

PART 2:

DOCUMENTATION OF THE SYNTACTIC LAYER

| | |
|--|-----------|
| INTRODUCTION | 5 |
| 1 THE SYNTACTIC LAYER | 5 |
| 1.1 THE CONTENTS OF THE LEXICON | 5 |
| 1.2 LINGUISTIC DESCRIPTION AT THE SYNTACTIC LAYER | 6 |
| 1.2.1 <i>Basic principles of valency-boundness</i> | 6 |
| 1.2.2 <i>The description of valency</i> | 7 |
| 1.2.3 <i>Treatment of the control phenomena</i> | 8 |
| 2 THE SYNTACTIC DESCRIPTION OF WORD CLASSES..... | 8 |
| 2.1 VERBS..... | 8 |
| 2.1.1 <i>Complements vs. adjuncts/free modifiers</i> | 8 |
| 2.1.2 <i>Syntactic functions of complements</i> | 8 |
| 2.1.3 <i>The arity of verbs and the numbering of the positions in constructions</i> | 9 |
| 2.1.4 <i>The arity of reflexive verbs and phrasal verbs</i> | 10 |
| 2.1.5 <i>Valency pattern types of verbs - An overview</i> | 10 |
| 2.1.6 <i>Phrasal verbs - Treatment of particles</i> | 12 |
| 2.1.7 <i>Treatment of prepositions</i> | 12 |
| 2.1.8 <i>Syntactic units and verb alternations</i> | 12 |
| 2.2 NOUNS..... | 13 |
| 2.2.1 <i>The valency of nouns</i> | 13 |
| 2.2.2 <i>Optionality of complements and encoding strategy</i> | 15 |
| 2.2.3 <i>Syntactic functions of complements</i> | 15 |
| 2.2.4 <i>The Self element for nouns</i> | 16 |
| 2.2.5 <i>Valency frames of nouns – An overview</i> | 16 |
| 2.2.6 <i>Noun as a complement of mass entity nouns</i> | 17 |
| 2.2.7 <i>Prepositional phrases</i> | 17 |
| 2.2.8 <i>Clausal complements</i> | 17 |
| 2.2.9 <i>Syntactic units and noun alternations</i> | 18 |
| 2.3 ADJECTIVES | 19 |
| 2.3.1 <i>The syntactic encoding of adjectives</i> | 19 |
| 2.3.2 <i>The valency of adjectives</i> | 20 |
| 2.3.3 <i>The arity of adjectives and numbering of the positions in constructions</i> . | 20 |
| 2.3.4 <i>Predicative constructions with clausal complements</i> | 20 |
| 2.3.5 <i>Optionality of complements</i> | 21 |
| 2.3.6 <i>Optionality in syntactic units and descriptions</i> | 21 |
| 2.3.7 <i>Syntactic functions of complements</i> | 22 |
| 2.3.8 <i>Valency patters of adjectives</i> | 23 |
| 2.3.9 <i>Syntactic units and alternations</i> | 26 |
| 3 THE DATA: STO SYNTAX REPRESENTED AS XML ELEMENTS | 27 |
| 3.1 THE STRUCTURE OF THE STO SYNTAX XML FILES | 27 |
| 3.1.1 <i>Mu_Synu, Mu_Id and Spelling elements</i> | 27 |
| 3.1.2 <i>Synu, Description and Construction elements</i> | 28 |
| 3.1.3 <i>Self elements</i> | 31 |

| | | |
|----------|---|-----------|
| 3.1.4 | <i>Self for verbs</i> | 31 |
| 3.1.5 | <i>Self for nouns</i> | 31 |
| 3.1.6 | <i>Self for adjectives</i> | 31 |
| 3.1.7 | <i>Self for adverbs</i> | 31 |
| 3.1.8 | <i>The elements describing valency patterns</i> | 33 |
| 3.2 | THE NUMBER OF MAIN XML ELEMENTS..... | 35 |
| 3.3 | AN EXAMPLE OF XML..... | 36 |
| 4 | APPENDIX | 38 |
| 4.1 | THE XML SCHEMA FILE FOR THE STO SYNTAX, STO_SYNTAX.XSD: | 38 |

Introduction

The acronym **STO** is the abbreviation of the Danish title of the present lexicon, and it stands for ‘SprogTeknologisk Ordbase’, Lexicon for Language Technology Applications. This acronym is used in the file names of the delivered data material.

The data material is provided as three XML files as follows (size in bytes):

| | |
|---------------------|---------|
| STO_Syntax_1_v1.xml | 4437723 |
| STO_Syntax_2_v1.xml | 4872856 |
| STO_Syntax_3_v1.xml | 4488728 |

The data files can be validated with the XML Schema which can be found in Appendix 1. (File name: STO_Syntax.xsd, size 21865 bytes).

The present documentation is the second part of the complete Documentation of the STO lexicon, the first part provides a general description of the STO lexicon and describes the Morphological Layer. Part 2 of a Documentation on the STO Syntactic Layer is structured in the following way:

Firstly, some general decisions concerning the syntactic layer are described briefly (Chapter 1), secondly, the linguistic principles adopted for the encoding of syntactic features are presented for each word class in detail (Chapter 2), and finally we describe the format of the XML-files (Chapter 3).

The Documentation contains also a literature list, which may be useful for the user not familiar with the linguistic theories mentioned in the document or with the Danish language. For details of the Danish grammar please consult Allen et al. *A Comprehensive Grammar of Danish* (1995).

1 The syntactic layer

1.1 *The contents of the lexicon*

The vocabulary provided with a syntactic description is a proper subset (app. 45,000 words) of the whole vocabulary covered in STO (app. 81,300 words). Thereof approximately 34,000 words are selected from the general language corpora of STO on the basis of their frequency. The rest, approx. 11,000 words belong to the domain languages covered. All syntactic information contained in STO is encoded on the basis of corpus evidence, where preference is given to more frequent syntactic patterns in cases, where an exhaustive encoding (including single or sparse occurrences of a single word) was not possible because of the overwhelming number of syntactic constructions. As an example can be mentioned a group of very common verbs (e.g. *komme* (to come), *tage* (to take)) having different syntactic constructions in such numbers that the list including also their infrequent constructions would be too comprehensive.

The methodology adopted in the description of syntactic features of words is mainly based on the valency theory. Main attention is paid to various grammatical structures of words consisting of a dominant word (noun, adjective or verb) and its complements. Thus, the syntactic layer of STO comprises the description of valency patterns for nouns, verbs and adjectives. Adverbs are not provided with syntactic descriptions in this data delivery (Version 1), although the appropriate structure of XML elements is implemented (cf. Chapter 3 The Data: STO syntax represented as XML elements).

1.2 Linguistic description at the syntactic layer

1.2.1 Basic principles of valency-boundness

The central features in the linguistic description at the syntactic layer concern the valency of words, and the syntactic behaviour of words are described in terms of valency patterns.

The approach to valency adopted in STO is highly inspired by the model of six distinct degrees of valency-binding, developed in Somers (1987). This model is based on the observation that there are different degrees by which constituents depend on the head. Most dependent are the lexically determined integral complements which are typically parts of the predicate e.g. in collocations; thereafter come the obligatory and optional complements. ‘Middles’ (ibid., p. 27) are on the borderline between complements and modifiers where adjuncts or modifiers are the less dependent.

The encoding strategy for complements and middles vs. adjuncts is based on the above sketched model. Since collocations (and other types of multi-word units) do not form part of STO yet, integral complements are not encoded. All **obligatory** and **optional complements** are encoded while adjuncts are not seen as part of the valency pattern, unless they are frequent.

In general ‘**middles**’ are encoded in cases where they are noticeably frequent in the corpora e.g. *en markedsandel på 20%* (a market share of 20%) and if they are pertinent to the central meaning of the word (e.g. *et mindre afbræk i produktionen* (lit.: a minor break in the production). The treatment of ‘middles’ varies slightly, dependent on the word class in question. The strategy adopted for the individual word classes is described in the relevant sections.

A few examples below illustrate some prototypical syntactic constructions that are regarded as valency-bound and accordingly, they are described in STO. (The optional complements are in brackets.)

- Verbal constructions
læse (bøger) (read (books)): divalent (transitive)
sende et brev (fra København) (til Århus) (send a letter (from Copenhagen) (to Århus)): trivalent
- Nominal constructions

en kasse æbler (a box of apples): monovalent

en søster til Peter (a sister of Peter)

- Adjectival construction

afhængig af indtægter (dependent on income): divalent

Contrary to the above examples, phrases of the type *spise* (på restaurant/ om morgenen/ tre gange om dagen) (eat (at a restaurant/ in the morning/ three times a day)) are not encoded, because the **locative** and **temporal adjuncts** in connection with the verb *spise* (eat) are not pertinent to the meaning of the verb but they are completely arbitrary (free modifiers).

1.2.2 The description of valency

The word is considered to be the syntactic head of the described construction,

- its valency is given in a **valency pattern** by listing and describing the constituents the word subcategorizes for and their characteristics and
- the characteristics of the **head itself** in the current context is given.

A valency pattern is described in a syntactic **description** and encoded in a **syntactic unit**. Thus different syntactic descriptions of the same word are encoded in different syntactic units. Exceptions are valency patterns which are considered to be syntactic alternations. These alternating syntactic descriptions are encoded in the same syntactic unit. Examples of encoded syntactic alternations are the following

- Verbs: dative shift alternation, e.g.

Jeg giver hende en bog/jeg giver en bog til hende

(I give her a book/I give a book to her)

- Adjectives: infinitive-clause without/with expletive, e.g.

Det er attraktivt at bo i København/At bo i København er attraktivt

(It is attractive to live in Copenhagen/To live in Copenhagen is attractive)

In other words, a valency pattern of a word therefore contains information about

- the arity of that word, viz. how many complements it governs (between 0 and 4)
- the syntactic function of each governed complement (e.g. subject, object)
- the syntactic construction potential of each complement expressed in terms of syntactic categories (e.g. noun phrase, prepositional phrase, clause).

In STO further linguistic details are specified, such as

- the preposition of each governed prepositional phrase and the construction potential of the governed element
- the control type in case of infinite clauses (viz. subject or object control)
- obligatoriness of each governed complement.

1.2.3 Treatment of the control phenomena

As mentioned above, the STO syntax provides information about control phenomena. For example the STO syntax indicates explicitly **subject control**. In the example below, the unexpressed subject of the infinitive *at ringe* (to ring up) in the sentence is the same as the subject of the matrix clause, i.e. *jeg* (I).

Ex.: *Jeg ønskede at ringe til hende* (I wished to ring her up)

Another control type encoded is the **object control**. In the following example, the unexpressed subject of the infinitive *at ringe* in the sentence is the object of the matrix clause, i.e. *Maria*.

Ex.: *Jeg anbefalede Maria at ringe til hende* (I recommended Maria to ring her up)

2 The syntactic description of word classes

2.1 Verbs

This section describes the syntax of verbs. Firstly, some examples illustrate the distinction between complements and adjuncts/free modifiers. Secondly, a list of syntactic functions of the complements is provided and commented. Thirdly, the arity and valency patterns of verbs are described. Fourthly, the treatment phrasal verbs and reflexive verbs are sketched. Finally, the verbal alternations are listed.

2.1.1 Complements vs. adjuncts/free modifiers

According to the general principle (cf. 1.2 Basic principles and description elements), all obligatory and optional complements are encoded, whereas adjuncts are not seen as part of the valency pattern. The strategy adopted for the treatment of middles (Somers, op. cit.) is based on the following observation: these elements are semantically determined by the governing verb, while their syntactic function and construction types may vary.

Middles are encoded in the valency pattern in the following cases:

- If they tend to be obligatory
e.g. *han bor i Spanien/på landet* (he lives in Spain/ in the countryside)
- If they express direction with motion verbs
e.g. *han gik (ned/op/ud ... til gården)* (he went down/up/out ... to the yard)
- If they are closely related to the core meaning of the verb and occur significantly in the corpus
e.g. *jeg bød ham 1000 DKK for jobbet* (I offered him 1000 DKK for the job).

2.1.2 Syntactic functions of complements

The description comprises the following syntactic functions:

- SUBJECT
- OBJECT
- INDIRECT_OBJECT

- PREPOSITIONAL_OBJECT
- OBJECT_PREDICATE
- FORMAL_SUBJECT
- ADVERBIAL
- SUBJECT_PREDICATE

Some of the above listed functions deserve further explanation.

