Eliminating external Merge

Jan-Wouter Zwart

version 2 (February 2017)

1. Internal Merge

The central idea in this paper is that syntactic structure (as commonly understood) is the result of a process that turns an unordered set of elements into an ordered pair (or ordered n-tuple) of these same elements (Zwart 2004, 2009a; also Fortuny 2008). The process is iterative, applying with each step to the largest unordered set of elements in the structure. The process is a movement process, merging one element from the unordered set of elements to the set as a whole:

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

More precisely, the process depicted in (1) is a case of internal Merge (Chomsky 1995), respecting the Extension Condition.

(2) Internal Merge
Combine $\alpha \in \Gamma$ and Γ

(3) Extension Condition
In (2), there is no Γ', Γ' ≠ Γ, such that $\Gamma \in \Gamma'$

Given a model of grammar where we have the three elements in (4):

(4) Elements of the model of grammar
(i) a set of elements (Numeration)
(ii) an operation creating structure (Narrow Syntax)
(iii) an externalization procedure (interpreting the structure for sound and meaning)

we will say that (ii), Narrow Syntax, contains just Internal Merge, i.e. (1).

Given Kuratowski's Law (Kuratowski 1921), the output of (1) is the set-theoretical definition of an ordered pair:

(5) $\{ \alpha, \{ \alpha, \beta \}\} \equiv \langle \alpha, \beta \rangle$

Hence, Internal Merge of α in (1) suffices to turn the set { α, β } into an ordered pair, i.e. to reach our set goal of turning an unordered set into an ordered pair.

I will assume that the externalization procedure (4iii) will see only the ordered pair. That is, when the externalization procedure encounters { α, { α, β } }, it will see ⟨ α, β ⟩. I will assume that the default realization of ⟨ α, β ⟩ will be a string αβ, and the default interpretation of ⟨ α, β ⟩ will be an asymmetric dependency relation (with β dependent on α).

Since the externalization procedure only sees ordered pairs, it will never 'see' the two copies of α created by Internal Merge in (1). Hence there is no need for any stipulations ensuring that only one of the two copies gets spelled out. (Also, there is no need or room for theories capitalizing on selective spell-out of the copies resulting from Internal Merge.) I take this to be a major improvement over existing theories of movement (for a reformulation of A-movement and A'-movement, see below).

In standard descriptions of the structure building process, assuming the elements of the model of grammar in (4), two forms of Merge are distinguished, Internal Merge and External Merge. External Merge differs from Internal Merge in that it introduces a new element:

(6) External Merge
Combine $\alpha \notin \Gamma$ and Γ

This requires a separation of the Numeration and what we may call the object under construction (OUC), such that OUC = Γ, and the Numeration is a resource of elements distinct from Γ. On this conception, structure building involves not only Internal Merge (2), but also a transfer of elements from the Numeration to the OUC.

The metaphorical status of this 'transfer' conception of structure building was already noted in Bobaljik (1995). As Bobaljik points out, Narrow Syntax minimally needs to enrich the Numeration with information on relations between Numeration members. So, instead of taking α and β from the Numeration and merging them in a separate OUC, we could dispense with the metaphor and state that after merger of α and β, the Numeration now also contains the combination of α and β (αβ). While the system contemplated here deviates from Bobaljik's in taking merger to be Internal Merge (2), it does take over from Bobaljik's system that the OUC and the Numeration are the same object, minimally transformed at each step in the derivation.

If we are right, there is no room for External Merge (6) in a syntactic derivation.

2. Questions of priority

Assuming a distinction between Internal Merge and External Merge raises the question which of the two takes precedence over the other. This question (discussion of which has remained fruitless) does not occur in the system contemplated here.

Chomsky (1995) initially proposed that since External Merge is an inevitable component of structure generation, it must be considered cost-free, and hence preferable over Internal Merge, which requires a trigger (to overcome inertia). This explains that we find (7a) instead of (7b).

(7) a. There seems <*e*> to be a man in the room
b. * There seems a man to be <*e*> in the room

Assuming a bottom-up structure building process, the position marked by <*e*> in (7a) could be created by Internal Merge of *a man* or External Merge of *there*. Internal Merge of *a man* followed by External Merge of *there* in the subject position of the main clause yields (7b), a derivation to be excluded. Positing External Merge over Internal Merge ('Merge over Move') gives the desired result. (*A man seems to be in the room* starts from a different Numeration, lacking *there*, and hence is not likewise blocked.)

More recently, however, Chomsky (2008?) has argued that the earlier characterization of Internal Merge as being somehow marked was wrong, as elements available in the structure could only be barred from undergoing Internal Merge by stipulation. (This assumes that movement is the automatic outcome of a process of 'Minimal Search' by a 'Probe', a feature of the root node of the OUC in need of elimination by some 'Goal' element eligible for Internal Merge on the basis of its feature composition.) Moreover, Internal Merge may sometimes be needed to create objects that can be interpreted in the externalization process (4iii)(Chomsky 2013). (This assumes that the externalization process needs to be able to read categorial information off of the structure, and hence projection of category features must be ensured during the syntactic derivation; Internal Merge, leaving an empty category, creates the asymmetry needed to allow the sister of the moved element to project its category features.)

While this shows that good, theory internal reasons exist for not preferring External Merge over Internal Merge, some indeterminacy remains. In the system contemplated here, where no External Merge exists, that problem disappears completely.

3. The trigger for Merge

Any finite derivational procedure needs a trigger to overcome inertia (a motor) and a halting condition (a stop).

We mentioned some potential triggers for Merge that are currently prominent in the theoretical literature (involving the Probe-Goal mechanism or the Feature Projection requirement). But these triggers are motivated by highly specific assumptions about interpretability requirements relevant to the externalization process (4iii). It is inherent to minimalism (as defined by Chomsky 1995) that interpretability at the interface components (dealing with sound and meaning) should be the only factor triggering operations in Narrow Syntax. But this interpretability requirement may lead to any number of specific hypotheses about how interpretability is endangered or ensured, enriching the ontology of features and operations of Narrow Syntax. None of these features and operations have the desired quality of inevitability (what has been called the 'third factor' quality).

