A proposal for an extended feature system

By Ole Togeby, EUROTRA-DK

This is a discussion paper about a system of semantic features which can serve 5 different disambiguation machineries: 1) content word disambiguation in analysis, i.e. computation of which among several readings of a given noun, adjective or verb is the one actualised in the given sentence; 2) disambiguation of prepositions, i.e. computation of the semantic relation of a pp to its govenor; the disambiguation should be based on the combination of preposition and the semantic features of the noun; 3) structural disambiguation in analysis, i.e. selection of the correct IS tree structure among several correctly generated ERS tree structures; 4) disambiguation in synthesis, i.e. seléction of the best word among two or more possible target language translations of the same reading from the same source language word; 5) time disambiguation, i.e. computation of the IS time value on the sentence node as a function of both the ERS tense value of the verb, and the time value of the (prepositional and other) modifiers.

The main idea in this paper is, that it is the same set of semantic features that serve all the 4 purposes, and that the semantic feature system only works if combined with a set of semantic relations on circumstantials and perhaps on arguments too.

As a basis for the discussion I will take the semantic feature system proposed by the German group by Cornelia Zelinsky-Wibbelt. This system is primarily designed for a purpose not mentioned here: transfer disambiguation. This purpose is not discussed here because there will be no use of comparison of semantic features in transfer, if the translation is carried out in the way it is described in the Reference Manual. So here I will suggest some changes and extensions made necessary by the 4 purposes which have been mentioned.

In the following I will first present the whole system of features, and give the definitions of the features. Then I will give a presentation of the four different disambiguation strategies, and argue that they will make it necessary to design the semantic feature system the way I propose.Then I will present a system of semantic relations for modifiers and its use in disambiguation.

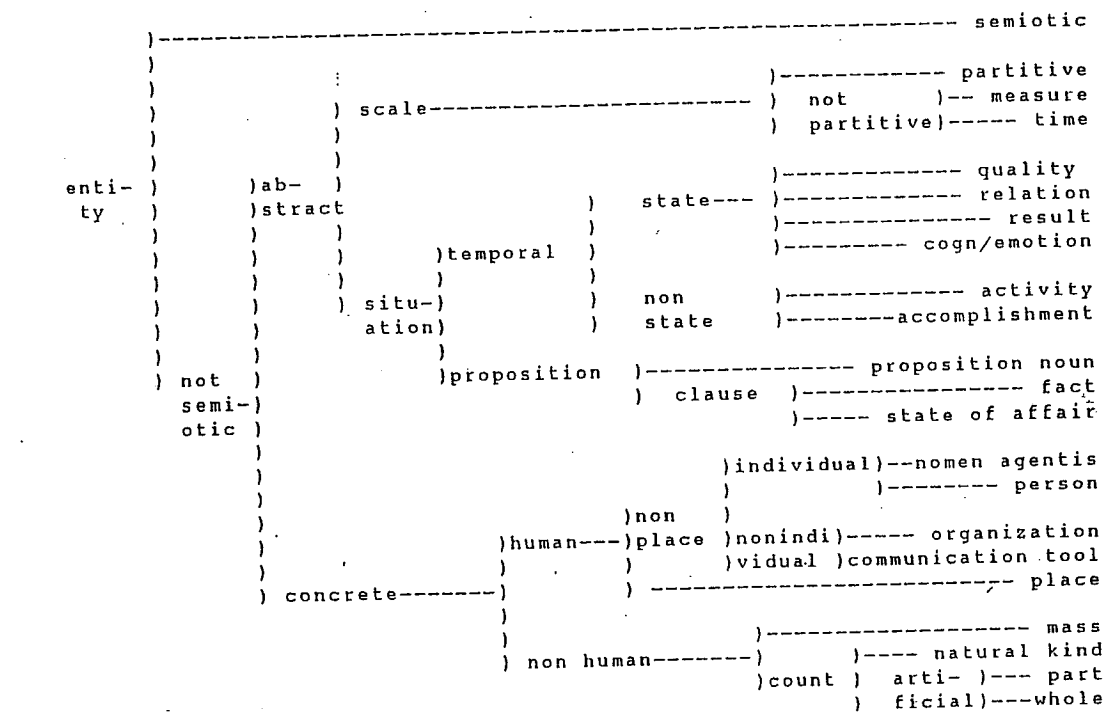
I The system of features for nouns

The system of features is a hierarchical ordered set of values of one semantic feature. There is only one hierarchical relation, viz disjunction; in this system there is no conjunction.

There are terminal feature values and non terminal feature values. To a noun in a given reading there will be assigned one and only one terminal feature value, never two feature values to one reading, and never only a nonterminal value. The nonterminal values are used only in the frames of verbs, nouns, adjectives and prepositions governing the noun bearing the feature.

The concept of polysemy is defined by the system so that a word has two readings if it in one context will have one of the terminal feature values, and another in another context.

Consequently the feature value is defined by the distribution of the noun to which it is assigned , not by the conceptual meaning of the noun. The decision of which feature is assigned to a given noun is therefore taken on basis of tests of the type: if you can put the word X in the context Y X Z, then it has the feature value f1; if it cannot, it has the feature value f2. In addition to the tests there will normally be a conceptual definition primarily ment as a mnemotechnical help for the linguist.



The system is not complete. In fact it is possible to make indefinitly many distinctions in a given vocabulary. In this system destinctions are only made if they prove useful in disambiguation of at least one (corpus type) example.

The system is not consistent from a cognitive point of view. Although semiotic nouns could be divided into PART NOUNS and WHOLE NOUNS as ARTIFICIAL nouns are, they are not divided in this system, because it has not proved to make any disambiguation possible, i.e. we have not found any frame bearing word that select only PART SEMIOTIC nouns, or only WHOLE SEMIOTIC nouns.

From a distributional point of view the system has to be consistent. It means that if we have a frame selection rule that selects a given feature value, it will select all members of the set of words with that feature value, and no other words.

The feature matching system is not a killer system, but a preference system, which works in the following way:

If the feature value of one of the readings of a noun is the same as the value specified in the frame, this reading will unify before other readings with feature values not identical to the specified value. If none of the competing readings have the same feature value as the frame, the reading with the value with the nearest ancestor in the feature value tree common with the value specified in the frame, will be selected. And if two readings have the same ancestor common with the value specified in the frame the reading which is nearest relative to the frame value (i.e. first descendent of the common ancestor) will be selected.

In section III in this paper it is described how this counting system can be implemented. In section IV it will be shown how the linguist can simulate the counting mechanism when constructing the lexical frames on nouns, varb and prepositions.

II. The features

The definitions given in this paragraph are first attempts to make a consistent set of definitions. They, no doubt, have to be elaborated and changed when more empirical material is taken into account.

It is not essential for the functioning of this system that the features are universal, but it is believed that it is possible to define a set of features common to all the EUROTRA languages, and in any case, it will be very efficient if the features are defined in the same way for all the languages.

Perhaps the system of features is not enough fine grained to capture all the distinctions necessary for disambiguation. New distinctions can be added, especially distinctions of the terminal values, if they prove to be useful.

SEMIOTIC nouns
If the word X fit in at least one of the contexts

she read X,
he interpreted X,
they translated X

then it has the feature value semiotic; if not, it has the feature value NOT SEMIOTIC. Semiotic nouns are both concrete and abstract, viz in the token and type sense repectively.

Examples: 'afsnit', 'forslag', 'aftale', 'liste', 'betegnelse'.

ABSTRACT-CONCRETE
a not semiotic noun is CONCRETE if it fits in all the contexts:

They saw X disappear
(i.e. x is the accusative in an accusative + infinitive construction, this test does not go for Greek):

They saw X pass by
He looked directly at X
It was placed on X
It was to the right of X

'Concrete' means here 'perceivable'.

If the word does not fit into any of the contexts, it is ABSTRACT.

SCALE - SITUATION
Of the abstract nouns some have arguments, some have not; All abstract nouns with arguments are situation nouns. All abstract nouns with explicative clause ('the fact that...') and all np which are clauses or infinitives are situation nps too. Abstract nouns which are not situations are scale nouns. Scale nouns have no arguments, they denote 'time','measure' or 'parts', they are scales denoting the extension of other things.

PARTITIVE - NOT PARTITIVE
Among scale nouns the partitive nouns have allready been recognized on ECS as the nouns wich scat=class or scat=specifier which in the surface structure are heads in the dependency structure, but in the IS structure modifiers to the noun, which was its modifier in the surface, e.g.

ECS		IS	
part of	the production =>	production	part
np	pp	gov	mod

PARTITIVE nouns
fit in the context:

only one X of something

Examples: 'sektor', 'side', 'halvdel', 'omr]de', 'aspect'.

MEASURE - TIME
Abstract nouns which are not partitive, are either measure or time.

MEASURE nouns
are nouns that have an appositional slot in the np structure on ECS, and denotes measure in any dimension or field. They can always be modified by a numeral, because they all denote units of measurement which

can be counted

Examples: 'meter', '$', 'decibel', 'grad', uge, minut, dag.

Measure nouns are divided into UNITS and TIME UNITS.

If x fits into one of the contexts

it lasted for two x (necessarily idefinite)
it was done in two x (necessarily indefinite)

it is a time unit noun.

TIME nouns
are abstract not partitive scale nouns which denotes periods of time. They fit in at least one of the contexts:

it happened in X (possibly definite)
it took place in x (possibly definoite)

Examples: 'efterkrigstiden', 'genopbygningsperiod', 'Xmas', 'the future'. Time nouns are sort of proper names of time points or periods in the history, and consequently always definite in meaning, and somtimes in form too.

TEMPORAL - PROPOSITION
Abstract situations can be either temporal or propositions.