FORMAL_SUBJECT is a place-holder or an impersonal subject occurring in *der* and *det* constructions in POSITION 1, the real SUBJECT occupies POSITION 2 or POSITION 3 depending on the other constituents of the construction. Exceptions are weather verbs which do not refer to a subject.

Ex.: *Der var tre mænd i haven* (There was three men in the garden)
Det interesserer mig om du kommer (lit: It interests me whether you come)
Det regner (lit: It rains; It's raining.)

INDIRECT_OBJECT is used for a group of ditransitive verbs, and it can occur in POSITION 2 or POSITION 3.

Ex.: *Jeg giver hende bogen* (I give her the book)
Jeg giver bogen til hende (I give the book to her)

PREPOSITIONAL_OBJECT refers to an object introduced by a valency-bound preposition, as in:

Ex.: *Vi tror på Gud* (We believe in God)

ADVERBIAL is encoded if it is governed by the verb

Ex.: *Han behandler hende godt* (He treats her well)

2.1.3 The arity of verbs and the numbering of the positions in constructions

The arity feature registers the number of valency-bound semantic arguments occurring in a given construction. This feature may be relevant for human inspection or in the case of semantic encoding, and it is registered in the identifier of the *Description* and *Construction* elements, e.g. Dv₃... or Cv₂... . The following verb arity types are registered: zero-, mono-, di-, tri- and tetravalent (see the examples below). In the following, the strategy adopted for the counting of arity in verb valency patterns are explained and illustrated. Some elements of particular construction types are not included into the semantic arity number (e.g. the FORMAL_SUBJECT of weather verbs), but they are represented within the syntactic description of the construction by an appropriate POSITION.

FORMAL_SUBJECT is not included into the semantic arity, although represented as a POSITION in the construction

Ex.: *Det regner* (lit: It rains; It's raining). Weather verbs are considered zerovalent.
Der var tre mænd i haven (There were three men in the garden). This is considered to be a monovalent construction.

OBJECT_PREDICATE is not included into the arity because it is regarded as being a part of the OBJECT, although it is represented as a POSITION in the construction. The same applies for accusative constructions with infinitive.

Ex.: *De kaldte ham klog* (lit.: They called him clever)

Jeg hørte ham komme (lit.: I heard him come)

Both examples are considered divalent constructions

SUBJECT_PREDICATE is considered a semantic argument even if it is not realized as such.

Ex.: *Du ser træt ud* (You look tired): this is considered divalent

ADVERBIAL is counted as a complement if it is governed by the verb

Ex.: *Han behandler hende godt* (He treats her well): is considered trivalent.

The valency pattern refers to an individually numbered position for each of the above mentioned elements.

2.1.4 The arity of reflexive verbs and phrasal verbs

A **true reflexive verb** consists of a verb and a reflexive pronoun, which form a unit, thus it is treated as a particular *Self* type (cf. the section 3.1.4 on *Self* for verbs). The reflexive pronoun is usually semantically empty in case of true reflexivity, and it is often not translated (cf. the first example below). Therefore the pronoun is not encoded as an individual position in the valency pattern but in the *Self* and the verb is considered intransitive.

Ex.: *Han brokkede sig* (He complained)

Peter morede sig (Peter enjoyed himself)

In contrast, in other reflexive constructions the reflexive pronoun does occupy the position of the OBJECT and the construction is considered transitive.

Ex.: *Han vaskede sig* (He washed himself) a reflexive NP

Peter vaskede sin søn og sig selv (Peter washed his son and himself):

divalent (transitive), the OBJECT contains two co-ordinated NP's, whereof the second one (*sig*) is a reflexive pronoun.

In case of **phrasal verbs**, the particle is treated in the same way, i.e. particle verbs are treated as a particular *SELF* type. The particle is not encoded as an individual position in the valency pattern but belongs to the *Self* element. (See further section 2.1.6 on Phrasal verbs – The treatment of particles).

Ex.: *Fristen løber ud i morgen* (lit.: The respite runs out tomorrow)

2.1.5 Valency pattern types of verbs - An overview

In the following a number of typical examples are listed in order to illustrate each encoded verb arity type. The complement having the SUBJECT function is not explained in the examples below because the SUBJECT is obligatory and it occupies the

POSITION 1 in constructions with canonical word order.(The valency-bound complements are underlined, optional complements are in brackets, alternating structures are in square brackets with a slash between them.)

2.1.5.1 Zerovalent

Verb constructions with FORMAL_SUBJECT only

Ex.: *Det sner i dag* (It is snowing today)

2.1.5.2 Monovalent verb constructions

Verb constructions with SUBJECT, incl. also FORMAL_SUBJECT

Ex.: *Endelig dukkede han op* (Finally he showed up)

Isen smelter i solen (The ice melts in the sun)

Peter morede sig i går (Peter enjoyed himself yesterday)

Det forlyder, at han ikke må rejse (It is said that he is not allowed to leave.)

Det nytter ikke at klage (It doesn't help to complain)

Det lønner sig at arbejde (It pays to work)

2.1.5.3 Divalent

Verb constructions with SUBJECT incl. also FORMAL_SUBJECT and OBJECT, INDIRECT_OBJECT or PREPOSITIONAL_OBJECT.

Ex.: *Han læser (en bog)* (He reads (a book)) (object, optional)

Han afprøver programmet (He tests the software) (object)

Peter vaskede sig i går (Peter washed himself yesterday) (object)

Han troede hende (He trusted her) (indirect object)

Jeg frydes over sangen (lit.: I'm delighted over the song) (prepositional object)

Der hører brød til maden (lit.: there belongs bread to the food)

Brødet hører til maden (The bread belongs to the food)

De kaldte ham klog (lit.: They called him clever) (object predicate)

2.1.5.4 Trivalent

Verb constructions with SUBJECT incl. also FORMAL_SUBJECT, OBJECT and INDIRECT_OBJECT or PREPOSITIONAL_OBJECT, ADVERBIAL...

Ex.: *Marie gav bogen til drengen* (Marie gave the book to the boy)

Marie gav drengen en bog (Marie gave the boy a book)

Jeg fraråder dig at rejse (I dissuade you from leaving)

Han sladrede (til politiet) (om flugten) (He babbled (to the police) (about the flight))

Hun rejser (fra København) (til London) (She travels (from Copenhagen) (to London))

2.1.5.5 Tetravalent

Verb constructions with SUBJECT incl. also FORMAL_SUBJECT, OBJECT and INDIRECT_OBJECT or PREPOSITIONAL_OBJECT(S), ADVERBIAL...)

Ex.: *Firmaet importerer kaffebønner (fra Afrika) (til de nordiske lande)*

2.1.6 Phrasal verbs - Treatment of particles

Phrasal verbs consist of a verb and a particle which is encoded in the *Self* for the verb in question (cf. the section about Self elements), as mentioned before. A number of verbs, mainly expressing motion, can combine with various local or directional particles. In such cases a general value ‘**dir**’ and/or ‘**loc**’ is encoded.

Ex.: *Han gik ud/op/ned på taget/i gården*
(lit.: He went out/up/down [dir] to the roof/ in the courtyard [loc])

Note: The particular directional particle *hen* is not member of the ‘dir’ particle group because it usually cannot stand alone but combines with an obligatory adverbial. (It is not rendered in English, cf. the example translated below.)

Ex.: *Han gik hen til huset* (He went to the house)

In order to identify phrasal verbs, the method of the so-called ‘loss of stress test’ is employed (cf. Scheuer 1995 and Harder, Heltoft & Thomsen 1996). ‘Loss of stress’ is a term of phonetics designating the phenomenon that phrasal verbs have unity stress, i.e. the verb loses its individual stress. Verb + particle co-occurrences without a loss of stress are not encoded as particle verbs.

Ex.: *Han blev ’væk* ((lit.: He stayed away) (He disappeared))
The ‘ marks the stress, here only the particle is stressed, vs.
Han ’blev ’væk ((lit.: He stayed away) (He didn’t come))
In this latter example, both the verb and the particle are stressed.

Note: A number of verbs allow for a preposition without an NP (viz. a particle) and also for a prepositional construction with an NP. In such cases both constructions are encoded.

Ex.: *Han stod af* (He got off/ he opted out) which is a phrasal verb vs.
Han stod af bussen (He got off the bus).

2.1.7 Treatment of prepositions

In case of locative verbs like *bo* (live) various **locative** prepositions may occur. A general value ‘**loc**’ stands for *i* (in), *på* (on, in), *ved* (at, by), *under* (under), *over* (over), *foran* (before, a head of), *bag* (behind), *bag ved* (behind). An identical encoding principle is adopted in case of motion verbs, where the prepositional phrase is preceded by a directional particle (cf. above).

Ex.: *Vi bor på en ø/i København/ved skoven* (We live on an island/in Copenhagen/ by the forest)

Note: the **directional** prepositions *til* (to), *fra* (from) and *over* (over) are spelled out.

2.1.8 Syntactic units and verb alternations

As has been briefly mentioned in the introductory section on basic principles and description elements, syntactic alternations are encoded in the same syntactic unit. For verbs, the set of alternations encoded as such is limited to the following four (for the sake of clarity, we include the description identifiers in the examples):

- Dative alternations:
Dv3indNN - *jeg giver hende bogen* (I give her the book) and
Dv3NPind - *jeg giver en bog til hende* (I give the book to her)
- 'There' alternations
Dv1 – *en anden verden eksisterer* (another world exists) and
Dv1exderN – *der eksisterer en anden verden* (there exists another world)
- Reciprocal alternations:
Dv1 – *vi samarbejder* (we collaborate) and
Dv2P-med – *jeg samarbejder med dig* (I collaborate with you)
- Reflexive alternations (NB: not for true reflexives):
Dv3xindNN-ind - *jeg bilder ham noget ind* (I made him believe something) and
Dv3xindrefNN-ind – *jeg bilder mig noget ind* (I made me believe something)

2.2 Nouns

This section describes the syntax of nouns, the following main topics are discussed. Firstly, the valency of nouns and the distinction between complements and adjuncts/free modifiers and the adopted encoding strategy is explained. Secondly, a commented list of syntactic functions of the complements is provided. Thirdly, the noun complementation types and particular elements of the valency patterns are explained and exemplified.

2.2.1 The valency of nouns

The general principle (as mentioned in Section 1.2) is adopted as a basis for the syntactic description of nouns. However, it is far more difficult in case of nouns to differentiate between 'middles' and modifiers. The strategy for encoding of 'middles' is based on the fact that computational language processing tasks (e.g. recognition of syntactic structures parsing) benefit from an encoding which is broadened to include this type too. Middles are encoded if they were significantly frequent in the corpus and if they were pertinent to the central meaning of the word. The latter has a particular relevance for the treatment of the domain-related vocabulary.

Ex.: *et mindre afbræk i produktionen* (lit.: a minor break in the production)
en markedsandel på 20% (a market share of 20%) vs.
en bog (på 1 kg) (a book (of 1 kg)) where the quantification does not form a part of the central meaning of the noun, thus it is not encoded.

Furthermore, almost every complement of nouns is optional, thus obligatoriness is not a useful criterion for the identification of valency bound complements. Basically, a noun describes an entity (and not a relation), consequently the vast majority of simple (concrete and abstract) nouns are said not to subcategorise for any complement. On the other hand, nouns derived from verbs or adjectives often take over the valency of the word from which they are derived. (For a further discussion on the valency of nouns please see Grimshaw (1990).)

In the following, a list of valency bearing noun types is given:

Simple, concrete nouns

- Family relationship
Ex.: *Peters søster/søster til Peter* (Peter's sister/the sister of Peter)
- Pictures, semiotic artifacts
Ex.: *billedet af Mona Lisa* (the picture of Mona Lisa)
bogen om landbruget (the book about agriculture)
- Direction nouns
Ex.: *stien fra huset til stranden* (the path from the house to the beach)

Simple, abstract nouns

- Property
Ex.: *husets farve* (the colour of the house)
- Mass
Ex.: *flertallet af indbyggerne* (the majority of the inhabitants)

Deadjectival nouns

These nouns inherit at least one valency-bound complement of the adjective from which they are derived, viz. the external argument of the adjective (cf. Section 2.3.3). In some cases, also the internal argument is inherited.

