The aim of this paper is to discuss the effect of the compositionality principle on the form of grammatical rules. More specifically, we will investigate how the by now standard analysis of coreference via Reinhart's non-c-command condition can be incorporated in a framework of PTQ-style Montague Grammar with a strong version of the compositionality principle.

In the first part we will extensively discuss the principle of compositionality of meaning and its formalization in Montague Grammar. We will point out that the compositionality principle as such only gains its strength through other restrictions on the form of grammatical rules. This forces us to be more explicit about what kinds of rules the grammar contains. In particular, it motivates the discussion of the role played by morphological features in grammatical rules.

In the second part we will, on the basis of the discussion of the compositionality principle, first pay some attention to the question of whether a Chomsky-approach to coreference by wellformedness conditions on outputs at the level of Logical Form is compatible with a Compositional Montague Grammar (2.1.). Further, we will investigate the relation between compositionality and a strong version of Partee's wellformedness constraint. This version prohibits the use of indexed pronouns in the syntax, and therefore it leads to an analysis which is radically different from Montague's original analysis (Section 2.2.). In Section 2.3. we will consider the original (weaker) version of the wellformedness constraint, which allows indexed pronouns and is better behaved with respect to compositionality. Starting with Montague's original substitution rule and Bennett's analysis of reflexives, we will consider several modifications which incorporate Reinhart's non-c-command condition. At the level of syntax, this theory roughly makes the same predictions regarding coreference as does Reinhart's original theory. At the semantic level, however, our approach will be seen to have several advantages.

As will be apparent from this short survey, our interests in this paper are of a rather abstract and theoretical nature. It is the effect of compositionality on the form of the grammar that we are concerned with.
here, rather than problems of a more empirical character. Thus, for example, we will not present an alternative to Reinhart's analysis of coreference, but investigate the theoretical consequences of incorporating such an analysis in Montague Grammar.

I. Compositionality

Linguistic theory deals with intuitions of native speakers about their languages. In particular, part of the theory has to deal with semantic intuitions. What are these, what should we require a semantic theory to deal with? It is not the aim of this paper to try to give a complete answer to this question. But there are some obvious points one can make: it is evident that speakers have intuitions about correct arguments, i.e. about entailment relations between sentences. So let us require that semantics should at least give a theory of entailment. It is also evident that there are intuitions about word meaning, or constituent meaning, i.e. about which words are synonymous, which words have more than one meaning, and for which words it holds that the meaning of the one is contained in the meaning of the other. (However, speakers do not seem to have clear intuitions about what kinds of entities these (constituent) meanings are.) Obviously, intuitions about constituent meanings and intuitions about the entailment relation are tightly connected, so let us require that semantics should clarify this connection.

There is an interesting claim on this connection (going back to Frege), the principle of compositionality, which says that the meaning of a compound expression is a function of the meaning of its constituents, and of the way these constituents are combined.

We will not enter into the philosophical discussion on what kind of entities meanings are. We accept as such the concept of intension from possible world semantics (i.e. functions from indices to extensions), which, besides being an intuitively acceptable concept, has at present no serious rival in explaining a diversity of semantical facts. We can then formulate the compositionality principle more tentatively as follows: the intension of a compound expression is completely determined by (a) the intensions of the parts, and (b) the way these parts are put together (cf. Janssen (1978)).

Following Janssen's interpretation of compositionality, the principle admits only these grammars in which the meaning of an expression $\alpha$ can be calculated by means of a restricted and explicitly defined class of operations from the meanings of the expressions occurring in the
derivation of $\alpha$, together with the operations on these meanings induced by the syntactic rules used in the derivation of $\alpha$.

It should be stressed that the compositionality principle as such has no empirical content: it only excludes unwanted meanings if severe restrictions are put on what possible derivations can be, and in particular, on which operations on meanings can be induced by the syntactic rules. For example, if in some way reference to derivational history is allowed in the formulation of syntactic rules, the compositionality principle loses much of its impact. It will be possible to be more explicit about this point after having presented a more formalized version of the compositionality principle.

In order to obtain such a formalization of the compositionality principle, let us briefly review the organization of Montague Grammar (MG) (following Montague (1973), henceforth PTQ). The syntactic rules of MG build up complex expressions out of less complex ones, while simultaneously the semantic interpretations of expressions are built up: every expression corresponds to a modeltheoretic object, its meaning (or intension). In PTQ this correspondence is mediated by a translation function, which translates every lexical item into a formula of a proper logical language (here the language of intensional type logic, IL); the translation is then modeltheoretically interpreted, thus giving the meaning of the original expression. An essential consequence of the compositionality constraint is that this logical language IL is dispensable: semantics deals with intensions (modeltheoretic objects) of natural language expressions; a logical language is used only to refer to these intensions more easily. In fact, Montague (1970a) does without it.

Every rule of the syntax combining syntactic items (strings or structures) to produce a new one corresponds with a semantic rule giving the meaning (or the translation) of the produced item from the meanings (translations) of the input terms. This is one of the essential features of MG: rules always come in pairs $(S, T)$. $S$ is an $n$-place function mapping $n$-tuples of strings (PTQ) or labelled bracketings (e.g. Partee (1973)) onto a string or labelled bracketing (henceforth lb) $\alpha = S(\alpha_1, \ldots, \alpha_n)$. Further, there is a function mapping syntactic categories to semantic types, and conforming to this function, $T$ maps $n$-tuples of expressions $(\epsilon_1, \ldots, \epsilon_n)$ of IL (of the appropriate types) onto an expression $\epsilon = T(\epsilon_1, \ldots, \epsilon_n)$ of IL. If $\alpha'_i$ is the translation of $\alpha_i (i = 1, \ldots, n)$, then $T(\alpha'_1, \ldots, \alpha'_n)$ is to be the translation of $S(\alpha_1, \ldots, \alpha_n)$. The only constraint proposed by Montague on the form of the syntactic rules $S$ or the translation rules $T$, is the compositionality constraint. Using the terminology introduced above, we formulate compositionality as follows:
Compositionality Principle (CP).
Let $T$ be a translation rule (thus corresponding to some syntactic rule $S$), and let $M$ be a model for IL. Then if $e_1, \ldots, e_n$ and $\gamma_1, \ldots, \gamma_n$ are expressions of IL of the appropriate types (to function as inputs for $T$), such that $e_i$ is equivalent to $\gamma_i$ ($i = 1, \ldots, n$) in $M$ (i.e. $e_i$ and $\gamma_i$ have the same intension in $M$), then $T(e_1, \ldots, e_n)$ is equivalent to $T(\gamma_1, \ldots, \gamma_n)$ in $M$.

It follows that although $T$-rules are defined as 'syntactic' operations in the grammar of IL, they are essentially operations on meanings, not on formulas. If $T$ assigns a meaning $m$ to a complex expression, it does not matter which IL-formula serves as the translation of that expression, as long as it is an IL-formula which is interpreted as $m$. It is in this sense that the level of the logical language is inessential. In Montague (1970b) this constraint is satisfied by representing both syntax and semantics as algebras, and the translation operation as a homomorphism from the syntax algebra to the semantics algebra, involving only polynomial operations.