PROPOSITIONS are clauses, infinitives or words that can be explicated by a that-clause or an infinitive: 'the fact that...', 'the problem that he was never there',

TEMPORAL nouns are nouns with an argument frame.

STATE - NON STATE
Temporal nouns are divided into states and non states.

NON STATE nouns
are nomina actionis, and fit into the contexts

X takes place
X finder sted
X foregaar

Non states nouns can be either activity nouns or accomplishment nouns.

Temporal nouns which do not fit into the contexts are states.

States are divided into quality, relation, result and cogn/emotion.

QUALITY nouns

are nomina qualitatis, i.e. deadjectival nouns (or other nouns) with only arg_1, viz the subjective genitive:

Examples: 'identitet', 'stoerrelse', 'laengde'.

RELATION nouns

are nomina relationis, i.e. nouns derived from stative verbs (or other nouns) with arg_1 (possibly subjective genitive) and arg (2) (never objective genitive). The stative verb from which the noun is derived cannot be passivized. Nomen relationis fit into the context:

Der eksisterede (en) X [preposition] [np]

Der eksisterede en uafhaengighed af verdensmarkedet

Under this category of relation goes non human nomina agentis e.g. 'factor'.

Examples: 'afhaengighed', 'uafhaengighed', 'faktor', 'position (?)'.

RESULT nouns

are nomina resultatis, i.e. nouns derived as an inner object of a transitive accomplishment verb, meaning 'the result of what is done' by the verb, e.g. 'produce the production', 'invest the investment'. Nomina resultatis fit into at least one of the contexts:

foretage X
lave X
goere X
der sker X

Examples: 'verdensproduktion', 'undtagelse', 'investering_2, 'initiativ' (?), 'virkning', konsekvens, 'struktur', 'omskiftelse_2', foranstaltning_2.

Note that many deverbal nouns are ambigous; they can be both result nouns and accomplishment or activity nouns, e.g. 'investering_1', 'investering_2', 'foranstaltning_1', 'foranstaltning_2'.

COGN/EMOTION nouns
are nouns with an arg 1 (possibly subjective genitive) and arg 2 (never objective genitive).They fit into the context:

af X

being a modifier denoting the psychological cause of an act or of behavior:

'De kom af interesse'
'De boejede sig af frygt'

No examples found in corpus.

ACTIVITY nouns
non state noun, derived from acticity verbs, nouns which fit into the context:

X varede i saa og saa lang tid
X var svaer at udfoere

Examples: 'databehandling', 'fabriksautomation', 'processtyring', 'telekommunikation', 'anvendelse', 'indflydelse', 'produktion_1', 'forskning'.

ACCOMPLISHMENT nouns
non state nouns which fit into the context:

X tog saa og saa lang tid
X var svaer for dem at gennemfoere

accomplishment nouns are nouns derived from accomplishment and achievement verbs.

Examples: 'revolution', 'investering_1', 'omskiftelse_1', 'foranstaltning_1', genopbygning', 'udforskning'.

PROPOSITIONS
are NPs which are either proposition nouns or clauses or infinitives.

PROPOSITION nouns
are nouns which can be explicated by either a that clause or an infinitive. They fit into the context:

Den X, at...

Examples: 'fordel', 'mulighed', 'problem', 'tendens', 'situation'.

HUMAN - NONHUMAN

concrete nouns are divided into human and non human.

HUMAN nouns
fit in one or more of the contexts:

X gjorde det med vilje
X holdt op med at goere det
X tog initiativet til at goere det
X siger at ...

In human nouns are included:

(concrete) nomina agentis which denote human beings, organizations, which can act like human beings, communication tools, which can be said to communicate like humans
persons, normally called their names,
places, which by metonymy can act like organizations.

NOMINA AGENTIS
are human nouns which have an argument structure (if they are derived from transitive verbs, nomina agentis derived from intransitive verbs normally have no argument structure, e.g. 'the runner', although they can have the inner object of the verb as an argument: 'the runner of the mile'.

Examples: 'fabrikanten af ...', 'tilskuer til ...', 'herre over ...'.

ORGANIZATION nouns
are human nouns which denote groups of persons acting like a (juridical) person. They fit into the contexts:

personerne i X
Xs medlemmer
repraesentanter for X.

They do not fit into contexts like:

*X doede af det og det
*X drak to glas vand

Excamples: 'hjemmemarked', 'datterselskab', 'selskab', 'oekoenomi', 'industri', 'regering'

COMMUNICATION TOOL nouns
are machines which can be said to communicate like humans and even take decisions. They have no argument structure, and do not fit into the contexts

for persons:

*X doede af tuberkulose
*X drak to glas vand.

Examples: 'persondatamat', 'videobaandoptager'.

PLACE nouns
are a subgroup to human because of the frequently used metonymi with a city acting like an organization; but this category has to be distinguished from the other human nouns. Place nouns fit into the contexts:

fra X,) denoting directions
til X)

Note that all persons will pass this test because all persons have a position in space. The subcategory of (geographical) place fits into the context:

X ligger nord for Y

Examples: 'Europa', 'USA', 'Japan', 'verden', 'ildlinjen'

Non human are divided into count (countable) and non count.

NON COUNT nouns
cannot be both singular and plural, they fit into the contexts

noget x
meget X

COUNT nouns
can be both plural and singular (although both countable singulare tantum and prurale tantum do exist), they do not fit into the non count context.

Countable nouns are divided into natural kind and artificial:

Examples (not from corpus) 'vand', 'luft', 'sand'.

ARTIFICIAL
fits into the contexts:

De frembragte X
de lavede X
X er lavet af det og det
De faerdiggjorde X

(Note that all semiotics will pass this test too,

but are filtered out earlier in the procedure.)

NATURAL KIND nouns
are nouns which do not fit into artificial test.

Natural kind nouns are divided into part nouns and whole nouns.

PART nouns
are artificial nouns which have an argument structure with the corresponding whole as arg_1.and always the possibility of a place preposition: 'the door of/in the house' , ' the heel of/on the shoe'.

Examples: 'kredsloeb'

WHOLE nouns
are artificial nouns without this argument structure.

Examples: 'dataanlaeg', 'hoejteknologi, 'elektronik, 'infòrmationsteknologi'.

III. Implementation of the unification mechanism for nouns with inherited features

by Carsten K. Olsson and Anders Nygaard.

1.Introduction.
This paper is based on mul discussions with Ole Togeby, chapter 3.6.3 "Semantic Features of Nouns" in the new reference manual 4.0, and on the work done so far by Anders Nygaard and Carsten K. Olsson. The goal has been to introduce feature inheritance into the E-framework in a way which does not disrupt current usage of the framework.

In section 2 we present the main idea and in section 3 we report on possible (and necessary?) extensions to the basic approach.

2. Proposal.
The main idea is the introduction of a type of features that have hierarchically ordered values (fig 1).

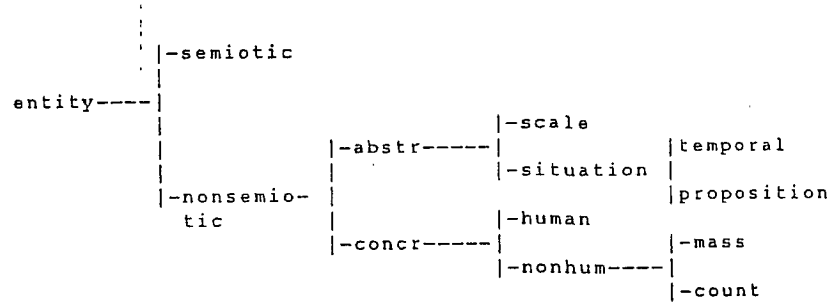


fig 1.

In such a tree, two values are compatible if one is a descendant of the other or they are equal. As long as we restrict ourselves to demanding compatibility in all translation situations, we can retain unification as the basis for our machine by representing such features as list-values with a tail variable. We would have e.g.

```
abstr      <->  [entity, nonsemiotic,abstr|_]
temporal   <->  [entity, nonsemiotic,abstr,situation,temporal|_]
```

which unify giving the latter as result. There are two main problems involved in such an approach: The design of a suitable user language for defining such features and the implementation of a conversion mechanism for changing e.g.

```
semfeat=temporal
```

into

```
[entity, nonsemiotic,abstr,situation,temporal|_]
```

for the rule interpreter. The latter is essentially trivial and will not concern us further here. We will, however, give a proposal for the user language. As the structure to be described is a tree, the most natural notation seems to be the familiar one of rewrite rules. The tree in fig. 1 could be described by the following entry in a .fd file:

```
entity = semiotic, nonsemiotic;
  nonsemiotic <- abstr, concr;
  abstr <- scale, situation;
  concr   <- human, nonhum;
  situation <- temporal, proposition;
  nonhum   <- mass, count .
```

where 'entity' is the name of the attribute, and the rest - as usual terminated by a dot - list the legal values for the attribute. The only difference from the current state of

affairs is that a number of inheritance properties are stated at the same time: If something is 'scale' it is also 'nonsemiotic', etc. You will note that values which, like 'abstr', both depend on others and have others as dependents appear twice; this seems to be unavoidable if the syntax is to be unambiguous as well as legible.

3. Extensions.
It is not possible in all cases of translation, or in all cases of analysis, to demand total compatibility between source and target level features, respectively frame and complement information. Let us take an example:

The new framework solves all problems

where

```
'solves'      has frame for subject  'entity = human'
  human       <->  [entity, nonsemiotic,concr,human|_]

'framework'   has  'entity = semiotic'
 semiotic     <->  [entity, semiotic|_]
```

We see that there is a mismatch between the feature demanded by the frame of the verb and the corresponding feature in the subject - reflecting the metaphorical use of the verb.