I assume that an inevitable component of interpretability is asymmetry. An asymmetric pair of elements is an ordered pair; an asymmetric set of *n* elements is an ordered *n*-tuple. (See Zwart 2011a for more discussion of the relation between structure and order in this context.) If the Numeration is an unordered set, as we have assumed, then what Narrow Syntax needs to achieve minimally is an ordering of the elements in that set. This is what Internal Merge in (1) does.

I will assume, then, that the trigger to overcome inertia is the existence of unordered elements in the Numeration, and that the halting condition is reached as soon as no unordered elements are left (i.e. the initial set is empty). (More precisely: Internal Merge turns a set into a set that is read as an ordered pair by the externalization procedure, and will continue to operate for as long as the set cannot be read as an exhaustively ordered n-tuple.)

4. Iteration

Consider a Numeration containing more than two elements:

(8) $\{ \alpha, \beta, \gamma \}$

Applying Internal Merge to (8) yields (9).

(9) $\{ \alpha, \{ \alpha, \beta, \gamma \} \}$

This orders α with respect to β and γ, but leaves β and γ unordered with respect to each other. In other words, (9) can be read as (10):

(10) $\langle \alpha, \{ \beta, \gamma \} \rangle$

After this first application of Internal Merge, the Numeration (8) is not yet exhaustively ordered, triggering a second application, targeting the unordered set $\{ \beta, \gamma \}$:

(11) $\{ \beta, \{ \beta, \gamma \} \} \equiv \langle \beta, \gamma \rangle$

This now yields the ordered n-tuple in (12):

(12) $\langle \alpha, \langle \beta, \gamma \rangle \rangle \equiv \langle \alpha, \beta, \gamma \rangle$

As there are no unordered elements remaining, the derivation halts.

This derivation is clearly similar to the one involving 'Split-Merge' in Zwart (2009a), and hence may be viewed as a 'top-down' derivational procedure. However I think it is more correct to say that the derivation is neither 'bottom-up' nor 'top-down', as those denominations presuppose a distinction between the Numeration and the OUC, a narrative this proposal surpasses (see section 9).

One might wonder why, at the stage illustrated by (10), Internal Merge does not combine β with the entire object $\langle \alpha, \{ \beta, \gamma \} \rangle$. The answer is that this would revoke the asymmetry between α and β created by the first operation Internal Merge. [PROOF]

To see this, consider that what Internal Merge creates is a record of asymmetries. What the externalization procedure needs is for that record to be complete, transitive, and antisymmetric (Kayne 1994). In our example, the first operation Internal Merge creates the record in (13).

(13) $\langle \alpha, \beta \rangle \langle \alpha, \gamma \rangle$

The next operation adds $\langle \beta, \gamma \rangle$, yielding

(14) $\langle \alpha, \beta \rangle \langle \alpha, \gamma \rangle \langle \beta, \gamma \rangle$

i.e., a complete record. On the other hand, merging β to $\langle \alpha, \{ \beta, \gamma \} \rangle$ would add $\langle \beta, \alpha \rangle$, yielding a record that contains both $\langle \alpha, \beta \rangle$ and $\langle \beta, \alpha \rangle$, violating antisymmetry.

Notice that the operation of Internal Merge in our example is iterative, not recursive, rejecting the notion that in minimalism, every application of Merge is a recursive step (see Zwart 2011b for discussion).

5. What are the elements in the Numeration?

So far we have left the status of the elements in the Numeration open. Here, various options exist:

(15) Elements in the Numeration may be:
a. of a single unique type (say, words)
b. of any type, but homogeneous (all words, all morphemes, all features, etc.)
c. of any type (heterogeneous)

Barring stipulation, we expect (15c) to hold.

Typically, though, the Numeration is considered to contain only words (cf. Zwart 2015). It is easy to see that this cannot be right, as complex left branch elements must be more complex. Consider a simple example like (16).

(16) The world changes

Here *the world* is a complex left branch element (the subject). That is, *the world* is a constituent, and is interpreted as such by the externalization procedure.

A minimal threshold of adequacy for any structure building mechanism is that it derives the constituent parts of a structure. In our system, that means that *the world* must be one of the elements in Narrow Syntax, i.e. either an element in the Numeration, or a set of elements in the Numeration, or one of the element-set combinations that the externalization procedure interprets as an ordered pair.

Suppose now that the Numeration contains only words. Then the Numeration for (16) is:

(17) { the, world, changes }

Applying Internal Merge to *the* yields the ordered pair (18):

(18) 〈 the, { world, changes } 〉

In (18), *the world* is not an element, nor a set, nor an ordered pair, so its constituent status is not derived. This conclusion remains also if we decide to apply Internal Merge to one of the other elements in (17) first. (Applying Internal Merge to *changes* first yields *the world* as a constituent, but the constituency problem returns if the verb is transitive, as in *changes us*; Internal Merge of *changes* would then fail to yield the constituent *changes us*. (Applying Internal Merge to *us* first and to *changes* then would fix that, but only if the internal argument is not itself complex; etc.).)

It follows that on the system contemplated here, the elements in the Numeration must be heterogeneous. For example, the Numeration for (16) must be:

(19) { [the world], changes }

Now *the world* in (19) is clearly complex. Assume (20):

(20) Any element that is structured is derived by Narrow Syntax (4ii).

We may strengthen this to:

(21) Any element that is structured is derived by a grammar as modeled in (4).

It follows that a simple sentence like (16) involves (at least) two derivations, one generating *the world* and a second one generating *the world changes*. In the second derivation, *the world* is a single (atomic) item, as reflected in the Numeration in (19).

I assume, then, that derivations are typically complex (layered, cf. Zwart 2009a), and that they interact. More precisely, I propose that the interaction takes place in a maximally simple way, such that the output of one derivation may be included in the Numeration for another derivation.

As I argue in Zwart (2011b), this interplay of derivations (the output of one derivation featuring as part of the input for another) is the hallmark of recursion, apparently present in all human languages and in many other manifestations of human cognition as well (Hofstadter 2007).

