Synonymy and Translation*

Franciska de Jong Lisette Appelo

Philips Research Laboratories
Eindhoven - The Netherlands

This paper is meant to give some insight into the interaction between on the one hand theoretical concepts in the field of formal semantics, and on the other hand linguistic research directed towards an application, more specifically, the research within the Rosetta Machine Translation Project. The central notion is **synonymy**. It will be used to discuss sameness of meaning for expressions belonging to **different languages**.

1 Introduction

Within the Rosetta-project we are facing the question whether and how translation can be performed automatically. There is no well-defined theory with respect to this process. Human translation does not provide enough interesting clues. Actually there is hardly more to observe than that two expressions are presented or accepted as translations of each other. Therefore, in our effort to let a machine perform the task of translating natural language, we will not aim at an empirical reconstruction of the human activity of translation, but at the construction of a formal system that defines the translation relation in accordance with human intuitions on acceptable translations.

A first move towards our goal is to answer the following question: what does it mean to claim that expression a is a translation of expression b? The translation relation should be based on considerations of meaning: two expressions are translations of each other if they have the same meaning. In the Rosetta- framework, the notion of meaning has a model-theoretic definition, which implies -more or less- that two expressions are considered translations of each other if they are true in the same set of models. It also implies that the pursued preservation of meaning is supposed to be independent of extra-linguistic knowledge: according to the definition of the translation relation above, Rosetta can be seen as a system that aims at a formal account for a special kind of synonymy, namely sameness of meaning for expressions belonging to different natural languages. As such the Rosetta-output amounts to the generation of equivalence-statements, which by nature are non-contingent. Their evaluation does not require any reference to extra-linguistic facts.

Knowledge of extra-linguistic facts is obviously a prerequisite for adequate translation, especially in view of the approach to ambiguity. Ambiguous sentences are supposed to have more than one meaning as they apply to more than one kind of situation. Consequently, they have more than one translation. Therefore, in case of ambiguous input, the part of the Rosetta system that makes use of linguistic knowledge only will define a set of possible translations.[1]

From the preceding remarks it follows that our current research activities consist mainly of the construction of linguistic modules, i.e. the definition of grammars and of the relation between grammars. In this paper the emphasis will be on the manner in which the the grammars for the languages involved are to be designed, in order to treat those languages as belonging to one and the same semantic system. In section 2 we will first sketch the requirements for synonymy of complex expressions in further detail, on the basis of the discussion in Carnap (1947) of a fairly trivial example of synonymy in the realm of formal languages. In section 3 a brief introduction to the Rosetta-framework will be given. This is followed by a more detailed discussion in section 4 on the role of the Isomorphy Principle in the preservation of meaning during the translation process. The complexities of the Rosetta grammars will be elucidated in section 5 by a discussion of some

non-trivial translations that Rosetta is supposed to deal with. One of them will be addressed again in section 6, where a brief sketch is given of some descriptive requirements for the presumed semantic theory. Finally, section 7 will focuss on the relevance of the Rosetta framework from a more general linguistic point of view.

2 The Notion of Synonymy

Consider the following three examples.

(1) 7 > 3
(2) Gr[VII, III]
(3) Gr[Sum(II, V), III]

The above statements can be seen as expressions with identical truth conditions. Each of them is true if and only if seven is greater than three. But in spite of the fact that they have corresponding parts that are equivalent, they are not three synonyms as a pairwise comparison will indicate.

The statements (2) and (3) are both true under the assumption that seven is greater than three. Hence, a transition of (2) into (3), or vice versa, will be meaning preserving in extensional contexts. The fact that the number of basic expressions differs for (2) and (3) causes a crucial difference: in intensional contexts they cannot be substituted freely.
The statements (1) and (2) do not only have identical meanings and equivalent corresponding parts, but they are also both built by means of the same number of basic expressions and operations. The differences in surface syntactic ordering and structuring devices, do not affect their meaning. In intensional contexts they can be substituted for each other freely. Paraphrasing Carnap: (3) has an intensional structure that is not isomorphic to that of (1) and (2).

> If two sentences are built in the same way out of designators [...] such that any two corresponding designators are L-equivalent, then we say that the two sentences are **intensionally isomorphic**. (o.c.:p.56)

This definition refers to Carnap's notion of L(ogical)-equivalence, which can be informally paraphrased as follows: Σ_i is **L-equivalent** to Σ_j if the truth of $\Sigma_i \equiv \Sigma_j$ can be established on the basis of semantical rules alone, without any reference to (extra-linguistic) facts. (o.c.:p.10)

Statement (3) contains an argument expression that is not isomorphic to the corresponding part in (1) and (2), therefore the pair (1) and (3) and the pair (2) and (3) fail to fulfil the isomorphy requirement implied by the above quotation. As only isomorphic expressions can be regarded as true synonyms, (3) is not a synonym of either (1) or (2).
From Carnaps identification of synonymy and intensional isomorphism it follows that there are two requirements for synonymy:

- equivalence
- isomorphism

In the arithmetical example above, the semantic systems referred to need not be reconstructed empirically. They are defined independently. The equivalence of the expressions involved is obvious. In the context of such examples it is almost trivial to decide on the presence or absence of synonymy. Carnap:

> We find that [these expressions] are isomorphic by **establishing** the L-equivalence of corresponding signs. (o.c.:p.58)

As a guideline for the development of computerized translation devices, this remark is by no means sufficient. First of all there is no uniformity concerning the grammatical structure of natural languages: there is no such well-defined principle that decides a priori what strings are to count as (basic or complex) expressions. Moreover, on the level of surface syntax there are numerous sources of mismatches, even in similar languages such as Dutch and English. Consequently it is not self-evident, and hence not easy to establish what is to count as an example of corresponding signs. So for our aims Carnap's heuristics of synonymy needs some revision. A formal account of synonymy between natural languages requires the **stipulation of synonymy**: expressions are isomorphic if we treat them as such. This may sound less informative than it is. In section 5 it will be argued that there is indeed a lot of stipulated synonymy needed in order to deal with larger fragments of natural language as is aimed for within Rosetta. In the sections preceding section 5 an introduction will be given to the Rosetta framework (section 3), and the way preservation of meaning is pursued in this framework (section 4).

