May 13th, 2010

The following text is a piece of my forthcoming book

Scheer, Tobias forth. How morpho-syntax talks to phonology. A survey of extra-phonological information in phonology since Trubetzkoy's Grenzsignale. Berlin: Mouton de Gruyter.

The text is "wild" in the sense that cross-references to other pieces of the book are not repaired and appear with an error message; also, there is no guarantee that the reference section contains all items quoted in the text.

---

1 2. PF – a strange hermaphrodite animal

2 2.1. Clean syntax, dirty phonology/PF?

3 2.1.1. Minimalism shrinks syntax

A pervasive effect of the minimalist programme is that a whole lot of phenomena and mechanisms that were held to be syntactic in earlier practice are unloaded from syntax. As indicated by its name, the minimalist logic is indeed

(1) "to examine every device (principle, idea, etc.) that is employed in characterizing languages to determine to what extent it can be eliminated in favor of a principled account in terms of general conditions of computational efficiency and the interface condition that the [language] organ has to satisfy for it to function at all." Chomsky (2004:106)

This is to say that anything that syntax is supposed to do beyond the concatenation of pieces must have an extra-syntactic reason, and the two reasons that Chomsky acknowledges are interface conditions ("bare output conditions") and computational efficiency. We have come across the latter in §**Erreur ! Source du renvoi introuvable.**: derivation by phase is motivated by the suspicion that the computation of an entire sentence is too complex given the limitations of active memory. Sentences are thus computed in successive smaller pieces.

In Chomsky's biolinguistic approach, language is "perfect" just like other natural systems are (in reference to Galileo's "nature is perfect", see Chomsky 2004:105, also Chomsky 1995a:221, Jackendoff 1997:19). Pursuing the goal of a "perfect" FLN (faculty of language in the narrow sense, Hauser *et al.* 2002, see §**Erreur ! Source du renvoi introuvable.**), the minimalist programme therefore ambitions to reduce the FLN to just those properties that cannot be derived from computational efficiency and constraints imposed by the two interfaces. This also implies that LF and PF do not count as really linguistic (§**Erreur ! Source du renvoi introuvable.**): they are imperfect and lie outside of the FLN, which reduces to narrow syntax.

The elimination of conceptually unnecessary devices from syntax does not mean that they are simply beamed out of sight into some extra-grammatical space, though. It means that they are poured into the interfaces, i.e. into what is called LF and PF.

4 2.1.2. Minimalism pumps up PF

The minimalist transfer of activity from syntax to the interfaces loads PF much more than LF. Under this charge, the status of PF changes quite dramatically. While in the classical inverted T that generative grammar has lived with since the 60s (§**Erreur ! Source du renvoi introuvable.**) PF was more or less coextensive with phonology, i.e. the phonological computational system, it is now pumped up with a whole lot of operations and items that have got nothing to do with what phonologists call phonology. In the minimalist landscape, PF has become "phonology plus something", and as we will see below nobody really knows what either the whole or the "something" is.

Syntacticians are quick and happy to dump all kinds of things into the PF-dustbin, but do not care much whether the dustbin is suited to receive the parcel, or whether the parcel can be handled by

those who know the dustbin from the inside. That is, PF is often treated as a black box: syntacticians do not look into what what actually happens to the parcel when it is treated "at PF". The only thing that matters is that "PF" somehow manages to do things that syntacticians do not want to do in clinically pure syntax.

5 2.1.3. Dumping into the PF dustbin and hoping that it is big enough

A typical case of syntax-to-PF outsourcing is clitic placement, which is traditionally managed in syntax. Bošković (2001) and Franks & Bošković (2001) for example argue for a PF-based solution: clitics in Bulgarian, Macedonian and Serbo-Croatian are generated in several locations by the syntax, and then PF "chooses" the copy that is actually pronounced.

Elaborating on Chomsky's (1993) copy and delete where only the highest copy is pronounced and all others are deleted "at PF", Franks & Bošković (2001) propose an analysis whereby

(2) "a chain is pronounced in the head position, with lower copies deleted in PF, unless pronunciation in the head position would give rise to a PF violation. If the violation can be avoided by pronouncing a lower copy of the chain, then the lower copy is pronounced instead of the head of the chain." Franks & Bošković (2001:176)

One wonders, then, what a relevant PF violation looks like, and whether it has anything to do with phonology. The authors explain the conditioning factor as follows.

(3) "This is the well-known Tobler-Mussafia (TM) effect. It results from a phonological difference between the relevant Bg [Bulgarian] and Mac [Macedonian] clitics: in Bg these elements are strictly enclitic, whereas in Mac they are not. When no nonverbal lexical material that could support the clitics occurs in front of the clitics, the verb must precede the clitics in order to provide phonological support for them. This 'repair' strategy of pronouncing the verb to the left of the clitics occurs in Bg, but not Mac." Franks & Bošković (2001:175)

The PF violation that makes the pronunciation of the highest copy illegal thus falls into two components: 1) the fact that in the particular language at hand an item is an enclitic, and 2) the fact that in this language there is no "repair" strategy which places a word before the enclitic. The result is an enclitic without host, which leads to ungrammaticality.

Looking at this analysis, one wonders about the use of the word "phonological": Franks & Bošković say that the effect is based on a "phonological difference", but there is nothing that a phonologist would call phonological, or that any phonological theory could manage. Clitichood is certainly not anything that is defined in phonology, or that is manipulable by phonological computation, and repairs that move entire morphemes are certainly not the result of phonological computation either.

6 2.1.4. The syntacticians' phonology is not what phonologists call phonology

The preceding section has shown that there is quite some confusion around the "phonological" causes of what happens "at PF", and the example quoted is not isolated. Bošković' (2001) book is called "On the nature of the syntax-phonology interface. Cliticization ond related phenmena", but the interested phonologist will not find anything that he would call phonological or due to phonological computation in the phenomena or analyses presented.

