A discourse is a sequence of sentences: a text or a conversation
A discourse is made of words or phrases that refer to things: the discourse entities
A discourse normally links the entities together to address topics
Within a single sentence, grammatical structures provide with a model of relations between entities.
Discourse models extend relations to more sentences
Discourse entities – or discourse referents – are the real, abstract, or imaginary objects introduced by the discourse. **Referring expressions** are mentions of the discourse entities through the text.

1. *Susan* drives a *Ferrari*
2. *She* drives too fast
3. *Lyn* races *her* on weekends
4. *She* often beats *her*
5. *She* wins a lot of trophies
## Discourse Entities

<table>
<thead>
<tr>
<th>Mentions (or referring expressions)</th>
<th>Discourse entities (or referents)</th>
<th>Logic properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Susan, she, her</td>
<td>’Susan’</td>
<td>’Susan’</td>
</tr>
<tr>
<td>Lyn, she</td>
<td>’Lyn’</td>
<td>’Lyn’</td>
</tr>
<tr>
<td>A Ferrari</td>
<td>X</td>
<td>ferrari(X)</td>
</tr>
<tr>
<td>A lot of trophies</td>
<td>E</td>
<td>$E \subseteq {X \mid \text{troy}(X)}$</td>
</tr>
</tbody>
</table>
Named entities are entities uniquely identifiable by their name.

Some definitions/clarifications:

- Named entity recognition (NER): a partial parsing task, see Chap. 10;
- Reference resolution for named entities: find the entity behind a mention, here a name.

As it is impossible to set a physical link between a real-life object and its mention, we use unique identifiers or tags in the form of URIs instead (from Wikidata, DBpedia, Yago).

<table>
<thead>
<tr>
<th>Words</th>
<th>POS</th>
<th>Groups</th>
<th>Named entities</th>
</tr>
</thead>
<tbody>
<tr>
<td>U.N.</td>
<td>NNP</td>
<td>I-NP</td>
<td>I-ORG</td>
</tr>
<tr>
<td>official</td>
<td>NN</td>
<td>I-NP</td>
<td>O</td>
</tr>
<tr>
<td>Ekeus</td>
<td>NNP</td>
<td>I-NP</td>
<td>I-PER</td>
</tr>
<tr>
<td>heads</td>
<td>VBZ</td>
<td>I-VP</td>
<td>O</td>
</tr>
<tr>
<td>for</td>
<td>IN</td>
<td>I-PP</td>
<td>O</td>
</tr>
<tr>
<td>Baghdad</td>
<td>NNP</td>
<td>I-NP</td>
<td>I-LOC</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>O</td>
<td>O</td>
</tr>
</tbody>
</table>
Cambridge: England, Massachusetts, or Ontario? Saussure has 11 entries in Wikipedia. Given the text (from Wikipedia):

One of his translators, Roy Harris, summarized Saussure’s contribution to linguistics and the study of language in the following way...

Which Saussure?

- Ferdinand de Saussure:
  - Wikidata: http://www.wikidata.org/wiki/Q13230
  - DBpedia:
    http://dbpedia.org/resource/Ferdinand_de_Saussure

- Henri de Saussure: http://www.wikidata.org/wiki/Q123776

- René de Saussure: http://www.wikidata.org/wiki/Q13237
Disambiguation of Named Entities

Given:

*One of his translators, Roy Harris, summarized Saussure’s contribution to linguistics and the study of language...*

Disambiguation is a classification problem dealing with mention-entity pairs:

<table>
<thead>
<tr>
<th>Mention</th>
<th>Entity</th>
<th>Q number</th>
<th>T/F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Saussure</td>
<td>Ferdinand de Saussure</td>
<td>Q13230</td>
<td>1</td>
</tr>
<tr>
<td>Saussure</td>
<td>Henri de Saussure</td>
<td>Q123776</td>
<td>0</td>
</tr>
<tr>
<td>Saussure</td>
<td>René de Saussure</td>
<td>Q13237</td>
<td>0</td>
</tr>
</tbody>
</table>

Feature vectors represent pair of mentions and entities:

- Cosine similarity between the mention context and the named entity page in Wikipedia and bag-of-word vectors of the mention context.
- Training set built from Wikipedia markup:
  ```
  [[Ferdinand_de_Saussure|Saussure]]
  ```
Graph databases are popular devices used to represent named entities, especially the resource description framework (RDF). Entities are assigned unique resource identifiers (URIs) similar to URLs (as in HTTP addresses) and can be linked to other data sources (Linked data).