3 The Rosetta Framework

The linguistic framework of Rosetta can be characterized by a number of 'working principles'. The Isomorphy Principle is only one of them. It will be discussed in detail in the next section. The current section is meant as a rough description of the Rosetta-framework. Rosetta will not be discussed here extensively, but just as much as is needed as a background for the understanding of the role of the Isomorphy Principle. In addition to a discussion of some of the Rosetta-principles, we will introduce here the levels of representation that will be referred to in the remainder of this paper.

3.1 Some Rosetta Principles

Note that the role of the principles to be discussed here, is to provide a guideline for systematic research on the possibilities for automatic translation and to be a support in the actual construction of the systems.

- **Principle of Explicit Grammars:** The translation relation is defined by means of explicit grammars for both source and target language. So the

wellformedness of input and output sentences results from independent sets of rules. The actual rules of the grammar are strongly influenced by the next principle.

- **Compositionality Principle:** The meaning of an expression is a function of the meaning of its parts and of the way in which they are combined. This principle is adopted from Montague Grammar, at least in spirit. We will regard compositionality as an obvious ingredient for a translational system based on a formal notion of meaning and assume that the given definition is explanatory enough. For further comment, cf. Landsbergen (1987).

- **One Grammar Principle:** A multilingual bidirectional machine translation system requires for every language an analysis component (for its function as source-language) and a generation component (for its function as target language). In Rosetta these two components are based on one and the same grammar of which the rules are reversible: they can be used both for generating and for analyzing sentences.

- **Isomorphy Principle:** Two sentences are considered translations of each other if their meanings are derived in the same way from the same basic meanings. In the following sections, the function of the Isomorphy Principle in a framework for automatic translation based upon the notion of meaning will be discussed in more detail.

3.2 Representations in Rosetta

The translation process is divided into an analysis phase and a generation phase. Both phases are defined by three components of the compositional grammars of the Rosetta-system (which are called M-grammars): a morphological component, a syntactic component and a semantic component. In order to elucidate how the principles sketched above interact, and also to facilitate the reading of the next sections, this section will be addressed to a brief introduction to some of the levels of representation employed in Rosetta. Attention will be restricted to the representations of the syntactic and the semantic component of the M-grammars. In the sequel of this paper we will refer to the following three levels, of which the first two are defined by the syntactic component, and the third level by the semantic component. Figure 2 at the end of this section may serve as a schematic outline of the organisation of Rosetta.

- Surface syntactic trees
- Syntactic derivation trees
- Semantic derivation trees

The **surface syntactic trees** (S-trees) are defined by the rules of the syntactic component, which are called M-rules. These M-rules yield sentential as well as constituent structures. The S-trees represent the syntactic structure of complex

expressions. The nodes of these trees are labelled with syntactic categories and attribute-value pairs. The branches are labelled with relations.
The process of deriving a surface tree starting from basic expressions by applying syntactic combination rules recursively, is represented in a **syntactic derivation tree** (synt. D-tree) with basic expressions as terminal elements, and the names of the applied rules at the non-terminal nodes. To each node of the derivation tree corresponds an intermediate S-tree. In Rosetta, the distinction between meaningful operations and purely syntactic operations is reflected in the distinction between **rules** (in the syntactic derivation trees represented as R'_n) and **transformations** (in the syntactic derivation trees represented as T_n). The left part of figure 1 specifies the syntactic derivation tree, as well as the intermediate S-trees for the sentence *Oscar is sleeping.*[2]

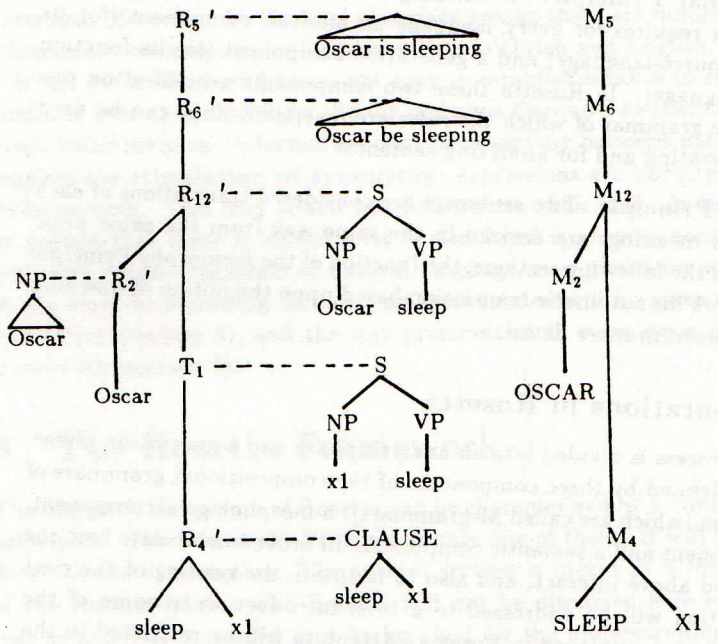


Figure 1: syntactic and semantic derivation for *Oscar is sleeping*

Surface strings of a certain language, which of course always exhibit language specific features, are mapped onto surface strings of another language via the representation of their common meaning. According to the Compositionality Principle the process that results in well-formed surface strings is in correspondence with the derivation of the meaning of the generated string. Therefore, the meaning of a complex expression can be represented in a **semantic derivation tree** (sem. D-tree): a tree with the same geometry as the syntactic derivation tree but labelled with the names of the meanings of the basic expressions as terminal elements, and the names of the meanings of the syntactic rules at the

non-terminal nodes. In the right part of figure 1 an example of a semantic derivation tree is given. As purely syntactic operations by definition have no impact on the semantics of the expressions involved, transformations are irrelevant for establishing synonymy. They are so to speak **translationally irrelevant**. For a detailed discussion of the formal distinction between rules and transformation see Appelo, Fellinger & Landsbergen (1987). For the role of semantic derivation trees as interlingua, see Appelo & Landsbergen (1986).