See Zwart (2009a) for the general architecture of derivation layering. Importantly, no element passes from one derivation to the next without undergoing the externalization procedure (4iii). Crucially, we do not allow elements from the Numeration or Narrow Syntax of one derivation to 'move' to the Numeration or Narrow Syntax of another derivation. The reason we do not allow this, is that such movement operations cannot be formulated on the definition of Internal Merge in (2), and we assume that Internal Merge is all there is.

We may develop this ban on interderivational movement (call it 'migration ban') into a conceptual argument against External Merge (6). In the absence of phenomena necessitating migration (in this sense), External Merge is not supported.

6. A network of derivations

We have now established that derivations interact (in a highly restricted way), even for very simple sentences like (16). This is supported by the findings in Ackema and Neeleman (2004) who show many examples of heterogeneous merge (words with morphemes, words with phrases, phrases with morphemes, etc.).

Kayne (2017) argues that even phonemes or the phonological features making up phonemes must be thought of as undergoing Merge. Assuming standard bottom-up merge (see below section 9), this leads Kayne to conclude that phonemes (or phonological features) must be the building blocks of syntax (i.e. must be in the Numeration), even if locality constraints must then be invoked to exclude merger of phonemes to phrases. A similar conception is advocated in Boeckx (2015), who takes the elements undergoing Merge to be much smaller than words, what he calls 'Lexical Precursor Cells'.

Kayne also takes his conclusion about phonemes undergoing Merge to argue against 'late insertion' models of morphology, such as Distributed Morphology (Halle and Marantz 1993) or any model of grammar in which morphological realization is part of the externalization procedure (4iii).

Derivational interaction (layered derivations) as defined in section 5, provides a natural locality constraint on merger of phonemes or phonological features. Assuming phonemes to be structured in the sense of (21), they must be the output of a derivation as modeled in (4). Such a phoneme may then be included in a Numeration containing other phonemes to yield morphemes in another derivational run. And morphemes may be included in the next derivation, etc. (In principle, nothing excludes the merger of phonemes with morphemes, or even words and phrases, given the potential heterogeneity of Numerations (15c). I assume that interpretability in the externalization procedure (4iii) limits the range of possibilities here, leaving the door ajar for marked phonology-syntax interactions of the type identified in Inkelas and Zec (1990).)

However, this system where phonemes or phonological features participate in derivations is not in principle incompatible with 'late insertion', or morphology-after-syntax more generally. To see this, consider the workings of a 'morphology-after-syntax' model in more detail.

Consider once more (1), where α = *the world* and β = *changes*. *The world* is inherently third person singular (3SG). It is quite possible that these features are generated separately from the lexical items *the* and *world*; if so, the Numeration yielding *the world* contains these features in addition to the elements *the* and *world*. Now in a morphology-after-syntax model, *the* and *world* lack phonological features, so we should perhaps write them differently, as the and world. The same with *changes*, which should be changes. Changes also has the features 3SG, but here the features are not inherent to the verb, but derived in some way from the world through subject-verb agreement. 3SG being expressed by the suffix *-es*, and continuing to assume morphology-after-syntax, we should write the verb without the suffix as change.

Importantly, it is much harder to think of the 3SG features of *changes* as being merged with change, than it is to think of the 3SG features of *the world* as being merged with the world. This is because the 3SG of *changes* are acquired in the course of the derivation, and hence should not be part of the Numeration. Assume a process of feature sharing (Koster 1987) to accompany Internal Merge, in the sense that the first member in the ordered pair resulting from Merge shares its features (marked superscript *F*) with the second member of the ordered pair:

(22) $\{ \alpha^F, \beta \} \rightarrow \{ \alpha^F, \{ \alpha^F, \beta \} \} \equiv \langle \alpha^F, \beta \rangle \rightarrow \langle \alpha^F, \beta^F \rangle$

If so, the externalization procedure receives for (16) an input like (23).

(23) $\langle [\text{the world}]^{3SG}, \text{change}^{3SG} \rangle$

Morphology after syntax entails that the externalization procedure involves a process of insertion, or better: replacement, such that the elements the procedure encounters as output of Narrow Syntax are replaced by vocabulary items. In the case of elements with inflectional features, these vocabulary items are taken from paradigms, such that the element from the paradigm that best matches the inflectional features is selected for insertion/replacement.

More exactly, in our example change points to a vocabulary item which is in fact a set of inflected forms, including *changes*. This presupposes a repository of forms, organized in paradigms in some nontrivial way. The question is, where do these forms come from?

This question now receives a simple answer. Since the forms are complex, they must be derived syntactically via (4). Nothing precludes that they are. We only need to allow that derivations feed into the vocabulary, in addition to supplying Numerations with complex elements. It would in fact require a stipulation to bar this possibility of derivations feeding into the vocabulary.

This shows that extending merger to features, phonemes, and morphemes is not inconsistent with 'late insertion' (morphology-after-syntax) of the type detailed here.

We conclude:

(24) The output of a derivation (4) is accessible to the Numeration (4i) and to the Externalization Procedure (4iii)

Since External Merge (6) is barred, the output of a derivation is not directly accessible to Narrow Syntax (4ii).

We have to stipulate what 'accessible' in (24) means. 'Accessible to the Numeration' means: can be included in a Numeration. 'Accessible to the Externalization Procedure' may mean various things, depending on the nature of the externalization process. In the example discussed, 'accessible' means: available for vocabulary replacement. But this is limited to the lexicalization aspect of externalization. This raises the expectation that the output of a derivation may also in some way be accessible to the interpretative aspect of the externalization procedure (i.e. for something we would have called 'LF-insertion' in the Government-Binding framework or perhaps 'Late Merge' in earlier minimalism). We return to this in section 7.3.

7. Movement

The hypothesis advanced in this paper is that syntactic structure is the outcome of a derivation as in (4), where Narrow Syntax turns an unordered set into an ordered n-tuple via Internal Merge (1). Each derivation is in fact a network of such derivations, where the output of one derivational run may feed into the Numeration for the next derivational run, as well as into processes of the Externalization Procedure (4iii).

The system described here limits movement processes to the transformation in (1), externalizing a member of the initial set. This leaves no room for movement processes considered in standard generative theory, such as passivization, object shift (A-movement), topicalization, wh-movement (A'-movement), and head movement. This section shows that eliminating these movements from the model of grammar does not entail a loss in explanatory adequacy.

