Volume 3 *U/MASS OCCASIONAL PAPERS IN LINGUISTICS -- VOLUME III*

Article 8

1978

Montague Grammar and Classical Transformational Grammar

Emmon Bach

Montague Grammar and Classical Transformational Grammar

Emmon Bach

*Presented at the conference on Montague Grammar, Philosophy, and Linguistics, State University of New York, Albany, April 23-24, 1977.

0. Introduction

In an important paper, Cooper and Parsons (1976) have shown how to obtain transformational grammars of two kinds that are strongly equivalent to a fragment of English grammar given by Montague (1973) (henceforth PTQ, with references to the reprint in Montague, 1974). By 'strongly equivalent' they mean that the pairings of English sentences and their interpretations differ only when one of the systems gives an interpretation which is logically equivalent to an interpretation given by the other. Both grammars are variants of the form of transformational grammar which developed after the publication of Chomsky's Aspects (1965). It has been remarked that the earliest versions of transformational grammar (Chomsky, 1975 [1955], 1957) were closer in spirit and in detail to the system of Montague grammar than later versions of the theory (the 'standard' theory of Chomsky, 1965, and its derivatives). The points of resemblance are three: (1) the classical theory included the notion of a 'transformation marker' (T-marker), which is in some ways like the analysis trees of Montague grammar (Partee, 1975). (2) The systems of 'semantic theory' envisaged for the theory by Katz and Fodor, 1963 (but not later writings by Katz, or Katz and Postal, 1964) included a type of 'projection rule' (Type 2) which is very close to the translation rules of PTQ. (3) Earlier transformational grammars observed a kind of 'local grammaticality' which is parallel to but not in general identical with the 'wellformedness constraint' usually observed in Montague's fragments and considered important by more recent workers in this tradition.[1]

Since a good deal of the motivation that went into the revision of transformational theory undertaken in Aspects arose from the attractiveness of a hypothesis that appears to be wrong (the 'Katz-Postal' hypothesis), it would seem to be worthwhile to explore the earlier model once again. Moreover, studies of the generative capacity of the 'standard' theories have shown that they are too powerful, since they are equivalent in power to various systems defining all recursively enumerable sets (Peters and Ritchie, 1973). The 'classical' theory can be shown to be weaker. The power of the modified system explored here is unknown.

The least satisfactory aspect of all transformational theories has been their treatment of quantification. The present paper is devoted to exploring a modification of the classical theory that makes it possible to handle quantification in a way that parallels the treatment in PTQ almost exactly. The particular grammar given below satisfies a 'local grammaticality' constraint with the sole exception that some structures are generated which underlie no well-formed expressions of English, since they contain variable elements which are 'filtered out' by the transformational component. It should be noted that all versions of interpreted transformational grammars use infinite sets of variables, the differences only arise as to the place in the systems of representation where these variables are used.

In Section 1 I sketch a general theory of modified transformational grammars of the sort just alluded to. We may call them KT-grammars (where KT is mnemonic for the idea of a Kernel set of basic structures that is extended by a Transformational component). In Section 2, I give a KT-grammar for the fragment of English described in PTQ. In 3 I undertake a comparison of the two grammars, and in 4 introduce some improvements that carry the grammar a little closer to descriptive adequacy and extend its coverage of English grammar.

1. General theory of KT-grammars

A KT-grammar is a formal system (Smullyan, 1961) which defines an infinite set of pairs, each consisting of a syntactic structure and a translation of the structure into an interpreted intensional logic. The syntactic rules of each grammar are of several kinds: the kernel rules which correspond roughly to context-free rewriting rules in a phrase structure grammar; several types of transformational rules which convert and combine structures defined by the kernel rules into a larger set of structures that underlie (in part) sentences and noun phrases of the language described. By 'structure' I mean a string of symbols which are either members of the vocabulary of the language or members of a finite set of labeled brackets. For each syntactic rule we require that there be a unique translation rule which is a function from the translations of the input structures to the translation of the structures obtained by applying the syntactic rule.

The system differs from classical transformational theory in several ways:

1. Classical systems define a language as a set of sentences. KT-grammars define a language as a set of sentences and noun-phrases.

2. Correspondingly, KT-grammars include not only transformations for embedding transforms of sentences into sentences but also transformations for embedding noun-phrases into sentences.

3. KT-grammars supplement the finite vocabularies of classical systems with an infinite set of indexed pro-forms of several types. These pro-forms (which are translated into expressions containing variables in the intensional logic) are used not only for ordinary quantification of noun phrases (as in PTQ), but also as substituends for other types of embedding rules (like the 'dummy symbols' of Fillmore, 1963).

4. The sets of rules of classical grammars (and 'standard' grammars) are finite, those of KT grammars are infinite since some of the "rules" are actually rule schemata. That is, I allow rule schemata of the form

.....pro-form$_i$...

where the grammar includes an infinite number of indexed pro-forms, <u>pro-form$_0$</u>, <u>pro-form$_1$</u>,... Such a schema then stands for the infinite set of rules gotten by replacing 'i' by some integer. In this respect they are like Montague grammars.

In addition, I allow (but do not use here) conjunction rules which apply to an indefinite number of instances of structures meeting the conditions of the rule. Such rules have been used in informal presentations of the standard theory as well as the classical theory.

I depart from more traditional formulations in several further ways. Rather than defining derivations and several levels of syntactic structure to which rules of interpretation apply, I give direct recursive definitions for the generated objects, each a pair consisting of a syntactic structure and its translation. Further differences in details will be noted below.

Each grammar makes use of a number of different sets of symbols, of which some are given by the general theory for every grammar, some are particular to a grammar. The former includes a finite set of labeled right and left brackets, including in particular "$[_S$," "$]_S$," "$[_{NP}$," "$]_{NP}$," and two infinite sets of "meta-variables" which are (i) variables standing for arbitrary strings of symbols and brackets (including the null string <u>e</u>), and (ii) variables which stand for arbitrary well-formed labeled bracketings.

The particular vocabularies include a set of kernel-terminal elements, and a partially overlapping set of T-terminal elements, elements which are terminal for the transformational component. The former but not the latter includes an infinite set of indexed pro-forms of various kinds, including, in particular, pro-sentences and pro-noun phrases. I define the language generated by the grammar as that set of strings over the T-terminal vocabulary which is obtained by removing all brackets from the first member of a pair included in the set of structural descriptions of sentences and noun phrases (syntactic structure plus translation) defined by the grammar.

In addition, I make use of a suggestion of Partee's (forthcoming) and define recursively certain properties of expressions of the syntax for gender, case-marking and verb-morphology. Some of the rules which do this and spell out the effects are included in the last set of rules (M-rules) which map the output of the transformations into well-formed English sentences.

The grammar will then take the following form:

Lexicon: a listing of lexical items each given with its assignment to a syntactic category and a translation into intensional logic.

Kernel rules: a recursive definition of a set of pairs, the first being a syntactic structure (labeled bracketing string), the second its translation into intensional logic. These are the kernel structural descriptions.

Transformations: rules which extend the set of structural descriptions for noun phrases and sentences. These transformations fall into four types:

Preliminary singulary transformations which apply to simple structural descriptions which underlie kernel sentences and noun-phrases and which apply to embedded structures before they are embedded and to their matrices after embedding.

Embedding transformations which substitute transforms of sentences for various kinds of indexed pro-forms or substitute noun-phrases for indexed pro-noun-phrases.

Conjoining transformations which form coordinate structures or reduce coordinate structures.

Final Singulary Transformations and Morphological Rules which are extrinsically ordered and apply after all other rules to map the set of outputs from the previous rules into English sentences.

I shall not give an explicit formalization for the general theory of grammars of this kind (since I think it's too early to make a lot of particular decisions about the general theory). However, the grammar given in the next section is intended to be completely explicit and to be understood as it stands. This aim leads to a rather cumbersome formulation for a number of rules, which could be simplified considerably if the grammar were embedded, as it should eventually be, in a general theory of grammars of this kind. I'll point out places where I think such simplification can be expected to be possible. I hope that the fragment will meet the objections of people like Montague and Michael Bennett, who have complained about the inexplicitness of transformational grammars.

2. A KTG for PTQ. The following grammar G is (with the exceptions noted at the beginning of Section 3) strongly equivalent to PTQ. For the intensional logic I will simply adopt without change the language of PTQ, Section 2, and its interpretation.

