Parallel Work Spaces in Syntax and the Inexistence of Internal Merge

Abstract: If we can maintain that the Work Space in which a phrase is built up prior to being initially merged in a derivation does not 'perish' with that operation but may be availed of for remerge, we do not need Internal Merge. We can simplify Grammar in other ways too: PIC, the 'escape hatch', partial spell out of phrases etc. can be dropped from the theory.

Key words

internal merge

phrasal movement

pied-piping

PIC

Word Count

8,030 words

1. Introduction

As is well-known, current minimalist theory does not make a distinction between Move and Merge. Move is only remerge, and nothing more.

It seems to be commonly agreed that the first-time merge of a phrase is from a different Work Space where the phrase is built up; for this reason, it is called External Merge. The picture we have is that of different Work Spaces that exist in parallel in the mind. One of these is of course dedicated to the main derivation. Regarding the other Work Spaces, there appears to be an assumption – which has never been articulated or examined, to my knowledge – that they are (in a peculiar way) ephemeral: Once the phrase built up in a certain Work Space has been plugged into the main derivation, that Work Space disappears. For this reason, when we want to remerge a phrase, we are forced to source it from its merged position in the main derivation; this is why remerge is called Internal Merge. Even when we want to remerge a phrase with a syntactic object that exists completely independently in another Work Space – as in proposals about sideward movement/parallel merge – we source it from its merged position, because the Work Space where it was originally built up has disappeared.

But the recovery of a phrase from somewhere within an already built-up structure for the purpose of remerge is not a trivial matter. The problems attendant on this operation have in fact been realized and commented on (occasionally!); but they have been largely ignored because there seemed to be no way to avoid them – if we wanted to keep Move at all in the Grammar.[1] But if we can say that the ‘other’ Work

[1] See Koster (2007) for an awareness of the problems of Internal Merge. We examine these problems in § 2.3. (See also de Vries 2013:146.)

Spaces need not all be so ephemeral, but that they can stay and be used for remerge, we get a very different picture of Move: we can dispense with Internal Merge. In fact the consequences for the organization of Grammar are immense. This is the main thesis of this paper.[2]

The paper is organized as follows. In § 2, we state our proposal regarding remerge from parallel Work Spaces; we clarify some points regarding the content of these parallel Work Spaces; and then discuss the problems of Internal Merge. In § 3, we illustrate how our system works by showing how we handle pied-piping. In § 4 and § 5, we address two prima facie challenges to our proposal, namely successive-cyclic movement and extraction islands, and show how we can deal with them in our system. In § 6, we discuss parasitic gaps. In § 7, the conclusion, we argue that PIC, the "escape hatch", and 'partial' spell-out of phases can be dropped from the theory.

2. The proposal

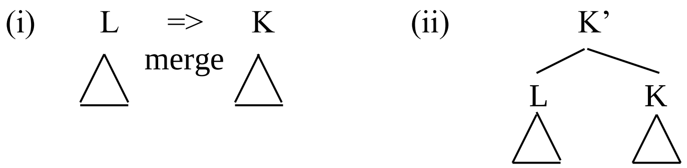
2.1. The idea of remerge from parallel Work Spaces

Consider the situation of L merging to K (external merge). If L is a lexical item, it is a copy of a member of the Numeration. But if L is a phrase, it must have been built up in a different Work Space (Nunes & Uriagereka 2000, Collins 2002, Stroik 2009, and many others).[3] Currently we think of it as being moved into a position in K' (extension of K):

[2] See Epstein, Kitahara & Seely (2014) for important additional arguments for unifying External and Internal Merge.

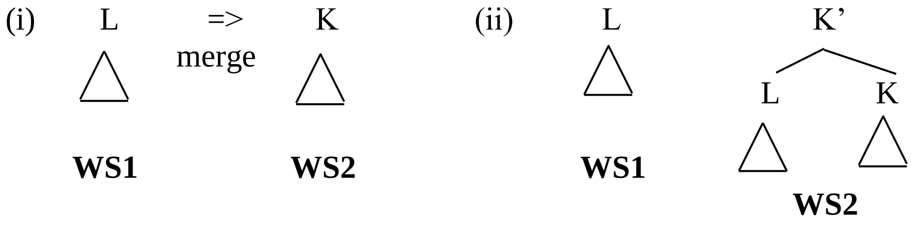
[3] Presumably, each such 'derivational space' has its own Numeration or sub-array.

(1)



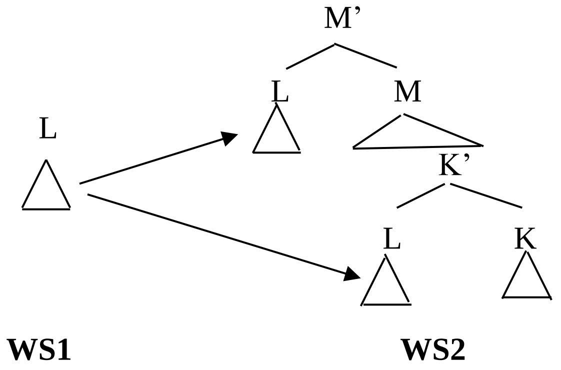
But there is another way to conceptualize this operation: L is merged from the Work Space where it is built up; but this Work Space (containing L) can remain after the operation, so that L is still available for remerge from it. The operation of merge will look like this when we represent both Work Spaces simultaneously. (WS2 is the main derivation.)

