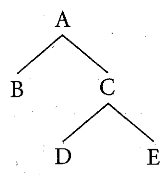
Roots, constituents, and c-command

Robert Frank, Paul Hagstrom, and K. Vijay-Shanker
Johns Hopkins University / Boston University / University of Delaware

1. Background

At the core of syntactic theory is the question of how grammatical structures are properly characterized. It has long been clear that sentences have constituents and hierarchical structure, and these abstract structures have generally been described in terms of trees such as the one shown in (1) for the surface string *BDE* separable into two constituents, *B* and *DE*, where *B* is hierarchically superior to both *D* and *E*.

(1)



In grammatical explanation, the relation of c-command is of fundamental importance; movement is allowed if and only if the moved element c-commands its trace, antecedents must c-command pronouns and anaphors, and so forth. Traditionally, c-command is defined in terms of dominance: α c-commands β iff every node dominating α dominates β (and neither dominates the other). However, the dominance relation is much less central in syntactic explanation than c-command. In fact, our hypothesis here (following Frank & Vijay-Shanker 2001) is that syntactic structures ought to be characterized directly in terms of a primitive c-command relation, as opposed to a primitive dominance relation.

Frank and Vijay-Shanker (2001) show that the class of tree structures that can be characterized in terms of primitive c-command is a subclass of those that can be characterized with primitive dominance including all of the linguistically relevant ones. This match between the expressiveness of primitive c-command and the range of natural language structures provides support for the primitive c-command hypothesis. However, since we take the existence of primitive c-command to imply the non-existence of primitive dominance, we must deal with the consequence that there is no way for grammar to refer to dominance in this sense at all. In this paper, we aim to demonstrate the tenability of our claim that dominance does not figure into grammatical explanation. We show first that the cases in which dominance was traditionally considered crucial can be translated into statements about c-command, and, second, that viewing these cases in terms of c-command furthers our understanding.

Dominance has been used primarily to define two notions, *roots* and *constituents*. In traditional terminology, the root of a tree (e.g., *A* in (1)) is the node which dominates all other nodes, and a constituent (e.g., {*C*, *D*, *E*} in (1)) is the set of nodes all of which are dominated by a specified node (e.g., *C* in (1)). We will consider these two concepts in turn, looking at what makes them important, how they can be described in terms of c-command, and what light it sheds on the phenomena involved.

2. Roots and substitution

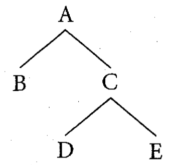
There are two properties that we normally associate with the *root* of a structure like (1). First, the root is the node which determines the category (or features) of the (sub-)tree as a whole. Second, it is the node to which further attachments occur (in a cyclic derivation). Intuitively, the root of a structure is the node that is closest to the top, the node that is least deeply embedded in the structure. We can formalize this intuition with the LESS EMBEDDED relation in (2), which approximates the classical "dominance" relation in a certain range of cases (Frank & Vijay-Shanker 2001).

(2) A node *x* is **LESS EMBEDDED** than a node *y* iff	$x \leq y$ iff
x does not c-command *y*, and	$\neg x \mathrm{C} y \wedge$
every node which c-commands *x*	$\forall z\, [\, z \mathrm{C} x$
also c-commands *y*	$\rightarrow z \mathrm{C} y\,]$

There are two parts to the definition in (2). First, it says that *x* can only be LESS EMBEDDED than *y* if *y* is indeed embedded; if *x* c-commands *y*, then *y* is certainly not embedded below *x* – *x* could not "dominate" *y* in traditional terms. The second part says that *x* can only be LESS EMBEDDED than *y* if everything

that c-commands *x* also c-commands *y*. To give concrete examples, consider some nodes from the tree in (1), repeated below.

(1) (*repeated*)



In terms of primitive c-command, this structure is characterized by the following set of c-command relations:

(3)	*D*	mutually c-commands	*E*	(*D* c-commands *E*, *E* c-commands *D*)
	B	mutually c-commands	*C*	(*B* c-commands *C*, *C* c-commands *E*)
	B	c-commands	*D*	
	B	c-commands	*E*	

Given the definitions in (2), we can see that *C* is LESS EMBEDDED than *D* here because (i) *C* does not c-command *D*, and (ii) everything that c-commands *C* (namely, *B*) also c-commands *D*. The reverse does not hold; *D* is not LESS EMBEDDED than *C* because there is something which c-commands *D* (namely, *E*) that does not c-command *C*. By the same reasoning, we can see that *A* is LESS EMBEDDED than *B* but not vice-versa, since (vacuously) everything that c-commands *A* also c-commands *B*, but there is a node (*C*) which c-commands *B* but not *A*.

To identify the root node of (1), we can use the intuitive idea that the root is the least embedded node in terms of this formal definition of relative embeddedness and conclude that *A* is the root; for any node *N* in (1), *A* is LESS EMBEDDED than *N* and *N* is not LESS EMBEDDED than *A*. The complete list of c-command relations from (1) is given in (3).

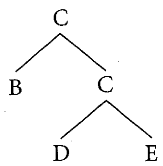
At least for the case of the structure (1), the definition of LESS EMBEDDED in (2) gives us the same results we would have had using traditional "dominance."

3. Roots and adjunction

Adjunction structures differ from the simpler substitution structure (1) considered above in terms of their c-command relations. In (4), *D* and *E* are sisters, and *B* has been adjoined to *C*.[1] Most of the c-command relations are as before, but notice that whereas *B* and *C* stood in a mutual c-command relation in (1),

B asymmetrically c-commands C in (4). That is, an adjunct c-commands its "sister" but not vice-versa (following Kayne 1995: 16).

(4)



(5)	D	mutually c-commands	E	(D c-commands E, E c-commands D)
	B	c-commands	C	
	B	c-commands	D	
	B	c-commands	E	

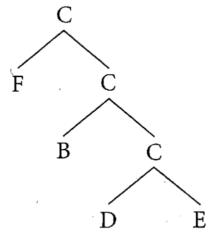
Running through the same sort of calculation as before, we discover unsurprisingly that C is LESS EMBEDDED than both D and E: C does not c-command either D or E, and the only node c-commanding C, namely B, also c-commands both D and E. More interestingly, we also see that B is not LESS EMBEDDED than C, since B c-commands C, but neither is C LESS EMBEDDED than B since there is a node (B) which c-commands C but not B. The question then arises: What is the root of the structure in (4)? There is no node which is LESS EMBEDDED than all other nodes; it cannot be C (since C is not less embedded than B), nor can it be B (since B is not LESS EMBEDDED than C). We would not like to consider (4) to be a rootless structure, since it must always be possible to determine the category of such a tree, and this is one of the functions of the root node.

There is a slightly weaker way we can think of the root of a structure. There are two nodes in (4) for which we can say that there is at least no *other* node LESS EMBEDDED than it. There is no node LESS EMBEDDED than B, nor is there a node LESS EMBEDDED than C. Following this idea, we define the root node as follows.

(6) A node N is a **ROOT** iff there is no other node M such that M is less embedded than N.   N is a ROOT $\leftrightarrow \neg\exists M.M \leq N$

Using this definition, *both* B and C are ROOTS of (4). In fact, generally, this way of looking at roots means that more adjuncts directly leads to more ROOTS in the structure. So, in a structure like (7), the ROOTS are C, F, and B.