"Deletion at PF" is also popular with pieces that are not involved in a chain, and which may be much bigger than simple clitics: the analysis of ellipsis and one specific form of this phenomenon, sluicing, may delete words, phrases or sometimes entire CPs (e.g. Merchant 2001, Fox & Lasnik 2003). The standard analysis today is that ellipsis reduces to non-pronunciation at PF (Gengel forth).

It was mentioned that the triggering conditions of Franks & Bošković' (2001) analysis are called phonological but have got nothing to do with phonology. The same is true for the action that is taken at

PF: phonologists may perhaps delete features or segments, but certainly not words, VPs or CPs. No phonological theory is suited for the manipulation of this kind of object, which phonologists look at like an ant looks at a jumbo jet.

Under the minimalist pressure, PF-sinking of syntactic problems has become a natural and unquestioned move over the past decade. Interestingly, though, this trend is by and large confined in syntactic quarters: syntacticians talk about syntactic phenomena and may use the word "phonology" in relation with PF, but nothing that phonologists would call phonology is involved. In other words, the PF world that takes on more and more importance among syntacticians is quite waterproof with respect to phonology as conceived of by phonologists.[1]

The following section documents the ambient confusion around the new minimalism-born PF.

7 2.2. Confusion and mistiness: what does PF mean, what does it contain?

When I first heard of PF as a student, I thought that "Phonological Form" was meant, and was quite surprised when I learned later on that PF actually stands for "Phonetic Form". This is at least the way Chomsky (1995a:2) spells out PF. The confusion lingers on in my mind, and I still need to make sure I think of phonetics when hearing and talking about PF. Maybe I am not the only one who gets confused by the meaning of the P. Chomsky for example does not make much difference between phonology and phonetics. In the quote under (4) for instance, phonology is missing, and phonetics appears instead.

(4) "We may think of the language, then, as a finitely specified generative procedure (function) that enumerates an infinite set of SDs [structural descriptions]. Each SD, in turn, specifies the full array of phonetic, semantic, and syntactic properties of a particular linguistic expression." Chomsky (1995a:14f)

In the quote below, one senses that Chomsky actually means the parametric variation of phonology, rather than of phonetics: parameter settings are a typical property of the former, but not really of the latter area.

(5) "At the PF level, properties of the language can be readily observed and variation is possible within the fixed repertoire of phonetic properties and the invariant principles of universal phonetics." Chomsky (1995a:27)

Finally, the confusion is overt in the following quote: if there is any distinction between phonology and phonetics at all, syllable structure is a property of the former.

(6) "The PF representation π is a string of phonetic primes with syllabic and intonational structure indicated, derived by a computation from σ." Chomsky (1995a:35)

The three quotes mentioned are all from the first chapter of The Minimalist Program, i.e. actually from a 1993 paper that Chomsky co-authored with Howard Lasnik. In later chapters of the book, Chomsky does make a difference between phonology and phonetics, as shown by the quote below.

(7) "Similarly, the phonological component contains no rules to express special cases of general properties of universal phonetics or of phonetic representations." Chomsky (1995a:152)

---

[1] As far as I can see, the only phenomenon that sometimes comes up in PF-based analyses of syntacticians, and which could qualify for phonological status, is intonation (sentence stress). This is not surprising since intonation is the single one phonological phenomenon that syntax is classically concerned with. It is not so sure, however, that intonation is the result of any phonological computation at all (more on this in §XXX).

He now talks about a phonological component, and this component seems to be different in kind from phonetics. One may thus incline to conclude that phonetics is not part of PF.

At the outset of chapter four, one reads the following statement, which is puzzling in more than one respect.

(8) "Notice that I am sweeping under the rug questions of considerable significance, notably, questions about what in the earlier Extended Standard Theory (EST) framework were called 'surface effects' on interpretation. These are manifold, involving topic-focus and theme-rheme structures, figure-ground properties, effects of adjacency and linearity, and many others. Prima facie, they seem to involve some additional level or levels internal to the phonological component, postmorphology but prephonetic, accessed at the interface along with PF (Phonetic Form) and LF (Logical Form)." Chomsky (1995a:220)

It seems to be safe to consider that PF is what Chomsky calls a "level", and he again is explicit on the fact that the P means "phonetic". One also learns that there is a "phonological component" which is ordered after morphology but before phonetics. This component, then, appears to be a piece of PF. So far, so good – but how could this sub-division of PF then itself host additional internal levels? In a note to the last sentence under (8), Chomsky states that the phenomena at hand (which are supposed to be accommodated by the "additional level or levels internal to the phonological component") cannot be directly done at PF, but could perhaps be handled by "elements formed in the course of the mapping of syntactic objects to a PF represetnation."

(9) "The PF level itself is too primitive and unstructured to serve this purpose, but elements formed in the course of the mapping of syntactic objects to a PF representation might qualify." Chomsky (1995a:379)

It is hard to imagine what kind of item could be at the same time external to PF but internal to the "phonological component", which itself is a piece of PF.

In other writings of Chomsky's, one senses that PF and the "phonological component" are the same thing.

(10) "There are some reasons to suspect that a substantial core of head raising processes […] may fall within the phonological component." Chomsky (2001:37)

"'Stylistic' operations might fall within the phonological component." (Chomsky 2000:144)

This selection of self-contradictory, puzzling and sometimes cryptic statements about PF could probably be pursued and extended to other authors. The purpose of their review is not to point out the inconsistency, but the fact that this inconsistency is not innocent: people talk about PF in these terms because PF is a misty and ill-defined intermundia. And people may get confused as I do about the P and the relationship of the strange PF-animal with what phonologists call phonology (and what Chomsky probably refers to as the phonological component).