This and similar cases can be handled by allowing a slight change in the rule interpreter's semantics: If a certain (sub-)translation does not produce any results as a consequence of a mismatch in tree-valued features, chop off one element off the tail end of the list and try again, and continue in this manner until it succeeds or there is nothing left. The rationale behind this approach is that a failed translation is most likely due to some sort of metaphorical use of a word; the best translation/analysis can be found by locating the word which needs the least coercion (in the form of eliminating false or misleading specificity) in order to fit.

In the example given, we will probably want to get a result saying merely that the subject (and corresponding frame information on the verb, which presumably should still unify) has 'entity=entity'.

4. Final remarks.
The suggestion in section 2 is quite straightforward, but a number of questions arise about the extension proposed in section.

1) Should this retry-mechanism be used for all translations, or only the translations going from ERS to IS?

2) As the unification process involves two feature bundles, there are at least three possibilities for doing the 'trimming' procedure: Either of the two feature bundles can

be trimmed by itself, or both simultaneously. Which option should be chosen?

3) How does one handle the simultaneous appearance of several tree-valued features in the same feature bundle? Should the longest one be reduced first or both concurrently?

4) What happens in the case of several compatible translations, or, in general, several equally compatible (in the sense of appearing at the same stage of 'cutting') translations appearing? Should some means of distinguishing between them be introduced?

5) The mechanism as described is only able to choose among single feature bundles. Can it be extended to include trees, e.g. to distinguish between pp-attachments?

IV. Disambiguation of nouns,adjectives and verbs an analysis

(by Ole Togeby)

The disambiguation machinery works in the following way:

The german word Absatz has 4 different readings:

German	English	Danish
Absatz_1	=paragraph	= afsnit
Ansatz_2	=sale	= afsaetning
Absatz_3	=heel	= hael
(Treppen)absatz_4	= landing (of a staircase)	= afsats

Each of the four readings has a one-to-one translation into both English and Danish, so the destinction in German between the four readings is highly relevant in a translation project.

Now the four readings occur unambiguously in the following four german sentences:

Die Ergebnisse sind in Absatz 5 dieses Artikels aufgefuehrt.
Der pro-kopf Absatz hat sich erhoeht.
Der Absatz an meinem shuh ist kaputt.
Der Absatz wird renoviert.

The semantic feature system should now be designed to make it possible to disambiguate in these four contexts in source language analysis so that it can be calculated which reading is the actual one in each of the 4 examples.

Each of the 4 readings is in the source language dictionary described by a value from the semantic feature system:

```
Absatz_1 (paragraph) : semiotic
Ansatz_2 (sale): accomplishment
Absatz_3  (heel): part
Absatz_4 = (landing): part
```

And the other relevant words in the 4 example sentences, i.e. the nouns, verbs, adjectives and prepositions with frames, are described (in the source language dictionary) with frames with the feature values specified for each of their arguments, and for the modifiers, like the following:

```
auffuehren_1 (anfuehren, mention): arg-1=hum, arg_2=entity, mod_type=
            semiotic place,
auffuehren_2 (errichten, build): arg_1=hum,
arg_2=artificial,mod_type=place_where,

sich erhoehen: arg_1 =temporal (Frede er det rigtigt?)

in_3(SEMIOTIC PLACE): arg_1=semiotic
in-1(PLACE WHERE): ARG-1=place

pro-kopf: arg_1=temporal

an_7 (PLACE WHERE): arg_1=concrete

renovieren: arg_1=hum, arg_2=non-human
```

Comments on the list: 'Ein Stueck auffuehren' is taken as a fixed phrase because no other words than 'das Stueck' (or the name of the play) can be arg_2 of 'auffuehren' in this reading. I have not made descriptions of 7 different 'an' and 4 different 'in', but it is probable that it can be done (se section V, below). The capital letters after the prepositions indicate the semantic relation between the modifier pp and its govenor. (Se V, below!)

Then in the analysis of each sentence the semantic distance from the semantic feature value selected by the frame bearer (in this case the verb) to the semantic value of each of the readings of the slotfiller (in this case 4 readings of Absatz) is calculated, and the reading with the shortest semantic distance is chosen as the best analysis of the sentence. Example:

```
     Absatz_1
 Der Absatz_2 wird renoviert
     Absatz_3
     Absatz_4
```

In this example 'Absatz-1' is not selected because the distance from NON-HUMAN to SEMIOTIC is longer than the distance from NON-HUMAN to PART; and

'Absatz_2' is not selected because the distance from NON-HUMAN to ACCOMPLISHMENT is longer than the distance from NON-HUMAN to PART.'Absatz_3' and 'Absatz_4 will both be chosen because 'renovieren' selects NON-HUMAN as arg_2, and both 'Absatz_3' and Absatz_4' are coded as PART. Then the distance from NON-HUMAN to PART is the same in both cases.

The example shows that the semantic feature system is specified enough to exclude 'Absatz_1' and 'Absatz_2' but not god enough to select between 'Absatz_3 and _4. In this case we need an extra feature value distinguishing between two types of ARTIFICIAL: BUILDING - NOT BUILDING. I do not find it necessary to introduce this destinction in our system.But it can be added witout any other changes in the system.

By the same feature match rule both the verb and the noun in the same sentence can be disambiguated; see the following example:

```
     Absatz_1          aufgefuehrt_1
Der  Absatz_2 5 ist hier aufgefuehrt_2
     Absatz_3
     Absatz_4
```

In this example 8 different combinations of readings of the two ambiguous words have to be compared in the following way:

Absatz feature	auffuehren arg_2 1: entity	_2:artificial
1. semiotic	0.1	5.1
2. accomplishment	0.6	4.4
3. part	0.6	0.1
4. part	0.6	0.1

The figures indicate how the linguist can simulate how the system unify the inherited feature values, by 'walking' in the tree and counting the number of steps taken, coming from the verb frame feature value to the noun feature value. Every step upwards in the tree counts as 1.0 and every step downwards in the tree as 0.1.

From The figures it can be seen that the best readings are combinations of 'Absatz_1' and 'auffuehren_1', or of 'auffuehren_2' and 'Absatz_3 or 4'. In other words this example will be translated into either: 'the paragraph is mentioned here' or 'the heel is build here' or 'the landing is build here'. The first and the third translations are both correct translations - the sentence is ambiguous - and the second one is not wanted, but made because the semantic feature system does not destinguish between

the 'heel' reading and the 'landing' reading of 'Absatz'. If this destinction is made 'auffuehren_2' would select BUILDING as arg_2, and it would be the best reading with the semantic distance: 0.0.

Notice that is is necessary to use the frames of the adjectives too. In the example sentence

```
              Absatz_1
der pro-kopf  Absatz_2 hat sich erhoeht
              Absatz_3
              Absatz_4
```

It is possible to specify the feature restrictions on the arg_1 of the adjective, which is its govenor in this sentence where it is attributive.

pro-kopf: arg_1(govenor)=temporal

	distance:
Absatz_1:semiotic:	4.1
Absatz_2:accomplishment	0.2
Absatz_3:part	3.5
Absatz_4:part	3.5

There should not be any technical problems with this solution.

V. Disambiguation of prepositions

An even more important porpose with the feature system than disambiguation of the nouns and verbs is disambiguation of the prepositions. But this disambiguation of the preposition can not be made unles the semantic feature system is combined with a system of semantic relations between the modifying pps and the govenors. I will first show how the disambiguation will work in analysis, then I will introduce the system of semantic relations on modifiers.

Take the original example with 'Absatz' and 'auffuehren':

Die Ergebnisse sind in Absatz 5 dieses Artikels aufgefuerht

I take for granted that 'in' has at least 6 different readings in German, each reading selecting a semantic feature value of its arg_1

in_1 (dat)(PLACE WHERE), 'In Berlin': arg_1=place
in_2 (acc)(PLACE WHERETO) 'ins Zimmer kommen':arg_1=concrete
in_3 (SEMIOTIC PLACE) 'in diesem Buch': arg_1=semiotic
in_4 (TIME DURING) 'im Fruehling': arg_1=time
in_5 (MANNER) 'in Gold Bezahlen'arg_1=mass or quality (?)
in_6 (CIRCUMSTANCE) 'im Traum': arg_1=cogn/emotion (?)

The SR value is percolated from the selected reading of the preposition to the pp node, so that the modifier (and the pp in question can only be a modifier because the valency bound prepositions have been selected on ERS because of the ERS frame and the value of the preposition feature (se section VI, below).

The calculation of the semantic distances will be a two dimensional calculation again:

		Absatz reading nr.			
'in' reading	arg 1	1SEM	2ACCOMP	3PART	4 PART
1. PLACE WHERE	PLACE	4.1	3.5	1.4	1.4
2. WHERETO	CONCRETE	2.1	1.5	0.2	0.2
3. SEM PLACE	SEMIOTIC	0.0	1.6	1.6	1.6
4. TIME DUR	TIME	5.1	3.4	4.5	4.5
5 MAN	MASS/QUAL	4.1/6.1	3.5/2.2	1.3/5.5	1.3/5.5
6 CIRCUM	COGN/EMOT	6.1	2.2	5.5	5.5

From the figures it can be seen that it is the reading 1 of 'Absatz' combined with the reading 3 of 'in' which will be chosen as the combination with the shortest distance in semantic space.
We then have the situation with the sentence:

Die Ergebnisse sind in_3 Absatz_1 5 dieses artikels aufgefuehrt_1
aufgefuehrt_2

We then have not only to specify which semantic feature values will be selected in the arguments of the two readings of 'auffuehren' but also the types of modifier each of the readings will accept, and which types they reject.The rules will be something like the following:

auffuehren_1 (anfuehren, mention): arg_1=hum,
erg_2=entity, modifier=SEMIOTIC PLACE, modifier=~PLACE WHERE

auffuehren_2 (errichten, build): arg_1=hum,
erg_2=artificial, modifier= PLACE WHERE, modifier=~SEMIOTIC PLACE

Ergebnis: RESULT

But unfortunately this mechanism will not work; it is not possible in a preference system to specify what is not accepted, and the different semantic relation types, i.e. PLACE WHERE, TIME WHEN and SEMIOTIC PLACE are not structured in a hiararchical system like the semantic features.

