COMPOSITIONALITY: THE VERY IDEA

MARCUS KRACHT

A . Compositionality is often considered a fundamental principle of semantics for natural language. Yet only fairly recently has there been something of a theory of compositionality which allows to prove actual results. Basically, the lack of progress has been due to an improper understanding of what syntactic and semantic structures actually are. Many linguistic theories in one way or another confuse them by importing semantic notions into syntax or—conversely—adding syntactic detail to semantic structures. In this paper I shall outline a theory of semantic and syntactic structures and show how it avoids the problems that beset the previous theories. A particular benefit of this approach is that it allows to show results on sentential structure.
compositionality, syntax, semantics

1. I

The present paper defends a particular view on language. It is that compositionality can serve as a tool to gain insight into sentential structure. Rather than having to rely on purely syntactic criteria of constituency it allows to adduce genuine semantic facts to decide between different constituent structures. The present paper is mainly conceptual; the technical apparatus is developed in [Kracht, 2007]. Here I will try to motivate the proposal on purely conceptual grounds. Though the final verdict will as much depend on technical detail, it is important to see how much intuitive appeal can be given to it in the first place.

Compositionality is undoubtedly a very important concept in linguistic theory. Unfortunately, it is often just a slogan and not an object of serious research. The literature on compositionality is rather small and for a long time there was actually no theory of compositionality worth the name. Recently, however, the situation has changed. Compositionality has enjoyed something of a renaissance in theoretical linguistics, especially through [Hodges, 2001] and [Kracht, 2003b], which provide the nucleus of a theory of compositionality.

Compositionality is very popular in linguistics. Linguists claim to prefer a compositional analysis over a noncompositional one. Yet, I suspect that

Date: August 30, 2009.

very few people have a clear idea of what compositionality actually buys them, if anything. For there have been up to recently next to no promising formal results concerning compositionality. Those that were known were of the negative kind: they suggest that compositionality comes for free. [Janssen, 1997] showed that every recursively enumerable (= r.e.) language is compositional, and [Zadrozny, 1994] that *every* language is compositional (even if not r.e.). While [Westerståhl, 1998] criticises Zadrozny on the grounds that he changes the underlying semantics, the results by Janssen have largely remained unchallenged. They mean, however, that any claim about someone's approach being compositional is without any theoretical (and empirical) value. While the proof is formally correct, I have argued in [Kracht, 2001] that the notion of compositionality used there does not match our intuitions. Hence the problem is that triviality of a concept is shown that does not match the intuitive concept of the linguist. The way the principle is formally defined does not square with the intuitive understanding. Additionally, the intuitive concept requires that we know what the expressions and their meanings are, in other words, what the language actually is. In practice, many linguists would like to view questions of the latter sort to be open to arbitrary decisions. Part of my mission is to convince you that this view is mistaken. Clearly, the identity of a language is an empirical matter: one may debate whether this or that expression is an expression of, say, English, and if so, whether or not it has a given meaning. There is a growing acceptance of the view that language is full of microvariation, to the point that no two people actually speak exactly the same language. This makes it hard to define a single language for a given community (even computer languages are far from uniform).

Yet, there is a limit to the arbitrariness. I will argue below that there are clear intuitions as to what part of the representation belongs to semantics and what part does not, what part belongs to syntax and what part does not. This continues efforts begun in [Kracht, 2003b] where I have put down constraints on syntactic operations, and the subsequent [Kracht, 2007] where I have done the same for semantics. In what is to follow I shall expose the basic philosophy of the research and some of its conclusions. My own stance is as follows. Compositionality is a nontrivial property, and yet natural languages *are* compositional. Thus there are languages which simply are not compositional, although I believe that they are not the ones we speak. I am furthermore convinced that compositionality is not simply a "nice to have", it is a "must have". Without it we have a hard time understanding how the meanings of parts figure in the meaning of the entire sentence.

In addition, if we assume that compositionality holds we get deep insights into the structure of language. This opens a research agenda that is very different from mainstream linguistic research, which—following structuralist theory—used syntactic data at the exclusion of any other. Here we shall use semantic data in conjunction with syntax to reach conclusions about the structure. GB theory actually made up for the lack of semantics by introducing some semantics into syntax such as indices and thematic roles. It was claimed—contrary to ordinary understanding—that these were syntactic notions.[1] I will deal below with this assumption. Here I just point out that in the 80s Generative Grammar did posses a diagnostic tool for syntactic structure other than the substitution method and the then partly obsolete arguments from transformations: binding theory. Its removal has left the theory void of any diagnostic tool for structure. Here I will offer what I consider to be the first serious contender for such a criterion: *compositionality*. To make this a convincing case, however, I have to show that it is a nontrivial principle and it actually tells us a lot about how language is structured. [2]

2. E R

Compositionality is the thesis of autonomy of semantics. Here is what I consider to be a commonly agreed definition (taken from [Partee *et al.*, 1990]).

> The meaning of a compound expression is a function of the meanings of its parts and of the syntactic rule by which they are combined.

Let me go right into the exegesis of this definition. The definition attributes meanings to *expressions*, and not, as is commonly done in theoretical work, to analysis terms. This means that since several analysis terms can unfold to the same expression we have to deal with ambiguity. This is somewhat tangential to the argumentation, so I shall assume that meanings are calculated from analysis terms and not from expressions, though you may find me saying that expressions have meanings. The definition also speaks of "the meanings of its parts", which we therefore read as speaking about the meaning of the subterms of its analysis term. It clearly says that the parts already have a meaning. The way I render this into formal talk is as follows: a **language** is a set of signs; a sign consists of (at least) an expression

[1]Let me be clear that θ-roles were officially claimed to be syntactic and therefore void of semantic content as far as syntax is concerned. This view is compatible with mine; but the confusion over the exact nature of θ-roles was very real. Most people basically used them as if they had semantic content attached to them.

[2]Special thanks to Ruth Kempson and Glyn Morrill for giving me a forum for my ideas at their workshop "Foundations of Grammar Formalisms". Many thanks also to a reviewer for valuable comments. I have benefitted from talking to Ed Keenan, Hans-Martin Gärtner, Ben Keil, Udo Klein, Greg Kobele, Philippe Schlenker, Marcus Smith, and Ed Stabler. The responsibility for errors is, however, entirely my own.

and a meaning. A *grammar* is a finite set of constructions to form signs. More concretely, a **signature** is a pair $\langle M, \Omega\rangle$ (denoted by Ω), where M is a set and $\Omega : M \to \mathbb{N}$ assigns a number to every member of M (0 is also a number!). A **grammar** is a pair $G = \langle\Omega, \mathfrak{I}\rangle$, where Ω is a signature and $\mathfrak{I}$ a function assigning to each $f \in M$ a partial function from $S^{\Omega(f)}$ to S, where S is a set of signs. Since the functions may be partial, the identity of S is of little relevance. We shall assume however that $S = E \times M$, where E is a set of expressions and M a set of meanings. S just needs to be big enough. We also write f^G in place of $\mathfrak{I}(f)$. M is the set of **modes** or **constructions**. The **lexicon** is the set of all f with $\Omega(f) = 0$. $L(G)$ is the set of all signs that can be generated by G. This can be inductively defined. Let $L_0(G)$ be the set $\{f() : \Omega(f) = 0\}$ (lexical items). Then $L_{n+1}(G)$ is the set of all signs σ such that there is an $f \in M$ and signs $\sigma_1, \sigma_2, \ldots, \sigma_{\Omega(f)}$ such that

(1) $$\sigma = \mathfrak{I}(f)(\sigma_1, \sigma_2, \cdots, \sigma_{\Omega(f)})$$

Finally,

(2) $$L(G) := \bigcup_{n=0}^{\infty} L_i(G)$$

G is **compositional** if for every f there are f^ε and f^μ such that

(3) $$f^G(\langle\vec{x}_1, m_1\rangle, \cdots, \langle\vec{x}_{\Omega(f)}, m_{\Omega(f)}\rangle) = \langle f^\varepsilon(\vec{x}_1, \cdots, \vec{x}_{\Omega(f)}), f^\mu(m_1, \cdots, m_{\Omega(f)})\rangle$$

(As functions are partial this means that both sides are defined on the same input and yield the same result. The right hand side is defined if both f^ε and f^μ are defined.)