What the CP formalizes is the principle that the translation is a means to refer to the meaning of an expression and nothing more than that: if, for example, $\phi(j)$ represents this meaning adequately, then so does $\exists x(\phi(x) \land x = j)$, since both expressions refer to the same model-theoretic object. This idea implies that no conditions on wellformedness referring to variables, constants, or other structural properties of IL-formulas which are not preserved under equivalence, can be imposed. There is no such level as 'Logical Form' in Montague Grammar. This is formulated explicitly in the following extension of the CP:

Extended Compositionality Principle (ECP):

(i) CP: as above.
(ii) for any condition $c$ on outputs $T(e_1, \ldots, e_n)$ of translation rules, and any model $M$ for IL: if $e_i$ is equivalent to $\gamma_i$ in $M(i = 1, \ldots, n)$, then if $T(e_1, \ldots, e_n)$ satisfies $c$, so does $T(\gamma_1, \ldots, \gamma_n)$.

It follows from the ECP that there is no formula $e$ satisfying the following condition on the output of $T$, $e = T(e_1, \ldots, e_n)$: 'e must be the one and only natural translation (or the logical form) of the syntactic expression', since this condition—in its intended meaning (see for example Keenan and Faltz (1978))—is not preserved under equivalence. For example, the two translations $(\lambda P((\lambda Q \forall x (Qx \rightarrow Px)) (\text{man'})))$ (walk')
and $\forall x (\text{man}'(x) \rightarrow \text{walk}'(x))$ of the sentence every man walks cannot be distinguished in the grammar, since they refer to the same meaning. Compositionality is incompatible with such a notion of 'naturalness of the translation'.

Another consequence of the compositionality principle is the distinction between the structure and the derivation of sentences. As pointed out at the beginning of this section, semantics has to deal with entailment relations, and hence it has to account for scope ambiguities. Scope ambiguities make it necessary to associate distinct meanings with a sentence (or a structure). The compositionality principle then implies that, if arguments can be given that the ambiguity is neither structural nor lexical, one can only obtain these different readings through a derivational ambiguity. In other words, we have to associate different derivations with such ambiguous sentences. Thus, an essential feature of a compositional theory is the distinction between derivation-trees and structure-trees (cf. Janssen (1978)). For example, there are independent reasons for assigning one structure to the sentence everyone in this room knows one language (that is, the ambiguity is not structural). The different readings cannot originate from different intensions of the parts, since both readings should be produced from the same parts (the ambiguity is not lexical). So, by compositionality, we can only distinguish its two meanings if we derive them differently.

A common argument against this view on scope ambiguities is that semantic theory need not associate two readings with such a sentence. It would suffice to produce only one reading (with everyone having wide scope over one language), and state (pragmatic) conditions under which the interpretation of this sentence is narrowed down to the other reading, which entails the generated one. However, this does not work in cases where the different readings are logically independent. We will come back to this at the end of Section 2.

As has already been mentioned, another consequence of the compositionality principle is the lack of a level of 'Logical Form'. This has important consequences for the analysis of anaphora in Montague Grammar. These consequences will be extensively discussed in the next section.

First, however, we need to be more explicit about the relation between the compositionality principle, and the organization of the grammar in general. At the beginning of this section it was pointed out that the strength of the compositionality principle depends heavily on what other restrictions we impose on the form of grammatical rules. A place in the grammar where this becomes crucial is at the point where
morphology fits into syntax: in morphological features. Let us digress briefly to explain this in more detail. We take such features to have a strictly morphological function: they are to serve as inputs for morphological operations mapping completed labelled bracketings onto final forms ('surface structures', 'phonological representations', etc.). But even if such features are meant to be morphological, there are many syntactic (or semantic) functions they can perform. For example, if it is possible to mention features in the input of syntactic rules, we can encode rule ordering in features (derivational history). Further, features could be used to arbitrarily filter out structures at the level of syntax; and if we do not forbid the encoding of arbitrary information about aspects of derivational history in features, Partee's constraint "do not refer to arbitrary aspects of derivational history" (Partee (1979a), (1979b)), clearly loses its meaning. This, in turn, affects the compositionality principle. For the essential part of the CP says that we have to operate with logical formulas (translations) as if we are operating with meanings; in particular, semantic operations on meanings should not depend on the way these translations are derived. Thus, we have to restrict the role played by features. It goes beyond the scope of this paper to present a theory of features that achieves this. Such a theory is proposed in Landman and Moerdijk (1982).

We now state some minimal assumptions about the form of the rules. It is not our aim to present a full formal description of the grammar; we only outline a minimal framework in which we can formulate our own proposals concerning the analysis of anaphora (Section 2), and in which the CP does not lose its strength (thus, although we do not discuss features we do have to restrict the relation between syntax and morphology).

Following Partee (1979a), we take a rule $R$ of the grammar to have the form

$$R = (F_R, G_R, S_A_R),$$

where $F_R$ is a syntactic operation, $G_R$ is a semantic operation, and $S_A_R$ is a structural analysis. $R$ applied to lb's $\alpha_1, \ldots, \alpha_n$ of specific categories, translating into $\alpha'_1, \ldots, \alpha'_n$ respectively, yields a complex lb $F_R(\alpha_1, \ldots, \alpha_n)$ of some specified category with translation $G_R(\alpha'_1, \ldots, \alpha'_n)$, only if $(\alpha_1, \ldots, \alpha_n)$ satisfies $S_A_R$. If one wants a theory of grammar of this form to be explanatorily adequate in any sense of the word, it has to be clarified

(i) what forms the $F_R$-rules can take,
(ii) what forms the $G_R$-rules can take,
(iii) what properties (of lb’s) the \( \text{SA}_R \)'s can express,
(iv) given \( R \), what the relation between \( F_R \), \( G_R \) and \( \text{SA}_R \) is.

Note that the answers to these questions cannot be independent of each other. Until now, we have only adopted the category-type agreement of PTQ and the ECP. Some further answers to these questions are discussed in Partee (1979a) and Partee (1979b). In this paper, we will only use two kinds of basic syntactic operations; concatenation to build up structure, and substitution which preserves structure.

A concatenation rule is a rule which has as input categories a ‘function category’ \( \alpha/\beta \) and an ‘argument category’ \( \beta \), and as output category the ‘value category’ \( \alpha \). A concatenation rule transforms labelled bracketings \([\xi]_{\alpha/\beta} \) and \([\xi]_{\beta} \) into a more complex labelled bracketing \([[[\xi]_{\alpha/\beta} [\xi]_{\beta}]_{\alpha}] \).

A substitution rule substitutes an lb \([\xi]_{\alpha} \) for some \((k-)\)indexed sub lb’s \([\eta_k]_{\alpha} \) in an lb \([\ldots [\eta_k]_{\alpha} \ldots [\eta_k]_{\alpha} \ldots]_{\beta} \). The result of substituting \([\xi]_{\alpha} \) for \([\eta_k]_{\alpha} \) in this lb depends of course on the particular rule used. We do not go into the problem of what possible forms these results can take here.

Note that this set of possible types of rules only admits rules in which either the output category is one of the input categories (substitution), or it is the ‘multiplication’ of the input categories (concatenation).