(7)



However, even in multi-ROOTed structures like (7), two of these ROOTS have a distinguished status. Specifically, C is the ROOT which does not c-command any other nodes (whereas both B and F c-command C), and F is the ROOT which is not c-commanded by any other nodes (whereas both B and C are c-commanded by F).

Recall that there are two things that the ROOT is important for: determining the label or category of the tree as a whole, and determining the site of cyclic attachment. In (7), C is the node which determines the category, so it is good that we can distinguish C from among the three ROOTS of (7).

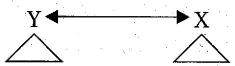
(8) The **CATEGORIAL ROOT** is
the ROOT which does not c-command any other nodes.

Since we can also distinguish F in an equally general way, we expect to see that F plays a similarly important role. We will see shortly that it plays the other major role of the traditional "root" node, determining the site of cyclic attachment. Before we make this connection, however, we must take a detour to discuss how these structures come about derivationally.

4. Substitution and adjunction with multiple ROOTS

Starting with a substitution structure like (9), we can characterize the "merger" (i.e. MERGE, in the sense of Chomsky 1995) of the subtrees <X> and <Y> as the assertion of a mutual c-command relation between every pair of ROOTS, where one member of the pair is taken from <X> and the other is taken from <Y>.[2]

(9)



Substitution

Adjunction, on the other hand, as in (10), only asserts c-command in one direction; so, for every pair of ROOTS (one from <X> and one from <Y> and where the <X> subtree is being adjoined to the <Y> subtree), the <X> ROOT will c-command the <Y> ROOT.

(10) 

Adjunction

Notice too that if we accept that structures are combined using c-command relations (only), then these two modes of attachment, substitution and adjunction, exhaust the logically possible means of combination: c-command can be established either in both directions or in just one direction.

Another interesting result follows from this and from a natural well-formedness condition on trees stated in (11), requiring that the categorical status of the tree be determinable.

(11) **Categorial Identity Condition**
A well-formed tree has a unique category-determining ROOT (a CATEGORIAL ROOT).

Recall that the category of the tree is determined by the CATEGORIAL ROOT, the ROOT that does not c-command any other nodes. This means that an adjunction to a tree cannot change the well-formedness of that tree with respect to (11). In (10), when <X> is adjoined to <Y>, the CATEGORIAL ROOT of <Y> does not c-command any new nodes (since c-command is only asserted from <X> ROOTS to <Y> ROOTS), so it remains the ROOT which does not c-command any other nodes. The <X> ROOT(s) on the other hand now (each) c-command the ROOT(s) of <Y> and so can no longer serve as CATEGORIAL ROOTS.

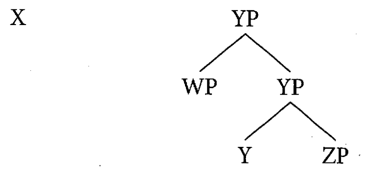
Substitution, however, establishes a mutual c-command relation between the ROOTS of the two subtrees. Saying nothing further, the result of (9) would be a tree which does not satisfy the Categorial Identity Condition (11). In order to have a well-formed tree, a new node must be added to the representation, one which does not c-command any other nodes. This new node is the label of the combined tree; it is the node which "projects" (in the terminology of Chomsky 1995).

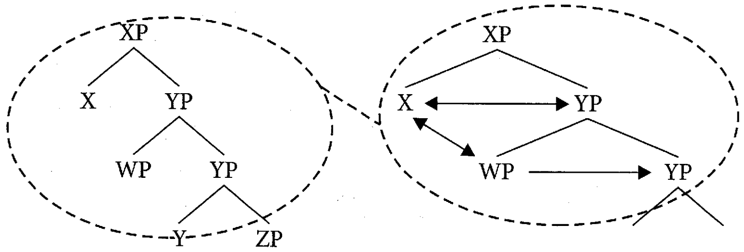
As a consequence of this interaction between the need for a CATEGORIAL ROOT and the mechanisms underlying adjunction and substitution, we no longer have any need to posit a distinction between "segments" and "categories" (May 1977; Chomsky 1986). In effect, "segments" don't exist; no new node is necessary in an adjunction structure. Only under substitution is a new node added to the representation.

Having laid out an overview of the representational system, we will turn in the next few sections to some particular cases of adjunction and substitution in action.

5. Substitution with multiple ROOTS

In (12), a single node *X* is about to be merged with a multiply ROOTED (*YP* and *WP*) structure. This will establish mutual c-command relations between *X* and *YP* and between *X* and *WP*. The result of this merge is shown in (13).

(12) 

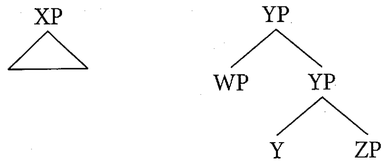
(13) 

As a result of the need to satisfy the Categorial Identity Condition, a node (*XP*) must be added to the structure. At this point *XP* is not in a c-command relation with any other node. It becomes the only ROOT of the resulting structure. One question that arises in this connection is how the properties of the newly projected node are determined; that is, why in (13) does the node labeled XP take its features from *X* rather than *YP*? One possibility is that the choice is free in the syntax, with incorrect choices being filtered out by uninterpretability at the LF interface. An alternative, suggested by Chomsky (2000), is that the possibility of substitution is regulated by the existence of a selection relation. Since this mode of combination does not impose any structural asymmetry, there must be some substantive asymmetrical relation between the combined elements, a relation that Chomsky takes to be selection.[3]

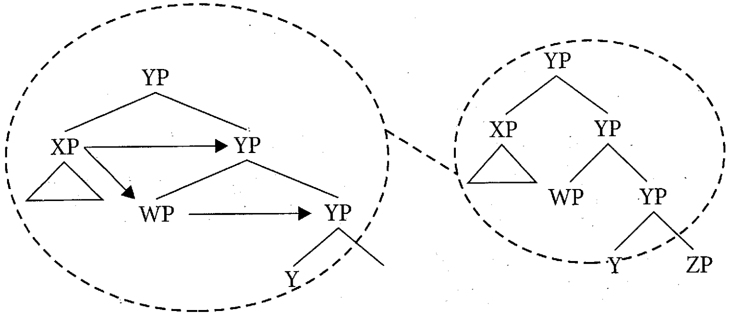
6. Adjunction with multiple ROOTS – XP adjunction

The situation is more interesting where we adjoin a subtree <XP> to the same multiply-ROOTed structure. Whereas in the case above, a mutual c-command relation was established between the ROOTS of the two structures, here only unidirectional c-command relations are established. In the resulting structure (15), *XP* c-commands both *YP* and *WP*. No new node is necessary to maintain well-formedness.

(14)

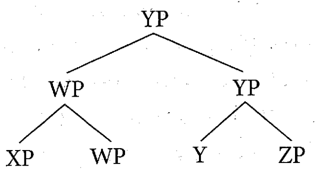


(15)



The interesting thing about the structure in (15) is that the c-command relations are the same as those in the structure in (16). To put it another way, (15) is *nondistinct* from (16).

