most of the information on the web is in natural language, it is unlikely that it will ever be marked up for semantic retrieval, if that entails hand annotation. Natural language programs will have to process the contents of web pages to produce annotations. Remarkable progress has been made in the last decade in the use of statistical techniques for analysing text. However, these techniques, for the most part, depend on having large amounts of annotated data, and annotations require an annotation scheme. Hence, in addition to developing the necessary tools for temporal analysis, it is important to enable for seamless integration into existing and emerging ontologies, such as OWL. Interest in temporal analysis and event-based reasoning has contributed to the development of a specification language for events and temporal expressions and their orderings (TimeML). Some issues relating to temporal and event identification have remained unresolved, however, and *ISO-TimeML* has been designed to address these issues. Specifically, four basic problems in event-temporal identification have been addressed in the design of *ISO-TimeML*:

— time anchoring of events (identifying an event and anchoring it in time);

— ordering events with respect to one another (distinguishing lexical from discourse properties of temporal ordering);

— reasoning with contextually underspecified temporal expressions (temporal functions such as “last week” and “two weeks before”);

— reasoning about the persistence of events (how long does an event or the outcome of an event last).

The specification language, *ISO-TimeML*, is designed to address these issues, in addition to handling basic tense and aspect features.

Linking a formal theory of time with an annotation scheme aimed at extracting rich temporal information from natural language text is significant for at least two reasons. It will allow us to use the multitude of temporal facts expressed in text as the ground propositions in a system for reasoning about temporal relations. It will also constitute a forcing function for developing the coverage of a temporal reasoning system, as we encounter phenomena not normally covered by such systems, such as complex descriptions of temporal aggregates.

6 Basic concepts and metamodel

Regarding the temporal information in a document, a distinction can be made between (1) the temporal metadata, regarding when the document was created, published, distributed, received, revised, etc., and (2) the temporal properties of the events and situations that are described in the document. The former type of information is associated with the document as a whole; information of the latter type will be associated in annotations with parts of the text in the document, “markables” such as words and phrases.

Temporal objects and relations have been studied from logical and ontological points of view; well-known studies include those by Allen (1984), Prior (1967), and more recently Hobbs and Pan (2004); see also the collection of papers in Mani *et al*. (2005). The most common view of time, which underlies most natural languages, is that time is an unbounded linear space running from a metaphorical “beginning of time” at minus infinity to an equally metaphorical “end of time” at plus infinity. This linear space can be represented as a straight line, the points of which correspond to moments in time; following Hobbs and Pan (2004), we will also use the term “instant” to refer to time points. From a mathematical point of view, the points on the time line are line segments of infinitesimally small size, corresponding to the intuition that a moment in time can, in principle, be determined with any precision that one may wish.

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

For linguistic and philosophical reasons, several classifications have been proposed of verbs describing various types of states or events, the Vendler classification being the best known (Vendler, 1967). For the annotation of temporal information in text, not only verbs with their tenses and temporal modifications should be considered, but also nouns, since nouns may also denote events and situations (“The meeting tomorrow”; “The six o'clock news”). In TimeML, Pustejovsky *et al*. (2007) have proposed a classification of states and events into seven categories. In the literature, a distinction is often made between events and states, where events are commonly characterized as occurring at a point in time or during a certain definite interval, whereas states may obtain for any indefinite stretch of time (“The Mediterranean Sea separates Europe from Africa”). On a terminological note: the term “event” will henceforth be used as a generic term that also covers such notions as “state”, “situation”, “action”, “process”, etc.; this broad notion of event has also been termed “eventuality” (Bach, 1986).

In reality, nothing happens in infinitesimally small time; every event or state that occurs in reality (or in someone's mind) requires more than zero time, although natural languages offer speakers the possibility to express themselves as if something occurs at a precise instant (such as “I will call you at twelve o'clock”). Since instants are formally a special kind of interval, a consistent approach to modelling the time that an event occurs is to always use intervals, where it may happen that the interval associated with a particular event is regarded as having zero length, and thus being an instant. This is reflected in the metamodel presented in Figure 1, which uniformly relates events with temporal intervals.

The length of an interval can also occur as temporal information in a text, as in “I used twelve hours to read that book” and “It takes seven minutes to walk to the station”. An expression such as “seven minutes” does not denote an interval, but the length of an interval. It is the temporal equivalent of spatial distance (“seven miles”). To describe the length of a temporal interval, one needs a unit of measurement, which may be combined with a numerical expression to obtain an amount of time. The metamodel presented below therefore includes the concept of an amount of time, related to intervals through the function *length*, and the auxiliary concepts of temporal units and real numbers. (Moreover, in the *ISO-TimeML* semantics, different temporal units are related through a conversion function, stipulating such things as 1 hour = 60 minutes; 1 day = 24 hours, etc. An amount of time can be characterized equivalently by as many pairs <*numeral, temporal unit*> as there are temporal units, the equivalence being defined through the numerical conversions between units [see Bunt (1985)].

Regarding the temporal anchoring of events in time, it may be noted that the association of a temporal interval with an event does not necessarily mean that the event took place during every moment within that interval. When someone says “I've been working on my presentation from 8.30 to 12 o'clock”, that presumably does not mean that the speaker has been working on his presentation for every single moment between 8.30 and 12 o'clock; there must have been interruptions for having some coffee, going to the bathroom, etc. In such a case it is more accurate to anchor the event at the time span starting at 8.30 and ending at 12 o'clock, a “time span” being understood as a period of time that may have “holes”, where the event was interrupted. The metamodel shown in Figure 1 does not distinguish time spans, but reflects the assumption that whether an event occurs during an interval, with or without any interruptions, can only be decided on a case by case basis, and is best modelled as a property of the temporal anchoring relation applied to a specific event.

ISO 24612:2011 insists on the use of stand-off annotation, i.e. the construction of annotations in separate files, separate from the document containing the primary language data, as contrasted with in-line annotation. Stand-off annotations refer to specific locations in the primary data by addressing byte offsets, linguistic elements such as words, or times associated with recorded data, to which the annotation applies. Compared to in-line annotation, stand-off annotation has the advantages of respecting the integrity of the primary data and of allowing multiple annotations to be layered over a given primary document. Since semantic annotations typically occur at a relatively high level in a layered annotation structure, they do not necessarily refer directly to segments in the primary data, but may also refer to structures in other annotation layers. The generic term “markable” is used to refer to the entities that the annotations are associated with.

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

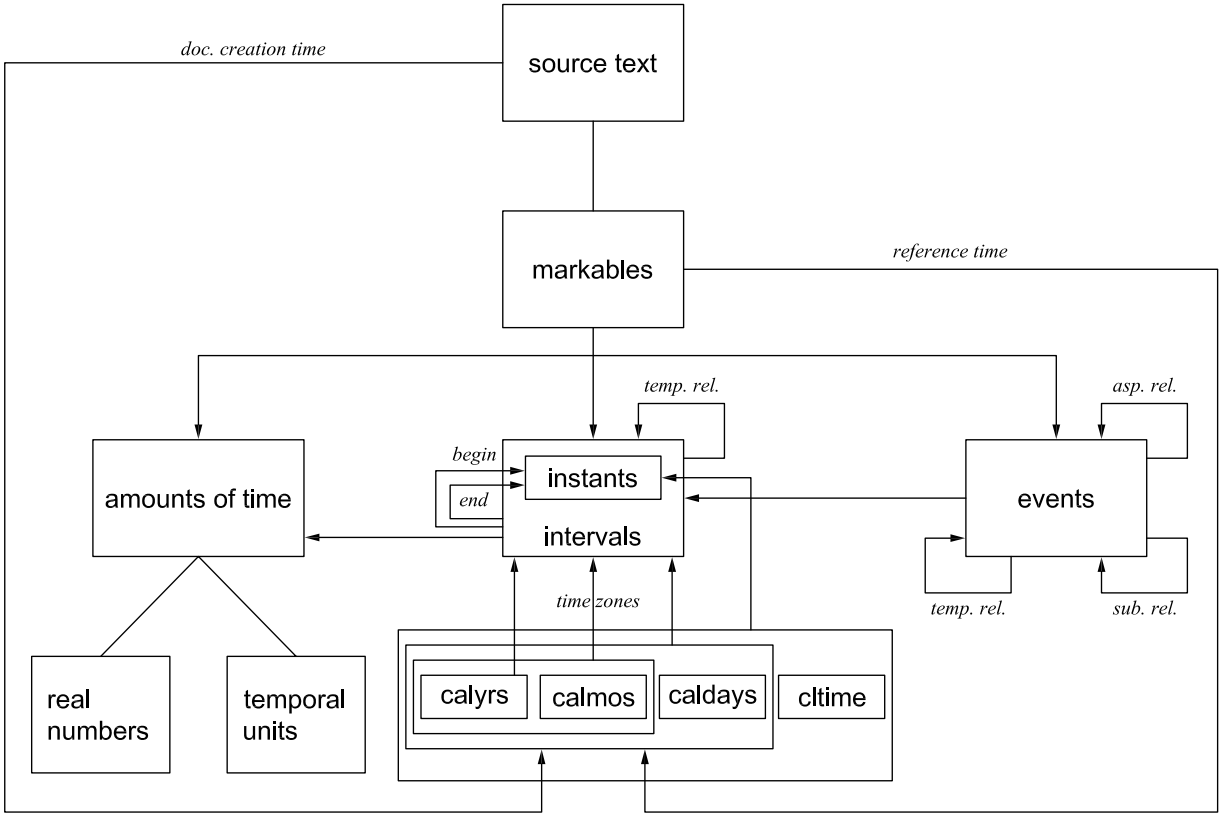

Figure 1 — Metamodel

Markables are derived from documents, which will have certain metadata that are particularly important for the interpretation of temporal annotations. For interpreting the tenses of verb forms and adverbial temporal deixis in a text (“yesterday”; “next week”), for instance, one must know when the text was produced. This will often be defined by the document creation time, and more precisely by the combination of a creation time and a creation location, since the latter defines the time zone within which the creation time is precisely defined. In many documents, the time and place of the document creation will be those of all the markables that may be derived from the document, but it may also happen that the text in a document introduces other times and places relative to which the annotations of the markables should be understood. A time zone, such as Greenwich Mean Time (GMT), can be seen as a way of segmenting the time line into named segments of particular lengths, such as (calendar) years, months, days, hours, and minutes. Accordingly, time zones show up in the metamodel as functions mapping a calendar year (“2008”), a combination of a calendar year and a calendar month (“May 2008”), a date (“May 25, 2008”), or a date plus a clock time (“May 25, 2008, 12.30 p.m.”) onto a temporal interval (in the latter case, an instant).

A markable may refer to more than one, related event, as in “She started to laugh” (two aspectually related events); “John drove to Boston after the concert” (two temporally related events); or “Will you attend the meeting on Tuesday?” (one event having a subordination relation to another). For expressing such relations, the metamodel includes the corresponding classes of relations showing up as inter-event links. Temporal relations between events may also be stated between intervals, hence they show up again in the metamodel.

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

7 Specification of ISO-TimeML

7.1 Overview

ISO-TimeML implements the fundamental distinction between the concepts of ***annotation*** and ***representation*** which is made in ISO 24612:2011 (*LAF*) [see also Ide and Romary (2003)]. The term ***annotation*** is used to refer to the process of adding information to segments of language data, or to refer to that information itself. This notion is independent of the format in which this information is represented. The term ***representation*** is used to refer to the format in which an annotation is rendered, for instance in XML, independent of its content. According to *LAF*, annotations are the proper level of standardization, not representations. This part of ISO 24617 (*SemAF-Time*) therefore defines a specification language for annotating documents with information about time and events at the level of annotations and then for representing these annotations in a specific way, namely XML. This language is called *ISO-TimeML*.

The distinction between annotations and representations is reflected in the specification of *ISO-TimeML* given below, where, following Bunt (2010), an ***abstract syntax*** is defined in addition to a ***concrete syntax***. The abstract syntax specifies the kinds of elements which make up the information in annotations, and describes how these elements may be combined to form complex annotation structures; these combinations are defined as set-theoretical structures, independent of any particular representation format. There are infinitely many ways in which these structures can be represented. In line with other ISO/TC 37/SC 4 standards, and in particular with *LAF*, *ISO-TimeML* has an XML-based concrete syntax for temporal and event-related information. *ISO-TimeML* has a semantics associated with its abstract syntax, which defines the meanings of *ISO-TimeML* annotation structures. All concrete representations of *ISO-TimeML* annotations inherit the semantics of the abstract syntax.

The specification of *ISO-TimeML* consists of three parts, mirroring the *LAF* distinction between the two levels of annotations and representations:

— Part 1 describes the abstract syntax of *ISO-TimeML* annotations (see 7.2);

— Part 2 describes a concrete syntax, which specifies a format for representing these annotations in XML [*ISO-TimeML* (see 7.3, Annex A and Annex H)];

— Part 3 describes the semantics of *ISO-TimeML* (see Clause 8) and is again split into two parts:

  — one part describes the semantics of the XML representation of the annotation of temporal expressions using Interval Temporal Logic, a first-order logic for reasoning about time;

  — the other part describes a more general, event-based semantics for the abstract *ISO-TimeML* syntax.

The annotations shall be implemented as specified in Annexes A and H.

7.2 Abstract syntax

7.2.1 Introduction

The abstract syntax of *ISO-TimeML* defines the set-theoretical structures that constitute the information about time and events that may be contained in annotations. The definition of the abstract syntax consists of

1) a specification of the elements from which these structures are built up, called a “conceptual inventory”, and

2) a set of syntax rules which describe the possible combinations of these elements. What these combinations mean, i.e. which information they capture, is specified by the semantics associated with the abstract syntax (see 8.4).