(2)



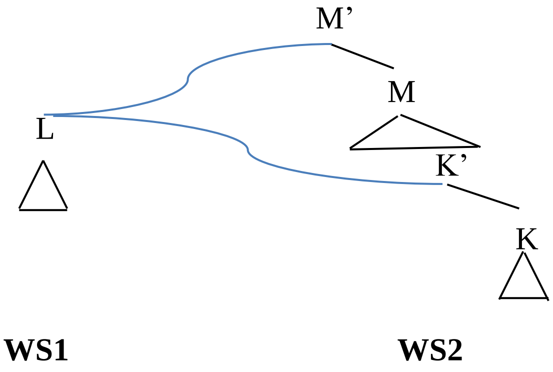
A picture of multiple merge (under this view) would be:

(3)



(3) – likewise (2) – invokes the Chomskyan notion of copy-merge (Chomsky 2003:307, 2008:140). But we are not forced to appeal to copies; we can as well merge L directly to multiple mother-nodes as shown in (4):

(4)[4]



Let us make clear what (3)/(4) says about the way we conceptualize displacement. In early transformational grammar, displacement was conceived of as analogous to the physical movement of an object. But currently, and since a long time, the assumption is that when a constituent α is moved from Position A to Position B, it does not cease to be in Position A. While we assume this for Internal Merge (and for sideward movement/parallel merge), we still continue with the old conception of movement for External Merge. What (3)/(4) does is to do away with this anomaly and

[4] At a first glance, (4) may not look so very different from Internal Merge, especially if one were to argue that a 'Work Space' is nothing more than the object created in it. But the difference of our system emerges very clearly when we consider a case like remerging an argument of an argument, in a configuration like (i) (where A is an argument of the main derivation and B is an argument of A):

(i) … $[_A$ … $[_B$ …] …] …

The Internal Merge theory will do subextraction here; i.e. it will remerge a term of A to the root of the tree. But in our system, B will also be in a parallel Work Space, the same as A; and B will be directly merged from its Work Space to the main derivation. Importantly, this will be a root-to-root merge and not a term-to-root merge which is what Internal Merge is. (This is illustrated in § 3, in our account of pied-piping.)

treat Internal Merge and External Merge alike. We can now maintain that there is only one type of displacement in natural language.

A question arises: Is L (in (3)/(4)) interpreted in all three positions? Let us say that L is already interpreted in WS1, i.e. the place where it is built up; but it is interpreted for its θ-related meaning in the first-merge position, and for its discourse-related meaning (its 'edge features', Chomsky 2008) in the criterial position, both of which are in WS2. (The derivation being in two Work Spaces simultaneously is not a problem: this is what we assume in the case of parallel merge.) The phrase will be linearized with respect to the rest of the derivation under either mother node.

The first noteworthy consequence of our proposal is the following:

(5) All merge is external merge.

What would be the advantage of maintaining (5)?[5]

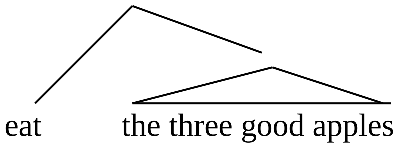
2.2. The content of parallel Work Spaces

[5] The idea is not without antecedents. In fact, Chomsky (2008:145-146) briefly considers, and rejects, this idea; but the execution he had in mind was in terms of generating the same phrase afresh for every merge, and he rightly says that this involves too much computation.

The implementation of "multiple Merge" that comes closest to our proposal is the "Survive Minimalism" of Stroik (1999, 2009), Putnam & Stroik (2010). In this system, a phrase is merged in its base position from a Work Bench. If the phrase has a feature which is unchecked in the base position, it "will return to (i.e. be copied in) the Work Bench for Remerge" (Stroik 2009:44-45). It will now be remerged to every new head that extends the derivation, until this feature is checked. So in a sense, every remerge has external merge as its last step.

Before we proceed, let us make clear what syntactic objects we expect to find occupying parallel Work Spaces in a derivation. Obviously, the arguments and adjuncts of a derivation will be represented in such spaces, since these are built up separately and merged as chunks. But the heads that project the ‘skeleton’ of the main derivation are taken directly from the Numeration; and these never come to be in such a space.[6] Similarly, the phrases that are built up in the main derivation itself – i.e. in its own Work Space – also never come to be in a parallel Work Space. To illustrate: In the following structure:

(6)



the three good apples will be represented in a parallel Work Space, since it is an argument; but neither *eat* nor *eat the three good apples* will be in such a space. These distinctions will become important in § 3, where we shall see that they make exactly the right predictions about what phrases can, and what phrases cannot, be moved by pied-piping.

These distinctions may also be important for another reason. Note that it is only the elements taken directly from the derivation’s own Numeration (and not from the Numeration of a different Work Space) – e.g. *eat* in (6) – that determine labels, i.e.

[6] Thus we maintain a distinction between merge from the Numeration and merge from a parallel Work Space. For a system that does not maintain this distinction, and which envisages a Numeration with “complex elements” (i.e. already-built-up phrases) in it, see Putnam & Stroik (2010), Stroik (2009:44).

project. It seems to be the case that phrases brought in from other Work Spaces (and their heads) are inert in the task of projection. Thus, when V takes an internal argument, it is V that projects; again, when vP takes an external argument in its Spec position (using 'Spec' only for expository purposes; see Starke 2004, Jayaseelan 2008, Chomsky 2013), the resulting structure is labelled 'vP'. (The same algorithm holds good in a VP which has two internal arguments: 'put [on the table]' is a VP, and '[the book] [put on the table]' is also a VP, because 'put' is the only element here which is taken from the derivation's own Numeration.)

2.3. Recovering a phrase for remerge: the problems of Internal Merge

Consider how Chomsky (1995:243) describes the operation of Merge:

> Applied to two objects α and β, Merge forms the new object K, *eliminating* α and β. [emphasis added – KAJ][7]

If we take 'eliminating' literally, α and β should no longer be available for a syntactic operation after K is formed. This is surely the simplest way to conceptualize Merge: after every merge operation, the derivation should be able to 'forget' the earlier syntactic objects that it handled. But unfortunately this conception is not viable, for if we wish to remerge (say) α later in the derivation, we must be able to "reanimate" α. (Koster 2007 makes this point.) For this, the derivation must hold α and β in its memory. Since the remerging of α may take place after *n* operations of merge (of other

[7] The idea is repeated further down on the same page: "Suppose a derivation has reached state Σ = { α, β, δ_i … δ_n }. The application of an operation that forms K [where K = { γ, { α, β }}, γ the label – KAJ] … converts Σ to Σ' = { K, δ_i … δ_n }, including K but not α, β."