7.1 Head movement

I assume with Chomsky (2001) and Zwart (2005, 2017) that head movement is not an operation of Narrow Syntax. As Chomsky (2001) argues, head movement violates the

Extension Condition (2) and has only limited effects on the Externalization Procedure (4iii)(lacking any semantic effects). Zwart (2005) argues that head movement (in particular verb movement) can be understood as a function of Merge, if it is defined as a morphological process marking of the left edge of the sister of some constituent (the first constituent, in verb-second configurations). Zwart (2017) argues that auxiliaries in periphrastic tenses are absent from Narrow Syntax (a periphrastic form being the output of a separate derivation feeding the vocabulary), which would show that verb movement (as demonstrated by auxiliaries) cannot possibly be a process of Narrow Syntax, given our current understanding of the model of grammar (including 'morphology-after-syntax').

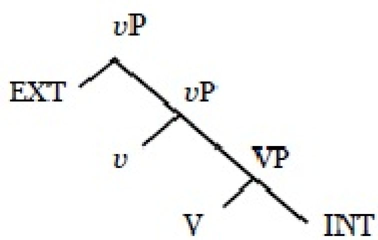
If these observations are sufficiently general, there is no need to complicate Narrow Syntax (4ii) by including a mechanism of head movement.

7.2 A-movement

A-movement in Government-Binding Theory and Minimalism is a mechanism linking thematic positions and grammatical function positions. Since grammatical functions are expressed almost uniquely by noun phrases, A-movement describes a kind of noun phrase dependency (identified with binding in Government-Binding theory).

Let us retain from this tradition that noun phrases need both thematic and grammatical function properties in order to be interpretable as key components of a clause. In standard approaches, thematic properties are a function of structure in the following way: a noun phrase that is merged with a verb (or verbal root) is interpreted as an internal argument of that verb; a noun phrases that is merged with a projection of little *v* is interpreted as an external argument of the little *v*-verb complex. (Little *v* is the agentive/causative component of a transitive or unergative verb.) Thematic properties, then, are associated with specific syntactic positions:

(25)



Grammatical functions (subject, object) are traditionally associated with specifiers of functional projections: initially the designated projections AgrSP (for subjects) and AgrOP (for objects), later TP and *v*P, respectively. Movements from thematic positions to these specifier positions are strictly local, bound to a single finite clause domain (*pace* Ura 1994).

The existence of these movements (A-movement) is based on the premiss that argument noun phrases are generated in positions associated with the *v*-V complex (25). In a layered derivations framework, this premiss can no longer be sustained. As already noted by Hale and Keyser (1993), the *v*-V complex, while structured, is also clearly a lexical item with often highly idiomatic properties. These properties can only arise during the Externalization Procedure (4iii), and are in fact one of the hallmarks of derivation layering (Zwart 2009a).

Assume, then, that a transitive verb like *kill* is the output of a derivation built on the

Numeration in (26).

(26) { *v*, $\mathbb{kill}$ }

Narrow Syntax (4ii) turns (26) into the ordered pair (27), which is externalized as the single verb *kill*.

(27) ⟨ *v*, $\mathbb{kill}$ ⟩ → *kill*

Kill is then included in the numeration for the next derivational run (one that might yield a transitive clause with *kill* as its core). But the thematic positions associated with the *v*-V complex in (25) are not represented in (27). So the arguments of *kill* cannot be part of the derivation in which *kill* is derived (i.e. (26)-(27)). They are merged in the next derivation (the one that yields a transitive clause with *kill* as its core).

Crucially, this removes any evidence or conceptual argumentation for generating the arguments of *kill* (or any verb) in close proximity to the verb. Minimally, we need the information that *kill* can be meaningfully associated with two arguments (arguably, this information is a function of the derivational history of *kill* illustrated in (26)-(27), the agentive element deriving from the presence in the Numeration of little *v*). This information now must be expressed as a structured list of features of *kill*:

(28) *kill* $_{\langle \mathrm{AG,TH} \rangle}$

Consider now the derivation of a transitive clause with *kill* as its core. We start from the Numeration in (29), where *kill* is the output of a previous derivation as discussed above.

(29) { spiders, mosquitos, kill $_{\langle \mathrm{AG,TH} \rangle}$ }

Internal Merge of *spiders* gives us the ordered pair in (30).

(30) ⟨ spiders, { mosquitos, kill $_{\langle \mathrm{AG,TH} \rangle}$ } ⟩

I assume that the ordered pair in (30) gives rise to a limited range of interpretations at the Externalization Procedure (cf. Zwart 2009a), among which is the interpretation that *spiders* is the subject of its sister (which we may call its predicate). More generally, we may claim as a default case that

(31) The first element merged is interpreted as subject

I moreover assume that the subject-predicate interpretation of (30) is an instantiation of a more general phenomenon, which is that the asymmetry between the members of an ordered pair generated by Internal Merge creates a dependency, such that the second element is a dependent of the first (Zwart 2004). (Feature sharing, discussed above, is another function of the dependency between the members of an ordered pair.)

On these assumptions, the grammatical function of *spiders* is a direct result of Internal Merge. (In earlier parlance, the noun phrase is base-generated in its subject position.) But at this point it lacks an argument role. Let us mark the dependent as 'owing' a thematic role to the subject:

(32) $\langle$ spiders, $\{$ mosquitos, kill $_{\langle AG,TH \rangle}$ $\}_{(SU=?)}$ $\rangle$

Similarly after Internal Merge of the object *mosquitos*:

(33) $\langle$ spiders, $\langle$ mosquitos $\{$ kill $_{\langle AG,TH \rangle}$ $\}_{(SU=?)(OB=?)}$ $\rangle \rangle$

Thematic interpretation of the noun phrases is now simply a function of (ordered) feature resolution, which I will assume takes place after Internal Merge of the verb *kill*:

(34) $\langle$ spiders, $\langle$ mosquitos $\langle$ kill $_{\langle AG,TH \rangle}$ $\{ \varnothing \}_{(SU=AG)(OB=TH)}$ $\rangle \rangle \rangle$

Essentially, the process described is one of thematic role binding (REF).