It is *grammars* which are compositional or not; compositionality of *languages* is a derived notion: a language is **compositional** if it has a compositional grammar. Thus, the meanings are given beforehand, they are not subject to arbitrary choice on our part. Contrast this with [Janssen, 1997], Page 427: "The choice of what the meaning of a part is, might depend on what we consider a suitable ingredient for building the meaning of the whole expression." This does not go as far as saying that we can do anything we please. Yet it grants us the right to adjust the meanings of the parts as long as we get the right sentential meanings. I shall return to that issue below. Here I shall deal briefly with the word "part". [Janssen, 1997] simply says the grammar determines what the parts are. Again this is too liberal. Basically, for $\vec{x}$ to be a part of $\vec{y}$, the material of $\vec{x}$ (the multiset of letters contained in it) must be contained in the material of $\vec{y}$. Otherwise linguistics becomes impossible to do. One piece of evidence is actually that we are otherwise unable to do any historical linguistics. Genetic relationships cannot be based on meaning coincidence alone; nor can they be established by pure sound coincidence regardless of meaning. Rather, you must find words (or phrases) in the respective languages that not only have similar enough meanings—they also must have similar enough sound structure. How could we provide the theoretical foundations for such a discipline if we allow deletion (or similar operations) into our syntactic process?[3]

For a long time linguists have experimented with operations that allow deletion of material (but most of them have given up on deletion and use plenty of empty material instead). The problem with deletion is that it obscures the part-of-relation. If anything can be done to strings, then you may claim that "compassion" is part of "wrench". There is nothing that rules that out a priori. It is to be noted, though, that the principle of additivity advocated here is not without problems. Some of them have been addressed in [Kracht, 2003b], Chapter 6.3. As a reviewer points out, Celtic languages militate against this. If we combine Welsh "am" and "merch" we get "am ferch", and not the expected *"am merch". This is indeed a moot point and will have to be dealt with elsewhere. My answer is roughly that additivity does not hold at the surface phonological level, only at the deep phonological level. This requires abstractness in the phonology. However, deep phonology is not far from surface phonology. Basically, I require that phonological processes are contact phenomena: sound changes at the juncture are permitted, as long as they are controlled by (morpho)phonological rules. This evidently needs careful investigation. What is important from a formal point of view is that deep phonology and surface phonology are not far apart. The restrictions I have in mind conspire to the effect that there is a finite state transducer from one to the other (see [Kracht, 2003a]).

Arguments in this paper against empty categories of a certain sort thus remain valid despite the abstractness of the actual level at which strict "surface" compositionality is supposed to hold, since the categories in question would require at least a context free transducer to pass between these levels.

3. G P P

In the literature one frequently finds claims that one analysis is superior to another on the grounds that it is compositional. The problem consists in knowing whether or not a given proposal is compositional. For example, [Kallmeyer and Joshi, 2003] claim to have provided a compositional

[3]This should not be dismissed on the grounds that I am confusing synchrony with diachrony. For I insist that a proper theory of language must be able to explain not only how languages can be like but also how they can change. In other words, it must provide a theoretical basis on which historical linguistics can be founded. If deletion is freely available we must provide an argument as to why it does not show in the historical development. Or else we must show why deletion provides predictions of the right sort of changes, something I think is quite impossible to do.

semantics. How can we see whether they are right? And what would be the rationale for our decision? A glance at the literature reveals that the problem lies in a complete lack of standards concerning *semantics* and *syntax* alike. It seems that we are free to declare syntactic positions to be part of semantics (as is done in [Kallmeyer and Joshi, 2003]) as we are free to declare indices to be part of syntax as is done in Generative Grammar.

This is where all the confusion starts. If anything, semantics should deal with *meaning* and syntax (together with morphology and phonology) with *form*. I shall be more precise below and develop concrete ideas about both of them. Let us first of all see what happens if we remain totally agnostic about syntax and semantics. Suppose that syntax may contain some parts of meaning and semantics some parts of form. In this case syntax and semantics can effectively spy onto each other. Compositionality becomes vacuous.[4] Let's look into the problem. Suppose that L is a language, that is, a set of pairs $\langle \vec{x}, \mu \rangle$, where $\vec{x}$ is, say, a string, and μ its meaning. For simplicity we assume that L is unambiguous, that is, $\langle \vec{x}, \mu \rangle, \langle \vec{x}, \mu' \rangle \in L$ implies $\mu = \mu'$.[5] Now let's change the language somewhat and put

(4) $$L' := \{\langle \vec{x}, \langle \vec{x}, \mu \rangle\rangle : \langle \vec{x}, \mu \rangle \in L\}$$

There is a straightforward compositional grammar for L' if there is a grammar for the string language $\varepsilon[L]$ of L.

(5) $$\varepsilon[L] := \{\vec{x} : \text{there is } \mu{:}\langle \vec{x}, \mu \rangle \in L\}$$

Notice that $\varepsilon[L] = \varepsilon[L']$. Suppose that there is a grammar $G = \langle M, \mathcal{I} \rangle$ that generates $\varepsilon[L]$. (This means that the functions f^G are partial functions on strings only.) All we need is the following. Let h be the (unique) function such that for all $\vec{x} \in \varepsilon[L]$: $\langle \vec{x}, h(\vec{x}) \rangle \in L$. Now let f be an m-ary function on strings. Let $\sigma_j := \langle \vec{x}_j, \langle \vec{x}_j, \mu_j \rangle\rangle$. Define $\mathcal{K}(f)$ by

(6)
$$\mathcal{K}(f)(\sigma_1, \cdots, \sigma_m) := \langle f(\vec{x}_1, \cdots, \vec{x}_m), \langle f(\vec{x}_1, \cdots, \vec{x}_m), h(f(\vec{x}_1, \cdots, \vec{x}_m))\rangle\rangle$$

Finally, let $H = \langle \Omega, \mathcal{K} \rangle$. H is compositional in the sense defined above. The semantic function can see the entire string and so knows exactly what it has to do. And this is for a trivial reason: the string is already present in the "semantics". (For given a meaning μ, the associated string is its first projection, and the ordinary meaning its second projection.) The semantics of [Kallmeyer and Joshi, 2003] is a clear case of a semantics that contains

[4] Some theories, HPSG is an instance, do not distinguish a semantic level from a form level. There the notion of compositionality does not make much sense and I shall therefore refrain from commenting on HPSG.

[5] The proof works only if there is a bound on the number of meanings a given string can have.

structural information. (They do not deny that this is the case.) What the discussion above shows is that this makes compositionality trivial.

Importing semantics (or parts thereof) into syntax is also often seen. In Generative Grammar the leading idea is that syntax is autonomous (the counterpart of compositionality saying that the modes of composition in syntax pay no attention to the semantic properties); moreover, once the derivation is complete it has also compiled a logical form for the expression. This means that syntax is actually doing part of the job that semantics is supposed to do, not to mention the fact that minimalist grammars use an armada of functional categories whose labels are actually of semantic nature.[6] Define

(7) $$L'' := \{\langle\langle\vec{x},\mu\rangle,\langle\vec{x},\mu\rangle\rangle : \langle\vec{x},\mu\rangle \in L\}$$

This is a completely symmetrified language in which syntax contains a full record of semantics and conversely. This allows compositionality to be obtained in a completely trivial fashion: syntax has all information it needs anyway, and likewise for semantics.

Now, in the formal setting the question of what L is settled beforehand. However, given an already existing language it is not always straightforward to decide what its signs are. Also, it may not always be easy to separate form from meaning. What I definitely reject is the idea that either of them can be freely defined to suit theoretical needs or otherwise. Thus we must develop clear definition of what we think syntax and semantics actually are. Otherwise pathological examples like the above cannot be excluded and compositionality is indeed vacuous. Thus, the following two questions must be answered:

❶ What are semantic representations and how are they manipulated?
❷ What are syntactic representations and how are they manipulated?

