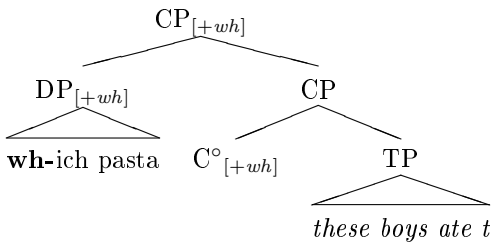
On the Inexistance of Specifiers and the Nature of Heads

michal.starke@nyu.edu [v0.95]

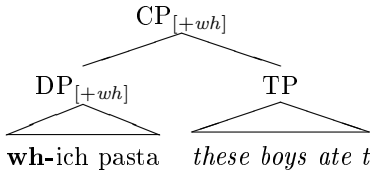
This paper argues that there is no such thing as a "specifier". What we mistook for specifiers are heads (in the sense that they provide the "label" for the merger they enter into). Put differently, there is no such thing as second merge in any theoretically relevant sense: every instance of merge has the properties of first merge.

Traditional representations postulate a "specifier" accompanied by an often invisible head terminal, both containing the same feature α, as in (1) with α=*wh*. Eliminating the invisible terminal causes the occurence of α inside the specifier to directly label the merger, with no duplication, as in (2):

(1) I wonder ...



(2) I wonder ...



In (2), the DP 'which pasta' is the head of CP, in the same sense as the V° terminal 'ate' is the head of VP. No "C°" terminal is involved.

To put the proposal in perspective, consider this informal description: given a "small" representation, such as S'-S-VP or CP-TP-VP, specifiers are essentially a way of "adding space" around heads, so as to accomodate all the necessary material (subjects, wh-phrases, etc.). Once the representations become more

refined (Pollock (1989); Cinque (1999); Rizzi (1997), etc.) however, space becomes abundant and the need to enlarge representations disappears. Space in fact becomes overabundant relative to the elements spelled out, and most "heads" become invisible. At the same time, given that specifiers agree with heads (whether directly, or indirectly with AGREE + EPP), specifiers contain the same feature as heads do. They are thus just as capable of "projecting" a label as the "head" is. Gone the need to add space, and with "specifiers" capable of doing the job alone, postulating an invisible head below each specifier becomes an anachronism, advantageously replaced by (2). The "specifier" has now taken over the role of the head, it has become a complex, non-terminal head. I will argue below that this simplification has many virtues beyond the significant clean-up of the theory that it provides.

Eliminating the notion "specifier" carries one step further the reductionist programme initiated by minimalism: minimalism eliminated the looser notion of governement in favor of the more restricted notions of specifier-head relations. This paper amounts to eliminating specifer-head relations also, and restricting all syntactic relations to the most basic: head-complement relations.

1.1 The 'Doubly Filled Nothing' Generalisation

As a first step in motivating the elimination of "specifier", consider the following generalisation: given finer-grained representations, there is almost no XP in which both the "specifier" and the "head" are filled.[1] Let us strenghten this to what we might call the "Doubly Filled Nothing" generalisation, coming back to apparent exceptions below:

(3) Doubly Filled Nothing: no projection can have both its head-terminal and its specifier present at the same time.

If this is an accurate depiction, recent detailed empirical research is in effect telling us: the theory predicts two positions, but we only ever see one. Why?

Clearly, the optimal answer is that we see only one position because there *is* only one position. And this is what the spec-less (2) implements: If there are no "specifiers", the sole available position is the position of the "labeller", the "head" position. What we are seeing are the heads of the successive projections, with no additional material around them.

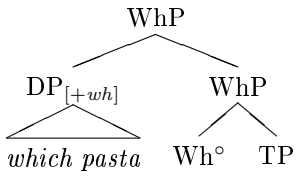
1.2 Why can't "specifiers" project?

The same conclusion comes from an inspection of current theories. As it turns out, the spec-less (2) is the result of pulling out undesirable tacit stipulations.

It is a common assumption the "specifier-head" relation is an identity - 'agreement', 'matching', 'checking' - relation with respect to some feature: both the specifier and the head contain the same feature f, and the "specifier-head" relation builds on this identity. That feature is furthermore the feature which labels the resulting constituent (it is immaterial whether this relation is a primitive, or whether it is derived, as in *Agree* followed by *EPP*).

The fragment in (1) is thus properly represented as:

(4) I wonder ...



If the same feature f is contained in specifiers and in terminals acting as heads, the obvious question to ask is: why does f in specifers never project while the same f in the corresponding terminal does project? What is it that prevents f inside the specifier from projecting?

The answer cannot be in terms of accessibility or usability of f inside the specifier; ie. it cannot be along the lines of 'f inside an XP is too deeply embedded to be visible outside of XP'. This is because the specifier-head relation requires f inside the XP-specifier to be visible outside that XP: f must be visible to the head with which it will enter into a checking relation. Given that f inside the specifier *is* visible to outside processes, why is it usable by the spec-head

relation but not by labelling? Ie. why do 'specifiers' never project?

