Surface Compositionality and Incremental Semantic Interpretation*

Wolfgang Sternefeld, October 25, 2008

1 Introduction

Serious efforts have been recently undertaken to develop a semantic theory that is "psychologically real" in the sense of being closer to actual syntactic and semantic processing. In particular, the prospective Tübingen research group SFB 833, which is on its way to being established, centers around questions of meaning and its processing in complex sentences, and it is against this background that I want to clarify a number of theoretical issues that might be relevant for future empirical research in this area.

One of the questions that I consider as central to this new initiative is the following: given that standard techniques of semantic interpretation require highly abstract structures remote from the syntactic surface, and assuming that these structures are highly problematic as a basis for any realistic concept of syntactic (and semantic) processing, what are approproate alternatives to the standard techniques when applied to a few but crucial sample sentences of German?

In this (preliminary) paper of clarification, I will present and discuss a number of such alternatives, i.e. alternatives to the usual techniques employed in Logical Grammar in the tradition of standard text-book semantics, as exemplified by the introduction text book Heim and Kratzer (1998). While discussing various methods and techniques that might help to make semantics more "surfacy", I will not address one problem left to future research, namely the issue of actual online feasibility of these methods, i.e. the question whether or not these techniques are compatible with, or even corroborated by, actual semantic processing. This is a long term question to be pursued in the context of the future SFB.

*The syntactic part of the paper was presented at the Yonsei University Seoul on 19th Feb. 2008 and the University of Hiroshima on 22th Feb. 2008. Thanks to the audiences for discussion and also to the Deutsche Forschungsgemeinschaft DFG for funding within the SFB 433.

1.1 Motivation: Against Quantifier Raising

One thorn in the eye of surface true semantics is quantifier raising (QR). QR as such is discussed to some extend in the introductory textbook of Heim & Kratzer, of which I assume familiarity. The authors state a number of arguments in favor of QR, but they also demonstrate that QR massively overgenerates and that it is incompatible with a sober syntactic analysis, ie. one motivated by syntactic considerations alone. To give an example of the first problem of overgeneration, it is well known that QR cannot apply in cases like

(1) His father loves every boy

Intuitively, the pronoun *his* cannot be bound by *every boy*, but that is precisely what QR would predict in the LF (2):

(2) every boy$_j$ [his$_i$ father loves t$_j$]

The problem is solved by an ad-hoc restriction, saying that LF binding is possible only when there is surface binding. This basically implies that the LF cannot be too far away from the surface; the null hypothesis would of course be that there is no difference at all.

But even under this hypothesis, H&K still assume movement in order to solve the problem of quantifiers in object position, so they assume that *every boy* nonetheless moves at LF, but, as a consequence of the above mentioned restriction, *his father* also has to move:

(3) his father$_i$ every boy$_j$ t$_i$ loves t$_j$

AS shown by H&K, this kind of analysis must apply not only to objects of verbs, but to all other categories as well, as they all may express two place relations having a complement that cannot be interpreted in situ. This leads us to the second problem already mentioned above: incompatiblility with a sober syntactic analysis. As H&K argue, under their anaylsis, we need silent subject positions for PPs and DPs, but apart from QR there seems to be no independent evidence for such positions in syntax.[1]

Another problem, in my view, is that QR is a kind of restructuring operation which from the point of view of syntax should also have restructuring effects. It is well known that if we need to revise our assumptions about syntactic structure in order to make a sentence interpretable, this should have measurable disturbing effects on complexity of the task. However, no such effects can be observed with

[1]Montague's semantics for quantifiers in object position, namely type shifting accompanied with semantic reduction via a meaning postulate would have avoided the problem, and it is a pitty (and a shame) that this unproblematic solution is not even mentioned by H&K.

string vacuous QR; in consequnce, there seems to be something fundamentally wrong in the design of the theory.

1.2 Overview

This paper attemps to go two small steps towards a more realistic interface theory. It shows how to do away with QR and it attemps to interpret sentences from left to right incrementally. The sample language will be German, Dutch and Swiss German, these languages are known to have a context sensitive syntax and display an interesting mismatch between syntax and semantics in that c-command in a simple syntactic analysis does not always agree with the scope relations in semantics.

Another third step has been taken elsewhere in Sternefeld (2001) where I have shown how to design an in situ semantics for topicalized structures like

(4) [his_i father $]_j$ everyone adores_i t_j

The method used here in the present paper is basically the same, it builds on a slight and systematic type shifting operation.