Examples of databases using the RDF format:

**DBpedia**: A database of persons, organizations, locations, etc. DBpedia is automatically extracted from Wikipedia semi-structured data (info boxes).

**Geonames**: A database of geographical names (a gazetteer).

SPARQL is a database query language that enables a programmer to extract data from a graph database (similar to Prolog or SQL).
Garcia Alvarado, 56, was killed when a bomb placed by urban guerrillas on his vehicle exploded as it came to a halt at an intersection in downtown San Salvador.
Anaphora, often pronouns

_Pronouns: it, she, he, this, that_

Cataphora

_I just wanted to touch it, this stupid animal._

_They have stolen my bicycle._

Antecedents

Ellipsis is the absence of certain referents

_I want to have information on caterpillars. And also on hedgehogs._
The MU Conferences have defined a standard annotation for noun phrases. It uses the \texttt{COREF} element with five possible attributes: \texttt{ID}, \texttt{REF}, \texttt{TYPE}, \texttt{MIN}, and \texttt{STAT}.

- \texttt{<COREF ID="100">Lawson Mardon Group Ltd.</COREF>} said
  \texttt{<COREF ID="101" TYPE="IDENT" REF="100">it</COREF>}

- \texttt{<COREF ID="100" MIN="Haden MacLellan PLC">Haden MacLellan PLC of Surrey, England</COREF> ...}
  \texttt{<COREF ID="101" TYPE="IDENT" REF="100">Haden MacLellan</COREF>}

\section*{Coreference Annotation}
Vandenberg and Rayburn are heroes of mine. Mr. Boren says, referring well to Sam Rayburn, the Democratic House speaker who cooperated with President Eisenhower.
In the MUC competitions, coreference is defined as symmetric and transitive:

- If A is coreferential with B, the reverse is also true.
- If A is coreferential with B, and B is coreferential with C, then A is coreferential with C.

It forms an equivalence class called a coreference chain. The TYPE attribute specifies the link between the anaphor and its antecedent. IDENT is the only possible value of the attribute. Other types are possible such as part, subset, etc.
Coreferences define a class of equivalent references
Backward search with a compatible gender and number
98% of the antecedents are in the current or previous sentence
Focus: an integer attached to all objects, incremented when:
- It is mentioned: subject, object, adjunct
- It is visible or pointed at.
The focus is decremented over time
Constraints are also applied: subject $\neq$ object, grammatical role
Anaphora is resolved by taking the highest focus
A Simplistic Method

Garcia Alvarado, 56, was killed when a bomb placed by urban guerrillas on his vehicle exploded as it came to a halt at an intersection in downtown San Salvador.
Instead of manually engineered rules, machine learning uses an annotated corpus and trains the rules automatically. The coreference solver is a decision tree. It considers pairs of noun phrases \((NP_i, NP_j)\). Each pair is represented by a feature vector of 12 parameters. The tree takes the set of NP pairs as input and decides for each pair whether it corefers or not. Using the transitivity property, it identifies all the coreference chains in the text. The ID3 learning algorithm automatically induces the decision tree from texts annotated with the MUC annotation standard.
The coreference engine takes a pair of extracted noun phrases \((NP_i, NP_j)\). For a given index \(j\), the engine considers from right to left, \(NP_i\) as a potential antecedent and \(NP_j\) as an anaphor. It classifies the pair as positive if both NPs corefer or negative if they don’t.
Some Features

- **Positional feature:**
  1. Distance (DIST): This feature is the distance between the two noun phrases measured in sentences: 0, 1, 2, 3, ... The distance is 0 when the noun phrases are in the same sentence.