2.1 Lexicon. The lexicon of G is a set of triples, the first an expression, the second a category label (to be used to define the set of labeled brackets), and the third a translation of the expression into the intensional logic. The basic bracket labels for lexical elements correspond to the basic categories of PTQ as follows:

G	PTQ
Vi	B_{IV}
N	B_T (!)
Vt	B_{TV}
AvV	B_{IAV}
CN	B_{CN}
AvS	$B_{t/t}$
Prep	$B_{IAV/T}$
Vthat	$B_{IV/t}$
Vto	$B_{IV/IV}$

From the point of view of transformational grammar certain phrase-types of PTQ are superfluous (for this fragment but not necessarily for a richer one) and hence are omitted. They are the phrase types corresponding to Vt, AvS, Prep, Vthat, and Vto. For all others we include among the bracket labels for every X on the above list a label XP (corresponding to P_X for B_X as above). In addition I use S corresponding to P_t and Aux with no corresponding category in PTQ.

I will not give a complete lexicon. With the exception of pronouns (given below) it consists of the set of basic expressions of PTQ, each followed by a label according to the above table, and its translation according to the translation rules of PTQ (but that and to are introduced by rules rather than as part of a basic expression). For example, the lexical entry for find is this:

$$\langle \text{find, Vt, find}' \rangle$$

I use the following indexed pro-NP's (reasons for the particular choices will be made clear below):

$$\left.\begin{array}{l} \langle \text{her}_{6i}, \text{N}, \hat{P}\, P\{x_{6i}\} \rangle \\ \langle \text{him}_{6i+2}, \text{N}, \hat{P}\, P\{x_{6i+2}\} \rangle \\ \langle \text{it}_{6i+4}, \text{N}, \hat{P}\, P\{x_{6i+4}\} \rangle \end{array}\right\} \quad i = 0, 1, 2, \ldots$$

(the individual concept variables introduced in the rules below will be chosen from the set $\underline{x}$, $\underline{y}$, $\underline{x}_5$, $\underline{x}_7$, ...)

In addition to the above variables for individual concepts, the rules will introduce the following variables:

$that_i$ translated as p_i (propositional variables of type s, t

$\left.\begin{array}{l} such_i \\ to_i \\ one_i \end{array}\right\}$ translated as q_i (variables over properties of individual concepts: $\langle s, \langle\langle s, e\rangle, t\rangle\rangle$)

(These are of the same type as Montague's $\underline{P}$, $\underline{Q}$, and I'll assume that they are disjoint from these by letting $\underline{P}$, $\underline{Q}$, etc. be indexed by even numbers, the above by disjoint sets of odd numbers).

2.2 Kernel Rules. The set of kernel structural descriptions of $\underline{G}$ is the smallest set $\underline{SD_K}(\underline{G})$ defined by the lexicon and the rules KR0-7.

KR0. If $\langle \alpha, A, \alpha' \rangle$ is in the lexicon of $\underline{G}$, then $\langle [_A \alpha], \alpha' \rangle$ is in $\underline{SD_K}(\underline{G})$.

(This is a general rule for any grammar and would be part of the definition of a grammar and its language according to a fully formalized general theory.)

The following rules are part of the particular grammar $\underline{G}$. I will give them in an abbreviated form which can be made clear by giving the 'official' form of the first rule (it's easy but tedious to spell out this abbreviation explicitly). (I omit labels on right brackets where it's clear).[2]

If $\langle [_{AvS} X_1], \alpha \rangle \in SD_K(G)$, then $\langle [_S [_{AvS} X_1]\ that_i],$
$\alpha(p_i) \rangle \in SD_K(G)$

KR1. $S \rightarrow AvS\ that_i$ $\qquad AvS'(p_i)$

KR2. $S \rightarrow NP\ Aux \left\{\begin{array}{l} ViP \\ to_{6i+1} \end{array}\right\}$ $\qquad$ (a) $NP'(\hat{}ViP')$ (b) $NP'(q_{6i+1})$

KR3. $ViP \rightarrow \left\{\begin{array}{l} Vto\ to\ to_{6i+1} \\ Vthat\ that\ that_i \\ Vt\ \ NP \\ Vi \\ to_{6i+1}\ \ AvVP \\ be\ a\ CNP \end{array}\right\}$

(a) $Vto'(q_{6i+1})$
(b) $Vthat'(p_i)$
(c) $Vt'(\hat{}NP')$
(d) Vi'
(e) $AvVP'(q_{6i+1})$
(f) CNP'

KR4.	AvVP $\rightarrow$	AvV	(a)	AvV'
		Prep NP	(b)	Prep'(^NP')
KR5.	NP $\rightarrow$	NP or NP	(a)	$\hat{P}[NP'_1(P) \quad NP'_2(P)]$
		N	(b)	N'
		every CNP	(c)	$\hat{P} \bigwedge x\ [CNP'(x) \rightarrow P\{x\}]$
		the CNP	(d)	$\hat{P} \bigvee x\ [\bigwedge y[CNP'(y) \leftrightarrow x = y]\ \&\ P\{x\}]$
		a CNP	(e)	$\hat{P} \bigvee x\ [CNP'(x)\ \&\ P\{x\}]$
KR6.	CNP	CNP $such_{6i+3}$	(a)	$\hat{x}[CNP'(x)\ \&\ q_{6i+3}\{x\}]$
		one_{6i+5}	(b)	$\hat{x}[q_{6i+5}\{x\}]$
		CN	(c)	CN'
KR7.	Aux	Pres		no translation rule

Here and below I assume that improper binding of variables will be avoided by replacing a translation by a logically equivalent one by changing the subscripts on variables which are bound in an expression to be combined with another expression whenever the second expression contains variables (of the same type) with the same subscript. E.g. when we combine an expression containing a bound variable $\underline{u_i}$ with an expression containing $\underline{u_i}$ (free or bound), we change the first expression by substituting $\underline{u_m}$ for $\underline{u_i}$ throughout where $\underline{m}$ is the least odd or even integer such that $\underline{u_m}$ occurs in neither expression, according as $\underline{i}$ is odd or even and m = i (modulo 6). Details of the rules will be commented on below in Section 2.4.

2.3 Transformations. The transformations will be given in an explicit form. The only abbreviations will be these.

(1) In place of "$[_A W_i]_A$" (where $\underline{W_i}$ is a variable over well-formed labeled bracketings) I will write "A_i" (without subscript if only one such variable with a given bracket label occurs).

(2) I will use numbered curly braces to abbreviate sets of partially similar rules in a way that is familiar in the linguistic literature:

$$\ldots \left\{ \begin{matrix} X \\ Y \\ Z \end{matrix} \right\}_i^{} \ldots \left\{ \begin{matrix} A \\ B \\ C \end{matrix} \right\}_i \ldots$$

abbreviates the three rules or statements

... X ... A ...
... Y ... B ...
... Z ... C ...

(3) $\underline{pro}_i$ is an abbreviation for

$$\left\{ \begin{matrix} him_i \\ her_i \\ it_i \end{matrix} \right\}_j$$

Readers of the linguistic literature should be warned that the formalism departs from the usual formalism in this respect: When we have a rule starting "If $\langle [_S A\ B\ X], \alpha \rangle$ is in SD(G)" where $\underline{A}$, $\underline{B}$ are bracket labels or brackets or constants then the description is not met by any labeled bracketed strings in which something, even only brackets, separates $\underline{A}$ and $\underline{B}$. (I relax this somewhat in the M-rules.)

(4) $\underline{X}$, $\underline{X}_1$, $\underline{X}_2$, ... are "meta-variables" standing for arbitrary strings of elements and brackets and $\underline{W}$, $\underline{W}_1$, $\underline{W}_2$, ... are variables for arbitrary well-formed labeled bracketed strings.

Finally, I assume throughout a general convention by which the structure appearing in the then-clause of a rule is to be the reduction of the structure actually given (see Peters and Ritchie, 1973a), that is the result of applying the following reduction <u>red</u> until no longer applicable

1. <u>red</u> $([_A\ ]_A) = \underline{e}$ (the empty string)
2. <u>red</u> $([_A[_A W\ ]_A]_A = [_A\ W\ ]_A)$ (where W is as usual a well-formed labeled bracketing).

By (1) we eliminate 'empty brackets'; by (2) we eliminate redundant labeled brackets.

The set of transformationally extended structural descriptions of $\underline{G}$ is the smallest set $\underline{SD(G)}$, such that its membership follows from T.0 - T12 and M1 - 10.

T.0 If $\langle [_{NP} X\ ], \alpha \rangle$ is in $\underline{SD}_{\underline{K}}(\underline{G})$, then it is in $\underline{SD}(\underline{G})$ also.

If $\langle [_S X], \varphi \rangle$ is in $\underline{SD}_{\underline{K}}(\underline{G})$, then it is in $\underline{SD}(\underline{G})$ also.