- External argument
Ex.: *træets højde* (the height of the tree) from *træet er højt* (the tree is high)
- External and internal argument
Ex.: *faderens godhed mod dyr* (lit.: the goodness of the father to animals) from *Faderen er god mod dyr* (lit.: the father is good to animals)

Deverbal nouns

These nouns may inherit none, one or more valency-bound complements of the verb from which they are derived, depending on the type of nominalization (cf. Kirchmeier-Andersen (1997:59-86). In STO the encoding strategy is based on the distinction between 'process (or event) nominals' and 'result nominals' as stipulated in Grimshaw (1990, Chapter 3).

- Process nouns have a dynamic semantic content and thus they allow for realization of one or more valency bound complements
Ex.: *Lægens behandling af patienten for lungebetændelse varede længe*
(Lit.: The doctor's treatment of the patient for pneumonia took a long time)
- Result nouns have a static semantic content and they are aivalent i.e. they do not inherit any complements of the verb
Ex.: *Bygningen findes i byen* (The building is in the city): result reading
- A large number of deverbal nouns has both a process and a result reading as well
Ex.: *Bygningen af hovedkvarteret varede flere år*
(The building of the head quarter took several years): process reading.

Compound nouns

Compound nouns in Danish are written as a single word without blanks between its elements (cf. the Section on morphology). In certain cases, the first element of the compound may be an incorporated complement of the second element (a), but this is by far not always the case. Even though the first element is not an incorporated complement (b), compound nouns trend to have a less rich complementation than simple nouns. The encoding is also in this case based on corpus evidence.

Ex.: a. *program* (til tegning) -> *tegneprogram*
(lit.: program (for drawing) -> drawing-program) vs.
b. *standardprogram* (til tegning) (a standard program (for drawing))

2.2.2 Optionality of complements and encoding strategy

Generally, complements of nouns are optional, there are only a very few nouns which cannot occur without complements. Obligatory complements are thus explicitly marked in the syntactic description.

Ex.: *kvindernes enemærker* (women's preserves)

Because of the general optionality of complements, it is very often impossible to differentiate between the result reading of a noun and its process reading in an elliptic construction (i.e. where the complements are omitted). Thus, in order to avoid unnecessary quasi-duplicates in the database and over-generation in computational language processing, a generalisation strategy of providing 'broad descriptions' is adopted in the following cases:

- Nouns having both process and result reading are provided with one single syntactic description covering both readings

Ex.: (*Kommunens*) *administration* (af klausulerne)
(lit: (the municipality's) *administration* (of the clauses)), process reading
Administrationen har engelsk som arbejdsprog
(The administration has English as its working language), result reading.

- Nouns which subcategorise for various combinations of optional complements are provided with one single description, viz. the broadest possible complementation pattern, even though a construction with all complements realised is not observed within the corpus.

Ex.: a. (*deres*) *flytning* (til England) ((their) moving (to England))
b. *flytning af arbejdspladser fra København*
((moving (of working places) (from Copenhagen))
c. *flytning (af hæren) (fra Århus) (til Krarup)*
((moving (of the army) (from Århus) (to Krarup))

2.2.3 Syntactic functions of complements

- REL_GEN denotes a NP in genitive (relational genitive), from which, in case of deverbal or deadjectival nouns, the subject of the verb or adjective the noun is derived.

Ex.: *Hansens erkendelse af fejlen* (Hansen's acknowledge of the mistake)

- PREPOSITIONAL_OBJECT
Ex.: *Columbus opdagelse af Amerika* (Columbus' discovery of America)
- CLAUSCOMP denotes clausal complements (that- or infinitive clauses)
Ex.: *Den metode at anvende datamater til forskning* (the method of using computers in research)
- SPEC_N denotes a nominal complement of mass entities
Ex.: *en kasse æbler* (a box of apples)

2.2.4 The Self element for nouns

The *Self* for nouns contains one element only, *Definite_Suffix_Allowed*, which can take the values YES or NO. This feature concerns the definiteness of nominal complements (viz. the use of clitic). The value is NOT in case of a nominal complement of a mass entity, in all other cases YES (which is seen as the default value.)

Ex.: *en kasse æbler* (a box of apples)

2.2.5 Valency frames of nouns – An overview

In the following, the various noun complementation types are described and for illustration purposes, a few examples are provided with a detailed syntactic description. Noun patterns may contain three complement types: genitive, noun and prepositional phrase. A list of prototypical examples illustrates the encoding of noun valency in STO.

Genitive or an alternating prepositional phrase

Simple, concrete nouns are considered monovalent nouns

- Family relationship
Ex.: a.) *en søster til Peter* (a sister of Peter) PREPOSITIONAL_OBJECT
(Position: 1; Function: PREPOSITIONAL_OBJECT; Optional: YES;
Syntactic_Label: PP; Introducer: TIL; Clause_Type: NULL; NP_Type: N;
Control: NOCONTROL; Coref: NOCOREF)
b.) *Peters søster* (Peter's sister) REL_GEN
(Position: 1; Function: REL_GEN; Optional: YES; Syntactic_Label: NP; Casus: GENITIVE)

The subgroups below are very similar to the Family relationship subgroup

- Picture nouns
Ex.: *Mona Lisas billede* (Mona Lisa's picture) REL_GEN
billedet af Mona Lisa (the picture of Mona Lisa) PREPOSITIONAL_OBJECT
- Professions
Ex.: *foreningens formand* (lit.: the association's president) REL_GEN
formanden for foreningen (the president of the association)
PREPOSITIONAL_OBJECT

Simple, abstract nouns

- Properties may denote one of the following four property categories (cf. Lenci et al. 2000:228-241): quality, social property, physical property and psychical property.

Ex.: *husets farve* (the colour of the house)

Deverbal nouns

Nouns derived from verbs often subcategorise for a subjective or an objective genitive and may also subcategorise for a prepositional phrase.

Ex.: *Hansens erkendelse af, at der var en fejl*
(lit.: Hansen's acknowledge of that there was a mistake)

In all other cases, the genitive is not encoded as complement, especially because of the fact that it is very difficult to differentiate between relational and possessive genitives.

Ex.: *landets sprog* (lit.: the country's language)

2.2.6 Noun as a complement of mass entity nouns

Mass entity nouns may be of three different subtypes: container, partitive and numeral. It holds for each subtype that the complement is obligatory and it must be indefinite and undetermined, and the complements of container nouns must be countable nouns in plural. Partitive nouns allow only for nouns in singular. Numerals allow only for countable nouns in plural.

Ex.: *en kasse æbler* (a box of apples) : container nouns
et stykke kage (a piece of cake): partitive noun
et tusinde år (one hundred years): numeral

Their syntactic construction is described in the database in the following way:

(Self: definite_suffix_allowed: NO,
Position: 1; Function: SPEC_N; Optional: NO; Syntactic_Label: NP; Casus: UNMARKED; NpIndex: N)

2.2.7 Prepositional phrases

Nouns, like verbs may subcategorise for one or more prepositional phrases (the governed elements may be: NP, that-clause, interrogative clause, wh-clause or infinitive) with the function PREPOSITIONAL_OBJECT. Accordingly, the method of encoding of prepositional complements in noun valency patterns is similar to that of verbs, below a few illustrating examples are provided.

- Mass nouns
Ex.: *flertallet af indbyggerne* (the majority of the inhabitants)
- Semiotic artifacts
bogen om landbruget (the book about agriculture)
- Direction nouns
Ex.: *stien fra huset til stranden* (the path from the house to the beach)

2.2.8 Clausal complements

A number of nouns subcategorises for clausal complements: a that-clause or an infinitive.

Ex.: *Den metode at anvende datamater til forskning*

(lit.: the method to use computers for research): INFINITIVE

Den kendsgerning at han aldrig kommer for sent

(lit.: the fact that he never comes too late): THAT_CLAUSE

In the example provided below, the noun *diskussion* illustrates the four different phrase types governed by the preposition ‘om’ (about):

Ex.: *publikums diskussion med panelet om den nye lov*: NP

(lit.: the audience’s discussion about the new act)

diskussionen om, at slutningen nok var ironisk: THAT_CLAUSE

(lit.: the discussion about that the end might be ironic)

diskussionen om, hvorfor færre syge danskere kan se frem til..: WH-CLAUSE

(lit.: the discussion about why fewer Danes can look forward to...)

præsidentens diskussioner med rådgivere om at bryde ind: INFINITIVE

(lit.: the president’s discussion with advisers about to intervene)

This is reflected by a complex description of these constructions

Position: 1; Function: REL_GEN; Optional: YES; Syntactic_Label: NP; Casus: GENITIVE; Position: 2; Function: PREPOSITIONAL_OBJECT, Optional: YES, Syntactic_Label: PP, Introducer: MED, Clause_Type: NULL; NP_Type: N; Ccontrol: NOCONTROL; Coref: NOCOREF;

Position: 3, Function: PREPOSITIONAL_OBJECT; Optional: YES, Syntactic_label: PP;

Introducer: OM; Clause_Type: NULL; NP_Type: N; Control: NOCONTROL; Coref: NOCOREF/

Introducer: OM; Clause_Type: THAT_CLAUSE; NP_Type: NULL; Control: NOCONTROL; Coref: NOCOREF/

Introducer: OM; Clause_Type: WH-CLAUSE, NP_Type: NULL; Control: NOCONTROL; Coref: NOCOREF/

Introducer: OM; Clause_Type: INFINITIVE; NP_Type: NULL; Control: WITHOUTCONTROL; Coref: WITHOUTCOREF)

2.2.9 Syntactic units and noun alternations

A word that has two valency patterns, is generally encoded in two syntactic units (SynU’s), each with one syntactic description, e.g. in the case of different prepositions.

Ex.: Syntactic unit 1: *variation af temaet* (variation of the theme)

Syntactic unit 2: *variation over temaet* (variation on the theme)

In case of homographs, the different semantics of the same syntactic realisation does not imply two syntactic units. E.g. the word *krampe* is encoded in one syntactic unit though it has two senses ‘staple’ and ‘convulsion’ since the syntactic realisation of the two senses is identical.

The following noun alternation types are encoded in STO (In the examples below, the description identifiers ‘Dn...’ reflecting the alternation are shown for the sake of clarity.)

- Genitive alternation

Two syntactic descriptions can be encoded in one syntactic unit if the two constructions are alternations, i.e. the semantic reading of the constructions is the same though the syntactic expressions differ.

Ex.: *forpagter* (tenant)

Syntactic unit 1

Dn1Pn-af *forpagteren af gården* (the tenant of the farm)

Dn1G *gårdens forpagter* (lit: the farm's tenant)

- Reciprocal constructions

Ex.: *forlovelse* (engagement)

Syntactic unit 1

Dn1G *Peter og Susannes forlovelse* (Peter and Susanne's engagement)

Dn1Pn-mellem *forlovelsen mellem Peter og Susanne*

(the engagement between P. and S.)

Dn2GPn-med *Peters forlovelse med Susanne*

(Peter's engagement with S.)

- Family relations

Ex.: *bror* (brother)

Syntactic unit 1

Dn1G *Jens' bror* (Jens' brother)

Dn1P-til *En bror til Jens* (lit: a brother to Jens)

2.3 Adjectives

This section describes the syntax of adjectives. Firstly, a general introduction to the syntactic encoding of adjectives is given. Secondly, the distinction between complements and adjuncts/free modifiers is briefly discussed. Thirdly, a list of syntactic functions of the complements is provided and commented. Fourthly, the arity of adjectives is discussed and a specification of the particular elements of the valency patterns is given. Finally, the valency patterns of adjectives are illustrated by prototypical examples.

2.3.1 The syntactic encoding of adjectives

Normal adjectives (i.e. all adjectives except quantifiers and numerals) are provided with one or more syntactic descriptions. The syntactic description of an adjective contains syntactic information about the adjective itself and about its valency pattern(s).

Adjectives can be used in a given construction attributively or predicatively. This feature is encoded in the *Self* element. Thus, adjectives that are used both in attributive and predicative function are connected to two *Self* elements.