Equivalently for our purposes, why must each insertion of an XP ("specifier") be preceded by an insertion of a corresponding terminal? Why can't a merged XP be "self-standing", as in (2)?

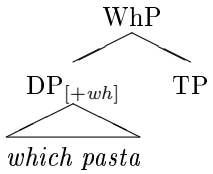
There are currently no answers to this question; current theories resort to tacit assumptions stipulating the need for a terminal along each specifier, or the impossibility for f in a specifier to project, along the lines of either of:

(5) a. Asymmetric Projection [AP]: a feature f in an XP node cannot legitimate its mother (it cannot 'project'), but the same f in an X° node can legitimate a maximal projection (it can 'project')

b. Merging an XP requires prior merging of a corresponding terminal

Of course, such assumptions are undesireable, and are to be derived or eliminated once recognised. Simply eliminating (5) yields:

(6) I wonder ...



Below, I provide half a dozen of other reasons to dispense with the notion of "specifier", before discussing what syntax would look like, once specifiers are a thing of the past.

1.3 The Specifier-of relation is expletive

A third property making specifiers suspicious is a curious asymmetry between the specifier-of relation and the head-complement relation.

The head-complement relation is the basic syntactic relation, with the expected properties: it instantiates the core notion of concatenation (merge) - mapping onto (non-trivial) compositionality - and the set resulting from that composition is subject to the usual labelling algorithm, creating an asymmetry among the members of the set.

The specifier-of relation on the other hand has neither of these expected properties: it maps onto a semantically vacuous identity relationship; and it plays no role in labelling. This makes the specifier-of relation look like an "expletive relation", playing no meaningful role; on a par with expletive pronouns, uninterpretable features or structural case.

If so, the hypothesis that there are specifiers needs serious justification; and again, the optimal state of affairs is that there is no such vacuous relation.

1.4 Specifiers complicate Locality.

Locality is standardly understood as a restriction on identity of two constituents with respect to some feature (quantificational features for A'-chains, agreement or case feature for A-chains). From the point of view of locality principles then, the specifier-head relation is just another case of "identity with respect to some feature". In fact, the specifier-head relation is indistinguishable from an antecedent-trace relation, as far as locality is concerned.

To take an example, consider the classical analysis of a wh-moved element: XP_{wh} $\mathrm{X}^{\circ}{}_{wh}$ ... t_{wh}. The relation between the anteposed XP_{wh} and its trace t_{wh} is restricted by locality because it is an identity wrt wh-features (or quantificational features). The specifier-head relation in CP - $\mathrm{R}(\mathrm{XP}_{wh}, \mathrm{X}^{\circ}{}_{wh})$ - is however also an identity wrt. wh-features, and as far as the locality principle is concerned, it is indistinguishable from the antecedent-trace relation.

It then comes as a surprise that the locality of the specifier-head relation is completly different from the locality of wh-movement. Specifiers thus provoke a locality paradox, forcing some (unresolved) complication to current locality theories.[2]

Again, the whole issue disappears if syntax does not include second-mergers (and no 'checking' configurations need to be enforced).

1.5 Ordering Stipulations.

Another rarely noted fact is that assuming the existence of specifiers forces extrinsing rule-ordering. Some mechanism must ensure that the merger of the would-be head is ordered before the merger of the would-be specifier. Nothing within current theories however forces that - although they all assume such an outcome. To achieve this, either the linearisation algorithm must be made more complex, or some independent stipulation must be added (as in (5)). Again, this complication does not arise if syntactic structures are made only of heads and complements.

1.6 Redundancies

Duplication of every feature. A side effect of the idea that there are specifiers - and its accompanying tacit stipulation (5) - is the duplication of features: every feature present in an XP must be present twice, as a corresponding (local) head must bear it too. Again, once the logic of functional projections is followed through, the scale of the issue becomes non-trivial. As before, no such consequence holds in a specifier-less syntactic structure.

Double-Articulation of Phrase Structure. As a seventh point, consider that fact that specifiers require phrase structure to be split into two hierarchies: nodes are first assembled into "xBar units", and these mini-structures are then assembled into a rigid sequence of projections. There is no clear relationship between these two layers, as contemporary investigations have been able to make very little sense of this, mostly leaving the issue aside for future scrutiny. Again, retiring specifiers relieves us of this unwanted side-effect (by collapsing the two layers into one).

1.7 Triggering insertion: Specifiers versus fseq

Specifiers thus create a lot of trouble for the theory. It therefore comes as a surprise that they are also unnecessary in order to achieve the results they are

used for.

In current theories, the only job of the apparatus surrounding specifiers, ie. (5) and the checking mechanism, is to trigger insertion (merge/move). This is based on the fact that if (5) holds, every insertion of an XP into phrase-structure creates an invariant local configuration between that XP and a corresponding head (the spec-head configuration). It is then trivial to equate the cause of the insertion with the need to create that configuration, which is what all current approaches do (cf. 'checking', 'criteria', etc.).