2.3. Properties of PF: what kind of animals live in the intermundia?

2.3.1. Internal structure of PF

One conclusion of the preceding discussion is that if anything, PF is a heterogeneous space. This is what is meant when the literature (namely the one related to Distributed Morphology) talks about "the PF branch", rather than just about PF. This opens up an entire workspace, indicating that there are many things going on "at PF", not just (the phonologists') phonology, and not just phonetics. It was mentioned in §**Erreur ! Source du renvoi introuvable.** that Distributed Morphology locates operations such as fission, fusion and impoverishment (which are unknown in syntax) in PF, and that

PF movement obeys locality conditions that are distinct from the ones that control regular movement in (narrow) syntax (§**Erreur ! Source du renvoi introuvable.**).

Therefore Pak (2008:26) says that "[t]he PF branch […] is thus viewed as a highly articulated derivational component, which yields a number of intermediate structural representations before the final surface form is reached". This points to the manifoldness of the PF landscape: a lot of things are done, and a lot of different things are done. Like Chomsky and much of the literature, Pak does not make explicit whether she means a computational system in the sense of a Fodorian module when she talks about a "derivational component". The fact that there is a long way to go from the upper limit of PF that intersects with (narrow) syntax and its output is illustrated by the interesting term "surface PF" that Pak introduces in a footnote.

(11) "The term *PF* is used to refer both to the derivation along the branch and to the surface form produced at the end of the branch; for clarity, I will use the term *surface PF* when this latter meaning is intended." Pak (2008:26, note 1, emphasis in original)

2.3.2. What happens "at PF"

Let us now look at the kind of operations that syntacticians suppose are managed "at PF", i.e. those that do not concern regular phonology and, if included in PF, regular phonetics. The survey below of course does not ambition to be exhaustive: new things are constantly added, and as a phonologist my visibility is rather poor. This being said, the most typical thing that syntacticians want PF to do is certainly deletion (of copies, or involving ellipsis and sluicing). In the previous sections we have also come across other candidates for a management at PF: topic-focus, theme-rheme, figure-ground, linearity, head movement and stylistic operations.

Richards (2004) adds an interesting hypothesis regarding the limitations of PF action.

(12) "The assumption that PF cannot drive syntactic operations ties in with the more general thesis, presented in section 2.5.1.3, that PF can only operate with the structures that the syntax provides to it: in particular, PF cannot build extra structure, that is, create new positions not licensed during the syntactic part of the derivation." Richards (2004:12, note 2)

Hence PF movement for example can move phonological terminals around, but only to positions that are inherited by syntax.

As was mentioned, Distributed Morphology is also concerned with the PF branch: this is where morphology-specific operations such as allomorphy, fission, fusion and impoverishment take place. Summarising Embick & Noyer (2001, 2007), Pak (2008) provides a list of operations that are assumed to take place at PF in order to derive "the surface form at the end of the branch [which] clearly has very different properties from the syntactic structure" (Pak 2008:26).

(13) "PF operations (unordered list):
a. *Structural readjustments*, a limited set of movement, rebracketing, and deletion/insertion operations whose surface effects are often recognized as 'syntax-morphology mismatches'
b. *Vocabulary insertion*, which adds phonological content to function morphemes
c. *Linearization operations*, which establish linear order between/across structures"
Pak (2008:26, emphasis in original)

Under (13)a, one senses that DM is aware of the fact that additional movement, deletion, insertion etc. is a source of dramatic overgeneration (see §**Erreur ! Source du renvoi introuvable.**), and therefore prophylactically talks about "a limited set" of operations. As far as I can see, though, nobody knows in which way exactly the operations "at PF" are limited, let alone the reason for such limitations.

Linearisation (and its interleaving with phonology) is further discussed in §16 below. Its occurrence at (the beginning of) PF is largely consensual also outside of DM quarters (except in Kayne's 1994 system).

11 2.4. Trying to make sense of PF from the modular point of view

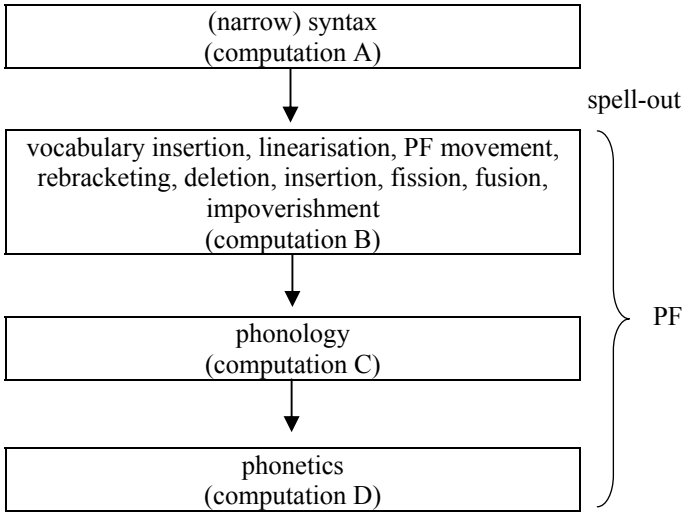
12 2.4.1. PF is a cover term for a number of serially ordered computational systems

Let us now try to make sense of PF and its internal structure from the modular point of view. This amounts to identifying individual computational systems that qualify for modular status, i.e. are domain specific. We have seen that various designations are used in the literature in order to refer to PF as a whole, or to its sub-systems: level, (phonological) component, module (Pak 2008:6, 29 calls the entire PF branch the "PF module"). Authors typically do not specify, though, whether they talk about computational systems, i.e. input-output systems, or just informally about a functional entity.

Let us start with two points that are undisputed: 1) (narrow) syntax is a computational system with modular status of its own, and 2) the input to PF is created by the spell-out of (narrow) syntactic structure. Another non-negotiable item is a phononlogy, i.e. a computational system with modular status that computes what *phonologists* call phonology. This system is domain specific because it works with its own proprietary vocabulary (labiality, occulsion etc.) which is distinct from other linguistic vocabulary. On traditional (generative) assumptions, this list is rounded off by a distinct computational system that is in charge of phonetics, and which is ordered after phonology. There is a fair amount of debate, especially in the OT literature, regarding the question whether phonetics and phonology are merged into one single computational system, i.e. one single constraint hierarchy (Kingston 2007 for example provides an informed overview). This is an expression of OT's scrambling trope (§§**Erreur ! Source du renvoi introuvable.**,**Erreur ! Source du renvoi introuvable.**). For the sake of exposition, below a distinct computational system appears for phonetics, but nothing hinges on that.