These are grand questions indeed. Obviously, I shall not even get close to an exhaustive answer; what I shall do, however, is delineate the boundaries of both syntax and semantics. I shall argue that there are things that do not belong to syntax: indices are a case in point, and so are θ-roles. Likewise, there are things about which we can say with certainty that they do not

[6]As far as I can make out there are basically two camps: one which really *does* believe them to be completely formal and their labels to be completely arbibtrary. And another camp for which the labels express correlations with the content of the label. The proponents of the first interpretations must explain how it is that all negative elements share the same component that will make them end up in, say, NEG^0. If semantics plays no role, such regularities must be extremely rare. Or else abstract classes do not exist; each lexical element is a class of its own. Then the problem disappears but Generative Grammar becomes entirely lexical, which it emphatically refuses to be.

belong to semantics: any reference to order (in the string) and multiplicity is beyond the scope of semantics.

I shall also develop some positive notion of what I do think possible semantic representations look like and how they may be manipulated. Similarly for syntax. What will emerge is that if we believe that syntax and semantics are genuinely separate tiers and that compositionality holds then we can actually get a window into the sentence structure; for it then follows that certain meanings for sentences cannot be obtained other than by assuming a particular sentence structure. This is like the dream come true for the linguist: that we need not refer to hopelessly unclear notions as "property of the mind" or "I-language" to establish sentence structure; rather, that we can use (E-)language—however *with* semantics—to do the very same thing. This eliminates much of the arbitrariness in thinking about sentence structure and language in general.

The rest of the paper is divided as follows. I shall start with a list of negative examples; I shall say why I think that certain frameworks fall short of embodying compositionality. After that I expose my own ideas about what I think the correct solutions should look like and point to examples in the literature.

4. T T M G

Montague was arguably the first to popularise the principle of compositionality.[7] He also proposed a grammar that he claimed meets the standards. I shall discuss in depth two shortcomings that beset his approach and turn to later developments.

The first problem with Montague Grammar (MG) is that it is not what [Hausser, 1984] calls *surface compositional*. This has to do with the rules for quantification. Montague wanted to get type raising and alternative scopes without complicating the type hierarchy. He therefore resorted to the following trick, which has become popular in Generative Grammar: the verb is first fed pronouns, and these pronouns are later traded in for actual quantifiers. This allows to keep the type of the quantifier unique. However, it creates the need for external bookkeeping. Montague uses the semi-English pronouns he_i, where i is a natural number; these 'pronouns' are not allowed to occur on the surface. Rule S14 is responsible for their elimination. It states that an occurrence he_i is replaced either by a true pronoun of English (with the correct ϕ-features) or by a quantified DP binding the index i. I object to this mechanism on two grounds: first, it makes use of deletion (not only of the index but also of most occurrences of "he") and

[7] As a reviewer kindly pointed out, Montague was not the first to use this principle. But he arguably initiated a renaissance of this notion through his work.

thus obscures the part-of relation for structures. Second, what looks like a single rule (applying the quantified expression to sentence denotations) is actually a parametrised set of rules so that the function base is actually infinite. Although one can formulate a single rule, as Montague has done, the parameter it uses is not explicit in the representation (as I claim) and thus cannot figure in its definition.

Montague's approach has been refined in later developments, but the problem basically remains. As long as there is a need for an explicit accounting device people have felt the need to use indices. Yet, indices are a mere convenience. Whether or not you use the pronoun he_7 or he_{165} should really not matter. All that matters is whether you choose the same number for identical variables. I shall have more to say on this issue below.

Let me turn to a second objection against MG, which concerns the use of *types*. Many linguists and logicians alike seem convinced that the type system is grounded in reality. However, already at its very beginning it was laden with problems. To be able to give a homogeneous account for both individuals and quantified expressions Montague argued that names actually denote sets of properties. Thus, even though the universe had individuals in it (to have denotations for the type e) there was no way to refer to them as such; constants would consistently refer to the sets of properties that these objects satisfy. This led to the idea of type raising: each object of type α could alternatively be seen as a function of type $(\alpha \to \beta) \to \beta$ for each β. Although technically viable it leaves us with a simple question: what is the *actual* meaning of an expression: the one with the basic type or its raised version? Of course, one may say it is the basic type, and that the raised type is only derived for technical convenience; and that it is derived through a mode of composition. This is a problematic viewpoint because it destroys the naturalness of the type assignment. Unless type raising is eliminated types must be considered a mere convenience rather than an essential (!) property of the meaning. Notice, though, that the type raising mode is also parametric (with parameter β). This creates problems for its formulations and its elimination (see [Kracht, 2003b]) but they can also be overcome (see [Steedman, 2000] for a discussion within Combinatory Categorial Grammar).

The preceding discussion has shown that type assignments are far from unique in semantics. This in itself is not the source of complaint so long as the types assigned to the meanings result from genuine semantic considerations. What I object to is adjusting the type assignment to the needs of syntax. There is an additional problem I need to address here. Consider the Gaelic word "faic" 'to see'. The syntax of Gaelic is VSO. Under standard assumptions of categorial grammar (indeed, also Generative

Grammar), constituents must be continuous; thus the verb must form a constituent with the subject in Gaelic, while it is claimed to form a constituent with the object in English. There are three possible answers to this:

(1) The Gaelic word "faic" translates into $\lambda x.\lambda y.\mathsf{love}'(x, y)$ whereas English "to love" translates as $\lambda y.\lambda x.\mathsf{love}'(x, y)$.
(2) The translation assigned in Gaelic is the same as in English, but the mode of composition is different: the verb first composes with the subject rather than the object.
(3) The constituent structure is the same in both languages; thus, Gaelic has a discontinuous constituent comprising the verb and its object.

The first approach is the least attractive one for many reasons. The most important one is that it claims that "faic" cannot be translated as "to love" because the two mean different things. This is highly unintuitive. The second puts the work into the modes of composition. It is disfavoured by many because if we assume that, categorial grammar is not uniform across languages. It is a basic assumption of categorial grammar that the array of modes is constant across languages (Montague assumed only forward and backward application, for example). The third option is the most attractive one. Recently, proposals along this line have been advanced. One is the framework of Abstract Categorial Grammars of [de Groote, 2001] or the Lambda Grammars of [Muskens, 2001] and another is the work by Hoffman on Turkish (see [Hoffman, 1995]); the latter however does not address questions of semantics. A third proposal, somewhat similar to [de Groote, 2001], has been put forward under the name *de Saussure Grammars* in [Kracht, 2003b].

5. G G

Generative grammar is not directly compositional. What it claims, rather, is that the generative process yields a structure, LF, which can be interpreted compositionally.[8] This is to say that the structure can be interpreted bottom-up; but it is not produced that way. Much of recent Generative Grammar is actually very similar to Montague Grammar, so the criticism levelled against the latter applies more or less verbatim to Generative Grammar. Let me therefore seize the opportunity to look in more detail at the use of free variables, since this turns out to be a central issue. Sooner or later all approaches produce constituents of the following form, with e_5 and e_{186} being empty elements.

(8) $$e_5 \text{ loves } e_{186}$$

[8] See [Heim and Kratzer, 1998], on which I largely base myself.

These constituents are standardly interpreted in first-order models, which are triples $\langle M, I, \beta \rangle$ such that M is a set, I an interpretation of the constants, functions and relations, and β a valuation, that is, a function from variables to the domain. (8) is interpreted as follows.