- **Grammatical features:**
  2. $i$-Pronoun (I_PRONOUN): Is $NP_i$ a pronoun i.e. personal, reflexive, or possessive pronoun? Possible values are true or false.
  3. $j$-Pronoun (J_PRONOUN): Is $NP_j$ a pronoun? Possible values are true or false.

- **Lexical feature:**
  12. String match (STR_MATCH): Are $NP_i$ and $NP_j$ equal after removing articles and demonstratives from both noun phrases? Possible values are true or false.
Training Examples: The Positive Examples

The classifier can be a decision tree or logistic regression. It is trained from positive and negative examples extracted from the annotated corpus. The positive examples use pairs of adjacent coreferring noun phrases. If \( NP_{a1} - NP_{a2} - NP_{a3} - NP_{a4} \) is a coreference chain in a text, we have

<table>
<thead>
<tr>
<th>Noun phrases</th>
<th>Coreference chains</th>
</tr>
</thead>
<tbody>
<tr>
<td>( NP_{a1} )</td>
<td>Chain 22</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>( NP_{a2} )</td>
<td>Chain 22</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>( NP_{a3} )</td>
<td>Chain 22</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
<tr>
<td>( NP_{a4} )</td>
<td>Chain 22</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

The positive examples correspond to the pairs: \((NP_{a1}, NP_{a2}), (NP_{a2}, NP_{a3}), (NP_{a3}, NP_{a4})\)
Training Examples: The Negative Examples

The negative examples consider the noun phrases $NP_{i+1}, NP_{i+2}, \ldots, NP_{j-1}$ intervening between adjacent pairs $(NP_i, NP_j)$.

<table>
<thead>
<tr>
<th>Noun phrases</th>
<th>Coreference chains</th>
<th>Relation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$NP_i$</td>
<td>Chain 22</td>
<td>Antecedent</td>
</tr>
<tr>
<td>$NP_{i+1}$</td>
<td>Not part of Chain 22</td>
<td></td>
</tr>
<tr>
<td>$NP_{i+2}$</td>
<td>Not part of Chain 22</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$NP_{j-1}$</td>
<td>Not part of Chain 22</td>
<td></td>
</tr>
<tr>
<td>$NP_j$</td>
<td>Chain 22</td>
<td>Anaphor</td>
</tr>
</tbody>
</table>

For each positive pair $(NP_i, NP_j)$, the training procedure generates negative pairs:

- They consist of one intervening $NP$ and the anaphor $NP_j$: $(NP_{i+1}, NP_j)$, $(NP_{i+2}, NP_j)$, \ldots, and $(NP_{j-1}, NP_j)$.
- The intervening noun phrases can either be part of another coreference chain or not.
At this point, it is useful to have the current performances in mind:

- Morphological parsing can parse correctly 99% of the words in many languages (Koskenniemi 1984).

  *Bilolyckorna* "bil#olycka" N UTR DEF PL NOM

- Part-of-tagging reaches and exceeds 97% (Church 1991).

  *En bilolycka med tre bilar*

  En/dt_utr_sin_ind bilolycka/nn_utr_sin_ind_nom med/pp tre/rg_nom bilar/nn_utr_plu_ind_nom

- Sentence parsing reaches 85% in Swedish (Nivre 2006) – labeled dependencies.
Conversion of a sentence into a predicate–argument structure. The F-measure reaches about 80 (CONLL 2009).

\[ \text{Judge She} \text{blames [Evaluator the Government] [Reason for failing to do enough to help]} \]

blames(judge, evaaluee, reason)
blames('She', 'The Government', 'for failing to do enough to help').