One of the main motivations for H&K is QRs ability to account for scope ambiguities, and this will also be a topic of the present paper. Basically, two cases must be accounted for. One is the problem of wide scope of $\exists$ in $\forall\exists$, the second is the problem of wide scope of $\forall$ in $\exists\forall$. I consider these completely different and independent problems. A solution for the first problem will be presented in the next section; this is completely indepent of all other sections.

A solution for the second problem will

... to be continued

2 A semantics for scope independent interpretation

2.1 The formal system

It has ever been observed in the literature that theories of QR run into trouble when explaining the unrestricted wide scope possibility of existential quantification: Although QR is essentially clause bound when considering strong quantifiers like *every*, this does not hold for indefinite quantifiers like *a* or *some*. The wide scope reading of these quantifiers has usually been called the specific reading. Several attemps have been made to account for these readings, some in terms of choice functions, others in terms of pragmatic conditions on reference. Influential as these attemps have been, one problem still remaines unsolved, namely the existence of intermediate readings with the indefinite having wide scope with respect to some quantifiers but narrow scope with repect to others, cf. Kratzer (1995). In my view none of these previous attemps have successfully treated intermediate readings of existential quantification in a completely local way, without introducing hidden existential quantifiers (dubbed "existential closure") non-locally (and also non-compositionally) somewhere in the LF structure. Let us therefore look for a solution that is truely local in not adding unmotivated syntactic machinery to logical representations.

The main idea is that the phenomenen as such is not a matter of wide scope taking but rather a case of scope independent interpretation in the sense of Liu (1990), i.e. a specific *in situ* interpretation that might ignore the previous interpretation of other quantifiers. I briefly indicate how this can be implemented in a slight modification of a standard semantics for predicate logic, taking the text book semantics of Heim and Kratzer (1998) as a point of departure. The basic assumption of H&K is that assignments are partial, they are built up step by step each time when the interpretation encounters a new quantifier. So if g is an assignment and we encounter a quantification (Qx), then g+$\langle x, a\rangle$ is an extension of g which in addition assigns a to x. Let $g\backslash Y$ be that g' identical with g except for the possible difference that g' is not defined for the variables in Y; we thus define:

(5) a. $g + \alpha :=$ the concatenation of g with α
    b. $g \setminus Y := \{\langle\alpha, \beta\rangle \in g : \alpha \notin Y\}$

The main idea to interpret scope independence is boroughed from game theoretical semantics (cf. Saarinen (1979)). We try to adjust this framework to a more conservative framework with universal quantifiers being interpreted with respect to assignments g. Scope independence then means that the interpretation of a potentially scope dependent quantifier (here: the existential quantifier) is independent of a previous quantifier (∀y), so that the assignment of g for y is information that cannot be relevant for the interpretation of α. As we will be made precise below, this will be implemented by removing {y} from g, as defined in (5-b). We somehow have to formally identify the set Y of variables irrelevant for the interpretation of α, so we assume that (6-a) and (6-b) are well-formed expressions:

(6) a. $(\exists x \mid Y : A, B)$
    b. $(\forall x : A, B)$

A is the restriction of a quantifier and B its nuclear scope; the value assignment for x will be chosen independently from that of all variables in Y. Universal quantification is always scope independent, hence there is no Y in (6-b). The set

Y can be empty, we then simply write $(\exists x : A, B)$. For the moment, the choice of Y is free, but as will be seen below in connection with (7-d) there may be far reaching empirical restrictions.

As a result, existential quantification is always desambiguated with respect to scope (in-)dependency. Accordingly, the readings we get for (7) by indexing are those in (a)-(d):

(7) every man bought every girl a book

a. every$_x$ man bought every$_y$ girl a$_{z|\{x,y\}}$ book =
There is a book such that every man bought it for every girl (wide scope)

b. every$_x$ man bought every$_y$ girl a$_{z|\{y\}}$ book
for every man there is a book and he bought it for every girl (intermediate scope)

c. every$_x$ man bought every$_y$ girl a$_z$ book
for every man and girl there is a book he bought for her (narrow scope)

d. every$_x$ man bought every$_y$ girl a$_{z|\{x\}}$ book
for every girl there is a book such that every man bought it for her (intermediate scope)

I doubt that the reading in (7-d) exists, but as can be easily verified it is predicted by the truth conditions to be laid down below. In order to exclude such a reading, one would need additional constraints on the use of variables in Y, see below.

Let us now go on and see how the apparatus developed so far can be applied to the semantics of quantification. The next step is to define a function S that choses an individual depending on the assignments to individuals of the relevant previous quantifiers. The working of S can be compared to a choice function, but since the term is already taken, I resort to game theory, where such a function is called a *winning strategy*. So the next step is to define a winning strategy S:

(8) A winning strategy S is a two place function, taking a variable x and an assignment g as arguments and yielding a new assigment $g' = g + \langle x, a\rangle$ for some $a \in D$. Thus, $g' = S(x, g)$.