(Again, this is to be part of the general theory of KT-grammars. This rule says that every kernel NP or S structure is also in the set to be transformationally extended.)

2.31 Preliminary Singulary Transformation.

T1. Future tense

If $\langle [_S NP\ [_{Aux} Pres]\ ViP\ ], \varphi \rangle \in SD(G)$, then $\langle [_S NP\ [_{Aux} will]\ ViP], W\varphi \rangle \in SD(G)$

T2. Perfect tense

If $\langle [_S NP\ [_{Aux} Pres]\ ViP\ ], \varphi \rangle \in SD(G)$, then $\langle [_S NP\ [_{Aux} Pres\ have]\ ViP], H\varphi \rangle \in SD(G)$

T3. Negative

If $\langle [_S NP\ Aux\ ViP], \varphi \rangle \in SD(G)$, then $\langle [_S NP\ Aux\ not\ ViP], \neg\varphi \rangle \in SD(G)$

T4. Pronominalization (obligatory)

If $\langle [_S X_1\ pro_i\ X_2\ pro_i\ X_3\ ], \varphi \rangle \in SD(G)$

$\langle [_S X_1\ pro_i\ X_2\ pro\ X_3\ ], \varphi \rangle \in SD(G)$

where pro is him if pro_i is him_i

her if pro_i is her_i

it if pro_i is it_i

This rule is assumed to iterate until no longer applicable.

2.32 Embedding Rules

For the purposes of the embedding transformations, I assume the following recursive definition of gender:

1. man, John, Bill, him_i are Masculine
2. woman, Mary, her_i are Feminine
3. it_i and all other members of CN are Neuter.
4. Let α range over Masculine, Feminine, or Neuter

If X_1 is α so are

(i) $[_{CN} X_1]$

(ii) $[_{CNP}[_{CN} X_1]\ ]$

(iii) $[_{CNP}[_{CNP} X_1]\ X_2]$ (where the whole thing is a well-formed 1b)

(iv) $[_{NP} X_2\ [_{CNP} X_1]]$ (where the whole thing is a well-formed 1b)

(v) $[_{N}\ X_1]$

(vi) $[_{NP}\ X_1]$

(I leave unsettled the question of the status of phrases like John or Mary.) If two expressions are both Masculine, Feminine, or Neuter, we say that they agree in gender.

T5 That-embedding

(a) If $\langle [_S \text{ AvS that}_i], \gamma \rangle$ and $\langle S, \varphi \rangle \in SD(G)$

than $\langle [_S \text{AvS S}], \lambda p_i \gamma\ ({}^\wedge\varphi) \rangle \in SD(G)$

(b) If $\langle [_S NP\ [_{Aux} \text{ Pres}]\ [_{VIP} \text{ Vthat that that}_i]\ ], \varphi \rangle$

and $\langle S, \psi \rangle \in SD(G)$

then $\langle [_S NP\ [_{Aux} \text{ Pres}\ ]\ [_{ViP} \text{ Vthat that S}\ ]\ ],$

$\lambda p\ \ \varphi ({}^\wedge\psi) \rangle \in SD(G)$

T6 To-embedding

(a) If $\langle [_S NP\ [_{Aux} \text{ Pres}] \text{ to}_i\ ], \varphi \rangle$ and $\langle [_S\ [_{NP}\ [_N \text{ pro}_j]]$

$[_{Aux} \text{ Pres}\ ] \text{ ViP}\ ], \psi \rangle \in SD(G)$

then $\langle [_S NP\ [_{Aux} \text{ Pres}] \text{ ViP}\ ],\ \ \lambda q_i\ \varphi\ (\hat{x}_j \psi) \rangle \in SD(G)$

where NP and pro_j agree in gender

(b) If $\langle [_S NP\ [_{Aux} \text{ Pres}]\ [_{ViP} \text{ Vto to to}_i\ ]\ ], \varphi \rangle$ and

(second member as in (a)) $\in SD(G)$

then $\langle [_S NP\ [_{Aux} \text{ Pres}]\ [_{ViP} \text{Vto to ViP}\ ]\ ], \lambda\ \ q_i\ \varphi (\hat{x}_j \psi\ ) \rangle$

$\in$ SD(G) where NP and pro_j agree in gender.

(c) If $\langle [_S NP\ [_{Aux} \text{ Pres}]\ [_{ViP} \text{to}_i \text{ AvVP}]\ ], \varphi \rangle$ and (second member

as in (a)) $\in SD(G)$

then $\langle [_S NP\ [_{Aux}\ Pres]\ [_{ViP} ViP\ AvVP]\ ],\ \lambda\ q_i\ \varphi\ (\hat{x}_j \psi)\rangle$ $\in$ SD(G) where NP and pro_j agree in gender.

T7 Relative clause

If $\langle [_A\ X_1\ [_{CNP} CNP\ such_i]\ X_2], \alpha\rangle$ and $\langle [_S X_3\ pro_j\ X_4], \varphi\rangle$ $\in$ SD(G) where A = NP or S; and pro_j and CNP agree in gender then $\langle [_A\ X_1\ [_{CNP} CNP\ such\ that\ [_S X_3\ pro\ X_4]]\ X_2],\ \lambda q_i \alpha (\hat{x}_j \varphi)\rangle$ $\in$ SD(G) (and pro is as in T4)

T8 One-embedding

If $\langle [_A\ X_1\ one_i\ X_2\ ], \varphi\rangle$ and $\langle [_S\ [_{NP}\ [_N\ pro_j\ ]\ ]\ [_{Aux} Pres]$ $[_{VP}$ be a CNP$]\ ], \psi\rangle \in$ SD(G) where A = S or NP then $\langle [_A\ X_1\ CNP\ X_2\ ],\ \lambda\ q_i\ \varphi\ (\hat{x}_j \psi)\rangle \in$ SD(G)

T9 Quantification

If $\langle [_S\ X_1\ pro_i\ X_2], \varphi\rangle$ and $\langle NP, \alpha\rangle$ $\in$ SD(G) then $\langle [_S X_1\ NP\ X_2],\ \alpha\ (\hat{x}_i \varphi)\rangle \in$ SD(G)

I require (as part of the definition of a KTG and its operation) the following general conventions on generalized transformations involving subscripted pro-forms:

Uniqueness convention: If an embedding transformation mentions one or more indexed pro-forms than it is defined only for structures satisfying the rule in a unique way (that is for particular choice of i there can be only one occurrence of pro-form_i in the input structures).

This convention ensures that T4 must have been iterated exhaustively on an input structure before it is available for embedding into a structure by T6 (to-embedding), T7 (relative clause), or T8 (one-embedding) or before a NP can be embedded into it by T9 (quantification). It also ensures that one and only one occurrence of a given $that_i$, to_i, $such_i$, or one_i will be contained in the matrix for the embedding rules involving these proforms.

2.33 Rules of conjunction and disjunction (CT)

T10 And-conjunction

If $\langle S_1, \varphi\rangle$ and $\langle S_2, \psi\rangle \in$ SD(G) then $\langle [_S\ S_1\ and\ S_2\ ], \varphi \wedge \psi\rangle \in$ SD(G)

T11 Or-conjunction

Same as T9 but with or and $\vee$ replacing and and $\wedge$, respectively.

T12 Conjunction reduction (ViP conjunction)

If $\langle [_S [_S [_{NP} [_N \text{pro}_i]] [_{Aux} \text{Pres}] \text{ViP}_1 \begin{Bmatrix} \text{and} \\ \text{or} \end{Bmatrix}_1^{\;1}$

$[_S [_{NP} [_N \text{pro}_i]] [_{Aux} \text{Pres}] \text{ViP}_2], \varphi \rangle \in \text{SD(G)}$

then $\langle [_S [_{NP} [_N \text{pro}_i]] [_{Aux} \text{Pres}][_{ViP} \text{ViP}_1 \begin{Bmatrix} \text{and} \\ \text{or} \end{Bmatrix}_1^{\;1}$

$\text{ViP}_2]], \varphi \rangle \in \text{SD(G)}$

2.34 Final Singulary Transformations and Morphological Rules. These rules deal primarily with getting the forms of pronouns and the verbs to come out right. Again I make use of certain recursively defined properties of expressions. Since these rules deal with government, i.e. properties determined by the external environment of the expressions in question, the recursive definitions work 'top-to-bottom.'

M1 Nominative case marking:

If NP occurs in the environment: $[_S X_2 \_\_\_\_ X_3]$, where X_2 = e or AvS
then NP is Nominative

M2 Finite Verb marking

If ViP occurs in the environment: $[_S X_2 [_{Aux}\text{Pres}] \_\_\_\_ X_3]$,
then ViP is Third Person Singular Present and Pres is deleted.