In order to facilitate that a common semantic derivation tree is assigned to two synonymous strings belonging to different languages and with different surface syntactic properties, they must be derived in a parallel way, but regardless of the part of the derivation that is marked as meaningless, namely the part defined by the transformations. M-rules may perform various syntactic operations at once, and also the descriptive content of the M-rules is language-specific to a certain amount. Therefore, a careful division of the syntactic content over the various steps in the derivations is required in order to allow the pursued mapping of synonymous strings.
Note that the syntactic derivations to be displayed below will be given in a reduced form. In general only the meaningful part of the derivational history will be represented, so syntactic transformations are left out. (As a consequence the syntactic D-trees will be of the same geometry as the corresponding semantic D-trees.) Also some minor meaningful rules will be ignored. For example, the rule that defines the NP-node dominating proper names, and sometimes the rules replacing argument variables for full NPs or vice versa, will be omitted in the sequel of this paper.

To summarize: the syntactic rules and the basic expressions define what the chunks of meaning are, and application of these rules specifies surface trees that express a.o. language specific syntactic generalisations. The delicacy of the relation between syntactic and semantic derivation is elucidated extensively in the next section. The current section will be concluded with a schematic representation of the Rosetta translation process (figure 2).

4 Isomorphy

According to the Isomorphy Principle, sentences that are to be regarded as translations of each other, must be *derived in the same way*, i.e. by fully parallel processes. Consequently, the reduced syntactic derivation trees will have the same geometry. In order to guarantee that the mapping relation is indeed defined for equivalent expressions, the various steps in the derivation must be designed carefully, in such a way that each basic expression can be mapped onto its (stipulated) equivalent, and that there is a proper correspondence for each of the rules. (Remember that a source language derivation tree is mapped onto its target language equivalent via their common meaning derivation tree, with which they are isomorphic.) This process of careful design is called the **attuning** of (rules of)

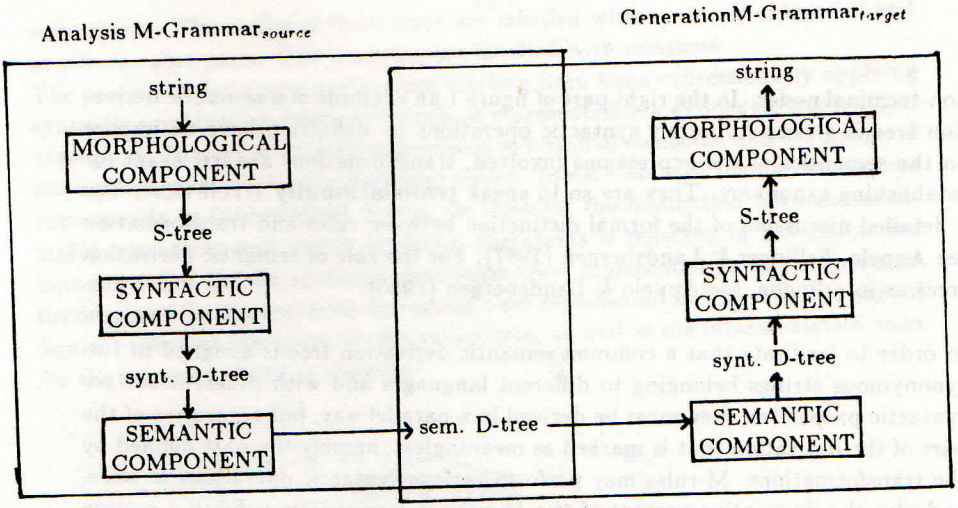
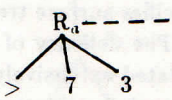


Figure 2: the Rosetta translation process

grammars.

Applied to the examples (1) and (2) of Carnap above, it must be guaranteed that if *Gr* is taken as a basic expression, > is a basic expression as well. Alternatively, a syncategorematic introduction of the one designator would require a syncategorematic introduction of the other as well. In general, synonymous expressions should have derivation trees with corresponding basic expressions as leaves, and representations of corresponding rules as nodes. The following syntactic derivation trees for (1) and (2) would obey the requirements imposed by the Rosetta Isomorphy Principle.



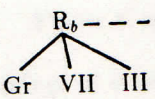


Figure 3: syntactic derivation trees for (1) and (2)

In Rosetta, information about the actual content of syntactic operations is not represented in the derivation trees, therefore derivation trees do not reflect differences as those between (1) and (2) directly. This is in accordance with Carnap's analysis:

> [...] the use of a functor preceding the two argument signs instead of one standing between them may be regarded as an inessential syntactical device.

(o.c.:p.56)

> For the intensional structure, in contrast to the merely syntactical structure, only the order of application is essential, not the order and manner of spelling. (o.c.:p.59)

Until now, we have not yet mentioned anything concerning the nature of the meaning representations in Rosetta. It was noticed before that the translation of synonymous expressions is effectuated via the mapping of their syntactic derivation trees, with a mapping onto their common semantic derivation tree as an intermediate step. In Rosetta, the basic meanings and meaning operations need not be made explicit. However, they are supposed to be compatible with the semantics as defined in Montague Grammar. In section 6 we will return to this issue in more detail.
The next section will be concerned with mappings that are far less trivial than the mapping of (1) and (2). The non-trivial nature of the process of attuning grammars will be demonstrated on the basis of several cases of mismatches between languages. Some of them even require the stipulation of synonymy of basic expressions while at first sight identity of meaning for these basic expressions does not exist.