We are thus left with a derivational sequence as under (14) below.

(14) what PF is made of



PF thus consists of three independent and serially ordered computational systems. So far this view would be consistent with modularity.

13 2.4.2. The minimalism-born intermundia violates domain specificity

It was already pointed out in §**Erreur ! Source du renvoi introuvable.**, however, that there is trouble with computation B, i.e. the misty intermundia that was created by minimalism where all those syntactic-looking things are managed that are rejected from clean narrow syntax. Computation B is

past vocabulary insertion; this means that phonological material is present and, according to PF movement, forms the terminal elements of the morpho-syntactic tree, which is also still available.

As was shown in §**Erreur ! Source du renvoi introuvable.**, however, this cannot be reconciled with domain specificity: computation B would have to access the morpho-syntactic labels of the tree, the tree geometrics and phonological vocabulary at the same time. Also, the tree labels would be the projection of nothing: on standard assumptions hierarchical structure is a projection of terminal elements. In a PF movement tree, however, phonological terminals would cohabitate with morpho-syntactic structure and labels: this does not make any sense. Computation B is thus a modular alien.

Another issue is linearisation: one wonders what it means to linearise upon spell-out from (narrow) syntax, but to leave the tree in place, and to allow for (PF) movement along its structure. A linearised string does not look like a tree, does it? Or does linearisation in fact take place at the end of computation B when PF movement has applied and a linear input to computation C, the phonologists' phonology, needs to be created? Linearisation is examined at greater length in §16 below. One thing is for sure, though: what phonologists call phonology requires a linear string as an input.

Finally, one could think of the whole of PF, i.e. computations B, C and D, as a single macro-module which accommodates a number of sub-modules. This is visibly what Pak (2008:6, 29) means when she calls the entire PF branch a "PF module". Nested computation (or computational systems), however, should be excluded in a modular environment: it violates (informational) encapsulation (see §**Erreur ! Source du renvoi introuvable.**, also §**Erreur ! Source du renvoi introuvable.**).

14 2.4.3. Mixing phonology and the intermundia

Minimalism-created intermundia, i.e. computation B, is already violating domain specificity by itself. There are proposals that go even further by mixing computation B with computation C, the real phonology. It was already mentioned in §**Erreur ! Source du renvoi introuvable.** that DM, through PF movement, promotes a radical form of direct syntax. It was also mentioned that Pak (2008:42ff, 60ff) sets out to do away with the Prosodic Hierarchy: she correctly argues that prosodic constituency is superfluous if phonology has direct access to morpho-syntactic information.

In fact the core of her proposal is a much stronger violation of modularity: she argues that the action of computation C (phonology) is serially interleaved with computation B (the intermundia).

(15) "I propose that phonological rules apply at *various points* in the PF derivation. Specifically, phonological rules are interleaved with different kinds of *linearization procedures*, which apply in PF in order to convert abstract hierarchical structures into fully linearized strings." Pak (2008:6, emphasis in original)

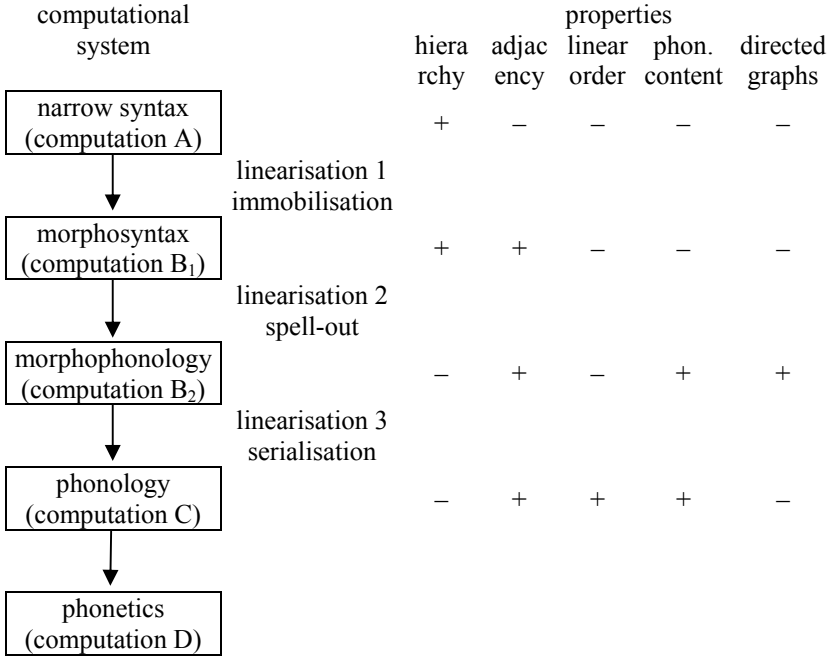
"The hypothesis pursued in this dissertation is that phonological rules may also use Concatenation statements . as well as other kinds of linearization statements . as their domains. In other words, *phonological rules are interleaved with linearization operations*." Pak (2008:28, emphasis in original)

DM thus turns out to be a strong modularity offender (see §**Erreur ! Source du renvoi introuvable.**). Linearisation is further discussed in §16 below.

15 2.4.4. The internal structure of the intermundia: two distinct derivational stages, morphosyntax and morphophonology (Idsardi & Raimy forth)

Idsardi & Raimy (forth) (followed by Samuels forth) argue that what appears as the intermundia under (14) (computation B) in fact falls into two distinct and serially ordered systems as under (16).

(16) internal structure of computation B according to Idsardi & Raimy (forth)



Their proposal is linearisation-driven: the two additional stages correspond to the idea that linearisation is a three-step operation: it falls into what they call immobilisation, spell-out and serialisation.

