MODULE 3 OPERATIONS AND STAGES IN THE TRANSFORMATION PROCESS

In this module, you will be exposed to some basic ideas on the structure of the Phrase marker in the Minimalist Program. We shall discuss the phrase architecture as it builds from the lexical base up to the phrasal top. We shall also discuss some of the changes in the categorical labels on the phrase markers. You will learn about the movement mechanism: how it is initiated, and how it is represented.

UNIT 1 BARE PHRASE STRUCTURE (BPS)

CONTENTS

1.0 INTRODUCTION

The native speakers derive their syntactic structure by merging words together to form phrases.Categorical labelling of those phrases comes up after the structures have been fully formed.In this unit, we will discuss how the native speakers achieve this.

2.0 OBJECTIVES

At the end of the unit, you should be able to:

- explain what is meant by bare phrase structure; and
- state how to use it in analysing simple language data.

HOW TO STUDY THE UNIT

a. Read this unit as diligently as possible.
b. Find meaning of unfamiliar words in the unit using your dictionary.
c. As you read, put major points down in a piece of paper or jotter.
d. Do not go to the next section until you have fully understood the section you are reading now.
e. Do all the Self-Assessment exercises in the unit as honestly as you can. In some areas where it is not feasible to provide answers to Self-Assessment exercises, go to the relevant sections of the unit to derive the answers.

3.0 MAIN CONTENT

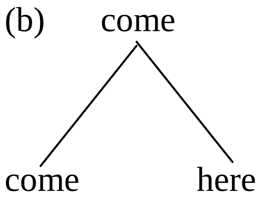
Bare Phrase Structure (often abbreviated BPS) is a major development of MP. This theory contrasts with X-Bar Theory, which preceded it, in four important ways.

3.1 The Derivational Structure of the BPS

This implies that the structure is built from the bottom up, word by word to derive a phrase. X-Bar Theory, on the other hand, is representational because the tree is first constructed before words are inserted into its end nodes. For instance, the X-Bar Theory will start its structural representation from the top by stating the phrasal category (i.e. VP) which will now yield its constituents (i.e. V and NP) before lexical items (i.e. *see* and *John*) will be inserted. The BPS, on the other hand, will start with the lexical items (i.e. *see* and *John*) which will merge to form the phrase (i.e. VP). This is what we meant by *bottom up*.

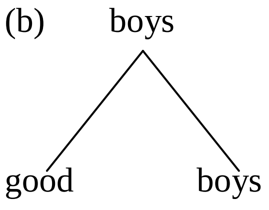
1 (a) come

(b)



2 (a) boys

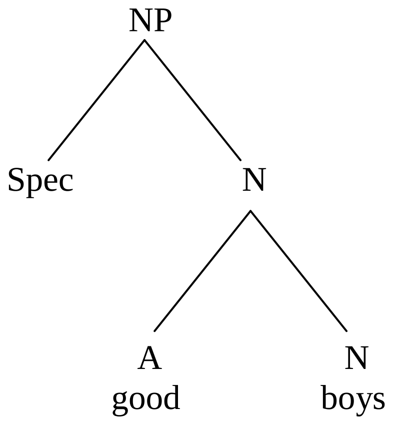
(b)



3.2 Flexibility of the Phrase Marker

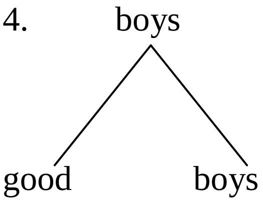
BPS does not have a rigid preconceived structure in MP, while in X-BarTheory, every phrase should have a Specifier and a complement. If these nodes are not filled, we will still assume they exist in the structure.

3.



While the Specifier position in (3) has to be assumed vacant just because the position must exist in GB analysis, the minimalist framework will not acknowledge any position that does not have lexical representation right from the lexicon.

4.



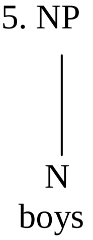
The principle simply adds one word to another, and it merges them into trees. The selection of words precedes the formation of syntactic trees. Since we use only the word we have alone in forming the tree, we cannot generate any vacant slot in the tree formed. Why do we have to create a Specifier position for *good boys* when we know that the derivation does not have any Specifier right from the outset?

3.3 Binary-branching Trees

BPS has only binary branching.

On the other hand, the minimalist framework presents its data on phrase markers using only binary branching nodes. When an item occurs alone, it will not be mapped into any tree diagram. You cannot have a tree diagram until you have at least two items. This idea is different from what obtains in GB where a single item (*boys*) will still be put under a non-branching NP tree as seen below.

5.

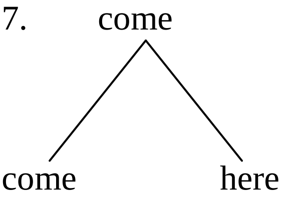


See how the minimalist framework phrases the following items with BPS:

6 (a) come
(b) come here
(c) <u>come here tonight</u> .

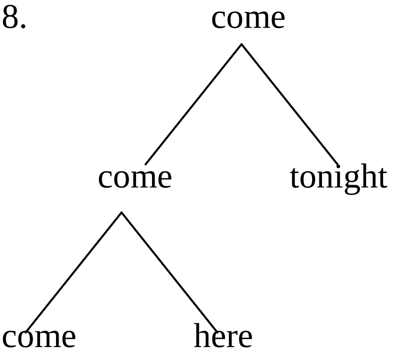
For 6(a) tree formation cannot be used yet because this is a single item. For 6(b) and 6(c), there are more than one item in the structure, therefore trees can be formed. See these trees in (7) and (8) below.

7.



(come + here)

8.



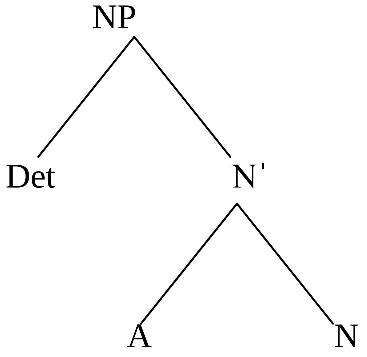
(come here + tonight)

Do you wonder why the phrase markers in (7) and (8) are labelled *come* rather than *here* or *tonight*? This happens because the phrase is a verbal structure which could have been represented with VP in GB framework. Here, the head word of the category is used instead of the V label. You can proceed to Section 3.4 for more discussion on this.

3.4 Phrasal Label

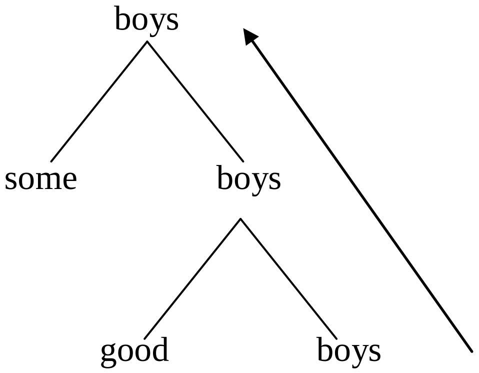
BPS does not distinguish between a "head" and a "terminal", but X-bar generates head-driven structures. In the diagram below, the use of N' (N-bar) has clearly shown that N (noun) is the head of the phrase. No wonder, the phrase is labelled NP (Noun Phrase). This is how phrases are described in using the X-bar theory of the GB.

9.



In the case of Minimalist Program, Bare phrase structure labels the representation by the actual words involved in derivation. The native speaker is less concerned about what is the head, but the line of the derivation from the base (bottom of the tree) can determine this. This is not a mere diagram but a logical path of reason reproduced from the line of derivation. See this minimalist concept in the diagram below, and compare it with the X-Bar concept above. You can see how the minimalist representation creates a path of reason for the word *boys* in (10) below.

(10)



Boys is the most important word in the derivation. This word keeps merging with other words right from the base until a desired syntactic structure is formed.

3.5 Operation Merge in the BPS

BPS has two important operations; these are Merge and Move. When Merge operates on two objects (say α and β), it merges them into an unordered set with a label (either α or β, in this case α). The label identifies the properties of the phrase.

Merge $(\alpha, \beta) \rightarrow \{\alpha, \{\alpha, \beta\}\}$

(Merge alpha and beta, to derive another alpha which is a conglomeration of alpha and beta)

We can assume that Operation Merge is being applied to these lexical items: *build* and *houses*. Let *build* be α, while *houses* represents β, we are going to have the following:

Merge (build, houses) → {build, {build, houses}}

Deciding on which of the two items should take the label of the phrase is a simple thing in this case. The phrase *build houses* can easily permit an elliptical representation of *build* than it will ever permit *house.* This means that *build* and *build houses* behave more alike than *houses* and *build houses* actually do. We therefore represent the phrase build house with the keyword *build* as follows.

