

Current Grammar

1. Organization

Our grammar has several types of rules, which are organized roughly as in (1):

(1) Phrase Structure Rules \Rightarrow **Deep Structure** \Leftarrow Lexicon

Transformations
Form Rules
 \Downarrow
Surface Structure

DEEP STRUCTURE is the representation that is the result of PHRASE STRUCTURE RULES and LEXICAL INSERTION. TRANSFORMATIONS apply, one by one, to derive new structures. The FORM RULES also apply (at some point in the derivation) to add morphological features to some of the elements. The structure that is the result of applying all the applicable transformations and the form rules is called SURFACE STRUCTURE.

DEEP STRUCTURE interfaces with the Semantic Component. Thus, transformations do not change meaning. SURFACE STRUCTURE is the input to the phonology; hence, everything needs to be in the right order and all necessary morphological information must be in place by Surface Structure.

Remember that Deep Structures do not always correspond to grammatical sentences - sometimes obligatory transformations and/or form rules need to apply. Also, Surface structures do not have to conform to the Phrase Structure Rules - the Transformation may move things around, etc., in a way that would not have been directly generated by the Phrase Structure Rules.

2. Phrase Structure Rules

- (2) S → {NP, CP} VP
NP → {(DP) (AP) N (PP), pn, Name}
DP → {D, NP Poss}
VP → {(V_m) (V_{have}) (V_{b-A}) V (NP) (AP) (PP) (CP)
or- VP {PP, Adv}
PP → P NP
AP → (Int) A
CP → C S

3. Lexical Entries

Lexical entries contain everything that is idiosyncratic to particular lexical items. This includes the phonological shape (which we represent with the standard English spelling), idiosyncratic morphological information (e.g. irregular verb forms, plurals, etc.), the syntactic category (N, V, P, etc.), subcategorization, and lexical semantics (what the word means). We are particularly interested one aspect of the a verb's meaning - the list of participants (SEMANTIC ROLES) entailed to occur with any use of the verb. This is the 'who did what to whom' part of the semantics. For example, any event denoted with the verb *eat* will entail an 'eater' and an 'eatee':

- (3) *eat*, V, [__ (NP)]
(eater, eatee)

Note that *eat* subcategorizes for an optional NP. Nevertheless, any sentence that uses *eat* will entail an eatee, even if it is not expressed syntactically. Also note that *eat* subcategorizes for one phrase (NP), but there are two semantic roles in the argument structure. This is because one of its semantic roles is assigned to the subject (and verbs do not subcategorize for subjects - they can only subcategorize for sisters). By 'subject' I mean the NP that is the daughter of the S-node.

In general, verbs have one more semantic role than the number of subcategorized arguments. Here are some possibilities:

- (4) a. *faint*, V, [__]
(fainter)
b. *see*, V, [__ NP]
(seer, thing)
c. *fall*, V, [__ (PP)]

(thing, location)

d. *send*, V, [__ NP (PP)]
(sender, thing, location)

e. *put*, V, [__ NP PP]
(putter, thing, location)

In order to hook up semantic roles with syntactic structures, we assume the following linking rules:

(5) LINKING RULES

- a. Associate the first semantic role with the subject of the sentence.
- b. Associate any remaining semantic roles with the verb's (subcategorized) sisters, left to right.

These linking rules apply as part of the interface between Deep Structure and the semantic interpretation. This establishes 'who did what to whom' in order to feed into the semantic component. Transformations may move things around, delete things, etc., but the 'who did what to whom' will have already been established.

4. Transformations and Form Rules

Transformations alter phrase structure; that is, they delete things, insert things, and move things around. The result is a new phrase structure. Again, note that the output of a transformation does not need to abide by the phrase structure rules.

Transformations consist of a STRUCTURAL DESCRIPTION (SD), which identifies what kinds of structure a transformation can apply to, and a STRUCTURAL CHANGE (SC), which says how the phrase structure is to be changed.

(6)PASSIVE: (optional)

	NP	X	V	{NP, CP}	Y
SD:	1	2	3	4	5
SC:	4	2	V _{b-B} +3	∅	5 + [PP <i>by</i> 1]

This transformation applies to any sentence that begins with an NP, perhaps followed by some auxiliaries, then a verb and an immediately following NP (or CP). It moves the NP (or CP) from its position after the verb to the subject position, it adds the auxiliary V_{b-B} before the verb, and builds a PP that contains the preposition *by* and the NP that was the old subject. This new PP becomes a sister to the last element in Y (indicated by '+')

Transformations can be optional or obligatory.