(16)



We are used to thinking of (16) as coming about as a result of a different derivational history, one in which XP first adjoins to WP and then the complex adjoins to YP (e.g., in the proposals about multiple *wh*-movement in Ackema & Neeleman 1998; Grewendorf & Sabel 1996). People have argued for structures like (16), although the structures have always appeared somewhat odd, seemingly resulting from movement which targets things which should not, cyclically speaking, be targets.

To take one example, consider the well-known case of "Absorption" of *wh*-words and quantifiers, discussed by Higginbotham and May (1981), May (1985). The operation of Absorption, given in (17), turns a structure like (15) into a structure like (16) for the purposes of interpretation.

(17) *Absorption* (May 1985:21, following Higginbotham & May 1981)
$\ldots [\, NP_i\, [\, NP_j \ldots \rightarrow \ldots [\, NP_i\, NP_j\,]_{i,j} \ldots$

However, if representations are described only in terms of c-command, as proposed here, no such operation is required, since the structures are already nondistinct.

Another well-known example of a structure like (16) was proposed by Rudin (1988) to account for the properties of multiple *wh*-movement in certain Slavic languages, including Bulgarian and Romanian. In these languages, all *wh*-words move to the front of their clause, forming an unbreakable constituent. Thus, (18b) is degraded compared to (18a) because the *wh*-words are separated by the adverb *vchera* 'yesterday'.

(18) a. **koj kogo** vchera e udaril
who whom yesterday has hit
'Who hit whom yesterday?'
b. ??**koj** vchera **kogo** e udaril
who yesterday **whom** has hit
'Who hit whom yesterday?' (Marina Todorova, p.c., cf. Rudin 1988)

Rudin (1988) analyzed this as multiple adjunction of *wh*-words to SpecCP, although from our understanding of other cases of *wh*-movement, we expect to find *wh*-words moving to, or targeting, CP itself, rather than SpecCP. Additionally, movement that targets SpecCP also constitutes a violation of cyclicity when understood in terms of the extension condition of Chomsky (1992). From our perspective, however, these two apparently distinct sorts of movement are actually the same (under the further assumption that there is no distinction between specifiers and adjuncts, e.g. as argued by Kayne 1995). We can thereby avoid the need to posit a movement that targets an embedded element.

On the basis of this discussion, one might be tempted to conclude that in our system the locus of attachment for a second adjunction is indeterminate among the multiple ROOTS, with all possibilities yielding identical results. The Bulgarian data just reviewed, however, indicate that that multiply fronted *wh*-elements form a single, unbreakable constituent, suggesting that each successive adjunct attaches to the one that precedes it. Observe that the first adjunct, WP in (14), is not only a ROOT of its structure, but is in fact the unique ROOT that is distinguished by being c-commanded by none of the other ROOTS. We label this distinguished root the ATTACHMENT ROOT.

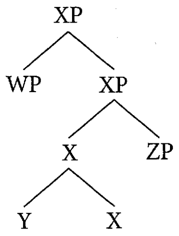
(19) The ATTACHMENT ROOT is
the ROOT not c-commanded by any other nodes.

We will see that the ATTACHMENT ROOT provides the locus of cyclic attachment (the role of the traditional notion of "root" not covered by the CATEGORIAL ROOT).[4]

7. Adjunction and multiple ROOTS – head adjunction

As our final example, we consider the head-adjunction structure in (20).

(20)



The complex head, considered alone as a subtree, has two ROOTS, *X* and *Y*. Recall that the definition of ROOT is based on the intuition that roots are minimally embedded; this means that in (20), *X* and *Y* are (equally) minimally embedded in the complex head subtree.

There is reason to think that both *X* and *Y* are local enough to *WP* in Spec*XP* to check features. For example, consider the licensing conditions on N-words like French *personne*. As is well-known, *personne* requires the presence of a local negative element. This requirement can be instantiated by the assertion that *personne* contains an uninterpretable negative feature ([Neg]) that must be checked for convergence. We assume that the negative head *ne* also contains an instance of [Neg] that is capable of checking *personne*'s feature. Following Pollock (1989) in the assumption that the negative head *ne* is generated between T and V, this would imply that in an example like (21), the Neg head adjoins to T (perhaps in a complex together with the verb), after which *personne* moves to SpecTP.[5]

(21) Personne n'est venu.
No one NEG-is came
'No one came.'

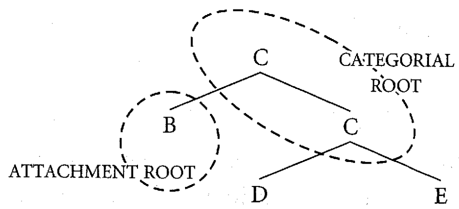
Such a derivation, combined with the empirical fact in (21), entails that the [Neg] feature of *ne* is close enough to *personne* in SpecTP to check, despite the fact that the [Neg] feature is part of a complex head with T and the copula (*n'est*).

Returning to (20), notice that, once this complex head <X> (containing *X* and *Y*) has been merged with the complement <ZP> (the subtree with categorial root *ZP*), both *X* and *Y* c-command <ZP> (*X* and *Y* are both ROOTS of <X> and merging asserts mutual c-command relations between every ROOT of <X> and every ROOT of <ZP>). However, thinking further back in the derivation, if *Y* had moved to adjoin to *X* from within <ZP>, this means that *Y* still c-commands its trace.[6] Looking at it this way allows us to maintain the view that a moved element must c-command its trace.

8. Recap: Rootedness of well-formed structures

Let us return to the issue we started with, the properties traditionally attributed to the root of a tree structure. Recall that the root determines the category or features of the tree as a whole, as well as determining the point at which further cyclic derivational attachments occur. We have seen that in these adjoined structures, the two properties of roots are dissociated; the category determination is taken by the CATEGORIAL ROOT (*C* or *YP* in (22)), while the site of cyclic attachment is determined by the ATTACHMENT ROOT.

(22)



9. Constituents

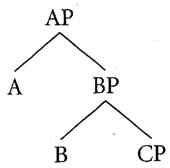
So far, we have concentrated on showing that although dominance was useful for the purposes of identifying the root of a structure, it conflated two notions of "root" which the proposed c-command-based view distinguishes (the site of cyclic attachment and the CATEGORIAL ROOT). We conclude that there is no need to refer to the dominance relation for the purposes of ROOT determination. However, there is another important role traditionally played by dominance: the identification of constituents.

As traditionally understood, a constituent is a collection of nodes that are picked out in some way by one of the nodes in the tree. This is usually accomplished by associating with each node n the constituent that is the collection of nodes dominated by n.

(23) *Traditional definition of constituent*:
For a node r, Constituent(r) = { m | r dominates m }

The question we will address in the next few sections is what becomes of the concept of "constituent" if we re-interpret it in terms of c-command. We will see that we can maintain a close-to-traditional view of constituency, while at the same time providing insight into certain phenomena that have remained puzzling under the traditional view.

Consider the subtree in (24). We want to be sure that whatever our new interpretation of constituency is, it gives us the constituents listed in (25).