elements), the derivation needs to remember syntactic objects that it handled *n* steps earlier. Also, if we continue with the idea that phrases may be built up in different derivational spaces and merged as chunks, α may have been merged with β in a different derivational space. Now the main derivation must access the memory of *that* derivational space. In effect, the derivation must remember every term of every phrase within a phase (besides the "left edge" of the earlier phase), even if we take into account the delimitation proposed by the phase theory (Chomsky 2000). Now some of these terms will be heads taken directly from the Numeration which project the 'skeleton' of the derivation. Some of these terms will be phrases built *within* the derivation. The current system has no way of sorting these out from the phrases which have been built in other derivational spaces and merged as chunks. Reverting to our illustration (6), the derivation must hold in its memory *eat*, (each term of) *the three good apples*, and *eat the three good apples*. Therefore in the current system with Internal Merge, the memory burden on the derivation is very great.[8]

But the real problem (for the current theory) is that the derivation *cannot* have those memories, because all those memories have been wiped out – this wiping out being done by our assumption that when a phrase is merged from a different Work Space, that Work Space disappears. The phrase now exists only in its merged position.

[8] The phase theory is intended to limit this problem. But it does not help very much, for note that even within the delimitation of a phase, the derivation must remember every term without exception – even though many (or most) of these terms are not eligible to move. (An example would be intermediate projections.) In our system, as we shall see, the derivation has to hold in its memory only the phrases in parallel Work Spaces, and these too only as 'chunks'. I.e., it must remember just the potentially moveable elements.

The problem with this way of doing things becomes clear when we look at an element that we are seeking to move which is multiply embedded. We examine this problem in the next section.

3. Pied-piping

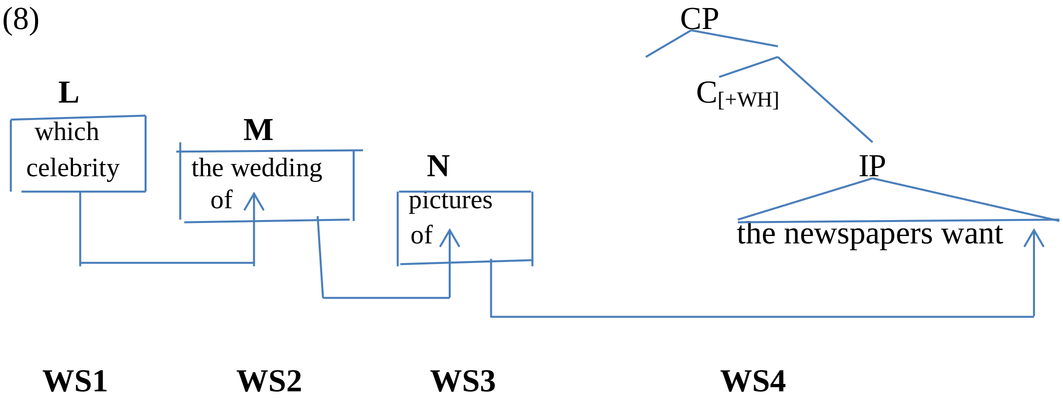
Consider a *wh*-element which is multiply embedded. (7) is a ‘pre-movement’ structure represented in terms of the current theory that assumes Internal Merge:

(7) the newspapers want [pictures of [the wedding of [[which] celebrity]]]

The grammar allows the movement of any of the bracketed elements except the innermost one. The D^0 element *which* cannot move by itself.

Given our proposal about simultaneously present parallel Work Spaces, (7) will have the representation (8) in our theory:

(8)



For ease of reference we have labelled the phrases on the left of the main derivation L, M, and N. L is built up in a separate Work Space prior to M; similarly, M is built up prior to N. It is N which is base-merged in the main derivation.

If Work Spaces 1, 2, and 3 may coexist with Work Space 4, the ‘null’ expectation – i.e. assuming that no stipulation is introduced to somehow prohibit such an operation – is that either L, or M, or N can be merged in Work Space 4 to satisfy $C_{[+WH]}$’s need for a *wh*-phrase in its specifier position. And this is indeed right, cf. the three results:[9]

(9) a. Which celebrity do the newspapers want pictures of the wedding of?

b. The wedding of which celebrity do the newspapers want pictures of?

c. Pictures of the wedding of which celebrity do the newspapers want?

Note that there is no notion of extracting a smaller phrase from a larger phrase in our system; i.e. there is no such thing as subextraction. The only possibility is that of giving a phrase that exists in a parallel Work Space a new mother node; this is our implementation of phrasal movement.

Our method is cognitively simpler, because it restricts its operations to roots. By contrast, subextraction involves many complications. Consider (7), which has a deeply embedded *wh*-word *which*. The dominant line of thinking about *wh*-movement is that a $C^0_{[+WH]}$ acts as a probe that seeks to match its [+WH] feature with an element in its

[9] (9b) and (9c) would be considered instances of heavy pied-piping. We comment on how we can deal with the differing acceptability judgments about heavy pied-piping later in this section.

domain (Chomsky 2000:101, 119).[10] Strictly, the matching is with a *wh-* feature; but since a feature cannot be moved by itself, a larger syntactic object that contains the *wh-* feature is moved. The mechanism that determines how large this syntactic object should be, is the core of any account of pied-piping.

The earliest idea in this regard was feature percolation (Chomsky 1973): the *wh-* feature is copied upwards from the *wh*-word to the phrases that contain it. But the problematic nature of the copying involved here has long been recognized. The feature must travel not only along the spine of the derivation, but must be able to 'jump' from specifier to head, cf. (10) (see Heck 2009:99 for a discussion of the problems of this movement); also, it must be able to span adjunct island boundaries, cf. (11):

(10) [Whose book] did you read?

(11) [The book which WHO wrote] were they unhappy about?

But the most damaging evidence against feature percolation (to my mind) is that the copying must be discontinuous – because VP and TP cannot be pied-piped, whereas CP can be. These well-known facts about pied-piping are illustrated below:

(12) * [Want pictures of which celebrity] do the newspapers?

(13) * [The newspapers had pictures of which celebrity] did you say that?

(14) [That WHO came] did you find surprising?

[10] We comment on the 'free movement' idea of Chomsky (2013) later.