2.3.2 The valency of adjectives

Obligatory and optional complements are encoded, while free modifiers in accordance with the general strategy adopted are not registered. For identification of valency-bound complements, various linguistic tests were used (e.g. such as topicalisation tests). In such cases where the distinction based on those tests was not clear enough, the element was regarded as a ‘middle’ and its encoding followed the same guidelines as those described in the section on verbs.

Ex.: *de var glade for deres venner* (they were happy with their friends):

complement encoded

de var glade fra morgenstunden (they were happy from early morning):

modifier not encoded.

2.3.3 The arity of adjectives and numbering of the positions in constructions

The syntactic description of adjectives records the syntactic elements the adjective can combine with. However, we also provide the arity information giving the number of valency-bound semantic arguments occurring in a given construction. This feature may be relevant for encoders especially if they want to add semantic information. Arity is registered in the identifier of the *Description* and *Construction* elements, e.g. Da₃... or Ca₃... . The following adjective arity types are registered: mono, di- and trivalent (cf. examples below).

In the following, the **external** and **internal** argument types are exemplified and the strategy adopted for counting of arity in adjective valency patterns is explained and illustrated.

The **external** argument of an adjective is the noun which is modified by the adjective within an attributive construction is the argument of that adjective at the semantic layer. The same holds for the noun phrase of predicative constructions. Thus, the external argument noun is included into the number of arity being the first element of the valency frame. For this reason, the example below is monovalent. In the valency frame it occupies the Position 1, and its syntactic function is EXTERNAL_COMP.

Ex.: *en ny bil* (a new car): attributive structure with external argument ‘*bil*’

. *bil er ny* (the car is new): predicative structure with external argument ‘*bil*’

Internal arguments are complements that are subcategorised for by the adjective; the example below contains an external and an internal argument, thus it is divalent. The external argument (NP: Peter) in Position 1 is treated in the above described way. It occupies the Position 2 and it has the function PREPOSITIONAL_OBJECT; the preposition ‘*for*’(with) governs an NP.

Ex.: *Peter er glad for sin nye bil* (Peter is happy with his new car)

2.3.4 Predicative constructions with clausal complements

A few adjectives may have clausal complements (syntactic function: CLAUSCOMP), and these complements may also occur in extraposed construction, this means that the sentential subject appears in final position, the subject position, Position 1 is filled by a

FORMAL_COMP, which does not correspond to any semantic argument and is not counted in the arity value, thus the example below is monovalent.

Ex.: at ryge er farligt (lit.: to smoke is dangerous)
det er farligt at ryge (lit.: it is dangerous to smoke)

2.3.5 Optionality of complements

Two types of optionality are distinguished in STO:

- Complements being syntactically optional, but semantically obligatory (i.e. the argument is semantically implied)
Ex.: *de forskellige meninger* (the different opinions),
viz. it is semantically implied that there are at least two meanings that are different from each other in some respects)
Vi er forskellige (fra hinanden) (we are different (from each other))
Hans æbler er forskellige (fra de andre) (i smag)
(His apples are different (from the others) (in taste))
- Complements being both syntactically and semantically optional
Ex.: *pigen er sød (mod dyrene)/(ved børnene)/(over for sine venner)*
(the girl is nice (to the animals/children/her friends) viz. the complementation is optional as the adjective does not imply a semantic argument.

In the valency frame, an optional complement may have the value YES or YES_GEN for the feature OPTIONAL. The value YES_GEN is applied if the optional complement gives rise to more than one single description of the same construction as explained below.

2.3.6 Optionality in syntactic units and descriptions

The following examples specify the encoding of the various combinations of complements and optionality in appropriate syntactic units.

(a) Adjectives with complements being syntactically optional, but semantically obligatory are provided with one single divalent valency frame comprising the optional complement. Usually, this type of adjectives subcategorizes for one particular preposition (e.g. *for*'), thus the adjective is described with one single syntactic unit which reflects the construction with an optional PREPOSITIONAL_OBJECT in Position 2, the value of the Optional element is YES.

Ex.: *Børnene er bange (for hunden)* (The children are afraid (of the dog))

(b) Adjectives with complements being syntactically and semantically optional are provided with one monovalent valency frame without the optional complement, i.e. the adjective is described with one single syntactic unit which reflects this construction.

Ex.: *pigen er sød* (the girl is nice)

Further, the adjective is provided with one (or more) valency frame(s) comprising the optional complement(s). Often, this type of adjectives subcategorizes for more than one particular preposition (e.g. '*ved*', '*mod*' and '*over for*' see example below). Accordingly,

an appropriate number of syntactic units (comprising one description each) describe the various constructions. (The relationship between a syntactic unit and description(s) is presented in Chapter 3, The data). The adjective 'sød' subcategorizes for a prepositional phrase which may be introduced by three different prepositions, thus the adjective is provided with three syntactic units describing one particular construction each. The optional complement is always a PREPOSITIONAL_OBJECT in Position 2, the value of the Optional element is YES_GEN.

Ex.: *pigen er sød (mod dyrene)* (the girl is nice (to/ towards the animals))
pigen er sød (ved børnene) (the girl is nice (to/ towards the children))
pigen er sød (over for sine venner) (the girl is nice (to/ towards her friends))

According to the general method adopted in STO, syntactic differences between constructions of a word must be reflected by different descriptions. This results in more syntactic units if the optional complement of an adjective can be realized with various syntactic structures.

In the example below, if the prepositional phrase (viz. the optional complement) is represented in the construction, there is a prepositional object control in evidence (viz. the NP contained in the prepositional phrase is the subject of the infinitive clause.) Of course, this is not the case if the optional complement is omitted.

Ex.: (a) *det er godt (for hende) at rejse* (lit.: It is good (for her) to travel)
 (b) *at rejse er godt (for hende)* (lit.: to travel is good (for her))

In order to treat the control phenomenon appropriately, four descriptions are needed in STO: two for constructions with the prepositional complement incl. the appropriate control and different syntactic structures of the obligatory complement, and accordingly two without the optional complement.

Ex.: (a) *det er godt for hende at rejse* (with control)
 (b) *det er godt at rejse* (without control)
 (c) *at rejse er godt for hende* (with control)
 (d) *at rejse er godt* (without control)

2.3.7 Syntactic functions of complements

The internal arguments of adjectives have various syntactic realizations, these are the following:

- ACOMP denotes nominal complements
- Ex.: *Han er hende taknemmelig* (lit.: He is her grateful)
- PREPOSITIONAL_OBJECT denotes prepositional complements
- Ex.: *Han er god til at tegne* (lit.: He is good to draw)
Maria er bange for hunde og heste (Maria is afraid of dogs and horses)
- CLAUSCOMP denotes clausal complements
- Ex.: *Det er godt at du er kommet/At du er kommet er godt*
 (lit.: It is good that you came/ That you came is good)
- FORMAL_COMP denotes formal subjects
- Ex.: *Det er usædvanligt at ryge* (lit.: It is unusual to smoke)

- EXTERN_COMP bruges for eksterne argumenter
Peter er glad; de glade børn (Peter is happy; the happy children)

2.3.8 Valency patterns of adjectives

In the following, the most representative valency frame types are presented. For many of these frames we also provide the most relevant features encoded in the syntactic description.

Monovalent adjectives

1. Valency frames with an external argument:

- Adjectives that can be used attributively:
Ex.: *Annes røde bil* (Anne's red car)
(Adj_Function: ATTRIBUTIVE;
Position: 1; Function: EXTERN_COMP; Optional: NO; Syntactic_Label: NP; Casus: UNMARKED)
- Adjectives that can be used predicatively:
Ex.: *Annes bil er rød* (Anne's car is red)
(Adj_Function: PREDICATIVE;
Position: 1; Function: EXTERN_COMP; Optional: NO; Syntactic_Label: NP; Casus: NON_GENITIVE)

2. Valency frames without an external argument

A Clause is the syntactic subject. Semantically this subject corresponds to an internal argument. These valency frames have an alternate frame representing the corresponding extraposed constructions. The formal subject of the extraposed constructions is the expletive *det*. The Clause in these constructions corresponds still to an internal argument in a semantic representation.

- Frames for constructions without extraposition:
Ex.: *at han kom i dag, er tilfældigt* (lit.: that he came today is accidental)
(Adj_Function: PREDICATIVE;
Position: 1; Function: CLAUSECOMP; Optional:NO; Syntactic_Label: CLAUSE;
Clause_Type:THAT_CLAUSE; Control: NOCONTROL; Coref: NOCOREF)
Ex.: [*at du kom/at købe en smoking*] *er fint* (lit.: that you came/to buy a dinner jacket is fine)
(Adj_Function: PREDICATIVE;
Position: 1; Function: CLAUSECOMP; Optional:NO; Syntactic_Label: CLAUSE;
Clause_Type: THAT_CLAUSE; Control: NOCONTROL; Coref: NOCOREF;
Syntactic_Label: CLAUSE; Clause_Type:INFINITIVE; Control: WITHOUTCONTROL; Coref: WITHOUTCOREF)
- Frames for the corresponding extraposed constructions:
Ex.: *det er tilfældigt at han kom i dag* (lit.: it is accidental, that he came today)

(Adj_Function: PREDICATIVE;
 Position: 1; Function: FORMAL_COMP; Optional:NO; Syntactic_Label: NP;
 expletive: DET; Casus: NOMINATIVE;
 Position: 2; Function: CLAUSECOMP; Optional:NO; Syntactic_Label: CLAUSE;
 Clause_Type:THAT_CLAUSE; Control: NOCONTROL; Coref: NOCOREF)

Divalent adjectives

1. Valency frames with an external nominal argument and an internal argument which is a prepositional phrase. This prepositional phrase may be obligatory or optional:
 - The complement within the prepositional phrase is a noun (this is the default value)

Ex.: *jeg er tom for ideer* (lit.: I'm empty of ideas)
 (Adj_Function: PREDICATIVE;
 Position: 1; Function: EXTERN_COMP; Optional:NO; Syntactic_Label: NP;
 Casus: NON_GENITIVE ;
 Position: 2; Function: PREPOSITIONAL_OBJECT; Optional:NO;
 Syntactic_Label:PP; Introducer: for; NP_Type: N)
 Ex.: *huset ligger bekvemt (for hendes arbejde)* lit.:(the house is situated favourably (for her work)
 (Adj_Function: PREDICATIVE;
 Position: 1; Function: EXTERN_COMP; Optional:NO; Syntactic_Label: NP;
 Casus: NON_GENITIVE;
 Position: 2; Function: PREPOSITIONAL_OBJECT; Optional:YES_GEN;
 Syntactic_Label:PP; NP_Type: N)
 - The complement within the prepositional phrase is a noun phrase or an infinitive:

Ex.: *han er hurtig til [madlavning/at lave mad]*
 (lit.: he is quick to [cooking/to cook])
 (Adj_Function: PREDICATIVE;
 Position: 1; Function: EXTERN_COMP; Optional:NO; Syntactic_Label: NP;
 Casus: NON_GENITIVE; NPIndex: INDEXI;
 Position: 2; Function: PREPOSITIONAL_OBJECT; Optional:YES_GEN;
 Syntactic_Label:PP;
 Introducer: TIL; NP_Type: N; Clause_Type: NULL; Control:NOCONTROL;
 Coref: NOCOREF/
 Introducer: TIL; NP_Type: NULL; Clause_Type: INFINITIVE; Control:
 SUBJECTCONTROL; Coref: COI)
 - Further divalent types with an external nominal argument and an internal argument which is a prepositional phrase
 - The complement within the prepositional phrase is an infinitive with subject control
 - The complement within the prepositional phrase is a noun phrase, a that-clause or an infinitive with subject control.

- The complement within the prepositional phrase is a noun phrase, a that-clause or an interrogative clause.
2. Valency frames with an external argument and an internal argument which is a Clause:
 Ex.: *Engelsk er let at lære* (lit.: English is easy to learn)
 (Adj_Function: PREDICATIVE;
 Position: 1; Function: EXTERN_COMP; Optional:NO; Syntactic_Label: NP;
 Casus: NON_GENITIVE; NPIndex: INDEXI;
 Position: 2; Function: CLAUSECOMP; Optional:NO;
 Syntactic_Label:CLAUSE;
 Clause_Type: INFINITIVE; Control: WITHOUTCONTROL; Coref: WITHOUTCOREF)
 3. Valency frames without an external argument. The syntactic subject is a clause and the adjective subcategorizes also for a prepositional phrase. These adjectives may also occur in extraposed constructions.