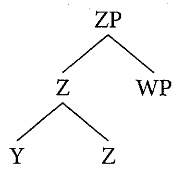
(24)


(25) {AP, A, BP, B, CP} ("AP")
{A} ("A")
{BP, B, CP} ("BP")
{B} ("B")
{CP} ("CP")

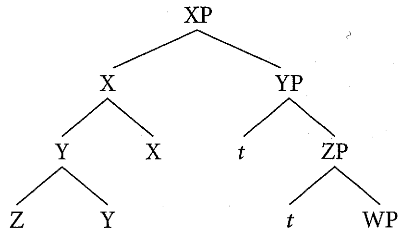
A natural place to begin is to carry over the notion of LESS EMBEDDED, used in the preceding sections to approximate dominance for the purposes of determining ROOTS. The idea would be to define a constituent something like in (26). For reasons we will turn to directly, however, (26) is insufficient as a definition of constituent.

(26) *Revised definition of constituent (first attempt)*:
For a node r, Constituent(r) = { m | r is LESS EMBEDDED than m }

For the structure in (24), the definition of constituent given in (26) gets the correct results. The nodes picked out by *AP* are {*AP*, *A*, *BP*, *B*, *CP*}, those picked out by *BP* are {*BP*, *B*, *CP*}, that picked out by *B* is {*B*}, and so forth. The definition in (26) runs into problems when we consider adjunction structures, such as (27) below.

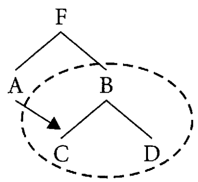
(27)


Recall from earlier discussion that *Y* is not less embedded than *Z*, nor vice versa. As a consequence of this, there is no node r that will pick out exactly the nodes {*Y*, *Z*} under the definition in (26). Rather, we find that the "constituent" picked out by *Z* is {*Z*} and that picked out by *Y* is {*Y*}. This runs counter what we know about the structure of complex heads, however. Specifically, the members of a complex head move together in iterated head movement, so they must form a constituent (assuming that movement can only involve constituents). The facts lead us to expect *Z* to pick out the constituent {*Y*, *Z*}, and the failure of (26) to provide this result is fatal.

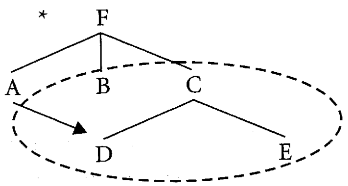
(28)


Y and X move together, hence must form a constituent

We presented (26) as a possibility because it is a natural extension of the preceding discussion, but thinking about constituency in terms of c-command, there is an equally natural alternative conception of constituent: A constituent is the collection of nodes that are picked out by some specific node by virtue of being c-commanded by that node. That is, as in (29) below, *A* picks out the constituent {*B*, *C*, *D*} because these are the nodes that *A* c-commands. Moreover, no node picks out exactly the set of nodes {*A*, *C*, *D*}, which is the desired result since {*A*, *C*, *D*} is not a constituent.

(29)


This definition is nearly correct, but there is one case that requires consideration, illustrated in (30). Were we to be faced with a ternary branching structure, such a definition would incorrectly allow two branches together to count as a constituent, picked out via c-command by the third.

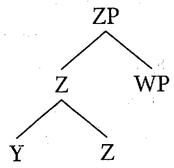
(30) *


A slight refinement solves this problem; we need only require that the "root" of the constituent (*C*, above) does not c-command any of the elements of the constituent (specifically, excluding *B* from the constituent). The final definition of CONSTITUENT we will adopt is given below in (31).

(31) *Revised definition of Constituent (final version)*:
CONSTITUENT(r)= $\{n \mid g \text{ C } n$ and $\neg\, r \text{ C } n\}$ for a node g
such that r C g and g C r.

Walking through (31), it says that the CONSTITUENT with "root" *r* includes those nodes which are c-commanded (or "picked out") by a certain node *g* and are not c-commanded by *r*, where *g* is in a mutual c-command relation with the "root" *r*.

Also, notice what this node *g* in (31) is, structurally. It is, essentially, the *governor* of the constituent. We will return to the significance of this shortly, but first let us try the definition (31) on the adjunction structure that caused trouble for the previous potential definition back in (26). The final definition of CONSTITUENT yields the constituents listed in (32) for (27).

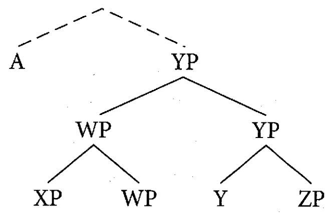
(27) (*repeated*)


(32)

CONSTITUENT(—)	is	governor
ZP	Ø	—
Y	{Y}	WP
Z	{Y, Z}	WP
WP	{WP}	Y, Z

Of particular note, *Z* is not a member of CONSTITUENT(*Y*) (with governor *WP*) because *Y* c-commands *Z*. Also notice that the entire tree is not a CONSTITUENT, since there is no node which can serve as a governor *g* to pick it out.

To return to the discussion of the Bulgarian data from Section 6, recall that (18a–b) suggest that *wh*-words in a multiple question form an unbreakable constituent at the front of the clause. We noted that two possible structures for these questions, (15) and (16), are nondistinct with respect to their nodes and c-command relations. Yet, only (16) provides an explanation for the "unbreakability" of the *wh*-words. Interestingly, the definition of Constituent provided above picks out (16) as the correct representation of these c-command relations. To see this, consider (33), which is (16) merged with another node to provide a governor. The constituents in (33) are given in (34).

(33)


(34)

CONSTITUENT(—)	is	governor
YP	{XP, WP, YP, Y, ZP}	A
WP	{XP, WP}	A
XP	{XP}	A
Y	{Y}	ZP
ZP	{ZP}	Y

The constituency in (34) is consistent with (16) (i.e. (33)) but not with (15).[7]

10. Using constituents: Movement

Now that we have a way to identify constituents in a tree, let us turn to consider why constituents are necessary. Primarily, constituents are the unit on which operations can be performed. We will consider three primary operations here. In this section we will look at movement, returning to ellipsis and conjunction in upcoming sections. We argue that constituents figure into the movement operation in the following way:

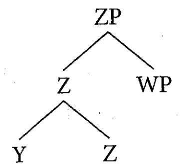
(35) All and only CONSTITUENTS may be moved.

In the remainder of this section, we consider the restrictions on certain subcases of movement, but the explanations of each will conform to a common theme: where movement of a certain part of a tree is not allowed, it is because that part of the tree does not constitute a constituent.

10.1 Head movement

The first case we will consider is that of head movement. Recall from the list in (32) that the X^0-level CONSTITUENTS are $\{Z, Y\}$ and $\{Y\}$, but not $\{Z\}$.

(27) (*repeated*)

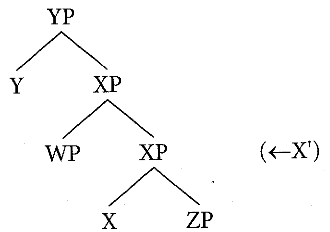


This set of CONSTITUENTS conforms to our expectations based on the observed behavior of head-movement. Moving a complex head is allowed (as predicted, $\{Z, Y\}$ is a CONSTITUENT), as is excorporation of Y out of the complex head (again as predicted, $\{Y\}$ is a CONSTITUENT);[8] however, moving the head away leaving the adjunct behind is impossible (because $\{Z\}$ is not a CONSTITUENT).