CP pied-piping is not considered very felicitous in English, which in fact ‘dis-prefers’ heavy pied-piping; but many languages – e.g. Basque (Ortiz de Urbina 1990), Malayalam (Jayaseelan 2001) – standardly pied-pipe an embedded CP containing a *wh*-phrase to a position in the matrix clause, in order to give the *wh*-phrase matrix scope.

Note that the Internal Merge theory, which essentially seeks to move a phrase containing the *wh*- feature, has no way of distinguishing between VP/TP and CP. On the other hand, our system *predicts* this distinction. V and T are elements taken directly from the Numeration of the main derivation, and VP and TP are structures built up *within* that derivation. Therefore they will never appear in any parallel Work Space. How is CP different? An embedded CP, like any other argument or adjunct of the matrix V, is built up in a parallel Work Space and merged from it. The crucial difference here is that VP and TP are not arguments or adjuncts, but an embedded CP is.

The same considerations should explain why, in a structure like (7), the *wh*-word *which* cannot be moved by itself. This functional word is taken directly from the Numeration (subarray) of *which celebrity*, and therefore will not be represented in any parallel Work Space. (Recall that our options for phrasal movement are strictly limited to giving a new mother node to a structure that exists in a parallel Work Space.)[11]

There have been notable attempts to do pied-piping without feature percolation, e.g. Heck (2009), Cable (2010). Heck’s system explains pied-piping by a “local agree”

[11] N-to-D movement or V-to-T movement – head movement, more generally – obviously does not fall within the purview of pied-piping. We need to make a strict differentiation between this type of movement – which has its own very distinct properties: it is morphologically motivated, strictly local, and can induce “roll-up” – and phrasal movement (Jayaseelan 2010).

condition, the essential idea being that the pied-piping movement seeks to minimize the number of XP boundaries that separate the probe ($C^0_{[+WH]}$) and the goal (the *wh*-word). The inadmissibility of cases like VP pied-piping is put down to a constraint: "Pied-piping of β by α is possible only if movement of α from β is blocked" (p. 92). But this does not explain why CP pied-piping is possible – perhaps infelicitously or marginally in a language like English but quite standardly in some other languages.[12]

In Cable (2010)'s system, it is not the *wh*-word itself that is probed by the interrogative C head, but a Q(uestion)-particle that adjoins to, or takes as complement, a phrase containing the *wh*-word. *Wh*-fronting is a secondary effect of the movement of the Q-particle to $C^0_{[+WH]}$. The Q-particle can be generated above any phrase that contains the *wh*-word. Therefore this system cannot explain the "cut" between VP/TP and CP: why can't the Q-particle take VP or TP as complement when it can take CP as

[12] It can be easily shown that in these latter languages, TP pied-piping is completely impossible. Cf. the following Malayalam data, showing the completely normal (and standard) CP pied-piping and an attempted TP pied-piping:

(i) *CP pied-piping to cleft focus*

[$_{CP}$ John AAR-e kaNDu ennə] aaNə nii ***t***$_{CP}$ paRaññ-atə ?
John who-ACC saw COMP COP you said-NOM
Lit. '(It) is that John saw WHO, that you said?' (= 'Who did you say that John saw?')

(ii) *TP pied-piping to cleft focus*

** [$_{TP}$ John AAR-e kaNDu] aaNə nii ***t***$_{TP}$ ennə paRaññ-atə ?
John who-ACC saw COP you COMP said-NOM

Extraction of a *wh*-phrase from an embedded clause (by Spec-to-Spec movement) is not possible in Malayalam; this fact may appear to support Heck's constraint. But then, why can't the derivation be saved by TP pied-piping to the cleft focus?

complement? Cable in fact notes this problem: “[T]he data from Tlingit are consistent with a generalization that the language permits pied-piping only of DP, PP and CP, a generalization that also seems applicable to languages like English and does not obviously follow from the Q-based account” (p. 581: fn. 20).

So we see that our account of pied-piping is the only one in the literature that makes the right distinction between VP/TP and CP in a principled way. This is strong support for our way of implementing phrasal movement.[13]

Let us conclude this section with an account of heavy pied-piping. Clearly the ‘orthodoxy’ in the vast literature on pied-piping is that pied-piping is perfect only when the *wh*-word is at the left edge of the pied-piped phrase (Webelhuth 1992, Grimshaw 2000, Horvath 2007, and many others); this claim has even been formalized as an “edge generalization” (see, e.g., Heck 2009:89). Cable (2010), who was trying to account for Tlingit pied-piping where this generalization certainly does not hold, faced the same dilemma as we ourselves face, namely that our basic mechanisms of pied-piping do not distinguish between phrases in which the *wh*-word is at the left edge and phrases in which it is not. Cable draws a distinction between what he calls “limited pied-piping languages” (like English) and languages that are not similarly limited (like Tlingit); the former class of languages are the ones in which “pied-piping past adjunct islands and

[13] We note the fact that there is VP-fronting, or more generally predicate fronting:

(i) Kiss her, I didn’t.

(ii) Innocent, he is not.

However a fronted VP/predicate needs a linguistic antecedent (Phillips 2003). And so does VP/predicate deletion. We may assume that this type of fronting/deletion involves a very different mechanism of the grammar which is extra sentential, and does not involve phrasal movement.

pied-piping past lexical categories is not permitted" (p. 581). He accounts for this constraint as follows: in the English-type languages – but not in the Tlingit-type languages – there is a requirement that the Q-particle must agree with the *wh*-word; and if this agreement is to take place, the *wh*-word "cannot be dominated in the sister of Q by adjunct islands or lexical categories" (p. 584; see Cable 2010 for the feature-checking mechanism that implements this constraint). We can readily adapt Cable's proposal into our system although we do not postulate a Q-particle. We can say that in English-type languages, $C^0_{[WH]}$ must agree with the *wh*-word in the phrase that is sought to be merged in its Spec position. The other details of Cable's proposal can be kept intact. The way this constraint will work with the sentences of (9) is the following: (9a) is fine because the *wh*-word is not dominated by any lexical category; but (9b) and (9c) will be marked as unacceptable because the *wh*-word is dominated by one or more nouns.[14]

4. Successive-cyclic movement

[14] There is more to be said about heavy pied-piping in the Germanic languages. First of all, the judgments are gradient; even within English, there are speakers who find examples of heavy pied-piping completely acceptable – e.g. (famously) J. R. Ross (Ross 1967). Also (as is well-known), heavy pied-piping is better in matrix questions than in embedded questions, and in non-restrictive relative clauses than in restrictive relative clauses; which suggests that there are factors like sentence stress at play here. Again there are many rather complicated differences *within* the Germanic family, between English, Dutch and German (see de Vries 2005). All this suggests that the constraints on heavy pied-piping – whatever they are – should not be part of the 'core' account of pied-piping. In our system there is no special rule of pied-piping. Pied-piping is simply an instance of the completely normal operation of remerge of phrases from parallel Work Spaces.