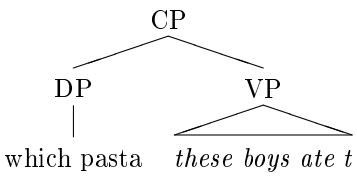
Should there be some other cause for insertion however, the concept of "specifier" would *ipso facto* become jobless and merrily walk into retirement. And there is such a cause.

All modern theories of syntax assume that there is an ordering of projections: S is above VP and not vice versa; or CP is above TP and not vice versa, TP is above VP (and not VP above TP), etc. There are various ways to formulate this restriction, none particularly succesful *qua* explanation. Let us then adopt a simple description:

(7) there exists an 'fseq' - a sequence of functional projections - such that the output of merge must respect fseq.

In the simplified syntactic structures considered so far, fseq = <C, T, V>. Now consider the consequence of not moving the subject DP with a wh-question on the object such as (1-2). In a structure without specifiers, the result is that the T layer is never projected:

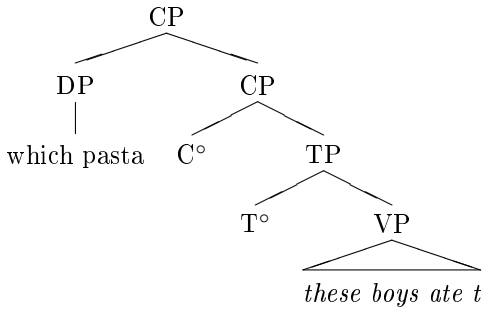
(8) I wonder ...



This produces fseq = <C, V>. But this is illegal, as fseq must be <C, T, V>.[3] In a spec-less theory, insertion is triggered directly by fseq: not moving or over-moving creates the wrong fseq.

Contrast this with a specifier-based structure. In the latter, the absence of the specifier has no bearing on the presence of the corresponding layer, the projection is already present via the prior insertion of a terminal-head:

(9) I wonder ...



That is, in syntactic structures based on the head-complement relation, every insertion (or failure thereof) alters the fseq expressed by the structure, while this is not true in a specifier-based theory. This difference is a direct result of the fact that a head-complement based theory does not duplicate features, each feature is represented only once.

Since (7) regulates possible fseqs, (7) dictates what can (and must) be inserted in a head-complement based representation. The fseq requirement itself thus triggers merge/move. The beauty of this result is that fseq is independently needed in every theory. We thus get a trigger mechanism "for free".[4]

The eighth and final reason to dispense with specifiers is thus that they do not work for us. Worst, they obfuscate a wasy of making the independently needed fseq work for us. Taken together, these eight points show specifiers under a new light: as both useless and harmful.

2 fseq as a replacement specifiers

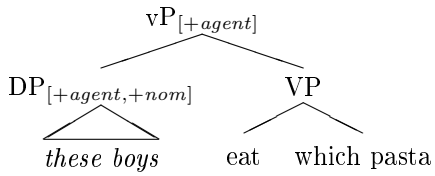
Let us thus proceed with the assumption that "specifiers" (non-first mergers) are an archaic concept, superceeded by rich syntactic structures of the type initiated by Pollock (1989) and followed up by such work as Beghelli and Stowell (1997), Rizzi (1997), Cinque (1999), among many others.

To get a handle on the structures without specifiers, here is the derivation of

(1-2), adopting the common assumption the the object merges with V whereas the subject merges with an agentivity head v, so that the simplified fseq is <C, T, v, V>.

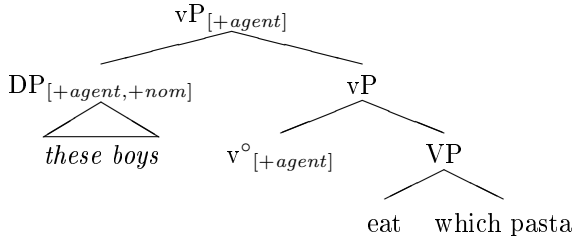
Given the VP={V, object} merger, of no interest here, the starting fseq = <V>. The only legal continuation of this is <v, V>, and the first step must therefore merge some v (agentivity) feature with VP. The DP 'these boys' provides such a property, resulting in:

(10)



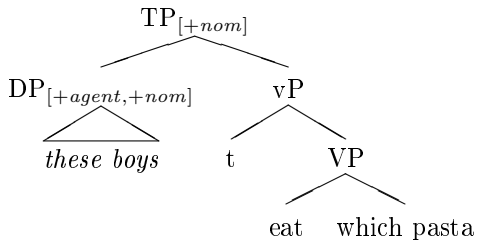
This step is parallel to a specifier-based structure in which the agentive property of the DP-subject enters into specifier-head agreement with an extra v° head:

(11)