M3 Past Participle Marking

If ViP occurs in the environment: $[_S X_2 \text{ have}]_{Aux} \text{ (not) } \_\_\_ X_3]$,
then ViP is Past Participial.

M4 Property Spreading

(a) Let A = NP or VP and α = Nominative, Third Person Singular Present, or Past Participial

If $[_A A_1 \begin{Bmatrix} \text{or} \\ \text{and} \end{Bmatrix} A_2]$ is α, so are A_1 and A_2

(b) If $[_{NP} N]$ is α so is N

(c) If $[_{ViP} \text{ViP AvVP}]$ is α so is ViP

(d) If $[_{ViP}$ Vx X_2] is α so is Vx where Vx = Vi, Vt, Vthat, Vto, or <u>be</u>

M5 $[_S\ X_1\ [_{Aux}$ Pres] (not) X_2 <u>be</u> X_3] $\Rightarrow$ $[_S\ X_1\ [_{Aux}$ be] (not) $X_2\ X_3]$ when $\underline{X_2}$ is a string of brackets.

M6 Do-support

$[_S\ X_1\ [_{Aux}$ Pres] X_2] $\Rightarrow$ $[_S\ X_1\ [_{Aux}$ Pres do] $X_2]$

M7 Have / do / be morphology

Pres have $\Rightarrow$ has
Pres do $\Rightarrow$ does

M8 Pronoun morphology

The Nominative forms of <u>her</u>, <u>him</u>, $\underline{her_i}$, $\underline{him_i}$ are <u>she</u>, <u>he</u>, $\underline{she_i}$, and $\underline{he_i}$ respectively.

M9 Verb-morphology

(a) The Third Person Singular Present form of <u>be</u> is <u>is</u>, otherwise is it $\underline{\alpha s}$ when α is in Vt, Vi, Vthat, Vto

(b) The Past participial forms of <u>run</u>, <u>rise</u>, <u>find</u>, <u>lose</u>, <u>eat</u>, <u>be</u> are <u>run</u>, <u>risen</u>, <u>found</u>, <u>lost</u>, <u>eaten</u>, <u>been</u>, respectively, for all other members α of <u>Vt</u>, <u>Vi</u>, <u>Vthat</u>, <u>Vto</u> it is $\underline{\alpha d}$ if α ends on a vowel, otherwise it is αed

M10 <u>a/an</u> alternation

a $\rightarrow$ an in the environment of a following vowel (this is a morphological rule applying to the morpheme 'a')

2.4 <u>Comments on the grammar</u>

<u>Lexicon</u>. The basic form of pronouns is taken to be the non-nominative or objective form in line with what seems to be the correct and simplest treatment of English. Note that extending the grammar to cover other constructions will still give us the right forms (for colloquial English).

Syntactically distinct pro-forms of the same semantic type are given disjoint indices in order to make the statement of the rules simpler. Note, for example, that this device ensures that the uniqueness condition on embedding rules (a syntactic condition) has the consequence that the lambda-abstraction in the translation will never bind the translation of a syntactically different expression. For example, although $\underline{one}_i$ and $\underline{such}_j$ are of the same semantic type a rule embedding for one will never "catch" a translation of the other.

KR-rules. Some of the peculiar looking rules are included to make sure we get a grammar as closely equivalent to PTQ as possible. Classical grammars did not treat in any detail some of the constructions included here and the theory was somewhat equivocal (at least in practice) as to whether the set of kernel sentences was to be finite or not. Our grammar would define a finite kernel language save for the inclusion of case (a) of KR5 (disjunction of NP's). This is necessary if we are to get the 'direct generation' reading of sentences like John seeks Mary or Bill.

The only substantive innovation is contained in KR3 (f) which provides a new translation for sentences like John is a fish: j*(^fish'). I have not provided any NP structure for the predicate nominal because it doesn't make any difference here (and I'm not too sure we need it in general since predicate nominals are notoriously un-NP-like). It seems to me likely that something like this treatment of be is required for adjectives anyway, as a number of writers have suggested (e.g. Lee, 1974). This rule is required to get the effects of common-noun quantification into the grammar (via one-embedding). It is perfectly possible to write a rule within our formalism that will assign a NP structure to the predicate nominal, namely this:

$$\text{If } \langle \text{CNP}, \alpha \rangle \in SD_K(G), \text{ then } \langle [_{Vip} \text{be } [_{NP} \text{a CNP}]], \alpha \rangle \in SD_K(G).$$

This shows that our formalism allows rules not statable in the form of phrase-structure rules although the effect of the rule could be achieved if the kernel rules were allowed to be context-sensitive (an option usually deemed necessary in classical grammars).

Transformations. The rules are stated in a detailed enough fashion that certain orders of application are excluded simply by the form of the rules. The following points are worth noting:

T1-T3 (the meaning changing singulary transformations) are stated in such a way that they can only apply to a simple sentence before it is embedded or conjoined (except for the case where T12 (conjunction reduction) has reduced a sentence to a form in which there is a single Pres). This ensures that we get the same relationships of scope for the sentential operators $\neg$, W, H and necessarily as in PTQ (with necessarily always outside the operators).

T5-6 (that- and to-embedding) are stated in such a way that embedding must take place before any of T1-T3 can apply to the matrix. This brings about for these rules an order of application postulated as correct by Fillmore (1963) and for which some evidence can be given. On the other hand T7 (relative clause) and T8 (one-embedding) as stated could be applied at any time subject only to the requirement (from the uniqueness condition) that if the second structure (the constituent) contains several like-indexed pro's, they must undergo T4 (pronominalization) before the rules can apply, and in the case of relative clauses T1-T3, and T5 and T6, cannot apply to a relative clause after it is embedded.

However, it is not the case that all and only the correct orders of application follow from the rules (I mean correct in the sense of getting a grammar equivalent to PTQ). For example, without some extrinsic ordering constraints it would be possible to form a future tense, then quantify in, then do negation, thus obtaining a translation for a sentence like the following not available from PTQ:

Every unicorn will not walk $\neg$[every unicorn'($\hat{x}$ [W[walk'(x)]])]

I will thus assume that the rules are ordered in blocks (as in Fillmore's 1963 theory) at least in such a way that the PST must apply one after the other (if applied) before a sentence is subject to quantification and that after quantification a sentence cannot undergo the PST again unless by virtue of its being embedded in a larger structure (where only T4 might affect it). I have already stated that the M-rules are extrinsically ordered to apply after all others (and must themselves apply in the order given).

3. Comparison of PTQ and G. In this section I will first discuss the relationship between the two grammars as far as their strong generative capacity (here: pairings of strings and translations) is concerned, then some general differences between the two models.

3.1 Strong equivalence. The language of G is the same as the members of P_t and P_T and the two grammars are strongly equivalent with the following exceptions:

1. L(G) contains no strings with indexed pro-forms, that is, L(G) is the set of English sentences and noun-phrases in P_t and P_T. This is because the indexed pro-forms are not in the T-terminal vocabulary of the grammar and hence by definition no strings containing them are either.

2. G does not assign readings to sentences or noun-phrases corresponding to vacuous applications of the rule S14, S15, S16 (quantification) in PTQ. This follows because the relevant rules are not functions from all members of the sets of expressions P_t, P_{CN}, and P_{IV} but only from the subset of these expressions which contain appropriate indexed pro-forms.

3. Similarly, L(G) does not contain any sentences with relative clauses containing no pronouns such as John loves a woman such that Bill is a fish. This is so for the reasons given under (2).

4. G makes available a reading for sentences like the following that is not available from PTQ:

Every woman does not love her (=herself)

By PTQ this can only come by quantifying in every woman (S14). Hence, there is no reading under which the negation operator is outside the scope of the subject NP, that is, no translation corresponding to the more natural English rendition Not every woman loves herself (cf. Partee, 1975; and for some discussion Bach, 1976, and below, Section 4.4).

5. The use of recursive properties of expressions in the M-rules makes it possible to correct two mistakes in PTQ. (i) My rules will get the right forms of verbs in conjoined verb phrases, and distinguish two structures in examples like these (See Bennett, 1974):

John walks and talks. (PTQ: John walks or talks)
John tries to walk and talk.
John tries to walk and talks.

(ii) My rules will distribute the correct case forms of pronouns in disjoined NP's where PTQ incorrectly gives phrases like see Mary or he, see Mary or he_3 (Barbara Partee pointed out this error to me).