Chomsky (2000, 2001) implemented successive cyclic movement by using two devices, both embedded in the idea of a ‘phase.’ CP and vP are taken to be the phases. At the end of each phase, the complement of the head of the phase – i.e., the complement of C^0 or v^0 – is “transferred” to Spell-Out and semantic interpretation; thereafter, this part of the structure is inaccessible to operations like extraction. This postulate is called the ‘Phase Impenetrability Condition’ (PIC). A consequence of PIC is that any phrase that needs to move out of the phase eventually, should have already moved to the ‘edge’ of the phase – Spec,CP or Spec,vP – before the end of the phase.

But PIC by itself does not force movement. Movement is actually implemented by Chomsky’s second device: every intermediate C^0 and v^0 in the path of a long movement is given an EPP feature; and this enables the C^0/v^0 to pull up the phrase to its Spec.

In the system of Chomsky (2000, 2001), EPP targets a phrase because it contains an uninterpretable feature (‘uF’) that makes it ‘active’. For A-movement, Chomsky assumes the uF to be Case; for *wh*-movement, it is a [+ WH] feature. But the system of Chomsky (2008, 2013) dispenses with any uF on phrases which move (Chomsky 2008:151). The suggestion in Chomsky (2008) is that the EF of the phase head can ‘indiscriminately’ pull up any phrase in its domain to the edge position of the phase; the interpretation of the moved phrase will depend on the final position it reaches. An interrogative *wh*-phrase will be labeled by an *interpretable* interrogative *wh*-feature; and if this phrase does not reach the Spec position of an interrogative C, it will simply be interpreted as deviant. The system of Chomsky (2013) proposes “free Merge”.

It should be fairly clear that implementing successive-cyclic movement in our system presents no particular problem. If an intermediate phase head has an EF that induces merge, it can induce the merge of a phrase that exists independently in a parallel Work Space even more easily than of a phrase in an embedded position.

In the "free Merge" proposal of Chomsky (2013), so far as I understand the system, what drives successive-cyclic movement appears to be only PIC. But if PIC is dropped from the theory (a move which has been argued for; see, e.g., Stjepanović & Takahashi 2001), what can implement successive-cyclicity? A plausible alternative explanation of successive-cyclicity (I suggest) is that there is a need for a long-moving phrase to repeatedly activate itself in the derivational memory. At the end of each phase it reactivates itself by merge (merge being "free"). Note that this idea also gives us a natural explanation of Superiority effects (such as is illustrated in: 'Who bought what?'/ * 'What did who buy?'). A phrase that was merged/remerged recently is more 'active' – and therefore more 'accessible' – than a phrase merged/remerged earlier. This new formulation replaces the idea of 'distance' in the tree measured in terms of the number of intervening nodes, which was natural in a system with Internal Merge.

5. Extraction islands

After Ross's (1967) discovery of some syntactic configurations that resist extraction from within them, there have perhaps been two main approaches to explaining this phenomenon. The early approach was in terms of 'bounding nodes' (which were variously defined). The idea was that a phrase trying to come out of one of these opaque domains was 'stopped' by a bounding node. The main exemplars of this approach were the Subjacency idea of Chomsky (1973) and its subsequent re-workings upto the “barriers” idea of Chomsky (1986). Takahashi (1994) is a latter-day variation of (essentially) the ‘bounding node’ idea. This type of explanation is peculiarly dependent on there being Internal Merge in the theory.

While Chomsky’s later work in the minimalist framework did not come back to the question of extraction islands, there has more recently been a very different type of proposal about this problem, which appeals to the organization of the Grammar. The idea of the new proposal is that a constituent resists extraction from within it because it has already been spelt out at the time of the attempted extraction. See Uriagereka (1999), Epstein (1999), Nunes & Uriagereka (2000). Nunes & Uriagereka (2000) make the following claims:

A. Phrases in Specifier positions are already spelt out: they have been spelt out prior to being merged in these positions.

B. All phrases that have moved have already been spelt out; or in other words, phrases are spelt out before they are moved.

C. Adjuncts are spelt out before they are merged in the main derivation.

In effect, only a complement can be sub-extracted from, because this is the only constituent in a tree that has not been spelt out.

The Nunes-Uriagereka proposal seeks to subsume the left-branch condition (A), the freezing principle (B), and the Adjunct Island Condition (C). Huang’s (1982) Condition on Extraction Domains (CED), which was an attempt to give a unified explanation of adjunct islands and subject islands – more generally, of non-complements as islands – is also subsumed by this proposal.

Recently, it has been argued that adjunct islands and subject islands should not be unified (Stepanov 2001a, 2007). The most compelling reason for saying this is that there appear to be languages which allow extraction out of subjects, whereas the Adjunct Condition is universal.

Here we do not group together Subject and Adjunct. Instead, we group Subject with Topic and Focus. All three are known to be islands for extraction (Ross 1967, Lasnik & Saito 1992, et. al). Cf. (15), (16) and (17):

(15) *Subject Island*

a. [Which politician]$_i$ would the voters like [a picture of t$_i$]?

b. * [Which politician]$_i$ would [a picture of t$_i$] please the voters?