11 (a) They planned to build houses

(b) They planned to build

(c) *They planned to house

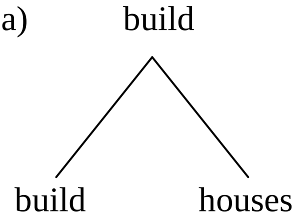
12 (a) We found good boys

(c)We found boys

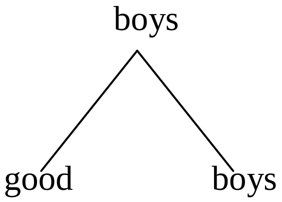
(c)*We found good

If we study the underlined items in the sentences above, we will notice that *build* can replace *build* or *build house,* but *house* cannot replace any of these. We can also see that *boys* can replace *good boys,* but *good* cannot. With this, we can justify the bare phrase structure of representation as they appear in (13) below.

13 (a)



(b)



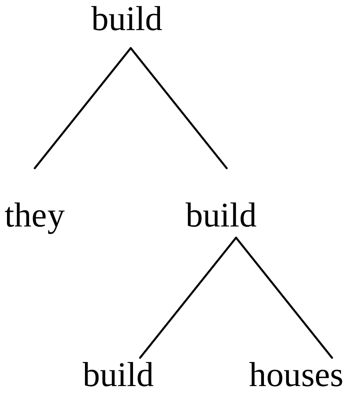
In (13), the VP level is represented with *build* while the NP level is represented with *boys*. This is how merging operation is represented in the BPS.

It is also possible to merge items with structures already built. In this case, we assume that a new syntactic element say γ combines with an already formed phrase $\{\alpha, \{\alpha, \beta\}$ in order to form a larger phrase $\{\gamma, \{\gamma, \{\alpha, \{\alpha, \beta\}\}\}\}$ where γ projects as the head. This can be properly represented symbolically as follows:

$\text{Merge}\ (\gamma, \{\alpha, \{\alpha, \beta\}\}) \rightarrow \{\gamma, \{\gamma, \{\alpha, \{\alpha, \beta\}\}\}\}$

If we want to continue the building of the tree structure in 13(a), we can take the pronoun *they* as the new syntactic item to be merged to our earlier-formed phrase: *build houses.* This recent merger results in the derivation of this newly-formed phrase: *they build houses*.

In this particular phrase *they build houses, build* is still the keyword of the derivation. So we can say *build* 'projects' as the label (This is what is stillreferred to as 'head' in GB).



SELF-ASSESSMENT EXERCISE

i. Describe to a friend why BPS trees are considered bottom up in derivations.
ii. Use this structure read books to compare the BPS bottom up with the X-Bar top down derivations.

4.0 CONCLUSION

The BPS is a useful tool in the Minimalist framework. It keeps the analysis closer to the actual language data rather than the categorical labels. At each level of the derivation, the analysis has a keyword from the language data which heads represent the major category in the structure.

5.0 SUMMARY

In this unit, we have seen how syntactic structures are formed through a unified generalised transformation. We have also peeped into two important Chomskian concepts in Minimalist Program. These are bare phrase structure and binary branching nodes.

6.0 TUTOR-MARKED ASSIGNMENT

Analyse the following in BPS

1. Sing songs
2. Bad eggs
3. Come to John
4. John eats bread
5. He can sing

7.0 REFERENCES/FURTHER READING

Chomsky, Noam (1994). “Bare Phrase Structure”. MIT Occasional Papers in Linguistics #5, Cambridge, Mass.: MIT Department of Linguistics and Philosophy. (Also published as Chomsky, Noam. 1995. “Bare Phrase Structure.” In: Webelhuth, Gert (Ed.).

Chomsky, Noam (1995). *The Minimalist Program*. Cambridge, Mass.: MIT Press, sections 1.3.2, 3.2, 33 (188-191), 4.3, and 4.4 (241-276).

Cook, Vivian & Mark Newson (1996). “Chapter 9: A Sketch of the Minimalist Programme.” In: *Chomsky’s Universal Grammar: An Introduction.* 2nd Edition. Oxford: Blackwell Publishers, 338-344.

Nunes, Jairo (1995). “The Copy Theory of Movement and Linearisation of Chains in the Minimalist Program.” University of Maryland at College Park, PhD dissertation, 60-64, 67-75.

Radford, Andrew (1997). *Syntactic Theory and the Structure of English: A Minimalist Approach*. Cambridge: Cambridge University Press. Chapter 3.

UNIT 2 PHRASAL CATEGORIES IN THE MINIMALIST PROGRAM

CONTENTS

1.0 INTRODUCTION

There are two types of phrasal categories in the Minimalist Program. These are the lexical and the functional categories. The lexical phrasal categories are verb phrases (VPs), noun phrases (NPs) and adjectival phrases (APs), adverbial phrases (ADPs) and prepositional phrases (PPs). The functional phrasal categories on the other hand include determiner phrases (DPs), the vp shell, Tense Phrases (TPs) and complementiser Phrases (CPs).

In the brief survey we need in this study, we just have to restrict ourselves to few of the common functional phrases. The Minimalist Program (MP) does not in any way make syntax to be more difficult. Rather it helps to simplify the principles in the earlier models of Transformational Generative Grammar (TGG).

2.0 OBJECTIVES

At the end of this unit, you should be able to:

- describe how syntactic trees are formed in the MP;
- distinguish between lexical and functional categories; and
- draw and label phrase markers (tree diagrams) in the MP.

HOW TO STUDY THE UNIT

a. Read this unit as diligently as possible.
b. Find meaning of unfamiliar words in the unit using your dictionary.
c. As you read, put major points down in a piece of paper or jotter.

d. Do not go to the next section until you have fully understood the section you are reading now.

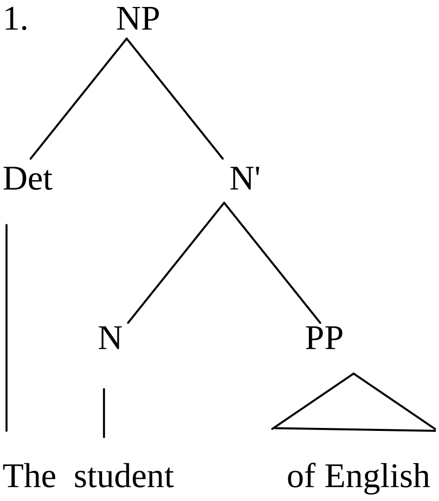
e. Do all the Self-Assessment exercises in the unit as honestly as you can. In some areas where it is not feasible to provide answers to Self-Assessment exercises, go to the relevant sections of the unit to derive the answers.

3.0 MAIN CONTENT

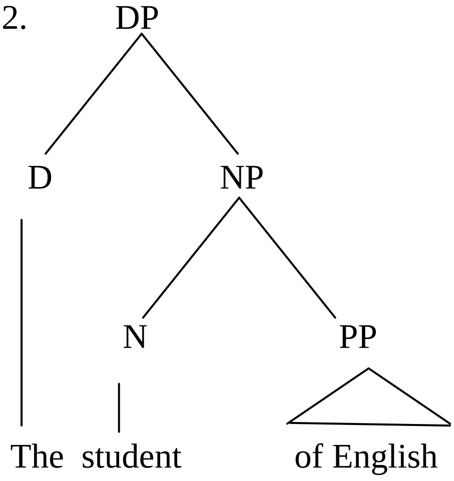
3.1 The Determiner Phrase (DP)

In the minimalist framework, what used to be called the noun phrase (NP) has been redefined. The determiner is a head which selects the nouns as its complement. The determiner, which used to be a Specifier in the NP, is now a functional head of the Determiner Phrase. The NP is a complement of the DP. So, we can say that the core of the nominal category is no longer the noun but the determiner. Abney (1987) work is one of the important works that laid the foundation of the DP hypothesis. DP hypothesis will be more adequately discussed in Module 4. It is mentioned here to briefly justify the use of DP for what could have been ordinarily labelled NP.

1.



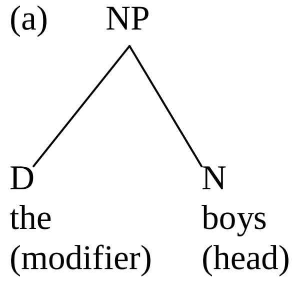
2.



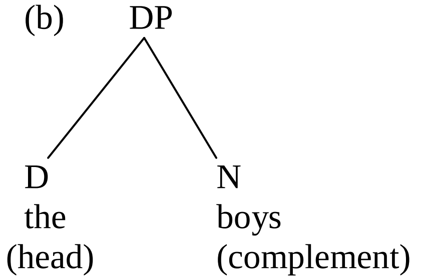
Apart from the inclusion of the pronoun in the determiner group, the rest of the determiners are those traditionally labelled as determiners (i.e. a, the, that, those, this, these, my, our and your). These are the traditional articles, demonstrative adjective, and possessive adjectives.