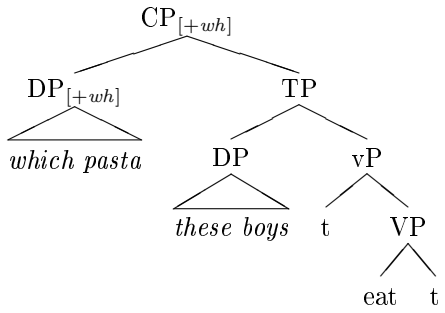
Given current fseq=<v, V> and the target fseq=<C, T, v, V>, the next step must involve projecting a T. There is currently no consensual approach as to how the subject DP relates to T, but for convenience let us adopt a simplification of the idea that the nominative case properties of DP are the relevant nominal feature that are involved in TP 'checking' (adapting from Pesetsky and Torrego (2000)). If so, movement of the DP allows the nominative feature to projects T:

(12)



Again, this is stricly parallel to moving the subject to specTP, but without the specifier and checking machinery. Finally, given this <T, v, V> structure and the target fseq=<C, T, v, V>, the last step must project C, which is achieved by moving the object wh-phrase:

(13)



Generally, then, the logic is that given fseq = <f_n ... f_1>, and a phrase-marker whose top node is F_kP (1<k<n), insertion of f_{k+1} as the next step is forced by the need to build a legal fseq.

2.1 *fseq: laissez faire!*

Now that fseq has come out of the closet, and plays a central role in triggering insertion, let us give it some deserved attention. The identity of the features within fseq is not paramount to the current line of reasoning (although it is of central importance to any current P&P approach). What rather needs to be known is how much variation fseq allows, if any. Let us distinguish two questions:

- is there more than one distinct fseq? (where two fseqs are distinct if they are not in a subset relation)

- given an fseq, must all its features always be present, or are subsets legal instances of fseq?

Consensus seems to have developed around the most restrictive answer to the first question: there is one and only one legal fseq, universally (Starke (1995), Rizzi (1997), Cinque (1999), and numerous others). The second question is more controversial, with three types of answer repeatedly proposed: the 'rigid' approach - everything must always be present (e.g. Starke (1995), Cinque (1999)); the 'peeling' approach - projections can be missing, but only by peeling off from the top (e.g. Radford (1990), Rizzi (1994), Platzack (1996), Cardinaletti and Starke (1999)); or finally the *'laissez-faire'* approach - any projection can be missing (e.g. Wexler (1994)). No argument has been put forward one way or another (pace Cinque (1999), p.133).

Here are however two argument for the third, least normative, line. Consider what happens to NegP in positive clauses. According to the first two approaches, there must be a null 'assertion' operator paralleling the negative operators (i.e. NegP is a PolarityP). The [-positive] operators however have some distinct properties, among which the quality of inducing weak islands ('negative islands'). All things being equal we would expect the [+positive] assertion operators to have the same property. Any value of PolarityP would thus induce weak islands, and hence all clauses should always be weak islands, a *reductio ad absurdum* of the idea of 'positive operators' in a PolarityP of every clause. Is there any evidence that this abstract reasoning is correct? Factives provide a direct indication: it is a traditional observation that verbal predicates whose semantics involves asserting that their complement denotes a fact (*regret*, *know*) induce island effects (the 'factive' island) while their non-fact-asserting counterparts do not (*believe*, *think*); an observation traditionally described with a factive (assertion) operator inducing weak islands. Since the assertion operator is a positive operator ('it is a truly fact that...'), factives indicate that positive operators induce weak island to the same extent as negative operators. The approaches to fseq that force the presence of an assertion operator thus seem stuck with the absurd conclusion

that every clause is a weak island. (In fact, things get worse as the same logic applies to at least [±question] and [±focus]. Each clause is thus predicted to contain at least three weak islands.).

The same logic is massively required by Modifier-movement in Rizzi (2001). Rizzi postulates a third class of features relevant to *Relativised Minimality* to account for the fact that an adverb can be anteposed only if no other adverb intervenes on its path. If all positions were always present, a disaster would result: absence of an adverb would translate into a null adverbial/operator occupying the spec of the relevant projection, and that null adverbial/operator would block movement just as an overt adverbial does. The generalisation that only actual occuring adverbs block anteposition would be lost. The account of adverbial Relativised Minimality thus presupposes that absent adverbs translate into a syntactic structure with interveners not represented.

2.2 *fseq* as an External Interface Condition

We thus need a *laissez-faire* approach allowing a projection to be absent if no element in the enumeration requires it, both in order to allow for the absence of NegP/PolP in positive clauses, and to allow for the absence of unused adverbial positions. A radical implementation would replace (7) with:

(14) there exists an 'fseq' - a unique sequence of functional projections - such that the output of merge must be a subset of fseq.

But this won't do: (14) is both too permissive, allowing many unattested combinations, and expresses the wrong generalization - the generalization that 'anything can drop'. Which is not what the facts are telling us. The facts are telling us that there is some regularity behind 'who can drop'; and it is this regularity that the fseq-requirement needs to express.