(9) $$[\mathrm{e}_5 \text{ loves } \mathrm{e}_{186}]^{\langle M,I,\beta \rangle} = \begin{cases} \top & \text{if } \langle \beta(x_5), \beta(x_{186}) \rangle \in I(\text{loves}) \\ \bot & \text{otherwise} \end{cases}$$

Thus, the valuation can make the claim true or false, and similarly for the choice of indices. The problem is that this presupposes an unlimited array of pointing devices—not an innocent assumption. Suppose that in my derivation of "John loves Maria" I use different indices, say, I translate the VP as

(10) $$\mathrm{e}_{1001} \text{ loves } \mathrm{e}_{34}$$

Does my VP then have a different meaning from yours if you used (8) instead? Of course not; you and I use the same concept so the two occurrences should have the same meaning wherever they occur. In Generative Grammar, of course, the problem is obscured by the fact that "loves" occurs in (8) and (10) in three different constituents, all of which sound (and look on paper) the same. So, in defense of the theory we may simply say that the innermost occurrences in (8) and (10) actually do have the same meaning (say, $\lambda x.\lambda y.\mathsf{love}'(y, x)$), but the two VPs have different meaning. Granted, for most variables it also does not matter since they will be quantified away later. Some of them are not, however, and they cause concern. The thought that the choice of index does matter that much is troubling. Moreover, if we communicate with each other, indices do not get transferred (they are neither visible nor audible), and so either every message is completely closed, containing no free variables, or else it is possible to replace the variables by something else that does the job just as well without assuming concrete choices of names. It is this latter road that I shall be taking.

6. A G

Rather than building up structure from incomplete parts, adjunction operates on complete expressions and yields complete expressions again. This proposal had its advocates among other in Zellig Harris, Solomon Marcus and Aravind Joshi (see [Harris, 1979], [Marcus, 1967], [Joshi *et al.*, 1975] and much further work). It comes in two varieties. The contextual grammars by Marcus use string adjunction while tree adjunction grammars (TAGs) favoured by Joshi use tree adjunction—as the name says. The latter type of grammars has proved to be more popular. Because of the more explicit structural record one can define the necessary operations more easily.

What interests us here is whether there actually is a compositional grammar using adjunction of either kind. Consider the following examples.

(11) The police arrested a sailor and a barman.

(12) The police arrested a barman and a sailor.

These sentences are synonymous and have the same syntactic structure (apart from the terminal strings), which for illustrative puposes we take to be the following.

(13)
$[[\text{the}_{\text{D}}\text{police}_{\text{N}}]_{\text{NP}}[\text{arrested}_{\text{V}}[[\text{a}_{\text{D}}\text{sailor}_{\text{N}}]_{\text{NP}}\text{and}_{\text{Conj}}[\text{a}_{\text{D}}\text{barman}_{\text{N}}]_{\text{NP}}]_{\text{NP}}]_{\text{VP}}]_{\text{S}}$

(14)
$[[\text{the}_{\text{D}}\text{police}_{\text{N}}]_{\text{NP}}[\text{arrested}_{\text{V}}[[\text{a}_{\text{D}}\text{barman}_{\text{N}}]_{\text{NP}}\text{and}_{\text{Conj}}[\text{a}_{\text{D}}\text{sailor}_{\text{N}}]_{\text{NP}}]_{\text{NP}}]_{\text{VP}}]_{\text{S}}$

Now apply tree adjunction to the last N:

(15) The police arrested a sailor and a Portuguese barman.

(16) The police arrested a barman and a Portuguese sailor.

The synonymy is gone even though we have applied the same operation. So, adjunction breaks the symmetry, which should not be possible. One may of course question the assumptions I made concerning the identity of structures or the synonymy of the original sentences. Neither is a particularly convincing strategy because the examples reveal quite a deep problem; it is that complete expressions have complete meanings associated with them. If we disassemble them at some point we have to say at which point the semantics needs to be changed. This turns out to be totally impossible. It is certainly impossible under a Montagovian view: in MG, a sentence denotes a truth value, at best a function from worlds to truth values. A truth value has no internal structure and cannot be disassembled. To be able to do the latter we need to assume that meanings are structured entities. Now, suppose we grant that. (I am happy to assume that meanings are structured. This is anyway the mainstream philosophical view on propositions.) Suppose that the way semantics is structured is somehow similar to the syntactic structure; let's say it is a flattened image of the syntax tree, quite like f-structure in LFG. Let's assume even that its structure is the same. In the coordinated structure above the adjective can be adjoined to two nodes, and this makes a difference in meaning. The operation of adjunction has to be mimicked by a similar operation in semantics. This operation can also target two nodes; depending on what syntax is doing it must choose the one or the other. The bigger the tree gets the more adjunction sites are created and the bigger the need for information to ensure exact coordination between syntax and semantics. At this point the idea breaks down: the amount of information that needs to be shared between syntax and semantics must be unlimited. In a compositional grammar it can only be finite (the choice of the mode is the only shared information). This is the opposite of compositionality since it makes semantics dependent on syntax. I have performed a close analysis in [Kracht, 2008] and reached the conclusion that if TAGs were reformed to be compositional in the sense of the word then they would actually look more like Linear Context Free Rewrite Systems (LCFRSs). Again, this points us to the option of relaxing the constituent structure.[9]

7. T N B

Let us return to the initial questions: what are meanings, what are syntactic structures and how do they work? I shall first approach the question about the identity of meanings. Simply put I claim that meanings are *concepts*. However, the notion of a concept is somewhat vague. If you want more detail, I suggest to use representations from cognitive grammar (see [Langacker, 1987] or [Talmy, 2000] for examples). These are, in essence, two- or three-dimensional graphical formulae. What is important from a formal point of view is that these pictures lack any alphabetic tags to cross-reference positions. Since the pictures lack any linear order, it is not possible to identify elements by reference to some pre-defined order.

In [Kracht, 2007] I have proposed an approach which does not use pictures and instead abstracts from standard tuple-based semantics of predicate logic. The idea behind the semantics is that concepts are not actually true of tuples of individuals, but of the set of these individuals. The difference between a tuple and the set of objects contained in it is that the tuple allows for multiple occurrences and is sensitive to the order of appearance. There are—I believe—strong metaphysical reasons to reject the tuple based approach and favour the set based approach (see [Fine, 2000]). They have to do with the fact that there is no real world correlate of the ordering (or the places into which the relata are plugged). A relation R holds in the standard sense of the pair $\langle a, b\rangle$. Or we may say that it holds with a occupying the first position (or subject position, or whatever you want to call it) and b occupying the second (or object) position. We can give up the order once we have recovered the positions into which the objects are plugged. The question that arises is what the relatum of the first and second position is in the world. There is none that can exist a priori. This is because R holds of a occupying the first position and b the second if and only if $R^{\smile}$ holds of a occupying the second and b the first position. Thus the notion of first and

[9] The interpretation algorithm proposed in [Kallmeyer and Joshi, 2003] is actually bottom-up over the analysis tree (= analysis term). TAGs however produce the tree top-down: start with a single tree and iteratively adjoin adjunction trees. A bottom-up reformulation of TAGs inevitably leads to LCFRSs. See [Kracht, 2008] for a discussion.

second position cannot be independent of the relation that we are considering.

Now, metaphysical considerations are not enough when it comes to language. I do however suggest that our mental representations essentially function in the same way. There are some arguments in support of this position. The first concerns multiplicity. Consider the fact that "John loves John" is rendered in English as "John loves himself", involving an arity reducing device. This construction is found in language after language. This suggests that languages treat re-use of arguments rather differently. The second argument concerns the fact that languages by preference try to systematise argument positions. Languages do try to single out some notion of subject or actor, for example. This is not always successful, and it is far from easy to identify semantically what constitutes a canonical subject in any given language (see [Levin and Rappaport Hovav, 2005]). Even so, such efforts would be completely pointless in a tuple based semantics, since the positions to which an argument may be assigned are already wired into the relation. We already know what it is to be in first position in a given relation. If however we have no relations but something without preassigned positions, it makes sense to try and establish such a system. Additionally, one hardly finds a language having a relation in addition to its inverse. Not only would this be an unnecessary luxury (the language cannot express more facts this way); it would also wreak havoc to a system that establishes positions on the basis of semantic criteria.

In addition to positions and multiplicity, predicate logic uses another device that is suspect: indices. Indices identify variables. If we want to insert a variable into a representation we must choose an index. This is far from innocent. [Vermeulen, 1995] identified a very important deficit of Dynamic Semantics (DS). Like Discourse Representation Theory (DRT) in [Kamp and Reyle, 1993], DS had to rely on an external demon to insert the correct variable names. To see an example, look at (17).