Concerning the second question (what forms the \( G_R \)'s can take) and part of the fourth (the relation between the \( F_R \)-operations and the corresponding \( G_R \)'s) we will assume the following: if \( F_R \) is a concatenation rule, then \( G_R \) is the operation of functional application (cf. Partee (1979a)). Thus if \([\xi]_{\alpha/\beta} \) translates into \( \xi' \) and \([\xi]_{\beta} \) translates into \( \xi' \), then \( F_R([\xi]_{\alpha/\beta}, [\xi]_{\beta}) = [[[\xi]_{\alpha/\beta} [\xi]_{\beta}]_{\alpha}] \) translates into \( \xi'(" \xi'\). To state the translation rules corresponding to substitution, we take indexed expressions to translate into expressions containing only variables and logical operators (and in particular a free variable with the same index as the indexed expression), which seems reasonable (cf. Janssen (1979), Partee (1979b)). If \( F_R \) is a substitution rule substituting an lb \([\xi]_{\alpha} \) for an indexed lb \([\eta_k]_{\beta} \) in \([\phi]_{\beta} \), and \([\xi]_{\alpha} \) translates into \( \xi' \), \([\phi]_{\beta} \) into \( \phi' \), and \([\eta_k]_{\alpha} \) translates into an IL-formula \( \eta_k ' \) which contains an indexed variable \( x_k \), then the result of applying \( F_R \) has as translation \( \xi'(' \lambda x_k \phi') \).

In this way, for the types of grammatical rules used in this paper we have severely restricted the semantic operations; and these semantic operations all satisfy the compositionality principle. In line with the remarks about features made above, it is plausible to require that conditions on application of syntactic rules (\( \text{SA}_R \)'s) are purely structural, and cannot ask for the (non-)occurrence of certain features. They can
only refer to (sub)category labels, occurrences of indexed pronouns, and 'tree-structure properties'. These tree-structure properties are stated in a tree-language (or equivalently an lb-language), so that we may define familiar structural notions from Transformational Grammar. Examples will be given in Section 2.3. The effects that such restrictions on the form of the \( S\alpha_r \)'s have on the analysis of anaphora will be discussed in the next section.

2. The Analysis of Anaphora

2.1. Logical Form

Let us briefly consider the question whether Montague Grammar can adopt the methods of (recent versions, cf. Chomsky (1980), (1981) of) Transformational Grammar used for the analysis of coreference. In Transformational Grammar, an intermediate level of Logical Form (LF) is placed between the level of syntax (\( S\)-structures or surface structures) and that of semantics, at which coreference-structure and scope-relations are determined. At this level, syntactic structures, or rather their 'logical forms' appear, enriched with indices, variables, and quantifiers, which disambiguate the original syntactic structures (i.e., one structure is possibly mapped onto more than one logical form). In this respect the level of LF can be compared with the level of the logical language IL in Montague Grammar. Structural conditions referring to such indices, variables, and quantifiers appearing at LF are then used to filter out logical forms which are not 'wellformed', and through this mechanism overgeneration of sentences showing an ungrammatical anaphoric structure is blocked. Familiar examples of such conditions are the Propositional Island Condition, the Nominative Island Condition, and the Opacity Condition from Chomsky (1980), or the 'Pisa-axioms', cf. Chomsky (1981).

Now can such conditions be used in Montague Grammar? First recall that in Montague Grammar syntactic structures are derived with a certain meaning; the derivation tree can be considered as the disambiguated structure which is translated into IL, and model-theoretically interpreted. But the compositionality principle tells us that the IL-translation is nothing but a means to refer to an intension, and this implies that it is impossible to lay conditions on IL-expressions referring to the position of indexed variables, etc., in order to filter out certain expressions as 'incorrect translations': only conditions that are preserved under logical equivalence are allowed at this level.
Thus, we must conclude that such 'logical form methods' are inapplicable in Montague Grammar. In Section 2.3., we will take an approach which is more akin to earlier methods of Transformational Grammar, which used conditions on application of (structure-preserving) transformations, cf. Chomsky (1973), (1975). In fact, we will give an analysis of anaphora at a purely syntactic level, using conditions on application of substitution rules. (Such rules may be seen to correspond to the structure preserving transformations of TG, cf. Landman and Moerdijk (1981).)

2.2. Compositionality and the Wellformedness Constraint
In this section we will concentrate on the interaction of compositionality and a strong version of Partee's wellformedness constraint (Partee (1979b)), which we formulate as:

\[
\text{do not use syntactic expressions in the derivation of a sentence that are not constituent parts of wellformed surface expressions.}
\]

There is a growing list of articles (Cooper (1975), (1979); Engdahl (1980); Gazdar and Sag (1981); Bach and Partee (1981); Partee and Bach (1981), etc.) that contain analyses of anaphora that try to obey both compositionality and wellformedness, and consequently forbid the use of indexed variables ($he_0, he_1, \ldots$) in the derivation of sentences containing anaphora. Our discussion of such proposals will be based on the grammar presented in Bach and Partee (1981); Partee and Bach (1981), since these papers seem to contain the most detailed analyses. (Henceforth, these two papers will be collectively referred to as BP.) We will argue that this form of the wellformedness constraint is not compatible with the compositionality principle as formalized in Section 1 (and neither with limited relaxations of this formulation of compositionality). In the next Subsection 2.3. we will discuss the effects of a relaxation of the wellformedness constraint which does allow for indexed pronouns in the syntax.

Compositional theories of anaphora which obey the wellformedness constraint should show how sentences (1) and (4) with their respective meanings (2) and (5) can be derived in a compositional way, without making use of structures of the forms (3) and (6).

(1) John hates himself.
(2) $\text{hate}_4(j,j)$
(3) $he_n$ hates $he_n$
We will discuss the grammar of BP by using these examples. The
underlying framework of BP can best be introduced by means of the
following quotation (Partee and Bach (1981), p. 447): (i) We assume Montague’s
general theory (Montague, 1970b, UG) especially with respect to
compositionality, except as modified by Cooper (1975) to
permit the direct assignment of sets of interpretations to ambiguous
sentences without a level of a disambiguated language, including Cooper’s
‘storage’ device (which amounts to a limited relaxation of the com-
positionality requirement).
(ii) We assume the wellformedness constraint of Partee (1979b), strengthened to
exclude indexed forms like he0, he1, ... from the syntax.
(iii) The syntax is limited to a rich context-free language (...).
(iv) The semantics is a direct modeltheoretic interpretation of the syntax; an
intermediate level of translation into intensional logic is dispensable (...).
Simplifying somewhat, their grammar may be said to generate four-
tuples, consisting of a syntactic structure, its IL-translation, and two
sets: the quantifier store (QST) and the local pronoun store (LPST). We
will not discuss storage-mechanisms here; for this, the reader is referred
to Cooper (1975), Ladusaw (1980), Engdahl (1980). Let us turn to
eamples (1) and (3) as analyzed in the grammar of BP. In the lexicon,
we find entries like

\[ \langle \text{John}, \lambda \text{PP}\{^j\}, \phi, \phi \rangle \]
(QST and LPST are empty)

\[ \langle \text{he}, \lambda \text{PP}\{^i\}, \phi, \{^i\} \rangle \]
(LPST contains index \( ^i \), \( ^i \) a natural number)

\[ \langle \text{himself}, \lambda \text{PP}\{^i\}, \{\langle \text{self}, ^i\rangle, \{^i\} \rangle \]
(where \( \langle \text{self}, ^i\rangle = (\lambda Rx[R(x)(\lambda \text{PP}\{^i\}]), \{^i\} \rangle, \) so both QST and
LPST are nonempty; self has type \( \tau(\text{IV}), \tau(\text{IV}) \)\)

Note that (8) and (9) are ‘schemas’ for lexical entries; for every index \( ^i \) there are entries (8\( ^i \)) and (9\( ^i \)).

\[ \langle \text{hate}, \text{hate}', \phi, \phi \rangle \]
The rules of NP-storage and quantification are:

\[ \text{STORE } T: \langle T, T', \phi, \phi \rangle \Rightarrow \langle T, \lambda \text{PP}\{^i\}, \{T', i\}, \{i\} \rangle^3 \]
\[ \text{QUAN } T: \langle t, t', \{T', i\}, \{i\} \rangle \Rightarrow \langle t, T'(\lambda xT'), \phi, \phi \rangle \]

Further, there is a rule mapping intransitive verbphrases containing a
stored reflexive (i.e. QST contains stored parts of the meaning of
himself) onto intransitive verbphrases having this stored part quantified, as in (13).