On this approach, processes affecting the association of grammatical functions with thematic roles (such as passive, middle, inverse voice, as well as applicative, causative) center on the properties of the verb and its list of associated argument roles (suppressing the agent role in the case of passives, say). Burzio's Generalization (balancing the number of grammatical functions and thematic roles, such that an object cannot be merged when the agent role of the verb is suppressed) follows trivially.

Long-distance 'A-movement' is limited to (infinitival) complementation structures. The difference in opacity between complement clauses on one hand and adjunct/subject clauses on the other is easily understood within a layered derivation approach (cf. Zwart 2009a, Toyoshima 2007). Only subject/adjunct clauses must be the output of a separate derivation, and hence atomic and opaque, assuming:

(35) Generalized Integrity
Given a derivation $\Delta = \langle$ Num, NS, Ext $\rangle$,
operations in NS affect no element ε such that $\varepsilon \notin$ Num

According to (35), Merge (the only operation in Narrow Syntax (NS) may only merge elements that are in the Numeration (Num), not subparts of elements that are in the Numeration.

It follows that the elements of a transparent infinitival complement to a verb V must be members of the same Numeration of which V is an element. Thus, in (36) the Numeration must be as in (37).

(36) Spiders seem to kill mosquitos

(37) { spiders, mosquitos, seem, kill }

This would be undesirable if Merge, applying to (37), were to yield the wrong constituents, but they do not (cf. Zwart 2009a). (Alternatively, *seem to kill* is itself the output of a subderivation.)

Abstracting away from verb placement, Merge turns (37) into (38):

(38) $\langle$ spiders, mosquitos, seem, kill $\rangle$

The association of grammatical functions with thematic roles now follows if verbs like *seem* are not associated with any thematic roles:

(39) $\langle$ spiders, $\langle$ mosquitos $\langle$ seem $\langle$ kill $_{\langle \mathrm{AG,TH} \rangle}$ $\{ \varnothing \}_{(\mathrm{SU=AG})(\mathrm{OB=TH})}$ $\rangle \rangle \rangle \rangle$

If this is the correct analysis, it follows that Burzio's Generalization also applies to constructions with verbal complexes like *seem to kill* (as shown in Zwart 2001).

7.3 A'-movement

The model of A-movement described above is not suitable for the analysis of A'-movement, the syntactic process by which operator-variable constructions are standardly derived (Chomsky 1977):

(40) operator-variable constructions
a. *wh-movement*
   (I wonder) which mosquitos [spiders kill]
b. *relative clause formation*
   mosquitos (that) [spiders kill]
c. *'topicalization'*
   (but) mosquitos, [spiders kill]

In these constructions *(which) mosquitos* is an operator binding a variable inside the bracketed clause. (Alternatively, *(which) mosquitos* is associated with an empty operator binding the variable, as is the standard analysis of relative clauses, and has also been proposed for topicalization—we abstract away from this refinement here.) Both the variable and the operator are associated with the thematic role (THEME) and grammatical function (object) of *mosquitos* in (41).

(41) Spiders kill mosquitos

This is a standard argument for a transformational analysis, assuming that the operator is merged in the object position first, and then moved (fronted, cf. Kayne 1994) to the operator position, leaving as its 'trace' the variable. As a result, the operator carries the case morphology associated with the variable position (as can clearly be seen in case-marking languages).

It seems, then, that A'-movement is an additional, second-order process that cannot be easily reformulated in a 'base-generation' approach, at least not as easily as A-movement (see also Den Dikken to appear).

Formally, it would not be hard to add an A'-conversion process to the system described so far. This process would take the ordered n-tuple created in Narrow Syntax as its input and externalize one of the elements of the ordered n-tuple, mimicking traditional A'-movement. But this would add an entirely new operation to the model of grammar, raising questions of triggers of movement that we wish to avoid. What follows is an attempt to derive A'-movement using only the limited mechanisms defined so far (i.e. Internal Merge and derivational interaction).

Recall (from section 6) that we were compelled to describe derivations as networks of derivations. Moreover, we allowed for the output of a derivation to be accessible at various points in other derivations, notably:

(42) derivational interaction points
a. Numeration construction
b. Externalization procedure

That is, the output of a derivation may be included in the Numeration for another derivation, but may also feed the Vocabulary, which is accessed during Externalization.

Assume that Externalization operations are only triggered to enable interpretation for sound and meaning. One of these operations is vocabulary insertion, which is the process of replacing abstract features with phonemes/morphemes. I would like to propose a similar process relating to semantic interpretation, replacing an unbound variable by an operator-variable pair (as a necessary step towards λ-conversion).

This presupposes:

(43) a. Numeration may include a variable element (vbl)
b. a ban on vacuous quantification (i.e. *vbl* must be bound at some point)

The ban on vacuous quantification can only apply during the Externalization Procedure (Narrow Syntax being arguably blind to it).

Consider now a Numeration containing a variable (44) and the ordered n-tuple derived on the basis of that Numeration (45)(using *vbl* instead of the object *mosquitos*).

(44) { spiders, *vbl*, kill $_{\langle AG,TH \rangle}$ }

(45) ⟨ spiders, *vbl*, kill $_{\langle AG,TH \rangle}$ ⟩

As before, *spiders* is the agent/subject, and the variable replaces *mosquitos* as the theme/object. (45) violates the ban on vacuous quantification (43b), and we can mark it as such:

(46) ⟨ spiders, *vbl*, kill $_{\langle AG,TH \rangle}$ ⟩*e*

Assume now that we have available as the output of another derivation the phrase *which mosquitos*. Then what the Externalization Procedure can do is construe a designated binary set (47) and apply Internal Merge to that set in the usual fashion:

(47) { [which mosquitos], ⟨ spiders, *vbl*, kill $_{\langle AG,TH \rangle}$ ⟩*e* }

This will create the new ordered pair in (48).

(48) ⟨ [which mosquitos], ⟨ spiders, *vbl*, kill $_{\langle AG,TH \rangle}$ ⟩*e* ⟩

As before, the second element in the ordered pair is dependent on the first, and we may take the dependency to be interpreted as a variable binding relation. This removes the vacuous quantification, turning (46) into an interpretable object.