5 Mismatches

Presuming the correctness of Carnap's claim that word or constituent ordering should be regarded as inessential from a semantic point of view, the mapping of synonymous expressions of different languages need not be complicated by mismatches due to differences in surface syntactic ordering. These can be accounted for by the language specific parts of the grammars, e.g. the transformations or the descriptive content of a meaningful rule. However, mismatches can be demonstrated at various other levels as well. To start with a relatively simple one, consider the following two equivalent sentences:

(4) Oscar slaapt (Dutch)
(5) Oscar is sleeping (English)

These two sentences have differently organized predicates: (5) contains an auxiliary to express the progressive tense, whereas in (4) only a present tense morpheme occurs. In order to let the grammars provide isomorphic derivations for these sentences it should be decided whether or not the English auxiliary *is* a basic expression. A basic expression is supposed to have a corresponding basic meaning, and in this case, the concept of progressive tense is intuitively connected with sentential features rather than with the verb *be* as a basic expression. Therefore it seems natural to treat *is* in (5) as a syncategorematically introduced expression that lacks an independently defined basic meaning. If the present tense morpheme in Dutch is also considered to be introduced by a rule, the two

sentences can be derived isomorphically. Figure 4 contains the simplified syntactic derivation trees for (4) and (5). They illustrate the strategy pursued here: isomorphic syntactic derivations are gained by assuming two basic expressions in the derivations for both the Dutch and the English sentence. The rules R_5 and R'_5 respectively, combine the two basic expressions to form a clausal structure, while R_6 and R'_6 account for tense. The crucial difference between the surface strings (4) and (5) is the result of a difference in the descriptive content of R_6 and R'_6.

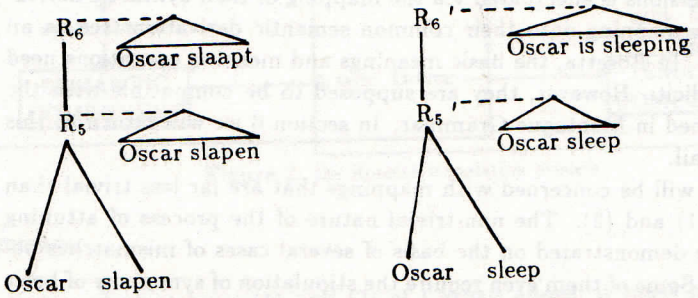


Figure 4: syntactic derivation trees for (4) and (5)

Other sources of mismatches are less trivial, because they require analyses and/or mappings that are counterintuitive in some sense. In this section we will discuss four examples. First two examples of mismatches of grammatical relations, one within the internal structure of NPs, and the other on the sentential level. The third example concerns the mapping of an adverb and a verb, and the fourth concerns the mapping of a one-word string onto a two-word string.

1. Genitive *-s* versus postnominal modification. Consider the following two equivalents (with the literal meaning: the book of Conchita).

(6) Conchita's boek (Dutch)
(7) el libro de Conchita (Spanish)

In (6) the possessive modification is expressed by a prenominal genitive NP. In (7) the prenominal structure contains a definite determiner, while the possessive modification is expressed by a postnominal PP. Stated otherwise, (6) and (7) illustrate that in Dutch and Spanish the possessor role is expressed by means of different syntactic relations, viz. a determiner relation in Dutch, versus a modifier or complement relation in Spanish. In Figure 5, isomorphic derivations for (6) and (7) are given.[3] These derivations presume several decisions.

- The introduction of modification and the introduction of definiteness is realized in two steps. A treatment along this line is motivated by the fact

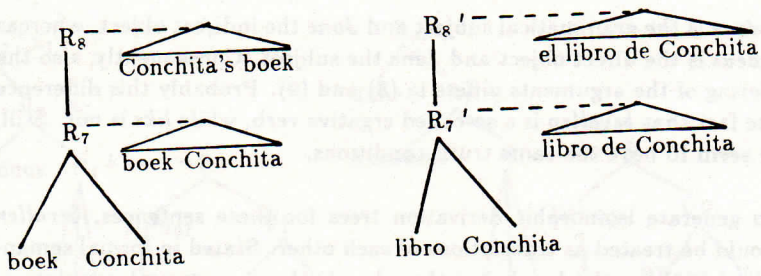


Figure 5: syntactic derivation trees for (6) and (7)

that in Spanish the string *libro de Conchita* must be available for the indefinite NP *un libro de Conchita* (a book of Conchita) as well. On the basis of the syntactic peculiarities of (6) alone there would be no need to assume a derivation in two steps. Note however that even from a monolingual point of view an analysis in two steps for (6) might be preferred in view of the fact that definiteness and possessive modification are separate semantic phenomena.

- The definite article *el* is introduced syncategorematically, rather than being treated as a basic expression. Obviously, this is not motivated by the analysis of Spanish NPs in general. However, if *el* were to be analysed as a basic expression, the derivation of (6) would require the deletion of the Dutch definite article. Under either solution, the analysis of either (6) or (7) is counter-intuitive and would probably not be chosen if Dutch and Spanish were to be analysed by a grammar of a monolingual system. In this example, it is an arbitrary question which of the two alternatives is to be preferred.

- A similar argument holds for the third decision implied by Figure 5: the preposition *de* in (7) is introduced syncategorematically, since its translation equivalent *van* does not show up in (6), but an analysis that presumes a *van*-deletion rule for Dutch might be preferred as well.

The synonymy of (6) and (7) indicates that grammatical relations (or categories) such as determiner and modifier, are in itself translationally irrelevant. Moreover, (6) and (7) illustrate the claim that the mapping of derivation trees is not always self-evident, but rather the result of a careful process of attuning, which might involve counter-intuitive decisions with respect to what is to count as a basic expression. The example involved in the next section is even more complicated, as it involves the mapping of basic expressions that intuitively are no synonyms at all.

2. Switching of arguments: *bevallen* versus *like*. Consider the sentences (8) and (9) that intuitively should be considered as acceptable translations of each other: both express the fact that the film Amadeus appeals to Jane.

(8) Amadeus bevalt Jane (Dutch)
(9) Jane likes Amadeus (English)

In (8) *Amadeus* is the grammatical subject and *Jane* the indirect object, whereas in (9) *Amadeus* is the direct object and *Jane* the subject. Consequently, also the surface ordering of the arguments differs in (8) and (9). Probably this difference is due to the fact that *bevallen* is a so-called ergative verb, while *like* is not. Still, (8) and (9) seem to have the same truth conditions.