(13) \[ \text{REFL: } \langle \text{hate himself, hate}'(\lambda PP\{x_i\}), \{(self, i), \{i\} \rangle \Rightarrow \langle \text{hate himself, } \lambda PP \{self(\lambda x_i \text{ hate}'(\lambda PP\{x_i\}))}, \phi, \phi \rangle \]

(this reduces to
\[ \langle \text{hate himself, } \lambda x_i \text{ hate}'(\lambda PP\{x_i\})(\lambda PP\{x_i\}), \phi, \phi \rangle. \]

Concatenation of a transitive verbphrase and a term yields an intransitive verbphrase (and takes the union of stores):

(14) \[ \langle \text{hate, hate}', \phi, \phi \rangle + \langle \text{John, } \lambda PP\{^j\}, \phi, \phi \rangle \Rightarrow \langle \text{hate John, hate}'(\lambda PP\{^j\}), \phi, \phi \rangle; \]
\[ \langle \text{hate, hate}', \phi, \phi \rangle + \langle \text{he, } \lambda PP\{x_i\}, \phi, \{i\} \rangle \Rightarrow \langle \text{hate him, } \text{hate}'(\lambda PP\{x_i\}), \phi, \{i\} \rangle. \]

It is in the rule for \( T + IV \Rightarrow t \) that the differences between reflexives and ordinary pronouns are expressed:

(15) \[ \langle T, T', Q_1, P_1 \rangle + \langle IV, IV', Q_2, P_2 \rangle \Rightarrow \langle T'IV, IV'(\lambda T'), Q_1 \cup Q_2, P_3 \rangle \]

where \( P_3 \) is the set of indices \( i \in P_1 \cup P_2 \) such that \( Q_1 \) or \( Q_2 \) contains a term meaning stored with index \( i \). (So \( P_3 \) does not contain index \( i \) if \( Q_1 \) or \( Q_2 \) only contains \( \{\text{self, } i\} \).

Condition: \( P_1 \cap P_2 = \phi \), the LPST of the subject does not share indices with the LPST of the verb.

Final sentences have empty quantifier stores.

For example, sentence (1) (with meaning (2)) is derived by first

(1) \( \text{John hates himself} \)

applying the rule underlying (14) to (10) and (9) \( (TV + T \Rightarrow IV) \), and then applying (13) to the result \( (IV \Rightarrow IV_{ref}) \), and finally applying (15) to (7) and the result of applying (13) \( (T + IV_{ref} \Rightarrow t) \). Sentence (16) cannot be derived.

(16) \* \( \text{John says that Mary hates herself.} \)

The only way to get \text{himself} coreferential with \text{John} and not with \text{Mary} is by not applying REFL to \text{hate himself} (since this would bind \text{himself} to \text{Mary} in the next step), but instead trying to build up (17).

(17) \[ \langle \text{say that Mary hates himself, say}'(\lambda \text{hate}(m, ^{x_i})), \{(self, i), \{i\} \rangle. \]

But the rule that builds up this IV involves an application of (15) to \( \langle \text{Mary, } \lambda PP\{^m\}, \phi, \phi \rangle \) and \( \langle \text{hate himself, } \text{hate}'(\lambda PP\{x_i\}), \{(self, i), \{i\} \rangle \)}
and this rule deletes the index $i$ from LPST (because QST only contains $\langle \text{self}, i \rangle$). Therefore, not (17) is produced, but (17):

$$\langle \text{say that Mary hates himself, say}'(\text{hate}(m, \, \prime x_1)), \{\langle \text{self}, i \rangle\}, \phi\rangle$$

where LPST no longer contains index $i$, and $\langle \text{self}, i \rangle$ cannot be quantified in, but is doomed to stay in LPST.

The condition on rule (15) blocks the derivation of (18). The bound

$$\langle \text{John hates him reading can only be obtained by using a stored John,} \rangle$$

$$\langle \text{John, } \lambda PP\{x_1\}, \{\lambda PP\{\prime j\}, \{i\}\} \rangle$$

and the result of applying (15) to (10) and (8):

$$\langle \text{hate him, hate}'(\prime \lambda PP\{x_1\}), \phi, \{i\} \rangle$$

But then the derivation is blocked, since both local pronoun stores contain index $i$. Indices of reflexives and pronouns are deleted from LPST at $t$-level; consequently, the condition that blocked derivation of (18) does not block derivation of (4),

$$\langle \text{John says that Mary hates him} \rangle$$

In Bach and Partee (1981) it is shown that this grammar makes almost the same predictions as Reinhart’s non-c-command condition (Reinhart (1979)) but combines this with PTQ’s predictions on scope-ambiguities. This is of course a result of considerable importance. There are some difficulties, however, with their grammar as well as with the underlying framework, which seem inherent to their proposal.

Starting with an empirical problem: the aim of their grammar is to give a semantic characterization of the possible relations between an antecedent and an anaphor (in terms of function-argument-structure, instead of by ‘tree properties’ of syntactic structures). Consequently, what the antecedent of an anaphor is is not expressed in the syntax, nor is it detectable through an antecedent-seeking rule (like REFL). This in turn implies that all properties shared by antecedent and anaphor should be essentially semantic properties (properties which can be expressed in modeltheoretic terms). It has been claimed (Cooper (1975)) that this is indeed the case in English: antecedent and anaphor have the same gender and number, and these are properties of the referents (gender is natural gender). In German, however, gender agreement between an antecedent and its anaphor involves grammatical gender: (21) shows grammatical gender-agreement, and (22) natural gender-agreement.
(21)  Der Lügendetektor sagt dem Mädchen dass es ihn betrügt.
The-Masc liedetector tells the-Neut girl that she-Masc cheats.

(22)  *Der Lügendetektor sagt dem Mädchen dass sie es betrügt.
The-Masc liedetector tells the-Neut girl that she-Fem it-Neat cheats.

Since German and English show roughly the same restrictions on possible anaphoric relations, these should be dealt with at the same level. If this level is the semantics, it seems to be necessary for the correct description of German pronoun distribution to claim that individuals (in the model) not only have properties like 'being male', but also like 'being denoted by a word with neuter grammatical gender' (as the word Mädchen 'girl' illustrates). This does not seem reasonable.

Let us now turn to BP's framework principles (i)–(iv). Before commenting on part (i), it is perhaps useful to be slightly more explicit about 'weak compositionality' in Cooper's sense. First recall that Montague generated a sentence with two non-equivalent readings by giving two derivations for this sentence (i.e., by associating two derivation trees with it at the level of the disambiguated language); every syntactic expression has an intension, and with every syntactic operation there is a semantic operation mapping the intensions of the input expressions onto the intension of the output expression. Compositionality restricts the class of possible semantic operations. An essential property of compositional semantics is the dispensability of the logical language (which should not be confused with the (non)dispensability of the 'disambiguated language', the level of derivation trees, cf. Section 1). In Cooper (1975) a grammar is presented which deals with ambiguities without a level of derivation trees (but with storage mechanisms). With each syntactic expression a set of intensions is associated (in case of a non-ambiguous expression, a singleton-set); and even if only non-ambiguous lexical expressions are used in a derivation, it should be possible to end up with an ambiguous complex expression (scope ambiguities). Therefore, for each syntactic operation there is a corresponding set of semantic operations. Weak compositionality—according to Cooper—now means that if one knows the intension of the parts of a complex expression, and one knows how to build up this complex expression, then one knows a set of intensions one of which is the actual intension.

