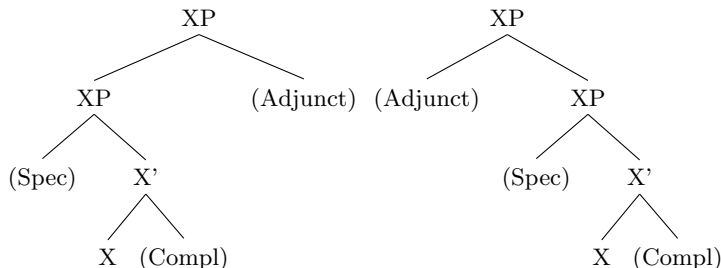# Ling 120B: Syntax I

Nico(letta) Loccioni

April 11, 2022

X-bar Theory I

Behind the range of diverse constructions that English and other languages allow, we find surprising uniformity and regularity.

- Last time we introduced the idea that the rule system that underlies our phrases is very simple.

- Every phrase looks the same!

- For English, syntax will be the iteration of:

This is called **X-bar Theory**.

X-bar Theory II

X-Bar theory is a theory that tries to eliminate phrase structure idiosyncrasies: every phrase looks the same.

- Every phrase is the projection of some lexical category: N, V, A, Adv...

- The system of projected constituency is common to all categories.

We need to distinguish between 4 types of elements:

- **heads**
- **complements**

- **specifiers**
- **adjuncts**

How do we define complements, specifiers and adjuncts?

(i) **complement**: sister to X.

(ii) **specifier**: sister to X', daughter of XP.

(iii) **adjunct**: sister to XP, daughter of XP.

Head

$\rightarrow$ Every phrase contains exactly one head of some category (N, V, Adj, Adv, D...)

$\rightarrow$ Every head is contained within a phrase of the same category.

$\rightarrow$ The head is present in all cases. Whether a complement or a specifier is needed is a case-by-case thing.

$\rightarrow$ The head of a constituent projects its label (lexical category) to that constituent.

$\rightarrow$ The head of a constituent tells us the distribution of that constituent.

$\rightarrow$ The head selects its sister constituent.
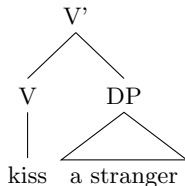
Complements I

→ Complements are selected by the head as their sisters.

→ They are merged into the structure as a result of <u>the complement rule</u>.

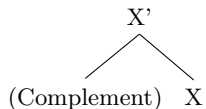Complement Rule (for English and head-initial languages)

X' → X (Complement)

```
        X'
       /  \
      X  (Complement)
```

Example

```
        V'
       /  \
      V    DP
      |    /\
    kiss  a stranger
```
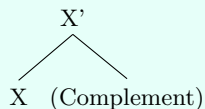
Complements II

Complement Rule (for head-final languages)

X' → (Complement) X

```
        X'
       /\
(Complement)  X
```

Example

(1) taro-wa inu-o   mita
     taro-TOP dog-acc saw
     'taro saw the/a dog' (**Japanese**)

```
        V'
       /\
     DP   V
     /\   |
   inuo  mita
```
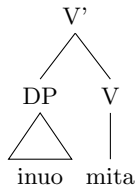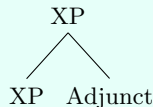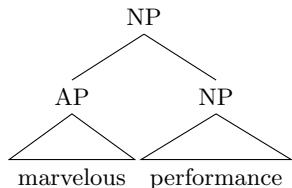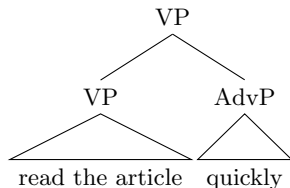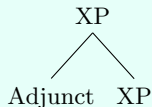
Adjuncts

→ Adjuncts are modifiers which modify the phrase.

→ They are sisters to phrases (XPs, not X' or X!).

→ They tend to be optional.

→ In English they can both precede and follow their sister.

Adjunct Rule.

XP → Adjunct XP   XP → XP Adjunct

```
        XP                                XP
       /\                                /\
      /  \                              /  \
   XP    Adjunct                   Adjunct   XP
```

```
              VP                               NP
             /  \                             /  \
            /    \                           /    \
          VP     AdvP                      AP      NP
         / \      /\                      / \     /  \
        /   \    /  \                     /  \   /    \
  read the article quickly          marvelous  performance
```
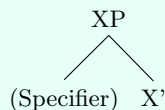
Specifiers

→ Specifiers are merged into the structure as the sister of X'.