In order to generate isomorphic derivation trees for these sentences, *bevallen* and *like* should be treated as translations of each other. Stated in formal semantic terms: it should be stipulated that they denote the same two-place relation, here represented as the semantic object LIKE. The argument of this 2-place relation LIKE is a pair, here consisting of the denotata of *Jane* and *Amadeus*. That is, the meaning of (8) and (9) should be a logical expression along the lines of (10). (Note that the choice of the ordering of the arguments is in fact arbitrary, although there must be some choice.)

(10) LIKE(JANE, AMADEUS)

In order to derive (10) as the result of a compositional process, we assume the semantic derivation that is represented in Figure 6.

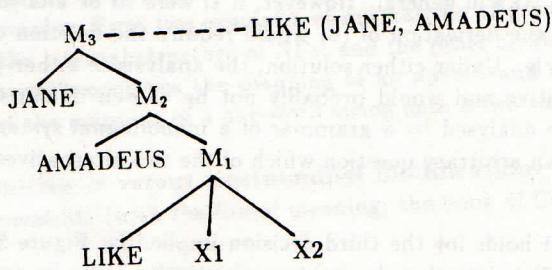


Figure 6: semantic derivation tree for (8) and (9)

Note that any syntactic derivation that would differ from the derivation in Figure 6 with respect to the number of steps involved in the formation of the propositional level from the basic expressions corresponding to LIKE, JANE and AMADEUS, will fail to provide a basis for isomorphic syntactic derivation trees.

Figure 7 exhibits (reduced) syntactic derivation trees for (8) and (9) that are isomorphic to the semantic derivation tree in Figure 6: the language specific syntactic rules that correspond to the common semantic rule M_1, i.e. R_1 and R'_1, specify the proper syntactic configurations for the occurrence of *like* and *bevallen*, respectively.

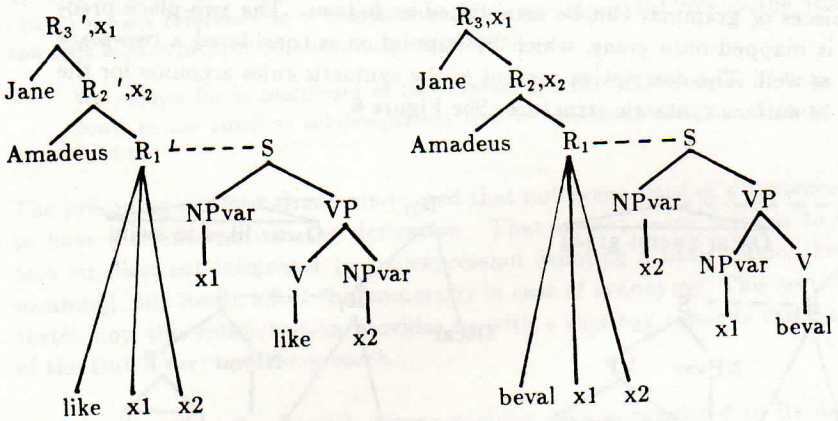


Figure 7: part of the syntactic derivation trees for (8) and (9) and derived S-trees

Two aspects of this analysis are prominent:

- in order to treat (8) and (9) as synonyms, the bi-lingual transfer dictionary of Rosetta must allow a translation of *bevallen* into *like* and vice versa. That is, it should allow the mapping of basic expressions that intuitively are no synonyms at all, but only by stipulation.

- According to the above treatment of (8) and (9), the syntactic bifurcation subject-NP versus VP is semantically empty, and given the semantic basis of Rosetta, translationally irrelevant. Within the analysis of Dutch alone it could be motivated that first the VP x_1 *beval* (x_1 to be substituted by *Jane*)is derived and then the sentence x_2 x_1 *beval* (x_2 to be substituted by *Amadeus*). In English the same could be said for the VP-constituent *like* x_2 and the sentence x_1 *like* x_2. But the required isomorphy excludes derivations that at any stage combine *like* and x_2, or *bevallen* and x_1, respectively. Again this illustrates that for a certain part the Rosetta-grammars are not founded in the syntactic and/or semantic analysis of a particular language. Instead the adopted analyses amount to the construction (or reconstruction) of synonymy.

3. The mapping of adverbials on verbs. A classic translation problem concerns the translation of the English sentence (11) into its Dutch equivalent (12).

(11) Oscar likes swimming
(12) Oscar zwemt graag

The problem consists in the fact that *like* should be considered a translational synonym for *graag*, while the categories for these two basic expressions are different: *like* is a verb and *graag* is an adverb. Due to this category mismatch, the sentences have to be assigned different surface structures as well: the surface

structure of (11) contains a sentential complement, while (12) is a simple sentence without an embedded sentence. But as will be clear by now, the Isomorphy approach offers an adequate tool for the mapping of syntactic derivation as well as the mapping of the meaning of such different structures. The isomorphy of the relevant pieces of grammar can be established as follows. The two-place predicate *like* is mapped onto *graag*, which by stipulation is considered a two-place predicate as well. The descriptive content of the syntactic rules accounts for the difference in surface syntactic structure. See Figure 8.

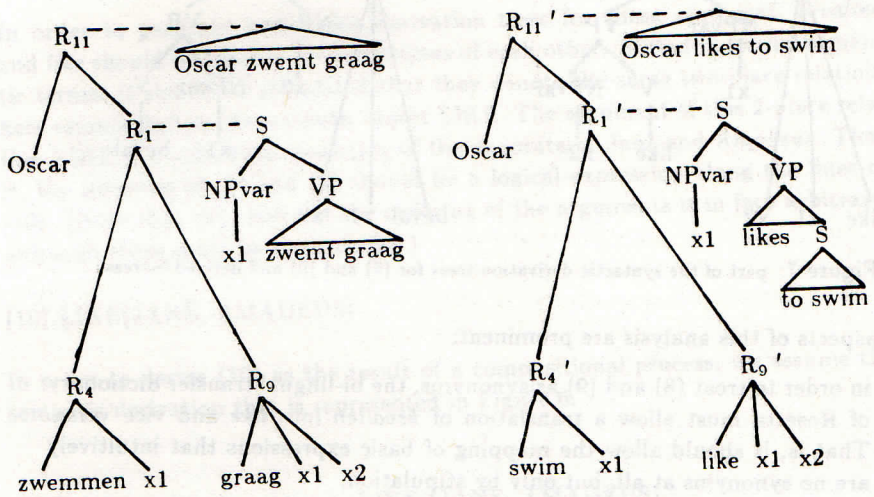


Figure 8: syntactic derivation trees for (11) and (12)

The urge to account for the synonymy of (11) and (12) enforces an analysis for certain adverbs as a two-place predicate. Note, however, that the relational nature of for example *graag* can be argued for on monolingual grounds as well. The occurrence of *graag* requires the presence of an animate subject: **het regent graag* (cf. '*it likes to rain'). The strong relation between *graag* and the sentential subject is also indicated by the impossibility to passivize a sentence with *graag* salva veritate.