The status of weak compositionality is somewhat less clear than that
of compositionality, due to the fact that it is not clear what kinds of semantic operations may be used. Without being more explicit about this, weak compositionality seems to be a very weak restriction indeed. Strictly speaking, however, weak compositionality is not satisfied by BP's grammar. For an essential feature of weak compositionality as formulated above is that stores of final sentences and of basic expressions are empty. (If not, the set of intensions of a complex expression is not solely determined by the intensions of the parts and the rules used, but also by something else hidden in the stores.) But the lexical entries of he and himself ((8) and (9) above) have non-empty stores (the store of himself even contains the essential part of the meaning of himself).

There is a related but more fundamental problem having to do with the dispensability of the logical language (claim (iv), p. 98). BP classify LPST as something in between a syntactic and a semantic mechanism. This somewhat misleadingly suggests that LPST is (part of) a new component of the grammar. But in fact, what LPST comes down to is a device for storing indices of variables occurring in the IL-translation of expressions. This however, amounts to saying that LPST is equivalent to a mechanism describing properties of IL-translations. The functioning of mechanisms like LPST may be described as:

1. With every basic expression a property of its IL-translation is associated.
2. Corresponding to syntactic rules operations on properties of IL-translations of the inputs are defined.
3. These operations are partial operations: there are conditions on the application of rules stated in terms of properties of IL-translations.

These indices of variables do not exist in the syntax (wellformedness), nor in the semantics (modeltheoretic interpretation), but only at the level of IL. The properties of IL-expressions do not correspond to properties of intensions. But this means that claim (iv) is false—essential use is made of the level of IL-formulas. Thus, the logical language is not dispensable and hence the grammar cannot be said to be compositional (either in the strong, or in the weak sense).

At a formal level, there are several ways to avoid these difficulties. One is to enrich the indices that form the domain of intensions; thus, intensions are now functions from $I \times J \times G \times H$ to extensions, where $H$ is some set of 'pronoun domains' to be defined, and $G$ is the set of variable-assignments. (This solution is hinted at in Groenendijk and Stokhof (1979).) Another is not to take intensions (functions from indices to extensions) as meanings, but something with more structure instead. For example, we could take as meanings not intensions, but
pairs consisting of an intension and a (set of) natural number(s): a variable \( x_i \) then has the meaning (the intension of \( x_i, i \)). Indices now appear as 'real objects' in the model, and the level of IL is again dispensable.

It seems doubtful, however, whether this would give a natural way of expressing modeltheoretically what LPST does. (Perhaps a change from intensions to really structured meanings can provide this; but that requires fundamental changes in the logic, a discussion of which would lead us too far, here.) And even if an enrichment of meanings with indices along these lines would suffice, this still results in a concept of meaning which lacks the intuitive elegance and motivation that the concept of intension has, and it implies the introduction of 'abstract markers' in the semantics. Thus, violation of wellformedness is transposed from the syntax to the semantics, in which 'indexed pronouns' now appear as objects in the model!

Therefore, we may conclude that the strengthening of the wellformedness constraint (no indexed pronouns in the syntax) has consequences which reach much further than just a mild relaxation of compositionality. These consequences are avoided in a framework with a weaker form of wellformedness (so as to admit indexed pronouns in the syntax). In such a framework, the compositionality principle can be maintained in its strongest form; there are no semantic conditions on applications of rules, and instead of introducing structured meanings, the structured syntax (labelled bracketings, which we have anyway) is exploited in analyzing anaphoric relations. We now turn to this.

2.3. Anaphoric Relations in Syntax

In the previous section we discussed theories that do not allow for indexed pronouns in the syntax, and the problems with the principle of compositionality that this gives rise to. We will now consider an approach to anaphora in a framework in which wellformedness is slightly relaxed, in favor of the compositionality principle. The lexicon now contains indexed pronouns, i.e. indexed basic expressions \( \text{PRO}_1, \text{PRO}_2, \ldots \) of category \( T \), with translations \( \lambda PP\{x_1\}, \lambda PP\{x_2\}, \ldots \). The grammar contains a rule which links occurrences of indexed pronouns as anaphors to other expressions. This rule will have the form of a substitution rule.

Let us once more stress that it is not our aim to give new explanations of facts. On the contrary, the rules and conditions that will be formulated are all (minor modifications of) wellknown analyses of anaphora. It is our purpose to see what form these rules and conditions
can take, given a compositional framework as sketched earlier. As a starting point, we take Montague's rule from PTQ, which can be formulated as (1) below. Later on, we will consider several modifications.

(1) if $\alpha$ is an expression of category $T$ and $\phi$ is an expression of category $t$, then $S_n(\alpha, \phi)$ is an expression of category $t$, where $S_n(\alpha, \phi)$ is defined as follows:

(i) if $\alpha$ is not some $\text{PRO}_n$, replace the first occurrence of $\text{PRO}_n$ in $\phi$ by $\alpha$, and all other occurrences of $\text{PRO}_n$ in $\phi$ by $\text{PRO}$, with appropriate case, number, gender, and person (in accordance with $\alpha$).

(ii) if $\alpha$ is $\text{PRO}_k$, replace all occurrences of $\text{PRO}_n$ in $\phi$ by $\text{PRO}_k$ with appropriate case.

For example, $S_3(\text{the man, PRO}_3 \text{ says that Mary loves PRO}_3) = \text{the man says that Mary loves him}$, and $S_5(\text{John, PRO}_5 \text{ loves PRO}_3) = \text{John loves him}$. The translation rule corresponding to $S_n(\alpha, \phi)$ translates $S_n(\alpha, \phi)$ into $\alpha'(hx, \phi')$, where $\alpha'$ and $\phi'$ are the translations of $\alpha$ and $\phi$. It is possible to generate one sentence with several nonequivalent translations, for example in order to treat scope ambiguities.

As has often been observed, this rule (1) has some obvious shortcomings. As the example shows, the rule does not handle reflexives. Also, the rule substitutes the antecedent for the first occurrence of the pronoun with the relevant index, which does not always yield correct results. And it is obvious that the morphological properties dealt with in the rule should be formulated much more systematically and explicitly. The phrase 'with appropriate case, gender, number and person' should be replaced by explicit operations on features. It is, however, immediately clear that this rule can in principle deal with any kind of agreement of syntactic and morphological properties of antecedent and anaphor, since these are linked at a syntactic level. Thus, the problems with grammatical gender discussed in the preceding section are avoided.

Let us first consider reflexivization. One of the first modifications of Montague's rule which treats reflexives by means of indexed pronouns is Michael Bennett's. Slightly simplified, his proposal (Bennett (1975)) for analyzing reflexives can be described as follows. His grammar produces strings (as in PTQ) instead of labelled bracketings, and the rules for reflexivization use a syntactic marker, the so-called 'Bennett-star'. This is a marker that is introduced at some level in the derivation and (if necessary) deleted at another level. Its function is to fulfil some of the tasks of structural conditions in a grammar that cannot explicitly refer to
structures (since it produces strings instead of labelled bracketings). The syntactic marker represents a certain syntactic property that a string has, during that part of the derivation at which the marker is present. Bennett’s rule \( TV + T \Rightarrow IV \) introduces a star, in case the termphrase is an indexed pronoun. For example, \( \text{love} + \text{he}_n \Rightarrow \text{love} *\text{him}_n \). The rule \( T + IV \Rightarrow t \) then takes care of reflexivization as is shown by the following paraphrase:

(A) (1) if the termphrase is \( \text{he}_n \) and the IV contains *\( \text{him}_n \), then *\( \text{him}_n \) is changed to *\( \text{himself}_n \).