Idsardi & Raimy consistently use the word "module" in order to refer to their systems, but they do not address the question which kind of proprietary vocabulary is used by each one of their computational systems. This is understandable since for them the need to further divide computation B stems from the idea that linearisation is progressive: each step in linearisation, then, must be a computational system by itself.

It may be noted that the minimal merit of Idsardi & Raimy's system is that they individuate the "real" phonology – i.e. what phonologists call phonology – from the misty intermundia "PF": their phonology (computation C under (16)) is distinct from computation B (or B_1, B_2), and they are explicit on the fact that we are facing distinct modules. This may not be unrelated to the fact that (unlike authors from syntactic or DM quarters) they are phonologists and look at the misty "PF branch" from the phonologist's perspective.

At variance with earlier writings of Raimy on directed graphs (Raimy 2000a,b, 2003, to be discussed in §21), Idsardi & Raimy (forth) call "phonology" (computation C under (16)) what appeared as phonetics before, i.e. the stage where directed graphs with loops (non-asymmetic representations as Raimy calls them) have been linearised to a strictly linear order of items without loops. Hence what Raimy called phonology (representations with loops) before now appears as morphophonology, and earlier phonetics (representations without loops) is now phonology.

Finally, for some reason Idsardi & Raimy restrict the label "PF" to "narrow" phonology, i.e. computation C under (16).

(17) "For the purposes of this paper, we will equate the phonological representation with PF and assume that there are further complex transformations in the phonetic modules which result in a representation interpretable at the SM [sensory-motor] interface." Idsardi & Raimy (forth:2)

This adds confusion to the overall landscape where PF is typically held to encompass everything after narrow syntax (the "PF branch").

Idsardi & Raimy's system is born from Raimy's (2000a,b) theory of (morpho-)phonology that is based on the analysis of reduplication (and infixation), and as was mentioned it is linearisation-driven. Their backdrop is thus significantly different from the one of Distributed Morphology, where the

motivation for inquiring on the internal structure of the PF-intermundia is the accommodation of operations such as allomorphy, fission, fusion and PF movement. It is therefore not really surprising that the results are quite different. For example, PF movement cannot be accommodated by Idsardi & Raimy's system: there is no single computational system under (16) that allows phonological content to cohabitate with hierarchical structure. The shift from computation B_1 (morphosyntax) to computation B_2 (morphophonology) eliminates hierarchical structure and introduces phonological content. In DM terminology, this is where vocabulary insertion takes place.

16 ## 2.5. Linearisation

17 ### 2.5.1. Introduction: no business of phonology

A question that was not addressed at all in Part I but which is constantly mentioned on the preceding pages is the linear order of the phonological string, as opposed to the non-linear character of syntax. The reason why linearisation was not discussed in Part I is simply that it is no business of phonologists – but as was mentioned before, theories of the interface between morpho-syntax and phonology are made by phonologists (with the half-exception of Distributed Morphology). What everybody agrees upon is the fact that the input to phonological computation is a linear string.[2] Or, in other words, phonological computation does not create linearity. This is why phonologists never talk about linearity and linearisation: they work on linear strings and thus take linearity for granted.

Today everybody agrees that some mechanism is required which transforms the hierarchical morpho-syntactic structure (the tree) into a linear string. The question is where exactly this mechanism is accommodated, how it works and eventually into how many sub-components it falls. Roughly speaking, Kayne's (1994) LCA (Linear Correspondence Axiom) places linearisation in narrow syntax, while all other proposals (which typically build on the LCA) locate this operation in PF (where "in PF" opens a rather large array of possibilities, as we have seen in §8).

18 ### 2.5.2. In minimalist times: no business of syntax either

In syntax, linearity was long taken to follow from syntactic constituent structure: word order is a function of constituent order. This was the case until (and including) GB: phrase structure rules were parameterised and responsible for language-specific variation in word order. In a language where prepositions precede nouns, the rule was PP → P, NP, while in a language where the opposite order is observed, the rule PP → NP, P was instrumental. Hence the difference between right-branching languages such as English where heads precede their complements and left-branching languages like Japanese where the reverse order is found is managed by a parameter setting, the *head parameter*.

In the minimalist perspective where trees are constructed by Merge and hence phrase structure rules are eliminated, a different means for deriving linear order needs to be found. Estranging syntax from linear order ties in with the observation that syntactic generalisations are about hierarchical organisation (command and dominance relations), not about linear order. Therefore Chomsky (1995a:334) concludes that syntax has got nothing to do with linearity, not any more than LF: linearity is only relevant for phonology. That is, it is imposed upon the linguistic system by external conditions of language use: linearity is the result of the constraints that follow from the transmission of human language whereby speech unfolds in a temporal sequence.

19 ### 2.5.3. Both syntax-internal and syntax-external linearisation is minimalism-compatible

Given that linear order is imposed upon language by a non-linguistic constraint, Chomsky (1995a:335) welcomes Kayne's (1994) antisymmetry in a minimalist perspective. Kayne's idea is to derive linear order *in the syntax* by leftward movement: at the end of all movement operations, the highest item is the leftmost in the linear order, and so on. On Kayne's take, this is true for all languages: SVO is the

[2] But even this notion needs to be further defined: we will see below (and have already seen in §15) that Raimy's (2003) and Idsardi & Raimy's (forth) system allows for different degrees of linearisation of the string.

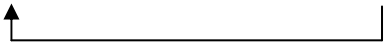
universal order underlyingly, and languages like Japanese with an overt SOV order are derived by leftward movement.

Incorporating the mechanism that creates linear order into the syntax may at first appear contradictory with the minimalist insight that syntax and linear order are independent. Chomsky (1995a:335), however, argues that syntax-internal linearisation is a prototypical implementation of minimalist thinking whose headstone is to reduce syntactic representations and computation to "bare output conditions". Hence linearity, imposed from the outside by language use, marshals syntax: movement needs to be carried out in order to satisfy its requirements.