(Technically, we could describe the process (47)-(48) as a regular derivation of the type in (4), but that would ignore the special character of this process, being limited to a pair of elements and being triggered by requirements of Externalization.)

In traditional terms, this proposal, perhaps shockingly, relegates wh-movement to the collection of ill-understood postsyntactic operations. However, in the model of grammar contemplated here, there is nothing irregular about it, given the premiss of derivational interaction as detailed in (42).

Moreover, it can be shown that locality effects associated with A'-movement follow naturally on this approach.

To see this, consider that the approach to A'-movement just outlined implies (informally):

(49) An ordered n-tuple containing a variable is merged with an operator binding that variable

We take this to entail that if the Externalization Procedure merges an ordered n-tuple containing a variable with some other element or set of elements than an operator binding that variable, interpretability is forever lost.

Consider standard wh-island constructions like (50), where variables are indicated by *e*.

(50) * Which mosquitos do you wonder why [spiders kill *e* *e*] ?

This example contains two variables, both within the bracketed element *spiders kill*. On our approach, the element in brackets is the output of a derivation presented to the Externalization Procedure, i.e. an ordered n-tuple containing two unbound variables. Merger with *why* during externalization takes care of one variable, but the other one remains unbound, yielding vacuous quantification. The derivation continues by including *why spiders kill* in the Numeration for another derivational run, together with *wonder*, *you*, etc., in violation of (49).

The only way to salvage multiple variables in an ordered n-tuple delivered to the Externalization Procedure would be to merge two operators (or a conjunction of operators). I submit that this underlies the phenomenon of multiple wh-movement (Richards 2001)(Tucking in effect).

Long-distance A'-movement is not constrained by (49), as long as the embedded clause is not construed in a separate derivation. Embedded clauses *must* be created in a separate derivation when they are in subject or adjunct position (like all complex left branch elements). But nothing forces complement clauses to be created in a separate derivation. For example, for (51) the Numeration can be as in (52).

(51) you think spiders kill mosquitos for fun/*vbl*

(52) { you, think, spiders, kill, mosquitos, [for fun]/*vbl* }

If the adjunct is not *for fun* but a variable, the n-tuple in (53) results.

(53) ⟨ you, think, spiders, kill, mosquitos, *vbl* ⟩*e*

This then must be merged with an operator in the Externalization Procedure to avoid vacuous quantification:

(54) { why, 〈 you, think, spiders, kill, mosquitos, *vbl* 〉*e* }

yielding the ordered pair

(55) 〈 why, [you think spiders kill mosquitos]*e* 〉

and the clause

(56) (I wonder) why you think spiders kill mosquitos

More generally, it is the immediacy of fixing unbound variables during Externalization by merging the element containing an unbound variable with a binding operator that yields locality.

Adjuncts containing an unbound variable cannot be included in the Numeration for another derivation without violating (49), giving us the Condition on Extraction Domains of Huang (1982).

(Truswell effects)

Conjuncts containing an unbound variable cannot be included in the Numeration for a derivation yielding a conjoined structure, giving us part of the Coordinate Structure Constraint of Ross (1967)(the part prohibiting extraction out of conjuncts). The other part (prohibiting extraction of conjuncts) is derived as well, since conjunctions containing a variable as one of their conjuncts cannot be included in any Numeration without violating (49); hence these cannot occur.

(ATB)(Kehler effects)

Weak islands (which can only be detected by subject or adjunct extraction) follow as well, on the assumption of Cinque (1990, chapter 3) that empty arguments in object position can be construed as empty pronominals, so that only in subject and adjunct position do we run the risk of vacuous quantification.

(wh-in-situ)

(intermediate landing effects)

8. The externalization procedure

We are now in a position to take stock as to the nature of the Externalization Procedure (4iii).

The Externalization Procedure takes as input an ordered n-tuple created in Narrow Syntax (4ii) through iterative application of Internal Merge (1). The procedure is driven by the need to assign sound and meaning properties to the input. We need to consider the question how the procedure assigns these properties, how the procedure is structured, and how much of its operation is constrained by the derivational history of the ordered n-tuple.

Minimally, the Externalization Procedure needs to:

(57) Externalization
- i. replace abstract syntactic elements by items from the vocabulary (lexical insertion)
- ii. provide an operator to bind unbound variables
- iii. linearize the ordered n-tuple in a modality specific organization (typically a string of sounds)
- iv. apply semantic interpretation rules (such as λ-conversion, reference resolution, etc.)

(57i) is necessitated by the principle of 'morphology-after-syntax' (late insertion). As we have seen, by accessing the vocabulary, the Externalization Procedure encounters elements that are the output of separate derivations (independently or in the context of inflectional paradigms).

(57ii) is proposed here.

(57iii) is an inevitable element of the model of grammar entertained in minimalism, and also here. I assume that by default, the ordering of elements in an ordered n-tuple translates directly into a linear string (Zwart 2011a):

(58) $\langle \alpha, \beta \rangle \rightarrow \alpha\beta$

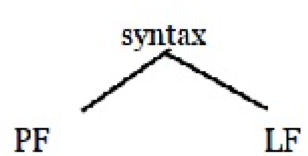
This replaces the Linear Correspondence Axiom of Kayne (1994), where string order is a function of the global computation of c-command relations among all the elements in a structure (Zwart 2004).

Kayne's generalization that wh-operators are left peripheral (Kayne 1992) follows from (58) as well, as does the typological generalization that subjects in the default case precede predicates (Tomlin 1986, Zwart 2009b). But many deviations from the straightforward conversion in (58) must be accounted for, notably having to do with verb placement (Chomsky 2001, Zwart 2005), but also phenomena like cliticization, scrambling, and the like.

(57iv) includes all processes traditionally associated with Logical Form. At minimum, it interprets the asymmetries inherent to the ordered n-tuple in terms of semantic interpretation (scope, binding, predication, quantification, ellipsis resolution, λ-conversion, etc.). Also, it should interpret the operator-variable relation in such a way that connectivity effects ('reconstruction') are accounted for, ideally as a function of the ordered pair created in the context of (57ii).