In a DP, the determiner is the head of the phrase. Hence, the DP phrase marker will replace 3(a) with 3(b) below.

3 (a)



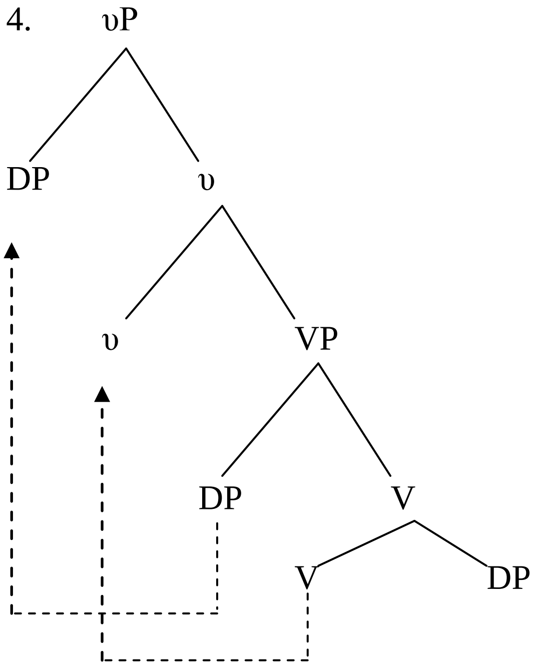
3 (b)



3.2 The VP Shell

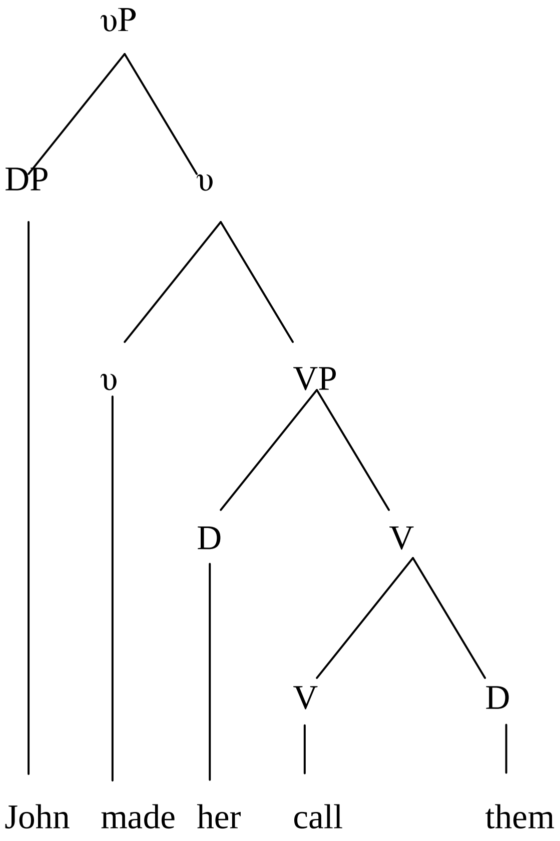
The verb phrase in the minimalist framework has a functional head written in a lower case ‘v’ label. This is called the light verb. Hence the functional vP differs from the lexical VP. This structure provides analytical basis for causative and double object constructions.

4.



Generally, it is assumed that the verb has to be copied to the functional head position before it can be liable to tense marking as shown in (4) above. However where causative construction is used, the light verb will host the causative verb; therefore the main verb can no longer be raised into the functional position provided by light verb. The use of light verb makes possible the analysis of causative construction in (5) below.

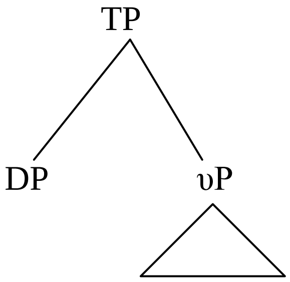
5.



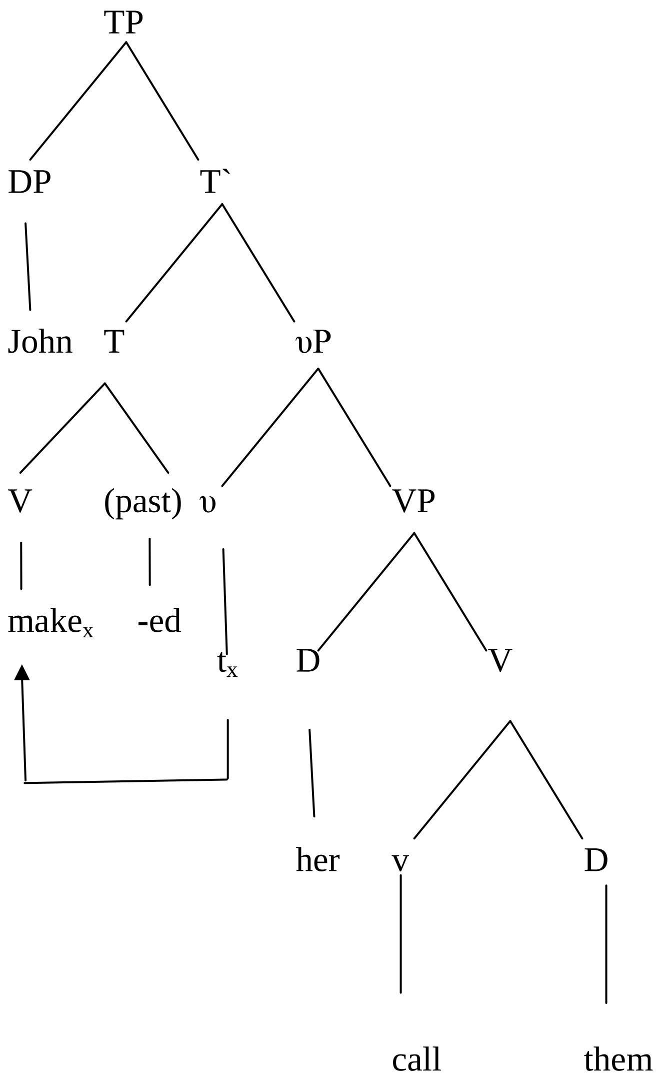
3.3 The Tense Phrase (TP)

The simple affirmative sentence is analysed as Tense Phrase (TP). So every VP need to become a TP in order to express tense marking.

6(a)



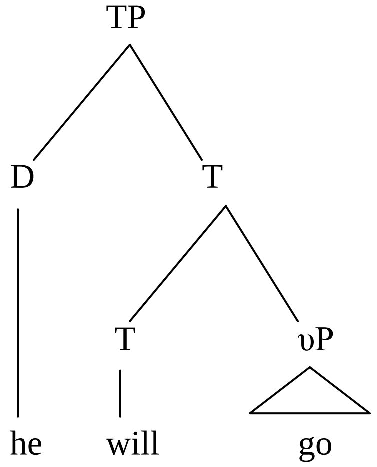
6(b)



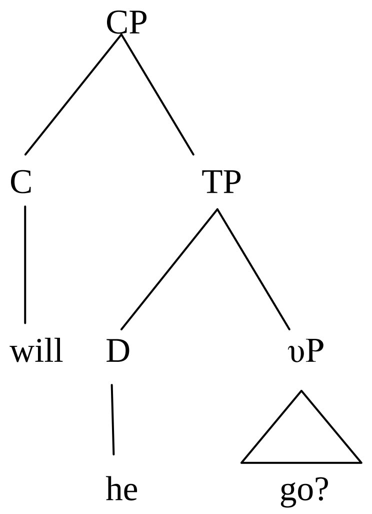
3.4 The Complementiser Phrase (CP)

Questions, inversions and sub-ordinate clauses are analysed with CP phrase marker. The inverted items are seen as being copied out of the TP (basic affirmative sentence) into a structural extension provided by the CP.