(16) *Topic Island*

a. I think (that) [a picture of Peter]$_i$ Mary would like t$_i$.

b. * Who$_j$ do you think (that) [a picture of t$_j$]$_i$ Mary would like t$_i$?

(17) *Focus Island*

a. It is [a picture of Peter]$_i$ that Mary wants t$_i$.

b. * Who$_j$ is it [a picture of t$_j$]$_i$ that Mary wants t$_i$?

What Subject, Topic and Focus have in common is that they are all exemplars of an ‘edge property’ (Chomsky 2008).[15] And what we have in the sentences exemplifying the Subject/Topic/Focus Condition is a ‘nested’ structure as regards edge

[15] In the framework of Chomsky (2008), all operations in a phase, including the movement of the Subject, must be driven by a feature inherited from the phase head. See also Rizzi (2005).

properties. Thus in (15b), *a picture of which politician* is Subject, and *which politician* (a proper term) is [+ WH].

Why is this ‘nesting’ of edge properties problematic? A plausible reason is that this creates a difficulty of interpretation. While nothing prevents a phrase as a whole having two edge properties – thus a phrase can be both Subject and Topic, cf. *This book$_i$, I don’t believe [t$_i$ has been read t$_i$]* – the assignment of distinct edge properties to a phrase and a proper subpart of it appears to be difficult. Our claim (then) is that the problem here is one of the interpretational mechanism of the grammar, and not something purely syntactic.

Coming to adjuncts, traditionally taken to be opaque domains for extraction, we may need to speak about different types of adjuncts (see Cattell 1976, Stepanov 2007 for a discussion). Thus instrumental adverbials, which are thematically adjuncts, are transparent for subextraction (cf. *What did you break the glass with?*). So are infinitival adjuncts of purpose (cf. *Who did she come here to look for?*). But time/place adverbials, and adverbials of cause are opaque (cf. * *Who did you see a snake near?*; * *Who did you arrive after?*; * *Who did she cry, because you hit?*). For our purposes, we don’t need to investigate this question in any depth. (The interested reader can see Boeckx 2008: § 1.3.2 for a wide-ranging discussion.) We can adopt any of the current proposals to explain the opacity of adjuncts, none of which conflicts with our basic proposal. Takahashi (1994) suggests that opaque adjuncts represent a kind of coordination structure. Another proposal is that adjuncts are late insertions: they are merged acyclically (or postcyclically); see Lebeaux (1988), Uriagereka (1999), Stepanov (2001a, 2001b, 2007). Their late insertion could be the reason why adjuncts (or at least some types of adjuncts) cannot make available a subphrase to the main derivation.[16] (However, as we shall see directly, the main derivation can make available a subphrase to an adjunct.)

6. Parasitic Gaps

For any framework that assumes that movement begins from a position in the tree, the parasitic gap construction – illustrated in (18) – presents a problem:

(18) a. Which paper$_i$ did you read t$_i$ [before filing t$_i$] ?

b. Which politician$_i$ would [a picture of t$_i$] please t$_i$?

How can a chain have two base positions in the tree? Or if there are two chains, how do they 'coalesce', such that there is only one chain-head? A natural solution – given this way of formulating the problem – was in terms of "chain-composition," which was proposed in Chomsky (1986). Another noteworthy attempt to solve the problem appeals to "across-the-board" movement (Williams 1990).

The solution that we find most interesting (and which is closest to our proposal) is one in terms of 'sideward movement' (Nunes 2001, 2004, 2013; Nunes & Uriagereka 2000). The essential idea here is that an adjunct and the main derivation are built up simultaneously and 'in parallel'; and a phrase merged in the adjunct can be copied and merged in the main derivation. This 'sideward movement' from one PS tree to another

[16] Note that this would be a 'third factor' explanation (in the sense of Chomsky 2005) of the opacity of adjuncts, unlike the bounding node idea.

must be done (importantly) *before* the adjunct is merged with the main derivation; because when the adjunct is so merged, it has already been spelt out and is now opaque to extraction (see our account of the Nunes-Uriagereka proposal in § 5). The new copy in the main derivation can be moved unproblematically – say, to Spec,CP. And the deletion of traces gives us the parasitic gap construction.

From our new perspective in this paper, we can see that the parasitic gap problem is an artifact of Internal Merge. If merge is from a parallel Work Space, we can say that a phrase that is present in a parallel Work Space can be merged in an adjunct as easily as in the main derivation. If adjuncts are built up late in the derivation, this is no problem: the main derivation is already in place when the adjunct is being built up. Again, a subject island violation as in (18b) is not a problem at least for the syntax (although there could still be an interpretation problem): the subject argument can be 'got ready' later than the object argument; the latter can be already present in a parallel Work Space, and it can be merged as a subphrase when the subject argument is being built up.[17]

[17] A traditional claim in the literature on parasitic gaps is that a parasitic gap cannot be inside an extraction island properly contained in an adjunct (or subject) – this was the main supporting evidence for an operator movement account of this phenomenon (Chomsky 1986). Cf.

(i) * Which book$_i$ did you like t_i [after meeting the man who wrote t_i] ?

But the following sentence is equally bad although there is no extraction island inside the adjunct:

(ii) * Which book$_i$ did you like t_i [after your wife said that she liked t_i] ?

A number of unclear factors seem to impinge on the acceptability of parasitic gaps. It is perhaps correct to say that a parasitic gap is 'perfect' only when the adjunct that contains it is nonfinite and has a PRO-subject; or at least if the subject of the adjunct is the same as the subject of the main clause; cf.:

(iii) a. Which books$_i$ do you read t_i before buying t_i ?

7. PIC, 'partial' phasal Spell-Out, and other matters

The Phase Impenetrability Condition (PIC) is usually sought to be justified in terms of third factor considerations: at the end of a phase, the complement of the phase head is transferred to Spell-Out and semantic interpretation; after this, the derivation can 'forget' this part of the structure, thereby reducing its memory burden. PIC also has another reason for its existence: it does half the work of implementing successive-cyclic movement, the other half being done by EPP. But in a system that dispenses with EPP and movement is 'free' (Chomsky 2013)[18], PIC – as far as I can see – is left to shoulder the entire burden of successive-cyclic movement.

b. Which books$_i$ do you read t$_i$, before you buy t$_i$?

c. * Which books$_i$ do you read t$_i$, before your students buy t$_i$?