We can now define truth as follows:

(9) Φ is true iff there is a winning strategy S and the empty assignment function g that satisfy $[\![\Phi]\!]_{S,g}$.

Next, for the recursive part of the definition of satisfaction, most definitions are straightforward. As for universal quantification, nothing new happens, cf. (10-a), and the same is true for binary sentential connectives. The only interesting condition is (10-b):

(10) a. $[\![(\forall x : A, B)]\!]_{S,g}$ iff for all $a \in D$ such that $[\![A]\!]_{S,g+\langle x,a\rangle}$ it holds that $[\![B]\!]_{S,g+\langle x,a\rangle}$

b. $[\![(\exists x \mid Y : A, B)]\!]_{S,g}$ iff $[\![A]\!]_{S,S(x,g\backslash Y)}$ and $[\![B]\!]_{S,g+\langle x,a\rangle}$ where a is the value assigned to x by $S(x, g\backslash Y)$.

Intuitively, Y cuts away all the modifications of g induced by quantifiers Q that should not play a role for the evaluation of the existential. The reduced assignment $g' = S(x, g\backslash Y)$ is independent of Q and therefore the same for all Q.

Note that at this point we could not say that there is an extension of g' that satisfies A and B. In a series of published papers on specific readings, whose authorship I will not reveal, I indeed found such a formulation as part of a truth conditions that was intended to capture the specific reading. However, such a condition cannot work. Bringing in existential quantification here crucially does not yield an interpretation independently of the previous quantifiers (although it might be independent of the current assignment!), because we may now chose a different extension for each g because we previously introduced different individuals when interpreting universal quanification. This is because the structure of our conditions formulated in the meta-language is still $\forall \ldots \exists$, so we (unintendedly) still say that for every modified assignment such that such and such ... there is an extension such that .... This is precisely what we wanted to avoid! It is for this reason that we need existential quantification far outside, namely by existential quantification over S in the definition of truth in (9).

It remains to account for negation. Intuitively, an existential can never outscope negation, and this is precisely what has to be implemented. If nothing else would be said, existential quantification would always outscope negation, clearly the wrong result.[2] We therefore define negation syncategorematically as in (11):

(11) a. If Φ is an atomic formula, S, g satisfy $\neg\Phi$ iff g does not satisfy Φ

b. $\neg\neg\Phi = \Phi$

c. $\neg(A \wedge B) = \neg A \vee \neg B$

d. $\neg(A \rightarrow B) = (A \wedge \neg B)$

e. $\neg(\forall x : A, B) = (\exists x : A, \neg B)$

f. $\neg(\exists x \mid Y : A, B)$ is undefined or illformed if Y is nonempty, otherwise $\neg(\exists x : A, B) = (\forall x : A, \neg B)$

That's all as far as truth conditions are concerned.

[2] It is at this point where a radical proponent of game theory could object that we only half heartedly adopted the theory: the difficulty with negation could be resolved in pure game theory in a much more elegant way. My reply is that the present formulation might still have some pedagical virtue: I does not lead us to far away from the standard mechanism and in addition it mkaes quite clear what the problem with negation is (and why game theory is as it is).

All I wanted to show is that scope independence does make sense and can be formally implemented in a local way.

The one open problem is the restriction on Y: you may verify that the reading (7-d) is indeed generated by the truth conditions layed down above. But it could of course also be generated by QR. Within the present framework, the obvious condition would be one that restricts the set of variables in such a way that it can only cut away variable assignments in such a way that the result is a variable assignment that has actually been used previously as an assignment function. In other words, we may profit from the fact that a variable assignment g is actually a sequence of pairs, not merely a set, and that the later variables are in the syntactic scope of the earlier ones. We may then say that

(12) $g \backslash Y$ is defined only if $g = g_1 + g_2$ and the Domain of g_2 is Y (and none of Y is in the domain of g_1).

The empirical prediction made by (12) has to be tested against further examples; I will discuss some potential counterexamples in the next subsection. One consequence, however, is immediate: Consider the sequences $\forall_1\forall_2\exists_{3|1}\exists_{4|2}$ and $\forall_1\forall_2\exists_{3|2}\exists_{4|1}$. These consititute well-known cases of so-called branching quantification. It is an immediate consequence of (12) that such cases are ruled out. I consider this a virtue: The branching readings discussed in the literature are invented by logicians. I claim that even trained semanticists don't get them: the intended readings are far too complex, and what (12) does, is reduce complexity. This is as it should be.