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

7.2.2 Conceptual inventory

The concepts that can be used to build *ISO-TimeML* annotations fall into five categories, all consisting of finite sets, along with the concepts of real number and natural number:

— a finite set of elements called “event classes”, “tenses”, “aspects”, “polarities”, and “set-theoretic types”;

— a finite set of elements called “temporal relations”, “duration relations”, “numerical relations”, “event subordination relations”, and “aspectual relations”;

— a finite set of elements called “time zones”;

— a finite set of elements called “calendar years”, “calendar months”, “calendar day numbers”, “clock times” (natural numbers ranging from 0000 to 0059; from 0100 to 0159; ... from 2300 to 2400);

— a finite set of elements called “temporal units”.

Natural numbers are needed for capturing the information expressed in English, for example, by “twice” and “three times”; real numbers are needed for cases such as “two and a half hours”.

7.2.3 Syntax rules

7.2.3.1 General principles

Annotation structures in *ISO-TimeML* come in two varieties, which we will refer to as ***entity structures*** and ***link structures***. Entity structures contain semantic information about a segment of source text; link structures describe semantic relations between segments of source text by means of links between entity structures.

The simplest kind of an *ISO-TimeML* structure is a single entity structure, which is a pair <m,a> consisting of a markable m and an annotation a, or a single link structure which relates two entity structures. More complex annotation structures consist of a set of entity structures and a set of link structures which link the entity structures together through temporal and inter-event relations.

More formally, an *ISO-TimeML* annotation structure consists of two sets M and L, where M is a set of pairs <*markable, entity structure*> and L is a set of triples <*markable, entity structure, entity structure*>, such that each element of L contains at least one entity structure that occurs in M (This latter condition ensures that the links in an annotation structure relate to the entity structures that it contains; they may, in addition, also contain links to elements outside the current annotation structure).

7.2.3.2 Entity structures

There are five types of annotations that may form an entity structure, containing information about (1) events; (2-4) temporal objects (intervals, instants, and amounts of time); (5) frequencies of events; and (6) explicit temporal relations (as, for example, expressed in English by temporal prepositions).

— An **event structure** is a 7-tuple <C, T, A, Σ, N, P_N, V> where C is a member of the set of event classes; T and A are a tense and an aspect, respectively; Σ is a set-theoretical type (such as an individual object or set of individual objects); N is a natural number (e.g. the number 2 for dealing with such examples as “John kissed Mary twice”); P_N is an amount of time (such as “two and a half hours”, for such examples as “John called Mary twice every two and a half hours”), and V is a veracity (claimed truth or falsity, corresponding to positive or negative polarity in natural language).

— An **instant structure** is either a triple <*time zone, date, clock time*>, where a date is a triple consisting of a calendar year, a calendar month, and a calendar day number; or a triple <*time amount structure, instant structure, temporal relation*> (“half an hour before midnight”).

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

— The following set-theoretical structures are **interval structures**:

  — a pair <t_1,t_2> of two instant structures, corresponding to the beginning and end points of the interval;

  — a triple <*time amount structure, instant structure, temporal relation*> (“three weeks before Christmas”; “two years from today”);

  — a triple <t_1,t_2,R> where t_1 and t_2 are either instant structures or interval structures, and where R is a duration relation (examples: “from 1992 until 1995”; “from 1882 through 1995”);

— A **time-amount structure** is a pair <n,u>, where n is a real number and u a temporal unit, or a triple <R,n,u>, where R is a numerical relation (such as greater than) and n and u are as before;

— A **temporal relation structure** is just a temporal relation.

7.2.3.3 Link structures

There are five types of link structures in *ISO-TimeML*: for temporal ***anchoring*** of events in time; for temporal ***ordering of events*** and/or ***intervals or instants*** relative to each other; for measuring the length of an interval; for ***subordination relations*** between events, and for ***aspectual relations*** between events.

— A **temporal anchoring structure** is a triple <event structure, interval structure, temporal anchoring relation>, or a triple <event structure, instant structure, temporal anchoring relation>;

— A **temporal relation structure** is a triple <event structure, event structure, temporal relation>, or a triple <interval or instant structure, interval or instant structure, temporal relation>;

— A **time measurement structure** is a pair <event structure, time-amount structure> or a pair <interval structure, time-amount structure>;

— A **subordination structure** is a triple <event structure, event structure, subordination relation>;

— An **aspectual structure** is a triple <event structure, event structure, aspectual relation>.

7.3 Concrete XML-based syntax

7.3.1 TimeML vs. ISO-TimeML: Stand-off annotation and other differences

A concrete syntax consists of the specification of names for the various sets forming the conceptual vocabulary, plus a listing of specific named elements of these sets, and a specification of how to represent the annotation structures defined by the abstract syntax.

The *ISO-TimeML* concrete syntax has been inspired by the TimeML format [Pustejovsky *et al*. (2003; 2007)]. and is reproduced below with a few amendments to do justice to the stand-off character of *ISO-TimeML* annotations and to the conceptual differences between *ISO-TimeML* and the original TimeML.

One of the differences between *ISO-TimeML* and TimeML is that *ISO-TimeML* adopts a stand-off annotation format, where TimeML has been using an in-line annotation format. The differences between the two formats are illustrated in the examples (1) and (2) below:

(1) John left on 31 December 2007.

```
John
 <EVENT  eid="e1"  eiid="ei1"  type="transition"  tense="past">left</EVENT>
 <SIGNAL sid="s1">on</SIGNAL>
 <TIMEX3 tid="t1" type="date" value="2007-12-31">31 December 2007</TIMEX3>
```


© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

(2) John left on 31 December 2007.

```
<s>
 <w xml:id="w1">John</w>
 <w xml:id="w2">left</w>
 <w xml:id="w3">on</w>
 <w xml:id="w4">31</w>
 <w xml:id="w5">December</w>
 <w xml:id="w6">2007</w>
</s>
 <EVENT eid="e1" eiid="ei1"target="#w2"  type="transition" tense="past"/>
 <SIGNAL xml:id="s1" target="#w3"/>
 <TIMEX3 tid="t1" target="#range(w4,w6)"  type="date" value="2007-12-31"/>
```

The first example shows the TimeML representation (somewhat simplified) in in-line format; the second shows the same annotation in stand-off format, together with the segmentation of the text.

For linking annotations to primary text in stand-off format, *ISO-TimeML* makes use of the @target attribute defined in TEI (2010) *P5 Guidelines*. This attribute can be used to point to units in the primary data preprocessed by another annotation scheme, such as tokenization, morpho-syntactic annotation, or syntactic annotation. It can also be used to point to word tokens identified in an in-line form of segmentation; see examples (3) and (4). Moreover, the TEI-defined functions “range” and “group” can be used to construct references to contiguous or non-contiguous text spans, respectively.

ISO-TimeML has adopted the following additional modifications of TimeML.

— The attribute names @eid, @tid and @sid have been changed simply to @xml:id, while marking up their values as they are.

— The element <MAKEINSTANCE> has been deleted along with its associated attribute @eiid or any other attributes and move some of its attributes such as @tense and @aspect to the element <EVENT>.

— The attribute @eiid has been deleted; each <EVENT> is interpreted in *ISO-TimeML* as denoting an event instance. As a result, the attributes such as @eventInstanceID and @relatedToEventInstance that occurred in the link elements such as <TLINK> and <ALINK> are replaced by such attributes as @eventID and @relatedToEvent.

— The attribute @pred has been introduced, having a value which is understood as denoting the content related to that event.

— Durations, such as “five seconds”, have been introduced as a separate type of temporal entity, different from instants and temporal intervals.

Here are two examples. In the first one, the segmentation of the text into words is annotated in an in-line format.

(3) John left on 31 December 2007.

```
<s>
 <w xml:id="w1">John</w>
 <w xml:id="w2">left</w>
 <w xml:id="w3">on</w>
 <w xml:id="w4">31</w>
 <w xml:id="w5">December</w>
 <w xml:id="w6">2007</w>
</s>
<isoTimeML>
 <EVENT xml:id="e1" target="#w2" pred="LEAVE" class="OCCURRENCE" type="TRANSITION" tense="PAST"/>
 <SIGNAL xml:id="s1" target="#w3"/>
 <TIMEX3 xml:id="t1" target="#range(w4,w6)" type="DATE" value="2007-12-31"/>
 <TLINK eventID="#e1" relatedToTime="#t1" signalID="#s1" relType="IS_INCLUDED"/>
</isoTimeML>
```

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

In the second example, the given text is assumed to be preprocessed by a tokenizer.

(4) John taught on Tuesday but not on Wednesday.

```
<isoTimeML>
 <EVENT xml:id="e1" target="#token1" pred="TEACH" class="OCCURRENCE" type="PROCESS" tense="PAST"
   polarity="POS"/>
 <SIGNAL xml:id="s1" target="#token2"/>
 <TIMEX3 xml:id="t1" target="#token3" pred="TUESDAY" type="DATE" value="2007-12-31"/>
 <EVENT xml:id="e2" pred="TEACH" class="OCCURRENCE" type="PROCESS" tense="PAST" polarity="NEG"/>
 <SIGNAL xml:id="s2" target="#token6"/>
 <TIMEX3 xml:id="t2" target="#token7" pred="WEDNESDAY" type="DATE" value="2007-01-01"/>
 <TLINK eventID="#e1" relatedToTime="#t1" signalID="#s1" relType="IS_INCLUDED"/>
 <TLINK eventID="#e2" relatedToTime="#t2" signalID="#s2" relType="IS_INCLUDED"/>
</isoTimeML>
```

7.3.2 Naming conventions

Inasmuch as XML is case-sensitive, it is necessary for *ISO-TimeML* to specify exactly the case of all its elements. This document follows the convention of indicating tag names and attribute values all in upper case (e.g. EVENT, PROGRESSIVE) and attribute names in lower or mixed case (e.g. @tense, @relatedToTime). Since attribute values are typically atomic (one-word) while attribute names often consist of multiple words, this convention would seem to maximize readability of the annotation.

Attributes that range over values of XML datatype URI references to IDs typically consist of the name of the element indexed, followed by “ID” (e.g. eventID) or a descriptive name (e.g. relatedToTime).

The values of the various ID attributes are specified as beginning with one or two characters, followed by an integer. This scheme is mandated by the syntax of XML. While attribute values of type ID can consist of any sequence of letters, digits, and the hyphen, underscore, and period characters, they shall begin with either an underscore or a letter. Therefore, e23 is a valid XML ID; but 23 is not. This naming convention also helps make the examples a bit more readable, especially in the case of link tags, which can contain multiple URIs of different kinds.

7.3.3 Example annotations

Though this document describes the full *ISO-TimeML* language, many of the example annotations provided show the result of annotation only through the output of initial automatic tagging combined with human annotation/editing, but do not include elements (e.g. attributes and/or attribute values) that may be introduced by later processing components (e.g. the closure tool). In particular, TIMEX3 tags that are treated as temporal functions typically appear in the examples in an underspecified form. However, those elements that do appear are sufficient for the output of manual annotation.

Finally, note that all examples in this document have been validated against an *ISO-TimeML* schema compliant with the specification presented in Annex H.

7.3.4 Basic elements: <EVENT>, <TIMEX3>, and <SIGNAL>

7.3.4.1 <EVENT>

The <EVENT> element is used to annotate those elements in a text that describe what is conventionally referred to as an eventuality. Syntactically, events are typically expressed as inflected verbs, although event nominals and nominalizations, such as “crash” in “... killed by the crash” and “examination” in “a recent examination of evidence”, should also be annotated as <EVENT>.

The <EVENT> tag is also used to annotate a subset of the states in a document, typically expressed as adjectives. This subset of states includes those that are either transient or explicitly marked as participating in a temporal relation. See Annex A, *Core annotation guidelines*, for more details.

Table 1 provides the syntax and definition for the <EVENT> element.

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

Table 1 — Specification of <EVENT>

target	specifies the destination of the reference by supplying one or more URI References.
pred	denotes the content related to that event through the indication of a lexical predicate.
class	characterizes each event as occurrence, perception, reporting, aspectual, state, I-state or I-action.
type	three event types are distinguished: state, process and transition.
tense	captures standard distinctions in the grammatical category of verbal tense.
aspect	captures standard distinctions in the grammatical category of verbal aspects such as perfective, imperfective, progressive and so on.
polarity	Boolean attribute that conveys the polarity associated with the event in question.
modality	conveys the modality associated with the event: different degrees of epistemic modality, deontic modality, etc.
comment	remarks made by an annotator.

— The value STATE occurs twice as a possible value of the attribute @class and also as a possible value of the attribute @type. As a result, we may have an example such as: <EVENT xml:id="e1" pred="LOVE" class="STATE" type="STATE">. For the present, there will remain a redundancy in the attribute specification, until the attribute @class is replaced with some ontological marker.

— The range of possible values of the @modality attribute is great, much depending on various features of particular languages. For instance, Italian requires CONDITIONAL as a modal value of @modality besides SUBJUNCTIVE, and Korean requires RESTROSPECTIVE and CONJECTURAL as possible values of the optional attribute @mood, which is not a strictly semantic attribute but which has implications for the semantic modality associated with an event. This will be shown in the annexes of this document.

— The attribute @comment is a part of all the *ISO-TimeML* tags, and is there for annotators to add clarifications and other observations about the text being marked.

Each <EVENT> tag represents a unique instance of an event, identified as the event instance identification number. If additional instances of an event are needed, a non-consuming <EVENT> tag can be created with a new event ID.

The attributes @tense and @aspect of the element <EVENT> are represented by specific attribute values within this tag. In addition, if the event is modified by a negation, this is indicated by the appropriate value in the @polarity attribute. The term “mood” in traditional grammar refers to SUBJUNCTIVE or INDICATIVE: “If I were (PRESENT SUBJUNTIVE) a bird, I would fly” vs. “If I am (PRESENT INDICATIVE) a bird, I can fly”, “If I had been (PAST SUBJUNCTIVE) in the airport, I would have died (PAST COUNTERFACTUAL CONDITIONAL sentence).” The optional attribute @mood is used when syntactic mood is expressed by inflectional morphology on the verb; @modality, on the other hand, is reserved for the semantic annotation of an explicit modal auxiliary verb, such as “should” or “must”. The attributes @tense and @aspect are expected to have their values filled in by a pre-processing program, according to the following paradigm. See Tables 2 and 3.