A construction without extraposition:

Ex.: *at redde barnet var modigt af ham*
 (lit.: to save the child was courageous of him)
 (Adj_Function: PREDICATIVE;
 Position: 1; Function: CLAUSECOMP; Optional:NO; Syntactic_Label: CLAUSE;
 Clause_Type: INFINITIVE; Control: PREPOBJECTCONTROL; Coref: CON;
 Position: 2; Function: PREPOSITIONAL_OBJECT; Optional: NO;
 Syntactic_Label:PP;
 Introducer: af; NPIndex: N; Clause_Type: NULL; Control: NOCONTROL; Coref: NOCOREF)

The corresponding extraposed construction:

Ex.: *det var modigt af ham at redde barnet*
 (lit.: it was courageous of him to save the child)
 (Adj_Function: PREDICATIVE;
 Position: 1; Function: FORMAL_COMP; Optional:NO; Syntactic_Label: NP;
 expletive: DET; Casus: NOMINATIVE;
 Position: 2; Function: PREPOSITIONAL_OBJECT; Optional: NO;
 Syntactic_Label:PP;
 Introducer: af; NPIndex: N; Clause_Type: NULL; Control:NOCONTROL; Coref: NOCOREF;
 Position: 3; Function: CLAUSECOMP; Optional:NO; Syntactic_Label: CLAUSE;
 Clause_Type: INFINITIVE; Control: PREPOBJECTCONTROL; Coref: CON)

Trivalent adjectives

1. Adjectives with two prepositional phrases, in the second prepositional phrase the complement may be a noun phrase, a that-Clause, an interrogative Clause or an infinitive.

Ex.: *han var enig (med moderen) (om [flytningen/at de skulle flytte/ hvornår de skulle flytte/at flytte])*

(lit.: he was agreeing (with the mother) on ([the removal/that they should move/when they should move/to move]). (NB: The adjective 'enig' is translated into English by the verb 'agree'.)

(Adj_Function: PREDICATIVE;

Position: 1; Function: EXTERN_COMP; Optional:NO; Syntactic_Label: NP; Casus: NON_GENITIVE; NPIndex: INDEXI;

Position: 2; Function: PREPOSITIONAL_OBJECT; Optional: YES;

Syntactic_Label:PP;

Introducer: MED; NP_Type: N; Clause_Type: NULL; Control: NOCONTROL;

Coref: NOCOREF;

Position: 3; Function: PREPOSITIONAL_OBJECT; Optional: YES_GEN;

Syntactic_Label:PP;

Introducer: OM; NP_Type: N; Clause_Type: NULL; Control: NOCONTROL; Coref: NOCOREF;

Introducer: OM; NP_Type: NULL; Clause_Type: THAT_CLAUSE; Control:

NOCONTROL; Coref: NOCOREF;

Introducer: OM; NP_Type: NULL; Clause_Type: INTERROGATIVE_CLAUSE;

Control: NOCONTROL; Coref: NOCOREF;/

Introducer: OM; NP_Type: NULL; Clause_Type: INFINITIVE; Control:

SUBJECTCONTROL; Coref: COI)

2. Adjectives with one nominal complement and one prepositional phrase. The prepositional complement may be a noun phrase, a that-Clause or an infinitive.

Ex.: *De var (mig) behjælpelig (med [kufferterne /at kufferterne blev hentet/at hente kufferterne])*

(lit.: They were (me) assistant (with [the luggage/ that the luggage was brought/to bring the luggage]))

2.3.9 Syntactic units and alternations

As it has been briefly mentioned in the introductory section on basic principles and description elements, syntactic alternations are encoded in the same syntactic unit. For adjectives, the set of alternations encoded as such is limited to cases where the adjectives occur in constructions, where the syntactic subject is a clause. These adjectives may also occur in corresponding extraposed constructions, where the syntactic subject is the expletive *det*, e.g.

Alternation within one Syntactic unit:

Ex.: D1t - *At han kom i dag, er tilfældigt* (lit. That he came today is fortuitous)

Da1ext - *Det er tilfældigt at han kom i dag* (lit. It is fortuitous that he came today)

An exception is the following alternation type, where only the first construction is encoded in STO:

Ex.: *Skibet er let at styre/Et let skib at styre*
(lit. The ship is easy to steer/an easy ship to steer): two alternate constructions
Da2Pi – *Skibet er let at styre*: the construction encoded

Constructions such as those in the following example are not coded as alternations, the two syntactic descriptions occur in two different syntactic units:

Ex.: *Hans mening er sagen uvedkommende/Hans mening er uvedkommende for sagen*
(lit.: His opinion is the subject irrelevant/His opinion has not bearing on the subject)

Syntactic unit 1: Da1exN - *Hans mening er sagen uvedkommende*

Syntactic unit 2: Da1exP-for - *Hans mening er uvedkommende for sagen.*

3 The data: STO Syntax represented as XML elements

This chapter describes the structure of the XML files in which the STO syntax is encoded (Section 1), then the number of relevant XML elements is provided (Section 2) and, finally, an example of syntactic encoding in XML is given (Section 3).

3.1 The Structure of the STO Syntax XML Files

The root element in the lexicon's XML file is *STO_Syntax*. The root element contains the following five elements:

1. *Morph_Syn_Units*
 2. *Descriptions*
 3. *Selfs*
 4. *Constructions*
 5. *Phrases*
- The *Morph_Syn_Units* element contains a sequence of *Mu_Synu* elements which connect STO syntactic units to STO morphological units through their identifier.
 - The *Descriptions* element contains a sequence of *Description* elements.
 - The *Selfs* element contains a sequence of *Self* elements.
 - The *Constructions* element contains a sequence of *Construction* elements.
 - The *Phrases* element contains a list of phrases (*NP*, *PP*, *Clause*, *ADVP* and *AP* elements).

In the following we describe how syntax is represented in STO through these XML elements.

3.1.1 *Mu_Synu*, *Mu_Id* and *Spelling* elements

Mu_Synu elements connect the syntactic descriptions of words, expressed in *Synu* elements with the identifiers of the morphological descriptions of the same words (*Mu_id* elements).

A *Synu* element stands for a syntactic unit and contains a description of a syntactic pattern of a word. A word may occur in different syntactic contexts, thus the same morphological unit can be bound to more *Synu* elements. The *Mu_Synu* element also contains one or more *Spelling* elements which give the spelling(s) of the described word. The structure of the *Mu_Synu* elements is given in Figure 1.

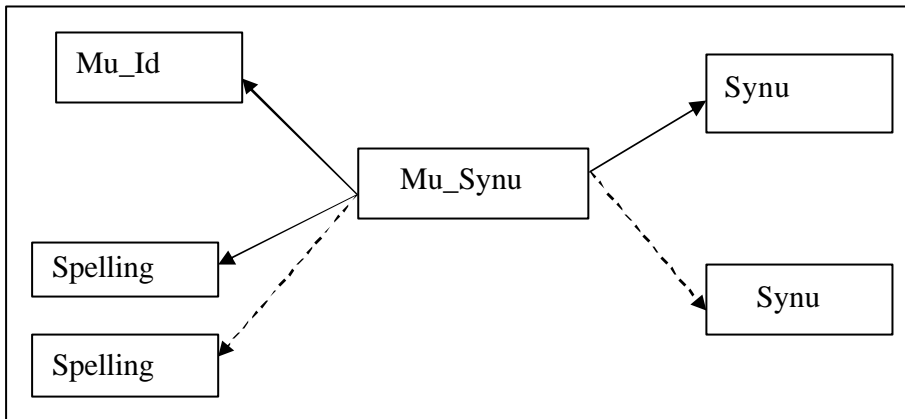


Figure 1: The structure of *Mu_Synu* elements

An example of an *Mu_Synu* is given in (1).

(1)

```

<Mu_Synu>
  <Mu_Id>"AFBETALE"</Mu_Id>
  <Spelling>afbetale</Spelling>
  <Synu Id="SYNU_AFBETALE_1">
  ...
</Synu>
  <Synu Id="SYNU_AFBETALE_2">
  .....
</Synu>
</Mu_Synu>
  
```

In (1) the word with Spelling “afbetale” and with the *Mu_Id* “AFBETALE” has two syntactic readings encoded in the two *Synus* with the identifiers “SYNU_AFBETALE_1” and “SYNU_AFBETALE_2”.

3.1.2 *Synu, Description and Construction elements*

A syntactic pattern may have different realizations, thus a syntactic unit can contain more syntactic descriptions (*Description* elements). The binding between a syntactic unit and a description is done through the *Synu_Description* elements. A syntactic description consists of the characteristics of the actual word (the head) in the syntactic pattern (*Self* element) and the valency pattern (*Construction* element). The structure of a syntactic unit (*Synu*) is presented in figure 2.

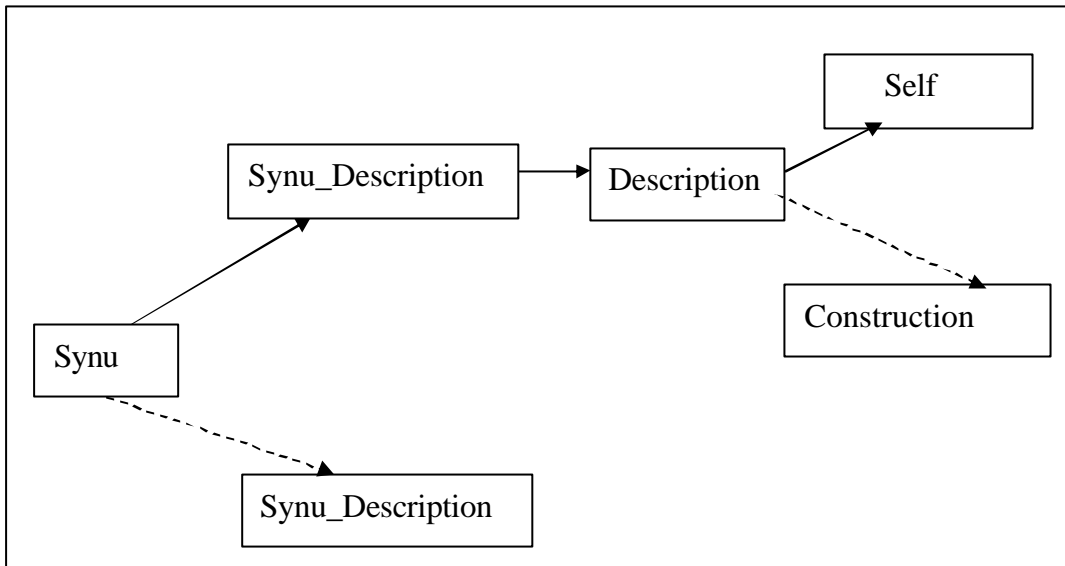


Figure 2: The structure of a syntactic unit (Synu)

An example of these elements in XML is given in (2).

(2)

```

<Synu Id="SYNU_AFBETALE_1">
  <Synu_Description Description_Id="Dv2N">
    <Example> fordi vi først skal afbetale gælden.</Example>
  </Synu_Description>
</Synu>
<Synu Id="SYNU_AFBETALE_2">
  <Synu_Description Description_Id="Dv2P-på">
    <Example> i stand til selv at købe og afbetale på
    huset</Example>
  </Synu_Description>
</Synu>

```

.....

```

<Description Id="Dv2N" Construction_Id="Cv2N" Self_Id="have_NO_NO">
  <Naming>divalent: NP, obligatory, NP</Naming>
  <Example> ..når de beregner den tre-delte statsskat med
  skattesatser på 6, 12, og ca. 50% Korpuskilde:
  BGH\AVIS40.88</Example>
</Description>

```

In (2), the two syntactic units from example (1) are shown, together with the *Description* for the first unit. The first syntactic unit refers to a description “Dv2N” which, as explained in the *Naming* element, describes a divalent construction (the word subcategorizes for two constituents). The *Example* element under the *Description*

contains a corpus example of a divalent syntactic pattern. The description refers to a *Self* element via the identifier “have_NO_NO” and a *Construction* element via the identifier “Cv2N”. The *Self* element says that the word described is a verb and that it takes “have” (have) as auxiliary. The *Example* element may contain either a standard example illustrating the construction type but not necessarily the particular entry word, or an individual example containing the entry word itself in its narrow context.