There is another problem with the use of a preference system for modifier types: The same reading of 'auffuehren' will be selected by two different disambiguation mechanisms: the distance from arg_2 of 'auffuehren_1': ENTITY to RESULT is 0.6, and from

ARTIFICIAL to RESULT is 4.5.

And the distance from PLACE SEMIOTIC to PLACE SEMIOTIC is in any organization of the SR-system shorter than the distance from PLACE WHERE to PLACE SEMIOTIC. What will happen if the two preference mechanisms do not yield the same result I do not know. But the mechanism will not work in such cases.

So the disambiguation of the modifiers in the sentence can not be done by the proposed preference system of semantic features. It has to be done by a killer system, in which only objects in which the modifier_types selected by the verb frame unify with the modifier type of the actual modifier, are generated.

To each verb, noun and adjective (put not preposition) is assigned a frame specifying which modifier types it will accept, written with alternation, marked by ';', between two or more acceptable values:

'auffuehren_1': [modtype=time_when;place_semiotic;
frequency;manner.

'auffuehren_2': [modtype=time_when;duration_verb;timeboundary;
place_where;frequency;manner.

This notation means that 'auffuehren_1 does not accept modifiers of the type 'timeboundary'.

To every modifier there will be assigned a mod_type value, either by percolation to the pp node from the p node and the np node, or - in case of adverbs - from the dictionary.

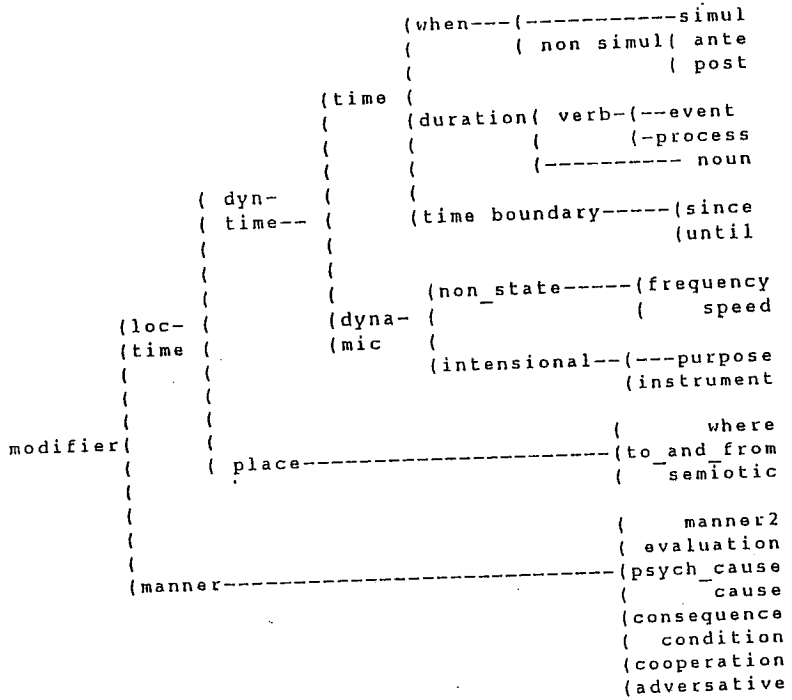
And only sentence trees in which the mod_type values selected by the frame match with the mod_type value of the adverbial, will be generated by the b-rules. There is no preference in this system; tree structures with no match will not be generated.

VI. The system of semantic relations of modifiers

The system of semantic relations on modifiers has to be adequate in relation to three purposes: 1) it has to contain as many types of semantic relations as is necessary for the disambiguation of prepositions, i.e. make it possible to distinguish between as many readings of the prepositions as are relevant for translation between the EUROTRA languages; 2) it has to be organized in a way so that it is easy to formulate the rule for rejection of modifiers;3) it has to yield time values of the modifies necassary for the calculation of the time value of the whole

sentence.

My tentative proposal is the following, but it certainly has to be revised and improved:



Note that the mod_type value of the modifier is written as a list of all the ancestors:

'now'
[mod_type=modifier,loctime,dyntime,time,when,simul],

and mod_type selected by the frame is written without the list of ancestors. The unification will only take place if the value selected by the frame is an ancestor of, or identical with the value of the modifier

Examples: TIME WHEN NONSIMUL: 'in 1982', DURATION VERB EVENT: 'in 3 days', DURATION VERB PROCESS: 'for 3 days', DURATION NOUN: 'of 3 days', TIME BOUNDARY SINCE: 'since the start of the project', TIME BOUNDARY UNTIL: 'until the end of the project', FREQUENCY: 'twice a week' SPEED: 'too slowly', 'in normal speed', PURPOSE: 'for fun', INSTRUMENT: 'with a knife', PLACE WHERE: 'in London', PLACE TO AND FROM: 'from Saar Bruecken', PLACE SEMIOTIC: 'in the document', EVALUATION: 'but effectively', PSYCH_CAUSE:

'of fear', CONDITION: 'in case of fire', ADVERSATIVE:
'against all odds', CAUSE: 'based on this idea',
CONSEQUENCE 'so that ...', COOPERATION 'with a friend'

VII. Structure disambiguation (pp attachment)

The problem of attachment can be stated as follows: any sentence of normal length will in analysis when using standard eurotra analysis modules create more than one attachment pattern, or as I would prefer to call it:geometry on ecs level. Of these different structures we only need one on IS level. Sometimes only one of the geometries is correct on the IS level, and in other examples more than one geometry is a correct analysis on IS. Where and how should the not wanted geometries on ECS be filtered out in analysis?

The Danish sentence

"Kommissionen har konstateret en tilstr{kkelig
Litt: The Commission has stated a satisfactory

anvendelse af teknologi, der forbedrer industriens
application of technology which improve the industry's

situation"
situation

will in analysis, using the Danish standard analysis module, create 32 different geometries on ECS, of which only one (or perhaps two) are acceptable as geometry of the interface structure.

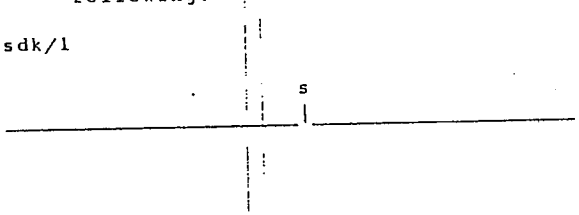
Take a more simple example: The Danish sentence

"Kommissionens konstatering n|dvendigg|r tilstr{kkelig
Litt: The Commission's statement necessitates satisfactory

udvikling af bistanden fra USA til Europa"
development of the assistance from USA to Europe

will give 14 objects on ecs when parsed with the standard Danish analysis module. The 14 different objects on ecs only differ in the attachment of the pps. The 14 different attachment structures are the following:

ecsdk/1



'of fear', CONDITION: 'in case of fire', ADVERSATIVE:
'against all odds', CAUSE: 'based on this idea',
CONSEQUENCE 'so that ...', COOPERATION 'with a friend'

VII. Structure disambiguation (pp attachment)

The problem of attachment can be stated as follows: any sentence of normal length will in analysis when using standard eurotra analysis modules create more than one attachment pattern, or as I would prefer to call it:geometry on ecs level. Of these different structures we only need one on IS level. Sometimes only one of the geometries is correct on the IS level, and in other examples more than one geometry is a correct analysis on IS. Where and how should the not wanted geometries on ECS be filtered out in analysis?

The Danish sentence

"Kommissionen har konstateret en tilstr{kkelig
Litt: The Commission has stated a satisfactory

anvendelse af teknologi, der forbedrer industriens
application of technology which improve the industry's

situation"
situation

will in analysis, using the Danish standard analysis module, create 32 different geometries on ECS, of which only one (or perhaps two) are acceptable as geometry of the interface structure.

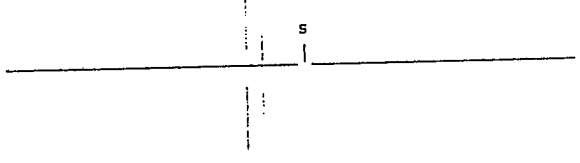
Take a more simple example: The Danish sentence

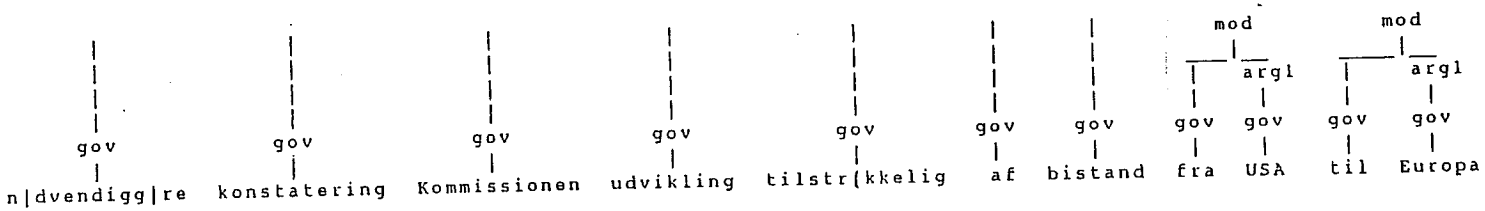
"Kommissionens konstatering n|dvendigg|r tilstr{kkelig
Litt: The Commission's statement necessitates satisfactory

udvikling af bistanden fra USA til Europa"
development of the assistance from USA to Europe