2.2 Scope independence and intensional contexts

It is well-known that intensional contexts create scope. Taking metaphysical necessity as an example, (13-a) expresses a contingent but essential property of all actual men, (13-b) expresses a necessary property of the species.

(13) a. $\forall x(\mathbf{man'}(x) \rightarrow \Box\mathbf{snore'}(x))$
b. $\Box\forall x(\mathbf{man'}(x) \rightarrow \mathbf{snore'}(x))$

It is also assumed that propositional attitude reports can be ambiguous between two readings; if John knows that students a, b, and c got grade A, and if a, b, and c are the ISCL-students in the class, then it is assumed that this can be reported as

(14) John knows that the/all ISCL-students got grade A

even if John is not aware of the fact that a, b, and c are ISCL-students. This is a description chosen by the speaker, not by John.

Logicians who first discovered the ambiguity, treated it as a matter of scope, so the question arises whether this is a case for QR. Linguists have pointed out that the different readings, often associated with de re and de dicto, can also be generated without invoking structural transformations; cf. Kamp (1971) for an early reference. When translating intensional logic into extensional two type logic, the formula in (14) transform into:

(15) a. $\forall x(\textbf{man'}(x, w) \rightarrow \forall w' \textbf{snore'}(x, w'))$
b. $\forall w' \forall x(\textbf{man'}(x, w') \rightarrow \textbf{snore'}(x, w'))$

But now, consider:

(16) $\forall w' \forall x(\textbf{man'}(x, w) \rightarrow \textbf{snore'}(x, w'))$

Under certain standard assumptions, namely that quantification is always over possible individuals (which means that the Barcan Formula and its converse are valid) it turns out that (16) and (15-a) are logically equivalent.

But this means that we have found a way to represent the ambiguity without invoking syntactic transformations. This is good, but probably not good enough. Let us look at a more complicated example.

Assume we have a surface order $\forall_1 \forall_2 \exists$ with $\forall_1$ quantifying over possible worlds (something like (17-a)) and assume that $\forall_2 \exists$ should be understood de re, giving rise to the reading $\forall_2 \exists \forall_1$ in (17-b):

(17) a. John knows$_1$ that every$_2$ man loves a woman
b. for every man there is a woman such that John knows that he loves her

I am not sure that such a reading really exists, but for the sake of the argument, let's try to represent the reading in a way that is consistent with the no movement hypothesis. Giving both quantifiers a "loaded" interpretation, and interpreting the attitude like a modal, we get something like (18-a) which we know is equivalent to (18-b):

(18) a. $\forall w' \forall x(\textbf{man'}(x, w) \rightarrow \exists y(\textbf{woman'}(y, w) \wedge \textbf{loves'}(x, y, w')))$
b. $\forall x(\textbf{man'}(x, w) \rightarrow \forall w' \exists y(\textbf{woman'}(y, w) \wedge \textbf{loves'}(x, y, w')))$

The crucial point now is whether (18-b) is equivalent to (19):

(19) $\forall x(\textbf{man'}(x, w) \rightarrow \exists y(\textbf{woman'}(y, w) \wedge \forall w' \textbf{loves'}(x, y, w')))$

(18) seems to permit that we chose different women for each w in order to satisfy **loves'**(x, y, w'), but among the different choices there must also be one for the actual world, and this will make (18) true. Since the direction from (18) to (19) is trivial, this establishes the equivalence.

However, we get a different outcome when we restrict the quantification in such a way that the actual world is not in the domain of $\forall w'$. Then the above argument goes no longer through and we get indeed different truth conditions. So for instance, if we replace knowing with believing, we get the following reading For every man x John holds a believe that x loves someone, whithout having some particular individual in mind and without knowing that this someone is a woman.

This implies that we cannot get the intended fully transparent reading (if real at all) by providing quantifiers with a loaded reading. And QR is not an option, raising the existential quantifier would also force the universal quantifier to raise, but QRing $\forall$ should be clause-bound. In this situation, scope independence is a way out: assuming that $\exists$ gets an interpretation that is scope independent from $\forall w'$ does the job. In order to see how this works, consider the two formulas in (20):

(20) a. $\forall w' \exists y \mid w'(\textbf{woman'}(y, w) \wedge \textbf{loves'}(x, y, w'))$
b. $\forall w' \exists y \mid w'(\textbf{woman'}(y, w') \wedge \textbf{loves'}(x, y, w'))$

Assume that we start evaluation with a non-empty assignment that assigns the actual world to the variable w. Then (20-a) is correctly interpreted in such a way that some woman gets wide scope over $\forall w'$. Consider next (20-b). Here the restriction of $\exists$ cannot be interpreted because we just removed the value for w' from the assignments. Hence only one interpretation is possible, and this is the intended one.