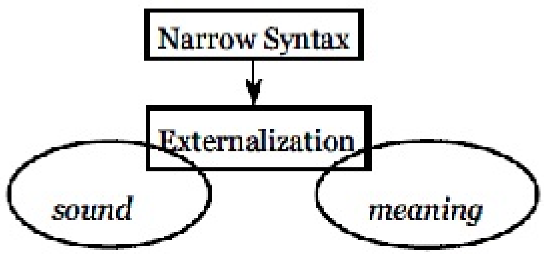
Traditionally (within generative grammar), the interpretative processes having to do with sound (Phonetic Form or PF) and meaning (Logical Form or LF) diverge after the syntactic part of the derivation (the so-called T-model):

(59)



On our approach, some processes relevant to semantic interpretation feed into processes dealing with sound (like the merger of the operator described in 7.3). Likewise, what limited effects of verb movement on semantic interpretation have been found (see Holmberg 2015 for a summary) suggest some impact of linearization on semantic interpretation. If so, the model in (59) is certainly too much of a simplification. I prefer to think of the Externalization Procedure as a single component interfacing with other components of the model of the mind/brain (circled in (60)).

(60)



These other components then must be thought of as being selectively sensitive to the output of the Externalization Procedure (barring any future recourse to Full Interpretation in the sense of Chomsky 1991).

9. The direction of structure building

In the model of grammar of standard minimalism (Chomsky 1995), structure is built from the bottom up, although top-down structure building alternatives have been proposed as well (Phillips 2003, Chesi 2007, Zwart 2009a, Den Dikken to appear). We close by considering the question how the system proposed here should be characterised in terms of direction of structure building.

Recall that our model differs from standard approaches in not assuming a process of transfer from the Numeration to the Object Under Construction (OUC). The Numeration *is* the OUC, undergoing a transformation from an unordered set to an ordered n-tuple. There is, therefore, no question of building up an OUC, either bottom-up or top-down.

Perhaps that means the question can be dropped altogether, possibly not a bad result. Still, we may wonder whether the question can be rephrased, abstracting away from the metaphoric concept of transfer for structure buidling.

Intuitively, structure building in a bottom-up derivation affects a node that has no mother, and structure building in a top-down derivation affects a node that has no daughters.

(61) *bottom-up*
a) node affected has no mother node
b) creates a new node that has no mother node

(62) *top-down*
a) node affected has no daughter node
b) creates a new node that has no daughter node

In our system, the node affected is the unordered (part of the) set that is the Numeration. The definitions of 'mother' and 'daughter' can be given in derivational terms as in (63):

(63) a. *mother*
α is the mother of β iff α is the result of Merge affecting β
b. *daughter*
α is the daughter of β iff β is the result of Merge affecting α

Given these definitions, the Numeration has no daughters: the Numeration is not the result of Merge. (In this model, the Numeration is a given set, and the model does not specify the process by which its members come to be in the set.) So the node affected (the Numeration) never has a daughter node, one of the hallmarks of top-down derivation (62a).

On the other hand, the Numeration does undergo Internal Merge at each step of the derivation, creating a new node as a result (the ordered pair), of which it is a daughter by definition (63b). This goes against the property of top-down derivations of creating a new node that has no daughter node (62b).

It is also wrong to say that the node affected (the Numeration) has no mother node (as the bottom-up process would require by (61a)). This is because for most of the derivation (all except the first step), the Numeration, being a member of an ordered pair, has a mother node that is the result of the Numeration undergoing Merge. So this would speak against bottom-up status.

Likewise, the operation Merge does not create a new node that is not the daughter of any other node (i.e. a root), given that Merge typically affects a Numeration that is a member of an ordered pair (that ordered pair itself being the result of Merge affecting the Numeration). So none of the properties of bottom-up derivation apply.

We are left with a mixed result, the process having more properties in common with top-down derivation, but nonetheless not representing a bona fide case of top-down derivation.

We may also apply another criterium, which refers to the internal complexity of elements in the structure in relation to the development of the structure over time. For this we may state:

(64) *bottom-up*
a. the internal structure of α undergoing Merge is given (actual)
b. Merge does not increase the internal structure of α undergoing Merge

(65) *top-down*
a. the internal structure of α undergoing Merge is not given (potential)
b. Merge does increase the internal structure of α undergoing Merge

On this criterium, the process contemplated here has much more in common with bottom-up derivation. It is in fact an inherent property of the system proposed here that members of the Numeration may have a derivational history, being potentially the output of a different derivation. So the internal structure of any element undergoing Merge (i.e. any α in (1)) is given, and is not further increased by Merge. This points unequivocally to bottom-up status, given (64)-(65).

All in all we may conclude that the process of structure building proposed here is not easily characterized as either a bottom-up or a top-down procedure, suggesting the concept of direction of structure building does not apply to the system outlined in this paper. Inasmuch as direction of structure building is a concept tied to the metaphor of transfer (of elements from the Numeration to the Object Under Construction), a process inherent to

External Merge which we propose to eliminate, moving beyond the issue of direction of structure building is consistent with the general approach taken here, and perhaps a result of some significance.

10. Conclusion

In this paper I have argued that syntactic structure is the result of a process that turns an unordered set of elements into an ordered pair (or an ordered n-tuple). This process is driven by the need to create an object that the Externalization Procedure (the interface components of Chomsky 1995) can process. It stops as soon as all the members of the unordered set (the Numeration) have become members of the ordered n-tuple.

The process by which the set is turned into an ordered n-tuple is Internal Merge. I have argued that this is the only process active in Narrow Syntax, eliminating External Merge. I have also argued that derivations are invariably networks of derivations, each a triple ⟨Numeration, Narrow Syntax, Externalization Procedure⟩ which yields an output that may be included in the Numeration for a next derivation (the layered derivations proposal of Zwart 2009a). In addition, the output of a derivation may feed into the Vocabulary needed for lexical insertion (as part of the Externalization Procedure) and may be called upon by the Externalization Procedure to act as an operator to an unbound variable. I have proposed that the latter process replaces A'-movement, showing that locality effects associated with A'-movement reduce to a simple ban on vacuous quantification applying at Externalization.