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

Table 2 — Active voice

verb group	tense	aspect
teaches	PRESENT	NONE
is teaching	PRESENT	PROGRESSIVE
has taught	PRESENT	PERFECTIVE
has been teaching	PRESENT	PERFECTIVE_PROGRESSIVE
taught	PAST	NONE
was teaching	PAST	PROGRESSIVE
had taught	PAST	PERFECTIVE
had been teaching	PAST	PERFECTIVE_PROGRESSIVE
will teach	FUTURE	NONE
will be teaching	FUTURE	PROGRESSIVE
will have taught	FUTURE	PERFECTIVE
will have been teaching	FUTURE	PERFECTIVE_PROGRESSIVE

Table 3 — Passive voice

verb group	tense	aspect
is taught	PRESENT	NONE
is being taught	PRESENT	PROGRESSIVE
has been taught	PRESENT	PERFECTIVE
has been being taught	PRESENT	PERFECTIVE_PROG
was taught	PAST	NONE
was being taught	PAST	PROGRESSIVE
had been taught	PAST	PERFECTIVE
had been being taught	PAST	PERFECTIVE_PROG
will be taught	FUTURE	NONE
will be being taught	FUTURE	PROGRESSIVE
will have been taught	FUTURE	PERFECTIVE
will have been being taught	FUTURE	PERFECTIVE_PROG
being taught	NONE	PRESPART

— The attributes @tense, @aspect, and @modality require additional values to deal with languages other than English, as illustrated in Annex C (Chinese), D (Italian), and E (Korean). This is also the case for the optional attribute @mood.

The values of the attributes @polarity and @modality are determined by modifiers found near the event in the text. For languages that encode mood in the morphology of the verb, the attribute @mood is used. Here are some examples:

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

(5) should have bought

<**EVENT xml:id**="e1" **target**="#token2" **pred**="BUY" **class**="OCCURRENCE" **type**="TRANSITION" **tense**="PAST" **aspect**="PERFECTIVE" **modality**="SHOULD" **polarity**="POS"/>

(6) did not teach

<**EVENT xml:id**="e2" **target**="#token2" **pred**="TEACH" **class**="OCCURRENCE" **type**="PROCESS" **tense**="PAST" **aspect**="NONE" **polarity**="NEG"/>

(7) must not teach

<**EVENT xml:id**="e3" **target**="#token2" **pred**="TEACH" **class**="OCCURRENCE" **type**="PROCESS" **tense**="PRESENT" **aspect**="NONE" **modality**="MUST" **polarity**="NEG"/>

7.3.4.2 <TIMEX3>

The <TIMEX3> element is used to mark up explicit temporal expressions, such as times, dates, and durations. It is modelled on Setzer's (2001) <TIMEX> element, as well as the TIDES [Ferro, *et al*. (2003)] <TIMEX2> element. Since it differs both in attribute structure and in use, it was given a separate name, which reveals its heritage while at the same time indicating that it is different from its forebears.

Table 4 provides the syntax and definition for the <TIMEX3> element.

Table 4 — Specification of <TIMEX3>

target	specifies the destination of the reference by supplying one or more URI References.
pred	denotes the content related to that time through the indication of a lexical predicate.
type	characterizes each temporal expression as denotation date, time, duration or set.
functionInDocument	indicates the function of the <TIMEX3> in providing a temporal anchor for other temporal expressions in the document.
beginPoint	indicates the beginning point of a period of type DURATION.
endPoint	indicates the end point of a period of type DURATION.
quant	specifies temporal quantification.
freq	specifies temporal frequency.
temporalFunction	indicates whether the element <TIMEX3> is used as a temporal function.
value	has a specific value of the type of duration, dateTime, time, date, gYearMonth, gYear, gMonthDay, gDay, or gMonth.
valueFromFunction	pointer to a temporal function that determines its value.
anchorTimeID	points to time expression to which the <TIMEX3> markable is temporally anchored. That is, the time expression needed in order to compute its attribute value.
comment	remarks made by an annotator.

— The optional attribute @functionInDocument indicates the function of the <TIMEX3> in providing a temporal anchor for other temporal expressions in the document. If this attribute is not explicitly supplied, the default value is NONE. The non-empty values take their names from the temporal metadata tags in the Prism draft standard, which is available at http://www.prismstandard.org and are intended to have the same interpretations:

There are several times that mark the major milestones in the life of text: The time the story is published, the time it may be released (if not immediately), the time it is received by a customer, and the time that the story expires (if any). Dates and times should be represented using the W3C-defined profile of ISO 8601:2004 [W3C-NOTE-datetime]. The elements are shown in Table 5.

AVA Documents
Document Sharing Hub

© ISO 2012 – All rights reserved

Full document details: BS ISO 24617-1:2012

Table 5 — Elements for time and date information

element	role
prism:creationTime	Date and time the identified resource was first created
prism:expirationTime	Date and time when the right to publish material expires
prism:modificationTime	Date and time the resource was last modified
prism:publicationTime	Date and time when the resource is released to the public
prism:releaseTime	Earliest date and time when the resource may be distributed
prism:receptionTime	Date and time when the resource was received on current system

— There can be as many instances of TIMEX3s containing a @functionInDocument attribute with a non-empty value as there are <TIMEX3>s that express different functions. In practice, there will probably be no more than two, one with <CREATION_TIME> and another with <PUBLICATION_TIME>, since these are likely to be the only attributes that will appear in the text of documents to be annotated.

— <RELEASE_TIME> does not indicate when the document was actually released. It is a specification of when the document is allowed to be released. This comes up in documents that are syndicated and where the issuing organization wants to delay publication by syndicators, so as not to be scooped.

— The Prism standard, at least in its temporal indicators, is interested only in the document as an artefact, a piece of intellectual property. This means that the Prism values do not indicate the function of a <TIMEX3> relative to the internal narrative of the document. The specification of the *ISO-TimeML* language can fill this gap by adding values for the @functionInDocument attribute that capture narrative functions. At present, we leave the specification of possible values as is, and will defer the obvious extension until annotation of existing texts indicates that this is a pressing issue.

— The optional attribute @temporalFunction indicates whether the element <TIMEX3> is used as a temporal function, e.g. two weeks ago. If this attribute is not explicitly supplied, the default value is false. It is used in conjunction with the attribute @anchorTimeID, which indicates the element <TIMEX3> to which its denotation is applied. It also appears with the attribute @valueFromFunction, a pointer to a temporal function that determines its value. As was noted above, <TIMEX3> tags that behave as temporal functions are often underspecified in the example annotations below.

— The data types specified for the value attribute - Period, Date, Time, WeekDate, WeekTime, Season, PartOfYear, PaPrFu - are XML data types based on the 2002 TIDES guideline, which extends the ISO 8601:2004 standard for representing dates, times, and durations.

— The attribute @mod is an optional attribute adopted from TIDES. It is used for temporal modifiers that cannot be expressed either within value proper, or via links or temporal functions.

Examples for the use of @type, @value, and @mod are shown below:

(8) no more than 60 days

<**TIMEX3 xml:id**="t1" **target**="#range(token0,token4)" **type**="DURATION" **value**="P60D" **mod**="EQUAL_OR_LESS"/>

(9) the dawn of 2000

<**TIMEX3 xml:id**="t2" **target**="#range(token0,token3)" **type**="DATE" **value**="2000" **mod**="START"/>

— The attribute @anchorTimeID is used to point to another <TIMEX3> in the case of expressions such as “last week”, which have a functional interpretation. The value of @anchorTimeID provides the reference point to which the functional interpretation applies.

— The attributes @quant and @freq are used to specify sets that denote quantified times in TIMEX3. The attribute @quant is generally a literal from the text that quantifies over the expression. The attribute @freq contains an integer value and a time granularity to represent any frequency contained in the set, just as a period of time is represented as a period.

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

Examples for the use of @quant and @freq are shown below:

(10) twice a month

<**TIMEX3** **xml:id**="t3" **target**="#range(token0,token2)" **type**="SET" **value**="P1M" **freq**="2X"/>

(11) three days every month

<**TIMEX3** **xml:id**="t4" **target**="#range(token0,token3)" **type**="SET" **value**="P1M" **quant**="EVERY" **freq**="3D"/>

(12) daily

<**TIMEX3** **xml:id**="t5" **target**="#token0" **type**="SET" **value**="P1D" **quant**="EVERY"/>

— The attributes @beginPoint and @endPoint are used to anchor periods (of duration) to other time expressions in the document. If there is no explicit @xml:id to assign to one of these values, then an empty <TIMEX3> tag is created to represent the unspecified point. Conversely, if both the beginning and end points of a period are explicitly stated in the document, an empty <TIMEX3> tag is created to represent the unspecified duration.

Examples:

(13) two weeks from June 7, 2003

<**TIMEX3** **xml:id**="t6" **target**="#token0 #token1" **type**="DURATION" **value**="P2W" **beginPoint**="#t61" **endPoint**="#t62"/>
<**SIGNAL** **xml:id**="s1" **target**="#token2"/>
<**TIMEX3** **xml:id**="t61" **target**="#range(token3,token5)" **type**="DATE" **value**="2003-06-07"/>
<**TIMEX3** **xml:id**="t62" **type**="DATE" **value**="2003-06-21" **temporalFunction**="true" **anchorTimeID**="#t6"/>

(14) 1992 through 1995

<**TIMEX3** **xml:id**="t71" **target**="#token0" **type**="DATE" **value**="1992"/>
<**SIGNAL** **xml:id**="s1" **target**="#token1"/>
<**TIMEX3** **xml:id**="t72" **target**="#token2" **type**="DATE" **value**="1995"/>
<**TIMEX3** **xml:id**="t7" **target**="#range(token0,token2)" **type**="DURATION" **value**="P4Y" **beginPoint**="#t71" **endPoint**="#t72"
temporalFunction="true"/>

7.3.4.3 <SIGNAL>

The element <SIGNAL> is used to annotate sections of text, typically function words, that indicate how temporal expressions or eventualities are to be related to each other. The specification is shown in Table 6.

Table 6 — Specification of <SIGNAL>

target	specifies the destination of the reference by supplying one or more URI References
pred	denotes the content related to that signal through the indication of a lexical predicate
comment	remarks made by an annotator

The material marked by <SIGNAL> constitutes several types of linguistic elements: indicators of temporal relations such as temporal prepositions (e.g. “on”, “during”) and other temporal connectives (e.g. “when”) and subordinators (e.g. “if”). This functionality of the <SIGNAL> element was introduced by Setzer (2001).

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

7.3.5 Link elements: <TLINK>, <SLINK>, <ALINK> and <MLINK>

7.3.5.1 <TLINK>

The element <TLINK> (temporal link) is one of the four *ISO-TimeML* link elements. Link elements encode the various relations that exist between the temporal elements of a document. The motivations for having multiple types of links are the following:

— to distinguish between event types, such as those introduced by conjunction, quantification, or negation;

— to adequately handle subordinating contexts involving modality and reported speech.

The specification is shown in Table 7.

While <TLINK> establishes the ordering of an event or temporal interval relative to another event or interval, <SLINK> introduces a syntactically and semantically subordinating context, such as that introduced by modality or reporting predicates. <ALINK> establishes an aspectual relationship between two events. Finally, <MLINK> establishes a measuring relation between a temporal expression and the event it measures.

Table 7 — Specification of <TLINK>

origin	attribute supplied by closure.
eventID	points to the event involved in a temporal link.
timeID	This is the ID of the event or the time involved in the temporal link.
signalID	points, when applicable, to the explicit signal of the temporal relation.
relatedToEvent	Pointer to the entity that is being related to the event if this entity is of an event type.
relatedToTime	Pointer to the entity that is being related to the time expression if this entity is of temporal type.
relType	Temporal relation holding between the entities.
comment	remarks made by an annotator.

The value of the optional attribute @origin will be supplied by closure. This information and the link ID are primarily used by the closure algorithm. All links in *ISO-TimeML* may have these two attributes, but neither will be included in the examples presented here.