(2) if the termphrase is not \( \text{he}_n \) or the IV does not contain *\( \text{him}_n \), then the rule simply concatenates \( T \) and IV.

(B) after this, the stars are deleted.

Thus an IV which contains a star has the \textit{syntactic} property that there is an occurrence of a pronoun in it which is ‘close’ enough to become reflexivized if the string is combined with a pronoun with the same index. At \( t \)-level all stars are deleted. Although say that Mary loves \( \text{him}_n \) contains an occurrence of an indexed pronoun, this occurrence is too ‘far’ for reflexivization. If a simple IV like love *\( \text{him}_n \) is combined with a subject termphrase like Mary, the resulting phrase no longer has the syntactic property represented by the star, hence the star is deleted, and consequently the complex IV say that Mary loves \( \text{him}_n \) does not have this property either. Finally, the substitution rule can substitute John for \( \text{he}_n \) in he\( _n \) hates himself\( _n \) and he\( _n \) says that Mary hates him\( _n \) which gives (2) and (3).

(2) John hates himself

(3) John says that Mary hates him

As mentioned above, devices like the Bennett star may be replaced by syntactic conditions in a grammar which produces labelled bracketings instead of strings. An advantage of such a replacement lies in the fact that the star itself tells us only very indirectly which aspects of the syntactic configuration (except the presence of the star itself) make up the difference between a reflexive and an ordinary pronoun. If we \textit{define} a syntactic property equivalent to the star, we are forced to make a decision about this difference, i.e., we have to give an explicit description of the structural properties determining the choice between a reflexive or a pronoun. We will give a formulation of Bennett’s rules with a simple syntactic condition below. The use of structure in the syntax makes syntactic markers dispensable, which is a desirable consequence in itself, because of the effects markers can have on the
expressive power of the grammar (for example, using markers it is easy to create rule ordering, cf. Landman and Moerdijk (1981)). Let us replace Bennett’s star by a structural condition:

**DEFINITION.** Let \( \gamma \) be an IV-node, and let \( \alpha \) be a node properly dominated by \( \gamma \). We say that \( \alpha \) is close in \( \gamma \) iff there is no t-node properly dominated by \( \gamma \), which properly dominates \( \alpha \). If \( \alpha \) is not close in \( \gamma \), we will say that \( \alpha \) is far in \( \gamma \).

We can now replace Bennett’s rule \( TV + T \Rightarrow IV \) by ordinary concatenation, \( [\alpha]_TV + [\beta]_T \Rightarrow [[\alpha]_TV[\beta]_T]_IV \), while his \( T + IV \Rightarrow t \) will now be as follows.

Given \( [\alpha]_T \) and \( [\beta]_IV \), let \( \gamma \) be the top-IV node of \( [\beta]_IV \); then

\[
[\alpha]_T + [\beta]_IV \Rightarrow [[\alpha]_T[\beta']_IV]_T,
\]

where

1. if \( \alpha \neq he_k \) (for some \( k \)), then \( \beta' = \beta \),
2. if \( \alpha = he_k \), then \( \beta' \) comes from \( \beta \) by replacing each occurrence of \( him_k \) that is close in \( \gamma \) by \( himself_k \).

**TV + T \Rightarrow IV** will produce \( [[\text{love}]_TV[\text{him}_k]_T]_IV \), in which \( \text{him}_k \) is close in the IV-node. So if this structure is combined with \( [he_k]_T \), \( \text{him}_k \) will be reflexivized. On the other hand, if \( [he_k]_T \) is combined with \( [\text{say that [Mary loves him}_k]\}_IV \), then \( \text{him}_k \) will not be reflexivized since it is far in the top IV-node.

If we compare these rules with Bennett’s original formulation, we observe that the new rules are considerably simpler, and that the difference between a reflexive and an ordinary pronoun is expressed more directly in the grammar. A reflexive and its antecedent occur at the same syntactic level (within a t-clause), while an ordinary pronoun and its antecedent occur at different syntactic levels. If we compare this result with BP, we find that the solution just presented mirrors their solution, which tries to distinguish different semantic levels at which antecedent and anaphor can occur. One of the advantages of the syntactic approach is that there are independent (although not totally theory-free) criteria for determining syntactic structure (the so-called structuralist tests). Another is the syntactic boundary nature of the category t. Such structure does not exist in the models of intensional logic.

At the empirical level, a difference between our modification of Bennett’s grammar and the grammar of BP is that the latter incorporates, at the level of LPST, a version of Reinhart (1979)’s c-command condition and our grammar does not. C-command is a structural notion,
defined as follows:

\( \alpha \) c-commands \( \beta \) iff \( \alpha \) does not dominate \( \beta \), \( \beta \) does not dominate \( \alpha \), and the first branching node dominating \( \alpha \) also dominates \( \beta \).

Sentences (4)–(8) admit coreferential readings in just those cases where no anaphor c-commands its antecedent.

(4) *he hates John
(5) John says that he is ill
(6) *he says that John is ill
(7) when John is ill, he goes to the doctor
(8) when he is ill, John goes to the doctor

In order to incorporate this in the grammar, we introduce the following notion:

(9) A possible n-antecedent in an lb \( \phi \) is an occurrence of \( \text{PRO}_n \) in \( \phi \) which is not c-commanded by any other occurrence of \( \text{PRO}_n \) in \( \phi \).

Using this notion, we define a new substitution rule ANA\( _n \) for anaphoric relations (and scope-ambiguities):

(10) ANA\( _n \) (first version): If \( \phi \) is an lb of category \( t \) and \( \xi \) is a termphrase, and \( \alpha \) is a possible n-antecedent in \( \phi \), then ANA\( _n,\alpha \)(\( \phi, \xi \)) is again of category \( t \), where ANA\( _n,\alpha \)(\( \phi, \xi \)) comes from \( \phi \) by replacing \( \alpha \) by \( \xi \) and every other occurrence of \( \text{PRO}_n \) (or \( \text{PROself}_n \)) in \( \phi \) by \( \text{PRO} \) (or \( \text{PROself} \)).

Condition: \( \xi = \text{PRO}_k \) iff \( k = n \).

Together with other rules, this rule makes roughly the correct predictions concerning reflexivization, and it can now deal with sentences (4)–(8). (4) cannot be derived, since it could only be derived from (11), which itself cannot be derived because the second occurrence of \( \text{PRO}_n \) is close in hate \( \text{PRO}_n \), and is reflexivized when this IV is combined with subject \( \text{PRO}_n \). In (12) John can only be substituted for \( \alpha \) (\( \beta \) is c-commanded by \( \alpha \)).