As an illustration, take negation, questions and focus again: we never have any reason to think that [+neg], [+wh] or [+foc] are absent from the syntactic representation but interpreted semantically. Such cases - e.g. *plum tastes good*

interpreted as *plum does not taste good* - indeed strike us as aberrant. (Cases where elements are phonologically null but syntactically present, as in the classical analysis of English null complementisers, are of course irrelevant.) The only corresponding cases that ever do suggest radical absence from the syntactic representation (with concomitant semantic interpretation) are [-neg], [-wh] and [-foc].

Building on the traditional observation that speakers have systematic and strong intuitions about 'markedness' of values whereby a declaration is 'unmarked' and an interrogation 'marked', asserting is 'unmarked' and negating 'marked', a flat statement is 'unmarked' and focusing 'marked'; the correct generalization seems to be:

(15) (only) unmarked values can drop

And therefore, descriptively:

(16) there exists an 'fseq' - a unique sequence of functional projections - such that the output of merge must respect fseq, modulo unmarked values.

Why should (16) hold? The fact that only unmarked values can 'drop', entails that dropped items are interpretatively recoverable (i.e. absent nodes are necessarily interpreted as receiving their unmarked value). (15) thus amounts to a 'recoverability condition' in the vein of Vergnaud (1974): an element of fseq can be dropped only if it is recoverable at the LF interface. At the interpretative level, *fseq* is thus always complete (having recovered all pieces missing in the syntactic representation via the markedness route).

This transparently suggests that fseq is not a syntactic principle, but rather an interface condition (a *bare output constraint*, in Chomsky's terminology). (16) can be simplified to:

(17) semantic representations must respect fseq
(i.e. there exists an 'fseq' - a unique sequence of functional projections - such that LF must be readable as an instance of fseq)

Which presupposes:

(18) the interpretive component's reading of syntactic representations is based on recoverability: absence from syntactic structures corresponds to unmarked values

It thus turns out that both the rigid and the *laissez-faire* approaches are right: the rigid approach is correct as far as the interpretative component goes (i.e. everything must always be present); the *laissez-faire* approach is correct for syntax but as an artifact of how interpretation of syntactic structures takes place.

As a second argument for the *laissez-faire* nature of the syntactic fseq requirement (and for the markedness-based formulation of that requirement), consider a massive and curious fact: only some values of a given feature trigger movement. In many languages a [+wh] feature triggers A'-movement to the CP zone, but in no language does a [-wh] feature trigger a similar movement; in many languages a [+foc] feature triggers A'-movement to the CP zone, but in no language does a [-foc] feature trigger a similar movement; or again, some constructions involve A' movement of [+neg] phrases (such as negative inversion in English), but no constructions involve a similar movement triggered by a [-neg] feature. Why?

Again, the generalization seems to correspond to speakers' intuitions about markedness: only 'marked' values ever trigger movement. But why? Why should the same feature be subject to movement under one of its values but not under the other? and why should the moved value systematically be perceived as the 'marked' value by speakers? A rigid view of fseq makes this mystery thicker: it postulates a functional projection corresponding to [-neg], [-foc], [-wh], ready to host movement, leading to the expectation that this projection will be used.

The *laissez-faire* approach on the other hand captures the contrast. Given our two assumptions that (i) features with marked values must be present in the

syntactic structure, but features with unmarked values can be absent and (ii) the incentive for remerger (movement) is to extend fseq so as to comply with (17); it follows that unmarked features will never trigger movement. This is because fseq does *not* need to be extended in the case of unmarked features, and thus movement/remerger is superfluous. Marked features on the other hand correspond to projections that cannot be dropped, and thus fseq needs to be extended, through movement or other means.

2.3 Dependent Insertion

If everything is a head-complement relation, and no additional geometric relation is used by syntax, there should be never be cases in which two different projections seem to interact. That is, there should never be anything looking like a specifer-head relation or a government relation.

Consider then cases like subject-aux inversion in English, (19a), or focus-particle insertion under XP-focalisation in Gun, (19b), from Aboh (1997). Such cases are representative of many other constructions:

(19) a. ... XP_{wh} aux [Subject t_{aux} ... t_{wh}

b. ... XP_{foc} wè [Subject ... t_{foc}

In these cases, the anteposed XP and the Aux/PRT belong to two different projections, and should a priori be entirely independent. It is thus unexpected that the insertion of the anteposed XP 'depends' upon the prior insertion an Aux/PRT. This is a run-of-the-mill case if longer-distance 'specifier-head' relations exist, but out of grasp of the more local 'head-complement' relations.

Notice however that there is a clear pattern to the issue: the insertion of a higher node is dependent on the prior insertion of a node immediately below it. Let us call this 'dependent insertion'.