(17) A man walked in. Then another man walked in.

The translation must be something like this:

$$\begin{aligned}(18)\quad & \exists x_0; \mathsf{man}'(x_0); \exists t_0; \mathsf{walk\text{-}in}'(t_0, x_0); \exists t_1; t_0 < t_1; \exists x_1; x_1 \neq x_0; \\ & \qquad \mathsf{man}'(x_1); \mathsf{walk\text{-}in}'(t_1, x_1)\end{aligned}$$

Any indexing suffices, however, as long as the index attached to the variable in the second sentence (not to mention the time variables) is different from the one chosen in the previous sentence. On the other hand, the words "a", "man", "walked" in the first sentence each use the same variable. How can this be accounted for? [Kamp and Reyle, 1993] assume that this is achieved by having the parser generate the indexing and pass that to the semantics. Vermeulen points out, though, that the implicit assumption is that every variable is global, and that merge will assume that variables are the same in both systems if they are the same string. Instead, Vermeulen offers the opposite view: variables are local by default, or “anonymous”. Unless otherwise stated, merge will make the variable sets disjoint. To prevent this, referent systems have the option of connecting a variable to a **name**. The name is communicated to the outside world and is visible in particular to the merge algorithm. If the referent systems have referents that have the same name, the substitution will make them the same, all others however are being made different (see [Kracht, 1999] for an elaboration of that theory). But notice that the names of the referents are part of the semantics; however, by design they encode nonsemantic (for example morphological) properties. In this way they provide semantics with a window—albeit a small one—into syntax. This semantics is therefore not compositional.

For similar reasons Kit Fine accused the semantics for predicate logic to be noncompositional (see his [Fine, 2003]). His source of complaint was that for all intents and purposes the choice of a variable does not matter. There is no difference in a proof that begins “Let PQR be a triangle” from a proof that begins “Let ABC be a triangle” as long as letters are consistently exchanged. In Fine’s words, predicate logic is not *alphabetically innocent*—but it should be. I am unsure about the conclusion that the semantics for predicate logic is not compositional. For the *actual* variables of predicate logic are of the form x_i, so exchanging the numbers does produce a genuinely different statement. Additionally, the semantics of (formal) predicate logic is codified. The semantics part of the language and cannot really be changed. However, this only applies to the formal theory; in actual practice we do not use predicate logic like that. Rather, we quickly start to use what are in fact metavariables. It is the use of the latter that makes predicate logic problematic. Moreover, inasmuch as predicate logic is used to describe natural language meanings, the following question becomes significant: does it make sense to have an a priori given set of variables? Fine answers the question differently from Vermeulen. He wishes to think of variables as acquiring meaning through the position at which they occur. In and of themselves, all variables have the same meaning: it is the entire range of values. There is no way to tell the difference between x and y. It depends entirely on their context of use.

My own solution of the problem is different from the previous two. To see what its motivations are, consider the problem of teaching your child the concept “to stab”. One way is to give a verbal definition. Another is to point at a scene in a film or a picture (!) and say: “See, *he*” (pointing your finger at Brutus) “is stabbing *him*” (now pointing at Caesar). You may also add that it necessarily involves a knife, and so on. The order that the arguments find themselves in your sentence helps us keep them apart (in other languages case marking can do that job as well). However, it has no correlate in any ordering that is defined for the picture. There is no order in three dimensional space that corresponds to the linear order in a sentence. The association between a linear position in a sentence and a figure in a picture is determined by other factors, and they have to do with the meaning of the concept itself. When you learn the concept of "stabbing" you also learn what it is to qualify as a subject of stabbing and what it is to qualify as an object. What I assume is that the picture is actually very much the way we should think of concepts; concepts are represented as image schemata, and there is no linear order in the positions. If you want a concrete example to hold on to, think of cognitive grammar, as in [Langacker, 1987]. The pictures also raise another point. Two occurrences of some graphic element necessarily denote two different real objects because they occupy different places. If you see Brutus twice in a picture you will assume that there are two men who look astonishingly alike.[10]

For example, suppose you need to explain the concept "seppuku". You could (somewhat inaccurately) say: "It is to stab oneself."; or, alternatively, you can point to a picture of Mishima and say: "See, he is committing seppuku." Finally, you may say: "He" (pointing at Mishima) "is stabbing himself" (again pointing at Mishima). The picture will not contain two copies of Mishima, even though the concept of stabbing seems to need two individuals. The duplicity of variables we have in the notation is not reflected in the picture.

This leads to the following definition. Rather than using sets of n-tuples to represent meanings we use a more abstract notion, where order and multiplicity of the relata is "suspended". The abstract entities are called *concepts*. This is a rather formal notion of concept, but nothing stated in this paper hinges on the particular version used. Concepts can involve objects of different types (which allows to have time points locations and so on), but we shall work here just with individuals. An n-ary *relation* is a subset of M^n, where M is the domain of the first-order structure. Let $R \subseteq M^n$ be an n-ary relation and $\pi : \{1, 2, \cdots, n\} \to \{1, 2, \cdots, n\}$ a permutation. Put

(19) $$\pi[R] := \{\langle a_{\pi(1)}, a_{\pi(2)}, \cdots, a_{\pi(n)}\rangle : \langle a_1, a_2, \cdots, a_n\rangle \in R\}$$

(20) $$p_i[R] := \{\langle a_1, a_2, \cdots, a_{n+1}\rangle : \langle a_1, a_2, \cdots, a_n\rangle \in R, a_{n+1} = a_i\}$$

(21) $$E[R] := R \times M$$

[10]You may also think that the picture has been manufactured, but this is a totally different issue. The picture here is intended to be just a replica of reality. I am interested in what we see and how it is represented. I would be surprised if our mental representations were such that two different occurrences of Brutus are stored in different places.

Write $R \approx R'$ if there is a relation R'' such that R and R' can be transformed into R'' using any combination of the above operations. Then put

$$[R]_{\approx} := \{R' : R \approx R'\} \qquad (22)$$

A **concept** is a set of relations of the form $[R]_{\approx}$ for some relation R. We say that R and R' **express the same concept** if $[R]_{\approx} = [R']_{\approx}$, that is, if $R \approx R'$.[11]

To give an example, let $M := \{a, b\}$. The set $\{\langle a, a\rangle, \langle a, b\rangle, \langle b, b\rangle\}$ expresses the same concept as $\{\langle a, a\rangle, \langle b, a\rangle, \langle b, b\rangle\}$ (since we can exchange the first and second position). The concept $\{\langle a, a\rangle\}$ expresses the same concept as $\{\langle a\rangle\}$, since the first and second are always identical. $\{\langle a, a\rangle, \langle b, b\rangle\}$ expresses the same concept as $\{\langle a\rangle, \langle b\rangle\}$, and we also have $[\{\langle a\rangle, \langle b\rangle\}]_{\approx} = [\{\varnothing\}]_{\approx}$ since $E(\{\varnothing\}) = \{\varnothing\} \times M$. (Here we use the familiar identity $\{\varnothing\} \times M \cong M$; they are not identical as sets, but considered identical here. Recall also that $\{\varnothing\} = 1$ in standard set theory.)

This has consequences worth pointing out. The relation denoted by "to the left of" is the inverse of the relation denoted by "to the right of". Likewise, the relation denoted by "to be seen by" is the inverse of the relation of "to see". If the above is right, then the two pairs, although denoting different *relations*, actually denote the same *concepts*. They only differ in the way their arguments are *linked* to positions.[12]

8. H A M M ?

Constituents denote concepts, which are sets of equivalent relations (not necessarily of same length). When two constituents are joined into a constituent, what happens to the concepts involved? In Montague Grammar the answer was: in that case one constituent denotes a function that can be applied to the denotation of the other one. There was only one map: function application. (Later versions of Categorial Grammar have added more functions, but they are more or less "derived" from this primitive one and type raising, which did not exist in MG.) If we unravel the functions into statements we basically perform a conversion from a functional signature to a relational signature. The analogue of function application is identification

[11] Careful analysis of these definitions will reveal that they do not entirely live up to the promise: it is at present not always possible to remove duplicate elements from a relation. For example, with $M = \{a, b, c\}$, $R = \{\langle a, a\rangle, \langle a, b\rangle\}$ is a relation that contains the pair $\langle a, a\rangle$, but is not reducible to a unary relation by these principles. This still awaits careful consideration. However, the conclusions reached in this paper do not seem to depend on the ultimate solution of this problem.