7(a)

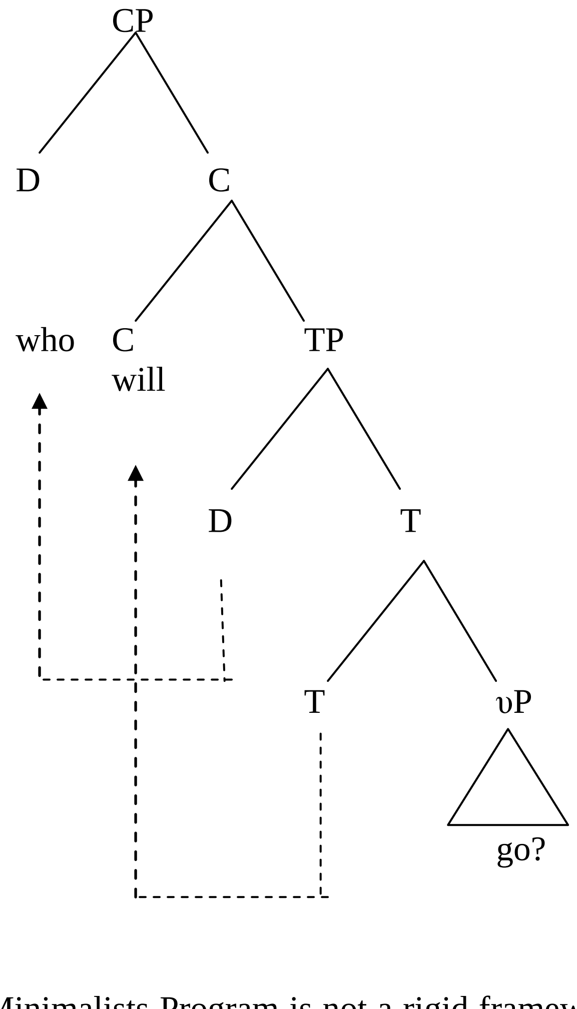


7(b)



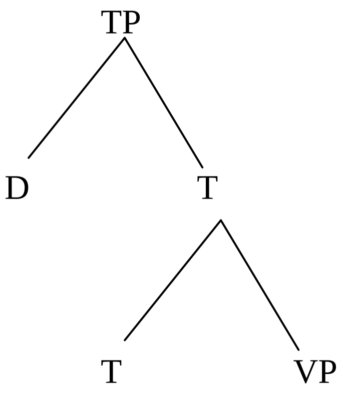
NB. Note that the modal auxiliary is usually placed under tense node here. This is because we are beginners and we can leave that there for now. The triangle under υP is a shorthand representation of the whole υP.

7(c)



The Minimalists Program is not a rigid framework. You may ignore the vP in your TP structure if it is not required to the analysis of your data. This is expected if you are not dealing with complex verbal structures such as causative and double object constructions. You can therefore have this.

8.

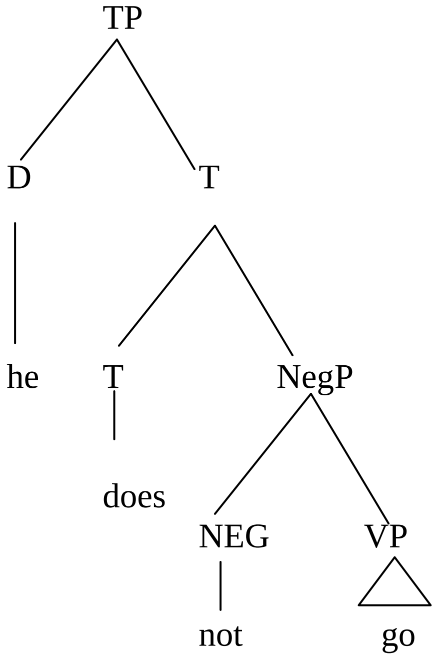


3.5 The Negation Phrase (NegP)

Negation is one of the functional categories one easily comes across in sentence analysis. This functional category has a wide spectrum which definitely our present knowledge of the model cannot cope with. Let us just be content with how we can represent simple sentence negation. In this case, negation will form a Negation Phrase NegP. You can see how the negative sentence in (9) is represented in (10). When we have a better grip on the theory we may want to look into constituent negation within the Minimalist framework.

9. He does not go.

10.



SELF-ASSESSMENT EXERCISE

i. Provide tree diagram analyses of the following in bare phrase structure.

   (a) sit down
   (b) come up
   (c) new students
   (d) at Lagos

ii. Convert the tree diagrams in (1) above to category-based labelled trees.

iii. Attempt a tree diagram analysis of the following using binary branching nodes with categorical labels.

   (a) sit down there

   (b) come up here

4.0 CONCLUSION

It is very important to understand the proper use of tree diagrams as phrase markers in the Minimalist Program. The mastery of the theory depends on one's ability in employing the phrase marker to reveal structural details that will make our analysis a dependable syntactic account.

5.0 SUMMARY

In this unit, we have seen how syntactic structures are formed through a unified generalised transformation. We have also peeped into two important Chomskyan concepts in Minimalist Program. These are bare phrase structure and binary branching nodes. We have also seen some of the common functional phrases: the DP, the TP and the CP. The Determiner Phrase (DP) has eventually replaced our earlier concept of Noun Phrase (NP). The pronoun has become a determiner which can function as the head of a Determiner Phrase (DP). The noun phrase (NP) is now a complement of the determiner. Going by this analysis, the noun will serve as a complement to the article or to the pronoun.

6.0 TUTOR-MARKED ASSIGNMENT

Analyse the following on labelled tree diagrams:

1. the men
2. our people
3. He knows the boy
4. we, the people
5. What did you say?
6. Did you say that?

7.0 REFERENCES/FURTHER READING

Abney, S. (1987). *The English Noun Phrase in its Sentential Aspect*: PhD Dissertation, MIT.

Bhatt, C. Löbel, E. & Schmidt, C. (Eds.).(1989). *Syntactic Phrase Structure Phenomena.* Armsterdam/Philadelphia: John Benjamins.

Radford, Andrew (1997). *Syntactic Theory and the Structure of English: A Minimalist Approach*. Cambridge: Cambridge University Press. Chapter 3.

Ouhalla, Jamal (1994). *Introducing Transformational Grammar from Principles to Parameter*. London: Edward Arnold.

UNIT 3 FEATURE CHECKING THEORY

CONTENTS

1.0 INTRODUCTION

We learnt earlier (in section 3.1of Unit 1, Module 2) that the move-alpha rule that normally moves items in the Government and Binding model has been laid aside in the Minimalist Program. Do you now ask yourself how items are still being moved in Minimalist Program? This unit will introduce you to the factors initiating movement in the framework.

2.0 OBJECTIVES

At the end of the unit, you should be able to:

- list the various features that are involved in syntactic computation; and
- describe the effects of morphosyntactic features.

HOW TO STUDY THE UNIT

a. Read this unit as diligently as possible.
b. Find meaning of unfamiliar words in the unit using your dictionary.
c. As you read, put major points down in a piece of paper or jotter.
d. Do not go to the next section until you have fully understood the section you are reading now.
e. Do all the Self-Assessment exercises in the unit as honestly as you can. In some areas where it is not feasible to provide answers to Self-Assessment exercises, go to the relevant sections of the unit to derive the answers.

3.0 MAIN CONTENT

3.1 Feature-Driven Movement

Movements are licensed in the minimalist program through the feature checking mechanism. We have already seen some things on the operation of greed as an economy principle. The claim that items moved because they have in them the potential to move is the basic thing we need to revisit now. What actually makes an item to move is the presence of un-interpretable features they possess. The demand that those un-interpretable features be interpreted before the derivation can converge (that is to be considered correct) will make the item having non-interpretable features to look for another item which can have those un-interpretable features valued. The item that can help to provide interpretable features is called a *probe*. The item that receives help before it can interpret its un-interpretable features is called a *goal*. The process through which a probe accesses a goal is called *probing.*

3.2 Types of Features

There are at least three types of features relevant to our current analysis. These are phonological features, semantic features and the morphosyntactic features. (See Module 1, Unit 1)

Phonological Features

The phonological features are the properties of the PF realisations. Natural languages cannot do without having speech forms. This speech form is a systematic combination of sounds for communication purposes. Theoretically, we may be talking of features like [+obstruent], [-coronal] and [-ATR] which are usually dealt with in Generative Phonology. In the present course, we may not really go beyond the fact that these features combine to derive the PF which is an important interface level in Syntax.

Semantic Features

These are the features through which semantic information is coded. They involve such features as {+male}, {+married}, {+young}. The value of these kinds of features is usually rooted in the context based on the understanding of the users of the code in question. This constitutes the sum of the LF interpretation.

Morphosyntactic Features

These are called formal features. They are the major features needed for syntactic operations. Morphosyntactic features comprise the followings: person, gender, number, and case. These are often called phi features.