(11) *\( \text{PRO}_n \) hates \( \text{PRO}_n \)
(12) \( \text{PRO}_n \) \( \alpha \) hates \( \text{PROself}_n \)

In (13), both \( \alpha \) and \( \beta \) are possible n-antecedents. Application of ANA\( _{n,\alpha} \) to (John, (13)) yields (7), and application of ANA\( _{n,\beta} \) to (John, (13)) yields (8).
Note that the distinction between reflexives and ordinary pronouns is formalized by complicating the rule $T + IV \Rightarrow t$. It would be in line with the considerations of the first section, however, to express the characterizing properties of antecedent-anaphor relations as conditions on application of the rule that links anaphors and antecedent. We therefore drop the notions close and far, return to the ordinary concatenation rule

$$[\alpha]_T + [\beta]_{IV} \Rightarrow [(\alpha)_{\gamma}[\beta])_{IV}]_I,$$

(consequently, there are no special concatenation rules, with conditions on their application) and reformulate the anaphoric rule $\text{ANA}_n$ by using the notion bound and free (familiar from TG). For occurrences $\alpha$ and $\beta$ of an indexed pronoun $\text{PRO}_n$ we define:

1. $\alpha$ is bound by $\beta$ in (a node) $\gamma$ iff $\beta$ c-commands $\alpha$, and there is no $t$-node $\delta$ dominated by $\gamma$ such that $\delta$ properly dominates $\alpha$, but not $\beta$ (and $\gamma$ dominates $\alpha$ and $\beta$).
2. $\alpha$ is bound in $\gamma$ iff there is an occurrence $\beta$ of $\text{PRO}_n$ in $\gamma$ such that $\alpha$ is bound by $\beta$ in $\gamma$.
3. $\alpha$ is free for $\beta$ in $\gamma$ iff $\alpha$ is not bound by $\beta$ in $\gamma$.
4. $\alpha$ is free in $\gamma$ iff $\alpha$ is not bound in $\gamma$.

To give an example, in (15) $\gamma$ is bound by $\beta$, because $\beta$ c-commands $\gamma$ and every $t$-node dominating $\gamma$ also dominates $\beta$.

$$\text{PRO}_n \text{ says that } [\text{PRO}_n \text{ hates PRO}_n]_I.$$

On the other hand, though $\alpha$ c-commands both $\beta$ and $\gamma$, there is a $t$-node dominating $\beta$ and $\gamma$ but not dominating $\alpha$, thus both $\beta$ and $\gamma$ are free for $\alpha$.

$$\text{ANA}_n \text{ (second version): If } \phi \text{ is an lb of category } t \text{ and } \xi \text{ is a termphrase, and } \alpha \text{ is a possible } n\text{-antecedent in } \phi, \text{ then } \text{ANA}_{n,\alpha}(\phi, \xi) \text{ is again an lb of category } t, \text{ where } \text{ANA}_{n,\alpha}(\phi, \xi) \text{ comes from } \phi \text{ by replacing } \alpha \text{ by } \xi, \text{ and replacing every other occurrence of } \text{PRO}_n \text{ in } \phi \text{ by PROself if it is bound in } \phi, \text{ and by PRO otherwise. Further, if } \phi \text{ translates into } \phi', \xi \text{ into } \xi', \text{ then } \text{ANA}_{n,\alpha}(\phi, \xi) \text{ translates into } \xi'^{(\lambda x_n \phi')}.$$

Condition: $\xi = \text{PRO}_k$ if $k = n$.

The condition guarantees a ‘faithfulness’-property of the rule $\text{ANA}_n$: it cannot change the coreference relations as expressed by the indices of the input, it can only convert coreference into ‘real’ anaphoric struc-
tasures. Without the condition we could, for example, derive (18) from (17) by applying $\text{ANA}_{n,a}(\text{PRO}_m, (17))$.

\[(17) \quad \text{PRO}_n \text{ said that } \text{PRO}_m \text{ liked } \text{PRO}_n\]

\[(18) \quad \text{PRO}_m \text{ said that } \text{PRO}_m \text{ liked } \text{PRO}_\beta\]

In (18), $\alpha$ is a possible $m$-antecedent, so we can apply $\text{ANA}_m$ and substitute Mary for $\alpha$; this yields

\[(19) \quad ^*\text{Mary said that she liked her}\]

\[(20) \quad \text{said}_\alpha'(m, \text{liked}_\alpha(m, m))\]

Thus, without the condition, the antecedent Mary in (20) acts twice as binding an anaphor which is too far from this antecedent to be reflexivized.

Let us briefly indicate how morphological operations, for example gender agreement, can be incorporated in this rule. For definitions and motivation of the concepts involved we refer to Landman and Moerdijk (1981), (1982).

The grammar now produces labelled bracketings of the form $[\alpha]_{C,f}$ where $\alpha$ is either a lexical string or itself a labelled bracketing, $C$ is a category label, and $f$ is a sequence of sets of features and of variables. Some features, like those for gender, number, and person are thought to be lexical, others, like features for case and morphological character (pronominal or reflexive) are structural. This is expressed in the grammar by generating basic expressions (members of the lexicon) with a feature sequence that contains the relevant lexical features, and has variables for the structural features. Thus the lexicon contains entries like:

\[
\begin{align*}
[\text{man}]_C & , \langle \{\text{masc}\}, \{\text{singular}\}, x_{\text{case}} \rangle \\
[\text{old}]_C & , \langle \{\text{masc}\}, \{\text{singular}\}, x_{\text{case}} \rangle \\
[\text{old}]_C & , \langle \{\text{masc}\}, \{\text{plural}\}, x_{\text{case}} \rangle
\end{align*}
\]

Syntactic rules can perform certain operations on feature sequences, most importantly, \textit{agreement}: take the union of the relevant feature sets of the inputs; and \textit{assignment}: substitute, relative to an assignment property of one of the input expressions, a feature set for all the relevant variables in the other input expression. An example of agreement is:

\[
\begin{align*}
[\text{old}]_C & , \langle ., \{\text{singular}\}, . \rangle + [\text{man}]_C , \langle ., \{\text{singular}\}, . \rangle \Rightarrow \\
[\text{old}][\text{man}]_C & , \langle ., \{\text{singular}\}, \{\text{singular}\}, . \rangle \\
( & = [[\text{old}][\text{man}]]_C , \langle ., \{\text{singular}\}, . \rangle)
\end{align*}
\]
Assignment is shown in the following example:

\[
\text{[with]} \text{PP/T, } \text{[the} \text{. } \text{x-case} \text{. } \text{man} \text{. } \text{x-case} \text{. }] \Rightarrow \\
\text{[with] } \text{[the} \text{. } \text{[dative]} \text{. } \text{man} \text{. } \text{[dative]} \text{. }] \text{. } \text{[dative]} \text{. }
\]

In the morphological component structures with feature sequences are mapped onto morphologically expanded forms. Morphological rules cannot expand a structure if it has an ambiguous morphological instruction. Such structures are ruled out. In this way feature agreement is dealt with in the morphological component: concatenation of old and man can only result in an expanded form if they have the same lexical features. If \(\text{old} \text{. } \text{[singular]} \text{.} \) is concatenated with \(\text{man} \text{. } \text{[plural]} \text{.} \), the resulting structure, \(\text{[old}[\text{man}] \text{. } \text{[singular, plural]} \text{.}\) has an ambiguous morphological instruction and is ruled out.

Indexed pronouns are also base generated with lexical features and variables for case and pronomial character. Thus the lexicon contains entries like:

\[
\text{[PRO7]} \text{T, } \text{[ masculine, plural, 3rd person, x-case, x-pron-char]}
\]

\[
\text{[PRO7]} \text{r, } \text{[ neuter, singular, 3rd person, x-case, x-pron-char]}
\]

As part of a sentential structure all case variables are replaced by case feature sets.