Just as the preceding examples of mismatches, the synonymy of (11) and (12) illustrates the restricted relevance of surface syntactic notions from the translational point of view. The Rosetta framework offers an adequate strategy towards this phenomenon, because the preservation of meaning is pursued via the derivation trees, rather than via a surface analysis. In order to elucidate the advantages of a stategy for machine translation that accounts for identity of meaning by a reconstruction or construction (in case of mismatches) of the corresponding isomorphism, this section will be concluded by discussing some examples of mismatches on the level of basic expressions.

4. The mapping of *zeer* onto *very much*. The following quotation from Carnap (1947) makes reference to a notion that plays an important role in the strategy towards isomorphism: smallest subdesignator.

> We require for isomorphism of two expressions that the analysis of both down to the smallest subdesignators lead to analogous results. (Carnap 1947:57)

The preceding sections already indicated that not every word of a sentence need to have a counterpart in the derivation. That is, not every word is to count as a smallest subdesignator (= an expression denoting a basic model-theoretic meaning), and hence affect the isomorphy in case of synonymy. This section will sketch how this relativization provides us with a strategy towards the mapping of the Dutch *zeer* onto *very much*.

The occurrence of the English degree-modifier *very* is restricted to its use as a modifier to adjectives and adverbs. It never occurs as a sentential constituent. Its Dutch counterpart *zeer* it not restricted in its distribution. As shown in (13) it can modify verbs as well. As a translation of (13) we need (14b) rather than (14a).

(13) Amadeus bevalt Jane zeer
(14) a. *Jane likes Amadeus very
b. Jane likes Amadeus very much

According to the presumptions described thusfar, the Rosetta grammars should provide isomorphic derivations for (13) and (14b). The problem we are facing here concerns the question what is to count here, in Carnap's terms, as the smallest subdesignator corresponding to *zeer*: (1) *very* and *much*, (2) *very*, or (3) *very much*. The first alternative would require a derivation for the Dutch *zeer* with two basic expressions instead of one as well: next to *zeer* as corresponding to *very*, a basic expression corresponding to *much* should be distinguished. In order to derive the correct surface string a (meaningless) rule would be needed to delete this counterpart of *much*. The second alternative with *zeer* and *very* as corresponding basic expressions requires the syncategorematic introduction of *much*. A third alternative would be to treat *very much* as a complex basic expression, synonymous with a simple basic expression in Dutch.

Each of these alternatives appeals to devices that are available in the Rosetta framework on independent grounds. The syncategorematic introduction of words, for example, was already demonstrated in the preceding sections. Deletion rules are needed for, among other things, the deletion of the subject arguments of infinitival complements. The incorporation of complex expressions is independently motivated by the treatment of idioms.
In a compositional framework, idioms need a special treatment, because their meaning cannot be composed out of the meaning of their syntactic parts. This is due to the fact that the parts do not have a meaning. For example, in the idiomatic reading of *kick the bucket*, the noun *bucket* does not refer to a bucket

at all. The meaningful part of the VP in a sentence such as *John kicks the bucket* is *kick the bucket*. Accordingly, this string is to be considered a basic expression, or in Carnap's words, a smallest subdesignator. The notion of complex basic expression is thus independently motivated for, irrespective of translational purposes. For a detailed discussion of the treatment of idioms in Rosetta, see Schenk (1986).
Applying this notion to *very much* would of course be a slightly different matter. From the perspective of the analysis of English alone, there is no need to consider *very much* a complex basic expression. Only the synonymy of *zeer* and *very much* induces such an approach. These special instances of complex basic expressions are therefore marked as **translational idioms**, i.e. expressions of which the idiomatic nature is motivated by translational purposes, rather than by monolingual analysis. Other examples of translational idioms are: *to rise early* which is considered a complex basic expression in view of its Spanish one-word counterpart *madrugar*, the Dutch *blijven staan* in view of *to stop*, and the Dutch and English *niet weten* and *not know*, respectively, in view of the Spanish *ignorar*. In addition to the derivation that is to be expected on the basis of monolingual analysis, these complex strings get an alternative treatment. The alternative analysis is obtained by extending the Rosetta dictionary entries with the complex basic expressions mentioned above.

6 VP-less Semantics

In the preceding section it was argued that for an adequate account of the synonymy of (8) and (9), it is necessary to do away with the VP-level in derivation.[4] This approach is characterized by the following features:

- The classical distinction between subject and predicate is present only in surface trees, if it plays any role at all. It is not the basis for semantic interpretation.
- No one-to-one correspondence between semantic roles and syntactic relations can be established in a generalized way. The first argument of a verb is not a priori the subject, nor are objects excluded from such an interpretation.
- As a consequence of the treatment described above it follows that there is no semantic level corresponding to the syntactic VP. This influences the treatment of what is usually dealt with as VP-modification. Moreover it complicates the account of some scope-phenomena.

In this section, we will address the third issue of VP-less semantics in some more detail.