[12] The idea that "x is to the left of y" is the same fact as "y is to the right of x" and therefore a relation is identical to its converse has been advanced in [Geach, 1957] and has subsequently been defended among others in [Williamson, 1985] and [Fine, 2000]. See also [Leo, 2006] for discussion. We shall return to the issue of "left" and "right" on Page 20.

of variables. Additionally, we need existential quantification. We are thus led to propose that when we merge two constituents C and D, the map that combines the meanings of C and D can do only two things: it can identify (possibly several) sets of variables, and it may quantify away some of the variables. In an algebraic setting (which is what we use here), identifying the variables j and k means intersection with the identity d^n_{jk}:

$$d^n_{jk} = \{\vec{a} \in M^n : a_j = a_k\} \tag{23}$$

Also, existential quantification means applying one of the projection functions p_i.

This idea must now be lifted from relations to concepts. To that effect, let C and D be constituents with meanings $\mathfrak{c}$ and $\mathfrak{d}$, respectively. The first stage is to pick appropriate relations $P \in \mathfrak{c}$ and $Q \in \mathfrak{d}$; then we form the product $P \times Q$ and intersect it with an appropriate identity relation. We may additionally apply projections, corresponding to existential closure. Finally, we take the concept that results from this relation.

What is important is that there are no variables; also, we have no first hand indication in which order the arguments are presented when we pick P and Q. What we have to do then is to find a way to make the definitions either independent of the choice of P and Q or make sure than we can actually pick P and Q uniquely. The latter is made possible through the introduction of a **linking aspect**. Let the **arity** $\alpha(\mathfrak{c})$ of a concept be the least n such that there is a $P \in \mathfrak{c}$ and $P \subseteq M^n$. Every relation of length $> \alpha(\mathfrak{c})$ is derived from a relation of length $\alpha(\mathfrak{c})$ by adding either a copy of some row or multiplying with the domain M. A linking aspect is a function that associates with each concept $\mathfrak{c}$ a finite set $A \subseteq M^{\alpha(\mathfrak{c})}$ such that there is a (necessarily unique) $P \in \mathfrak{c}$ of length $\alpha(\mathfrak{c})$ such that $A \subseteq P$ and for every $Q \in \mathfrak{c}$ of length $\alpha(\mathfrak{c})$: if $A \subseteq Q$ then $P = Q$. It is perhaps a surprising fact (but not difficult to show) that linking aspects always exist. The idea is this: pick any P you like (of length $\alpha(\mathfrak{c})$), and choose two positions j and k. Either it really does not matter whether we exchange columns i and j, or it does. If it does, there is an n-tuple $\vec{a}$ such that P holds of $\vec{a}$, but not of the result of exchanging the items a_j and a_k. Let $\vec{a} \in A$. We need one vector for each pair of variables at most. Or it does not, then it does not matter how we serialise the two positions with respect to each other. In that case, no vector is put into A for this pair. For example, a binary relation R is either symmetric and then you do not need to pay attention whether you say $R(a, b)$ or $R(b, a)$; or it is not, and then there is a pair $\langle a^*, b^* \rangle \in M^2$ such that $R(a^*, b^*)$ holds but not $R(b^*, a^*)$. In this case, the set $\{\langle a^*, b^* \rangle\}$ can be used as a value $[R]_\approx$ of a linking aspect.

Thus, the merge proceeds like this. Using the linking aspect, variables are temporarily dragged out of their anonymity. Identification of variables and existential closure is performed, and then we let the variables sink back into anonymity. It is known that this is sufficient. Any language that has functions in it can be reduced to a relational language; the equivalent of function application is identification of variables. Namely, if f is the function, and if it is applied to x, then the reduction of f will produce a statement of the form $y = f(u)$ and "application" consists in adding the equation $u = x$. The set theoretic coding of functions actually does exactly that. (The variables x and y will actually end up being coded by just one position, since they share the values.)

Now, if moving from relations to concepts does not change the expressive power, why bother? One answer is that there actually *is* a difference. If P and Q are in the same concept, they will actually be treated alike. This produces delicate interactions. I have said above that the meaning of passive sentences is the same as that of the corresponding active sentences. They denote the same concepts but different relations. To get them to behave differently we have to use their form.

9. S R

There is perhaps nothing more difficult as agreeing on minimal standards for syntactic representations. Yet, we have to try. Syntactic constituents have a **category** and an **exponent**.[13] Pushing aside a few concerns about segmentability (see also the short discussion on Page 5), I consider exponents to be sequences of strings. If the constituent is continuous, we just have a single string, but to have pairs of strings is not uncommon. The idea of manipulating tuples has been reintroduced through Literal Movement Grammars (LMGs) in [Groenink, 1997]. Linear Context Free Rewrite Systems (LCFRSs) are particular LMGs that use a context free grammar where the exponents are tuples of strings. It has been shown in [Michaelis, 2001] that Minimalist Grammars in the sense of [Stabler, 1997] can be reduced to LCFRSs. Thus, even if one likes to think of syntactic representations as trees, there is no reason to dismiss tuples of strings as insufficient. They serve the purpose just as well in Minimalism and—so I believe—elsewhere. For example [Pollard, 1984] has shown that head grammars, a variant of 2-LCFRSs, can deal with crossing dependencies. [Calcagno, 1995] has used head grammars with a categorial backbone to provide a compositional semantics of Swiss German, which until today the most elegant and simple

[13] As a reviewer correctly observed, the categories are de facto exempt from the non-deletion requirement. Two remedies suggest themselves. The first is to treat categories as types of strings, thus resorting to a system of types quite like the one employed in semantics. In that case, the category is not a substantive part of the string; rather, it forms part of the identity criterion for it. The other solution is to actually abandon categories altogether. This is more minimalistic in spirit but it is not clear to me whether it works.

solution I know of. This list can be prolonged. Basically, the weak equivalence of multicomponent TAGs with LCFRSs shown in [Vijay-Shanker *et al.*, 1987] is another indication. However, notice that the commitment to tuples of strings does *not* mean that we have to restrict ourselves to concatenation. A modicum of copying is in all likelihood needed (see [Michaelis and Kracht, 1997], whose basic insight remains valid in this connection despite [Bhatt and Joshi, 2004]).

After the nature of exponents has been discussed, let us now turn to categories. The categories form the actual grammar that links exponents with meanings (= concepts). A grammar rule of the form S → NP VP is in actual fact translated into a binary mode f that operates as follows. $f = \langle f^\varepsilon, f^\kappa, f^\mu \rangle$ where f^ε is a binary function on tuples of strings, f^μ is a binary function on concepts, and f^κ a binary function on categories, defined only on the pair ⟨NP, VP⟩ with result S. In English, for example, $f^\varepsilon(\vec{x}, \vec{y}) = \vec{x}^\frown \square ^\frown \vec{y}$ (with $\square$ representing the blank), but verb second in German will require a more complex function to be used.

An important principle is

> I I . If T and U are *n*-tuples of strings that occur in the same environments, then they have the same category.