Coreference solving reaches a MUC F-measure of \( \sim 60 \). Latest figures from CoNLL, Pradhan et al. (2011)
Discourse theories are used to develop organization models of texts. They have three objectives: represent, parse automatically, and generate a discourse. There are many ways to represent a text and competing theories. In 1992, Mann and Thompson compared 12 different representations obtained from experts in the field. The most significant are:

- Grosz and Sidner’s theory (1986) and Centering (1995)
- Rhetorical structure theory (RST) (Mann and Thompson 1988)
Discourse is described as a hierarchical tree.
Centers are entities that link one a sentence to another one. Grosz divides centers in a unique backward-looking center that is the most important entity in the segment and others forward-looking centers. Two relations link segments: dominance and satisfaction-precedence.
• Invention (*Inventio*).
• Arrangement (*Dispositio*): introduction (*exordium*), a narrative (*narratio*), a proposition (*propositio*), a refutation (*refutatio*), a confirmation (*confirmatio*), and finally a conclusion (*peroratio*).
• Style (*Elocutio*): emote (*movere*), explain (*docere*), or please (*delectare*).
• Memory (*Memoria*)
• Delivery (*Actio*).
The rhetorical structure theory is a text grammar that analyzes argumentation: A text consists of:

- **Text spans** that can be sentences or clauses
- **Rhetorical relations** that link the text spans

Relations are richer than with Grosz and Sidner.
Relations between segments can be symmetrical when spans have the same importance: Both spans are **nuclei**.

When relations are asymmetrical, we have a **nucleus** and a **satellite** where the nucleus is the most important.

The text analysis produces a tree of text spans that are linked by different relation types.
Example cited by Mann and Thompson (1987):

1. **Concern that this material is harmful to health or the environment may be misplaced.**

2. **Although it is toxic to certain animals,**

3. **evidence is lacking that it has any serious long-term effect on human beings.**
Spans can have a same importance and are linked by a sequence relation:

1. Napoleon met defeat in 1814 by a coalition of major powers, notably Prussia, Russia, Great Britain, and Austria.
2. Napoleon was then deposed
3. and exiled to the island of Elba
4. and Louis XVIII was made ruler of France.

Microsoft Encarta, cited from Simon Corston-Oliver (1998)
Mann and Thompson gave a formal structure to the graph that correspond to a parse tree:

1. The tree extends over the whole text;
2. Each text span part of the text analysis is either a terminal symbol or a node constituent;
3. A span has a unique parent;
4. Relations bind adjacent spans.
The original relations in RST are:

<table>
<thead>
<tr>
<th>Nucleus-satellite relations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circumstance</td>
</tr>
<tr>
<td>Solutionhood</td>
</tr>
<tr>
<td>Elaboration</td>
</tr>
<tr>
<td>Background</td>
</tr>
<tr>
<td>Enablement</td>
</tr>
<tr>
<td>Motivation</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Multi-nucleus relations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sequence</td>
</tr>
</tbody>
</table>
The number of relations is somewhat arbitrary. Mann and Thompson first proposed 15 relations, then 23. It is possible to group and simplify them.

Symmetrical (nucleus-nucleus) and asymmetrical relations (nucleus-satellite)

Group classes in a superclass
Definition of the Relations

The following text corresponds to an evidence relation that links a nucleus (segment 1) and a satellite (segment 2):

1. The program as published for calendar year 1980 really works.
2. In only a few minutes, I entered all the figures from my 1980 tax return and got a result which agreed with my hand calculations to the penny.

Mann and Thompson defined each relation in the RST model using a set of “constraints”.
<table>
<thead>
<tr>
<th>Relation name</th>
<th>EVIDENCE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constraints on the nucleus $N$</td>
<td>The reader R might not believe to a degree satisfactory to the writer W</td>
</tr>
<tr>
<td>Constraints on the satellite $S$</td>
<td>The reader believes $S$ or will find it credible</td>
</tr>
<tr>
<td>Constraints on the $N + S$ combination</td>
<td>R’s comprehending $S$ increases R’s belief of $N$</td>
</tr>
<tr>
<td>The effect</td>
<td>R’s belief of $N$ is increased</td>
</tr>
<tr>
<td>Locus of the effect</td>
<td>$N$</td>
</tr>
</tbody>
</table>
Is it possible to process automatically texts with these definitions? And how can we do?
The description of an evidence relation is:

*Reader believes Satellite or finds it credible*

How can we measure this?
The idea is to map a certain relation to certain words. Words like *and*, *so*, *but*, *although*, and commas denote frontiers and ideas in a text. The automatic text analysis uses these signs, *cues*, *cue phrases*, to segment a text and recognize relations.
Cues are often ambiguous. Example:

\[Karl \textbf{and} Jan \textit{came to the lecture}] \textbf{and} \textit{asked questions}\]

The first \textbf{and} has a syntactic role only. The second one defines a sequence.
We must use supplementary constraints like position constraints between spans to carry out the analysis.
Mann and Thompson describe a typical ordering between relations:

- **Satellite before nucleus**
  - Antithesis
  - Background
  - Concession
  - Condition
  - Justify
  - Solutionhood

- **Nucleus before satellite**
  - Elaboration
  - Enablement
  - Evidence
  - Statement
Corston-Oliver’s Method

Corston-Oliver (1998) used such position constraints and cues as a strategy to analyze texts. He recognizes an elaboration relation between two clauses where clause 1 is the nucleus and clause 2 the satellite using these constraints:

1. Clause 1 precedes Clause 2
2. Clause 1 is not subordinate to Clause 2
3. Clause 2 is not subordinate to Clause 1

and some cues that he ranks using heuristics
For elaboration, there are six heuristics. Two of them (simplified):

1. Clause 1 is the main clause of a sentence (sentence $k$) and Clause 2 is the main clause of a second one (sentence $l$). Sentence $k$ immediately precedes sentence $l$ and Clause 2 contains an elaboration conjunction (also, for example). (Heuristic 24, score 35)

2. Clause 2 contains a predicate nominal whose head is in the set \{portion, component, member, type, kind, example, instance\} or Clause 2 contains a predicate whose the main verb is in the set \{include, consist\} (Heuristic 41, score 35)
Corston-Oliver applied this method to analyze the article stem in the Microsoft Encarta encyclopedia:

1. *A stem is a portion of a plant.*

2. *Subterranean stems include the rhizomes of the iris and the runners of the strawberry;*

3. *The potato is a portion of an underground stem.*
With heuristic 41 and because of words **include** and **portion**, he could find the following rhetorical structure:
We saw that and can have a syntactic role and also a discourse role. A discourse relation, here contrast, can use two or more cues:

- The driver died **but** the passenger survived
- The driver died **and** the passenger survived

There can also be no cue to mark the relation

- The driver died. The passenger survived
### Contrast

<table>
<thead>
<tr>
<th></th>
<th>Contrast</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>Such standards would preclude arms sales to states like Libya, which is also currently subject to a U.N. embargo.</td>
<td>South Africa can afford to forgo sales of guns and grenades.</td>
</tr>
<tr>
<td>b</td>
<td><strong>But</strong> states like Rwanda before its present crisis would still be able to legally buy arms.</td>
<td><strong>because</strong> it actually makes most of its profits from the sale of expensive, high-technology systems like laser-designated missiles, aircraft electronic warfare systems, tactical radios, anti-radiation bombs and battlefield mobility systems.</td>
</tr>
</tbody>
</table>

![Contrast Diagram](image)  
![Explanation Diagram](image)
Marcu and Echihabi (2002) developed an unsupervised learning algorithm to identify rhetorical relations. The idea is to use words like but as a strong sign of contrast and to find automatically other contrast conditions using a corpus of one billion words.

<table>
<thead>
<tr>
<th>Contrast</th>
<th>Cause-evidence-explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>[BOS...EOS] [But...EOS]</td>
<td>[BOS... ] [because...EOS]</td>
</tr>
<tr>
<td>[BOS...] [but ...EOS]</td>
<td>[BOS Because... , ] [... EOS]</td>
</tr>
<tr>
<td>[BOS... ] [although...EOS]</td>
<td>[BOS...EOS] [BOS Thus...EOS]</td>
</tr>
<tr>
<td>[BOS Although... , ] [...EOS]</td>
<td></td>
</tr>
<tr>
<td>Condition</td>
<td>Elaboration</td>
</tr>
<tr>
<td>[BOS If... , ] [...EOS]</td>
<td>[BOS...EOS] [BOS ...for example...EOS]</td>
</tr>
<tr>
<td>[BOS If... , ] [then ...EOS]</td>
<td>[BOS...] [which ...EOS]</td>
</tr>
<tr>
<td>[BOS ...] [if ...EOS]</td>
<td></td>
</tr>
</tbody>
</table>
Determination of Discourse Relations