Specifier Rule (for English)

$XP \rightarrow$ (Specifier) X'

```
        XP
       /  \
(Specifier)  X'
```
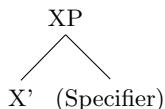
Specifier Rule (for languages such as Malagasy)
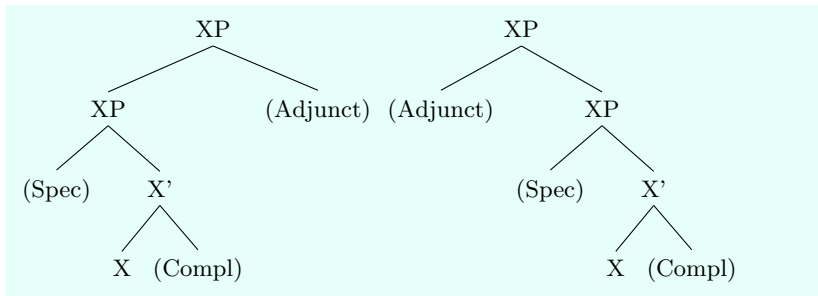
(2) Nahita ny mpianatra ny vehivavy
    saw   the student   the woman
    V   O   S
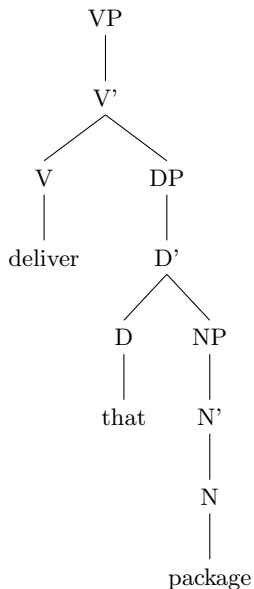   'the woman saw the student'   Malagasy (Austronesian)

$XP \rightarrow X'$ (Specifier)

```
        XP
       /  \
     X'  (Specifier)
```
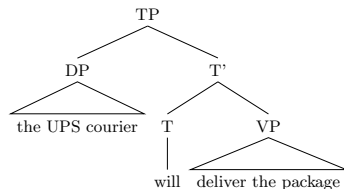
This means that the English syntax is the iteration of:

So [$_{VP}$ deliver that package] now looks like this:

```
                          VP
                          |
                          V'
                         /  \
                        V    DP
                        |     |
                     deliver  D'
                             /  \
                            D    NP
                            |     |
                          that    N'
                                  |
                                  N
                                  |
                               package
```
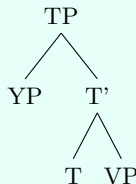
Sentences

Tense Phrases (TPs) I

TP
├── DP
│   └── the UPS courier
└── T'
    ├── T
    │   └── will
    └── VP
        └── deliver the package

The X' schema then will apply to TPs as well:

TP:
- The head: T
- Complement: VP
- Specifier: The subject (DP or CP)

TP
├── YP
└── T'
    ├── T
    └── VP

Tense Phrases (TPs) II

Bound morphemes in tree structures

- the bound morpheme is in T;
+pres or [-pres] (*-ed*, *-s* are also fine!)
- the VP is headed by the bare verbal form.

```
            TP
          /    \
        DP      T'
        /\     /  \
       /  \   T    VP
      Mary       |      |
            [−pres]    V'
                        |
                        V
                        |
                      leave
```
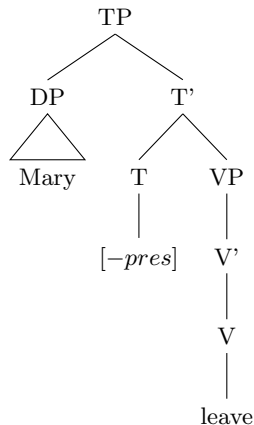
Tense Phrases (TPs) III

<u>Here are the lexical entries:</u>

to	T	free	selects DP/CP	c-selects VP	
will	T	free	selects DP/CP	c-selects VP	meaning: future
[+pres]	T	bound	selects DP/CP	c-selects VP	meaning: present
[-pres]	T	bound	selects DP/CP	c-selects VP	meaning: past

<u>Lexical entries</u> specify the basic properties of the syntactic atoms (heads), including the category and all the selectional properties. Only unpredictable properties that the speaker has to acquire should be included!