Consider the following example from Chomsky & Lasnik (1993) (cited by Nunes 2004:115)

(iv) * The book was filed without my reading first.

Chomsky and Lasnik cite this sentence as evidence that a trace of A-movement cannot license a parasitic gap; but this cannot be the right explanation, cf. (v) which has an operator trace to license the parasitic gap:

(v) * Which book did you file without my reading first?

Nunes's explanation (ibid.) of (iv) is that *the book* in the subject position of the main clause and its copy in the object position of *reading* cannot form a chain "because of the intervention of *my*". This explanation might be on the right track, for it seems to me that the gradient judgments about parasitic gaps are due to factors of interpretation and not of phrasal movement per se.

[18] See also Epstein, Kitahara & Seely (2014) for a discussion of this new conception of re-merge, which the authors call "simplest Merge".

We should bear in mind that PIC is predicated on a close identification of Spell-Out and Pronunciation. This identification itself is very clear; e.g., consider Chomsky's (2004) suggestion that movement can be ordered before or after Spell-Out, and that the ordering gives us overt or covert movement. The idea is that, if α has not moved at the time of Spell-Out, there is only one copy of α; and so this copy is spelt-out – and therefore pronounced. But if α has moved, there are two copies of α, and the lower copy is not spelt-out – and therefore not pronounced. (Chomsky 1995:228 says that it is marked as "invisible at the interface.") This assumes that what is spelt-out must be pronounced, and what is pronounced has been earlier spelt-out. Also note that the 'deletion' of traces is done – or at least determined – by Spell-Out, at the level of the phase.

This identification of Spell-Out and Pronunciation has many disadvantages. One problem is that there is also discourse-dependent deletion, e.g. VP-deletion; and obviously, Spell-Out cannot do this deletion. So in the current theory, trace deletion and discourse-dependent deletion cannot be unified.

But possibly, discourse-dependent deletion is another *type* of deletion and should not be unified with trace deletion. While this is plausible enough, note that phase-wise deletion of traces is problematic in itself: it gives rise to 'look ahead' in the multiple *wh*-construction in English. Consider (19):

(19) a. Who(m)$_i$ does John think that he is in love with t$_i$?

b. Who thinks that he is in love with whom ?

In (19b), *whom* should not start to move to Spec, vP in the embedded vP-phase; but this decision is dependent on whether or not there is another *wh*-phrase merged in the higher vP-phase.

Let us say (instead) that all copies of a phrase are spelt out (Bobaljik 2002, a.o.). And that deletion is done by Pronunciation – late in the derivation. This enables us to unify the two kinds of deletion and to avoid the look-ahead problem. Adapting this proposal to systems (like ours) that assume multi-dominance – i.e. the idea that a phrase can have many mother-nodes – is straightforward enough: we can say that a phrase can be pronounced under any of its mother-nodes, the choice of the mother-node being a late decision.

This way of looking at things naturally gets rid of the old "escape hatch" idea (Chomsky 1973), which has its latest implementation as the 'partial' spell-out of phases. Much dissatisfaction has been felt about this last-mentioned idea, cf. Boeckx & Grohmann (2007), Richards (2011).

But let us ask the question: Can the economy argument for phases still be retained? I believe it can. We suggested (in § 2) that a phrase is already interpreted – and let us now say, spelt-out – in the parallel Work Space where it is built-up. Let us say that in the main derivation, each phase (similarly) is interpreted and spelt-out, as and when it is completed.[19]

[19] In this proposal, successive-cyclic movement, i.e. the remerge of phrases to intermediate phase-heads in long movement, will be motivated solely by the need to 'reactivate' a phrase in the derivational memory.

References

Bobaljik, J. 2002. A-chains at the PF-interface: copies and 'covert' movement. *Natural Language and Linguistic Theory* 20 (2): 197-267.

Boeckx, C. 2008. *Bare Syntax*. Oxford: Oxford University Press.

Boeckx, C. & K. Grohmann. 2007. Putting phases in perspective. *Syntax* 10:204-222.

Cable, S. 2010. Against the existence of pied-piping. *Linguistic Inquiry* 41:553-594.

Cattell, R. 1976. Constraints on movement rules. *Language*, 52:18-50.

Chomsky, N. 1973. Conditions on transformations. In *A Festschrift for Morris Halle,* ed. S. Anderson & P. Kiparsky, 232-286. New York: Holt, Rinehalt, and Winston.

Chomsky, N. 1986. *Barriers.* Cambridge, Mass.: MIT Press.

Chomsky, N. 1995. Categories and transformations. In *The Minimalist Program,* N. Chomsky, 219-394. Cambridge, Mass.: MIT Press.

Chomsky, N. 2000. Minimalist inquiries: the framework. In *Step by Step,* ed. R. Martin, D. Michaels and J. Uriagereka, 89-155. Cambridge, Mass.: MIT Press.

Chomsky, N. 2001. Derivation by phase. In *Ken Hale: A Life in Language*, ed. M. Kenstowicz, 1-52. Cambridge, Mass.: MIT Press.

Chomsky, N. 2003. Replies. In *Chomsky and His Critics,* ed. Louise M. Antony and Norbert Hornstein, 255-328. Oxford: Blackwell.

Chomsky, N. 2004. Beyond explanatory adequacy. In *Structures and Beyond: The Cartography of Syntactic Structures*, ed. A. Belletti, 104-131. Oxford: Oxford University Press.

Chomsky, N. 2005. Three factors in language design. *Linguistic Inquiry* 36:1-22.

Chomsky, N. 2008. On phases. In *Foundational Issues in Linguistic Theory*, ed. R. Freidin, C. P. Otero, and M. L. Zubizarreta, pp. 133–166. Cambridge, MA: MIT Press.

Chomsky, N. 2013. Problems of projection. *Lingua* 130: 33-49.