The classical PTQ semantics, which is supposed to supply a basis for the contents of the semantic rules of Rosetta, would have to be extended in order to supply a suitable semantic rule corresponding to the sentence-formation rules R_1 and R'_1

as displayed in Figure 7, and to deal with the interpretation of non-sentential modifiers, as for example the so-called 'intensifiers'.
Traditionally, semantic frameworks such as Montague's PTQ treat non-sentential modifiers as VP-modifiers. Taken in isolation it is certainly possible to treat *zeer* in 'Amadeus bevalt Jane zeer' (13) and *very much* in 'Jane likes Amadeus very much' (14b) as VP-modifiers. However, as their respective VPs are not composed of synonymous basic expressions, we have to account for the intuitive synonymy of these sentences by enforcing the grammars to define isomorphic derivations for them as sketched above, and, moreover, to specify some suitable level other than VP, to function as the argument of these modifiers.
According to the derivations of (8) and (9) as represented in Figure 7 two other levels are available: the sentential level and the level with the verbal head of the construction as a terminal basic expression. As the former is not the most appropriate level for the expression of non-sentential modification, a solution can be sought in the incorporation of modification rules that apply to the verbal head. As the presumed intuitive synonymy of (8) and (9) already enforced the stipulated synonymy of *like* and *bevallen*, modification of the bare verb allows us to preserve the pursued isomorphy. See Figure 9 for a derivation tree along these lines.

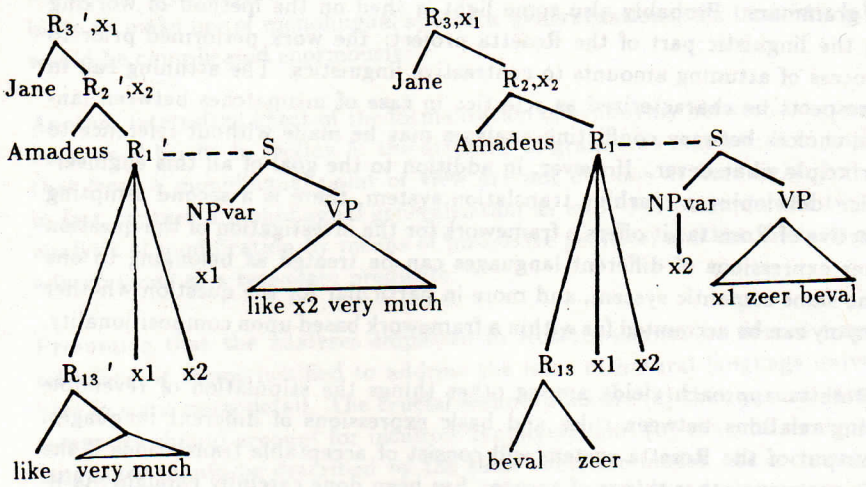


Figure 9: part of the syntactic derivation trees for (13) and (14a) and derived S-trees

In general, modifying expressions are assigned a semantic type of the scheme $< ty, ty >$: when a modifier is applied to an expression of type ty, the result is again an expression of type ty. Consequently, if verbs are of different semantic types, varying in their number of arguments, their modifiers should belong to different types as well. As one-place predicates and VPs are considered $< e, t >$-type expressions in classical semantic frameworks, the semantic type assigned to intensifiers is: $<< e, t >, < e, t >>$.

In order to account for the modification of verbs denoting a two-place predicate, for example *like* and *bevallen*, intensifiers should have an additional type-assignment based upon the type of two-place predicates. Transitive verbs are of type $<< e, < e, t >>$. Accordingly, the additional semantic type required for intensifiers is: $<< e, < e, t >>, < e, < e, t >>$.

Of course, introduction of this new type raises the question how to account for the systematic relation between the two readings for intensifiers. In general there are two approaches conceivable: either we assume that modifiers such as *very much* and *zeer* are ambiguous between the two type-assignments discussed here, or we make use of the so-called type-shifting devices as described for example in Geach (1972), and recently discussed in several other publications. In the former option we do not account for the systematic relation between the two readings. In the latter option we will have to make use of the principle of combinatory logic known as the Geach-rule, i.e. the rule which states that expressions of category x/y may be analysed as expressions of category (x/z)/(y/z).[5]

7 Discussion

In the preceding sections we have shown how judgements concerning the intuitive synonymy of expressions of different languages may direct and influence the design of the Rosetta grammars. What we deliberately tried to emphasize was the amount of engineering involved in the process of what we called the attuning of grammars. Probably also some light is shed on the method of working within the linguistic part of the Rosetta project: the work performed prior to the process of attuning amounts to contrastive linguistics. The attuning can in some respects be characterized as eclectic: in case of mismatches between languages, choices between conflicting analyses may be made without reference to any principle whatsoever. However, in addition to the goal of all this engineering, viz. developing a machine translation system, there is a second tempting perspective of Rosetta: it offers a framework for the investigation of the question whether expressions of different languages can be treated as belonging to one and the same semantic system, and more in particular for the question whether synomymy can be accounted for within a framework based upon compositionality.

The Rosetta approach yields among other things the stipulation of reversible mapping relations between rules and basic expressions of different languages. The output of the Rosetta system will consist of acceptable translations if the attuning -among other things of course-, has been done carefully enough. As illustrated by the examples of non-trivial attuning in section 5, it may be necessary to adopt analyses that are not motivated by the more limited goals of explanatory monolingual description. This final section is meant to provide some insight into the amount of linguistics in Rosetta and consequently, into the relevance of the presented analyses for the study of natural language semantics and universals.

First of all it should be stressed that the Rosetta framework distinguishes various levels of analysis. As indicated in section 3.2 there are two kinds of representations relevant for sentential analysis: the surface trees and the derivation trees. For the compositional mapping of expressions onto their semantic interpretation, and subsequently onto their target language counterpart, the derivation tree is the crucial level. In the actual elaboration of M-rules, the structure of the intermediate results as represented in the surface trees is important in several respects. Its role concerns both the translational system sec, and the method of labour during the design of the system. For example, the surface trees represent not only the result of the meaningful steps in the derivation, but also the translationally irrelevant features of a language that the derivation trees abstract away from. Consequently, the S-trees may exhibit surface syntactic configurations that are independently motivated for in a certain language. As such they are of interest for the study of natural language syntax. The language-dependent nature of the S-tree level has a practical advantage too. For the method of design it is crucial that there is a separate level for the representation of monolingual generalizations: in order to preserve control over the generated structures, the linguists who are the authors of the M-rules need the possibility to refer to the language-specific syntactic generalizations. For example, Dutch is more easily described if it is considered an SOV-language rather than an SVO-language. Therefore, in the Dutch intermediate S-trees the underlying order is SOV. Without the possibility to make use of monolingual syntactic generalizations, the linguistic labour would be complicated enormously.