<u>Selection</u> = the type of phrase that the head requires in its specifier (or subject) position.

<u>C-selection</u> = complement selection. It tells us what kind of sister a particular head is selecting for.

Complementizer Phrases (CPs) I

- Sentences can be embedded under others. Recursion is one of the core properties of human language.

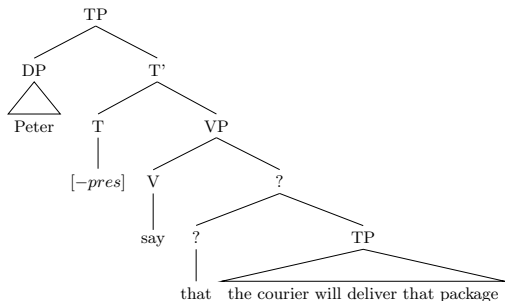 (3) Peter said that [$_{TP}$ the UPS courier will deliver that package]
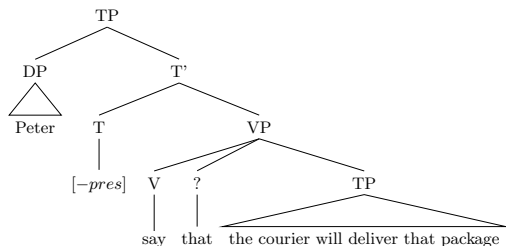
Complementizer Phrases (CPs) II

- We can run a new test we can use: *so* or *this* or *that* replacement.

  (4) Peter said <u>that the UPS courier will deliver that package</u> → Peter said **so**

  This shows that [that the UPS courier will deliver that package] is a constituent but it does not tell us what kind of constituent it is.
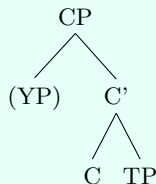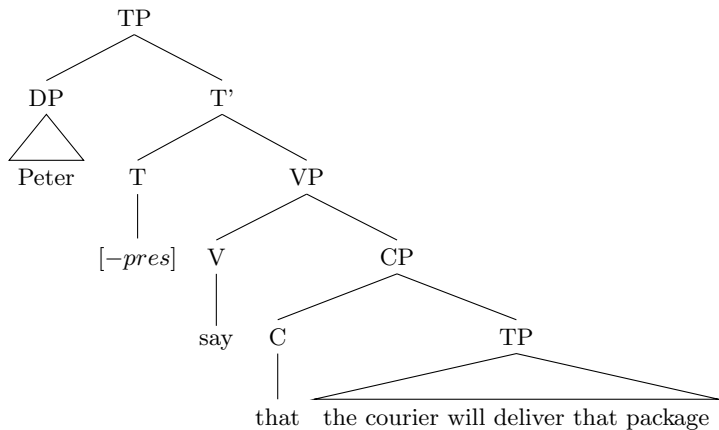
Complementizer Phrases (CPs) III

Here is the proposal:

- the complementizer *that* is the head of this constituent. It selects for a TP complement.

- The X' schema applies to CPs as well:

CP:
- The head: C
- Complement: TP
- Specifier: wh-phrases in wh-questions.

```
            CP
           /  \
      (YP)     C'
              /  \
            C     TP
```
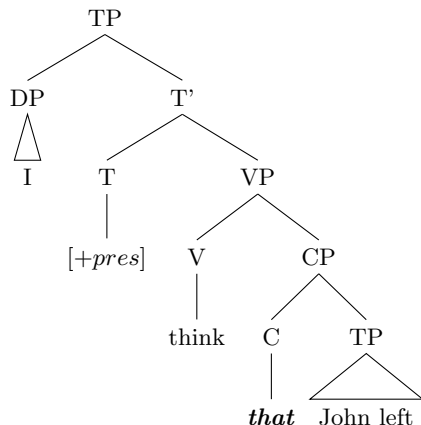
Complementizer Phrases (CPs) IV

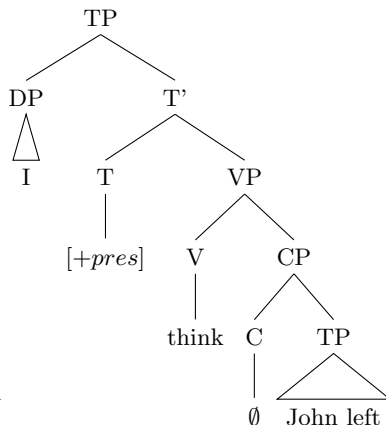# Complementizer Phrases (CPs) V
Realization of C

The C head can be unpronounced in some languages. In English, only '*that*' has this property:

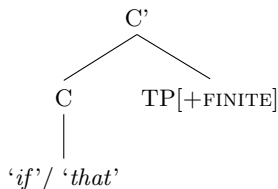(5) I think (that) John left (6) *I wonder/asked John left.