10.2 Non-movement of non-maximal projections

The second case we will consider is movement of non-head, non-maximal projections ("X′"). As mentioned earlier, we assume, following Kayne (1995), that there is no meaningful structural distinction between specifiers and adjuncts, or to put it another way, specifiers *are* adjuncts. With this in mind consider the structure given below in (36).

(36)



What would it mean to be able to move X' (the lower XP) in the structure above? It would mean that there would have to be a CONSTITUENT which contains XP, X, and ZP but to the exclusion of WP. There is, however, no such CONSTITUENT. The only CONSTITUENT (governed by Y) which contains all of these nodes also contains WP.

Thus, it is a straightforward prediction of this view that non-maximal projections, or indeed any phrase without adjuncts included, cannot be moved.

10.3 Movement of XPs

Finally, we turn to consider XP-movement, which can be divided into two cases, complement movement and specifier/adjunct movement.

In general, for an XP complement, the head of which it is a complement will act as a governor and license movement. In (36) above, the head Y is a governor for CONSTITUENT(XP) = $\{XP, WP, X, ZP\}$. To put it another way, phrasal complements of a head are always CONSTITUENTS and hence always extractable. In somewhat more traditional terms, we might say that phrasal complements of a head are "head governed" (in the sense of Rizzi 1990); but notice that we need make no statements about licensing of movement over and above the principle in (35) that all and only CONSTITUENTS can be moved.

As noted earlier, a consequence of our conception of constituent is that unembedded structures do not form CONSTITUENTS. In the current context, this leads us to expect that unembedded structures cannot undergo movement. Though such a result might strike the reader as tautological, we would like to suggest that it allows us to derive the stipulation made by Chomsky (2000) to the effect that feature checking cannot obtain under so-called "pure Merge", i.e., Merge that does not occur as a subpart of the movement operation. Chomsky's motivation for this stipulation is to prevent a DP merged into the specifier of *v*P as an external argument from checking accusative case. To see how this follows, consider the mechanism underlying licit cases of feature checking by phrasal elements, as occurs in feature-driven movement. First, the feature in need of checking, F say, identifies some other feature F′ that can check it. Since it is assumed that bare features are inaccessible to the movement operation, the derivational system must next determine what the minimal structure S is which contains F′ and which may be moved.[9] Under the assumptions that we are making in which syntactic structures consist of a set of nodes and c-command relations among them, determining such an S is a non-trivial problem, as these structures do not directly specify a notion like "containment". The most natural way to solve this problem is to exploit the conception of constituent under discussion. Suppose now that the feature F′, rather than coming from within the same syntactic structure as that which includes F, is present in an independently constructed structure S′. Suppose further that F′ is part of the head whose features project the (categorial) ROOT of S′. We might expect that F (or more properly its projection) should then Merge with S′. If, however, we require that this S′ be identified as a CONSTITUENT, such an instance of Merge will be impossible, as there is no governor for S′.[10]

As things stand so far, we also predict that XP-adjuncts and specifiers should be CONSTITUENTS and thus should be capable of movement quite generally. For example, in (36), CONSTITUENT(*WP*) = {*WP*} with governor *Y*, analogous to excorporation from a complex head as discussed above. However, we know empirically that movement of specifiers and adjuncts is much more limited than movement of complements. This is an issue which we will discuss in some depth after summarizing the basic proposal with respect to constituency.

10.4 Conclusion

In the preceding sections, we have proposed that the constituent status of each of the following entities is as listed below:

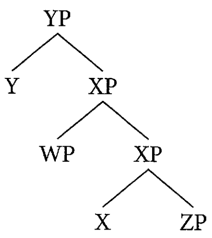
(37) a. An X^0 and everything adjoined to it (X^{0max}, in the terminology of Chomsky 1995) is a CONSTITUENT.
b. A maximal projection is a CONSTITUENT (*except* at the root).
c. A non-maximal projection **is not** a CONSTITUENT.

The existence of a *governor* plays a crucial role in defining a CONSTITUENT. This allows us to understand why we seemed to need the (head government condition of the) ECP; it is simply a result of the fact that CONSTITUENTS are only defined with respect to a governor, combined with the constraint that only CONSTITUENTS can move.

11. Movement of specifiers, constituency and conditions on government

As noted in Section 10.3, the proposal sketched above does not distinguish between the constituency and hence extractability of complements and specifiers/adjuncts. In (38), both *XP* = {*XP*, *WP*, *X*, *ZP*} and *WP* = {*WP*} are CONSTITUENTS, given that they both stand in a mutual c-command relation with the head Y.

(38)



Given this, what yields the well-known complement/adjunct and complement/subject asymmetries in extraction? In the subsections below, we speculate on a number of possibilities.

11.1 Possibility I: Restrictions on possible governors

Rizzi's (1990) explanation of contrasts between examples like (39) and (40) distinguishes *saw* from *that* in the ability to head-govern the trace of movement.

(39) Who do you think that John saw *t*?

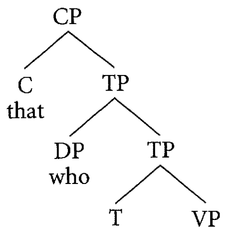
(40) *Who do you think that *t* saw John?

This proposal can be readily reformulated in our terms, by restricting what counts as a possible node *g*:

(41) CONSTITUENT(r)= $\{n \mid g \mathrel{C} n$ and $\neg\, r \mathrel{C} n\}$ for a node g such that $r \mathrel{C} g$ and $g \mathrel{C} r$ **where g is a governor.**

(42) Lexical verbs are governors. *That* is not a governor.

We will also include in our set of potential governors certain functional heads, e.g. empty complementizer and T, so as to allow for extraction of subjects (from *that*-less CPs) and adjuncts, respectively. Under this view, the source of *that-t* effects is the inability of *that* to define a CONSTITUENT, leaving the DP in (43) as a non-CONSTITUENT.

(43)


Notice that this also implies that the TP complement here is not itself a CONSTITUENT either, since the C node *that* would also play a crucial role in defining the TP constituent, if it existed. However, there is at least some evidence to support the hypothesis such TP complements are not CONSTITUENTS.

First, as seen in (44), note that TP, when the complement of *that*, cannot be extracted. Second, such a TP cannot be elided, as shown in (45) (cf. Lobeck's 1995 proposal that elided constituents must be governed; under our proposal this would simplify to "only CONSTITUENTS can be elided").

(44) *John left, I was told that *t*.

(45) *Even though Mary hopes that *e*,
she doubts that Bill is coming to the party.

Similar behavior is observed with the wh-complementizer *if*, which also fails to license extraction from the specifier of its TP complement:

(46) *John finished his novel, I asked Mary if *t*.

(47) *Even though Mary asked if *e*,
she is pretty sure that Bill won't be coming.

In the presence of empty complementizers, which license extraction of embedded subjects, both the extraction and elision of TP are possible:[11]

(48) John left, I was told [$_{CP}$ [$_{C}$ Ø] *t*].

(49) Even though Mary asked who [$_{CP}$ [$_{C}$ Ø] *e*],
she is sure that Bill is bringing Louise.