Given this, we can reformulate \(\text{ANA}_n\) (ignoring inessential details):

\[
\text{(21) ANA}_n
\]

Let \(\text{[\xi]} \text{T, } \text{[g-gen, f-num, f-pers, f-pron-char]}\) be an lb of category \(T, \phi\), an lb of category \(t\), \(\alpha\) a possible \(n\)-antecedent in \(\phi\),

\[
\alpha = [\text{PRO}_n] \text{T, } \text{[g-gen, f-num, f-pers, f-pron-char]}.
\]

Then \(\text{ANA}_{n,\phi}(\xi, \phi) = [\phi']\), where \(\phi'\) comes from \(\phi\) by replacing \(\alpha\) by \(\text{[\xi]} \text{T, } \text{[g-gen, f-num, f-pers, f-pron-char]}\) (so the lexical features of \(\alpha\) and \(\xi\) are united and \(f_{\text{pron-char}}\) is substituted for \(x_{\text{pron-char}}\))

and replacing every other occurrence \(\beta\) of \(\text{PRO}_n\),

\[
\beta = [\text{PRO}_n] \text{T, } \text{[g-gen, f-num, f-pers, f-pron-char]},
\]

by \(\text{[PRO]} \text{T, } \text{[g-gen, f-num, f-pers, f-pron-char, f-pron-char, k]}\)

where \(k = \{\text{reflexive}\}\), if \(\beta\) is bound in \(\phi\), and \(k = \{\text{pronominal}\}\), otherwise.

\text{Condition}: as before.

\text{Translation}: as before.

This rule enforces gender agreement between antecedent and anaphors: both the antecedent occurrence of \(\text{PRO}_n\) and the other (anaphor) occurrences of \(\text{PRO}_n\) must have the same gender as the term to be sub-
stituted, else the result contains ambiguous morphological instructions and the sentence is ruled out.

It appears that the theories of Reinhart, of Bach and Partee, and the analysis presented in this section make approximately the same prediction with respect to coreference relations. And improvements of one theory can easily be incorporated in others. What we try to show here is how the underlying theoretical framework determines the way in which the analysis is realized in the grammar. Thus, in Montague Grammar the principle of compositionality and the representation of meanings by intensions severely restrict the possibilities for imposing conditions at the level of semantic structures and lead to structural conditions on application of rules at a syntactic level.

Thus, as in Reinhart's theory, in this version of Montague Grammar the syntactic structure of a sentence restricts the possible coreference relations. In Reinhart's theory, however, semantic relations are completely determined by syntactic structure. For example, one quantifier can have wide scope over another iff the first quantifier c-commands the second. Consequently, sentence (22) has only one meaning, (23).

\[(22) \quad \text{Every man loves a woman} \]
\[(23) \quad \forall x (\text{man}_x(x) \rightarrow \exists y (\text{woman}_y(y) \land \text{love}_z(x, y))) \]

Our analysis takes over Montague's quantification mechanism, and therefore we also get

\[(24) \quad \exists y (\text{woman}_y(y) \land \forall x (\text{man}_x(x) \rightarrow \text{love}_z(x, y))) \]

Reinhart claims that (24) (which entails (23)) should be derived through a pragmatic mechanism that 'narrows' the interpretation (23) of (22) to (24). She justifies this 'narrowing'-mechanism by claiming that sentences having two logically independent readings (where narrowing does not work) form a marginal class. This, however, seems to be incorrect. Logically independent readings appear in all sorts of constructions. For example, scope-ambiguities involving an opaque context (de dicto-de re distinctions) cannot be expressed in Reinhart's framework: sentence (25) has two logically independent readings, (26) and (27), of which (26) is the reading that is predicted by the c-command requirement; but (27) cannot be derived through a narrowing-mechanism.

\[(25) \quad \text{Every schoolboy believes that a mathematician wrote 'Through the looking glass'} \]
\[(26) \quad \forall x (\text{schoolboy}_x(x) \rightarrow \text{believe}_z(x, \exists y (\text{mathematician}_y(y) \land \text{wrote}_z(y, 'Through the looking glass')))) \]
If both readings of (25) should be represented (and we think so), then there is no reason to assign (22) only one reading. On the other hand, it is a known fact that not every sentence containing every and a has two readings; (28) has only a reading with wide scope of a:

(28) A woman loves every man

Reinhart considers such examples as supporting her theory, which predicts that the c-commanding quantifier has wide scope. In her theory the other reading is not available, since it cannot be derived from the first one by means of 'narrowing' principles. However, there are many arguments against this view, of which we only mention the following. In sentence (29) the only available reading is the one with wide scope of every over a, which argues strongly against Reinhart's c-command approach with respect to scope.

(29) A medal is available at the finish for every participant

In our opinion, (28) and (29) show that the semantic (and pragmatic) properties of the verb interfere with scope phenomena. It is not difficult to represent this in our framework. In general both readings are available; semantic properties of the verb may sometimes induce restrictions on derivations, excluding one or another reading. In this way we get both the advantages of Reinhart's syntactic theory of anaphoric relations and Montague's explicit semantic theory of scope phenomena.

Let us briefly summarize what our grammar now looks like. There is a set of concatenation rules, which build up structure (TV + T ⇒ IV, T + IV ⇒ t, etc.). There is an anaphoric rule (schema), which is structure-preserving, and operates on a term, a sentence and an indexing structure in that sentence. In this rule the difference between reflexives and ordinary pronouns is formalized as that between bound and free occurrences of indexed pronouns. The wellformedness constraint is satisfied in a version which allows as 'abstract symbols' labelled brackets, and indices (and features). The semantic operations corresponding to our rule are well-defined IL-operations which preserve logical equivalence (functional application, λ-abstraction, etc.). It is clear that the compositionality principle is satisfied.
COMPOSITIONALITY AND ANALYSIS OF ANAPHORA 113

NOTES

* A preliminary version of part of this paper appeared in Landman and Moerdijk (1981). While writing this preliminary version and the present paper we have profited from helpful discussions with Renate Bartsch, Theo Janssen, Henk van Riemsdijk, and, in particular, Jeroen Groenendijk and Martin Stokhof. We would like to thank them all.

The first author gratefully acknowledges the support of part of this work by the Netherlands Organization for the Advancement of Pure Research (ZWO).

1 This restriction has to be weakened somewhat if substitution rules like the one in Landman and Moerdijk (1981) for Wh-substitution are incorporated in the grammar.

2 Indexed pronouns PRO0, PRO1, PRO2, ... (or more precisely [PRO0]r, [PRO1]r, [PRO2]r, ...) are the syntactic variables which are called he0, he1, he2, ... in PTQ. An occurrence of an indexed pronoun PROi in a labelled bracketing φ is a sub labelled bracketing [PROi]r of φ.

3 + and → are derivational symbols used in the formulation of the rules to indicate rule application: α → β means 'rule i applied to α yields β'; α + β → γ stands for concatenation: 'concatenation rule i applied to α and β yields γ'. For readability, rule names are suppressed.

4 Engdahl (1980) gives another semantic analysis of non-coreference phenomena, but the mechanism that takes care of this, the Store Address Convention (Engdahl (1980), p. 85) is far from compositional.

5 node α dominates node β iff there is a (possibly empty) path from α down to β (dominate is reflexive).

6 This notion of antecedent differs from the one discussed in Landman and Moerdijk (1981). There a notion of 'direct antecedent' is used, in order to obtain a 'stepwise' (one anaphor at a time) derivation of anaphoric structures.

REFERENCES


Language. Proceedings of the Third Amsterdam Colloquium, Mathematical Center Tracts, Amsterdam.