Complementizer Phrases (CPs) VI

C subcategorize for properties of TP.

- '*if*', '*that*', select finite TPs:

  (7) a. I hope [<u>that</u> [he/John win***s***]]
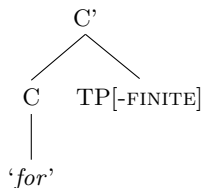   b.*I hope [<u>that</u> [him/John (to) win]]

  (8) a. I wonder [<u>if</u> he/John eat***s*** asparagus]]
      b.*I wonder [<u>if</u> him/John (to) eat asparagus]]

```
            C'
           /  \
          /    \
         C      TP[+FINITE]
         |
      'if'/ 'that'
```

- '*for*' select non-finite TPs:

  (9) a. I would like [<u>for</u> him to eat asparagus]]
      b.*I would like [<u>for</u> him to eat***s*** asparagus]]

Complementizer Phrases (CPs) VII

```
        C'
       /  \
      C    TP[-finite]
      |
    'for'
```

- '*whether*' seems to allow both -tense and +tense TPs:

(10) a. John wonders whether [Mary will win].
     b. John wonders whether [to win].

Here are the lexical entries for the four complementizers above[1]

that/∅	C	free	selects finite +tense TP complement
if	C	free	selects finite +tense TP complement
for	C	free	selects non-finite -tense TP complement
whether	C	free	selects TP complement

Practice: Embedded sentences and ambiguity

Consider the following ambiguous sentence. (i) Draw the two tree structures. (ii) Pair each tree up with the meaning of each sentence. (iii) Explain how at least one of the tests supports the structure of one tree but not the other.

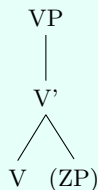(11) That plumber said that that new assistant went to the store quickly.

Verb Phrases (VPs) I

We detect VPs using:

> *do so* replacement test;
> topicalization;
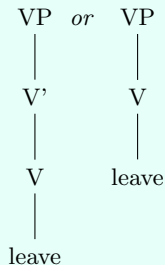> pseudo-clefting
> coordination
> ...

VP:
- The head: V
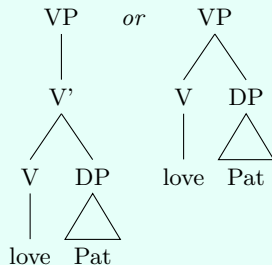- Complement(s): DPs, CPs, PPs, (TPs).
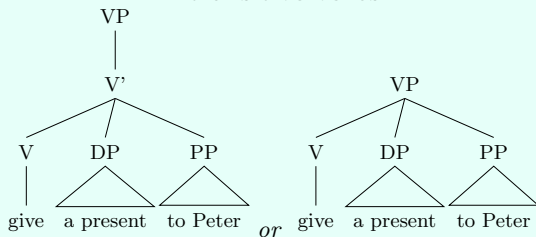- Specifier: none (for now)

```
        VP
        |
        V'
       / \
      V  (ZP)
```

Verb Phrases (VPs) II

Intransitive verbs

VP *or* VP
|
V' — V
|
V — leave
|
leave

Transitive verbs

VP *or* VP

V' V DP
V DP love Pat
love Pat

Ditransitive verbs

VP

V'

V DP PP
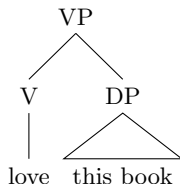give a present to Peter *or*

VP

V DP PP
give a present to Peter

Verb Phrases (VPs) III

→ **Complements** are selected by V. They fill a required slot in the lexical entry of the head.
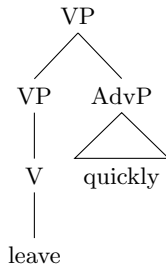
- They are in a *sisterhood relation* to V.