Examples:

(15) John taught on Monday and on Tuesday too.

```
<EVENT xml:id="e1" target="#token1" pred="TEACH" class="OCCURRENCE" type="PROCESS"
  tense="PAST" aspect="NONE" polarity="POS"/>
<SIGNAL xml:id="s1" target="#token2"/>
<TIMEX3 xml:id="t1" target="#token3" pred="MONDAY" type="DATE" value="xxxx-wxx-1"/>
<SIGNAL xml:id="s2" target="#token5"/>
<EVENT xml:id="e2" pred="TEACH" class="OCCURRENCE" type="PROCESS"
  tense="PAST" aspect="NONE" polarity="POS"/>
<TIMEX3 xml:id="t2" target="#token6" pred="TUESDAY" type="DATE" value="xxxx-wxx-2"/>
<TLINK eventID="#e1" signalID="#s1" relatedToTime="#t1" relType="IS_INCLUDED"/>
<TLINK eventID="#e2" signalID="#s2" relatedToTime="#t2" relType="IS_INCLUDED"/>
```


© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

(16) John left 5 minutes after the explosion.

```
<EVENT xml:id="e1" target="#token1" pred="LEAVE" class="OCCURRENCE" type="TRANSITION" tense="PAST"
  aspect="NONE" polarity="POS"/>
<TIMEX3 xml:id="t1" target="#token2 #token3" type="DURATION" value="PT5M" beginPoint="#t2" endPoint="#t3"/>
<SIGNAL xml:id="s1" target="#token4" pred="AFTER"/>
<EVENT xml:id="e2" target="#token6" pred="EXPLODE" class="OCCURRENCE" type="TRANSITION"
  tense="NONE" aspect="NONE" polarity="POS"/>
<TIMEX3 xml:id="t2" type="TIME" value="xxxx-xx-xx" temporalFunction="true" anchorTimeID="t1"/>
<TIMEX3 tid="t3" type="TIME" value="xxxx-xx-xx" temporalFunction="true" anchorTimeID="#t1"/>
<TLINK eventID="#e2" signalID="#s1" relatedToTime="#t1" relType="BEGINS"/>
<TLINK eventID="#e2" relatedToTime="#t2" relType="IS_INCLUDED"/>
<TLINK eventID="#e1" signalID="#s1" relatedToTime="#t3" relType="ENDS"/>
<TLINK eventID="#e1" relatedToTime="#t3" relType="IS_INCLUDED"/>
```

Treatment of temporal functions:

(17) John taught from September to December last year.

```
<EVENT xml:id="e1" target="#token1" pred="TEACH" class="OCCURRENCE" type="PROCESS"
  tense="PAST" aspect="NONE" polarity="POS"/>
<SIGNAL xml:id="s1" target="#token2" pred="FROM"/>
<TIMEX3 xml:id="t1" target="#token3" pred="SEPTEMBER" type="DATE" value="xxxx-09"/>
<SIGNAL xml:id="s2" target="#token4" pred="TO"/>
<TIMEX3 xml:id="t2" target="#token5" pred="DECEMBER" type="DATE" value="xxxx-12"/>
<TIMEX3 xml:id="t5" type="DURATION" value="P4M" beginPoint="#t1" endPoint="#t2" temporalFunction="true"/>
<TIMEX3 xml:id="t3" target="#range(token6,token7)" pred="LAST_YEAR" type="DATE" value="1995"                    t
temporalFunction="true" anchorTimeID="t4"/>
<TIMEX3 xml:id="t4" type="DATE" value="1996-03-27" functionInDocument="CREATION_TIME"/>
<TLINK timeID="#t1" signalID="#s1" relatedToTime="#t5" relType="BEGINS"/>
<TLINK timeID="#t2" signalID="#s2" relatedToTime="#t5" relType="ENDS"/>
<TLINK eventID="#e1" relatedToTime="#t5" relType="SIMULTANEOUS"/>
```

(18) John taught last week.

```
<EVENT xml:id="e1" target="#token1" pred="TEACH" class="OCCURRENCE" type="PROCESS"
  tense="PAST" aspect="NONE" polarity="POS"/>
<TIMEX3 xml:id="t1" target="#range(token2, token3)" pred="LAST_WEEK" type="DATE" value="XXXX-WXX"
  temporalFunction="true" anchorTimeID="t2"/>
<TIMEX3 xml:id="t2" type="DATE" value="1996-03-27" functionInDocument="CREATION_TIME"/>
<TLINK eventID="#e1" relatedToTime="#t1" relType="IS_INCLUDED"/>
```

The <TLINK> may also relate to <TIMEX3> expressions. This is the only representation that will adequately express the temporal anchoring of this event.

(19) John taught last week on Monday.

```
<EVENT xml:id="e1" target="#token1" pred="TEACH" class="OCCURRENCE" type="PROCESS"  tense="PAST"
  aspect="NONE" polarity="POS"/>
<TIMEX3 xml:id="t1" target="#range(token2,token3)" pred="LAST_WEEK" type="DATE" value="XXXX-WXX"
 temporalFunction="true" anchorTimeID="t2"/>
<SIGNAL xml:id="s1" target="#token4" pred="on"/>
<TIMEX3 xml:id="t3" target="#token5" pred="MONDAY" type="DATE" value="XXXX-WXX-
1" temporalFunction="true"/>
<TIMEX3 xml:id="t2" type="DATE" value="1996-03-27" functionInDocument="CREATION_TIME"/>
<TLINK eventID="#e1" relatedToTime="#t1" relType="IS_INCLUDED"/>
<TLINK timeID="#t3" signalID="#s1" relatedToTime="#t2" relType="IS_INCLUDED"/>
```

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

7.3.5.2 <SLINK>

The element <SLINK> represents a subordination link between two events that is used for contexts involving modality, evidentials, and factives. An <SLINK> is used in cases where an event instance subordinates another event instance type. These are cases where a verb takes a complement and subordinates the event instance referred to in this complement. The specification is shown in Table 8.

Table 8 — Specification of <SLINK>

eventID	points to the event involved in a temporal link.
subordinatedEvent	points to the event subordinated to an event in a subordination link.
relType	expresses the kind of subordination relation holding between the two events.
comment	remarks made by an annotator.

The following <EVENT> classes interact with <SLINK>: REPORTING, I_STATE or I_ACTION

Some lexical notes:

— Verbs that introduce I_STATE EVENTs that induce <SLINK>: “want”, “desire”, “crave”, “lust”, “believe”, “doubt”, “suspect”, “hope”, “aspire”, “intend”, “fear”, “hate”, “love”, “enjoy”, “like”, “know”;

— Verbs that introduce I_ACTION EVENTs that induce <SLINK>: “attempt”, “try”, “persuade”, “promise”, “name”, “swear”, “vow”.

Examples:

(20) If Graham leaves today, he will not hear Sabine.

```
<SIGNAL xml:id="s1" target="#token0" pred="IF"/>
<EVENT xml:id="e1" target="#token2" pred="LEAVE" class="OCCURRENCE" type="TRANSITION"  tense="PRESENT"
  aspect="NONE" polarity="POS"/>
<TIMEX3 xml:id="t1" target="#token3" pred="TODAY" type="DATE" value="XXXX-XX-XX" temporalFunction="true"/>
<EVENT xml:id="e2" target="#token8" pred="HEAR" class="OCCURRENCE" type="PROCESS" tense="FUTURE"
  aspect="NONE" polarity="NEG"/>
<SLINK eventID="#e1" subordinatedEvent="#e2" signalID="#s1" relType="CONDITIONAL"/>
<TLINK eventID="#e1" relatedToEvent="#e2" relType="BEFORE"/>
```

(21) Bill denied that John taught on Monday.

```
<EVENT xml:id="e1" target="#token1" pred="DENY" class="I_ACTION" type="PROCESS" tense="PAST" aspect="NONE"
  polarity="POS"/>
<EVENT xml:id="e2" target="#token4" pred="TEACH" class="OCCURRENCE" type="PROCESS" tense="PAST"
  aspect="NONE" polarity="POS"/>
<SIGNAL xml:id="s1" target="#token5" pred="ON"/>
<TIMEX3 xml:id="t1" target="#token6" pred="MONDAY" type="DATE" value="XXXX-WXX-1"/>
<TLINK eventID="#e2" signalID="#s1" relatedToTime="#t1" relType="IS_INCLUDED"/>
<SLINK eventID="#e1" subordinatedEvent="#e2" relType="NEG_EVIDENTIAL"/>
```


© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

(22) Bill wants to teach on Monday.

```
<EVENT xml:id="e1" target="#token1" pred="WANT" class="I_STATE" type="STATE" tense="PRESENT" aspect="NONE"
  polarity="POS"/>
<EVENT xml:id="e2" target="#token3" pred="TEACH" class="OCCURRENCE" type="PROCESS" aspect="NONE"
  tense="NONE" polarity="POS"/>
<SIGNAL xml:id="s2" target="#token4" pred="on"/>
<TIMEX3 xml:id="t1" target="#token5" pred="MONDAY" type="DATE" value="XXXX-WXX-1"/>
<TLINK eventID="#e2" signalID="#s2" relatedToTime="#t1" relType="IS_INCLUDED"/>
<SLINK eventID="#e1" subordinatedEvent="#e2" relType="INTENSIONAL"/>
```

(23) John is believed to have lived in Rome.

```
<EVENT xml:id="e1" target="#token2" pred="BELIEVE" class="I_STATE" type="STATE" tense="PRESENT" aspect="
NONE"
  polarity="POS"/>
<EVENT xml:id="e2" target="#token5" pred="LIVE" class="OCCURRENCE" type="STATE" tense="NONE"
  aspect="PERFECTIVE" polarity="POS"/>
<SLINK eventID="#e1" subordinatedEvent="#e2" relType="INTENSIONAL"/>
```

(24) Bill attempted to save her.

```
<EVENT xml:id="e1" target="#token1" pred="ATTEMPT" class="I_ACTION" type="PROCESS" tense="PAST"
  aspect="NONE" polarity="POS"/>
<EVENT xml:id="e2" target="#token3" pred="SAVE" class="OCCURRENCE" type="PROCESS"
  tense="NONE" aspect="NONE" polarity="POS"/>
<SLINK eventID="#e1" subordinatedEvent="#e2" relType="INTENSIONAL"/>
```

7.3.5.3 <ALINK>

The element <ALINK> is an aspectual link; it indicates an aspectual connection between two events. In some ways, it is like a cross between <TLINK> and <SLINK> in that it indicates both a relation between two temporal elements, as well as aspectual subordination. The specification is shown in Table 9.

Table 9 — Specification of <ALINK>

eventID	points to the event involved in a temporal link.
signalID	points, when applicable, to the explicit signal of the temporal relation.
relatedToEvent	points to the entity that is being related to the event if this entity is of an event type.
relType	temporal relation holding between the events.
comment	remarks made by an annotator.

Examples:

(25) The boat began to sink.

```
<EVENT xml:id="e1" target="#token2" pred="BEGIN" class="ASPECTUAL" type="PROCESS" tense="PAST"
  aspect="NONE" polarity="POS"/>
<EVENT xml:id="e2" target="#token4" pred="SINK" class="OCCURRENCE" type="TRANSITION"
  tense="NONE" aspect="NONE" polarity="POS"/>
<ALINK eventID="#e1" relatedToEvent="#e2" relType="INITIATES"/>
```

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

(26) The search party stopped looking for the survivors.

```
<EVENT xmlid="e1" target="#token3" pred="STOP" class="ASPECTUAL" type="TRANSITION"
  tense="PAST" aspect="NONE" polarity="POS"/>
<EVENT xml:id="e2" target="#token4 #token5" pred="LOOK_FOR" class="OCCURRENCE"
  type="PROCESS" tense="NONE" aspect="NONE" polarity="POS"/>
<ALINK eventID="#e1" relatedToEvent="#e2" relType="TERMINATES"/>
```

7.3.5.4 <MLINK>

The abstract syntax specified in *ISO-TimeML* makes a distinction between an interval and the length of the interval (an “amount of time”; see Bunt and Pustejovsky, 2010). The element <MLINK> is a link for measuring the length of a time span or the duration of an event. <MLINK> has the inherent relation type of MEASURES. A temporal expression such as “3 hours” is expressed. The example after Table 10 illustrates the use.

Table 10 —Specification of <MLINK>

eventID	points to the event involved in a temporal link.
signalID	points, when applicable, to the explicit signal of the temporal relation.
relatedToEvent	points to the entity that is being related to the event if this entity is of an event type.
relType	refers to measures.
comment	remarks made by an annotator.

EXAMPLE

(27) John taught for three hours.

```
<EVENT xml:id="e1" target="#token2" pred="TEACH" class="OCCURRENCE" type="PROCESS"
  tense="PAST" aspect="NONE" polarity="POS"/>
<TIMEX3 xml:id="t1" target="#range(token4, token5)" type="DURATION" value="P3H"/>
<MLINK eventID="#e1" relatedToTime="#t1" relType="MEASURES"/>
```

7.3.6 Other tags: <CONFIDENCE>, <CERTAINTY> and <ISO-TimeML>

7.3.6.1 <CONFIDENCE> and <CERTAINTY>

It has been noted to be useful to be able to mark confidence values for parts of an annotation. In general, it is useful to allow confidence values to be assigned to any tag, and, in fact, to any attribute of any tag.

This is conveniently done with the <CONFIDENCE> tag, which consumes no input, and which has the specification shown in Table 11.

Table 11 — Specification of <CONFIDENCE>

tagType	points to any element in *ISO-TimeML*
tagID	points to an element under consideration
attributeName	(OPTIONAL) points to a particular aspect in the element under consideration
confidenceValue	ranges over the rationals between 0 and 1, indicating the degree of confidence
comment	remarks made by an annotator

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

The attribute @tagType would range over the names of all the tags of *ISO-TimeML*. The attribute @tagID would range over the set of actual tag IDs within the current document (XML type URI). The attribute @attributeName would range over the names of all the attributes of all the tags of *ISO-TimeML*. The attribute @confidenceValue would range over the rationals (i.e. would have a floating point value) between 0 and 1.

So, for example, given this annotation:

The TWA flight crash-landed on Easter Island two weeks ago.

```
<EVENT xml:id="e1" target="#token3" pred="CRASHLAND" class="OCCURRENCE" type="TRANSITION"
  tense="PAST" aspect="NONE" polarity="POS"/>
<TIMEX3 xml:id="t1" target="#range(token7,token9)" pred="TWO_WEEKS_AGO" type="DURATION"
  value="P2W" beginPoint="#t2" endPoint="#t3"/>
<TIMEX3 xml:id="t3" type="DATE" value="1999-12-06" temporalFunction="true" anchorTimeID="#t1"/>
 .........
<TIMEX3 xml:id="t2" type="DATE" functionInDocument="CREATION_TIME" value="1999-12-20"/>
<TLINK eventID="#e1" relatedToTime="#t3" relType="IS_INCLUDED"/>
```

If we want to indicate that we are unsure to have annotated "two weeks ago" correctly, we may add:

```
<CONFIDENCE tagType="TIMEX3" tagID="#t1" confidenceValue="0.50"/>
<CERTAINTY target="#t1" degree="0.50"/>
```

Here the lack of the optional attribute @attributeName indicates that the confidence applies to the whole tag.

On the other hand, if we wanted to indicate that we weren't sure if the tense of “crash-landed” was really PAST, we could add this annotation:

```
<CONFIDENCE tagType="EVNT" tagID="#e1" attributeName="TENSE" confidenceValue="0.75"/>
<CERTAINTY target="#t1" degree="0.50"/>
```

Abstracting confidence measures as a separate tag frees the annotation from having to include a @confidenceValue attribute in every tag and eliminates the problem of uncertainty over the exact attribute of a tag the confidence value applies to.