6. The dual treatment of be given here provides alternative readings for some sentences of PTQ. Although this detail was included primarily to allow the grammar to capture the results of common-noun quantification by S15 without getting some unwanted results (see discussion of one-embedding below) it does, I believe, have some independent justification. Although most sentences with be in PTQ will, by meaning postulate (2) (p. 263), turn out to be equivalent to our translation, it seems to me that there is a sense of predicational sentences like John is a fish different from the one given by PTQ: 'there's a fish that John is identical with.' Notice the sentence John wants to be a fish and eat it which can arise by quantification into predicate nominal position in PTQ (and G). At best it seems to me that this sentence is peculiar and that its translation only fits one of the senses of the copular sentence John is a fish. Note further that PTQ (and G) allows a peculiar reading for John isn't a fish 'there's a fish that John isn't.' G allows the derivation of copular sentences which do not permit quantification into the predicate nominal position. It seems to me doubtful that if Montague had been a native speaker of almost any language but English he would have treated be as a transitive verb.

With these exceptions, then, G bears the following relationship to PTQ.

I. $\langle X, \varphi \rangle$ is a terminal structural description of G if and only if the debracketization of X, d(X), is in P_t or P_T and there is a derivation from PTQ that assigns the translation φ (or a logically equivalent one) to d(X).

Let's note first that there is no disambiguated language in this model, since we can have several translations for the same syntactic structure. We can, however, borrow the notion of a transformation-marker from classical transformational theory and use the T-markers of a sentence as a disambiguated language for showing equivalence.

Let's define a (generalized) T-marker as a tree rooted in a terminal structural description of either a sentence or a noun-phrase, whose leaves are structures gotten from the lexicon by using rule KR 0, and each subtree defined as follows: If a derivation uses a particular rule, then let the result of applying the rule be a tree whose root is the structural description corresponding to the <u>then</u> part of the rule, and its immediate branches the structural description(s) corresponding to the <u>if</u> part. For example, the T-marker for the simplest derivation of <u>John loves Mary</u> is this:

$$\langle\, [_S[_{NP}[_N\text{John}]]\ [_{ViP}\ [_{Vt}\text{loves}]\ [_{NP}[_N\text{Mary}]]]],\ j^*(^\wedge\text{love}(^\wedge m^*))\,\rangle$$

M-rules

$$\langle\, [_S[_{NP}[_N\text{John}]]\ [_{Aux}\text{Pres}]\ [_{ViP}[_{Vt}\text{love}]\ [_{NP}[_N\text{Mary}]]]],\ j^*(^\wedge\text{love}(^\wedge m^*))\,\rangle$$

$$\langle\, [_S[_{NP}[_N\text{John}]]\ \text{Aux}\ [_{ViP}[_{Vt}\text{love}]\ [_{NP}[_N\text{Mary}]]]],\ {}^*j(^\wedge\text{love}(^\wedge m^*))\,\rangle$$

$$\langle\, [_{NP}[_N\text{John}]],\ j^*\,\rangle \qquad \langle\, [_{ViP}[_{Vt}\text{love}]\ [_{NP}[_N\text{Mary}]]],\ \text{love}'\ (^\wedge m^*)\,\rangle$$

$$\langle\, [_N\text{John}],\ j^*\,\rangle \qquad \langle\, [_{Vt}\text{love}]\ \text{love}'\,\rangle\ \langle\, [_{NP}[_N\text{Mary}]],\ m^*\,\rangle$$

$$\langle\, [_N\text{Mary}],\ m^*\,\rangle$$

(I call this structure a 'generalized' T-marker because it differs from the classical notion, which was a tree whose leaves were K-terminal structures and which included no information about the translation.)

The proof of (I) goes like this: It can be shown that every T-marker determines a unique terminal structural description. We define a mapping from T-markers into analysis trees which takes place in several stages:

(i) Given a T-marker <u>t</u>, we construct a T-marker <u>t'</u> which is just like <u>t</u> except that in place of branches rooted by S-structures and immediately dominated by structures that result from embedding for $\underline{\text{one}}_i$ or $\underline{\text{to}}_i$ we have branches containing CNP or ViP structures in place of the original S-structures. This T-marker is a T-marker that would have resulted from a grammar which embedded CNP's and ViP's and which contained quantification rules for getting NP's into those structures and which included such structures in the extended set of structural descriptions SD(G)). For example, a branch schematically represented as (a) will be mapped into a branch of the form (b):

a. $\langle$ [John Pres Vto to ViP], λq_1 $(\hat{x}_4\ [\alpha[\hat{x}_j\ \ ])$ $\rangle$

$\langle$ [John Pres Vto to to_1], $\varphi\rangle$ $\langle$ [he_4 Pres ViP], $\alpha(\hat{x}_j\psi)$ $\rangle$

$\langle$ [John], $\beta\rangle\langle$ [V to to to'], $\sigma\rangle$ $\langle$ NP, $\alpha\rangle$ $\langle$ he_4 Pres [$_{ViP}X_1$ pro_j X_2], $\psi\rangle$

b. $\langle$ [John Pres Vto to ViP], λq_1 $\varphi(\hat{x}_4[\ [\alpha\ [\hat{x}_j\ \psi])\rangle$

$\langle$ [John Pres Vto to to_1], $\varphi\rangle$ $\langle$ ViP, $\alpha(\hat{x}_j\psi))\rangle$

$\langle$ [John], $\beta\rangle$ $\langle$ [Vto to to_1], $\sigma\rangle$ $\langle$ NP, $\alpha\rangle$ $\langle$ [$_{ViP}X_1$ $pro_j X_2$], $\psi\rangle$

(ii) We then define a second mapping which converts the results of (i) into a new set of T-markers which are just like the old ones, but wherever we have a subtree dominated by a structure resulting from embedding for $\underline{that}_i$, $\underline{to}_i$, $\underline{such}_i$ or $\underline{one}_i$ we form a branch which results from replacing the syntactic $\underline{\text{pro-form}}_i$ by the embedded structure throughout the branch corresponding to the matrix and changing the new leaves involving Vthat, AvVS, Vto, etc. to branches of the form

ViP

Vthat that S etc.

and deleting the branch for the embedded structures. To continue the above example, this mapping will give the tree (c) (leaving out the translations).

[John Pres Vto to Vip]

[John Pres Vto to Vip]

John Vto to Vip

Vto to ViP

NP [$_{ViP}X_1$ pro_j X_2]

We now have a T-marker which would result from a grammar with no embedding rules except quantification (into S, CNP, ViP) (such a grammar, which is not exactly a KT-grammar, is given in Appendix B). Moreover, the T-marker is a refinement of the analysis tree from PTQ that the mapping will produce.

The chief complication in step (ii) involves making sure that a free variable pronoun occurring in the embedded structure does not get 'caught' accidentally by a quantification which has occurred in the matrix structure. To prevent this (and we can always tell if it is going to happen) we replace all $\underline{pro_i}$'s which have been bound in the matrix by $\underline{pro_n}$'s where $\underline{n}$ is the least integer = $\underline{i}$ (modulo 6) such that $\underline{pro_n}$ occurs in neither the matrix structure nor the embedded structure.

(iii) It is now possible to map the result of (ii) into an analysis tree. The mapping is straightforward (let the map of each node expression by the debracketization of the first member of the pair, followed by the appropriate number for Montague's syntactic functions, which is completely recoverable from the form of the immediate branches). For the sample T-marker given at the beginning of this discussion we would have

```
                John loves Mary
                      |
                      |    M-rules
                      |
            John [Pres] love Mary
                      |
          John Aux  love Mary, 4
         /                     \
    John                   love Mary, 5
                          /           \
                      love             Mary
```

This leaves us with some single-branched subtrees resulting from pronominalization and the rules for tense and negation. The former will be of the form:

$$[\ X_1\ pro_i\ Y\ pro\ \ldots\ pro\ Z\ ]$$

$$[\ X_1\ pro_i\ Y\ pro_i\ \ldots\ pro_i\ Z\ ]$$

and occur immediately dominated by a node resulting either from quantification or relative clause formation. For these, we erase the upper structures in the single branched subtree.

Results of T1-T3 will be single branched structures like these

```
NP Pres Not ViP      NP will ViP        NP will not ViP
       |                  |                   |
NP Pres ViP, 4       NP Pres ViP, 4     NP will ViP
                                              |
                                        NP Pres ViP, 4
```

NP Pres have ViP	NP Pres have not ViP
|	|
NP Pres ViP, 4	NP Pres have ViP
	|
	NP Pres ViP, 4

These we replace by the debracketization of the result by applying M-10 to the first member of the top structure and write down the numbers 11, 12, 13, 14, and 15 respectively to the right. Finally, we delete all nodes which have no numbers with them except for the leaves (the basic expressions) and connect the lines thus lopped.