The goal of the analysis is to find systematically word pairs in relations. First, build the Cartesian product of words \((o_m \times o_n) \in S_p \times S_q\) where \(S_p\) and \(S_q\) are two text segments. Then, determine the discourse relation between two segments, \(S_1\) and \(S_2\) using the formula

\[
\hat{r} = \arg \max_k P(r_k | S_1, S_2)
\]

To simplify computation, use only nouns, verbs, and cue phrases.
Naïve Bayes

Bayes formula on conditional probabilities:

\[ P(A|B)P(B) = P(B|A)P(A) \]

For the rhetorical relations, we compute

\[ \hat{r} = \arg\max_k P(r_k)P(S_1, S_2|r_k) \]

The naïve application of Bayes’ principle yields:

\[ \hat{r} = \arg\max_k (P(r_k) \times \prod_{(o_m, o_n) \in S_1, S_2} P(r_k)P(S_1, S_2|r_k)) \]
Here pairs are unambiguous, but counts could be (sales, electronics): 19 contrasts, 23 explanations.
We estimate $P(o_1, o_2 | r_k)$ using the maximum likelihood estimate. The estimation is done with automatically extracted word pairs that belong to a relation.

Even with a corpus of one billion words, there are unseen pairs. Marcu and Echihabi used the Laplace rule to handle them.
When the program is compared with a manually annotated RST corpus, we have the results for two-way classifiers

<table>
<thead>
<tr>
<th></th>
<th>Contrast</th>
<th>CEV</th>
<th>Cond</th>
<th>Elab</th>
</tr>
</thead>
<tbody>
<tr>
<td>#</td>
<td>238</td>
<td>307</td>
<td>125</td>
<td>1761</td>
</tr>
<tr>
<td>Contrast</td>
<td>–</td>
<td>63%</td>
<td>80%</td>
<td>64%</td>
</tr>
<tr>
<td>CEV</td>
<td></td>
<td>87%</td>
<td></td>
<td>76%</td>
</tr>
<tr>
<td>Cond</td>
<td></td>
<td></td>
<td></td>
<td>87%</td>
</tr>
</tbody>
</table>

The classifier decides correctly in 63 % of the cases for the relations contrast and cause-evidence-explanation (CEV). Only 26 % of the contrast relations are marked with an unambiguous cue like *but*. The rest is discovered using probabilities.
Parsing Algorithm: An Overview

Parsing uses a bottom-up search strategy:

1. Identify segments
2. Generate all possible relations between segments
3. Order relations in increasing order using heuristics
4. For all segment pairs in increasing order, try to:
   1. Merge the highest pair that contains adjacent segments
   2. Replace the pair with the nucleus
5. Until all the segments are merged into the whole text
Results for whole texts are still preliminary
But we have seen that there are promising signs for a correct analysis
Improvements depend on models, formalisms, and use of gigantic corpora
Such text analysis should enable to turn computerized encyclopedia into knowledge bases and ask questions like:

- What are the causes of something?
- Are there contradictions in the text
Research on the representation of time, events, and temporal relations dates back the beginning of logic. It resulted in an impressive number of formulations and models. A possible approach is to reify events: turn them into objects, quantify them existentially, and connect them using predicates

\[ \exists \varepsilon [ \text{saw}(\varepsilon, \text{John}, \text{Mary}) \land \text{place}(\varepsilon, \text{London}) \land \text{time}(\varepsilon, \text{Tuesday}) ], \]

where \( \varepsilon \) represents the event.