As for how confidence values should be assigned in manual annotation, we feel that, in a large-scale annotation effort such as TIMEBANK, two conditions should be satisfied:

— Fairly high inter-annotator agreement on the tag assignment in the text;

— Ease of use and habitability of the tool from the annotator's perspective.

Therefore, the annotation of a scalar value such as confidence should have at least two features:

The choice of confidence values should be as clearly defined as possible to cover the options; this relates to the granularity and orders of magnitude as well. This would suggest a selection from a small set (e.g. low, mid, high; not_sure, sure, abolutely_sure). These could be interpreted or rescaled to a (0,1] range, if need be, for subsequent inference. There should be a default value specified [at high (=1)] so that it is not necessary to annotate all links and attributes for them with a confidence. The constraint on human annotators to a subset of the possible values should be documented in the annotation guidelines and implemented in the annotation tool. And it would probably be best if the annotation tool did not present numbers but rather natural language descriptions such as those suggested above, which would be represented in the underlying annotation numerically. For example, the annotator might pick “moderately certain”, which would enter the annotation as 0.5.

Moreover, for manual annotation, it does not seem that the 0 and 1 values will be used/useful. Presumably if the annotator does not trust an annotation at all s/he will not add it. And, as was suggested above, 1, at least for manual annotation, should be the default or unmarked value, and so need not be noted, since it would bulk up the files considerably, even if it were used only on entire tags.

Documents
Document Sharing Hub

© ISO 2012 – All rights reserved

Full document details: BS ISO 24617-1:2012

7.3.6.2 The root element <isoTimeML>

Inasmuch as every well-formed XML document must have a single root node, we supply the *<ISO-TimeML>* element as this node. For example, a sample annotated *ISO-TimeML* document might look like this:

Source text:

FAMILIES SUE OVER AREOFLOT CRASH DEATHS

The Russian airline Aeroflot has been hit with a writ for loss and damages, filed in Hong Kong by the families of seven passengers killed in an air crash. All 75 people died on board the Aeroflot Airbus when it ploughed into a Siberian mountain in March 1994.

Tokenized text:

ISO 24611 *Language resource management – Morpho-syntactic annotation framework* (MAF) allows the following tokenization in an in-line format:

```
<maf>
 <token xml:id="token0">FAMILIES</token>
 <token xml:id="token1">SUE</token>
 <token xml:id="token2">OVER</token>
 <token xml:id="token3">AREOFLOT</token>
 <token xml:id="token4">CRAQSH</token>
 <token xml:id="token5">DEATHS</token>
 <token xml:id="token6">The</token>
 <token xml:id="token7">Russian</token>
 <token xml:id="token8">airline</token>
 <token xml:id="token9">Aeroflot</token>
 <token xml:id="token10">has</token>
 <token xml:id="token11">been</token>
 <token xml:id="token12">hit</token>
 <token xml:id="token13">with</token>
 <token xml:id="token14">a</token>
 <token xml:id="token15">writ</token>
 <token xml:id="token16">for</token>
 <token xml:id="token17">loss</token>
 <token xml:id="token18">and</token>
 <token xml:id="token19">damages</token>
 <token xml:id="token20">,</token>
 <token xml:id="token21">filed</token>
 <token xml:id="token22">in</token>
 <token xml:id="token23">Hong</token>
 <token xml:id="token24">Kong</token>
 <token xml:id="token25">by</token>
 <token xml:id="token26">the</token>
 <token xml:id="token27">family</token>
 <token xml:id="token28">of</token>
 <token xml:id="token29">seven</token>
 <token xml:id="token30">passengers</token>
 <token xml:id="token31">killed</token>
 <token xml:id="token32">in</token>
 <token xml:id="token33">an</token>
 <token xml:id="token34">air</token>
 <token xml:id="token35">crash</token>
 <token xml:id="token36">.</token>
 <token xml:id="token37">All</token>
 <token xml:id="token38">75</token>
 <token xml:id="token39">people</token>
 <token xml:id="token40">died</token>
 <token xml:id="token41">on</token>
 <token xml:id="token42">board</token>
```


© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

```
 <token xml:id="token43">the</token>
 <token xml:id="token44">Aeroflot</token>
 <token xml:id="token45">Airbus</token>
 <token xml:id="token46">when</token>
 <token xml:id="token47">it</token>
 <token xml:id="token48">ploughed</token>
 <token xml:id="token49">into</token>
 <token xml:id="token50">a</token>
 <token xml:id="token51">Siberian</token>
 <token xml:id="token52">mountain</token>
 <token xml:id="token53">in</token>
 <token xml:id="token54">March </token>
 <token xml:id="token55">1994</token>
 <token xml:id="token56">.</token>
</maf>

<isoTimeML>
 <EVENT xml:id="e1" target="#token12" pred="HIT" class="OCCURRENCE" type="TRANSITION"
  tense="PRESENT" aspect="PERFECTIVE" polarity="POS"/>
 <EVENT xml:id="e2" target="#token21" pred="FILE" class="OCCURRENCE" type="PROCESS"
  tense="PAST" aspect="NONE polarity="POS"/>
 <EVENT xml:id="e3" target="#token31" pred="KILL" class="OCCURRENCE" type="TRANSITON"
  tense="PAST" aspect="NONE" polarity="POS"/>
 <SIGNAL xml:id="s1" target="#token32"/>
 <EVENT xml:id="e4" target="#token35" pred="CRASH" class="OCCURRENCE" type="TRANSITION"
  tense="NONE" aspect="NONE" polarity="POS"/>
 <EVENT xml:id="e5" target="#token40" pred="DIE" class="OCCURRENCE" type="TRANSITION"
  tense="PAST" aspect="NONE" polarity="POS"/>
 <EVENT xml:id="e6" target="#token41 #token42" pred="ON_BOARD" class="OCCURRENCE" type="STATE"
  tense="NONE" aspect="NONE" polarity="POS"/>
 <SIGNAL xml:id="s2" target="#token46" pred="WHEN"/>
 <EVENT xml:id="e7" target="#token48" pred="PLOUGH" class="OCCURRENCE" type="TRANSITION"
  tense="PAST" aspect="NONE" polarity="POS"/>
 <TLINK eventID="#e7" relatedToEvent="#e5" relType="INCLUDES"/>
 <TLINK eventID="#e5" signalID="#s2" relatedToEvent="#e7" relType="IAFTER"/>
 <TLINK eventID="#e7" signalID="#s3" relatedToTime="#t2" relType="IS_INCLUDED"/>
 <TLINK eventID="e7" relatedToEvent="#e4" relType="IDENTITY"/>
 <SIGNAL xml:id="s3" target="#token53" pred="IN"/>
 <TIMEX3 xml:id="t2"  target="range(#token55,#token56)" pred="MARCH_1994" type="DATE" value="1994-03"/>
  ...
 <TIMEX3 xml:id="t1" type="DATE" value="1996-03-27"/>
 <TLINK eventID="#e1" relatedToTime="#t1" relType="BEFORE"/>
 <TLINK eventID="#e2" relatedToEvent="#e1" relType="BEFORE"/>
 <TLINK eventID="#e3" relatedToEvent="#e2" relType="BEFORE"/>
 <TLINK eventID="#e3" signalID="#s1" relatedToEvent="#e4" relType="IS_INCLUDED"/>
 <TLINK eventID="#e7" signalID="#s3" relatedToTime="#t2" relType="IS_INCLUDED"/>
</isoTimeML>
```

Some further notes concerning changes from version TimeML 1.2 to *ISO-TimeML*:

1. The @nf_morph attribute that was part of <MAKEINSTANCE> has been changed to the attribute @pos (part of speech), and the PASTPART, PRESPART, INFINITIVE, and GERUNDIVE values of nf_morph have been redistributed to the attribute @vform. Both are syntactic, rather than semantic attributes, and therefore belong to another layer of annotation than *ISO-TimeML*. They are considered as optional attributes here. The same holds for the attribute @syntax – see next point.

2. The optional attribute @syntax was added to <SLINK>, <ALINK>, and <TLINK>. The attribute @syntax can be used to hold <CDATA>, but is generally only used by annotation programs to hold the data that led to the creation of the tag.

3. The optional attribute @comment was added to all *ISO-TimeML* elements, for the purpose of giving (human) annotators a place to put observations about annotated text.

© ISO 2012 – All rights reserved

AVA Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

8 Towards a semantics for ISO-TimeML

8.1 Overview

The syntactic apparatus of *ISO-TimeML* touches on several distinct areas of linguistic description. These include temporal semantics, as well as issues of modally subordinating contexts and the semantics of Aktionsarten (event classifications). In this clause, we focus on the semantics of the first of these areas, in particular: the representation of predicates as events; the representation of temporal expressions as intervals or quantification over intervals; and the relations between these interval structures. Besides the interval semantics of *ISO-TimeML* to be presented in clause 8.4, an event-based semantics will also be presented in 8.5.

8.2 Tense and aspect in language

8.2.1 Tense

Tense can be defined as “the grammaticalized expression of location in time” (Comrie, 1986). This grammaticalized expression involves marking, via change of form, of particular syntactic elements, e.g. the verb and auxiliaries. For example, in “John ran a marathon”, the past tense morpheme represented as “-ed” (producing the inflected verb form “ran”) is used to indicate that the event occurred at a time earlier than the speech time. In “John will run a marathon”, the modal auxiliary “will” is used to locate the event as occurring at a future time, i.e. later than the speech time. While tense is mainly marked on the verb and auxiliaries associated with the verb group, in some languages, such as the North American Indian language Nootka (Comrie, 1986), tense is expressed on the noun phrase.

Tense is not the only mechanism for expressing location in time. In languages such as Mandarin Chinese, which lacks tense morphemes, aspectual markers can be used to express location in time, though sometimes even these may be absent (Lin, 2003). There are also non-grammaticalized expressions of location in time given by temporal adverbials, e.g. “tomorrow”, “yesterday”, “two hours later”, etc. In the case of “tomorrow” or “yesterday”, the temporal location is with respect to the speech time. Temporal locations can also of course be expressed relative to a coordinate system given by a calendar, e.g. “1991 (C.E.)”, or a cyclically occurring event, e.g. “morning”, “spring”, or an arbitrary event, e.g. “the day after he married her”.

The few languages that lack tense altogether are not able to distinguish past from present or future. However, they all have a “realis/irrealis” distinction. In Burmese, for example (Comrie 1986[34]), events that are ongoing or that were observed in the past are expressed by sentence-final realis particles “–te”, “-th’a”, “-ta”, and “–hta”. In other cases, i.e. for unreal or hypothetical events (including future events, present events, and hypothetical past events), the sentence-final irrealis particles “–m’e”, “-ma”, and “–hma” are used.

ISO-TimeML expresses four values for the attribute @tense plus the value NONE. These are: FUTURE, PAST, PRESENT, and IMPERFECT.

8.2.2 Aspect

ISO-TimeML makes the traditional linguistic distinction between tense and aspect. The tense attribute values cover all the languages thus far examined by the *ISO-TimeML* Specification Working Group, but obviously there will be languages not covered by the current set.

While tense allows the speaker to relate the time of an eventuality to a deictic centre or some other reference point, grammatical aspect allows the speaker to represent the structure of an eventuality. Here there is a distinction between ***perfective aspect***, where an entire eventuality is presented without its internal temporal structure, e.g. “John built a house”, and ***imperfective aspect***, where the speaker represents internal phases of the eventuality, e.g. “John is building a house”. Perfective aspect can express termination or completion of an eventuality, while imperfective aspect can express the ongoing nature of an activity. It is important to realize that many of the traditional tenses, e.g. Spanish imperfective, as in “Juan leia cuando entre” (“John was reading when I entered”), may combine both tense and aspect, e.g. past and imperfective. The same is true of the ‘complex tenses' in English, such as present progressive, present perfect, etc. Grammatical aspect is expressed in systematic ways across languages, depending on the lexical aspect of the eventuality. The following cross-linguistic account, derived from arguments by Smith (1991), summarizes some of this systematicity.

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

In English and French, ***perfective aspect*** is signalled by verbal tense and aspect morphemes. Termination is expressed in activities, completion is expressed in accomplishments and achievements, and statives can either express termination (e.g. French ***passé composé*** tense morpheme) or not (English, e.g. “I have lived in Paris”). In Mandarin Chinese, which lacks tense markers but which does have the semantic notion of tense (Lin, 2003), the perfective is signalled by morphemes “–le” and “–guo”, usually indicating termination for activities, accomplishments and achievements; completion is indicated by a separate resultative morpheme “–wan”. In Russian, the perfective does not apply to statives, but is signalled by prefixes “po-” (short duration) and “pro-” (unexpected interval) in activities.

The ***imperfective aspect*** is signalled in English by the progressive morpheme “-ing”. It occurs in activities, accomplishments and achievements. In French, as in Russian, it is signalled by tense morphemes (e.g. the French ***imparfait***). In Mandarin, it is signalled by the progressive morpheme “–zai” and resultative morpheme “–zhe”. The particle “-le” can also have an imperfective use with atelic predicates (Lin, 2003).

ISO-TimeML expresses five values for the attribute @aspect, plus the value NONE. These are: PROGRESSIVE, PERFECTIVE, IMPERFECTIVE, PERFECTIVE_PROG, and IMPERFECTIVE_PROG. The values not mentioned here are described in the annexes for those languages introducing them.

8.3 Temporal relations

ISO-TimeML assumes the general framework of Allen's (1984) interval algebra. In Allen's interval algebra, there are 13 basic (binary) interval relations, where six are inverses of the other six, excluding equality.

a. before (b), after (bi); (1)

b. overlap (o), overlappedBy (oi);

c. start (s), startedBy (si);

d. finish (f), finishedBy (fi);

e. during (d), contains (di);

f. meet (m), metBy (mi);

g. equality (eq).