3.3 Feature-based Computations

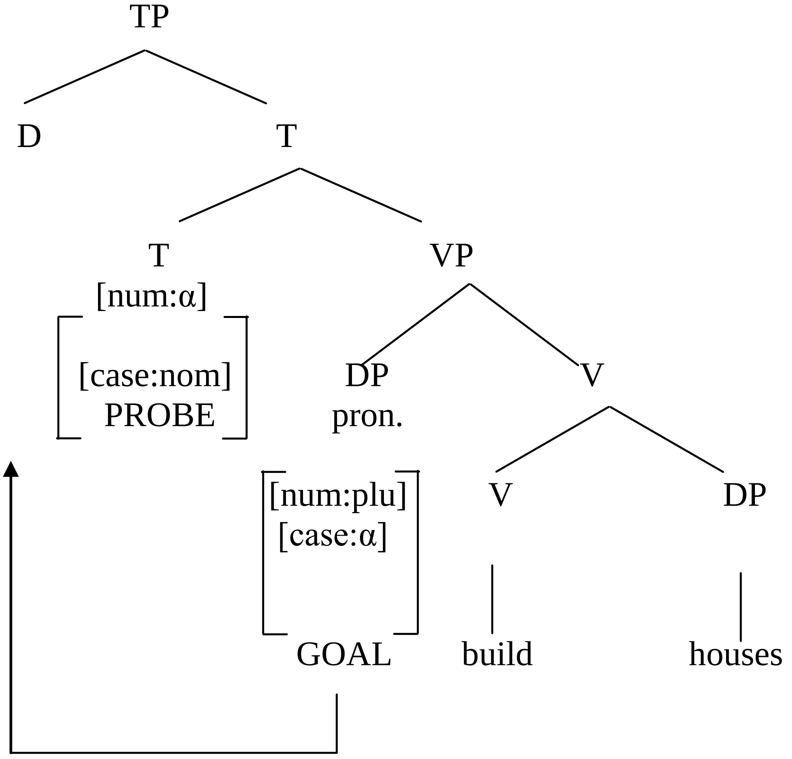
Let us go into the discussion as we consider why nouns have to move. Until a noun is used in a sentence, it is assumed that its case feature is un-interpretable. For this feature to be properly valued, the said noun must locate a verb that can help to interpret the un-interpretable case feature. This noun (as a probe) has to probe for that verb (which acts as a goal).

We will as well assume that our selection of words is initially done without a specific order. Then we can start with the keyword which is likely going to be a verb. We may decide to have the numeration as follows.

1. Σ:{build, houses, 3plu.}

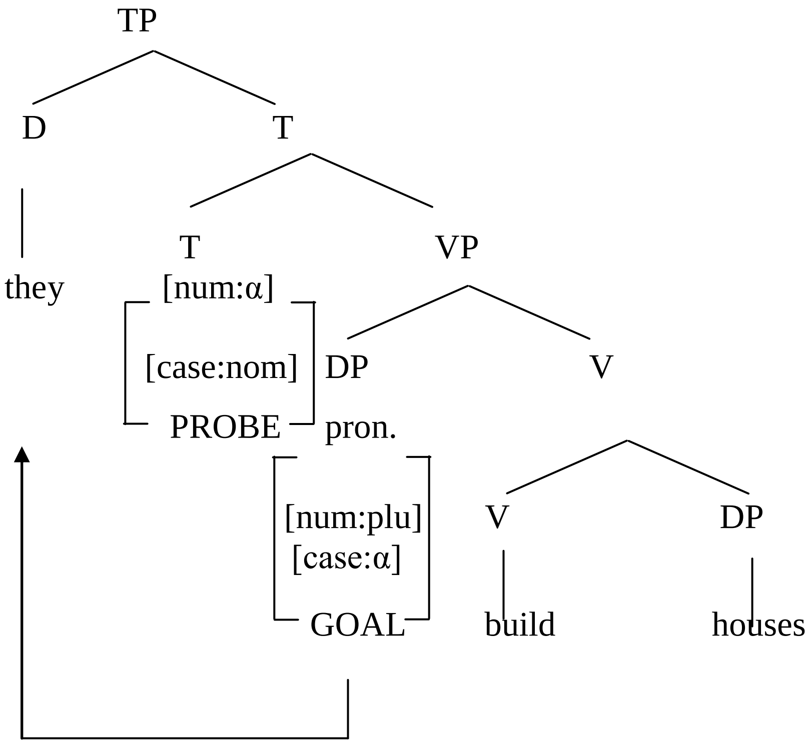
In the (1), the subject of the verb has the following features: [3plu]. Through this means we identify it as third person plural. It happens to be a pronoun; then it will be a third-person plural pronoun.

2.



We then assume that the pronoun needs nominative case. This case can be interpreted via the tense head. (It should be noted that two views exist on how nominative case can be valued. Some scholars think it must be with the tense head; others feel it should be with the verb. We are learners; therefore, we will adopt the first view. This does not mean that the second view is wrong. We just need to know that our knowledge of the subject matter is too low to really go into detailed theoretical argument now.

3.



In the diagram above, we can see how the pronoun was represented at the initial stage. There it takes a non-phonetic form. At that point, it could be pronounced either as *them* or *they*. We eventually choose to call it *they* when we have moved it to the Specifier position of the TP. That is where it is merged with T which has interpretable case feature.

SELF-ASSESSMENT EXERCISE

In a simple way, describe what you understand by the term feature-driven computation. Read 3.1 and 3.3 for the answers.

4.0 CONCLUSION

Minimalist Program is considered as feature-driven syntax due to the place of morphosyntactic features in syntactic derivations. Major syntactic operations are triggered when items seeks to check their un-interpretable features making structural links with another item where those features are interpretable.

5.0 SUMMARY

In this unit, we have considered the relevance of features to syntactic movements. We have also been introduced to the following types of features: phonological, semantic and morphosyntactic mentioned. The morphosyntactic features are very dwelt upon because they are needed in syntactic derivations.

6.0 TUTOR-MARKED ASSIGNMENT

1. Define the following terms:
   (a) Goal
   (b) Probe
   (c) Probing
   (d) Un-interpretable features.
2. Explain with examples from English how un-interpretable feature can cause syntactic movements.

7.0 REFERENCES/FURTHER READING

Chomsky, Noam (1995). *The Minimalist Program*. Cambridge, Mass.: MIT Press, 3.2 (176-180), 4.2.2 (230-241), 4.5.1 (276-278), 4.5 (276-312).

Lasnik, Howard (1999). "On Feature Strength: Three Minimalist Approaches to Overt Movement." *Linguistic Inquiry*.30: 197-218.

Radford, Andrew (1997). *Syntactic Theory and the Structure of English: A Minimalist Approach*. Cambridge: Cambridge University Press. Chapters 2 and 5.

UNIT 4 THE COPY THEORY OF MOVEMENT

In the last unit, we discussed syntactic movements. In this unit, we shall proceed to see the actual movement rule by comparing the trace theory with the copy theory.

CONTENTS

1.0 INTRODUCTION

In the GB, when an item is moved, it is believed that the entire item is moved while its shadow remains at the source of the movement. However, in MP, a word item creates another copy of itself, and this copy is moved. The original copy is subsequently deleted.

2.0 OBJECTIVES

At the end of this unit, you should be able to:

- distinguish between the movement in GB and the copy theory in the MP; and
- explain the copy theory of movement with examples from English.

HOW TO STUDY THE UNIT

a. Read this unit as diligently as possible.
b. Find meaning of unfamiliar words in the unit using your dictionary.
c. As you read, put major points down in a piece of paper or jotter.
d. Do not go to the next section until you have fully understood the section you are reading now.
e. Do all the Self-Assessment exercises in the unit as honestly as you can. In some areas where it is not feasible to provide answers to Self-Assessment exercises, go to the relevant sections of the unit to derive the answers.

3.0 MAIN CONTENT

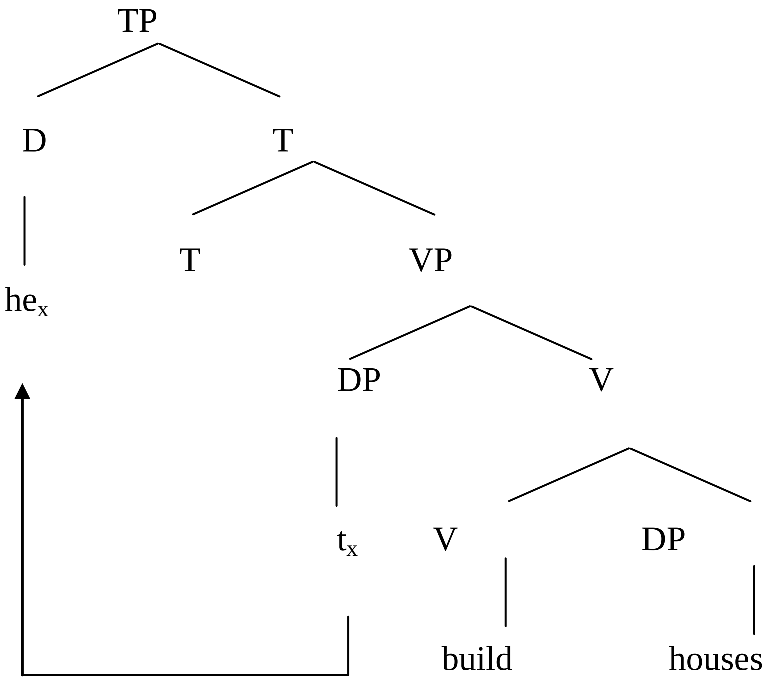
3.1 The Trace Theory in Government and Binding Theory

The trace theory holds that if an item is moved, it will leave behind a copy of itself. This copy is called trace. In GB, the position occupied by a trace is not vacant as a landing position for any other item moved into it. The way the binding theory perceives NP trace and the way subjacency condition respects trace occupied positions are good instances on the importance of trace in the theory.