- They tend to be obligatory [*Read the discussion in the framed box at page 112*]

```
        VP
       /  \
      V    DP
      |    /\
    love this book
```

Verb Phrases (VPs) IV

$\rightarrow$ **Adjuncts** are modifiers which modify the phrase.

- They are sisters to phrases.

- They tend to be optional.

```
        VP
       /  \
     VP    AdvP
      |     /\
      V   quickly
      |
    leave
```

Verb Phrases (VPs) V

Now we can go back to last week's puzzle!
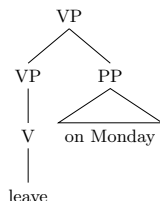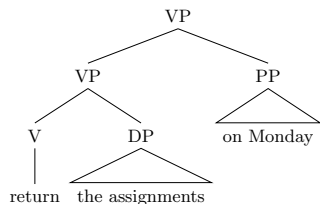
(12) Fred left on Monday.
   →Fred did so on Monday.

(13) Fred returned the assignments on Monday.
   →*Fred did so the assignments on Monday.

If [DP the assignments] is a complement of *return* whereas [PP on Monday] is an adjunct, we obtain the following structures:


***Do-so* can only replace VP. Vs cannot be replaced by do-so!**

Verb Phrases (VPs) VI

Practice
Based on (i) the structural difference between complements and adjuncts and (ii) what you know about *do so* replacement test, explain the results below:

(14) a. The professor will write a poem with a fountain pen.
    b. The professor will do so with a fountain pen.
    c.*The professor will do so a poem with a fountain pen.
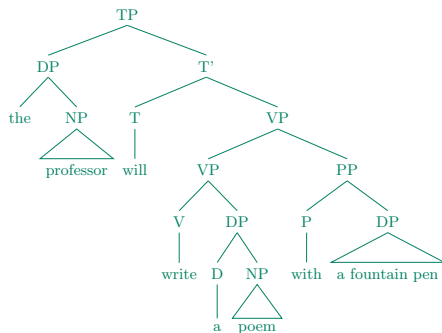
Verb Phrases (VPs) VII
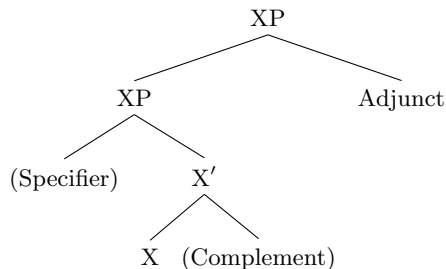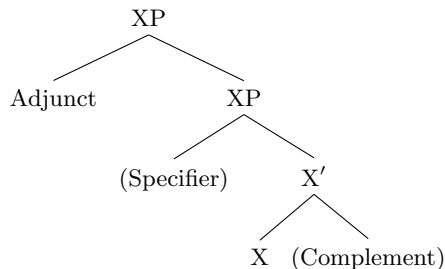
V complement adjunct order

The three rules we have so far:

(i) Complement rule (ii) Specifier rule (ii) Adjunct rule

are able to generate the following phrases:

This makes the prediction that when the adjunct is on the right side, it must follow the complement:

$$\text{Head} >> \text{Complement} >> \text{Adjunct}$$

Verb Phrases (VPs) VIII

(15) a. The professor will [$_V$ write] [$_{COMPL}$ a poem] [$_{ADJ}$ with the fountain pen]
    b.*The professor will [$_V$ write] [$_{ADJ}$ with the fountain pen] [$_{COMPL}$ a poem]

Summary: VP structure

- the head of the VP is the verb;
- *transitive* and *ditransitive* verbs select for complement(s): they are sisters of V (we merge them applying the **complement rule**);
- No specifier in the VP (not yet!)
- Adjuncts attach to the maximal projection (the VP)
- *do so* replaces VP (complement must be included).
    If a phrase need not be included as part of the sequence being replaced by do so, then it is an adjunct. If it must be included, then it is a complement.

Verb Phrases (VPs) IX

Here are the lexical entries:

leave	V	free		example 'John left'
kiss	V	free	c-selects DP	example 'Mary kissed John'
eat	V	free	(c-selects DP)	examples 'John ate (a cookie)'
give	V	free	c-selcts DP, PP	example 'John gave a present to Mary'

Practice: Draw a tree for the following sentence.

Use triangles for DPs, APs and AdvPs.

(16) Miriam thinks that you should definitely show Kevin that new book.