Another interesting effect of the formalisation of synonymy between languages is mentioned briefly in section 5: the attuning of grammars may result in analyses that from a monolingual point of view are not obvious at first sight, but that in fact capture a monolingual generalisation as well. For example, the two-step analysis of modification by means of possessive genitive, and the analysis of the adverb *graag* as a two-place predicate.

Presuming that the analyses employed in Rosetta cannot be denied linguistic relevance, it seems justified to address the issue of natural language universals here in some more detail. The crucial assumptions are (a) that natural language semantics should account for intuitive synonymy, and (b) that different natural languages should be described by the same semantic model. As a consequence the intuitive synonymy of expressions belonging to different languages should be accounted for by assigning them identical meaning representations. Given the framework described above, this requires parallel derivational histories for synonymous expressions.
As argued above, careful attuning of grammars may require arbitrary choices between an analysis based on language A instead of language B. As Rosetta is designed for a very small set of languages, nothing concerning the universality of a certain analysis can be concluded on the basis of a single derivation tree. If we conceive of a future Rosetta system which deals with a less restricted set of

languages, the choices to be made may become less arbitrary due to the fact that there will be more facts to reckon with.
For example, if *like* and *bevallen* should be analysed as synonymous predicates, then one suggestion concerning universality is already implied by the preceding sections: the notion of VP is a surface syntactic notion, irrelevant from a semantic point of view, and consequently not belonging to the set of shared universal linguistic categories. Support for this conclusion is provided by an analysis of Modern Irish: as argued in McCloskey (1979), Modern Irish could be considered a VSO-language, a language with an underlying stucture that lacks a syntactic VP.

A more general conclusion to be drawn on the basis of the particular way in which Rosetta treats synonymy between languages is that the establishment of the proper balance between syntactic and semantic analysis may require the distinction of more levels of representation. In addition to the representation of surface syntactic generalizations we need a level representing the derivational history. The correspondence between syntax and semantics should not be sought in the establishment of a correspondence between surface syntactic structures and semantic representations. Surface syntax need not be complicated by considerations of semantic nature. In this respect the critics of Montague (1974) were right. If we distinguish between syntactic structure and its derivation, each motivated by different facts, the relation between surface syntax and semantic representation might turn out to be even weaker than is often argued. Presuming that Montague did not intend to account for the syntactic well-formedness of the expressions involved in his analysis, the account of synonymy described here might even be considered as supporting Montague with respect to his totally ignoring surface syntax.

Notes

* We would like to thank Carel Fellinger, Jan Odijk and especially Jan Landsbergen for their comments on earlier versions of this paper.

1. Actually this restriction only holds for the system presently under design, i.e. Rosetta3. In the follow-up of this system, Rosetta4, knowledge of some specialized domain will be incorporated in order to select the right translation. In addition it will deal with remaining ambiguities by means of interaction with its users.

2. Note that all derivation trees in this paper, including the one in Figure 1 are in fact simplified versions of the kind of derivation trees actually employed in Rosetta. Also the S-trees are simplified: the labels on the branches are omitted, and sometimes parts of the structure are abbreviated by triangles.

3. Actually, in Dutch both configurations are available. In addition to *Conchita's book* there is the equivalent form *het boek van Conchita*. They are true synonyms. Hence isomorphic derivations for (6) and (7) might as

well provide the basis for a formal account of the synonymy of the Dutch equivalents.

4. In fact this claim is tentative as long as the presumed synonymy of *like* and *bevallen* is not checked in all relevant contexts. Especially scope phenomena may turn out to be complicating the analysis.

5. This amounts to the claim that an expression that applies to a y to result in an x, can also be analysed as an expression that applies to an expression of category y/z to result in an expression of category x/z.

References

Appelo, L., C. Fellinger and J. Landsbergen (1987), 'Subgrammars, Rule Classes and Control in the Rosetta Translation System'. Philips Research M.S. 14.131. To appear in: *Proceedings of 3rd Conference ACL*, Copenhagen.

Appelo, L. and J. Landsbergen (1986), 'The Machine Translation Project Rosetta'. Philips Research M.S. 13.801. Also in: *Proceedings of the First Conference on the State of the Art in Machine Translation*, Saarbruecken.

Carnap, R. (1947), *Meaning and Necessity: A study in Semantics and Modal Logic*. The University of Chicago Press, Chicago.

Geach, P. (1972), 'A Program for Syntax'. In: D. Davidson and G. Harman (eds.). *Semantics of Natural Languages*. Reidel, Dordrecht.

Landsbergen, J. (1984), 'Isomorphic Grammars and Their Use in the Rosetta Machine Translation System'. Philips Research M.S. 12.950. Also in: M. King, (ed.), *Machine Translation Today*. Edinburgh University Press, Edinburgh, 1987.

Landsbergen, J. *(1987), 'Montague Grammar and Machine Translation'*. Philips Research M.S. 14.026. To appear in: P. Whitelock et al. (eds.), *Linguistic Theory and Computer Applications*. Academic Press, London.

McCloskey, J. (1979) *Transformational Syntax and Model Theoretic Semantics*. Reidel, Dordrecht.

Montague, R. (1974) 'The Proper Treatment of Quantification in Ordinary English'. In: R. Montague, *Formal Philosophy*. Selected Papers of Richard Montague, edited by R.H. Thomason. Yale University Press, New Haven.

Schenk, A. (1986), 'Idioms in the Rosetta Machine Translation System'. Philips Research M.S. 13.508. Also in: *Proceedings of Coling '86*, Bonn.