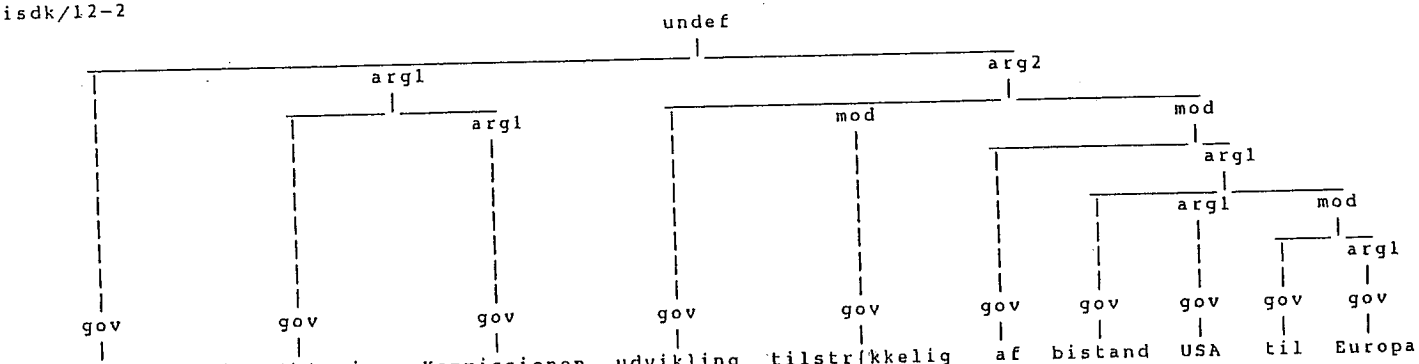
will give 14 objects on ecs when parsed with the standard Danish analysis module. The 14 different objects on ecs only differ in the attachment of the pps. The 14 different attachment structures are the following:

ecsdk/1

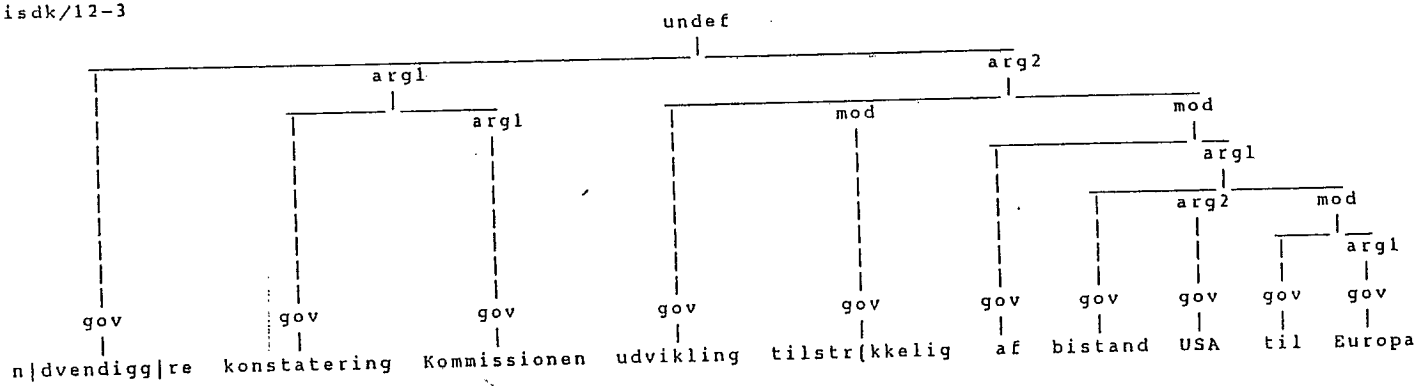




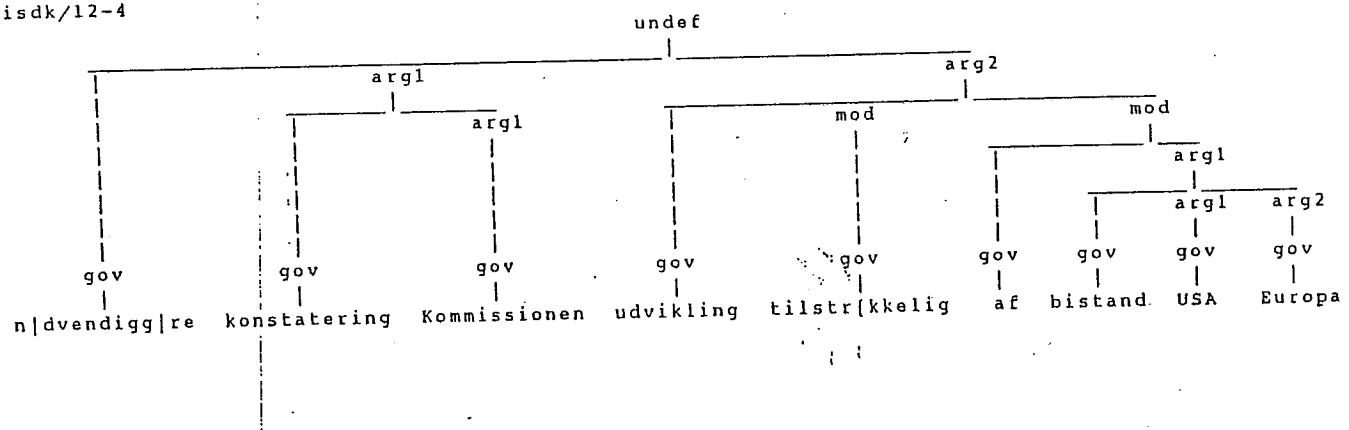
isdk/12-2



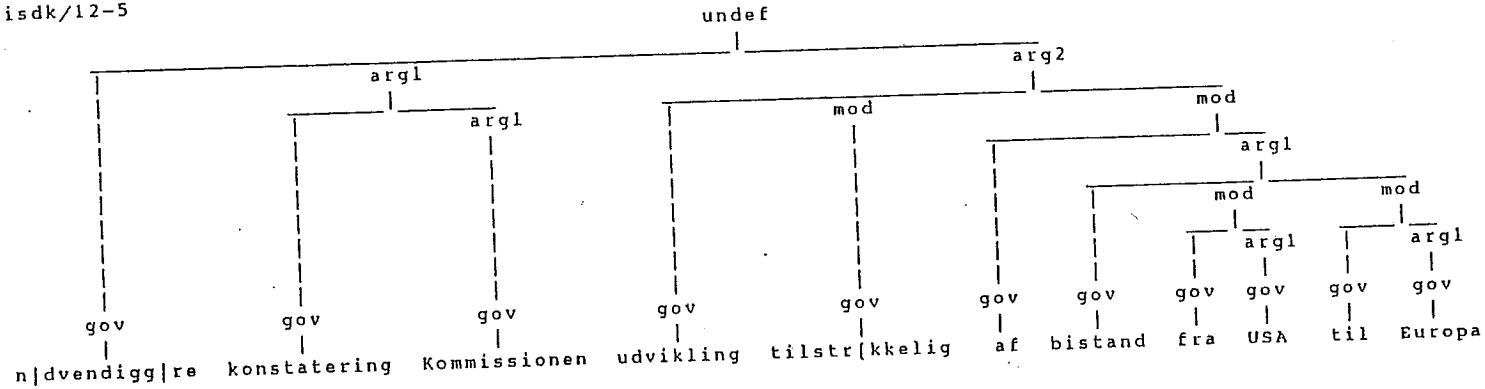
isdk/12-3



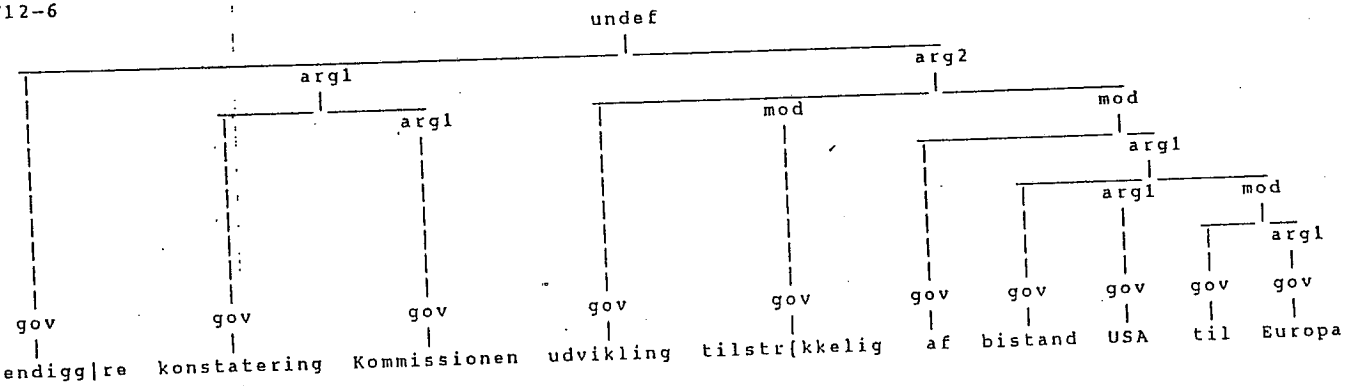
isdk/12-4



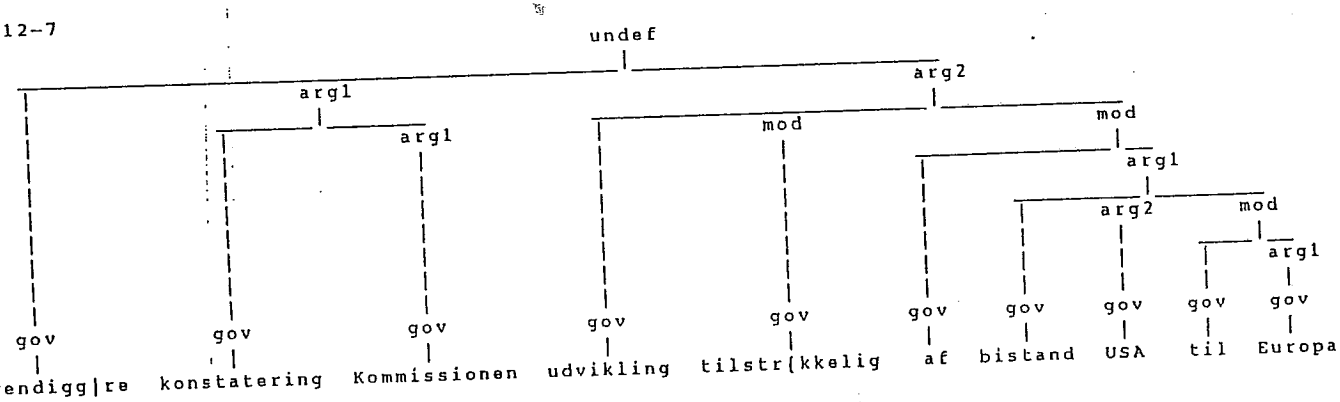
isdk/12-5



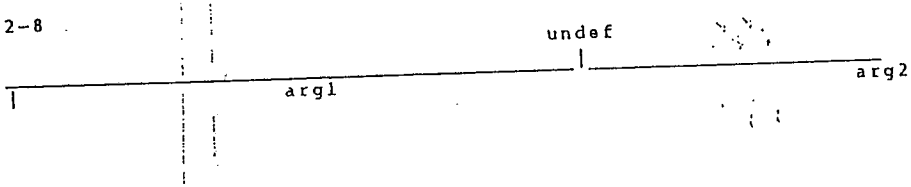
isdk/12-6

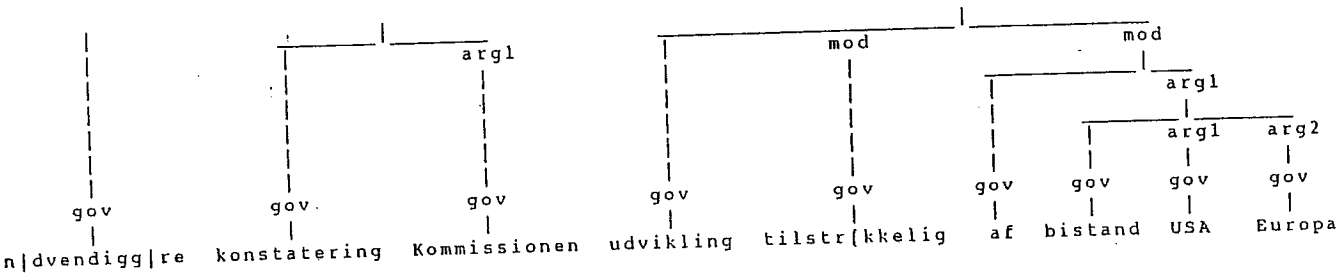


isdk/12-7

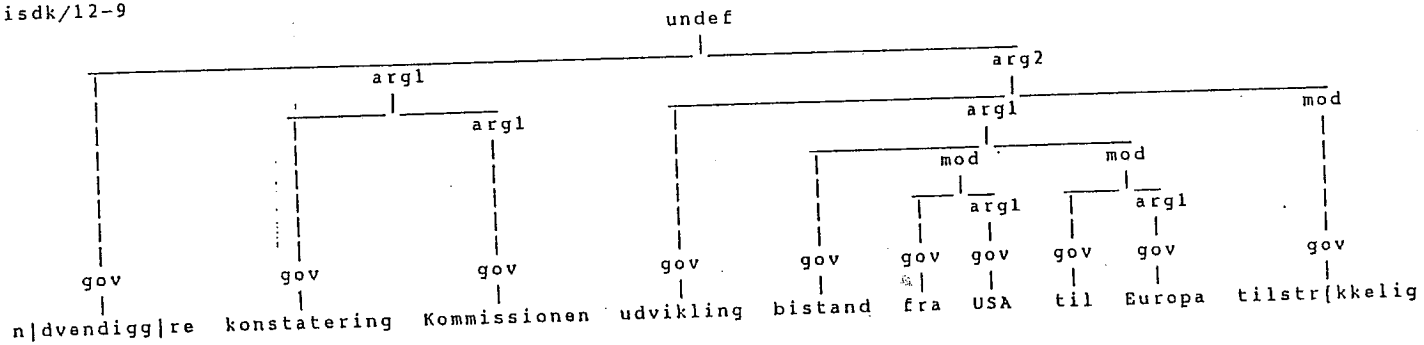


isdk/12-8

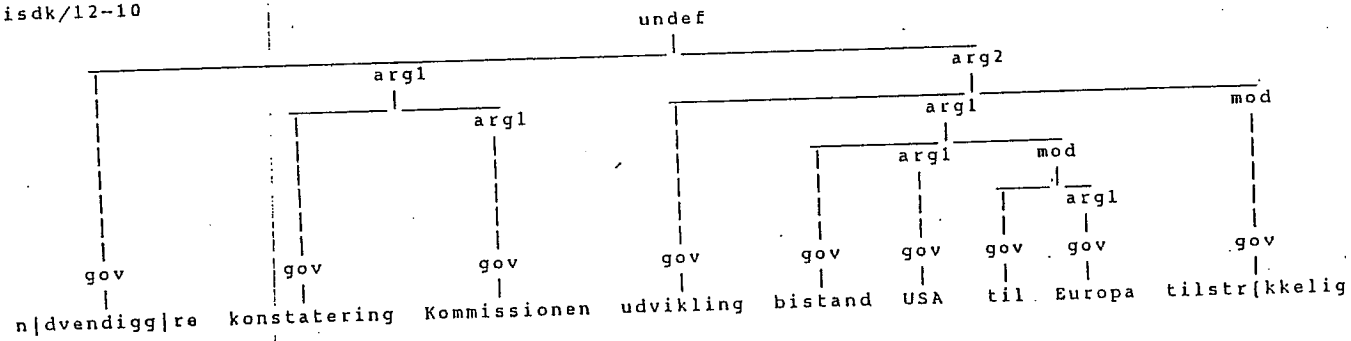