Chomksy threrefore writes that Kayne's perspective is "very much in the spirit of the Minimalist Program and consistent with the speculation that the essential character of C_{HL} [the computational system of human language] is independent of the sensorimotor interface" (Chomsky 1995a:335). A few pages later, however, he places the LCA outside of (narrow) syntax, in PF: "we take the LCA to be a principle of the phonological component that applies to the output of Morphology" (Chomsky 1995a:340, note that Chomsky operates with the traditional split between (narrow) syntax and morphology).

Another line of attack that follows Kayne's track is Fox and Pesetsky's (2004) analysis whereby successive cyclic movement (in narrow syntax) is derived from linearity requirements. That is, just like in Kayne's system and in line with the minimalist philosophy, syntax-internal movement is motivated by a necessity of a syntax-external cognitive system. The key idea of Fox & Pesetsky is that there is a mechanism that controls the result of linearisation at every phase and compares it with the linear order achieved at previous phases. In case there is a mismatch between former and current linear order, the derivation crashes. This is what Fox & Pesetsky call order preservation: linear order must be the same at every spell-out of every phase. For the sake of illustration, consider the example under (18) below (from Fox & Pesetsky 2004:5).

(18) [To whom will he [__say [$_{CP}$ __ that Mary [$_{VP}$ __ gave the book __]]]]?



Did the movement skip Spec,VP and went directly to Spec,CP as indicated, and if VP is a phase head, Fox & Pesetsky argue, the derivation will crash at PF since the result of linearisation in the VP phase (where the displaced item remains in situ) is different from the linearised string after the CP phase (where nothing will be left in situ). By contrast, if movement goes through Spec,VP, the VP will be linearised with the displaced item in Spec,VP at the lower phase as well as at all other phases: order preservation is satisfied. Therefore, Fox & Pesetsky (2004:8) argue, "[a]n architecture of this sort will in general force successive cyclicity when movement crosses a Spell-out domain boundary."

In sum, then, minimalist principles seem to be able to be used in order to locate linearisation in either (narrow) syntax or PF. Or rather, the competition between both options cannot be refereed on minimalist grounds.

2.5.4. Everybody but Kayne does linearisation "at PF"

Work in Kayne's original perspective set aside, it appears that the field has by and large followed Chomsky's indication that linearisation occurs "at PF".

It seems to me, though, that the question is treated without much argument in the literature: people acknowledge the existence of Kayne's LCA, but then do linearisation "at PF" without saying why it should or could not be done by movement in (narrow) syntax.[3] One argument is provided by Richards (2004): Kayne's LCA is based on c-command, i.e. on an asymmetric relationship between the items that are to be linearised.

[3] E.g. Bobaljik (2002) and Embick & Noyer (2001, 2007), discussed below, who do linearisation at PF "by hypothesis" or "assume" that it occurs at PF: "[a]ssuming that linear order is not included in the syntactic representation, PF operations, because they are responsible for creating the interface level that mediates betweeen syntax and the articulatory/perceptual systems, must at the very minimum be responsible for linearizing hierarchical structures" (Embick & Noyer 2007:293). Hence Kayne's syntax-internal option is dismissed without argument or discussion.

(19) "Linear Correspondence Axiom
If α asymmetrically c-commands β then (the terminals dominated by) α precede(s) (the terminals dominated by) β" version of the LCA given by Richards (2004:11)

What about cases of mutual c-command, then? Richards argues that the minimalist Bare Phrase Structure, which eliminates trivial, i.e. vacuous or unary-branching projections, regularly produces this kind of structure, for example in [John [ate it]] where *ate* and *it* are sisters and c-command each other. Kayne's LCA fails when trying to linearise this kind of structure. Richards therefore concludes that

(20) "the LCA cannot be a constraint on phrase-markers themselves, i.e. a property of Narrow Syntax, but must be a *linearization strategy* operative only after Spell-Out in the mapping of syntactic hierarchy onto phonotemporal order (cf. MP [Chomsky 1995a]: 340: 'We take the LCA to be a principle of the phonological component that applies to the output of Morphology'). This recasting of the LCA as a PF-mapping strategy (cf. the operation *Linearize* of Nunes 1999) conforms to the general principle that the 'horizontal' dimension of time and sequential ordering is relevant only in the phonological component, so that the syntactic component of C_{HL} [the computational system of human language] deals only in the 'vertical' dimension of hierarchical relations." Richards (2004:12, emphasis in original)

Richards (2004:23) then goes on to rehabilitate the old head parameter of GB in order to make it a parameter of the PF-located LCA.

Other syntax-based linearisation strategies also typically implement the head parameter in one way or another. Bobaljik (2002:216) for example says that "it should be clear that I am espousing a more or less traditional view of headedness parameters, for instance, that the German V' is head-final while the English V' is head-initial; this is the information encoded in the precedence rules."

Bobaljik's precedence rules are "a procedure that maps each node to an ordered pair: [X → Y] or [Y → X] (where the arrow is to be read as 'precedes')" (Bobaljik 2002:213). He then shows how a German SOV and an English SVO order can be derived from the same syntactic structure when reverse precedence rules are applied to certain nodes (the head parameter): German will have [$_{I'}$ VP → Infl] and [$_{VP}$ DP_2 → V], while English accommodates [$_{I'}$ Infl → VP] and [$_{VP}$ V → DP_2]. On Bobaljik's count, thus, each language possesses a language-specific set of precedence rules (that look much like good old phrase structure rules, except that there is only one item to the right of the arrow).

In Bobaljik's system, this all happens upon spell-out from (narrow) syntax to PF, whereby the spell-out process falls into four distinct operations (which Bobaljik 2002:214, note 16 calls "components"): "a) assignment of precedence conditions to syntactic nodes [the precedence rules mentioned], b) chain reduction (= trace or copy deletion), c) conversion to linear string of X°s, d) vocabulary insertion" (Bobaljik 2002:214).