These are shown schematically in Figure 2.

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

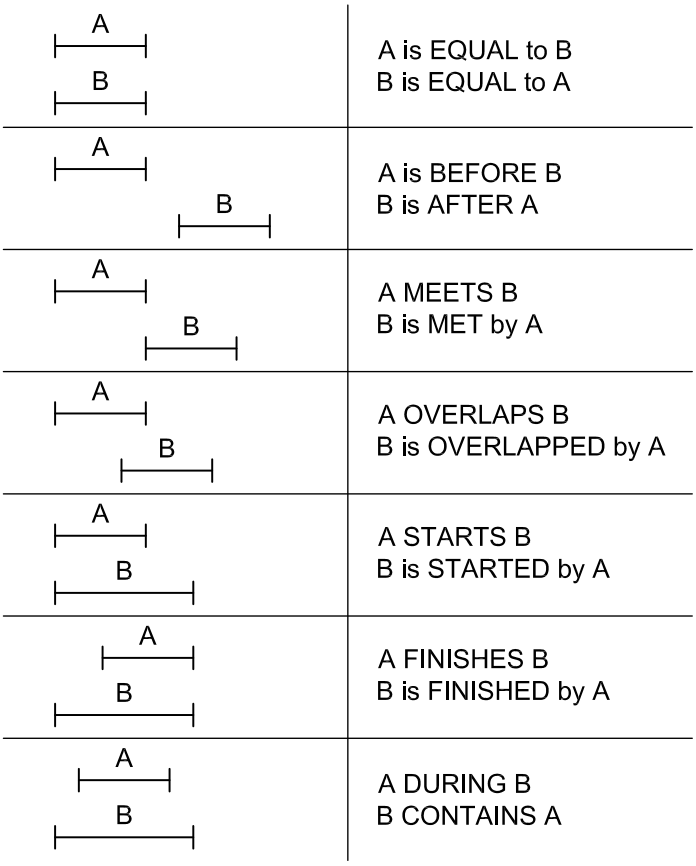

Figure 2 — Allen's interval relations

In the clauses that follow, we will follow the syntax for *ISO-TimeML* by assuming a single type of event class where there is no distinction made in Aktionsarten. Subsequent work within the *ISO-TimeML* community will likely extend the specification to account for differing event types in language, but this is not a necessary component to the current scope of the specification.

8.4 An interval-based semantics for ISO-TimeML

8.4.1 Technical preliminaries for interval temporal logic

We assume the usual apparatus and notation of the Simply Typed Lambda Calculus with primitive types i and t. The type i is identified with the set of non-empty compact, connected subsets of the reals (hereinafter: intervals), while the type t is identified with the set $\{\top, \bot\}$ (hereinafter: Booleans). The primitive types are assumed to be mutually disjoint. We further assume a variety of temporal and logical constants. For example, the constant $\wedge$ of type $(t(tt))$ will be used to represent conjunction in the usual way. Likewise, the constant Q_{EVERY} of type $((it)(it)t)$ will map two properties of intervals to $\top$ just in case every interval satisfying the first satisfies the second. Such constants will be introduced as and when required.

The semantics presented here concerns only a subset of *ISO-TimeML*. This is partly because some of the information recorded in *ISO-TimeML* is essentially syntactic rather than semantic in character, and partly because much of the semantic content of natural language texts has, for one reason or another, no clear rendition within the usual apparatus of formal semantics. Specifically, only three varieties of *ISO-TimeML* elements: <EVENT>, <TIMEX3> and <TLINK> are considered. In addition, only certain slots within these elements are taken to contribute to the semantics.

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

8.4.2 Basic event-structure

8.4.2.1 Sample annotation

Consider the following text:

After his talk with Mary, John drove to Boston. (2)

The *ISO-TimeML* annotation of this text contains a pair of <EVENT> elements and a <TLINK> element:

(3)

```
<EVENT xml:id="e2" target="#token2" pred="TALK"/>
<EVENT xml:id="e1" target="#token7" pred="DRIVE"/>
<TLINK eventID="#e1" relatedToEvent="#e2" relType="AFTER"/>
```

8.4.2.2 Representation in a first-order formula

At the same time, the event-structure of (2) can be captured by the first-order formula

$$p_{\mathrm{DRIVE}}\,(I_{e1}) \wedge p_{\mathrm{TALK}}\,(I_{e2}) \wedge R_{\mathrm{AFTER}}\,(I_{e1}, I_{e2}), \tag{4}$$

where the variables I_{e1} and I_{e2} range over intervals, the unary predicates p_{DRIVE} and p_{TALK} are interpreted as the sets of such intervals over which John drove to Boston and John talked to Mary, respectively, and the binary predicate R_{AFTER} is interpreted as the relation which holds between intervals $[a,b]$ and $[c,d]$ just in case $d < a$. The variables I_{e1} and I_{e2} may be assumed by default to be bound by existential quantifiers that take scope over the whole text; however, this default assumption may be overridden by further *ISO-TimeML* elements, as explained below. Making these global existential quantifiers explicit would add no information, and we do not do so.

Comparing the sample annotation of the two events and their temporal link (3) and its semantic form (4), we see that each attribute pred-value π gives rise to a unary predicate p_{π}, each xml:id-value ei gives rise to a variable I_{ei}, and each attribute relType-value r gives rise to a binary predicate R_{r}.

For compatibility with the more complicated cases treated below, we reformulate the semantics proposed in (4) using the syntax of higher-order logic. Under this regime, all constituents — logical constants, illogical constants and variables — are regarded as functions of (at most) one argument, and the application of a function f to an argument a is denoted $(f\ a)$. In addition, where convenient, we split a conjunction $\phi \wedge \psi$ in higher-order-logic notation: $((\wedge\ \phi)\ \psi)$ into the pair of formulas ϕ and ψ. Making use of this facility for one of the conjuncts in (4) yields the pair of higher-order logic formulas

$$(p_{\pi 1}\, I_{e1}),\ ((\wedge\ (p_{\pi 2}\, I_{e2}))(R_{\mathrm{AFTER}} I_{e1}) I_{e2})). \tag{5}$$

Representations in the style of (5) are easier to generate from *ISO-TimeML* annotations than those in the style of (4); for the sake of readability, however, we continue to give both forms.

8.4.2.3 Interpretation rules

We can generate the formulas (5) as follows. Consider any <EVENT> element with xml:id-value ei and pred-value π. We first create a new variable I_{ei}, of type i, and a new constant p_{π}, of type $(i\ t)$. In addition, we assign to ei a semantic value $\sigma_{ei,}$ which is an expression of higher-order logic featuring I_{ei} and p_{π}. (In the cases encountered in (3), all semantics values have Boolean type, so we can think of them as ordinary formulas.) The interpretation rule for simple <EVENT>-tags such as those encountered in (3) may be given as

<EVENT xml:id="ei" pred="π"/> $\rightarrow \sigma_{ei} := p_{\pi}\, I_{ei}$ (6)

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

Since any xml:id-value ei is allowed to occur in at most one EVENT-tag, these assignments cannot clash.

Suppose, then, that ei is the value of an attribute @xml:id in some element <EVENT>, so that σ_{ei} has been assigned. We take any <TLINK> element with an attribute @relatedToEvent with value ej to modify the expression assigned to σ_{ei}. In the case encountered in (3), this modification results simply in the addition of a conjunct expressing a temporal relation between two events, thus:

<TLINK eventID="#ei" relatedToEvent="#ej" relType="r**"/>** $\rightarrow \sigma_{ei} := ((\wedge\ \sigma_{ei})((R_r I_{ei})I_{ej}))$, (7)

where the R_r is the constant of type $(i(it))$ corresponding to r.

By applying rule (6) and subsequently rule (7) to example (3), we obtain the assignments

$\sigma_{e1} := p_{\pi 1} I_{e1}$
$\sigma_{e2} := ((\wedge\ (p_{\pi 2} I_{e2}))((R_{\mathrm{AFTER}} I_{e1})I_{e2}))$,

Collecting together these assignments yields the two formulas (5), as required.

In general, ei may be the relatedToEvent-value of many <TLINK>-tags such as the one encountered in (3). In that case, the corresponding instances of rule (7) may be applied in any order; the chosen order will not affect the representation of event-structure, modulo logical equivalence.

8.4.2.4 Negated events

We turn now to negated events. Consider

John did not drive to Boston, (8)

which has *ISO-TimeML* mark-up

<EVENT xml:id="e1**" target="**#token3**" pred="**DRIVE**" polarity="**NEG**"/>** (9)

We take (8) to assert that, within some contextually determined interval, no event of John's driving to Boston took place. If we represent our contextually determined interval using the variable I_{e1} (corresponding to the id-value of the relevant <EVENT>), then we can express these truth-conditions using the formula

$$\neg\exists I_1(R_{\mathrm{DURING}}(I_1, I_{e1}) \wedge p_{\mathrm{DRIVE}}(I_1)), \quad (10)$$

where the binary predicate R_{DURING} is interpreted as the relation which holds between intervals $[a,b]$ and $[c,d]$ just in case $c < a$ and $b < d$. Using the notation of higher-order logic, (10) may be re-written

$$(Q_{=0X}\lambda I_1.((\wedge\ ((R_{\mathrm{DURING}}\ I_1)\ I_{e1}))(p_\pi(I_1))), \quad (11)$$

where the unary quantifier $Q_{=0X}$, of type $((it)t)$, has the obvious interpretation: it maps a property of intervals to $\top$ just in case that property has no instances. The formula (11) can be generated from (9) by the interpretation rule

<EVENT xml:id="ei" pred="π" polarity=NEG> (12)
$\rightarrow \sigma_{\mathrm{ei}} := (Q_{=0X}\lambda I_1((I_1),((\wedge((R_{\mathrm{DURING}}\ I_{1})\ I_{ei}))(p_\pi(I_1)))$,

where I_1 is a variable of type i, and p_π is the constant of type (it) corresponding to e.

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

8.4.3 The interpretation of <TIMEX3>

8.4.3.1 Three types of <TIMEX3> elements

The use of the element <TIMEX3> comes in many varieties:

— Unquantified use: <**TIMEX3 xml:id**="t61" **type**="DATE" **value**="2003-06-07"/>

— Frequency: <**TIMEX3 xml:id**="t1" **pred**="EVERY_MONDAY" **type**="SET" **value**="XXXX-WXX-1" **quant**="EVERY"/>

— Quantified use: <**TIMEX3 xml:id**="t3" **pred**="TWICE_A_MONTH" **type**="SET" **value**="P1M" **freq**="2X"/>

Clearly, we cannot expect a uniform semantic treatment of these different uses of the element <TIMEX3>.

8.4.3.2 Unquantified <TIMEX3> element

Let us begin with the simplest case. Sometimes, <TIMEX3> can be treated analogously to <EVENT>. Consider, for example,

John drove to Boston on Saturday, 31st January, 2004. (13)

This text may be marked up as

(14)

```
<EVENT xml:id="e1" target="#token1" pred="DRIVE"/>
<TIMEX3 xml:id="t1" target="#range(token5,token10)" value="2004-01-31"/>
<TLINK eventID="#e1" relatedToTime="#t1" relType="DURING"/>
```

and its event-structure may be represented by the formula

$$p_{\text{2004-01-31}}(I_{t1}) \wedge p_{\text{DRIVE}}(I_{e1}) \wedge R_{\text{DURING}}(I_{e1},I_{t1}), \quad (15)$$

where the variable I_{t1} ranges over intervals, and the unary predicate $p_{\text{2004-01-31}}$ is interpreted as the (singleton) set of time-intervals coinciding with the 31st January, 2004. Re-writing (15) in the style of (5), we obtain

$$(p_{\text{2004-01-31}}(I_{t1})),((\wedge\ (p_{\text{DRIVE}}\ I_{e1}))((R_{\text{DURING}}\ I_{e1})I_{t1})). \quad (16)$$

This example suggests the following interpretation process. Given a <TIMEX3> element with @xml:id-value t, we create a new variable I_t of type t, and we assign the semantic value σ_t according to the rule

<**TIMEX3 xml:id**="t" **value**="v"/> (17)
$\rightarrow \sigma_t := (p_v I_t)$

where p_v is the constant, of type (it), corresponding to v.

In addition, we interpret the <EVENT> element in (14) using the rule (6), and the <TLINK> element using the following generalization of rule (7):

<TLINK (18)
{eventID|timeID}=x
{relatedToEvent|relatedToTime}=y
relType=r />
$\rightarrow \sigma_x := ((\wedge\ \sigma_x))((R_r I_x)I_y))$.

These rules generate the representation of event-structure (16) from (14), exactly as for example (2).

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

8.4.3.3 Quantifying <TIMEX3> element

Other <TIMEX3> constructs make a more complex semantic contribution. Consider, for example,

John drove to Boston twice. (19)

which has *ISO-TimeML* mark-up

(20)

```
<EVENT xml:id="e1" target="#token1" pred="DRIVE"/>
<TIMEX3 xml:id="t1" target="#token4" pred="TWICE" freq="2X"/>
<TLINK eventID="#e1" relatedToTime="#t1" relType="DURING"/>
```

Assuming that “twice” here has the reading “at least twice”, we take (19) to assert that, within some contextually determined interval, at least two events of John's driving to Boston took place. If we represent this interval using the variable I_{t1} (corresponding to the xml:id-value of the relevant <TIMEX3> element), then we can express these truth-conditions using the formula

$$\exists_{e2} I_{e1}(R_{\mathrm{DURING}}(I_{e1}, I_{t1}) \wedge p_{\pi}(I_{e1})), \tag{21}$$

where, as before, the predicate p_{DRIVE} is true of precisely those intervals over which John drives to Boston. Note that the variable I_{e1} arising from the EVENT-tag is bound by a quantifier in (21).