On the other hand, we now get into conflict with (12). One modification would be to say that the restriction holds only for variables of type e. That is, we can simply ignore the world variable in Y:

(21) $g \backslash Y$ is defined only if the following holds:
If x is of type α, $x \in Y$ and x is the domain of the n-th pair in g, then for any subsequent y of the same type α in the domain of g, $y \in Y$.

Note that scope independence is not allowed to skip an intensional opeator: by analogy to (7-d), if the first two quantifiers would be modal operators, modal independence of one might reverse quantifier scope as in the example. This should never happen, because the scope of operators is clearly fixed and determined by the surface. Nachrechnen...

2.3 QR without QR: a problem

In this section I would like to show that the system proposed above is not stronger than ordinary QR, that is: the readings we can get in it are often, though not always equivalent to those using a limited sort of QR, i.e. one that is not allowed to move across negation.

More precisely, the question we are asking is this: if there is a formula that contains a quantifier with a scope independent interpretation, is there an equivalent formula with no scope independent interpretation which can be obtained from the original structure by QR?

In simple cases, the answer seems to be Yes. Consider the following example.

(22) a. every man who loves a woman deams

b. $(\forall x : \mathbf{man}(x) \wedge (\exists y \backslash \{x\} : \mathbf{woman}(y), \mathbf{love}(x,y)), \mathbf{dream}(x))$

(22-b) represents the scope independent reading. This can be obtained by QRing the existential as shown in (23):

(23) $(\exists y : \mathbf{woman}(y), (\forall x : \mathbf{man}(x) \wedge \mathbf{love}(x,y), \mathbf{dream}(x))$

(23) is obtained from (22-b) by moving the existential and its restriction over the minimal scope of x and leaving the scope of the existential in situ. These formulas should be equivalent, but in fact they are not. The problem comes in when the restriction of the universal quantifier cannot be satisfied. In this case there are no men or no men who love a woman. Now (23) still asserts the existence of a woman, since the scope of the existential is trivially true, due to the emptyness of the domain of *every*. By contrast, (22) does not assert the existence of a woman, because the whole sentence is true simply by the fact that restiction cannot be satisfied. This does not imply the existence of a woman.

Of course, we can get the equivalence under a presuppositional semantics of the universal quantifer, but this does not solve the problem in general. We only have to look at formulas like

(24) $A \rightarrow (\exists x \mid Y : B, C)$

to see that as soon as A is false we cannot assert the existence of a B. This is of course not perculiar to the present implementation but is a common feature of similar accounts that we considered less adequate than the present one: eg. it also holds for the choice function approach, which basically assert the existence of a choice function f such that, in the same syntactic context, we get (25) as equivalent to (24)

(25) $A \rightarrow C(f(B))$

Here again, due to the fact that (25) is true as soon as A is false, the formula does not assert the existence of a B.

I do not believe that this outcome is devastating, but before commenting further on it it might be worth looking into the literature on choice function to see whether the problem has been noticed at all, and how it has been handled, if at all.

This is future work to be done.

2.4 Villalta on Wide Scope Plurals

It seems to be a common place among semanticists that questions like (26) are ambiguous between a number of possible readings:

(26) How many dogs did everyone feed?

This ambiguity, to be explained in a minute, has been taken as a starting point for one of the very few attemps to design a model of incremental semantic processing; this endeavour has been undertaken by Elisabeth Villalta (2003, 2006) in her dissertation which has been judged by the referees as a mile stone in procedural semantics and it is for this reason that I will discuss the issue here, arriving at conclusions very different from that of the oppinio commonis.

Now for an explanation of the ambiguity, consider the following story told by Villalta p. 2:

(27) In the music department, three trumpet students had to pass an exam last week. Every student had to play six pieces. The only requirement they had was that among these there were two pieces that ecerybody had to play [...] For the rest, the students were free to choose what they preferred.

In the context of the situation described above, the question described above, the question [...] can be answered truthfully in two possible ways [...]