The same picture is found in DM quarters. Embick & Noyer (2001, 2007) hold that linearisation is concomitant with vocabulary insertion.

(21) "By hypothesis, linear ordering is not a property of syntactic representations but is imposed at PF in virtue of the requirement that speech be instantiated in time (see Sproat 1985).It is therefore natural to assume that linear ordering is imposed on a phrase marker at the point in the derivation when phonological information is inserted, that is, at Vocabulary Insertion.

(8) *The Late Linearization Hypothesis*
The elements of a phrase marker are linearized at Vocabulary Insertion."
Embick & Noyer (2001:562, emphasis in original)

Embick & Noyer work with a notational variant of Bobaljik's precedence rules which they call "Lin" (or the reverse: Bobaljik's precedence rules are a notational variant of Lin):

(22) "linear order is a binary operator – represented by '*' – imposed by an operation Lin:

Lin [X Y] → (X*Y) or (Y*X)

This relationship is one of immediate (left-)adjacency; subsequent steps concatenate terminal nodes. Other types of conditions might be imposed by distinct linearization operations." (Embick & Noyer 2007:294).

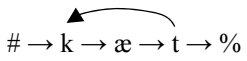
Embick & Noyer thus cut linearisation into (at least) two distinct operations, Lin and Concatenate (see Pak 2008:26ff for a summary). The linearisation procedures reviewed share the idea that linearisation in fact falls into a number of different operations that are serially ordered: (at least) two with Embick & Noyer (2001, 2007), three with Idsardi & Raimy (forth) (immobilisation, spell-out and serialisation, §15), four with Bobaljik (2002) (precedence conditions, chain reduction, linearisation of X°s, vocabulary insertion).

21 2.5.5. Linearisation in phonology in order to derive phonetics (Raimy)?

Raimy (2000a,b, 2003) proposes a view on linearisation that is quite different from what we have seen thus far: he looks at the issue through a phonological, rather than through a syntactic lense. Raimy's basic idea is that precedence relationships must be explicitly noted in all phonological representations. Linear order is implicit in the standard graphic notation: "kæt" (representing *cat*, the word used for the sake of illustration by Raimy) for example reads "k before æ before t", and the same is true for x-slots in an autosegmental representation. Raimy proposes to make this explicit and to note "# → k → æ → t → %" instead (where # indicates the beginning of the linear string, and % its end). He calls the result a *directed graph*.

What is this good for if the graphic material added can be deduced from "kæt" and is therefore fully redundant? Raimy argues that there are cases where arrows actually represent *morphological* information: morphological computation is one that adds precedence relationships, i.e. arrows, to lexical material. The operation is trivial when prefixes and suffixes are concatenated, but involves the addition of "loops" within a root in case of reduplication and infixation. A process whereby "kæt" is entirely reduplicated, producing "kætkæt", thus boils down to the addition of an arrow that originates in the *t* and points to the *k*, as shown under (23) below.

(23) linearisation of loop-including representations

a. 

b. # → k → æ → t → k → æ → t → %

The loop under (23)a thus represents the morphological operation of reduplication. The resulting structure (which Raimy calls non-asymmetrical) is then spelled out into a strictly linear string (an asymmetrical structure) as under (23)b: "[l]inearization eliminates non-asymmetrical precedence structures through repetition of segments which preserves the overall organization of a precedence structure while not causing problems of interpretation for the phonetics module" (Raimy 2003:133). The "problems of interpretation" that Raimy talks about are the loops: he assumes that phonology can live with loop-containing representations, but phonetics cannot – here a fully linearised string is needed.

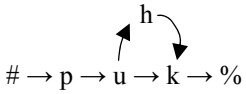
In sum, then, phonological representations are only partially linear (loops are non-linear elements); exactly analogous to the relationship between (narrow) syntax and PF, bare output conditions – the requirement of fully linearised strings at the phonetic level – provoke a spell-out.

(24) "Analogous to the syntactic LCA (Kayne 1994), phonology contains a *linearization* process which ensures that representations have *asymmetrical* precedence structures and are thus interpretable at the phonetics interface. Linearization in phonology repeats segments within a 'loop' in order to eliminate [non-]asymmetrical precedence relations." Raimy (2003:132, emphasis in original; obviously Raimy means to say *non*-asymmetrical in the last line, the absence of the *non*- is an error)

The operation that adds arrows to strings needs to "know" where to place the origin and the endpoint (anchor points). Adopting Raimy's directed graphs, Samuels (2009a:147ff) has studied the cross-linguistic variation of infixation in order to determine what kind of information the arrow-placing algorithm needs to be able to interpret. The list of anchor points that infixes look at in order to determine their landing site falls into two categories: edge-oriented and prominence-oriented. For the left edge for example, documented situations are "after the first consonant (or consonant cluster)", "after the first vowel", "after the first syllable" and "after the second consonant". Prominence-based attractors are stressed vowels, stressed syllables or stressed feet.

An example discussed by Samuels (2009a:179ff) is Tzeltal (Maya) intransitivisation whereby a verb is made intransitive by infixing an *-h-* after the first vowel: compare *puk* "to divide among" with *pu-h-k* "to spread the word", and *kuč* "to carry" with *ku-h-č* "to endure". The representation of this operation with directed graphs is as under (25) below.

(25)



As was mentioned in §15, Raimy has changed the labels of his system since 2003: in Idsardi & Raimy (forth) (also Samuels 2009b), he calls loop-containing representations morphophonology (instead of phonology), and representations from which loops have been eliminated phonology (instead of phonetics). This leaves the workings of his system unaffected, but does away with the idea that phonology can handle non-linear structure. That is, like all other approaches, the relabelled version of directed graphs holds that the input to "true" phonological computation, i.e. the one that phonologists call phonology and from which concatenation is absent, is strictly linear.