See below how the original positions of the moved items are represented with *t* (trace). Take note of how this trace is co-indexed to the moved item with an *x* subscript.

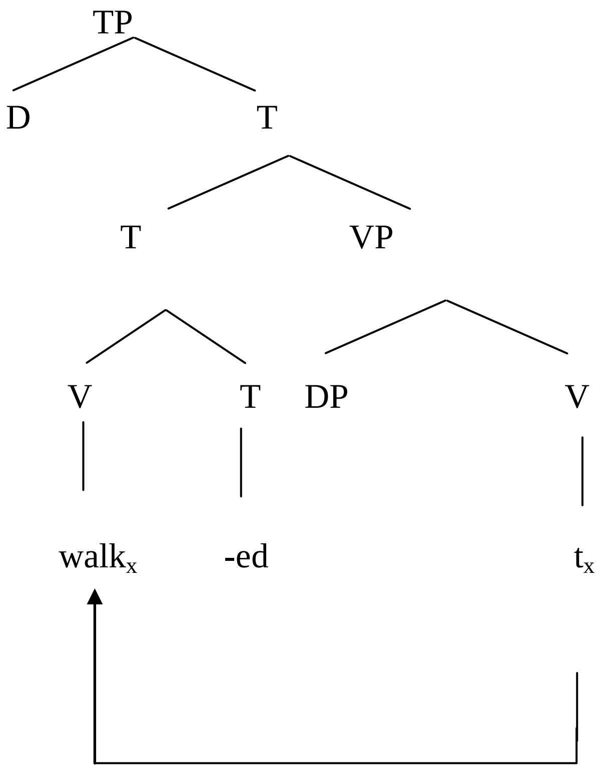
The trace theory will show the movement of the subject to the TP for nominative case as follows.

1.



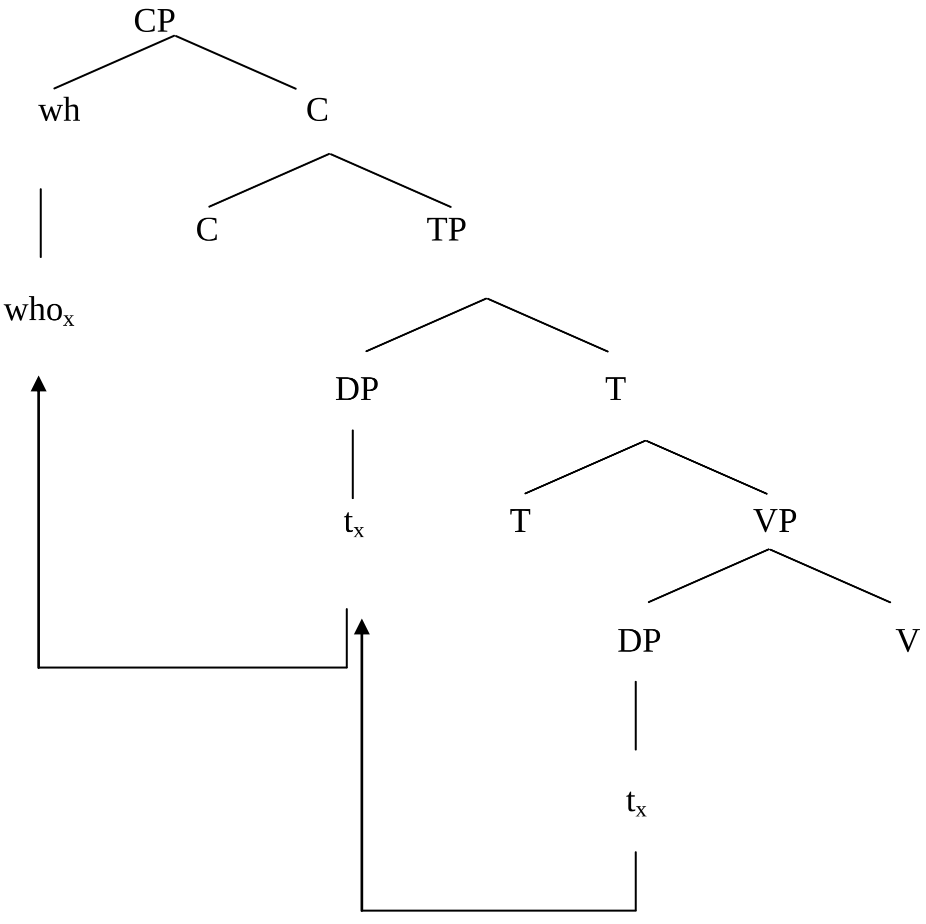
Verb movement can also be represented below as the movement of the verb to adjoin to T. This is called head to head movement implying that V the head of VP moves to T the head of TP.

2.



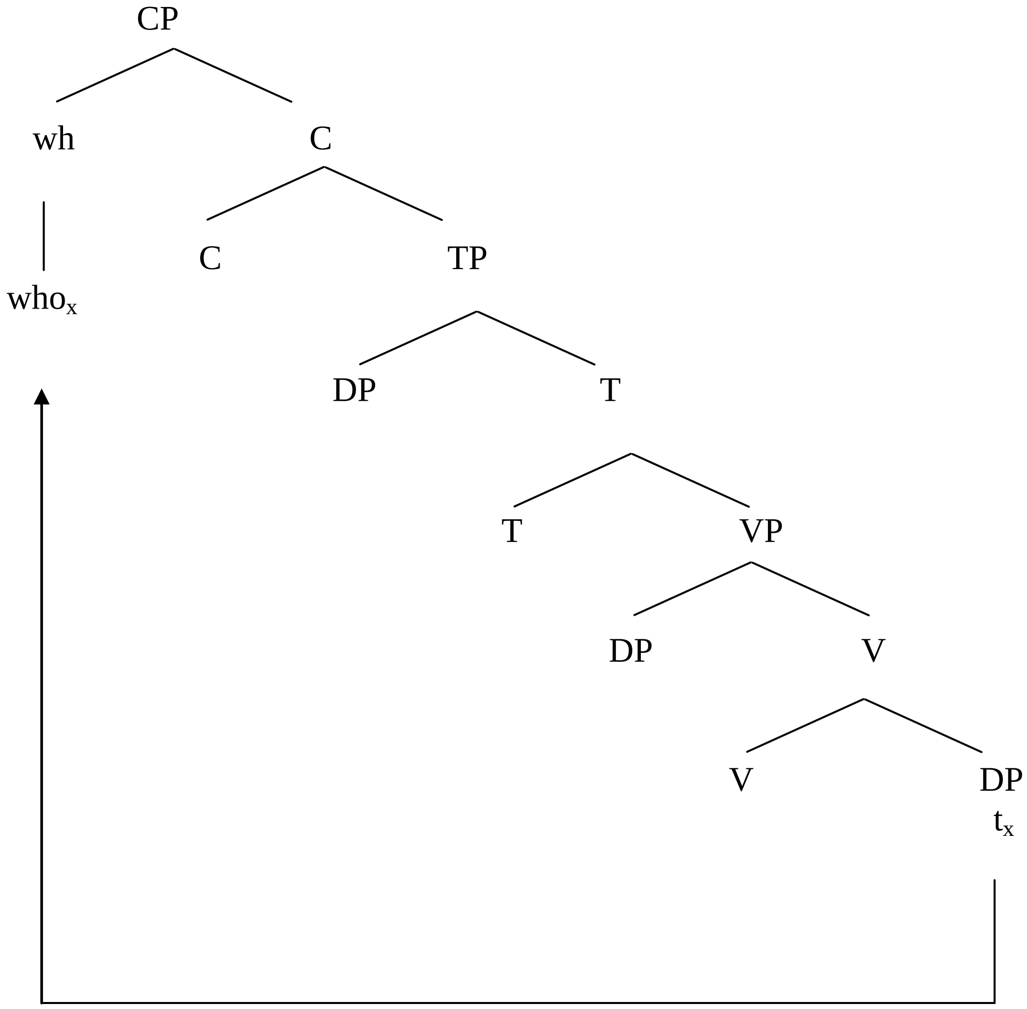
The movement of a wh- operator (wh-movement)

3.



wh-movement can apply to the subject and the object. In each case, the trace is left at the extraction point of the moved item.