One potential problem with the hypothesis that TP is not a CONSTITUENT is the fact that TPs appear to be conjoinable (50).

(50) I would never have believed that [Mary would leave] and
[Bill would stay]

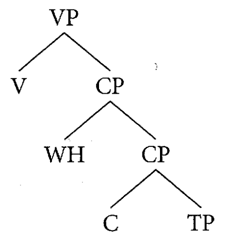
This becomes a non-problem, however, if we accept the proposal of Wilder (1994), who argues that conjuncts must be extended projections (CPs here) with ellipsis of repeated lexical material.

(50′) I would never have believed [that Mary would leave] and
[~~that~~ Bill would stay]

11.2 Possibility II: Selectional restrictions

A somewhat less conservative alternative approach to restricting the constituency, and hence extractability, of an embedded specifier builds on the selectional properties of the embedding head.

Our proposal entails that phrases with specifiers have multiple ROOTS (i.e., multiple minima in the LESS EMBEDDED THAN relation). So, for example, in (51), the CP complement of V has two ROOTS, the CATEGORIAL ROOT CP (which c-commands no other node in <CP>) and WH.

(51)


Syntactic heads have selectional properties which specify what type of complement they require. When considering the complement of V in (51), what property does it have that satisfies the selectional requirements of V? Certainly, selection must attend to the CATEGORIAL ROOT, which determines that <CP> has the categorial properties of a CP. Suppose, however, that there are two kinds of selection. The first dictates properties of the ROOT(s) of a proper complement, the second dictates properties of the CATEGORIAL ROOT (only) of a proper complement.

Consider, as an example, the difference between *say* (a bridge verb) and *whisper* (a non-bridge verb). If we suppose that *say* has the property that its complement must be a CP (interpreted as meaning that the CATEGORIAL ROOT of its complement must have the categorial features of a CP), whereas *whisper* has the (stricter) property that (all of) the ROOT(s) of its complement must have the categorial features of a CP (52), we derive the result that *whisper* does not tolerate successive-cyclic extraction through the specifier of its CP complement. Specifically, if a *wh*-word left an intermediate trace in the specifier of a

CP complement of *whisper*, the selectional requirement of *whisper* would be violated: The intermediate trace would itself constitute an additional ROOT of the CP complement, yet would not have the categorial features of a CP. The selectional properties of *whisper* in (52) practically forbid its CP complement to have any specifiers.[12]

(52) *Selectional properties of bridge verbs vs. non-bridge verbs*
 a. *say* CATEGORIAL ROOT: CP
 b. *whisper* ROOT: CP

(53) Why did you say/*whisper [t' that you had left early t]

As a second example of how this distinction between selection for a ROOT vs. selection for a CATEGORIAL ROOT, consider the difference between raising and control verbs.

(54) *Selectional properties of raising verbs vs. control verbs*
 a. *seem* CATEGORIAL ROOT: TP
 b. *try* ROOT: TP

Assuming that NP-movement but not control depends on the possibility of movement through specifier of TP (*pace* Hornstein 1998), only *seem* will allow it given the selectional properties above. Parallel to the bridge/non-bridge distinction, *try* effectively selects for a TP *with no specifier*, whereas *seem* selects for a TP regardless of whether it has a specifier.

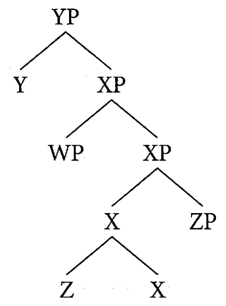
If this approach to the raising/control distinction is on the right track, there are a couple of implications worth mentioning. First, it avoids the otherwise unmotivated TP/CP distinction between the infinitival complements to raising and control verbs. A second implication is that "EPP" (in the sense of "TP must have a specifier" as the EPP is interpreted following Chomsky 1992 *et seq.*) cannot be a universal property of T, since under this proposal, TP complements of control verbs lack specifiers altogether.

11.3 Possibility III: Derivational symmetric c-command

Finally, let us consider a third alternative (as a proposal for how to restrict extraction of specifiers while allowing extraction of complements), one which is even more radical (and more speculative). In our discussion above (Section 10.3), we suggested that the constituency of a specifier (as well as the constituency of an adjoined head) derives from the existence of a mutual c-command relation with an "external" element. So, in (55), *WP* (the specifier of

XP) was taken to be a CONSTITUENT in virtue of the fact that it is in a mutual c-command relation with governor *Y*; likewise, *Z* is a CONSTITUENT in virtue of the fact that it is in a mutual c-command relation with governor *ZP*.

(55)



Earlier in (9–10), we took these mutual c-command relations to result from the multi-ROOT characterization of the merging of <X> and <Y>: *all* ROOTS of <X> c-command *all* ROOTS of <Y> (and vice versa). In this section, we will consider an alternative statement of the Merge operation:

(56) Alternative version of Merge (single-ROOT version):

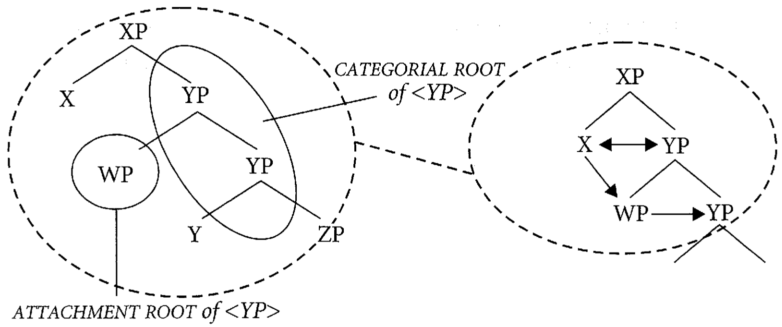
$$\begin{array}{ccc} X & \longleftrightarrow & Y \\ \triangle & & \triangle \end{array}$$

where x_C is the CATEGORIAL ROOT of <X>, and
where y_A is a ROOT of <Y>,
then x_C c-commands y_A

where y_C is the CATEGORIAL ROOT of <Y>, and
where x_A is a ROOT of <X>,
then y_C c-commands x_A

Consider what happens when merging <X> and a multi-ROOTed structure <YP>, as in (57) below, using this revised definition of Merge. <X> has only one ROOT node (*X*). <YP> has two ROOTS, *YP* being the CATEGORIAL ROOT. Merging <X> and <YP> will therefore establish three c-command relations according to (56): *X* c-commands *WP*, *YP* c-commands *X*, and *X* c-commands *YP*.

(57)

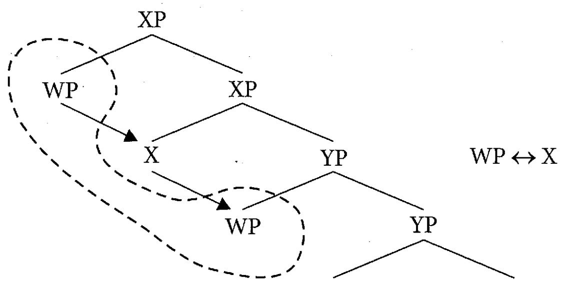


Notice in particular that *WP* does not c-command *X*, which means that *X* cannot serve as a governor to define *WP* as a CONSTITUENT. That is, *WP* is *not* a CONSTITUENT in (57). This is a step toward the solution of our problem (that is, how to restrict the movement of specifiers), since specifiers and complements are no longer symmetrical; specifiers are not necessarily CONSTITUENTS, whereas complements are. If only CONSTITUENTS can move, then we are not guaranteed to be able to move a specifier.