Chomsky, N. & H. Lasnik. 1993. The theory of principles and parameters. In *Syntax: An international handbook of contemporary research*, ed. J. Jacobs, A. von Stechow, W. Sternefeld, and T. Vennemann, 506-569. Berlin: Walter de Gruyter.

Collins, C. 2002. Eliminating labels. In *Derivation and Explanation in the Minimalist Program*, ed. S. D. Epstein and T. D. Seely, 42-64. Oxford: Blackwell.

Epstein, S. D. 1999. Un-principled syntax: the derivation of syntactic relations. In *Working Minimalism,* ed. S.D. Epstein & N. Hornstein. Cambridge, Mass.: The MIT Press.

Epstein, S.D., H. Kitahara & T.D. Seely. 2014. Labeling by minimal search: implications for successive-cyclic A-movement and the conception of the postulate ‘phase’. *Linguistic Inquiry* 45(3):463-481.

Grimshaw, J. 2000. Locality and extended projection. In *Lexical specification and insertion*, ed. P. Coopmans, M. Everaert, and J. Grimshaw, 115‑133. Amsterdam: John Benjamins.

Heck, F. 2009. On certain properties of pied-piping. *Linguistic Inquiry* 40:75-111.

Horvath, J. 2007. Pied-piping. In *The Blackwell Syntax Companion*, ed. M. Everaert & H. van Riemsdijk, 569-630. Oxford: Blackwell.

Huang, C.-T. J. 1982. Logical relations in Chinese and the theory of grammar. Doctoral dissertation, MIT,Cambridge, Mass.

Jayaseelan, K. A. 2001. IP-internal topic and focus phrases. *Studia Linguistica* 55:1, 39-75.

Jayaseelan, K. A. 2008. Bare phrase structure and specifier-less syntax. *Biolinguistics* 2.1:87-106.

Jayaseelan, K. A. 2010. On two types of movements. Paper presented at GLOW-in-Asia VIII, Beijing Language and Culture University, 12-14 August 2010.

Koster, J. 2007. Structure preservingness, internal merge, and the strict locality of triads. In *Phrasal and Clausal Architecture: Syntactic Derivation and Interpretation* (Linguistik Aktuell/Linguistics Today 101), ed. Simin Karimi, Vida Samiian & Wendy K. Wilkins, 188-205. Amsterdam: John Benjamins.

Lasnik, H. & M. Saito. 1992. *Move α*. Cambridge, MA: MIT Press.

Lebeaux, D. 1988. *Language Acquisition and the Form of the Grammar*. Doctoral dissertation, University of Massachusetts, Amherst.

Nunes, J. 2001. Sideward movement. *Linguistic Inquiry* 32:303-344.

Nunes, J. 2004. *Linearization of Chains and Sideward Movement*. Cambridge, Mass.: MIT Press.

Nunes, J. 2013. Sideward movement: triggers, timing, and outputs. In *Ways of Structure Building*, ed. Myriam Uribe-Etxebarria & Vidal Valmala, 114-142.

Nunes, J. & J. Uriagereka. 2000. Cyclicity and extraction domains. *Syntax* 3:20-43. Oxford: Oxford University Press.

Ortiz de Urbina, J. 1990. Operator feature percolation and clausal pied-piping. In *MIT*

Working Papers, vol. 13: Papers on Wh-Movement, ed. L. Cheng & H. Demirdash, 193-208.

Phillips, C. 2003. Linear order and constituency. *Linguistic Inquiry* 34:37-90.

Putnam, M. & T. Stroik. 2010. Syntactic relations in Survive-minimalism. In *Exploring crash-proof grammars*, ed. M. Putnam, 143-166. Amsterdam: John Benjamins.

Richards, M. 2011. Deriving the edge: What's in a phase? *Syntax* 14:74-95.

Rizzi, L. 2005. On some properties of subjects and topics. In *Contributions to the thirtieth "Incontro di Grammatica Generativa"* , ed. L. Brugé, G. Giusti, N. Munaro, W. Schweikert, G. Turano, 203-224. Università Ca' Foscari Vanezia.

Ross, J. R. 1967. Constraints on variables in syntax. Doctoral dissertation, MIT, Cambridge, Mass.

Starke, M. 2001. Move dissolves into merge: A theory of locality. Ph.D. dissertation, University of Geneva.

Starke, M. 2004. On the inexistence of specifiers and the nature of heads. In *The Cartography of Syntactic Structures*, vol. 3: *Structures and Beyond* (Oxford Studies in Comparative Syntax), ed. A. Belletti, 252-268. New York: Oxford University Press.

Stepanov, A. 2001a. Cyclic domains in syntactic theory. Ph.D. dissertation, University of Connecticut, Storrs.

Stepanov, A. 2001b. Late adjunction and minimalist phrase structure. *Syntax* 4: 94-125.

Stepanov, A. 2007. The end of CED? Minimalism and extraction domains. *Syntax* 10:1, 80-126.

Stjepanović, S. & S. Takahashi. 2001. Eliminating the Phase Impenetrability

Condition. Paper presented at the workshop on ‘Motivating movement: Explaining the displacement property of natural language’, University of Ulster.

Stroik, T. S. 1999. The survive principle. *Linguistic Analysis* 29:278-303.

Stroik, T. S. 2009. *Locality in Minimalist Syntax.* Cambridge, Mass.: MIT Press.

Takahashi, D. 1994. Minimality of Movement. Doctoral dissertation, University of Connecticut.

Uriagereka, J. 1999. Multiple Spell-Out. In *Working Minimalism*, ed. S.D. Epstein & N. Hornstein, 251-82. Cambridge, Mass.: The MIT Press.

de Vries, M. 2005. Possessive relatives and (heavy) pied-piping. *Journal of Comparative Germanic Linguistics* 9:1-52.

de Vries, M. 2013. Unconventional mergers. In *Ways of Structure Building*, ed. Myriam Uribe-Etxebarria & Vidal Valmala, 143-166. Oxford: Oxford University Press.

Webelhuth, G. 1992. *Principles and parameters of syntactic saturation*. Oxford: Oxford University Press.

Williams, E. 1990. The ATB theory of parasitic gaps. *The Linguistic Review* 6, 265-279.