Form rules (also known as morpho-syntactic rules) express generalizations about the form that a category has (= the way it is pronounced) depending on where it is located in constituent structure. Among the phenomena handled by these rules are subject-verb agreement, and verb forms. Form rules add a morphological feature or features to a category depending on a category that is immediately adjacent to it. These rules have one of the forms shown below:

- (7) a. $X \rightarrow X_{[A]} / _ _ Y$
 b. $X \rightarrow X_{[A]} / Y _ _$

Later, the morphology interprets the features as instructions for how (the words in) this category should be pronounced. Form rules are obligatory. Form rules cannot alter phrase structure; they cannot delete or add structure. They can only add morphological features.

5. The Organization of Transformations and Form Rules

The transformations and form rules are arranged in an ordered list. Once a Deep Structure is created, we simply go down the list and see which rules can apply. If a transformation can apply then we apply it (if it is optional we may apply it or not), creating a new structure. Then we continue down the list. The Form Rules come somewhere in the middle of the list. They all apply as a block. Then we continue down the list of remaining transformations until we reach the end. The result is a Surface Structure. Any Surface Structure our grammar can produce is predicted to correspond to a grammatical sentence. Anything that our grammar does not generate is predicted to be ungrammatical.

Since our phrase structure rules create Deep Structures with embedded clauses, we have a possibility that there may be more than one S-node in a structure. From the Deep Structure that underlies (8a), the following sentences may emerge:

- (8) a. Max believed that Pat wrote letters.
 b. Max believed that letters were written by Pat.
 c. That Pat wrote letters was believed by Max.
 d. That letters were written by Pat was proven by max.

- (8a): *no PASSIVE anywhere*
 (8b): *PASSIVE in the embedded clause*
 (8c): *PASSIVE in the main clause*
 (8d): *PASSIVE in the embedded and main clauses*

Therefore, we will assume that transformation may apply in both embedded and main clauses. The way this is assumed to happen is as follows:

(9) CYCLIC APPLICATION

- a. First, go through the list of transformations and form rules, in order, in the embedded clause.
- b. Then go up to the next clause up and go through the list all over again.

In this way, the transformations apply CYCLICALLY; that is, each clause defines a CYCLE in which the ordered list of transformations is applied. Thus, in (8b) passive applies only in the embedded clause on the first cycle; in (8c) it is skipped in the embedded clause (it is an optional transformation), but is applied in the main clause on the second cycle. In (8d) it is applied in the embedded clause on the first cycle, then again when its turn comes around on the second, main-clause cycle. For the purpose of cyclic applications ‘embedded clause’ may be either a CP or an S.

Also, we will assume that transformations are LOCAL; that is, they cannot reach into an embedded clause unless explicitly sanctioned. We implement this with the LOCALITY CONSTRAINT in (9):

(9) LOCALITY CONSTRAINT: In a transformation, no variable may contain a clause-boundary.

This means that unless a clause boundary is explicitly mentioned in the structural description of a transformation, the transformation will be clause-bounded.

Current Transformations and Form Rules

Here is the list of current transformation and form rules in order of application.

TO-PLACEMENT: (OBLIGATORY)

	$[_{CP} for$	$\{NP, CP\}$	(V_m)	X
SD:	1	2	3	4
SC:	1	2	$[V_M to]$	4

PASSIVE: (OPTIONAL)

	NP	X	V	$\{NP, CP\}$	Y
SD:	1	2	3	4	5
SC:	4	2	V_{B-B+3}	\emptyset	$5 + [_{PP} BY 1]$

AGENT DELETION: (optional)

	X	$V_{be-pass}$	Y	$[_{PP} by someone/something]$
SD:	1	2	3	4
SC:	1	2	3	\emptyset

EQUI: (obligatory)

	NP _i	X	V	[_{CP} C _{for}	NP _i	Y
SD:	1	2	3	4	5	6
SC:	1	2	3	∅	∅	6

EXTRAPOSITION: (optional for some verbs; obligatory for others)

	CP	VP	
SD:	1	2	
SC:	[i _{tpn}] _{NP}	2 # 1	(‘#’ means ‘daughter of’)

FORM RULES: (obligatory)

These all apply in a block during the relevant cycle.

V	→	V _{[past] or [present]}	/	NP	_____
V	→	V _[α person, β number]	/	NP _[α person, β number]	_____
V	→	V _[BARE]	/	V _m	_____
V	→	V _[-EN]	/	V _{have}	_____
V	→	V _[-ING]	/	V _{b-A}	_____
V	→	V _[-EN]	/	V _{b-B}	_____

WH-MOVEMENT: (obligatory)

	X	(wh)	Y
SD:	1	2	3
SC:	2+1	∅	3

Note: ‘wh’ stands an XP with a wh-cord in it

SAI: (obligatory if term 1 is present, otherwise optional)

	(wh)	NP	Aux	X
SD:	1	2	3	4
SC:	1	3 + 2	∅	3

Note: ‘Aux’ stands for V_m, V_{have}, V_{b-A}, V_{b-B}, and main verb *be*