(28) How many pieces did every student have to play at the exam?
Possible true answers:
 a. Six pieces
 b. Two pieces

These answers follow systematically from the standard theory of Logical form under the assumption that the set of pieces which is pied piped when questioning the number can either be reconstructed into the position of the trace (answer a.) or be interpreted more or less in situ, i.e. in a position that c-commands the universal quantifier (answer b.). Accordingly, the logical forms for these readings are assumed to be:

(29) a. For which number $n : \forall x : \mathbf{student}(x) \rightarrow \exists Y : \mathbf{pieces}(Y) \wedge card(Y) = n \wedge \mathbf{played}(Y)(x)$

b. For which number $n : \exists Y : \mathbf{pieces}(Y) \wedge card(Y) = n \wedge \forall x : \mathbf{student}(x) \rightarrow \mathbf{played}(Y)(x)$

These LF are standard in the semantics of questions, and repeated in every discussion on reconstruction and pied piping. However, in my view there is something fundamentally wrong in all the existing literature on the subject.

Consider:

(30) Wieviele Hunde hat jeder von euch gefüttert?

Given a situation in which A feeded a, b, and c, B feeded a, f, and d, and C feeded e and f. Then one possible answer would describe just this situation or model, and this is normally termed the paired list reading of the question. It is often described as quantifying *jeder von uns* into the question; cf. also Krifka (2001). I will not be concerned with this kind of answer.

Another possible answer would be the number 2, since this is the minimal number of dogs being fed by each of us. This corresponds to the LF where *dogs* has been reconstructed under the scope of *every*. However, on the alternative reading with *every* having narrow scope, a possible answer should be none, and this, so I content, does not correspond to a reading the question could have. This shows that there is something wrong in the standard theory adopted by Beck (1996), Stechow (1993, 1991) [check!!!], and many others.

Let us look at the nature of the problem more closely; the problem can be identified independently from pied piping and the theory of questions.

(31) 5 Hunde hat jeder von uns gefüttert

Is (31) ambiguous? The one reading which expresses that 5 is the minimum number of dogs being fed by anyone of us seems to be unproblematic. The more problematic reading seems to be the one with the dogs having wide scope over *every*. This also expresses that 5 is the minimum number of dogs being fed by anyone of us, but in addition this set of dogs must be the same for each of us. But is that a particular reading or only a special case of the other reading?

In another context, namely that of "every man loves a woman", this is an old problem that raises its ugly head in almost any empirical study of scope and possible readings. And still there is no received wisdom one could refer to as a standard solution. Uttered out of the blue, I have a fairly strong intuition that this is not a possible reading of (31). We can test further the case by

(32) Keinen Hund hat jeder von uns gefüttert

in a situation where there the intersection of dogs feeded by each of us is empty. Is an uttarance like (32) felicitous and true in such a situation?

I guess that the issue has never been tested empirically, but I would conjecture that there is a vast majority that rejects (32) in such a situation. If so, this tells against the standard theory of QR. QR is a movement operation which creates wide scope, so one would expect that, by the principle of Reversed Effability stated in (33),

(33) No LF-rule can perform a job which cannot be effectively performed in syntax in some human language. (cf. ?, p. 99)

the wide scope reading should also be possible as a result of S-structural movement. And by the principle of surface interpretability, the wide scope reading should even be the preferred one, contrary to fact.

Returning now to the pieces to be played at the exam, all reviewers of the dissertation cite the crucial sentence as

(34) How many pieces did every student play at the exam?

According to what we have said above, the corresponding German sentence

(35) Wieviel Stücke hat jeder gespielt?

cannot have the answer two. Now, this is in fact not what Villalta finds, and it is clear that she accepts the common view that (35) is ambiguous. What Villalta states is not that the purported reading does not exist, but that the reading is dispreferred. In fact, when comparing the original example (28) with (35), one may detect a subtle but important difference that has been overlooked. In the original example, there is an additional modal component that may contribute to a scope ambiguity that is independent of the potential ambiguity discussed so far. Consider (36):

(36) 2 Stücke müssen gespielt werden, die Wahl der übrigen Stücke ist frei.

Putting stress in the modal makes (36) unambiguous, and the rest of (36) is explanatory, but redundant. That two pieces are obligatory of course implies that every student has to play them, but this is surely independent of the meaning and scope of *every student*.

[to be continued]

3 Argument Structure and Theta Indexing

[Bla ... Overview, motivation; a brief discussion that coherent constructions have received a lot of attention in almost anyone's framework, see e.g. Bouma (2003) for HPSG; Kaplan and Zaenen (2003) for LFG; Kempen and Harbusch (2003) for Performance Grammar; Rambow (2003) for Tree Adjunction Grammar. Transformational (syntactic) analyses still prevail, but none of them is deep enough to deal with aspects of semantic interpretation in a satisfactory way ...]

3.1 SOV order and auxiliaries

Like many other languages, German is SOV in basic order. In other words: Theta role assignment by a predicate and its projections (the discharge of theta roles from a theta grid) always targets a left sister node.