In the notation of higher-order logic, (21) becomes

$$(Q_{2\mathrm{X}} \lambda I_{e1}.((\wedge\ ((R_{\mathrm{DURING}}\ I_{e1})\ I_{t1}))(p_{\pi}(I_{e1}))), \tag{22}$$

where $Q_{2\mathrm{X}}$ is the function mapping a property of intervals to $\top$ just in case it has at least two instances.

To generate (22) from the *ISO-TimeML*-tags in (20), we adopt the rule

$$\text{<TIMEX3 xml:id=t freq=q />} \rightarrow \lambda R \lambda p.(Q_{\mathrm{q}} \lambda I_1.((\wedge\ ((R I_1)\ I_t))(p\ (I_1))), \tag{23}$$

where R is a variable of type $(i(it))$, p a variable of type (it), I_1 a variable of type i, Q_{q} the constant of type $((it)t)$ corresponding to q, and I_t the variable of type i corresponding to t. We also adopt the rule

```
<TLINK                                                    (24)
  {eventID|timeID}=x
  {relatedToEvent|relatedToTime}=y
  relType=r />
```

$$\rightarrow \sigma_y := ((\sigma_y R_{\mathrm{r}}) \lambda I_x.\ \sigma_x)$$

$$\sigma_x := \text{undefined}$$

where R_r is the constant of type $(i(it))$ corresponding to r, and I_x the variable of type i corresponding to x.

Let us work through the example (20) to see how these rules produce the event-structure representation (21). First,

```
<EVENT xml:id="e1" pred="DRIVE"/>                         (25)
```

is processed by rule (6), yielding the assignment

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

$$\sigma_{e1} := (p_{\text{DRIVE}}\, I_{e1}), \tag{26}$$

while

<**TIMEX3 xml:id**="t1" **freq**="2X"/> (27)

is processed by rule (23), yielding the assignment

$$\sigma_{t1} := \lambda R \lambda p.(Q_{2\text{X}} \lambda I_{1.}((\wedge\ ((RI_1)\ I_{t1}))(p\ (I_1))). \tag{28}$$

Note that (28) is a function which takes a relation between intervals as argument, and returns a function which itself takes a property of intervals as argument. These assignments having been made,

<**TLINK eventID**="#e1" **relatedToTime**="#t1" **relType**="DURING"/> (29)

is processed by rule (24) to yield the re-assignment

$$\sigma_{t1} := ((\lambda R \lambda p.(Q_{2\text{X}} \lambda I_{1.}((\wedge\ ((RI_1)\ I_{t1}))(p\ I_1)))R_{\text{DURING}})\lambda I_{e1.}(p_{\text{DRIVE}}\, I_{e1})), \tag{30}$$

and the de-assignment of σ_{e1}. Thus, rule (24) supplies arguments for the function in (28), and gets rid of the assignment in (26) altogether. Routine calculation shows that (30) normalizes to (22); and since, following the de-assignment of σ_{e1}, there are no other semantic values to consider, this is the final semantics for (20), as required.

There is one further matter to discuss in relation to this example. As part of the process of de-assigning σ_{e1}, we assume that $e1$ is made inaccessible to any other TLINK-tags than the one which caused rule (24) to be applied. That is, no other TLINK-tags may have $e1$ as the value of any of the slots @eventID, @timeID, @relatedToEvent or @relatedToTime. This inaccessibility reflects the fact that we cannot meaningfully use a TLINK-tag to relate $e1$ to any other xml:id-value. After all, if such a TLINK-tag did involve $e1$, which of the two (or more) events of John's driving to Boston would the relation involve? *ISO-TimeML*-marked-up text violating this accessibility constraint is considered uninterpretable.

8.4.3.4 Quantifying <TIMEX3> element with unquantified complements

Moving on to a different kind of <TIMEX3> construct, consider

John drove to Boston every Monday. (31)

This text may be marked up as

(32)

<**EVENT xml:id**="e1" **target**="#token1" **pred**="DRIVE"/>
<**TIMEX3 xml:id**="t1" **pred**="EVERY_MONDAY" **type**="SET" **value**="XXXX-WXX-1" **quant**="EVERY"/>
<**TLINK eventID**="#e1" **relatedToTime**="#t1" **relType**="DURING"/>

and its event-structure may be represented by the formula

$$\forall I_1((p_{\text{XXXX-WXX-1}}(I_1) \wedge R_{\text{DURING}}(I_1, I_{t1})) \rightarrow \exists I_{e1}(p_{\text{DRIVE}}(I_{e1}) \wedge R_{\text{DURING}}(I_{e1}, I_1))), \tag{33}$$

where the free variable I_{t1} represents some contextually given interval to which the universal quantification in (31) is confined, the unary predicate $p_{\text{XXXX-WXX-1}}$ is satisfied by precisely those intervals which coincide with Mondays, and the predicates p_{DRIVE} and R_{DURING} are as above. Thus, we take (31) to exhibit two levels of quantification: explicit universal quantification over Mondays, and implicit existential quantification over events within each of those Mondays.

In the notation of higher-order logic, (33) may be written

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

$$((Q_{\text{EVERY}}\lambda I_1.((\wedge\ (p_{\text{XXXX-WXX-1}}\ I_1))((R_{\text{DURING}}\ I_1)\ I_{t1})))\lambda I_2.(Q_{1X}\lambda I_{e1}.((\wedge\ (p_{\text{DRIVE}}\ I_{e1}))((R_{\text{DURING}}\ I_{e1})I_2)))). \quad (34)$$

Here, Q_{EVERY} is the constant of type $((it)((it)t))$ expressing the binary universal quantifier, and Q_{1X} the constant of type $((it)t)$ expressing the unary existential quantifier. That is, for all p, Q of type (it): $((Q_{\text{EVERY}}\ p)Q) = T$ if and only if, for every interval I, $(pI) = T$ implies $(QI) = T$; and $(Q_{1X}p) = T$ if and only if, for some interval I, $(pI) = T$.

The question now is how to recover (34) from the *ISO-TimeML* tags in (32). We propose the rule

<TIMEX3 xml:id="t" **type**="SET" **value**="Rv" **quant**="q"/> (35)

$$\rightarrow \sigma_t := \lambda R\lambda p.((Q_{\text{q}}\lambda I_1.((\wedge\ ((R_{\text{DURING}}I_1)I_t))(vI_1)))\lambda I_2.(Q_{1\text{X}}\lambda I_3.((\wedge\ ((RI_3)I_2))(pI_3)))).$$

With the above rules at our disposal,

<EVENT xml:id="e1" target="#token1" pred="DRIVE"/> (36)

is processed by rule (6), yielding the assignment

$$\sigma_{e1} := (p_{\text{DRIVE}}\ I_{e1}), \quad (37)$$

while

<TIMEX3 xml:id="t1" **pred**="EVERY_MONDAY" **type**="SET" **value**="XXXX-WXX-1" **quant**="EVERY"/> (38)

is processed by rule (35), yielding the assignment

$$\sigma_{t1} := \lambda R\lambda p.((Q_{\text{EVERY}}\lambda I_1.((\wedge\ ((R_{\text{DURING}}\ I_1)I_{t1}))(p_{\text{XXXX-WXX-1}}\ I_1)))\lambda I_2.(Q_{1\text{X}}\lambda I_3.((\wedge\ ((RI_3)I_2))(pI_3)))). \quad (39)$$

These assignments having been made,

<TLINK eventID="#e1" **relatedToTime**="#t1" **relType**="DURING"/> (40)

is processed by rule (24), yielding the re-assignment

$$\sigma_{t1} := ((\lambda R\lambda p.((Q_{\text{EVERY}}\lambda I_1.((\wedge\ ((R_{\text{DURING}}\ I_1)I_{t1}))(p_{\text{XXXX-WXX-1}}\ I_1)))\ \lambda I_2.(Q_{1\text{X}}\lambda I_3.((\wedge\ ((RI_3)I_2))(pI_3))))R_{\text{DURING}})\lambda I_{e1}.(p_{\text{DRIVE}}\ I_{e1})), \quad (41)$$

and the de-assignment of σ_{e1}. But (41) normalizes to (34); and since there are no other semantic values to consider, this is the final semantics for (32), as required.

In some texts, the implicit existential quantification over events of the kind exhibited in (31) is made explicit. Consider, for example,

John drove to Boston twice a month. (42)

(Again, we assume that “twice” here has the reading “at least twice”.) This text may be marked up as

(43)

<EVENT xml:id="e1" **target**="#token1" **pred**="DRIVE"/>
<TIMEX3 xml:id="t1" **target**="#range(token4,token6)" **type**="SET" **value**="month"
quant="EVERY" **freq**="2X"/>
<TLINK eventID="#e1" **relatedToTime**="#t1" **relType**="DURING"/>

and its event-structure may be represented by the formula

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

$$\forall J((p_{\text{MONTH}}(J) \wedge R_{\text{DURING}}(J, I_{t1})) \rightarrow \exists_{\geq 2} I_{e1}(p_{\text{DRIVE}}(I_{e1}) \wedge R_{\text{DURING}}(I_{e1}, J))), \quad (44)$$

where the variable I_{t1} represents the interval to which universal quantification over months is restricted, and the unary predicate p_{MONTH} is true of exactly those intervals which are (calendar) months. Here we have two levels of quantification: universal quantification restricted to I_{t1}, and "at-least-twice" quantification restricted to calendar months included in I_{t1}. The universal quantification is somewhat obliquely signalled by the English "a month", but is encoded directly in the quant-value of the TIMEX3-tag.

We may re-write (44) using the notation of higher-order logic as

$$((Q_{\text{EVERY}}\lambda I_1.((\wedge\ (p_{\text{MONTH}}\, I_1))((R_{\text{DURING}}\, I_1)I_{t1})))\lambda I_2.(Q_{2\text{X}}\lambda I_{e1}.((\wedge\ (p_{\text{DRIVE}}\, I_{e1}))((R_{\text{DURING}}\, I_{e1})I_2)))). \quad (45)$$

This suggests the interpretation rule

<TIMEX3 xml:id="t" type="SET" value="Rv" quant="q" freq="q′ "/> (46)

$$\rightarrow \sigma_t := \lambda R\lambda p.((Q_q\lambda I_1.((\wedge\ ((R_{\text{DURING}}\, I_1)I_t))(vI_1)))\lambda I_2.(Q_{q'}\lambda I_3.((\wedge\ ((RI_3)I_2))(pI_3)))).$$

with other tags interpreted as before.

It may be verified that, after applying these rules and normalizing, σ_{t1} is assigned the expression (45), while σ_{e1} is de-assigned. Since there are no other semantic values to consider, (45) — or its more readable equivalent, (44) — is the meaning of the *ISO-TimeML* tags in (43).

8.4.3.5 Quantifying <TIMEX3> elements with quantified complements

In example (20), the EVENT-tag, corresponding to "John drove to Boston" contributes an unquantified expression ($p_{\text{DRIVE}}\, I_{e1}$), while the TIMEX3-tag, corresponding to the adverbial "twice", contributes a quantifier Q_{2x}, binding the variable I_{e1}. Similarly, in example (32), the EVENT-tag again contributes the unquantified expression ($p_{\text{DRIVE}}\, I_{e1}$), while the TIMEX3-tag, corresponding to the adverbial "every Monday", contributes an (implicit) existential quantifier Q_{1x}, again binding the variable I_{e1}. In both these cases, a single temporal adverbial provides a quantifier binding the variable in an unquantified event-reporting expression.

It can also happen, however, that a temporal adverbial modifies an expression which has already been explicitly quantified by another adverbial. Since both adverbials will be marked with TIMEX3-tags, additional rules are required to interpret these tags. Consider, for example,

One Monday, John drove to Boston twice, (47)

where the adverbial "One Monday" modifies the explicitly quantified John drove to Boston twice. This text may be marked up as

(48)

```
<TIMEX3  xml:id="t2" target="#token0 #token1" pred="ONE_MONDAY"
 type="SET" value="XXXX-WXX-1" quant="SOME"/>
<EVENT xml:id="e1" target="token4" pred="DRIVE"/>
<TIMEX3  xml:id="t1" target="token7" pred="TWICE" freq="2X"/>
<TLINK eventID="#e1" relatedToTime="#t1" relType="DURING"/>
<TLINK xml:id="t1" relatedToTime="#t2" relType="DURING"/>
```

with the two TLINK-tags forming, so to speak, a cascade: e1→ t1→ t2. The event-structure of (47) may be represented by the formula

$$\exists I_1(p_{\text{XXXX-WXX-1}}(I_1) \wedge R_{\text{DURING}}(I_1, I_{t2})) \rightarrow \exists_{\geq 2} I_{e1}(p_{\text{DRIVE}}(I_{e1}) \wedge R_{\text{DURING}}(I_{e1}, I_1))), \quad (49)$$

or, in higher-order logic notation,

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

$$((Q_{\text{SOME}}\lambda I_1.((\wedge\ (p_{\text{XXXX-WXX-1}} I_1))((R_{\text{DURING}}\, I_1)I_{t2})))\lambda I_2.(Q_{\text{2X}}\lambda I_{e1}.((\wedge\ (p_{\text{DRIVE}}\, I_{e1}))((R_{\text{DURING}}\, I_{e1})I_2)))), \quad (50)$$

with the binary quantifier Q_{SOME} interpreted in the obvious way.