The result of this mapping will be an analysis tree and PTQ will assign to it a translation which is logically equivalent to the translation given by G to the original T-marker (with the exceptions noted above). This will be so because the embedding rules for $\underline{such}_i$, $\underline{that}_i$, $\underline{one}_i$, $\underline{to}_i$ all involve lambda abstraction over a variable for senses of expressions applied to the sense of the expression translating the embedded structure.

The conversion c from M-trees to T-markers is an analogous process which takes subtrees of an analysis tree involving rules S3, S8, S10, S12, S15, S16 and constructs appropriate branches from (let us say) the next higher node which involves the rules that derive members of P_t and replaces the part of the original string corresponding to the structure embedded by an appropriate syntactic variable. Finally, labeled brackets are added to the string in a way that is completely determinate.

One difference between the two grammars lies in the build-up of relative clauses. Although there will always be a derivation from G of a sentence containing several relative clauses in one common noun phrase that corresponds to the 'bottom-to-top' build-up of PTQ, the grammar will always allow all permutations of the relative clauses to be derived for the same translation (but because of the logical equivalence of p & q and q & p the results will all be logically equivalent). If there is some semantic or implicative difference between these different phrases (as I believe there is) then this result must be considered a defect in G. I can think of various brute-force solutions but no very good general principle that might be adopted in the general theory. (E.g. we can build into the rule a condition to ensure embedding from left to right.)

3.2 Montague grammars, KT-grammars, C-grammars and I-grammars. In this section I will make some general remarks about the general frameworks illustrated by PTQ, G and the two 'standard'-derived transformational grammars given in Cooper and Parsons (1976).[3]

(i) In common with other transformational frameworks, the model explored here incorporates the hypothesis that significant generalizations about natural languages can best be captured by describing an abstract language that is mapped into the 'real' language by certain obligatory rules. This feature has been exploited here, rather minimally, in the treatment of tense and the use of late rules of a primarily morphological nature. In Section 4.4 I will take up a suggestion for treating negation in a more abstract way.

Montague's general theory as given in 'Universal Grammar' allows for this possibility (Partee, forthcoming) and apparently one could mimic G with a Montague grammar in which my M-rules are part of the relation R and the language of T-markers is the disambiguated language which is the domain of R. On the other hand, the model violates the wellformedness constraint of Partee (forthcoming) but in a fairly limited way. Motivation for the particulars of the grammar must largely come from extending the grammar to accomodate richer fragments of English. For example, the treatment of the Auxiliary in G (or the fuller treatment traditional since Chomsky, 1957) can be motivated by the success with which it dovetails with a treatment of questions and other sentences involving inversion.

However, the particularization of Montague's general theory illustrated by PTQ has the feature that all morphological details are built into the syntactic functions (and thus the fragment meets Partee's constraint except for the presence of the abstract forms $\underline{he}_i$ and $\underline{him}_i$). One could motivate the choice of objective base forms for pronouns on the basis of predictive power even within PTQ. If one extended the fragment to include double object verbs like give, topicalized and clefted sentences etc., the grammar would then make the right predictions (give it to her; give him it; Him, I saw yesterday, etc.), but the generalizations about the form of English pronouns given in my M-rules would still be splintered across 6 rules (S4 and the five functions of S17). Additions of other tenses to PTQ would carry along no predictions. Put another way, there would be no reason even with this modified PTQ to expect that English should be the way that it is rather than, say, requiring the distribution of pronouns in the following strings:

he walks
him will walk
he will not walk
he has walked
him has not walked
his does not walk

Questions of this sort have been discussed in the transformational linguistic literature largely under the heading 'explanatory adequacy.' They have not, so far, been treated in discussions of Montague grammar.[4]

(ii) Classical transformational grammar accorded special status to the sentence and implicitly carried with it the claim that a language could best be described as a system in which rules of a structure-dependent sort built up complicated structures from the structures underlying simple sentences. Many logicians and philosophers since Frege have accepted the idea that the logical structures of languages (artificial or natural) can best be understood by taking open sentences containing variable elements as well as closed sentences to be necessary building blocks. Transformational linguists (McCawley, 1971; Bach, 1968) originally incorporated Frege's insight at a time when the theory had developed the idea (not in the classical theory) that every sentence had some underlying 'deep structure' of a very abstract sort that was more or less directly related to a representation of the logical form of the sentence. More recently, Chomsky (1975, 1976) and others have been incorporating the insight by constructing rules which go from an enriched 'surface structure' to a representation of logical form (Cooper and Parson's C-grammar is like the formed development, their I-grammar like the latter, but without 'traces'). Montague grammar and classical transformational grammar both incorporate the notion that part of how we represent the meaning of a sentence is by means of a representation of the order in which it is derived rather than by means of some structural configuration. Montague used this device in PTQ to represent quantificational scope differences (this derives from Frege also; see Dummett, 1973). Moreover, his treatment of term-phrases as representing sets of properties allowed him to treat English sentences in a way that did justice to the integrity of the noun-phrase as a basic syntactic and semantic element in the construction of sentences. But transformational theory did not use this device to account for quantificational facts. The extension of the classical theory illustrated here unites these two historically quite different traditions, one derived from Frege, the other ultimately from historical linguistics (see Chomsky, 1955 [1975]: Introduction).

The extension of classical theory made here singles out noun-phrases and sentences as having a special status for the purpose of building complex structures and for representing complex logical forms. Barbara Partee has pointed out to me that these two categories correspond to the two categories that Frege considered to be the only complete expressions of a language: names and sentences. There are a number of respects in which these two categories seem to have a special status. As far as I know, every natural language does have these two syntactic categories, and it is hard to imagine a natural language that did not. (An example of an unnatural language that doesn't is given in Quine 1966: Ch. 23). The two categories are related in an intimate way to two necessary uses of language: to refer to objects and to make statements that can be true or false. Moreover, I believe that although native speakers of language do not in general have very firm intuitions about the syntactic structure of sentences in any very fine-grained way, they do have clear intuitions about what pieces of sentences are noun-phrases and what sentences and noun-phrases are well-formed, and this is so because these are the two categories that can be used alone in a relatively context-free way as complete utterances.

The use of sentence transforms to form VP-complements (required by the general theory) represents an interesting empirical hypothesis. Analyses like that of PTQ (with direct recursion on VP) have been developed in post-Aspects versions of transformational grammar (Brame, 1976). Avery Andrews (1976) has argued that in Icelandic, VP complements shoulf be derived from sentences and suggested that general linguistic theory should be constrained in such a way as to require this result. It should be made clear that this hypothesis predicts that no language will be found in which evidence can be found against such an analysis and is quite consistent with the fact (if it is a fact) that languages exist for which there is no direct evidence for a derivation of VP's from sentences (English may be such a language).

(iii) Standard transformational theories and their derivatives have made the assumption that the relation between syntax and semantics is to be made via a set of rules relating syntactic structures at some level to representations of semantic structure. Both of the grammars in Cooper and Parson's use this configurational approach. The fragment given here shows that the older model can be used along with a conception of the relation that is based on relationships between syntactic rules and translation rules instead.

(iv) Finally, it can be shown that the classical theory is mathematically less powerful than the standard theory and its derivatives. Thus there is some independent reason for choosing to work within this framework rather than the more recent versions of the theory. What I hope to have accomplished so far in this paper is to have shown that an interesting and intuitively satisfying theory of syntax can readily be extended to incorporate a rigorous and beautiful semantic theory.

4. Some extensions of G.

4.1 Variables. Suppose we add to G the rule (in the PST) (I'll give rules schematically in this section):

T105 Pro-form desubscripting

$$X\ \text{pro-form}_i\ Y\ \text{pro-form}_i\ Z \rightarrow X\ \text{pro-form}_i\ Y\ \text{pro-form}\ Z$$

where pro-form_i stands for $${}_1\left\{\begin{matrix} \text{that}_i \\ \text{to}_i \\ \text{such}_i \\ \text{one}_i \end{matrix}\right\}_1$$

and pro-form stands for $${}_1\left\{\begin{matrix} \left\{\begin{matrix} \text{that}_i & \text{if Y ends on } \underline{\text{necessarily}} \\ \text{it} & \text{otherwise} \end{matrix}\right\} \\ \underline{e} \\ \underline{e} \\ \text{one} \end{matrix}\right\}_1$$

and add to the M-rules the rule a one $\Rightarrow$ one.

Suppose further that we change the rules for that-embedding and to-embedding to a more general form by adding X_1 and X_2 to the if-part of each rule inside the end brackets for the matrix and changing $[_{Aux}$ Pres$]$ to Aux.