This principle says that categories should not make distinctions that go beyond the need of syntax. To see the effect of this principle, let us return to the distinction between active and passive. On the face of it, we might simply classify all active verbs as V[+act] and all passive verbs as V[−act]. This would allow active and passive verbs to link differently. However, consider a language in which passives are not syntactically distinct from actives; for example, suppose that passives are derived by having subject and object swap places (and case marking). Then, by the above principle, actives and passives cannot be distinguished by category. If that is the case, they are the same on all three levels, and the distinction disappears. In English, actives and passives actually are different syntactically. Passives fail to have a transitive object. This is the way syntax can distinguish them. Similarly, "buy" and "sell" are not mirror images of each other; their argument frames are actually quite distinct: you buy *from*, but you sell *to*. Altogether different is the pair "left" and "right", which seem to be perfect mirror images of each other in syntax. In order to treat them correctly, we have to do either of the following. (1) We choose different variables to be identified; or (2) we use a different linking aspect for the two. The linking aspect of "left", for example, must be effectively be such that it always picks the relational inverse of the one that is picked for "right".[14] The categories are the same, and so is the semantics. Still we are able to correctly identify the variables. For we postulate two binary functions on exponents: one that combines "left" with its complement, and another that combines "right" with its complement. The syntactic function is the same in both cases. In semantics we use different linkings. The trick is to make the functions on the exponents *partial*: one can *only* concatenate "left"—and not "right"—with its complement; the other can *only* combine "right" but not "left". In this way, semantics "knows" from the mode that is being applied what linking aspect to apply. Thus, unlike in silent passives, a surface distinction exists and can therefore be exploited to discriminate the two meanings despite the fact that the concepts are the same.

10. A H A S R M ?

First of all, tuples may be manipulated by concatenating their parts or by permuting their members. However, more complex operations are conceivable, the most important one being *copying*. There are, I think, genuine instances of copying, which include plural in Malay, yes-no-questions in Mandarin, and case stacking in Australian languages. Not all languages use it, but it is an available option. In [Kracht, 2003b] I have tried to give a few criteria of what constitutes a legitimate syntactic operation. First, there is no deletion and strings cannot be dissected. This means that every part of the tuple can be traced to at least one occurrence of this string as a substring of the entire constituent. (In the case of copying there may evidently be more.) Second, there are no syncategorematic symbols. This may be controversial; however, I know of very few exceptions to this rule. In writing, the blank is a syncategorematic symbol. In German compounds, certain sounds are inserted that have no semantic function (Fugen-s and others). Small exceptions as these aside, however, no symbol is truly syncategorematic. This is important and often underestimated: without such a principle it is impossible to verify that any symbol occurring in a sentence or any part of the sentence is actually an occurrence of a constituent. Third, the exponents do not contain any empty symbols; that is to say, empty elements really leave no trace in the representation. This may be discomforting in view of the fact that many theories (mostly variants of Generative Grammar) assume a plethora of empty categories. But it is not clear that their presence is really needed other than to remind the reader that some operation has been performed in the derivation. There is nothing wrong with empty exponents,

[14] The alert reader may note that it is perfectly acceptable to use several linking aspects concurrently. There is nothing that rules that out.

but their presence should actually be irrelevant for the definitions. For example, two signs which only differ in that one contains an empty category somewhere where the other does not, are identical. Also, there is no indexation. Hence you cannot tell which argument has been added, and so on. Also, empty categories may not be used when invoking the principle of identity of indiscernibles. Suppose that two constituents C and D differ in that C can occur in those contexts that differ from those D can occur in except that an empty pronoun has to be added. Then the principle requires them to have identical category. Empty elements in the context make no difference. Again, all those who deny the validity of this requirement will have to tell me how they can tell "good" from "bad" use of empty elements in distinguishing categories. I have not come across an example that would necessitate giving up this stance. Again, I should stress that I do not object to the use of empty elements in representations, as long as it is clear that they are for the eye only. This means that I contest that trees in Generative Grammar are stored verbatim in the head. Any alternative representation that serves the same purpose is a serious contender for "internal representation". Until hard evidence to the contrary comes up, I will therefore remain with tuples of strings and the above rules for their manipulation.

11. S W D T B U ?

The present theory substantially complicates life. Even translating a simple sentence like "Scipio stabbed Germanicus" involves several steps of 'aligning relations'. First, from the lexicon we know the concept denoted by the verb. Call it $\mathfrak{s}$. $\mathfrak{s}$ contains plenty of relations. However, it contains only two binary relations. Since we don't know which one to pick, we need to consult the linking aspect. Let Y be our linking aspect. Now, suppose $Y(\mathfrak{s}) = \{\langle b, c\rangle\}$, where b is Brutus and c is Caesar. We take the one binary relation $P \in \mathfrak{s}$ for which $\langle b, c\rangle \in P$. This assures us that whatever is the first member of a pair is the actor who stabs the second member of the pair.[15] Now we check whether or nor $\langle s, g\rangle \in P$, where s is Scipio and g is Germanicus. If so, the sentence is true. Otherwise it is false. The linking aspect seems like a very roundabout way to achieve this. However, it is actually very easy to apply. Suppose I have been shown a picture of Brutus stabbing Caesar; and that in addition I have been told that "he" (pointing at Brutus) "is stabbing him" (pointing at Caesar). If I have correctly grasped the concept from the scene, I have extracted an abstract image schema which I

[15]Notice that in order for us to know this it is enough to know that Brutus stabbed Caesar and not vice versa. This in turn tells us what Scipio did to Germanicus according to that sentence. All that is involved is substituting constituents in analogous positions.

can now invoke when I hear that Scipio stabbed Germanicus. I place Scipio into the schema where Brutus had been, and Germanicus where Caesar had been. This allows me to understand what it means that Scipio stabbed Germanicus. And it allows me to say whether this is actually true. What is important is that the entire process works without numbers or indices, it just uses positions whose identity is recovered by means of an "ideal situation" (perhaps this is effectively what Kit Fine had in mind). Abstractly, the "ideal situation" is provided by the linking aspect.

I have noted in [Kracht, 2007] that one is actually better off thinking of the linking aspect as a dynamically created object, and that this would allow for the entire process to be finitely computable. All one needs to understand is how to extend a linking aspect to new concepts. This may sound very complex but I claim that it actually is much closer to what language (and language processing) is really like.

However, there are also quite tangible benefits. I shall mention one, which I proved in [Kracht, 2007] on the basis of the assumptions so far:

Theorem 1. *There is no compositional context free grammar for Dutch.*

The theorem is true even if Dutch is weakly context free. Notice that while it has often been suggested that this is the case, here it actually falls out as a consequence of the theory.

This is far from trivial. First of all, all constructions involving raising infinitives seem to have distinct meanings. Therefore there exists a computable map from semantics to syntax. (This can abstractly be shown on the basis that there is a computable map from syntax to semantics.) In that case there is a computable compositional grammar that is context free. However, this works under the assumption that the semantic functions can be anything we please. However, we have argued that all semantics can do is identify positions in relations and quantify them away. Then the situation is quite different. Suppose we form a constituent using a verb, say "teach" and a noun, say "Tullius". The number of outcomes in semantics is severely limited. We have two basic options: we can identify one of the positions in the concept teaching with that of Tullius. This gives us the concepts of "teaching Tullius" or of "Tullius teaching (someone)". Or we may resist identifying the two variables, in which case we get the concept of "teaching someone and there is Tullius". Modulo existentially quantifying off some variables this is all that semantics can produce from the input. The proof now goes as follows. If Dutch crossing dependencies are treated as nested dependencies then one must refrain from identifying any NP-variables with any of the arguments of the verbs until the entire verb cluster is complete.

Once it is completed, however, one has lost any recollection of which argument appeared at which place in the structure.[16] And so one cannot unambiguously arrive at the correct meaning.

We have argued that semantics really is quite a simplistic creature. It basically needs the arguments in the correct order unless there is a way to tell them apart. As soon as the concept has two distinct positions that can be filled by different arguments we must have semantic means of telling which argument fills what place. If we don't we are lost.

12. C

The structuralist doctrine has it that syntactic structure can be assessed mainly if not exclusively through the study of syntax alone. Semantic considerations are at best viewed as giving hints as to where to look for evidence. Yet it seems that the syntactic fine structure unearthed in the Minimalist Program is de facto justified by semantic considerations, while the official doctrine makes it appear to be a consequence of internal structural principles, parts of which are of questionable theoretical status (for example, Kayne's antisymmetry thesis, see [Kayne, 1994]). I have argued here that by contrast, compositionality is a simple idea and is shared at least pretheoretically by linguists of many persuasions. Moreover, if we properly separate syntax and semantics then compositionality becomes a powerful tool for investigating the structure of language(s).