*John saw Mary in London on Tuesday*
Events are closely related to sentence’s main verbs. Different classifications have been proposed to associate a verb with a type of event, Vendler (1967):

- A state – a permanent property or a usual situation (e.g. be, have, know, think);
- An achievement – a state change, a transition, occurring at single moment (e.g. find, realize, learn);
- An activity – a continuous process taking place over a period of time (e.g. work, read, sleep). In English, activities often use the present perfect -ing;
- An accomplishment – an activity with a definite endpoint completed by a result (e.g. write a book, eat an apple).
<table>
<thead>
<tr>
<th>#</th>
<th>Relations</th>
<th>#</th>
<th>Inverse relations</th>
<th>Graphical representations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>before(a, b)</td>
<td>2</td>
<td>after(b, a)</td>
<td><img src="#" alt="Graph" /></td>
</tr>
<tr>
<td>3</td>
<td>meets(a, b)</td>
<td>4</td>
<td>met_by(b, a)</td>
<td><img src="#" alt="Graph" /></td>
</tr>
<tr>
<td>5</td>
<td>overlaps(a, b)</td>
<td>6</td>
<td>overlapped_by(b, a)</td>
<td><img src="#" alt="Graph" /></td>
</tr>
<tr>
<td>7</td>
<td>starts(a, b)</td>
<td>8</td>
<td>started_by(b, a)</td>
<td><img src="#" alt="Graph" /></td>
</tr>
<tr>
<td>9</td>
<td>during(b, a)</td>
<td>10</td>
<td>contains(a, b)</td>
<td><img src="#" alt="Graph" /></td>
</tr>
<tr>
<td>11</td>
<td>finishes(b, a)</td>
<td>12</td>
<td>finished_by(a, b)</td>
<td><img src="#" alt="Graph" /></td>
</tr>
<tr>
<td>13</td>
<td>equals(a, b)</td>
<td></td>
<td></td>
<td><img src="#" alt="Graph" /></td>
</tr>
</tbody>
</table>
TimeML is an effort to unify temporal annotation, based on Allen’s (1984) relations and inspired by Vendler’s (1967) classification. TimeML defines the XML elements:

- TIMEX3 to annotate time expressions (at four o’clock),
- EVENT, to annotate the events (he slept),
- “signals”.

The SIGNAL tag marks words or phrases indicating a temporal relation.
TimeML connects entities using different types of links. Temporal links, TLINKs, describe the temporal relation holding between events or between an event and a time. TimeML elements have attributes. For instance, events have a tense, an aspect, and a class. The 7 possible classes denote the type of event, whether it is a STATE, an instantaneous event (OCCURRENCE), etc.
All 75 people on board the Aeroflot Airbus died when it ploughed into a Siberian mountain in March 1994 (Ingria and Pustejovsky 2004):

All 75 people
<EVENT eid="e7" class="STATE">on board</EVENT>
<MAKEINSTANCE eiid="ei7" eventID="e7" tense="NONE" aspect="NONE"/>
<TLINK eventInstanceID="ei7" relatedToEvent="ei5" relType="INCLUDES"/>
the Aeroflot Airbus
<EVENT eid="e5" class="OCCURRENCE">died</EVENT>
<MAKEINSTANCE eiid="ei5" eventID="e5" tense="PAST" aspect="NONE"/>
<TLINK eventInstanceID="ei5" signalID="s2" relatedToEvent="ei6" relType="IAFTER"/>
All 75 people on board the Aeroflot Airbus died when it ploughed into a Siberian mountain in March 1994 (Ingria and Pustejovsky 2004):

<SIGNAL sid="s2">when</SIGNAL>
<EVENT eid="e6" class="OCCURRENCE">ploughed</EVENT>
<MAKEINSTANCE eiid="ei6" eventID="e6" tense="PAST" aspect="NONE"/>
<TLINK eventInstanceID="ei6" signalID="s3" relatedToTime="t2" relType="IS_INCLUDED"/>
<TLINK eventInstanceID="ei6" relatedToEvent="ei4" relType="IDENTITY"/>
<into a Siberian mountain
<SIGNAL sid="s3">in</SIGNAL>
<TIMEX3 tid="t2" type="DATE" value="1994-04">March 1994</TIMEX3>.