Now these rules will be able to apply to any level of embedding. G will now generate sentences like these:

1. Mary asserts that Bill is a fish and John believes it.

2. Mary wants to date a unicorn or a fish and Bill $\left\{\begin{array}{l}\text{does not}\\ \text{wants to}\\ \text{does not want to}\end{array}\right.$

3. Bill has seen a woman such that she walks and Mary has seen a man (=who walks)

4. John knows a woman such that she is a fish and Bill has seen one such that she walks.

(Actually, I'm cheating on the last example, since one_i is undefined for gender. If we let it be any gender, then the grammar will not only generate 4 but a similar sentence with the second she replaced by he or it.)

We could also consider the possibility of letting the proform of such_i be such, adding rules to get such a CNP from a CNP such, deleting it elsewhere and get sentences like (5).

5. John knows a man such that he walks and Bill knows such a woman.

(But Muffy Siegel and Greg Carlson have led me to be very suspicious of my treatment of such.)

I don't want to enter into an extended discussion of such examples in this paper (for some arguments for this treatment of VP's see Bach, 1976). All I want to suggest, primarily to transformational linguists, is that the model used here shares with Montague grammar the possibility of treating various kinds of anaphora as direct reflexes of syntactic variables. Transformational grammarians have argued endlessly about whether such phenomena (of one or another kind) should be handled by deletion rules or interpretive rules. There are arguments against each position. Deletion rules have problems involving so-called sloppy identity. Interpretive rules have problems because various extra material gets left around (e.g. Mary was seen by the police but Bill wasn't En by). Maybe both positions are wrong.

4.2 Gender. I'm not at all sure that the syntactic treatment of gender given in G or any syntactic treatment of gender in English can be defended (cf. Chapter 3 of Cooper, 1975). The necessity to get gender straight in G requires an extremely cumbersome formulation of various rules (notably to embedding). I understand that Lauri Kartunnen wants to treat English gender as part of conventional implicature. If this were done we could simplify our grammar greatly. The rules for to-embedding and relative clauses could look like this:

$$\left.\begin{array}{ll} \text{T6}' & X_1 \ \text{to}_i \ Y \\ & \text{pro}_i \ \text{Pres ViP} \end{array}\right\} \Rightarrow X_1 \ \text{ViP} \ Y$$

$$\left.\begin{array}{ll} \text{T7}' & X_1 \ \text{such}_i \ X_2 \\ & X_3 \ \text{pro}_j \ X_4 \end{array}\right\} \Rightarrow X_1 \ \text{such that} \ X_3 \ \text{pro} \ X_4 \ X_2$$

Then sentences like (6) would be syntactically and semantically O.K. but convey the wrong meaning:

6. The woman such that he walks.

If we take this approach then we must face the question of what gets "lambda"-ed in if we combine this treatment with the use of syntactic variables for VP's. Consider the sentence:

7. John wants to see himself but Mary doesn't want to.

There is certainly a reading of this sentence where we understand that Mary doesn't want to see <u>her</u>self. I don't think that this sentence in any way implicates that Mary <u>is</u> a man.

4.3 <u>Extending tenses</u>. I have treated tenses the way I have mainly to show how meaning-changing transformations can be used in the theory, and I'm not at all sure that it's the right way to do the Auxiliary. But to mimic the classical treatment of the Auxiliary requires either dropping the rule-by-rule treatment of semantics (and using something like the configurational rules of Cooper and Parsons, 1976) or removing the Auxiliary from the main body of a sentence as in the UCLA grammar adaptation of Fillmore's case grammar (Stockwell et e., 1973).

We can, however, make fairly minimal changes to the grammar to allow both <u>will</u> and <u>have</u> to occur in the same sentence. If T1 is changed by adding a variable W (for wflb's) after Pres in the Aux in parentheses (since <u>e</u> is not a wflb) the grammar will generate for every sentence it used to generate a sentence in which the Aux contains <u>will have</u> and the translation for the sentence will be $W(H(\phi))$. (A few changes are necessary in the M rules.) It is also fairly easy to change the rules for <u>to</u>-embedding to get things like <u>John wishes to have walked</u>.

4.4 <u>Negation</u>. As pointed out in Bach (1976), it is possible to build into the grammar an analysis of negation that was developed by Klima (1964). Suppose we replace T3 by the rules:

Negation

NP Aux ViP $\Rightarrow$ Neg NP Aux ViP

Neg-placement (obligatory)

Neg NP Aux ViP   NP Aux not ViP where NP $\neq$ $[_{NP}\left\{\begin{matrix}\text{a}\\ \text{every}\end{matrix}\right\}$ CNP]

and add these M-rules:

Neg $[_{NP}$a CNP] $\rightarrow$ $[_{NP}$no CNP]

Neg $\rightarrow$ not

Now L(G) will include these sentences with the order of Negation and quantification as indicated on the right:

1. A woman does not walk. $\bigvee x \neg \varphi$
2. No woman walks. $\neg \bigvee x\ \varphi$
3. Every woman does not walk. $\bigwedge x \neg \varphi$
4. Not every woman walks. $\neg \bigwedge x\ \varphi$

This follows because Neg-placement cannot apply if the subject of the sentence is a woman or every woman. Hence quantification must apply after Negation for those sentences in which the subject with a or every is followed by the negative.[5]

APPENDIX A: Examples of derivations from G.

John seeks a unicorn is in L(G). This is shown by the following proof for a structural description given by our grammar.

(a) $\langle [_S[_{NP}[_N\text{John}]]\ [_{Aux}\text{Pres}]\ [_{ViP}[_{Vt}\text{seek}]\ [_{NP}\text{a}\ [_{CNP}[_{CN}\text{unicorn}]]]]],$

$\hat{P}\ P\{\hat{}j\}\ (\hat{}\text{seek}'\ (\hat{P}\ \bigvee x\ [\text{unicorn}'(x)\ \&\ P\ \{x\}]))\rangle \in SD(G)$

Proof.

(i) $\langle [_{CNP}\ [_{CN}\text{unicorn}]],\ \text{unicorn}'\rangle \in SD_K(G)$ by Lexicon, KR 0

(ii) $\langle [_{NP}\text{a}\ [_{CNP}[_{CN}\text{unicorn}]]],\ \hat{P} \bigvee x[\text{unicorn}'(x)\ \&\ P\{x\}]\rangle \in SD_K(G)$

by KR5(e)

(iii) $\langle [_{ViP}[_{Vt}\text{seek}]\ [_{NP}\text{a}\ [_{CNP}[_{CN}\text{unicorn}]]]],\ \text{seek}'(\hat{}\hat{P}\ \bigvee x[\text{unicorn}'$

$(x)\ \&\ P\ \{x\})\rangle \in SD_K(G)$ by Lexicon, KR 0, and KR3(c)

(iv) (a) $\in SD_K(G)$ and hence $\in SD(G)$ by Lexicon, KR 0, KR2,

KR5 and T0

By M2 and M4(d) seek has the property ThirdPersonSingular Present, by M8. Pres is deleted (and empty brackets erased) giving us in place of the first member of (a) the structure (after adding "s" to seek)

$$[_S[_{NP}[_N John]]\ [_{ViP}[_{Vt} seeks]\ [_{NP} a\ [_{CNP}[_{CN} unicorn]]]]]$$

The debracketization of this is the desired sentence.

As the reader can check the following pairs are also in SD(G) all with the same first member as (a) (call it X):

(b) $\langle X, \hat{P} \bigvee x[\text{unicorn}(x)\ \&\ P\{x\}]\ (\hat{x}_4[\ j^*(\hat{}\text{seek}(\hat{}\hat{Q}\ Q\{x_4\}))])\rangle$

Pres

(from John seeks it_4 + a unicorn)

(c) $\langle X, j^*(\hat{x}_2[\hat{P}\ P\{x_2\}\ \ (\hat{}\text{seek}\ (\hat{}\hat{Q}\ \bigvee x[\text{unicorn}'(x)\ \&\ Q\ \{x\}]))])\rangle$

(he_2 Pres seek a unicorn + John)

(d)
(e) arising from quantifying in both NP's in either order.