R

[Bhatt and Joshi, 2004] Rajeesh Bhatt and Aravind Joshi. Semilinearity is a Syntactic Invariant. A Reply to Kracht and Michaelis 1997. *Linguistic Inquiy*, 35:683 – 692, 2004.

[Calcagno, 1995] Mike Calcagno. A Sign–Based Extension to the Lambek Calculus for Discontinuous Constituents. *Bulletin of the IGPL*, 3:555 – 578, 1995.

[de Groote, 2001] Philippe de Groote. Towards Abstract Categorial Grammars. In *Association for Computational Linguistics, 39th Annual Meeting and 10th Conference of the European Chapter*, pages 148 – 155, Toulouse, 2001.

[Fine, 2000] Kit Fine. Neutral relations. *The Philosophical Review*, 109:1 – 33, 2000.

[Fine, 2003] Kit Fine. The Role of Variables. *Journal of Philosophy*, 50:605–631, 2003.

[Geach, 1957] Peter Geach. *Mental Acts*. Blackwell Publishers, Oxford, 1957.

[Groenink, 1997] Annius V. Groenink. *Surface without Structure. Word Order and Tractability Issues in Natural Language Analysis*. PhD thesis, University of Utrecht, 1997.

[Harris, 1979] Zellig S. Harris. *Mathematical Structures of Language*. Robert E. Krieger Publishing Company, Huntington, New York, 1979.

[Hausser, 1984] Roland R. Hausser. *Surface Compositional Grammar*. Wilhelm Finck Verlag, München, 1984.

[16] More exactly: a bounded amount of knowledge may persist. Thus if the construction can be performed with arbitrary embedding, the argument goes through.

[Heim and Kratzer, 1998] Irene Heim and Angelika Kratzer. *Semantics in Generative Grammar*. Blackwell Publishers, Oxford, 1998.

[Hodges, 2001] Wilfrid Hodges. Formal features of compositionality. *Journal of Logic, Language and Information*, 10:7 – 28, 2001.

[Hoffman, 1995] Beryl Hoffman. *The Computational Analysis of the Syntax and Semantics of "Free" Word Order in Turkish*. PhD thesis, University of Pennsylvania, 1995.

[Janssen, 1997] Theo Janssen. Compositionality. In Johan van Benthem and Alice ter Meulen, editors, *Handbook of Logic and Language*, pages 417 – 473. Elsevier, Amsterdam, 1997.

[Joshi *et al.*, 1975] Aravind Joshi, Leon S. Levy, and Masako Takahashi. Tree Adjunct Grammars. *Journal of Computer and System Sciences*, 10:136 – 163, 1975.

[Kallmeyer and Joshi, 2003] Laura Kallmeyer and Aravind Joshi. Factoring Predicate Argument and Scope Semantics: Underspecified Semantics with LTAG. *Research in Language and Computation*, 1:3 – 58, 2003.

[Kamp and Reyle, 1993] Hans Kamp and Uwe Reyle. *From Discourse to Logic. Introduction to Modeltheoretic Semantics of Natural Language, Formal Language and Discourse Representation*. Kluwer, Dordrecht, 1993.

[Kayne, 1994] Richard S. Kayne. *The Antisymmetry of Syntax*. Number 25 in Linguistic Inquiry Monographs. MIT Press, 1994.

[Kracht, 1999] Marcus Kracht. Agreement Morphology, Argument Structure and Syntax. Manuscript, 1999.

[Kracht, 2001] Marcus Kracht. Strict Compositionality and Literal Movement Grammars. In Michael Moortgat, editor, *Logical Aspects of Computational Linguistics '98*, number 2014 in Springer Lecture Notes in Artificial Intelligence, pages 126–142, 2001.

[Kracht, 2003a] Marcus Kracht. Features in Phonological Theory. In Benedikt Löwe, Wolfgang Malzkorn, and Thoralf Räsch, editors, *Foundations of the Formal Sciences II, Applications of Mathematical Logic in Philosophy and Linguistics, Papers of a Conference held in Bonn, November 11–13, 2000*, number 17 in Trends in Logic, pages 123–149, Dordrecht, 2003. Kluwer.

[Kracht, 2003b] Marcus Kracht. *The Mathematics of Language*. Mouton de Gruyter, Berlin, 2003.

[Kracht, 2007] Marcus Kracht. The Emergence of Syntactic Structure. *Linguistics and Philosophy*, 30:47–95, 2007.

[Kracht, 2008] Marcus Kracht. Is Adjunction Compositional? 2008.

[Langacker, 1987] Ronald W. Langacker. *Foundations of Cognitive Grammar*, volume 1. Stanford University Press, Stanford, 1987.

[Leo, 2006] Joop Leo. Modeling relations. manuscript, Utrecht University, 2006.

[Levin and Rappaport Hovav, 2005] Beth Levin and Malka Rappaport Hovav. *Argument Realization*. Research Surveys in Linguistics. Cambridge University Press, 2005.

[Marcus, 1967] Solomon Marcus. *Algebraic Linguistics; Analytical Models*. Academic Press, New York and London, 1967.

[Michaelis and Kracht, 1997] Jens Michaelis and Marcus Kracht. Semilinearity as a syntactic invariant. In Christian Retoré, editor, *Logical Aspects of Computational Linguistics (LACL '96)*, number 1328 in Lecture Notes in Computer Science, pages 329 – 345, Heidelberg, 1997. Springer.

[Michaelis, 2001] Jens Michaelis. *On Formal Properties of Minimalist Grammars*. PhD thesis, Universität Potsdam, 2001.

[Muskens, 2001] Reinhard Muskens. Lambda Grammars and the Syntax–Semantics Interface. In Rob van Rooy and Martin Stokhof, editors, *Proceedings of the Thirteenth Amsterdam Colloquium*, pages 150–155, 2001.

[Partee *et al.*, 1990] Barbara Partee, Alice ter Meulen, and Robert Wall. *Mathematical Methods in Linguistics*. Kluwer, Dordrecht, 1990.

[Pollard, 1984] Carl J. Pollard. *Generalized Phrase Structure Grammar, Head Grammars and Natural Language*. PhD thesis, Stanford University, 1984.

[Stabler, 1997] Edward P. Stabler. Derivational Minimalism. In Christian Retoré, editor, *Logical Aspects of Computational Linguistics (LACL '96)*, number 1328 in Lecture Notes in Artificial Intelligence, pages 68 – 95, Heidelberg, 1997. Springer.

[Steedman, 2000] Mark Steedman. *The Syntactic Process*. MIT Press, Cambridge (Mass.), 2000.

[Talmy, 2000] Leonard Talmy. *Toward a Cognitive Semantics*, volume 1 & 2. MIT Press, Cambridge (Massachussetts), 2000.

[Vermeulen, 1995] Kees F. M. Vermeulen. Merging without mystery or: Variables in dynamic semantics. *Journal of Philosophical Logic*, 24:405 – 450, 1995.

[Vijay-Shanker *et al.*, 1987] K. Vijay-Shanker, David J. Weir, and Aravind Joshi. Characterizing structural descriptions produced by various grammar formalisms. In *Proceedings of the 25th Meeting of the Association for Computational Linguistics (ACL '87)*, Stanford, CA, pages 104 – 111, 1987.

[Westerståhl, 1998] Dag Westerståhl. On Mathematical Proofs of the Vacuity of Compositionality. *Linguistics and Philosophy*, 21:635 – 643, 1998.

[Williamson, 1985] Timothy Williamson. Converse relations. *The Philosophical Review*, 94:249 – 262, 1985.

[Zadrozny, 1994] Wlodek Zadrozny. From Compositional Semantics to Systematic Semantics. *Linguistics and Philosophy*, 17:329 – 342, 1994.

D L , UCLA,, 3125 C H ,, PO B 951543,, L A -, CA 90095-1543,, USA

E-mail address: kracht@humnet.ucla.edu