A valency pattern (*Construction* element) is described by the canonical position of its constituents (*Position* element), the syntactic function (*Function* elements) of this position, the optionality status of the position (*Optional* element) and the syntactic characteristics of the phrases which may occur in that position (*Constituent* elements). The constituents may be phrases of different types: nominal phrases (*NP* elements), clauses (*Clause* elements), prepositional phrases (*PP* elements), adjectival phrases (*AP* elements) and adverbial phrases (*ADVP* elements). The structure of a *Construction* is given in figure 3.

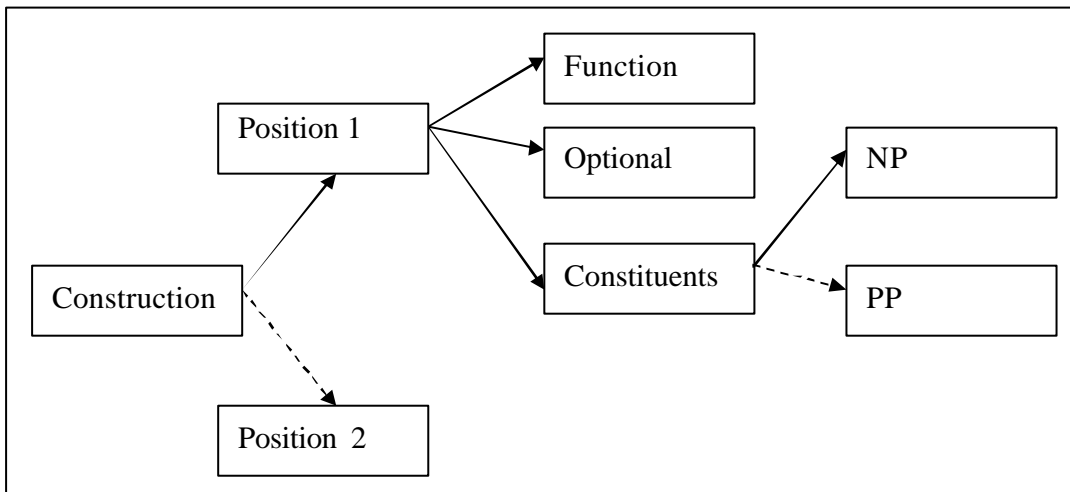


Figure 3: The structure of a Construction.

An XML example containing the Construction and the phrases from (2) is presented in (3) below.

(3)

```

<Construction Id="Cv2N">
  <Position Number="1">
    <Function>SUBJECT</Function>
    <Optional>NO</Optional>
    <Constituents>
      <Constituent Syntactic_Label="NP" Phrase_Id="NOM"/>
    </Constituents>
  </Position>
  <Position Number="2">
    <Function>OBJECT</Function>
    <Optional>NO</Optional>
  </Position>
</Construction>
  
```

```

        <Constituents>
            <Constituent Syntactic_Label="NP" Phrase_Id="ACC"/>
        </Constituents>
    </Position>
</Construction>

```

The *Construction* “Cv2N” in (3) consists of two positions: the first position has the syntactic function of subject, is obligatory (the *Optional* element contains the value “NO”) and refers to a phrase which is a nominal phrase. The *Phrase* element contains the characteristics of nominal phrases that may occur in this position (e.g. the case of the nominal phrase must be NOMINATIVE). The second position is also obligatory, it has the syntactic function “OBJECT” and the phrase in it must also be a nominal (and its case must be accusative).

3.1.3 Self elements

There are different types of elements that describe the Self for verbs, adjectives, nouns and adverbs.

3.1.4 Self for verbs

The *Self* for verbs (*Self* element with *Cat* attribute having value VERB) contains the following elements: *Reflexive* (values YES, NO), *Particle* (NO, or particle if the verb is phrasal), *Auxiliary* (values HAVE or VÆRE), *Passive* (values YES, NO, or UNMARKED), *Modal* (values YES or NO), *Aux* (saying whether the verb is an auxiliary with the values YES or NO).

3.1.5 Self for nouns

The *Self* for nouns (*Self* element with *Cat* attribute having value NOUN) contains one element, *Definite_Suffix_Allowed*, which can take the values YES or NO.

3.1.6 Self for adjectives

The *Self* for adjectives (*Self* element with *Cat* attribute having value ADJECTIVE) contains one element *Adj_Func*, which can take the values PREDICATIVE or ATTRIBUTIVE.

3.1.7 Self for adverbs

The *Self* for adverbs (*Self* element with *Cat* attribute having value ADVERB) contains the following elements: *Modifying*, *Modifiable*, *Context*, *Appear_In_Predicative*, *Fundament_Field*, *Prefinit_Field*, *Nexus_Field*, *Manner_Field*, *Predicative_Field*, *Bound_Adverbial_Field*, *Prepositional_Object_Field*, *Final_Field*, *Pre_Or_Postmodifies*, *Pre_Or_Postmod_Inf*, *Appear_In_Question*.

The *Modifying* element describes the expressions that the adverb can modify and can contain the following string values:

- ADJECTIVE
- ADJECTIVE PHRASE

- ADVERB
- PREPOSITIONAL PHRASE
- IKKE (viz. negation expressed by ‘no’)
- NOUN PHRASE
- VERB
- VERB PHRASE
- SENTENCE
- EXC (viz. the formal subject)

The *Modifiable* element describes whether the adverb can be modified and whether it can be modified by the adverb *meget* (very) and it contains one of the following string values: YES, NO, YES_MEGET.

The *Content* element describes whether the adverb occurs in particular contexts and it can contain the following string values: NO_RESTRICTIONS, POSITIVE or NEGATIVE.

The *Appear_In_Predicative* element describes whether the adverb can occur in predicative constructions and it contains the string values YES or NO.

The *Fundament_Field* element describes whether the adverb can occur in the fundament field and it contains the string values YES, NO or NA (Not Applicable).

The *Prefinit_Field* element describes whether the adverb can precede a finite verb and it contains the string values YES, NO or NA (Not Applicable).

The *Nexus_Field* element describes where in the nexus the adverb can occur and can contain the following string values:

- ONLY_THEME
- ONLY_FOCUS
- BOTH
- NEITHER
- NA

The *Manner_Field*, *Predicative_Field*, *Bound_Adverbial_Field*, *Prepositional_Object_Field*, *Final_Field* elements describe whether the adverb can occur in the manner, predicative, prepositional object and final field, respectively. All of them can contain the string values YES, NO or NA (Not Applicable).

The *Pre_Or_Postmodifies* element indicates whether the adverb precedes or follows the modified expressions and it can take the following string values:

- PREMODIFIES
- POSTMODIFIES
- BOTH
- NA

The *Pre_Or_Postmod_Inf* element indicates whether the adverb precedes or follows the modified infinitive and it takes the same string values as the *Pre_Or_Postmodifies* element.

The *Appear_In_Question* element indicates whether the adverb can occur in questions and it can take the following string values:

- YES_ONLY
- NO
- NO_RESTRICTIONS
- NA

3.1.8 *The elements describing valency patterns*

The *Position* in a Construction can range from 1 to 4 (monovalent, divalent, trivalent and tetravalent constructions). Zerovalent constructions do not contain any *Position* element.

The *Function* element can contain one of the following string values:

- SUBJECT
- OBJECT
- INDIRECT_OBJECT
- PREPOSITIONAL_OBJECT
- OBJECT_PREDICATE
- FORMAL_SUBJECT
- ADVERBIAL
- SUBJECT_PREDICATE
- REL_GEN
- EXTERN_COMP
- FORMAL_COMP
- CLAUSCOMP
- ACOMP
- SPEC_N
- SOM_PP.

The *Optionale* element contains one of the three string values: YES, NO, and YES_OVERG. YES_OVERG is used for the complements of adjectives and indicates that the description of the optional element can result in syntactic ambiguity.

In the following we describe the elements and the values which apply for each phrasal type.

NP elements

NP elements describe nominal phrases, and they are bound to elements describing the phrase's case, reflexivity, type (viz. expletive or no), definiteness and to an index (NPIndex) which is used to mark co-reference in constructions with control phenomena.

(such as NPIndex=I coref=COI). The possible content of the elements mentioned above is given in the tables below.

| ELEMENT | VALUES | | | | | |
|---------|------------|----------|------------|----------------------|-------------|----------|
| Case | NOMINATIVE | GENITIVE | ACCUSATIVE | ACCUSATIVE_INDIREKTE | NO_GENITIVE | UNMARKED |

| ELEMENT | VALUES | | |
|-----------|--------|----|----------|
| Reflexive | YES | NO | UNMARKED |

| ELEMENT | VALUES | | |
|-----------|--------|-----|----|
| Expletive | DET | DER | NO |

| ELEMENT | VALUES | | | | | |
|---------|--------|---|---|---|---|---|
| NPIndex | I | J | K | L | M | N |

CLAUSE elements

Clause elements contain information about the clausal type, control and co-reference, expressed in the subelements *Clause_Type*, *Control* and *Coref*. The possible content of these elements is given in the following tables.

| ELEMENT | VALUES | | | |
|-------------|-------------|----------------------|------------|--------------------------|
| Clause_Type | THAT_CLAUSE | INTERROGATIVE_CLAUSE | INFINITIVE | INFINITIVE_WITHOUT_INTRO |

| ELEMENT | VALUES | | | |
|---------|-----------|-----------------------|-------------------|---------------|
| Control | NOCONTROL | WITHOUTCONTROL | SUBJECTCONTROL | OBJECTCONTROL |
| | | INDIRECTOBJECTCONTROL | PREPOBJECTCONTROL | RAISING |

| ELEMENT | VALUES | | | | | | | |
|---------|---------|--------------|-----|-----|-----|-----|-----|-----|
| Coref | NOCOREF | WITHOUTCOREF | COI | COJ | COK | COL | COM | CON |

The values in the elements *Control* and *Coref* depend on the type of control. Only infinitive clauses can have control (value is not NOCONTROL).

PP

PP elements describe prepositional phrases through the subelements *Introducer*, *Clause_Type*, *NP_Type*, *Control*, *Coref*. The *Introducer* element has as its value the preposition which introduces the phrase. *Clause_Type* is reserved for prepositional phrases, where the complement is a clause. In this case the element *NP_Type* has the value NA (Not Applicable). If the prepositional complement is an NP, the *Clause_Type* element has value NA and *NPIndex* contain the index of the NP (as for the NP element). The values for the *Control* and *Coref* elements are the same as for clauses.

NOTE: *ADVP* and *AP* are empty.