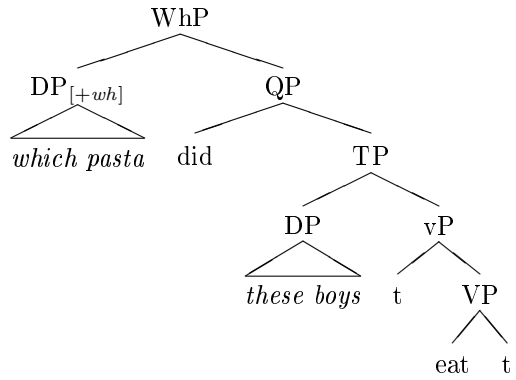
As it turns out, dependent insertion is already expressed in fseq: the fseq requirement dictates that insertion of a given feature depends on prior insertion of all features preceding it in fseq. Temporarily setting aside the optionality of

unmarked features, the logic is as follows: given fseq=<... α β γ ... >, attempts to insert α directly on top of γ will create an illegal fseq and will thus fail. Given fseq=<C, T, V> attempting to merge C directly with VP will fail, since T is missing from the sequence. The same logic applies to subject-verb inversion: if α is the projection of the wh-phrase, and γ the complement of the inverted verb, not inverting the verb creates <... α γ ... >, an illegal fseq. Dependent-insertion thus amounts to "fseq-filling".

The optionality of unmarked features provides an interesting refinement, and a stronger prediction. The fseq <... α γ ... > *is* a legal instance of <... α β γ ... >, provided that β has an unmarked value. The "fseq-filling" explanation thus predicts that dependent insertion will only occur when the sentence requires the *marked* value of the in-between features.

This in turn entails that the underlying cause of dependent insertion is dependence of the value of one feature on the value of anohter, a common phenomenon. As an example, suppose that in wh subject-aux inversion the verb occurs in Q° (a plausible hypothesis since inversion is found with negation, focus and wh):

(20)



In that case, the generalization is that projecting whP requires prior projection of a QP-like projection. The filling-in phenomenon is now a consequence of the fact that +wh requires +Q rather than -Q (together with the fact that Q occurs lower than Wh in fseq).

This type of feature-value dependencies abound in a Cinquean refined fseq: if

tense is split among several projections for instance, a value +future in one projection is 'dependent' on a value +tensed in another, etc. Again, an essentially interpretive phenomenon.

The "filling-in" approach to dependent-insertion makes different predictions with respect to the traditional spec-head approaches: the spec-head approach entails that the only case of dependent-insertion ever found is the case of a YP dependent upon the prior insertion of exactly one X°. The fseq-approach on the other hand has no such entailment: insertion of a YP can require prior insertion of any number of elements, depending on how many fseq items are missing between the insertion site and the top node of the tree to be extended. The elements inserted to fill-in fseq can furthermore be either heads or phrases, while it can only be a head in a spec-head approach.

The fseq-based prediction seems correct. Cases with more than one element filling-in are provided by the Scandinavian wh-constructions where *two* particles ('complementisers') appear between the wh-phrase and the rest of the clause. A case in which a maximal projection is the filler is provided by Jamaican Creole focus structures, in which a particle *precedes* the focused phrase in the CP area (Durrleman (2000)), i.e. a head triggers an XP-filler.[5]

3 fseq and ϕ

According to the fseq logic, the only syntactically relevant geometrical relationship is the complementation relationship linked to sisterhood, now understood as including the 'distant sisterhood' of remergers. This is because syntactic structures are nothing but raw layers of head-complement relationships, with each layer independent of the other.

There are however two types of unexpected relations which seem to occur. First, the agreement relation - though exceedingly rare - does occur with ϕ-features. A second type of case is illustrated by subject-verb inversion in root wh-questions: the insertion of one node (the inverted verb) depends on the

insertion of another node (the wh-phrase).

Both of these unexpected dependencies are straightforward to state in terms of spec-head agreement but seem unstatable with fseq, as they do not involve head-complement relations.

3.1 ϕ-feature "agreement"

How do we express the fact that the morphology of the verb covaries with the morphology of the subject? Let us set our goals higher: since ϕ-feature agreement is the only credible instance of agreement, we not only want to explain how the agreement between the subject and the verb comes about, but we also want to explain why overt morphological agreement only ever happens with ϕ-features.

An obvious observation in this context is that there is one other phenomenon in grammar which again involves covariation in ϕ-features but in no other features: binding. Simple contrasts such as 'she washes herself' vs. 'he washes himself' (cf. *'she washes himself') are formally identical to subject-verb agreement: the morphology of two nodes in the structure covaries with respect to ϕ-features (but no other features).

Why then do we have twice the same phenomenon? A natural answer is to reduce one to the other: verbal morphology enters into a binding relation with the argument of the verb. This captures the similarity between the two cases and solves the configuraional aspect of the problem: no direct geometrical relationship of the "spec-head" type is involved, and thus no such geometrical relation is needed in grammar. The verbal inflection and the covarying argument are in different projections, the relationship is established via binding.

(This line of thought is reminiscent of approaches which hold that verbal agreement is 'pronominal' - e.g. Rizzi (1982). In that terminology, the present argument is that the 'pronominal nature of agr' allows it to enter binding relationships with verbal arguments, and renders the idea of spec-head agreement redundant.)[6]

3.2 EPP

Another type of problematic case is the EPP: the EPP states that one particular projection in fseq is an exception in that it can never be dropped. Why?