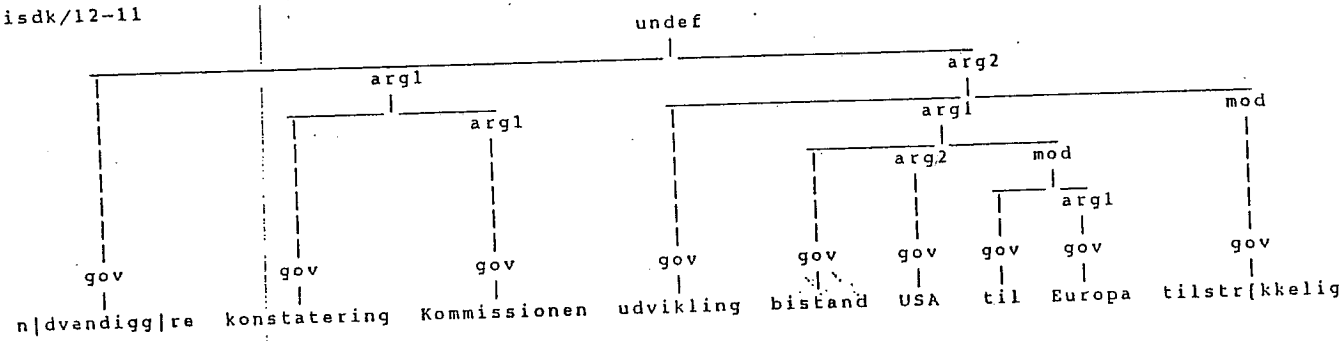
isdk/12-9



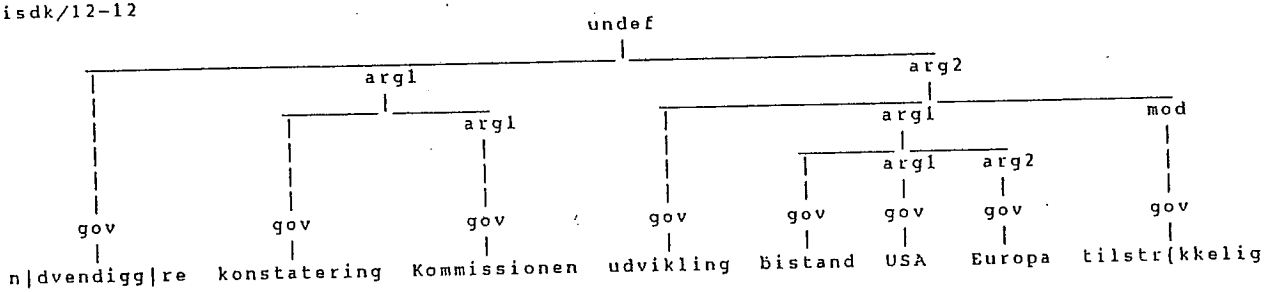
isdk/12-10



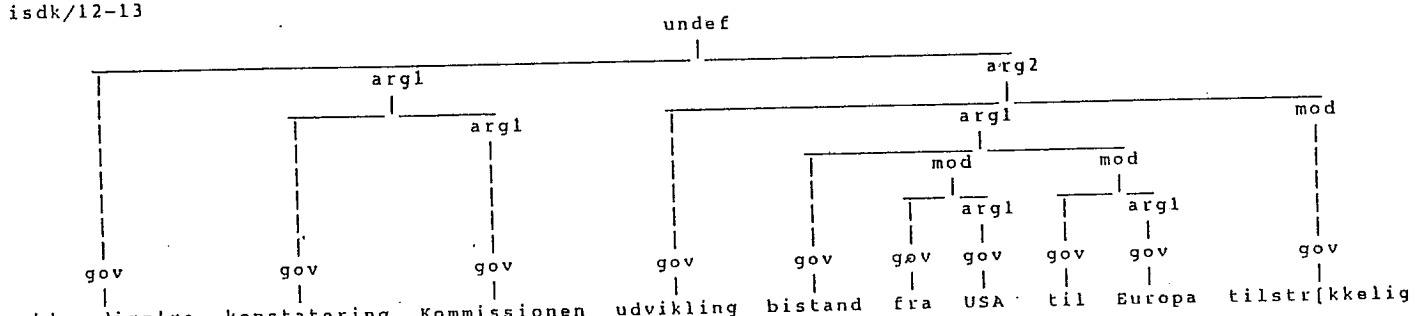
isdk/12-11



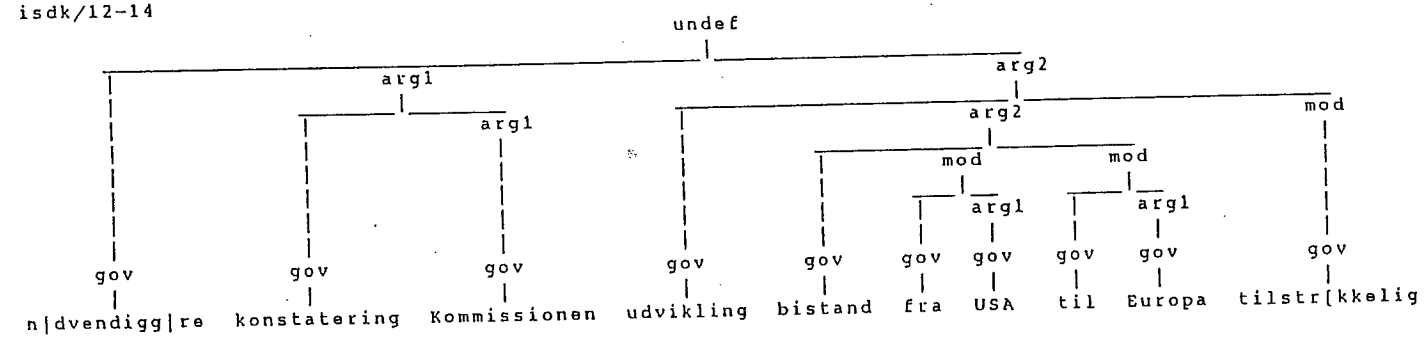
isdk/12-12



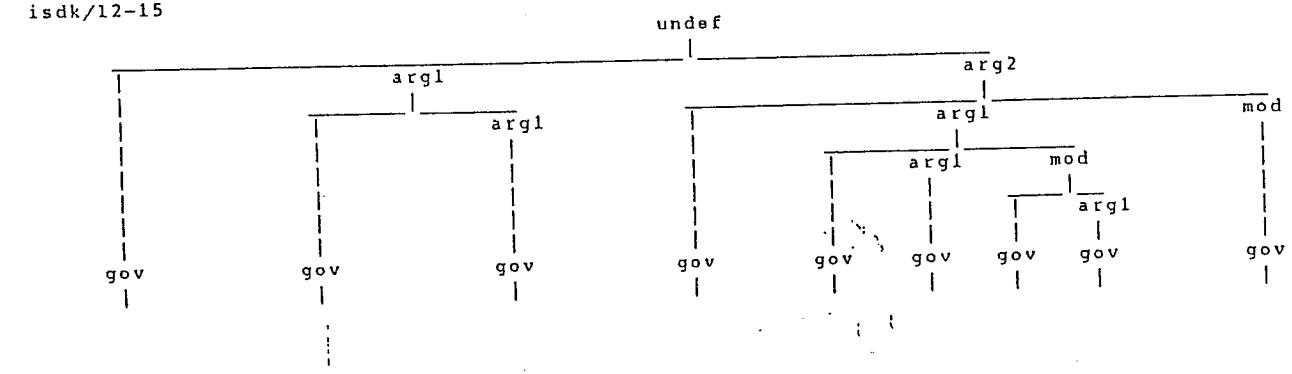
isdk/12-13



isdk/12-14

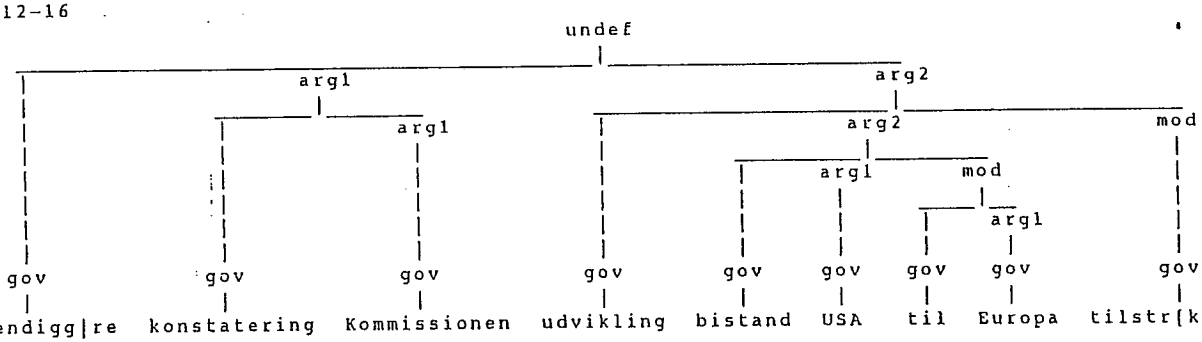


isdk/12-15

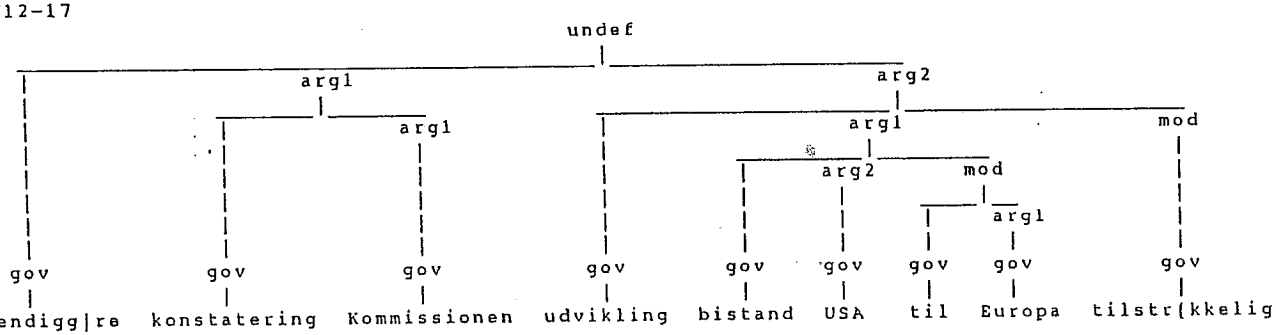


n|dvendigg|re konstatering Kommissionen udvikling bistand USA til Europa tilstr{kkelig