To see how (50) may be generated from (48), let us split the latter into

(51)

```
<EVENT xml:id="e1" target="#token4" pred="DRIVE"/>
<TIMEX3 xml:id="t1" target="#token7" pred="TWICE" freq="2X"/>
<TLINK eventID="#e1" relatedToTime="#t1" relType="DURING"/>
```

and

(52)

```
<TIMEX3 xml:id="t2" target="#token0 #token1" pred="ONE_MONDAY"
 type="SET" value="XXXX-WXX-1" quant="SOME"/>
<TLINK xml:id="t1" relatedToTime="#t2" relType="DURING"/>
```

Now, (51) is identical to (20), and will therefore be given the same semantics. Crucially, the expression (22) thereby assigned to σ_{t1} already quantifies over events; therefore, we should not proceed by analogy with (31), where the TIMEX3-tag is mapped to an expression providing implicit existential quantification. Instead, we propose the additional interpretation rules

<TIMEX3 xml:id="t" type="SET" value="Rv" quant="q"/> (53)

$\rightarrow \sigma_t := (Q_q\lambda I_1.((\wedge\ ((R_{\text{DURING}}\, I_1)I_t))(vI_1)))$

and

<TLINK (54)
{eventID|timeID}=x
{relatedToEvent| relatedToTime}=y
relType="DURING"/>
$\rightarrow \sigma_t := (\sigma_y\lambda I_x.\sigma_x)$ where σ_x := undefined.

Let us work through the example (48) to see how these rules produce the event-structure representation (50). We have already seen how the tags in (51) generate the assignments:

$$\sigma_{t1} := (Q_{2X}\lambda I_3.((\wedge\ ((R_{\text{DURING}} I_{e1})I_{t1}))(p_{\text{DRIVE}}\, I_{e1}))) \text{ where } \sigma_{e1} = \text{undefined}. \quad (55)$$

Turning therefore to the tags in (52), rule (53) results in the assignment

$$\sigma_{t2} := (Q_{\text{SOME}}\lambda I_1.((\wedge\ ((R_{\text{DURING}} I_1)I_{t2}))(p_{\text{XXXX-WXX-1}} I_1))), \quad (56)$$

and rule (54) results in the re-assignment

$$\sigma_{t2} := (Q_{\text{SOME}}\lambda I_1.((\wedge\ ((R_{\text{DURING}} I_1)I_{t2}))(p_{\text{XXXX-WXX-1}} I_1)))\lambda I_{t1}.(Q_{\text{2X}}\lambda I_{e1}.((\wedge\ ((R_{\text{DURING}} I_{e1})I_{t1}))(p_{\text{DRIVE}}\, I_{e1})))), \quad (57)$$

and the de-assignment of σ_{t2}. But (57) is identical to (50) up to variable renaming. And since there are no other semantic values to consider, this is the final semantics for (48), as required.

8.4.4 Interpretive rule summary

We summarize the interpretation rules employed above.

Rule 1: <EVENT xml:id="ei" pred="π"/>
$\rightarrow \sigma_{ei} := (p_\pi I_{ei})$

Rule 2: <EVENT xml:id="ei" pred="π" polarity="NEG" />
$\rightarrow \sigma_{ei} := (Q_{=0X}\lambda I_1.((\wedge((R_{\text{DURING}}\, I_1)I_{\text{ei}}))(p_\pi(I_1)))$

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

Rule 3: <TIMEX3 xml:id=t freq=q />

$\rightarrow \sigma_t := \lambda R \lambda p.((Q_q \lambda I_1.((\wedge\ ((R\ I_1)I_t))(pI_1)))$

Rule 4a: <TIMEX3 xml:id=t type="SET" value=R_v quant=q />

$\rightarrow \sigma_t := \lambda R \lambda p.((Q_q \lambda I_1.((\wedge\ ((R_{\text{DURING}}\ I_1)I_t))(vI_1)))\lambda I_2.(Q_1 \text{x} \lambda I_3.((\wedge\ ((RI_3)I_2))(pI_3))))_{ei}$

Rule 4b: <TIMEX3 xml:id=t type="SET" value=R_v quant=q />

$\rightarrow (Q_q \lambda I_1.((\wedge\ ((R_{\text{DURING}}\ I_1)I_t))(vI_1)))$

Rule 5: <TLINK
{eventID | timeID}=x
{relatedToEvent | relatedToTime}=y
relType=r />

$\rightarrow \sigma_x := ((\wedge\ \sigma_x)((R_r I_x)I_y))$

Rule 6: <TLINK
{eventID|timeID}=x
{relatedToEvent| relatedToTime}=y
relType=r/>

$\rightarrow \sigma_y := ((\sigma_y R_r)\lambda I_x.\sigma_x)$ where σ_x := undefined.

Rule 7: <TLINK
{eventID | timeID}=x
{relatedToEvent | relatedToTime}=y
relType="DURING">

$\rightarrow \sigma_y := (\sigma_y \lambda I_x.\sigma_x)$ where σ_x := undefined.

There is a general assumption that rules may only apply if the relevant types match.

8.5 An event-based semantics for ISO-TimeML

8.5.1 Introduction

Besides the interval-based semantics of the XML representation of annotation structures, described in 8.4, the *ISO-TimeML* language also has an event-based semantics associated with its abstract syntax. Just like the interval-based semantics provides an interpretation of annotation representations by translating them into the language of Interval Temporal Logic (ITL), the event-based semantics defines a mapping from the annotation structures defined by the abstract syntax into a first-order logic which has a richer ontology than ITL, including events and relations between them, and interval lengths (durations). Given that first-order logic has a well-established compositional formal semantics, this approach defines a compositional semantics for *ISO-TimeML* at the level of annotations. This semantic specification can be instantiated for each representation format, as illustrated for the XML representation format described in Clause 7, by Bunt & Overbeeke (2008). It includes a semantics for tenses, aspect, durations, subordination (‘SLINK') structures and aspectual (‘ALINK') structures, which are not covered by the interval-based semantics.

Before presenting the formal details of these semantics, we illustrate the way it works with a simple example. Consider the sentence “John started to read at half past ten”. The *ISO-TimeML* annotation structure for this sentence would be as follows, where we identify markables by the corresponding text segments in the sentence, and use self-explanatory names for particular elements of the conceptual inventory.

Entity structures:

— event structure for the “start” event: <“started”, <process,past,inchoative,individual,positive>>

— event structure for the “read” event: <“read”, <process, --, --, individual, positive>>

— instant structure for “half past ten”: <“10:30”, <CET, <2008, --, -->, 1030>>>

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

Link structures:

— subordination structure relating the two events:

<<“started”, <process,past,inchoative,individual,positive>>,

<“read”, <process, --, --, individual, positive>>, INITIATES>

— temporal anchoring structure relating the start event to the temporal instant:

<<“started”, <process,past,inchoative,individual,positive>>,

<“10:30”, <CET, <2008, --, -->, 1030>>>, AT>

The semantics of this annotation structure can be computed by combining the first-order logic representations of the subordination structure and the temporal anchoring structure (see Bunt & Overbeeke, 2008 for how to do this), with the result:

$\exists e_1.\ PROCESS(e_1) \wedge \exists t_1.\ calyear(CET, t_1)=2008 \wedge clocktime(CET(t_1)=1030 \wedge EV\text{-}TIME(CET(e_1))= t_1 \wedge$
$\exists e_2.\ INITIATES(e_1,e_2)$

8.5.2 Defining an event-based semantics

8.5.2.1 General

The following subclauses define an interpretation function *Infl* as a mapping from annotations to first-order logic with lambda abstraction (and real numbers). In the left hand side of the definition clauses, capital letters are used to designate elements from the conceptual inventory, and small letters to designate annotation structures.

8.5.2.2 Elements from the conceptual inventory

— To every event class C_i, tense T_j, and aspect A_m the interpretation function *Infl* assigns a predicate constant which is indicated by a quote ', e.g. *Infl*(T_j) = T_j'.

— Polarities and set-theoretic types as such are not represented in the first-order logic representations, but are interpreted through the stipulation of different interpretations of event annotations, depending on the values of these elements.

— To every temporal relation, temporal measurement function, duration relation, numerical relation, event subordination relation, and aspectual relation the interpretation function assigns a predicate constant which is indicated by a single quote (').

— To every time zone Z a function constant Z' is assigned; (functions that map the time line onto pairs consisting of a date and a clock time).

— To all calendar years, calendar months, calendar day numbers, and clock times the interpretation function assigns their usual numerical string name (such as “20090229” and “1245”).

— For every real number N, *Infl*(N) will be its usual string name (such as “5”).

— To every temporal unit, the interpretation function assigns an individual constant which is indicated by a quote ', *Infl*(u) = u'.

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

8.5.2.3 Event annotations

— *Infl*(<C, T, A, *indiv*, *pos*>) = $\lambda P.\ \exists e.\ C'(e) \wedge T'(e) \wedge A'(e) \wedge P(e)$

— *Infl*(<C, T, A, *indiv*, *neg*>) = $\lambda P.\ \neg\exists e.\ C'(e) \wedge T'(e) \wedge A'(e) \wedge P(e)$

— *Infl*(<C, T, A, *set*, *pos*>) = $\lambda P.\ \exists E.\ \forall e \in E.\ C'(e) \wedge T'(e) \wedge A'(e) \wedge P(e)$

— *Infl*(<C, T, A, *set*, *neg*>) = $\lambda P.\ \neg\exists E.\ \forall e \in E.\ C'(e) \wedge T'(e) \wedge A'(e) \wedge P(e)$

— *Infl*(<C, T, A, *set*, N, *pos*>) = $\lambda P.\ \exists_N e.\ C'(e) \wedge T'(e) \wedge A'(e) \wedge P(e)$

— *Infl*(<C, T, A, *indiv*, N, P_N, *pos*>) = $\lambda P.\ \forall I.\ Infl(P_N)(I) \rightarrow \exists_N e.\ C'(e) \wedge T'(e) \wedge A'(e) \wedge P(e)$

— *Infl*(<C, T, A, *set*, N, *neg*>) = $\lambda P.\ \neg\exists_N e.\ C'(e) \wedge T'(e) \wedge A'(e) \wedge P(e)$

— *Infl*(<C, T, A, *set*, N, P_N, *neg*>) = $\lambda P.\ \neg\forall I.\ Infl(P_N)(I) \rightarrow \exists_N e.\ C'(e) \wedge T'(e) \wedge A'(e) \wedge P(e)$

8.5.2.4 Interval annotations

— *Infl*(<t_1,t_2>) = $\lambda P.\ \exists T.\ Infl(t_1)(\lambda x.\ start(T) = x) \wedge Infl(t_1)(\lambda y.\ end(T) = y) \wedge P(T)$

— *Infl*(<<N,U>, T_1, R>) = $\lambda P.\ \exists T.\ \exists T'.\ end(T)=start(T') \wedge end(T)=start\ (T_1) \wedge length(T',U')=N' \wedge P(T)$

— *Infl*(<<t_1,t_2>,R>) = $\lambda P.\ \exists T.\ R'(T,\ t_1,t_2) \wedge P(T)$

8.5.2.5 Instant annotations

— *Infl*(<Z,d,T>) = $\lambda P.\ \exists t.\ date(Z'(t)) = Infl(d) \wedge clocktime(Z'(t)) = T' \wedge P(t)$

— *Infl*(<<N,U>,t_1>) = $\lambda P.\ \exists t.\ \exists T.\ t=start(T) \wedge t_1=end(T) \wedge length(T,U')=N' \wedge P(t)$

8.5.2.6 Time-amount annotations

— *Infl*(<N,U>) = $\lambda x.\ length(x,U')=N'$

— *Infl*(<R,N,U>) = $\lambda x.\ R'(length(x,U'),N')$

8.5.2.7 Temporal relation annotations

— *Infl*(R) = $\lambda R'$

8.5.2.8 Temporal anchoring structures

— *Infl*(<e,t,R>) = $\lambda x.\ \lambda T.\ R'(EV\text{-}TIME(e),T)$

8.5.2.9 Temporal relation structures

— *Infl*(<e_1,e_2,R_t>) = $\lambda e.\ \lambda e'.\ R_t'(e,\ e')$

— *Infl*(<t_1,t_2,R>) = $\lambda t.\ \lambda t'.\ R'(t,\ t')$

© ISO 2012 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012

8.5.2.10 Time measurement structures

— *Infl*(<t,<N,U>) = $\lambda T.$ *length*(T,U')=N'

— *Infl*(<e,<N,U>) = $\lambda T.$ *length*(*EV-TIME*(e),U')=N'

8.5.2.11 Subordination structures

— *Infl*(<e_1,e_2,R_s>) = $\lambda e.\ \lambda e'.\ R_s'(e,\ e')$

8.5.2.12 Aspectual structures

— *Infl*(<e_1,e_2,A>) = $\lambda e.\ \lambda e'.\ A'(e,\ e')$

8.5.2.13 Formal semantics of these representations

These first-order logic representations have a formal semantics which makes use of a model

M = <D,F>

where

D is the model structure, which is an octet D = <C_z, T, F_T, U, C_U, Z_T, *date, clocktime*>;

where

C_z is a set of event classes (such as *Process*, *Transition*, *State*);

T is an 8-tuple <Π, $\leq_T$, *start, end, cY, cM, cD, cT*> where Π is an infinite set of time points with a total ordering $\leq_T$; *start* and *end* are functions from temporal intervals to the time points defining their beginning and end; *cY, cM,* and *cD* are sets of intervals in T (calendar years, calendar months, calendar days, and clock times respectively); *cT* is a structure *cT* = < *clH, clM,..* > (of clock hours and minutes, extended, if necessary, with seconds, milliseconds, etc.);

U is a finite set of units for temporal measurement;

F_T is a set of temporal functions;

C_U is a pair consisting of a function from pairs of temporal units to real numbers, i.e. a conversion function between temporal units, and a function *length* that computes the length of a temporal interval, given a temporal unit (i.e. a function from pairs consisting of a temporal interval and a temporal unit to real numbers);

length is a function assigning a numerical value to a pair, consisting of a temporal interval and a temporal unit;

Z_T (time zones) is a set of functions from T to pairs of dates and times; event occurs at some place, the notion of time zones can also be applied to events); day numbers, and clock times, respectively;

date and *clocktime* are functions projecting each temporal value, defined by a time zone, onto its date and its clock time, respectively;

F is the interpretation function, which assigns elements of the model structure to the predicates in the first-order representations of annotation structures. The interpretation rules for first-order logic expressions are as usual.

© ISO 2012 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO 24617-1:2012