(37) DP_1 DP_2 ... V   (Subject, Object, Verb)

Theta role assignment will be essential for semantic interpretation, as will become clear below.

A few syntactic facts will also be helpful for a better understanding of the syntactic constructions we are going to analyse. In German and related languages to be discussed below, the subject agrees with the finite verb V or auxiliary AUX with respect to person and number. In what follows, this fact per se is irrelevant, but on the other hand, identifying the subject and the finite verb will turn out essential for analysing coherent structures, and for this reason it will facilitate our understanding of a construction if the subject and the agreeing verb are highlighted, which, by notational convention, will be achieved by printing the agreeing items in a font without serifes, as shown in (38):

(38) a. DP DP ... V
b. DP DP ... V AUX
c. DP DP ... V AUX AUX

As in English, auxiliaries determine (govern, select) the morphology of their verbal complements (participles, bare infinitives, infinitives with *zu* = 'to'), but contrary to English the direction of government in German is usually to the left, as one would expect in a SOV language. Concerning Case marking, government and agreement, we adopt the following notational conventions, again only for the purpose of increasing the readability of the examples:

— A DP without serifes (the subject) always bears nominative Case; dative and genitive DPs are indexed by *D* and *G* respectively on the head of the DP; all remaining DPs are accusative DPs without an index.

— A verb without serifes bears tense and the agreement features of the subject. Participles are indexed by *P*; all remaining verbs are infinitives with or without *zu* (='to').

— A verb with index n governs the morphology of a verb with the index $n+1$. The higher the index, the deeper the embedding:

(39) a. DP DP ... V_1
b. DP DP ... V_2 AUX_1
c. DP DP ... V_3 AUX_2 AUX_1
d. ...

These conventions are exemplified in (40):

(40)

der Mann	dem_D	Hund	einen	Knochen	$gegeben_P$	haben	muss
the man	the	dog	a	bone	given	have	must
					V_1	AUX_2	AUX_3

For the considerations to follow the indexing of the verbs is crucial, that is, the ordering of auxiliaries and main verbs is essential for the complexity of the resulting language. Before turning to this topic in section ??, a brief remark on main clauses in German is in order.

3.2 CPs and movement into C and SpecC

As usual, sentences are CPs, but contrary to pure SOV-languages like Korean or Japanese, the complementizer C of embedded clauses (*dass* = 'that') is left-peripheral:

(41) [$_{CP}$ (SpecC) C (=dass) [DP ... V/AUX]]

Subordinate CPs behave mostly as their English counterparts (complements, relative clauses, indirect questions etc. can all be analyzed as CPs very much like in English). In main clauses, however, the finite verb moves into the C-position:

(42) a. [$_{CP}$ V/C DP ... trace]
b. [$_{CP}$ AUX/C DP ... trace]

As in English, such verb first clauses are mostly interpreted as yes/no-questions.

A different interpretation is triggered by moving into SpecC. After having moved the finite verb into C, any major constituent can move into SpecC, yielding a verb second clause. Such V/2 clauses are interpreted as declarative sentences.

In what follows, I will totally ignore questions of clausal typing, and I discuss declarative main clauses only very briefly. What is crucual here is movement into SpecC and C. The latter is head movement, and it is normally assumed that head movement must be reconstructed in order to be interpretable. I do not think that this must be taken for granted. An in situ interpretation would in many cases be possible, if the verb that has been moved into C is interpreted in a Neo-Davidsonian style as a property of an event. Although I still believe that this kind of interpretation is perhaps to be preferred from the view of immediate left to right processing of semantic information, Neo-Davidsonism requires a completely unorthodox treatment of scope and quantification and I therefore refrain from going into the details of such a theory.

Rather, I would like to follow the main stream in assuming that movement to C has to be reconstructed. Reconstruction can be either syntactic or semantic, and I'm not aware of any argument that could distinguist between the two methods. As concerns movement of an argument to SpecC, two possibilities arise: either the argument is interpreted in situ, or it is reconstructed into its base position. Both possibilities must be available as they may lead to different scopal properties of the moved item: Crossing of another scope inducing operator will induce a scope ambiguity. This ambiguity can be resolved by either in situ interpretation or reconstruction. Again, as I have shown elsewhere (and see also below), reconstruction can be modeled within semantics by lambda-conversion. So there is no obstacle for an in situ interpretation.

So much for main clauses; in what follows, I will focus on the basic SOV(AUX*)-ordering of subordinate sentences.