Of course, we must not stop here, since it is certainly not the case (empirically) that specifiers can *never* move. Given that specifiers *can* sometimes move, and continuing to assume that only CONSTITUENTS can move, it must be the case that *WP* can become a CONSTITUENT in certain circumstances. For *WP* to become a CONSTITUENT means essentially that *WP* must come to c-command *X*. The question is: How might this come about?

Our suggestion is (somewhat paradoxically) *movement*. That is, as illustrated in (58), if *WP* were moved higher, the *WP* would c-command *X*. At this point, *WP* would be a CONSTITUENT (with governor *X*), since *WP* is in a mutual c-command relation with *X*. Put another way, movement is allowed if in the *end* what moved is a CONSTITUENT.

(58)



Opening up this possibility of "post-hoc" licensing of movement seems in one sense to have put us back where we started; any movement of a specifier to a c-commanding position would be licensed, since after the movement, there will a mutual c-command relation between the moved specifier and a governing node. So, what then accounts for the *restrictions* on the movement of specifiers? Our proposal is that movement of specifiers is in general *allowed*, though not for cases where there is no local landing site available for the movement. Under this hypothesis, the locus of the restriction on movement of specifiers is to be found in the locality theory; let us suppose that the relevant locality domain is the "phase" as proposed by Chomsky (2000).

This view implies, of course, that observed legitimate cases of movement from specifier involve only very local moves. This seems plausible. For example, in TP complements of ECM verbs, the embedded subject moves only as far as the accusative Case position (say, Spec*v*P). No phase boundary intervenes (phase boundaries being CP and *v*P). Similarly, if we suppose that raising verbs like *seem* have no *v*P projection (only a VP projection), then no phase boundary is introduced at *seem* and the higher TP remains local to elements in the complement of *seem*.

As another example which suggests that movement must be phase-local, consider the following agreement fact in Chamorro (as reported by Chung 1994:17). In (59), a *wh*-word has been moved from the object position in the embedded clause. Agreement morphology must appear on all verbs along the path of movement. In (59), the embedded clause is only a TP, yet the successive-cyclic agreement affects it as well. If *wh*-agreement is a reflex of having the trace of *wh*-word in Spec*v*P, the facts are explained.[13]

(59) Hafa malago'-mu [*t'* u-mafa'maolik *t*]?
what wh[OBL].want-AGR wh[NOM].AGR-be.fixed
'What do you want to be fixed?'

Lastly, what is the status of the principle that says "only CONSTITUENTS can move" in light of the hypothesis that something can *become* a CONSTITUENT via movement? One possibility is that the notion of CONSTITUENT is *not* a fundamental part of *syntax*, but rather an important requirement on the part of *pronunciation*. That is, perhaps constituency is verified at and useful for PF, while syntax (and even perhaps LF) are not sensitive to issues of constituency. If this speculation is on the right track, we would expect to find that post-Spell-Out movement (of specifiers, for example) would be unconstrained compared to overt movement (since it would no longer be important to move only constituents. This also raises an interesting possible take on why the "phase" would be the relevant locality domain as well. Suppose that constituency is necessary for pronunciation (e.g., to determine the location of prosodic boundaries), and suppose that the linguistic structure is handed off to the PF interface at every phase. We might then conclude that phase-internal movement of an XP is required for pronunciation if the XP in its base position is not itself a CONSTITUENT; this predicts a difference between direct objects and other arguments (which are presumed to be introduced into the structure in specifier position) in whether they must undergo phase-internal movement. Notice also that this need not be movement "to the edge" of the phase; it need only be movement to a position c-commanding the would-be governor, but within the phase. This makes a slightly different set of predictions about the impact of phases on derivations than the view proposed in Chomsky (2000) (for example, it predicts that direct objects need not move successive-cyclically, while subjects, adjuncts, and indirect objects must, and it predicts that movement to a c-commanding position further inside a phase boundary is possible for those things which must move successive-cyclically). Further pursuit of this and other implications of these suggestions are left for future study.

Notes

1. Note that we represent adjunction structures as is standard, involving two segments of the single category to which adjunction takes place. In primitive c-command terms, however, there is no corresponding distinction between segment and category. Rather, it is simply the case that the adjunct (asymmetrically) c-commands the node to which it is adjoined.

2. We will use the notation <X> to label subtrees in the text to distinguish them from node labels. <X> is a collection of nodes and c-command relations, whose CATEGORIAL ROOT is X.

3. For Chomsky (2000), the operation involved in substitution is what he calls "set merge", an operation that puts two structures into a single set. This is a notational variant of our proposal that substitution involves the assertion of mutual c-command. One sees these two variants in different formalizations of undirected graphs in mathematics: they may either be characterized by a symmetric relation on the set of nodes (a set S of pairs such that $(x,y) \in S$ iff $(y,x) \in S$) or by a set of 2-subsets of the set of nodes. In the case of adjunction, Chomsky's formulation of the operation as "pair merge" is identical to ours, as he takes it to involve the creation of an ordered pair. This is nothing but the addition of a single assertion to a (c-command) relation.

4. In Section 9, we show how the notion of constituent that we develop derives the conclusion that the structure that is derived from successive adjunction is not indeterminate between attachment at any of the ROOTS, but instead involves attachment at the CATEGORIAL ROOT. In other words, while (15) and (16) are nondistinct, constituency tells us that (16) is the correct way to draw the structure.

5. Example (i) below shows that [Neg] on *ne* need not be checked via overt movement, and is therefore not a strong/selectional feature. This eliminates the possibility of an alternative derivation for (21) in which *personne* moves first to SpecNegP, checking [Neg] on the Neg^0 head, after which *personne* moves on to SpecTP and Neg^0 moves on to head-adjoin to T^0. Such a derivation would not make our point, since there is a stage in the derivation where Neg^0 and *personne* would be in a direct Spec-head relation.

(i) Je n'ai vu personne.
I NEG-have seen noone
'I did not see anyone.'

6. Given the formulation of adjunction given in the text, this result will require the sort of interarboreal derivation discussed in Bobaljik and Brown (1997). An alternative to this approach might invoke a transitivity condition on the c-command relation of the sort in (i):

(i) For all nodes x, y, z, if xCy and yCz and not $x \leq z$, then xCz.

7. This has the further implication that in cases we may have thought were multiple adjunctions to XP, such as multiple adverbial phrases, are not. That is, although we can draw a tree like (15), the constituents will always come out as in (16). This result meshes with Kayne's (1995) proposal, under which multiple adjunction to XP is impossible, and forces us to a view along the lines of that proposed by Cinque (1999) for multiple adverbs.

8. Whether excorporation is allowed at all or in only these cases is under debate; see Roberts (1991) for discussion and an opposing view.