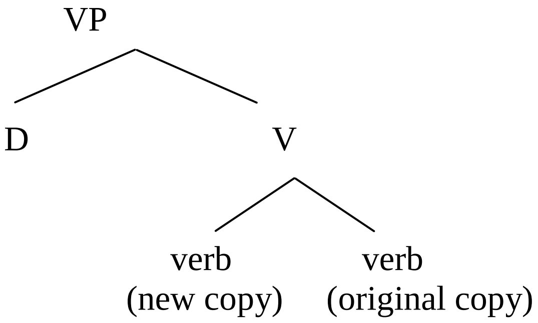
4.



3.2 The Copy Theory of Movement inChecking the Tense Feature of a Verb and the Nominative Case Feature of a Subject

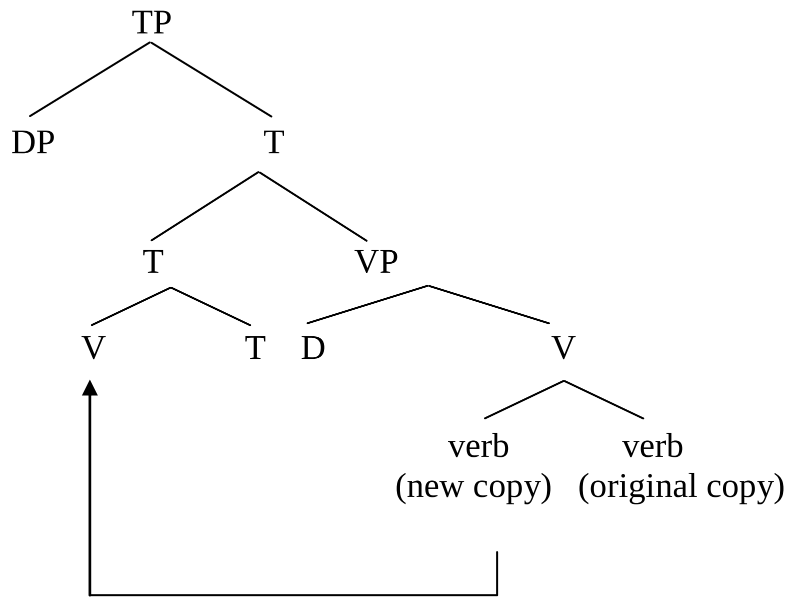
You first create a copy for the verb to be moved. See (5) below.

5.



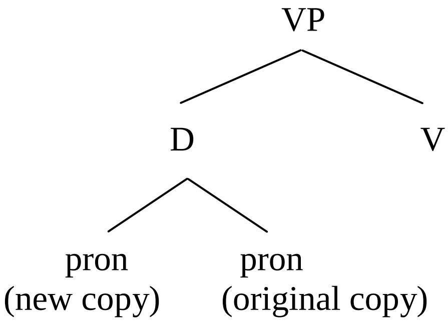
The VP will be merged with T to form a TP, and the new copy of the verb will be merged with the T head of the TP as shown below.

6



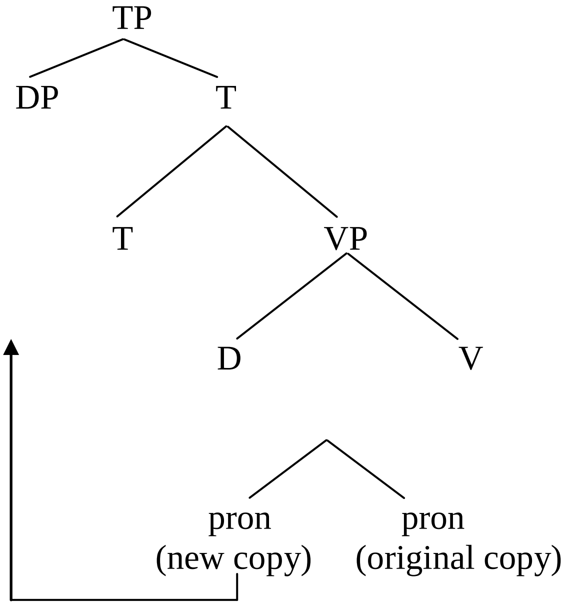
To check the nominative case of the subject, you need to create a copy for the subject to be moved.

7.



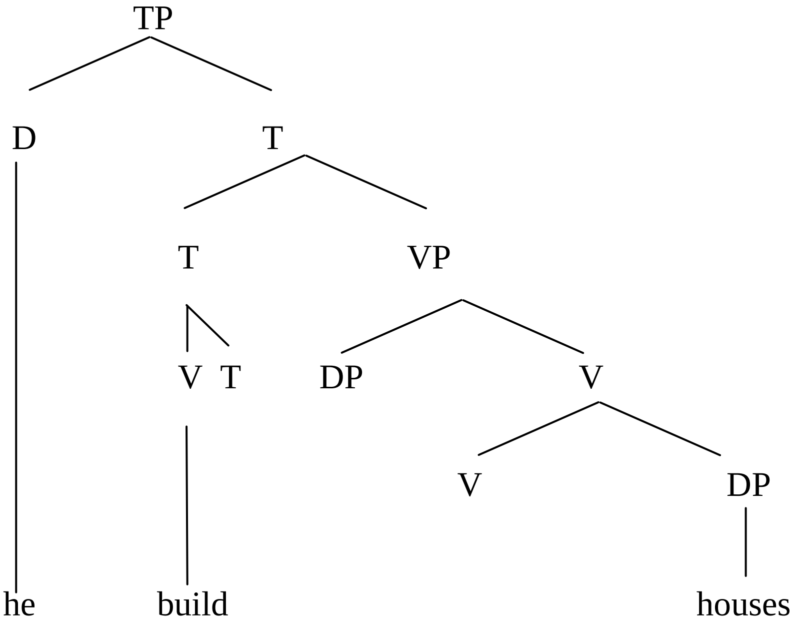
As the VP is merged with T to form a TP, the new copy of the subject will be moved into the TP as shown below.

8.



Before the spell-out stage, the derivation will delete all the copies of the moved item except the most recent copy of each movement. This deletion will include the original copy of the moved item as we can see in (9) below. You will note that the trace of the movements is not marked in this final version. This is a very important feature in the copy theory of movement, the earlier copies of the moved items are deleted on the PF, and the change represented in the final structure is mainly showing the PF. However, we keep using trace in our analysis so that students can find it easier to account for each derivation.

9.

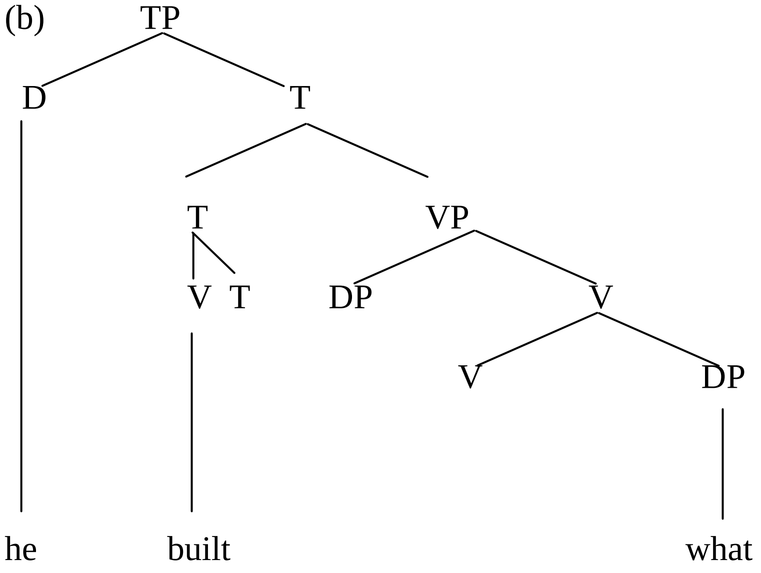


3.3 The Copy Theory of Movement on wh-movement

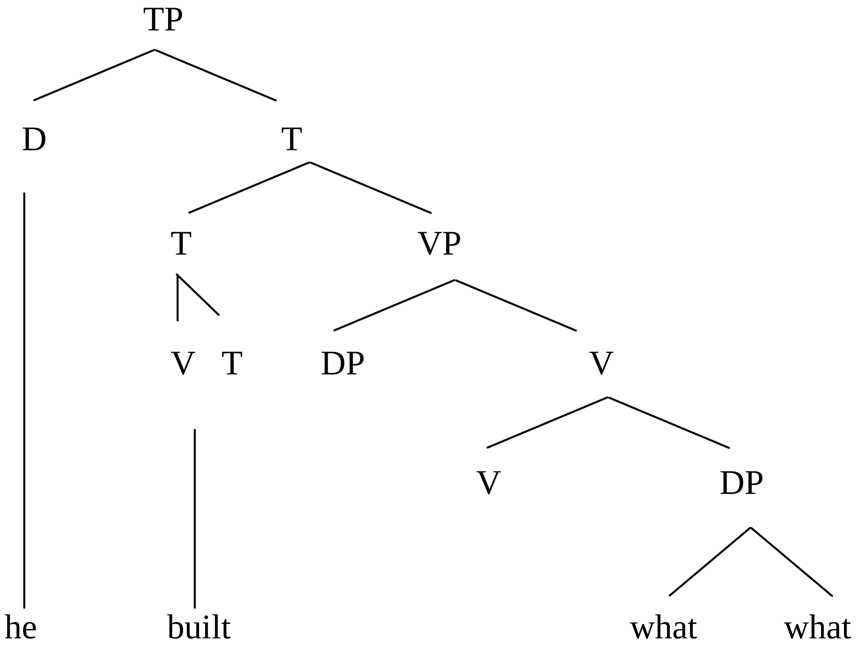
We may assume that we already have a TP in the process of the derivation. However, the derivation that develops a wh movement must have the moved item as *unknown*. In other word, the moved item is a wh-word. See a typical illustration in (10) below.