The process described cannot be simply characterized in terms of bottom-up or top-down derivation, suggesting this distinction is an artificial by-product of the commonly held separation between a Numeration and an Object Under Construction, requiring a process of transfer of elements from the Numeration to the Object Under Construction, already denounced in Bobaljik(1995). If we are correct, this transfer process, External Merge, is no more than a metaphor and should not be considered a fundamental process of structure building in a minimalist model of grammar.

References

Ackema, Peter and Ad Neeleman. 2004. *Beyond morphology: interface conditions on word formation*. Oxford: Oxford University Press.

Bobaljik, Jonathan David. 1995. In terms of merge. *MIT Working Papers in Linguistics* 27, 41-64.

Boeckx, Cedric. 2015. *Elementary syntactic structures*. Cambridge: Cambridge University Press.

Chesi, Cristiano. 2007. An introduction to phase-based minimalist grammars: why move is top-down and from left-to-right. *CICSL Working Papers on Language and Cognition* 1, 38-75.

Chomsky, Noam. 1977. On wh-movement. In Peter Culicover, Thomas Wasow and Adrian Akmajian, eds., *Formal syntax*, 232-286. New York: Holt, Rinehart and Winston.

Chomsky, Noam. 1991. Some notes on economy of derivation and representation. In Robert Freidin, ed., *Principles and parameters in comparative grammar*, 417-454. Cambridge:

MIT Press.
Chomsky, Noam. 1995. *The Minimalist Program*. Cambridge: MIT Press.
Chomsky, Noam. 2001. Derivation by phase. In Mike Kenstowicz, ed., *Ken Hale: a life in language*, 1-52. Cambridge: MIT Press.
Chomsky, Noam. 2008. On phases. In Robert Freidin, Carlos P. Otero and Maria Luisa Zubizarreta, eds., *Foundational issues in linguistic theory: essays in honor of Jean-Roger Vergnaud*, 133-166. Cambridge: MIT Press.(?)
Chomsky, Noam. 2013. Problems of projection. *Lingua* 130:33-49.
Cinque, Guglielmo. 1990. *Types of A'-dependencies*. Cambridge: MIT Press.
Den Dikken, Marcel. To appear. *Dependency and directionality*. Ms, Research Institute for Linguistics, Hungarian Academy of Sciences, Budapest.
Fortuny, Jordi. 2008. *The emergence of order in syntax*. Amsterdam: John Benjamins.
Hale, Ken and Samuel J. Keyser. 1993. Argument structure and the lexical expression of syntactic relations. In Ken Hale and Samuel J. Keyser, eds., *The view from Building 20: essays in linguistics in honor of Sylvain Bromberger*, 53-109. Cambridge: MIT Press.
Halle, Morris and Alec Marantz. 1993. Distributed morphology and the pieces of inflection. In Ken Hale and Samuel J. Keyser, eds., *The view from Building 20: essays in linguistics in honor of Sylvain Bromberger*, 111-176. Cambridge: MIT Press.
Hofstadter, Douglas. 2007. *I am a strange loop*. New York: Basic Books.
Holmberg, Anders. 2015. Verb second. In Tibor Kiss and Artemis Alexiadou, eds., *Syntax — theory and analysis: an international handbook*, 342-382. Berlin: De Gruyter.
Huang, C.-T. James. 1982. *Logical relations in Chinese and the theory of grammar*. Dissertation, MIT.
Inkelas, Sharon and Draga Zec, eds. 1990. *The phonology-syntax connection*. Chicago: The University of Chicago Press/Center for the Study of Language and Information.
Kayne, Richard, 1992. Word order. Keynote lecture at the GLOW Colloquium, Lisbon, April 14.
Kayne, Richard. 1994. *The antisymmetry of syntax*. Cambridge: MIT Press.
Kayne, Richard. 2017. What is suppletive allomorphy? On *went* and **goed* in English. Ms., New York University.
Koster, Jan. 1987. *Domains and dynasties: the radical autonomy of syntax*. Dordrecht: Foris.
Kuratowski, Kazimierz. 1921. Sur la notion de l'ordre dans la théorie des ensembles. *Fundamenta Mathematicae* 2, 161-171.
Phillips, Colin. 2003. Linear order and constituency. *Linguistic Inquiry* 34, 37-90.
Richards, Norvin. 2001. *Movement in language: interactions and architectures*. Oxford: Oxford University Press.
Ross, John Robert. 1967. *Constraints on variables in syntax*. Dissertation, MIT.
Tomlin, Russell S. 1986. *Basic word order: functional principles*. London: Croom Helm.
Toyoshima, Takashi. 1997. Derivational CED. *Proceedings of the West Coast Conference of Formal Linguistics* 15, 505-519.
Ura, Hiroyaki. 1994. Varieties of raising and the feature-based Bare Phrase Structure theory. *MIT Occasional Papers in Linguistics* 7.
Zwart, Jan-Wouter. 2001. Object shift with raising verbs. *Linguistic Inquiry* 32, 547-554.
Zwart, Jan-Wouter. 2004. The format of dependency relations. Bloomington SynFest lectures, Bloomington, June 22-July 1.
Zwart, Jan-Wouter. 2005. Verb second as a function of Merge. In Marcel den Dikken and

Christina M. Tortora, eds., *The function of function words and functional categories*, 11-40. Amsterdam: John Benjamins.
Zwart, Jan-Wouter. 2009a. Prospects for top-down derivation. *Catalan Journal of Linguistics* 8, 161-187.
Zwart, Jan-Wouter. 2009b. Relevance of typology to minimalist inquiry. *Lingua* 1589-1606.
Zwart, Jan-Wouter. 2011a. Structure and order: asymmetric merge. In Cedric Boeckx, ed., *The Oxford handbook of linguistic minimalism*, 96-118. Oxford: Oxford University Press.
Zwart, Jan-Wouter. 2011b. Recursion in language: a layered-derivation approach. *Biolinguistics* 5, 43-56.
Zwart, Jan-Wouter. 2015. Top-down derivation, recursion, and the model of grammar. In Andreas Trotzke and Josef Bayer, eds., *Syntactic complexity across interfaces*, 25-42. Berlin: De Gruyter.
Zwart, Jan-Wouter. 2017. An argument against the syntactic nature of verb movement. To appear.