3.2 The number of main XML elements

The number of the main elements in the STO Syntax lexicon are the following:

- Mu_Synu: 45309
- Synu: 57887
- Synu_Description: 60381
- Description: 1363
- Self: 103
- Construction: 985
- NP:11
- PP:113
- Clause: 9

3.3 An example of XML

In the following a complete example of the encoding of the verb *afbetale* is given as an example:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<STO_Syntax xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="STO_Syntax.xsd">
  <Morph_Syn_Units>
    ...
    <Mu_Synu>
      <Mu_Id>"AFBETALE"</Mu_Id>
      <Spelling>afbetale</Spelling>
      <Synu Id="SYNU_AFBETALE_1">
        <Synu_Description Description_Id="Dv2N">
          <Example> fordi vi først skal afbetale gælden.</Example>
        </Synu_Description>
      </Synu>
      <Synu Id="SYNU_AFBETALE_2">
        <Synu_Description Description_Id="Dv2P-på">
          <Example> i stand til selv at købe og afbetale på huset
          </Example>
        </Synu_Description>
      </Synu>
    </Mu_Synu>
    ...
  </Morph_Syn_Units>
  <Descriptions>
    ...
    <Description Id="Dv2N" Construction_Id="Cv2N" Self_Id="have_NO_NO">
      <Naming>divalent: NP, obligatory, NP</Naming>
      <Example> ..når de beregner den tre-delte statsskat med skattesatser
        på 6, 12, og ca. 50% Korpuskilde: BGH\AVIS40.88</Example>
    </Description>
    <Description Id="Dv2P-på" Construction_Id="Cv2P-på"
    Self_Id="have_NO_NO">
      <Naming>divalent: NP, oblique PP prep=på</Naming>
      <Example>Det er som et kludetæppe, som jeg har nørklet på hele
      livet,</Example>
    </Description>
    ...
  </Descriptions>
  <Selfs>
    ...
    <Self Cat="VERB" Id="have_NO_NO">
      <Reflexive>NO</Reflexive><Particle>NO</Particle>
      <Auxiliary>have</Auxiliary><Passive>NO</Passive>
      <Modal>NO</Modal><Aux>NO</Aux>
    </Self>
    ...
  </Selfs>
  <Constructions>
    ...
    <Construction Id="Cv2N">
      <Position Number="1">
        <Function>SUBJECT</Function> <Optional>NO</Optional>
        <Constituents>
          <Constituent Syntactic_Label="NP" Phrase_Id="NOM"/>
        </Constituents>
      </Position>
    </Construction>
  </Constructions>
</STO_Syntax>
```

```

        <Position Number="2">
            <Function>OBJECT</Function> <Optional>NO</Optional>
            <Constituents>
                <Constituent Syntactic_Label="NP" Phrase_Id="ACC"/>
            </Constituents>
        </Position>
    </Construction>
    <Construction Id="Cv2P-på">
        <Position Number="1">
            <Function>SUBJECT</Function> <Optional>NO</Optional>
            <Constituents>
                <Constituent Syntactic_Label="NP" Phrase_Id="NOM"/>
            </Constituents>
        </Position>
        <Position Number="2">
            <Function>PREPOSITIONAL_OBJECT</Function> <Optional>NO</Optional>
            <Constituents>
                <Constituent Syntactic_Label="PP"
hrase_Id="PÅ_NP_NOC_NOC"/>
            </Constituents>
        </Position>
    </Construction>
    ...
</Constructions>
<Phrases>
    <NP Id="NOM">
        <Casus>NOMINATIVE</Casus> <Reflexive>NO</Reflexive>
        <Expletive>NO</Expletive> <Definite>UNMARKED</Definite>
        <NPIndex>I</NPIndex>
    </NP>
    <NP Id="ACC">
        <Casus>ACCUSATIVE</Casus> <Reflexive>UNMARKED</Reflexive>
        <Expletive>NO</Expletive> <Definite>UNMARKED</Definite>
        <NPIndex>J</NPIndex>
    </NP>
    <PP Id="PÅ_NP_NOC_NOC">
        <Introducer>på</Introducer> <Clause_Type>NA</Clause_Type>
        <NP_Type>N</NP_Type> <Control>NOCONTROL</Control>
        <Coref>NOCOREF</Coref> </PP>
    </Phrases>
</STO_Syntax>

```

4 Appendix

4.1 The XML Schema file for the STO syntax, *STO_Syntax.xsd*:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:annotation>
    <xs:documentation>This is the export file of the syntax of the STO
      Danish lexicon.</xs:documentation>
  </xs:annotation>
  <xs:element name="STO_Syntax">
    <xs:annotation>
      <xs:documentation> The root element STO_Syntax contains the
        elements Morph_Syn_Units, Descriptions, Selsf,
        Constructions
        and Phrases</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="Morph_Syn_Units" />
        <xs:element ref="Descriptions" />
        <xs:element ref="Selsf" />
        <xs:element ref="Constructions" />
        <xs:element ref="Phrases" />
      </xs:sequence>
    </xs:complexType>
    <!-- key and keyrefs giving restrictions on ids and their references -->
    <xs:key name="des_id">
      <xs:selector xpath="Descriptions/Description"/>
      <xs:field xpath="@Id" />
    </xs:key>
    <xs:keyref name="des_id_ref" refer="des_id">
      <xs:selector
        xpath="Morph_Syn_Units/Mu_Synu/Synu/Synu_Description"/>
      <xs:field xpath="Description_Id" />
    </xs:keyref>
    <xs:key name="cos_id">
      <xs:selector xpath="Constructions/Construction"/>
      <xs:field xpath="@Id" />
    </xs:key>
    <xs:keyref name="cos_id_ref" refer="cos_id">
      <xs:selector xpath="Descriptions/Description"/>
      <xs:field xpath="Construction_Id" />
    </xs:keyref>
    <xs:key name="s_id">
      <xs:selector xpath="Selsf/Self" />
      <xs:field xpath="@Id" />
    </xs:key>
    <xs:keyref name="s_id_ref" refer="s_id">
      <xs:selector xpath="Descriptions/Description"/>
      <xs:field xpath="Self_Id" />
    </xs:keyref>
    <xs:key name="phr_id">
      <xs:selector xpath="Phrases/*" />
      <xs:field xpath="@Id" />
    </xs:key>
    <xs:keyref name="phr_id_ref" refer="phr_id">
      <xs:selector
        xpath="Constructions/Construction/Position/Constituents/Constituent"/>
      <xs:field xpath="Phrase_Id" />
    </xs:keyref>
  </xs:element>
</xs:schema>
```

```

</xs:element>
<xs:element name="Morph_Syn_Units">
  <xs:annotation>
    <xs:documentation> The element Morph_Syn_Units contains
      Mu_Synu elements which connect STO syntactic units to STO
      morphological units</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element maxOccurs="unbounded" ref="Mu_Synu"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Mu_Synu">
  <xs:annotation>
    <xs:documentation> The element Mu_Synu connects the id of a
      morphological unit (Mu_Id) and its possible spellings
      (Spelling element) to one or more syntactic
      units (Synu). </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Mu_Id" type="Short_String_Type"/>
      <xs:element maxOccurs="30" name="Spelling"
        type="Short_String_Type"/>
      <xs:element maxOccurs="unbounded" ref="Synu"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Synu">
  <xs:annotation>
    <xs:documentation> The element Synu contains a reference to
      one or more syntactic descriptions
      (Synu_Description).
  </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element maxOccurs="100" name="Synu_Description"
        type="Synu_Description_Type"/>
    </xs:sequence>
    <xs:attribute name="Id" type="xs:ID" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="Example" type="Long_String_Type"/>
<xs:element name="Descriptions">
  <xs:annotation>
    <xs:documentation> The element Description contains the
      sequence of Description elements (i.e. descriptions of
      syntactic valency) contained in the STO-lexicon.
  </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element maxOccurs="unbounded" name="Description"
        type="Description_Type"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="Selfs">
  <xs:annotation>

```

```

        <xs:documentation> The element Selfs contains the sequence
        of descriptions of valency heads (Self) in the STO-
base.
        </xs:documentation>
    </xs:annotation>
    <xs:complexType>
        <xs:sequence>
            <xs:element maxOccurs="unbounded" name="Self"
                type="Self_Type"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="Constructions">
    <xs:annotation>
        <xs:documentation> The element Constructions contains the
        sequence of construction elements (Construction) which
        express valency descriptions without information about the
        valence head.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
        <xs:sequence>
            <xs:element maxOccurs="unbounded" name="Construction"
                type="Construction_Type"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="Phrases">
    <xs:annotation>
        <xs:documentation> The element Phrases contains the
        sequence
        of Phrase elements constituting the syntactic
        complements
        (at the moment NP, PP, Clause).
        </xs:documentation>
    </xs:annotation>
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="unbounded"
                name="NP" type="NPP_Type"/>
            <xs:element minOccurs="0" maxOccurs="unbounded"
                name="PP" type="PP_Type"/>
            <xs:element minOccurs="0" maxOccurs="unbounded"
                name="Clause" type="ClauseP_Type"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:simpleType name="Short_String_Type">
    <xs:restriction base="xs:string">
        <xs:maxLength value="100"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="Synu_Description_Type">
    <xs:annotation>
        <xs:documentation> An element of Synu_Description_Type may
        contain an example (Example element) and a reference
        to the id of a Description.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
        <xs:element minOccurs="0" ref="Example"/>
    </xs:sequence>
    <xs:attribute name="Description_Id" type="xs:IDREF"
        use="required"/>
</xs:complexType>
<xs:simpleType name="Long_String_Type">
    <xs:restriction base="xs:string">

```



```

        <xs:maxLength value="1000"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="Description_Type">
<xs:annotation>
    <xs:documentation> An element of type Description_Type may contain
        an example (Example element) and may have a Name (Naming).
        It has a unique identifier (Id) and a reference to a
        description of the valency head (Self_Id) and to a valency
        pattern (Construction_Id)</xs:documentation>
</xs:annotation>
<xs:sequence>
    <xs:element minOccurs="0" name="Naming"
        type="Long_String_Type"/>
    <xs:element minOccurs="0" name="Example"
        type="Long_String_Type"/>
</xs:sequence>
<xs:attribute name="Id" type="xs:ID" use="required"/>
<xs:attribute name="Construction_Id" type="xs:IDREF"/>
<xs:attribute name="Self_Id" type="xs:IDREF" use="required"/>
</xs:complexType>
<xs:complexType name="Self_Type">
<xs:annotation>
    <xs:documentation> An element of type Self_Type has a unique
        identifier (Id) a category for the word and contains a list of
        elements which vary if the valency head is a verb, a noun, an
        adjective or an adverb.</xs:documentation>
</xs:annotation>
<xs:choice>
<!-- Self for Verbs -->
    <xs:sequence>
        <xs:element name="Reflexive" type="YN_Type"/>
        <xs:element name="Particle"
            type="Short_String_Type"/>
        <xs:element name="Auxiliary" type="Auxiliary_Type"/>
        <xs:element name="Passive" type="YNU_Type"/>
        <xs:element name="Modal" type="YN_Type"/>
        <xs:element name="Aux" type="YN_Type"/>
    </xs:sequence>
<!-- Self for Substantives -->
    <xs:sequence>
        <xs:element name="Definite_Suffix_Allowed"
            type="YN_Type"/>
    </xs:sequence>
<!-- Self for Adjectives -->
    <xs:sequence>
        <xs:element name="Adj_Func" type="Adj_Func_Type"/>
    </xs:sequence>
<!-- Self for Adverbs -->
    <xs:sequence>
        <xs:element name="Modifying" type="Modifying_Type" />
        <xs:element name="Modifiable" type="Modifiable_Type" />
        <xs:element name="Context" type="Context_Type" />
        <xs:element name="Appear_In_Predicative"
            type="YN_Type" />
        <xs:element name="Fundament_Field" type="YNNNA_Type" />
        <xs:element name="Prefinit_Field" type="YNNNA_Type" />
        <xs:element name="Nexus_Field"
            type="Nexus_Field_Type" />
        <xs:element name="Manner_Field" type="YNNNA_Type" />
        <xs:element name="Predicative_Field" type="YNNNA_Type"

```

```

/>

```

```

        <xs:element name="Bound_Adverbial_Field"
            type="YNNA_Type" />
        <xs:element name="Prepositional_Object_Field"
            type="YNNA_Type" />
        <xs:element name="Final_Field" type="YNNA_Type" />
        <xs:element name="Pre_Or_Postmodifies"
            type="Pre_Or_Postmodifies_Type" />
        <xs:element name="Pre_Or_Postmod_Inf"
            type="Pre_Or_Postmodifies_Type" />
        <xs:element name="Appear_In_Question"
            type="Appear_In_Question_Type" />
    </xs:sequence>
</xs:choice>
<xs:attribute name="Cat" type="Cat_Type" use="required"/>
<xs:attribute name="Id" type="xs:ID" use="required"/>
</xs:complexType>
<xs:simpleType name="Cat_Type">
    <xs:restriction base="xs:string">
        <xs:enumeration value="ADJECTIVE"/>
        <xs:enumeration value="ADVERB"/>
        <xs:enumeration value="NOUN"/>
        <xs:enumeration value="VERB"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="YN_Type">
    <xs:restriction base="xs:string">
        <xs:enumeration value="YES"/>
        <xs:enumeration value="NO"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="YNU_Type">
    <xs:restriction base="xs:string">
        <xs:enumeration value="YES"/>
        <xs:enumeration value="NO"/>
        <xs:enumeration value="UNMARKED"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="YNNA_Type">
    <xs:restriction base="xs:string">
        <xs:enumeration value="YES"/>
        <xs:enumeration value="NO"/>
        <xs:enumeration value="NA"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Optional_Type">
<xs:annotation>
    <xs:documentation> Type describing whether a complement is
        optional (YES or YES_OVERG), obligatory (NO). YES_OVERG is only
        used for adjectival complements and indicates that removing the
        optional complement results in syntactic ambiguities, because the
        given pattern is already in the STO base.</xs:documentation>
</xs:annotation>
    <xs:restriction base="xs:string">
        <xs:enumeration value="YES"/>
        <xs:enumeration value="NO"/>
        <xs:enumeration value="YES_OVERG"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Auxiliary_Type">
<xs:annotation>
    <xs:documentation> Type of auxiliary that the verb takes (have or
        være?)</xs:documentation>
</xs:annotation>