10 (a) He built what

10 (b)

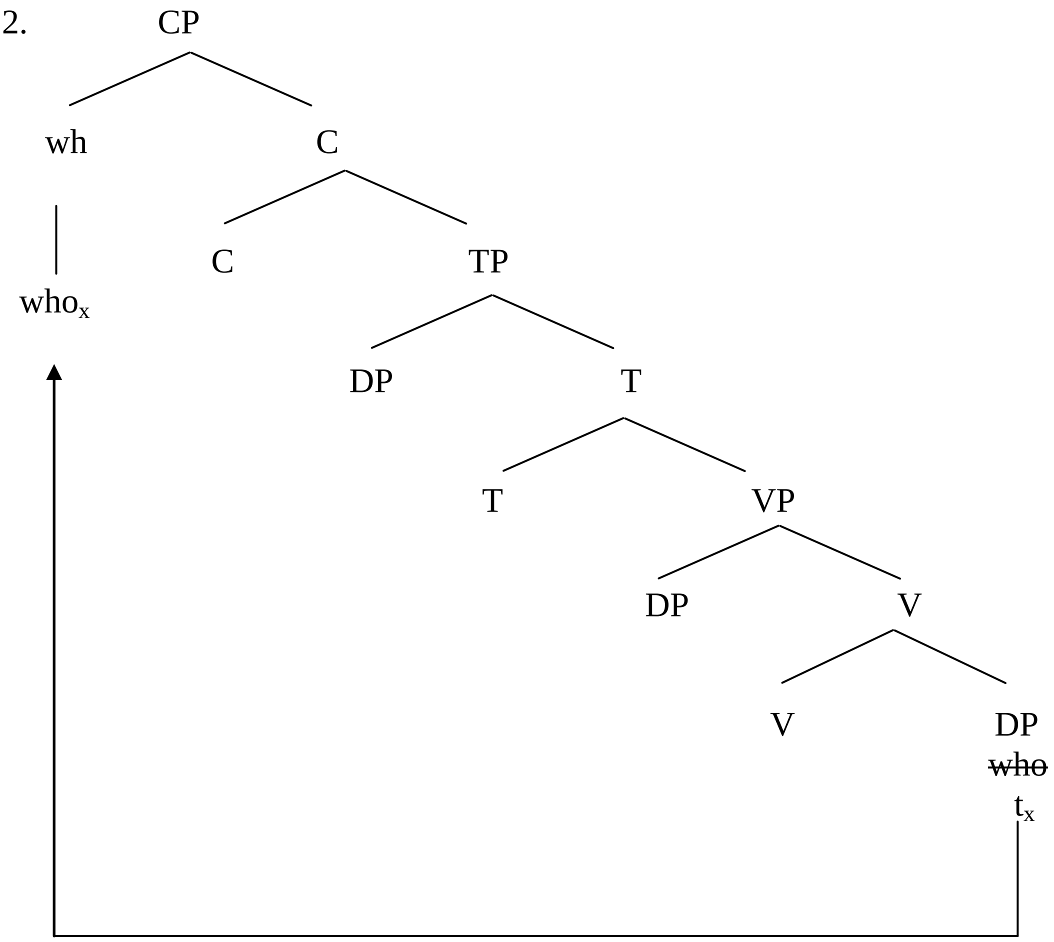


11.



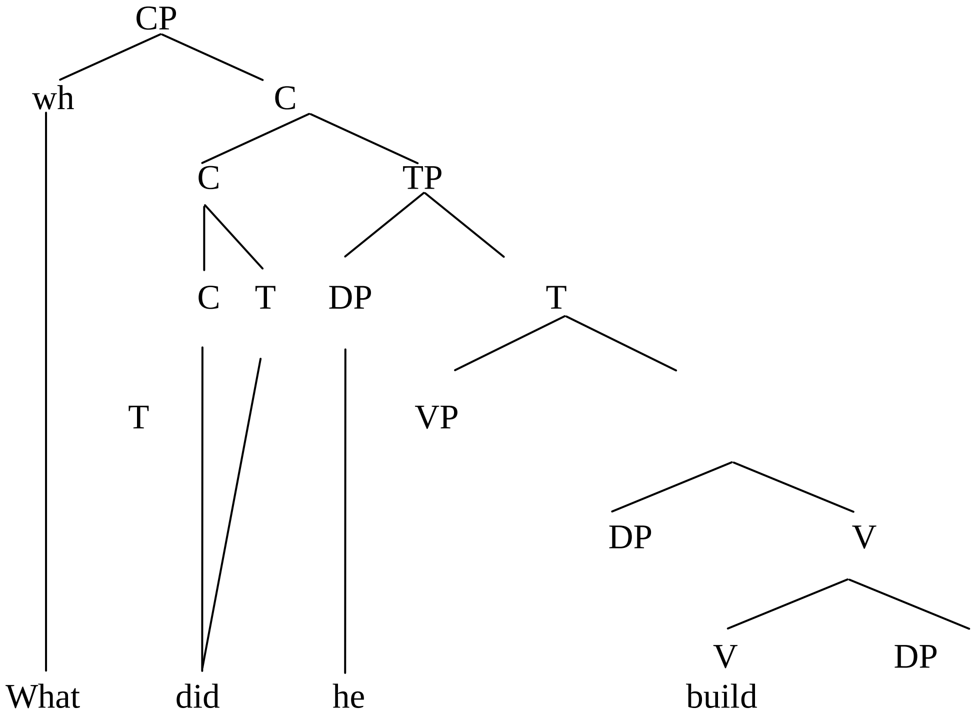
The TP will be merged to C (Complementiser); then the new copy of the wh –word will be moved to the CP. We try to represent all this in (12) below.

12.



Any copy that is not needed in the PF must have been deleted before the spell-out stage.

13.



In (13) above, you can see that the wh-word is not represented with any trace at the extraction point. You can also take note of the *do-insertion* strategy which we place under C node. The change from *do* to *did* is a proof that the tense feature at T is checked on *do* at C which result in *did*. The way we collapse C and T nodes to be represented by a single lexical item *did* is merely a graphical way of showing that the tense morpheme cannot be realised as a suffix. Hence we could have had something like *do* + *ed* .

SELF-ASSESSMENT EXERCISE

Explain to a colleague the major differences you think exist between the copy theory of movement and the GB Trace theory?

4.0 CONCLUSION

The copy theory of movement reveals that what we normally refer to as a trace is actually an authentic copy of the moved item. The original copy of what we move is not really the final copy that is usually retained in the speech, but the initial copy will always be tagged *t* (trace) in the GB framework.

5.0 SUMMARY

We have seen the concept of the trace theory within the GB framework. We have also seen the concept of movement in the Minimalist Copy theory of movement. Instances cited here illustrate the NP movement, verb movement and wh-movement.

6.0 TUTOR-MARKED ASSIGNMENT

1. Distinguish between the trace theory and the copy theory.
2. Analyse the following with copy theory:
   (a) Verb movement
   (b) Wh-movement

7.0 REFERENCES/FURTHER READING

Chomsky, Noam (1995). *The Minimalist Program*. Cambridge, Mass.: MIT Press, 4.5 (276-312).

Lasnik, Howard (1999). *Minimalist Analysis*. Oxford: Blackwell Publishers.

Nunes, Jairo (1995). “The Copy Theory of Movement and Linearisation of Chains in the Minimalist Program.” University of Maryland at College Park, Ph.D dissertation.

Radford, Andrew (1997). *Syntactic Theory and the Structure of English: A Minimalist Approach*. Cambridge: Cambridge University Press.