2.6. Conclusion: a minimalism-born monster

PF is an ex nihilo creation of minimalism: until the mid-90s PF was coextensive with phonology, but since then adds an ill-defined intermudia between (narrow) syntax and "true" phonology, i.e. what phonologists call phonology. Today one is forced to talk about "true" phonology or "what phonologists call phonology" because there is a great amount of confusion around the word *phonology*. Syntacticians want a whole lot of things to be "phonological" in order to get rid of them: clean syntax, dirty phonology. Most of the things that are called "phonological" in the syntactic literature where phenomena are dumped into the PF dustbin, though, are nothing that phonologists are familiar with or could handle.

While the upper limit of PF is clearly defined (PF is created by spell-out from (narrow) syntax), its lower limit and eventual internal structure are murky. It is consensual that phonological material is present as soon as PF structure is created: vocabulary insertion occurs at spell-out. What happens then, though, is left misty in the literature: a lot of operations that have got nothing to do with phonology, and then, at some point, "true" phonology. Idsardi & Raimy (forth) set aside (they are phonologists), I could not find a single explicit mention that what phonologists call phonology is a distinct computational system whose input must be a linear string that does not contain any morpho-syntactic arboreal structure anymore. When talking to syntacticians and asking the question where "true" phonology is in the wafting PF body, if any, one often hears that there must be a clearly distinct phonological system somewhere further down the road when all the pseudo-syntactic labour is done – but syntacticians are then unable to name a written source where this is made explicit.

When one tries to make sense of PF in terms of domain-specific computational systems, a separate (and "true") phonology (plus eventually a phonetic system) can be identified, but an alien

intermundia is left between spell-out and phonology (computation B under (14)). This intermundia cannot be a module because it mixes morpho-syntactic and phonological vocabulary. It also violates another fundamental principle of hierarchical structure: the nodes of the tree are projections of nothing, in any case not of the terminals, which are phonological.

Finally, linearisation is also unclear: it seems to be consensual (Kayne 1994 set aside) that it occurs upon spell-out and vocabulary insertion, but the mechanisms that are available on the market are based on speculation, rather than on data and/or analysis. Also, they reproduce the traditional head parameter in one way or another: instead of being linearised by phrase structure rules like in the times of GB, sisters in the tree are now linearised by linearisation statements (called precedence rules etc.) in PF.

The strangest animal in the PF zoo (which everybody seems to want to restrict, but nobody says what exactly may and what may not be done at PF) is certainly PF movement. One issue is linearisation, which is supposed to occur at spell-out, that is before PF movement. But what does it mean to "linearise" when the output of "linearisation" is a not a linear, but an arboreal structure along which items continue to be displaced? And how is a truly linear string created at the end of PF movement when "true" phonology is fed? After spell-out from (narrow) syntax, is there a second spell-out mechanism?

All these questions, and foremost the massive violation of modularity by the intermundia computation B, raise the question what the whole PF business really buys us: a clean syntax, but a lot of trouble "at PF". Is it really worthwhile to move undesirable phenomena out of sight of syntacticians (and of phonologists) into an ill-defined buffer where established principles of linguistic analysis do not hold? Dumping them into PF is not analysing or solving them, and being allowed to analyse them with all kinds of ad hoc mechanisms that are unheard of elsewhere may turn out to be a Pyrrhic victory rather than anything else.

References

Bobaljik, Jonathan 2002. A-Chains at the PF-Interface: Copies and 'Covert' Movement. Natural Language & Linguistic Theory 20: 197-267.

Bošković, Željko 2001. On the nature of the syntax-phonology interface. Cliticization ond related phenmena. Amsterdam: Elsevier.

Chomsky, Noam 1993. A minimalist program for linguistic theory. The view from Building 20. Essays in Linguistics in Honor of Sylvain Bromberger, edited by Kenneth Hale & Samuel Keyser, 1-52. Cambridge, Mass.: MIT Press.

Chomsky, Noam 1995. The Minimalist Program. Cambridge, MA: MIT Press.

Chomsky, Noam 2004. Beyond explanatory adequacy. Structures and Beyond. The cartography of syntactic structures, Volume 3, edited by Adriana Belletti, 104-131. Oxford: Oxford University Press.

Fox, Danny & Howard Lasnik 2003. Successive-Cyclic Movement and Island Repair: The Difference between Sluicing and VP-Ellipsis. Linguistic Inquiry 34: 143-154.

Franks, Steven & Željko Bošković 2001. An argument for multiple spell-out. Linguistic Inquiry 32: 174-183.

Gengel, Kirsten forth. Phases and Ellipsis. Linguistic Analysis 35.

Kayne, Richard S. 1994. The Antisymmetry of Syntax. Cambridge, Mass.: MIT Press.

Kingston, John 2007. The phonetics-phonology interface. The Cambridge Handbook of Phonology, edited by Paul De Lacy, 435-456. Cambridge: CUP.

Merchant, Jason 2001. The Syntax of Silence: Sluicing, islands, and identity in ellipsis. Ph.D dissertation, University of California at Santa Cruz.

Nunes, Jairo 1999. Linearization of Chains and Phonetic Realization of Chain Links. Working Minimalism, edited by Samuel Epstein & Norbert Hornstein, 217-249. Cambridge, MA: MIT Press.

Raimy, Eric 2000. The phonology and morphology of reduplication. Berlin: Mouton de

Gruyter.
Raimy, Eric 2000. Remarks on backcopying. Linguistic Inquiry 31: 541-552.
Raimy, Eric 2003. Asymmetry and linearization in phonology. Asymmetry in Grammar, Vol.2, edited by Anna Maria Di Sciullo, 129-146. Amsterdam: Benjamins.
Samuels, Bridget 2009. The structure of phonological theory. Ph.D dissertation, Harvard University.
Samuels, Bridget 2009. What can syntax do for phonology? A case study in Basque. Paper presented at the Yale Department of Linguistics, New Haven 4 December.
Sproat, Richard 1985. On deriving the lexicon. Ph.D dissertation, MIT.