```

```

        <xs:restriction base="xs:string">
            <xs:enumeration value="have"/>
            <xs:enumeration value="være"/>
        </xs:restriction>
    </xs:simpleType>
<xs:simpleType name="Adj_Func_Type">
<xs:annotation>
    <xs:documentation> The adjective appears in predicative or attributive
        position in the given Description.</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
        <xs:enumeration value="PREDICATIVE"/>
        <xs:enumeration value="ATTRIBUTIVE"/>
    </xs:restriction>
</xs:simpleType>

<!-- Types describing the Self of adverbs -->
<xs:simpleType name="Modifying_Type">
<xs:annotation>
    <xs:documentation> Constituents that the adverb can modify
        </xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
        <xs:enumeration value="ADJECTIVE"/>
        <xs:enumeration value="ADJECTIVE PHRASE"/>
        <xs:enumeration value="ADVERB"/>
        <xs:enumeration value="PREPOSITIONAL PHRASE"/>
        <xs:enumeration value="IKKE"/>
        <xs:enumeration value="NOUN PHRASE"/>
        <xs:enumeration value="VERB"/>
        <xs:enumeration value="VERB PHRASE"/>
        <xs:enumeration value="SENTENCE"/>
        <xs:enumeration value="EXC"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Modifiable_Type">
<xs:annotation>
    <xs:documentation> Can the adverb be modified?</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
        <xs:enumeration value="YES"/>
        <xs:enumeration value="NO"/>
        <xs:enumeration value="YES_MEGET"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Context_Type">
<xs:annotation>
    <xs:documentation> Type of context in which the adverb must occur
</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:string">
        <xs:enumeration value="NO_RESTRICTIONS"/>
        <xs:enumeration value="POSITIVE"/>
        <xs:enumeration value="NEGATIVE"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Nexus_Field_Type">
<xs:annotation>
    <xs:documentation> Where in the nexus field the adverb can occur
    </xs:documentation>
</xs:annotation>
    <xs:restriction base="xs:string">
        <xs:enumeration value="ONLY_THEME"/>

```

```

        <xs:enumeration value="ONLY_FOCUS"/>
        <xs:enumeration value="BOTH"/>
        <xs:enumeration value="NEITHER"/>
        <xs:enumeration value="NA"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Pre_Or_Postmodifies_Type">
    <xs:restriction base="xs:string">
        <xs:enumeration value="PREMODIFIES"/>
        <xs:enumeration value="POSTMODIFIES"/>
        <xs:enumeration value="BOTH"/>
        <xs:enumeration value="NA"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Appear_In_Question_Type">
    <xs:restriction base="xs:string">
        <xs:enumeration value="YES_ONLY"/>
        <xs:enumeration value="NO"/>
        <xs:enumeration value="NO_RESTRICTIONS"/>
        <xs:enumeration value="NA"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="Construction_Type">
<xs:annotation>
    <xs:documentation> An element of Construction_Type has an unique
        identifier referred to in the Description element
        and describes valency elements by the contained Position elements.
    </xs:documentation>
</xs:annotation>
    <xs:sequence>
        <xs:element name="Position" type="Position_Type"
            minOccurs="0" maxOccurs="4"/></xs:sequence>
        <xs:attribute name="Id" type="xs:ID" use="required"/>
</xs:complexType>
<xs:complexType name="Position_Type">
<xs:annotation>
    <xs:documentation> An element of Position_Type gives the canonical
        position of valency constituents (attribute Number) contained in
        the element Constituents, expresses the constituents function
        (element Function) and indicates whether the constituents are
        optional or obligatory in the element Optional.
    </xs:documentation>
</xs:annotation>
    <xs:sequence>
        <xs:element name="Function" type="Function_Type" />
        <xs:element name="Optional" type="Optional_Type" />
        <xs:element name="Constituents" type="Constituents_Type" />
    </xs:sequence>
    <xs:attribute name="Number" type="PNumber_Type" use="required"/>
</xs:complexType>
<xs:simpleType name="PNumber_Type">
    <xs:restriction base="xs:string">
        <xs:enumeration value="1"/>
        <xs:enumeration value="2"/>
        <xs:enumeration value="3"/>
        <xs:enumeration value="4"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="Constituents_Type">
    <xs:sequence>
        <xs:element name="Constituent" type="Constituent_Type"
maxOccurs="10"/>
    </xs:sequence>

```

```

</xs:complexType>
<xs:complexType name="Constituent_Type">
<xs:annotation>
  <xs:documentation> Syntactic label of a constituent and reference to
    the constituent phrase for NP,PP and Clauses.
  </xs:documentation>
</xs:annotation>
  <xs:attribute name="Syntactic_Label" type="Syntactic_Label_Type"
    use="required"/>
  <xs:attribute name="Phrase_Id" type="xs:IDREF"/>
</xs:complexType>
<xs:simpleType name="Syntactic_Label_Type">
  <xs:restriction base="xs:string">
    <xs:enumeration value="NP"/>
    <xs:enumeration value="PP"/>
    <xs:enumeration value="CLAUSE"/>
    <xs:enumeration value="ADVP"/>
    <xs:enumeration value="AP"/>
    <xs:enumeration value="OTHER"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Function_Type">
<xs:annotation>
  <xs:documentation> List of the possible complement functions
</xs:documentation>
</xs:annotation>
  <xs:restriction base="xs:string">
    <xs:enumeration value="ACOMP"/>
    <xs:enumeration value="ADVERBIAL"/>
    <xs:enumeration value="CLAUSCOMP"/>
    <xs:enumeration value="EXTERN_COMP"/>
    <xs:enumeration value="FORMAL_COMP"/>
    <xs:enumeration value="FORMAL_SUBJECT"/>
    <xs:enumeration value="INDIRECT_OBJECT"/>
    <xs:enumeration value="OBJECT"/>
    <xs:enumeration value="OBJECT_PREDICATE"/>
    <xs:enumeration value="PREPOSITIONAL_OBJECT"/>
    <xs:enumeration value="REL_GEN"/>
    <xs:enumeration value="SOM_PP"/>
    <xs:enumeration value="SPEC_N"/>
    <xs:enumeration value="SUBJECT"/>
    <xs:enumeration value="SUBJECT_PREDICATE"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="NPP_Type">
<xs:annotation>
  <xs:documentation> NP and its possible characteristics
  </xs:documentation>
</xs:annotation>
  <xs:sequence>
    <xs:element name="Casus" type="Casus_Type" />
    <xs:element name="Reflexive" type="YNU_Type" />
    <xs:element name="Expletive" type="Expletive_Type" />
    <xs:element name="Definite" type="Definite_Type"/>
    <xs:element name="NPIndex" type="NPIndex_Type" />
    <xs:element name="Optionalfeature" type="xs:string"
      minOccurs="0"/>
  </xs:sequence>
  <xs:attribute name="Id" type="xs:ID" use="required"/>
</xs:complexType>
<xs:simpleType name="Casus_Type">
  <xs:restriction base="xs:string">
    <xs:enumeration value="NOMINATIVE"/>

```

```

        <xs:enumeration value="ACCUSATIVE"/>
        <xs:enumeration value="GENITIVE"/>
        <xs:enumeration value="NO_GENITIVE"/>
        <xs:enumeration value="UNMARKED"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Expletive_Type">
    <xs:restriction base="xs:string">
        <xs:enumeration value="YES"/>
        <xs:enumeration value="NO"/>
        <xs:enumeration value="det"/>
        <xs:enumeration value="der"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Definite_Type">
    <xs:restriction base="xs:string">
        <xs:enumeration value="NO"/>
        <xs:enumeration value="UNMARKED"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="NPIndex_Type">
<xs:annotation>
    <xs:documentation> NP and its possible characteristics
    NPIndex and COREF are used express coreference between NPs
    in main clauses and in subordinate constructions
    </xs:documentation>
</xs:annotation>
    <xs:restriction base="xs:string">
        <xs:enumeration value="I"/>
        <xs:enumeration value="J"/>
        <xs:enumeration value="K"/>
        <xs:enumeration value="L"/>
        <xs:enumeration value="M"/>
        <xs:enumeration value="N"/>
        <xs:enumeration value="NOINDEX"/>
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="PP_Type">
<xs:annotation>
    <xs:documentation> PP and its possible characteristics
</xs:documentation>
</xs:annotation>
    <xs:sequence>
        <xs:element name="Introducer" type="Short_String_Type" />
        <xs:element name="Clause_Type" type="Clause_T_Type" />
        <xs:element name="NP_Type" type="NP_T_Type" />
        <xs:element name="Control" type="Control_Type"/>
        <xs:element name="Coref" type="Coref_Type" />
        <xs:element name="Optionalfeature" type="xs:string"
            minOccurs="0"/>
    </xs:sequence>
    <xs:attribute name="Id" type="xs:ID" use="required"/>
</xs:complexType>
<xs:simpleType name="Clause_T_Type">
<xs:annotation>
<xs:documentation> Types of clause that are prepositional complements
    </xs:documentation>
</xs:annotation>
    <xs:restriction base="xs:string">
        <xs:enumeration value="INFINITIVE"/>
        <xs:enumeration value="INTERROGATIVE_CLAUSE"/>
        <xs:enumeration value="THAT_CLAUSE"/>
        <xs:enumeration value="WH_CLAUSE"/>
    </xs:restriction>

```

```

        <xs:enumeration value="NA" />
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="NP_T_Type">
    <xs:restriction base="xs:string">
        <xs:enumeration value="N" />
        <xs:enumeration value="NA" />
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Control_Type">
<xs:annotation>
    <xs:documentation> Types of control</xs:documentation>
</xs:annotation>
    <xs:restriction base="xs:string">
        <xs:enumeration value="SUBJECTCONTROL" />
        <xs:enumeration value="OBJECTCONTROL" />
        <xs:enumeration value="INDIRECTOBJECTCONTROL" />
        <xs:enumeration value="PREPOBJECTCONTROL" />
        <xs:enumeration value="RAISING" />
        <xs:enumeration value="NOCONTROL" />
        <xs:enumeration value="WITHOUTCONTROL" />
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="Coref_Type">
    <xs:restriction base="xs:string">
        <xs:enumeration value="COI" />
        <xs:enumeration value="COJ" />
        <xs:enumeration value="COK" />
        <xs:enumeration value="CON" />
        <xs:enumeration value="WITHOUTCOREF" />
        <xs:enumeration value="NOCOREF" />
    </xs:restriction>
</xs:simpleType>
<xs:complexType name="ClauseP_Type">
<xs:annotation>
    <xs:documentation> Clause and its possible characteristics
    </xs:documentation>
</xs:annotation>
    <xs:sequence>
        <xs:element name="Clause_Type" type="Clause_T2_Type" />
        <xs:element name="Finite" type="YN_Type" />
        <xs:element name="Control" type="Control_Type" />
        <xs:element name="Coref" type="Coref_Type" />
        <xs:element name="Optionalfeature" type="xs:string"
            minOccurs="0" />
    </xs:sequence>
    <xs:attribute name="Id" type="xs:ID" use="required" />
</xs:complexType>
<xs:simpleType name="Clause_T2_Type">
<xs:annotation>
    <xs:documentation> Types of clause that occur as complements (not
    prepositional complements)</xs:documentation>
</xs:annotation>
    <xs:restriction base="xs:string">
        <xs:enumeration value="INFINITIVE" />
        <xs:enumeration value="INTERROGATIVE_CLAUSE" />
        <xs:enumeration value="THAT_CLAUSE" />
        <xs:enumeration value="INFINITIVE_NO_INTRO" />
    </xs:restriction>
</xs:simpleType>
</xs:schema>

```