(f) $\langle X, j^*\ (\hat{x}_2\ [\hat{P} \bigvee y\ [\text{unicorn}'(y)\ \&\ P\ \{y\}]\ (\hat{x}_4[\hat{Q}\ Q\{x_2\}(\hat{}\text{seek}'$

$(\hat{}\hat{P}_1 P_1\{x_4\}))])])\rangle$

(actually a more complicated expression equivalent to this)
(John pres to_1 + (a unicorn + (he_2 Pres seek it_4)))

(g) a derivation similar to (f) but with John quantified in and a matrix he_8Pres to_1

(h) our grammar gives a number of derivations not matched in PTQ (but with translations logically equivalent to ones given by PTQ, for example:

John Pres seek a one_5 $\langle\ j^*(\hat{}\text{seek}'(\hat{}\hat{P} \bigvee x\ [\hat{x}_2[q_5\{x_2\}](x)$
$\&\ P\ \{x\}]))\rangle$

it_4 Pres be a unicorn $\hat{P}\ P\ \{x_4\}$ $(\hat{}\text{unicorn})$ (via KR 3(f))

John seeks a unicorn (by one-embedding and M-rules)

λq_5 [the expression above] $(\hat{x}_4\ (\text{unicorn}'(x_4))$

which gives by conversion

$$j^*(\ \text{seek}'(\hat{}\hat{P}\ \surd x[\hat{x}_2[\hat{x}_4[\text{unicorn}(x_4)]\ (x_2)]\ (x)\ \&\ P\{x\}\ ]$$

which is logically equivalent to the second member of (a).

APPENDIX B: A T-grammar that is exactly equivalent to PTQ and almost exactly like it.

By a T-grammar I mean a grammar of the same general form as a KT-grammar except that it defines directly a set of structural descriptions by a set of unordered rules of the If-then form and maps them by M-rules into sets of expressions (of all categories) of the language.

R0 (=KR0)

R1 XP → X for X = Vi, N, AvV, CN

R2 $\text{NP} \rightarrow \left\{\begin{array}{l}\text{every}\\ \text{the}\\ \text{a}\end{array}\right\}\ \text{CNP}$

R3 $\left.\begin{array}{l}\text{CNP}\\ [_S X_1 (\text{pro}_n)\ X_2\ ]\end{array}\right\} \rightarrow [_{\text{CNP}}\text{CNP such that } [_S\ X_1, (\text{pro}) X_2]](\text{CNP},\ \text{pro}_n$

agree in Gender)

R4 $\text{S} \rightarrow \text{NP}\ [_{\text{Aux}}\text{Pres}]\ \text{ViP}$

R5 ViP → Vt NP

R6 AvVP → Pres NP

R7 ViP → Vthat that S

R8 ViP → Vto to ViP

R9 S → AvS S

R10 ViP → ViP AvVp

R11 $\text{S} \rightarrow \text{S}\ \left\{\begin{array}{l}\text{and}\\ \text{or}\end{array}\right\}\ \text{S}$

R12 $\text{VIP} \rightarrow \text{ViP}\ \left\{\begin{array}{l}\text{and}\\ \text{or}\end{array}\right\}\ \text{ViP}$

R13 NP → NP or NP

R14 $\left.\begin{array}{l}[_S\ X_1(\text{pro}_n)\ X_2\ ]\\ \text{NP}\end{array}\right\} \Rightarrow [_S X_1(\text{NP})X_2\ ]$ (NP, pro_n agree in Gender)

R15
R16 Same as 14 but with CNP and ViP for S

R17
through
R20 = T1-T4 in G

M1-
M10 T-terminal vocabulary to include $\underline{him_i}$, $\underline{he_i}$

The second member of each pair is gotten directly from Montague's translation rules for R0 - R16.

Except for the results of R17-20 a T-marker from this grammar will be an exact match for an analysis tree from PTQ and the translation will be identical, including the results of vacuous quantification and relative clause formation. R3 and R14-16 are to be read as follows:

case (a): the rule with parentheses removed.
case (b): the rule with parenthesized material removed.

The mapping to and from M-trees is trivial.

FOOTNOTES

[1]By 'local grammaticality' I mean the property of a grammar such that every sentence-structure embedded into a matrix must underlie some well-formed sentence of the language, cf. Bach, forthcoming, where I show that this property, for classical grammars, makes them no more powerful than the 'local-filtering' transformational grammars of Peters and Ritchie, 1973b. For the 'wellformedness constraint' see Partee, forthcoming.

[2]The KR rules 1-7 are given in the form of phrase structure rules and partially similar rules are collapsed by curly brackets. Except for KR7 they can be read 'backwards' as parts of a recursive definition of the set of left members of elements in the set of kernel structural descriptions. The expressions on the right give the translation of the resultant expression and "A" means "the translation of $[_A X]$." If two categories of the same kind appear on the right of a rule (KR5a) they are distinguished in an obvious way by subscripts.

[3]I disclaim any competence about "Universal Grammar."

[4]But see Partee, forthcoming, for some steps in this direction.

[5]My understanding of Montague's work has been aided immensely by discussions with Robin Cooper, Muffy Siegel, and especially Terry Parsons and Barbara Partee. An early version of this paper was presented in a

seminar conducted by Parsons and Partee and I have benefited greatly from class discussions and from personal communications, in particular, I'd like to thank Barbara Partee for her generosity in letting me bug her endlessly and helping me debug the paper. Naturally, all remaining bugs are my own. Completion of this paper was made possible by the Center for Advanced Studies in the Behavioral Sciences and a grant from the National Endowment for the Humanities.

BIBLIOGRAPHY

Andrews, Avery. 1976. The VP Complement analysis in Modern Icelandic. Recherches Linguistiques a Montreal 6:1-21 (NELS 6).

Bach, Emmon. 1968. Nouns and Noun Phrases. In E. Bach & R. T. Harms, eds., Universals in Linguistic Theory (New York).

__________. 1976. An extension of classical transformational grammar. In: Problems in linguistic metatheory (Proceedings of the 1976 conference at Michigan State University).

__________. Forthcoming. "The position of embedding transformations in a grammar" revisited. To appear in a volume of papers from the International Summer School of Computational Linguistics (Pisa), A. Zampolli, ed.

Bennett, Michael. 1974. Some extensions of a Montague fragment of English. Unpublished doctoral dissertation, University of California, Los Angeles.

Brame, Michael. 1976. Conjectures and refutations in syntax and semantics. New York.

Chomsky, Noam. 1957. Syntactic structures. The Hague.

__________. 1965. Aspects of the theory of syntax. Cambridge, Mass. (M.I.T. Press.)

__________. 1975. [1955]. The logical structure of linguistic theory. New York.

__________. 1975. Reflections on language. New York.

__________. 1976. Conditions on rules of grammar. Linguistic Analysis. 4.303-351.

Cooper, Robin. 1975. Montague's semantic theory and transformational syntax. Unpublished Ph.D. dissertation, University of Massachusetts, Amherst.

Cooper, Robin and Terence Parsons. 1976. Montague grammar, generative semantics, and interpretive semantics. In: B. Partee, ed., Montague Grammar (New York).

Dummett, Michael. 1973. Frege: Philosophy of Language. New York.

Fillmore, Charles J. 1963. The position of embedding transformations in a grammar. Word 19: 208-231.

Katz, Jerrold J. and Jerry A. Fodor. 1963. The structure of a semantic theory. Language 39:170-210.

Katz, Jerrold J. and Paul M. Postal. 1964. An integrated theory of linguistic descriptions. Cambridge, Mass.

Klima, Edward S. 1964. Negation in English. In Jerry A. Fodor and Jerrold J. Katz, eds., The Structure of Language (Englewood Cliffs: Prentice-Hall).

Lee, Kiyong. 1974. The treatment of some English constructions in Montague grammar. Unpublished doctoral dissertation. University of Texas, Austin.

McCawley, James D. 1971. Where do noun phrases come from. In: Danny D. Steinberg and Leon A. Jakobits, eds., Semantics (Cambridge).

Montague, Richard. 1973. The proper treatment of quantification in ordinary English. In: Montague, 1974.

__________. 1974. Formal Philosophy. Ed. by Richmond Thomason. New Haven.

Partee, Barbara H. 1975. Montague grammar and transformational grammar. Linguistic Inquiry

__________. forthcoming. Montague grammar and the well-formedness constraint. To appear in the proceedings of Third Groningen Round Table (1976) ed. by Frank Heny and H. Schnelle.

Peters, Stanley and R. W. Ritchie. 1973a. On the generative power of transformational grammars. Information Sciences 6:49-93.

__________. 1973b. Nonfiltering and local-filtering transformational grammars. In: K. J. Hintikka, J.M.E. Moravcsik, and P. Suppes, eds., Approaches to Natural Language (Dordrecht.)

Quine, W.V. 1966. Selected logic papers. New York.

Smullyan, Raymond M. 1961. Theory of formal systems. Princeton, N.J. (Annals of Mathematics Studies 47)

Stockwell, Robert P., Paul Schachter, and Barbara Hall Partee. 1973. The major syntactic structures of English. New York.