Contrary to the cases above, this problem is not particular to the fseq approach. Traditional theories are confronted to the same issue and need to appeal to stipulations such as asserting that a particular feature has the property of being 'strong', or simply stating the facts through an 'EPP' feature.

The analysis of ϕ-agreement in terms of binding however opens up a path for explanation: if verbal agreement enters a binding relationship with the argument, the relationship is either pronominal or anaphoric. Suppose it is anaphoric. It then follows that a local antecedent must be present. Supposing that it is a general property of language to have anaphoric agreement on the verb (often null), the EPP follows.

(This opens up an additional intriguing possibility for the well known correlation between null subjects and rich agreement. We know that anaphors are morphologically poorer than pronouns. Accordingly only languages with poor verbal inflection have an anaphoric inflection, and thus only infl-poor languages show EPP effects. Languages with rich inflection have a pronominal infl along traditional lines and thus escape the EPP requirement.)

3.3 Verb Movement as an artefact of ϕ-placement

Consider the following potentially serious objection: according to a Cinque (1999), verbs can occur anywhere in fseq. To provide these position, it seems that we need to double fseq. Instead of <a b c d>, we need an fseq along the lines of <V1 a V2 b V3 c V4 d>, with each Vn a potential landing site for V°. But of course, doing so basically recreates the null head positions that the specifier-head approach postulates.[7]

The problem disappears once we follow up on a further question: why is it that verbs have this special liberty of appearing anywhere? Why is it that within

a given "construction" of a given language, verb movement has such widespread optionality as it has according to Cinque's system, with no correlation with either semantics or morphology? Why are the boundaries of the 'optionality zone' of verb-movement arbitrary (again, no semantic or morphological correlate)? It is only because verb-movement turns out to have such bizarre properties that the above issue arises.

There are however two regularities that make the situation less desperate than seems at first sight.

First, not all 'verb-movement' is bizarre: subject-aux inversion in questions, V2 antepositions, etc. do not display the above bizarre properties. Such antepositions do not result in optionality, and are correlated to interpretation (±question, ±finite, etc.). It is only the "IP-internal" step of verb-movement which behaves oddly; suggesting that some property if the IP zone is responsible for this erratic behaviour.

Second, verbs are not the only category with these bizarre symptoms. Agreement and negation suffer from the same syndrome: "the evidence points to the possibility of generating a NegP on top of every adverb-related functional projection" Cinque (1999):126, and a similar conclusion is reached by Cinque for "DP-related" projections, i.e. AgrP, or more precisely ϕP.

A slight restatement of Cinque's solution is that ϕ (and neg) can be inserted anywhere in fseq. This (surprising) fact provides the solution to the V-movement issue: since verbs move to ϕ and ϕ can be inserted anywhere in fseq, it follows that verbs will seem to be anywhere in fseq. But of course, this is simply an artefact of the syntax of ϕ.

Given an fseq=<C Mod Adv1 ϕ Adv2 ...>, the verb will appear between Adv1 and Adv2, but if fseq=<C Mod ϕ Adv1 Adv2 ...>, the verb will appear between Mod and Adv1. No enrichment of fseq is involved. What is rather at stake is the bizarre and ill-understood capacity of ϕ to occur anywhere.[8]

(Reducing the verb-movement issue to the ϕ also provides an explanation of the first generalization above: the difference between the 'IP-related' part

of verb-placement and the higher parts derive from the fact that "liberal ϕ insertion" is a property of the IP zone of syntactic structures.)

4 Results

The core asset of the present proposal is to replace a more complex and more heterogeneous theory by a simpler and more homogeneous counterpart thanks to the elimination of the notion of specifiers. The older - traditional - theory takes 'merge', adds the distinction between first merge and non-first merge on top of it (ie. Asymmetric Projection, (5)), grafts an agreement/checking mechanism on top of this, and then proceeds to leave fseq lying around in a dusty corner. In doing so, it ends up with {merge; fseq; AsymmetricProjection; checking}, a curious bag of heterogeneous tools.

In contrast, eliminating specifiers results in everything being a head-complement relation. The corresponding tools are thus {merge; fseq}, which are part of every theory, and form a homogeneous pair: one creates an object, the other decides the type of the object just created.

The newer theory is simpler in that it is a proper subset of the old theory, but maybe more importantly it leaves us with a more homogeneous state of affairs. This gain is supplemented by a number of side-gains: no duplication of features, no gratuitous null heads, no bizarre locality, no expletive relation, no two intermixed theories of phrase-structure, an explanation of the "Doubly Filled Nothing" generalisation, etc.

Notes

[1] Counting traces (but ignoring traces of verb-movement until §1.6#). See Koopman and Szabolscsi (2000) for a recent example of this observation, albeit in a weakened form, restricted to phonologically overt material.

[2] Notice that this also holds if spec-head relations are recast as EPP-relations,

as long as the latter are understood as feature-checking mechanisms.