9. As we tentatively suggest in Section 11.3, the need for identifying a moveable CONSTITUENT may only pertain to structures that need to be pronounced, however.

10. An immediate question raised by this proposal is how expletive *there* is able to check T's EPP features under pure Merge. One possibility might build on the idea that *there* is a radically impoverished lexical item, being the realization of the single categorial feature D. Perhaps as a consequence the derivation can avoid the step of identifying a containing CONSTITUENT, as the bare feature itself is moveable. We would expect, then, that all such cases of feature checking under Merge will involve such radically impoverished lexical items.

11. To accept that (48)–(49) are the correct structures for these sentences requires that the empty complementizer is not being moved/elided in these cases. One argument, admittedly indirect, for the structure in (48) might center on Stowell's (1981) proposal that empty complementizers must occur in governed positions (to explain the fact that we get them in object CP complements, but not with sentential subjects). If this is right, the empty C couldn't front. Note, however, that this notion of "needing a governor" differs from the notion of "needing a governor" to be a CONSTITUENT; here, it is something like a selectional requirement, determining whether the empty complementizer can appear in the structure. Concerning (49), note that, although (i) is possible, (ii) is unexpectedly ill-formed. This indicates that something more complicated is going on in elision of this kind, but we have nothing further to offer here.

(i) Mary hopes (that) John will leave.

(ii) *Even though Mary hopes [$_{CP}$ [$_{C}$ Ø] *e*], she doubts that Max is coming to the party.

12. There is a loophole, however. If the phrase which occupies SpecCP in the complement of a non-bridge verb like *whisper* is *itself* a CP, then the selectional property of the verb could be satisfied despite the fact that its complement CP has a specifier. Sufficiently developed, this might make interesting predictions for languages like Basque (Ortiz de Urbina 1989), or Quechua (Cole 1982), in which "clausal pied piping" is allowed in *wh*-movement. For example, in Basque, *wh*-arguments can in general either move to the matrix SpecCP alone (i-a), or take the embedded clause along (i-b). We might expect to find that a version of (i) with a non-bridge verb would show obligatory clausal pied-piping, because its complement cannot have a specifier, yet without clausal pied-piping the *wh*-argument would have to land in the specifier of the embedded clause on its way to its matrix position. More interesting would be a case structured like (ii); here we would predict (given certain assumptions) that (ii-a) would be ill-formed because the DP must land in the specifier of the complement of *whisper* whereas in (ii-b) the entire subordinate clause stopped there. In (ii-b), *whisper*'s selectional requirements would be met, since what is in SpecCP of the complement of *whisper* is itself a CP. However, not only do we not have the relevant Basque data, it is also not completely clear that the assumptions necessary for these predictions to fall out are warranted; more research must be done before any solid conclusions can be reached. For example, it is not clear whether *wh*-words must move internal to a pied-piped CP (Ortiz de Urbina 1990) or not (Echepare 1995), nor is it clear that *wh*-words and pied-piped CPs share the same landing site (Echepare 1995). Development of this area must await further research.

(i) a. nor$_i$ esan duzu [t$_i$ etorri dela] ?
who say AUX arrive AUX-comp

b. [nor etorri dela]$_i$ esan duzu t$_i$?
who arrive AUX-comp say AUX
'Who did you say arrived?'

(ii) a. *who$_i$ whisper [t$_i$′ John say [t$_i$ arrived]] ?
(prediction, given certain assumptions)

b. [who arrived]$_i$ whisper [t$_i$′ John say t$_i$] ?

13. Others who have recently provided evidence for a VP-level trace of *wh*-movement include Fox (1999) (arguing on the basis of interactions between Condition C and scope reconstruction) and Nissenbaum (1998) (arguing based on properties of parasitic gap constructions).

References

Ackema, P. & A. Neeleman (1998). Optimal questions. *Natural Language and Linguistic Theory, 16*, 443–490.

Bobaljik, J. D. & S. Brown (1997). Interarboreal Operations: Head movement and the extension requirement. *Linguistic Inquiry, 28*, 345–356.

Chomsky, N. (1995). *The Minimalist Program*. Cambridge, MA: The MIT Press.

Chomsky, N. (2000). Minimalist Inquiries: The framework. In R. Martin, D. Michaels, and J. Uriagereha (Eds.), *Step by Step, Essays in Minimalist Syntax in Honor of Howard Lasnik*. Cambridge, MA: The MIT Press.

Chung, S. (1994). Wh-agreement and Referentiality in Chamorro. *Linguistic Inquiry, 25*, 1–44.

Cinque, G. (1999). *Adverbs and Functional Heads*. Oxford: OUP.

Cole, P. (1982). *Imbabura Quechua*. The Hague: North Holland.

Echepare, R. (1995). A Case for Two Types of Focus in Basque. In E. Benedicto, M. Romero and S. Tomioka (Eds.), *Proceedings of Workshop on Focus* [UMOP 21]. Amherst, MA: GLSA.

Fox, D. (1999). Reconstruction, Binding Theory, and the Interpretation of Chains. *Linguistic Inquiry, 30*, 157–196.

Frank, R. & K. Vijay-Shanker (2001). Primitive c-command. *Syntax, 4*(3), 164–204.

Grewendorf, G. & J. Sabel (1996). Multiple Specifiers and the Theory of Adjunction: On scrambling in German and Japanese. *Sprachwissenschaft in Frankfurt Arbeitspapier* 16. Johann Wolfgang Goethe-Universität, Frankfurt am Main.

Higginbotham, J. & R. May (1981). Questions, Quantifiers and Crossing. *The Linguistic Review, 1*, 41–80.

Hornstein, N. (1998). Movement and Chains. *Syntax, 1*, 99–127.

Kayne, R. (1995). *The Antisymmetry of Syntax*. Cambridge, MA: The MIT Press.

Lobeck, A. (1995). *Ellipsis*. Oxford: OUP.

May, R. (1985). *Logical Form: Its structure and derivation*. Cambridge, MA: The MIT Press.

Nissenbaum, J. (1998). Movement and Derived Predicates: Evidence from parasitic gaps. In U. Sauerland and O. Percus (Eds.), *The Interpretive Tract* [MITWPL 25]. Cambridge, MA: MIT Working Papers in Linguistics.

Ortiz de Urbina, J. (1989). *Parameters in the Grammar of Basque*. Dordrecht: Foris.

Ortiz de Urbina, J. (1990). Operator Feature Percolation and Clausal Pied-piping. In L. Cheng and H. Demirdash (Eds.), *Papers on Wh-movement* [MITWPL 13]. Cambridge, MA: MIT Working Papers in Linguistics.

Pollock, J.-Y. (1989). Verb Movement, Universal Grammar, and the Structure of IP. *Linguistic Inquiry, 20*, 365–424.

Rizzi, L. (1990). *Relativized Minimality*. Cambridge, MA: The MIT Press.

Roberts, I. (1991). Excorporation and Minimality. *Linguistic Inquiry, 22(1)*, 209–218.

Rudin, C. (1988). On Multiple Questions and Multiple WH Fronting. *Natural Language and Linguistic Theory, 6*, 445–501.

Wilder, C. (1994). Coordination, ATB and ellipsis. Ms., ZAS, Max-Planck-Gesellschaft, Berlin.