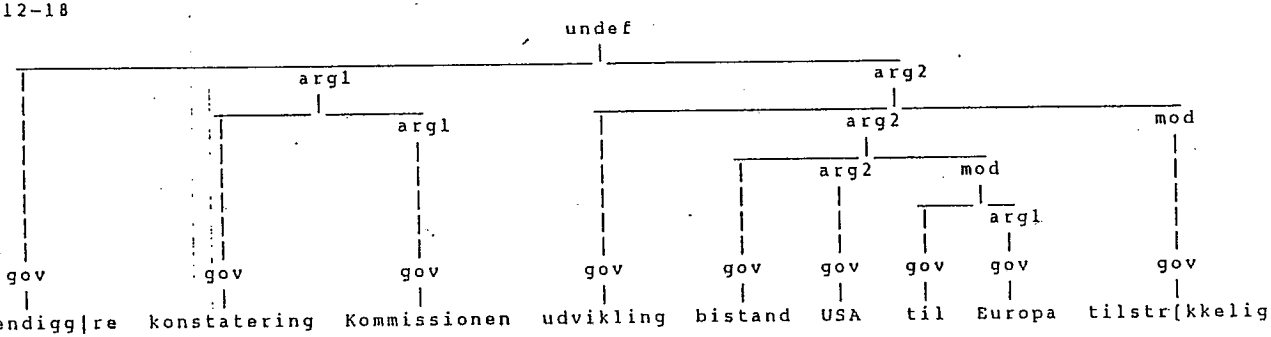
isdk/12-16



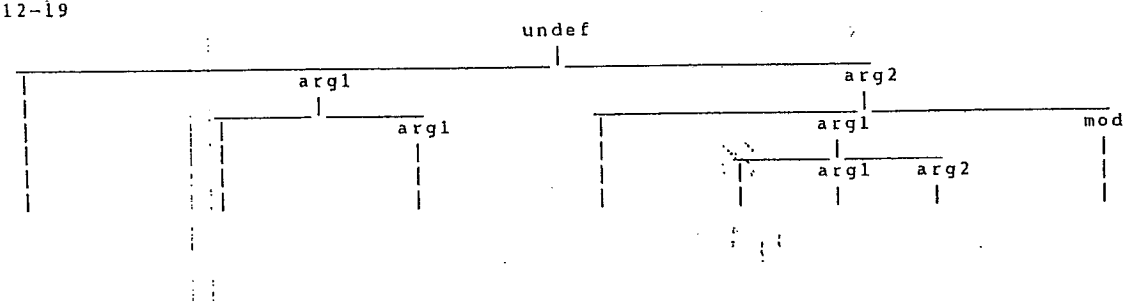
isdk/12-17

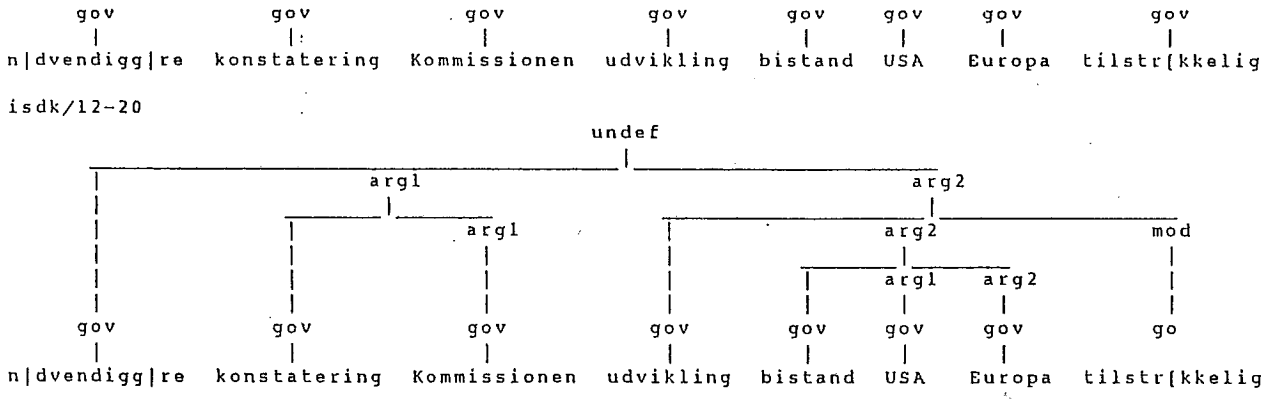


isdk/12-18



isdk/12-19





isdk/12-20

This is not the result of bad t-rules from ERS to IS; they are as they have to be. The main reasons for the diversity are the following:

The noun 'udvikling' has two frames: one with arg1, and one with arg1_arg2, and it is not possible in this sentence to decide which one is in use because only one candidate for argumenthood is present.

But if it in the frame of both 'udvikling_1' and 'udvikling_2' is described that arg_1 can never have prepositions, and for 'udvikling_2', that arg_2 can as the argprep have either 'af' or 'i', then all the readings where 'bistand fra USA til Europa' is arg_1 in relation to 'udvikling', can be excluded.

All the three pp's on ers/12 (the same geometry as ecs/12) can be both an argument and a modifier, and if the preposition matches with the frame for preparg, both arg1 and arg2. It can not in accordance with the existing IS-legislation be decided wether a certain pp is an argument or a modifier.

Theoretically it is not possible to make a rule saying: if a pp has a preposition equal to the argprep specified in the frame of the verb, it is an argument; if not, it is a modifier, because arguments, and especially arguments of nouns are optional, and if a proposition bound argument is not present it is possible to have a modifier with the same preposition.

```
 arg_1       arg_3
'han rejste til Paris'
 (he vent    to Paris)

 arg_1       modifier:TIME WHEN
'han rejste til jul'
 (he left    at xmas)
```

The linguistic rule which handle the attachment of pp's is not a constituent structure rule. The Danish ecs rules allow all the geometries that should be allowed and no more, and it is probably possible to find other acceptable examples which will fit all the 14 geometries.

Whether the problem of attachment is a question of relational structure depends on whether the syntactic valency of the verbs and nouns exclude certain attachments:

1. The verb 'noedvendiggoer' is in Danish a transitive verb which takes as a direct object either a noun or a that-clause, but it does not take any pp's as a complement. So all the structures without two np's should be exclude on ERS. There are no geometries from ECS that will create a structure with only one and zero nps. So in this sentence the linguistic rule is not a relational structure rule.

2. None of the nouns 'udvikling' and 'bistand' have obligatory complements, so in this sentence the linguistic rule is not a relational structure rule.

The linguistic rules that exclude the incorrect attachment structures has to be interface structure rules.

The govenors have the following lexical entries on IS.

```
'noedvendiggoere' = (gov, [lu='noedvendiggoere', syncat=v,
        frame=arg1_arg2, arg1=entity, arg2=situation, arg1prep=none,
         arg2prep=none, mod_type=when;duration_process;where;manner]).[],

'bistand_1' = (gov, [lu='bistand', syncat=n, scat=deverbal,
    semfeat=activity, frame=arg1_arg2_arg3_arg4, arg_1=hum,
    arg_2=hum,arg_3=proposition, arg_4=scale,
    arg1case=gen, arg1prep='fra', arg2prep='til', arg3prep='til',
    arg4prep=paa, mod_type=when;duration_noun;where]).[],

%%ex: Europas bistand til Danmark / til at opbygge
      sundhedsvaesenet

'udvikling_1' = (gov, [lu='udvikling', syncat=n,
    scat=deverbal, semfeat=activity,
    frame=arg1, arg1case=gen, arg1prep=none, mod_type=duration_noun;
    time_boundary;non_state;where]).[],

%% ex: barnets / sygdommens udvikling
(the development of the child, development in the decease).

'udvikling_2' = (gov, [lu='udvikling', syncat=n,
```

scat=deverbal, semfeat= accomplishment,
frame=arg1_arg2, arg_1=human, arg_2=entity,
arglcase=gen, arglprep=none, arg2prep=af;i, mod_type=duration_noun;
time_boundary;non_state;where}).[],

%% ex: industriens udvikling af nye metoder
(industry's development of new methods)

The atom rule for 'noedvendiggoere' will not exclude any of the 14 ERS geometries; the pps are not necessarily taken as arguments in any of them.

The linguistic problem is now: which rule makes the ERS geometry nr. 1., with all three pps as modifiers of the governing verb incorrect? In other words which rules make the following sentences unacceptable in Danish:

*"Af bistanden noedvendiggoer Kommissionens konstatering tilstraekkelig udvikling fra USA til Europa"

*"Fra USA noedvendiggoer Kommissionens konstatering tilstae{kkelig udvikling af bistanden til Europa"

*"Til Europa noedvendiggoer Kommissionens konstatering tilstraekkelig udvikling af bistanden fra USA"

Notice that these examples are unacceptable which means that it is not a question of preference rules but of strict a-rules. The incorrect geometries will in synthesis create non acceptable surface structures in Danish.

In this example the verb "noedvendiggoere" does not accept directional modifiers like "fra USA" (from USA) and "til Europa" (to Europe). I think it can be formulated as a general rule that no verb, noun or adjective accept a modifier which is 'to_and_from'. In other words: when we find a pp of this type it is always an argument. The rule could be fomulated in the following way: do not create IS objects with a modifier of the type: to_and_from. That is exactly what is done by the modifier selection mechanism bescribed in section V and VI, when this mod_type is not assigned to any verb frames in the lexicon.

The phrase "af bistanden" (of the assistance) is not a candidate for modifierhood, it can only be an argument to something, and probably only for noun like "udvikling"(development), "foroegelse" (increase).

But the result of the disambiguation process of the prepositions and nouns will in all cases yield some result; in this case probably a modifier of the type {mod_type=to_and_from} as the best (metaphorical)

reading among several possible readings of the preposition 'af' and the noun 'bistanden'. But the verb 'noedvendiggoere' does not accept a modifier of that type, so this tree structure will not be generated because the mod_type values of the verb frame and the pp, will not unify.

If these two rules will work, all the ERS (ECS) geometries except the wanted geometry nr. 12 will be excluded.Then it would be possible to parse the Danish sentence and only create IS objects from one object on ERS, and only from the correct one.

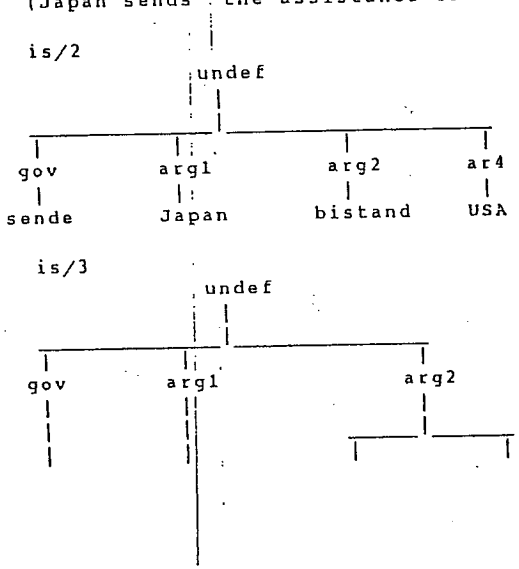
Then the same two rules will exclude 19 of the 20 IS objects created by the correct ersdk/12. All structures with 'af bistanden fra USA til Europa' is a modifier (1, 2, 3, 4, 5, 6, 7, 8) is excluded by the rule that the combination of 'af' + 'bistand' is not a possible modiefier.

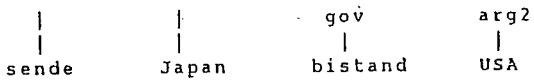
Of the rest all structures where 'fra USA' or 'til Europa' is a modifier will be killed by the rule that no verb accept a to_and_from modifier; this means that 9, 10, 11, 13, 14, 15, 16, 17, 18, are excluded.

At last the structures where'af bistanden fra USA til Europa' is arg_1 is excluded because no prepositions is accepted as arg_1 of 'udvikling'. The only structure that survive is 12-20.

In some cases two attachment patterns will both be acceptable on'is, e g.

Japan sender bistanden til USA
(Japan sends the assistance to USA)





In this case both geometries are acceptable on IS, but we do not want to compute both of them. Therefore it would be useful to introduce a preference rule so that if two objects both are created from the same surface sentence, (and none of them are killed by the killer rules, or selected by the preference rule), then the structure which is created first is chosen, as a sort of random choice.