[3]See below for a discussion of the obligatory or optional nature of elements within fseq. I take them to be obligatory here only to simplify the illustration of the logic

[4]Conversely, another advantage of this result is to squeeze more juice out of pre-existing assumptions: Since every theory has the fseq condition - implicitly or explicitly - leaving it lying around in a dusty corner, barely used, is the worst-case scenario.

[5]An attractive implementation of the fact that unmarked values are syntactically inert - or absent - would be to adopt the approach taken by Kaye et al. (1985) for broadly similar facts in phonology: the unmarked value simply does not exist, features are unary (and presence corresponds to the marked value). Transposing: the reason why the unmarked value never moves and can drop from *fseq* is that it doesn't exist.

Unfortunately this appealing approach does not seem to be workable in syntax, since realised instances of unmarked values of syntactic nodes abound. Notice however that such an answer would amount to a *laissez-faire* approach to fseq: only marked (ie. existing) features can project.

[6]There is an interesting question as to why some languages allow binding (agreement) only in some positions - e.g. Arabic is noted for having the verb covary with the subject only if the subject precedes the verb, and similar facts hold of French participles wrt. an object. This is however orthogonal to the present issue, and depends on two intersting areas: (i) why is coreference possible in some structural configurations but not others, a traditional issue of Binding Theory, (ii) the dependent-insertion phenomenon discussed below.

[7]I owe this formulation of the objection to Luigi Rizzi.

The metaphor of verb 'movement' is used here only for convenience. I am in

fact assuming the more elegant technology proposed by Brody (2000) whereby no actual movement of the verb occurs. This however makes no difference with respect to the argumentation.

[8] A slightly radical solution along the same general line is to allow insertion of the verb directly in the position of ϕ, discarding the traditional idea that verbs are generated in the lowest projection. I will not pursue this line here, but it is interesting to note that there is surprisingly little evidence for the common assumption that verbs must be generated very low.

References

Aboh, Enoch. 1997. From the syntax of Gungbe to the grammar of Gbe. Doctoral Dissertation, University of Geneva.

Beghelli, Filippo, and Tim Stowell. 1997. Distributivity and negation. In *Ways of scope taking*, ed. Anna Szabolcsi, chapter 3, 71–107. Dordrecht: Kluwer.

Brody, Michael. 2000. Mirror theory: Syntactic representation in perfect syntax. *Linguistic Inquiry* 31:29–56.

Cardinaletti, Anna, and Michal Starke. 1999. The typology of structural deficiency: On the three grammatical classes. In *Clitics in the languages of europe*, ed. Henk Van Riemsdijk, 145–290. Mouton de Gruyter.

Cinque, Guglielmo. 1999. *Adverbs and functional heads: A cross-linguistic perspective*. Oxford: Oxford University Press.

Durrleman, Stéphanie. 2000. The left periphery in jamaican creole. Ms., University of Geneva.

Kaye, Jonatha, Jean Lowenstamm, and Jean-Roger Vergnaud. 1985. The internal structure of phonological representations: A theory of charm and government. *Phonology Yearbook* 2:305–328.

Koopman, Hilda, and Anna Szabolscsi. 2000. *Verbal complexes*. Cambridge, Mass.: MIT Press.

Pesetsky, David, and Esther Torrego. 2000. T-to-c movement: Causes and consequences. In *Ken hale: A life in language*, ed. Michael Kenstowicz. Cambridge, Mass.: MIT Press.

Platzack, Christer. 1996. The initial hypothesis of syntax. a minimalist perspective on language acquisition and attrition. In *Generative perspectives on language acquisition*, ed. Harald Clahsen. Amsterdam: John Benjamins.

Pollock, Jean-Yves. 1989. Verb movement, Universal Grammar, and the structure of IP. *Linguistic Inquiry* 20.3:365–424.

Radford, Andrew. 1990. *Syntactic theory and the acquisition of English syntax: The nature of early child grammars of English*. Oxford: Blackwell.

Rizzi, Luigi. 1982. *Issues in Italian syntax*. Dordrecht: Foris.

Rizzi, Luigi. 1994. Some notes on linguistic theory and language development: the case of root infinitives. *Language Acquisition* 3:371–393.

Rizzi, Luigi. 1997. The fine structure of the left periphery. In *Elements of grammar: A handbook of generative syntax*, ed. Liliane Haegeman, 281–337. Kluwer Academic Publishers.

Rizzi, Luigi. 2001. Relativised minimality effects. In *A handbook of syntactic theory*, ed. Mark Baltin and Chris Collins, 89–110. Oxford: Blackwell.

Starke, Michal. 1995. On the format for small clauses. In *Small clauses*, ed. Anna Cardinaletti and Teresa Guasti, volume 28 of *Syntax and Semantics*, 237–269. New York: Academic Press.

Vergnaud, Jean-Roger. 1974. French relative clauses. Doctoral Dissertation, MIT.

Wexler, Ken. 1994. Optional infinitives, head movement, and economy of derivation. In *Verb movement*, ed. Norber Hornstein and David Lightfoot, 305–350. Cambridge: Cambridge University Press.