4 Semantic interpretation of Theta Indexing

4.1 Argument Structure

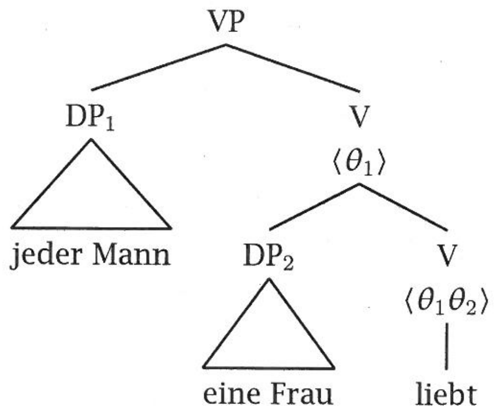
The aim of this paper is to develop an explicit surface semantic account of a construction type in German, Dutch, and Swiss German that has,due to a mismatch between syntax and semantics, been particularly difficult to deal with. As a first step towards that end I will look at the simplest construction types like simple transitive clauses and develop a semantic theory that will turn out useful for the treatment of the more complex cases. I will start with the analysis of (43):

(43) [$_{VP}$ eine Frau jeden Mann liebt]
a woman every man loves

I will assume that the verb *loves* has two theta roles to discharge, these are assembled in a theta grid. Theta grids are indexed sequences of theta roles. The identity of the theta role itself (whether it is an agent, a causer etc.) plays a role in linking theory which determines the sequence in the grid. Once this is done, the kind of theta role is irrelevant in syntax; all that counts is the order in the sequence. The theta roles themselves will then only be distinguished by indeces; this is what Dowty (1991) has called theta-indexing. The indeces themselves can be chosen abitrarily from a given array, provided that each theta role gets a new index. Thus, a verb like *love* has a theta grid $\theta\theta'$; taking it from an "enumeration" or out of the lexicon for the purpose of building up a syntactic tree implies that we chose indeces for the theta roles. By chosing 1 and 2 as indeces, we turn $\theta\theta'$ into $\theta_1\theta_2$. As we will see below, this numerical indexing will also be given a semantic interpretation.

Now, given an indexed theta grid, the individual theta roles will be projected and discharged, as shown in (44):

(44)



Discharging is traditionally encoded as an assignment of an index to the DP, within checking theory this can be re-interpreted as the checking of an already existing DP-index which can be implemented as a simple feature on the DP. This way the index of the DP and the index of its theta role must be identical and that's all that is needed for the semantic interpretation.

Now, the key to a solution of the more complex cases (to be discussed only later) is the following assumption: predicates are not Schönfinkeled, as in Montague's writings and in over many textbooks and as in Heim and Kratzer (1998), but are represented as open formulas, containing "“pointers” that are syntactically represented as the indices of theta-roles. The theta index of a DP has the function of telling the DP which argument index in the open proposition is to be bound. It is also called its binding index.

In order to develop this into a fully compositional semantics, a number of measures have to be taken. Let us start with the traditional (non-compositional) concept of binding into open formulas. Given an open formula $P(x_1, x_2)$ the index of the variable corresponds to the theta index. This means that a DP_i takes as an argument the open proposition and binds the variable with the index i, as shown in (45):

(45) a. to love = $\mathbf{love'}(x_1, x_2)$
b. to love [every woman]$_2$ = every woman $\lambda x_2.\mathbf{love'}(x_1, x_2)$ = $\forall x(\mathbf{woman'}(x) \rightarrow \mathbf{love'}(x_1, x)$
c. John$_1$ loves every woman = $\mathbf{John'}$ $\lambda x_1 \forall x(\mathbf{woman'}(x) \rightarrow \mathbf{love'}(x_1, x)$ = $\forall x(\mathbf{woman'}(x) \rightarrow \mathbf{love'}(\mathbf{John'}, x)$

This can be done in a completely compositional way if we treat open propositions as sets of assignment functions. Assignment are functions from variables to individuals; in order to get a compositional semantics we implement these as functions from numbers into individuals. This means that (45-a) now translates into:

(46) $\lambda g.\mathbf{love'}(g(1), g(2))$

It is important that the assignments are part of the object language, and the only thing we would have to do now in order to provide the usual semantics is to translate the classical techniques of variable binding via modified assignments into the object language. This program has been carried out in Bennett (1979), Sternefeld (2001), and Sternefeld (2006*b*); for lack of space I refer the reader to these works.

One obvious advantage is that quantification now becomes completely compositional, as is well-known in algebraic logic (s. Tarski and Vaught (1957) and the overview in Hodges (1998)). An expression like $\mathbf{love}(x_2, x_1)$ no longer denotes a truth value as in classical logic (which is the source for non-compositionality), but the set of assignments which satisfy the open proposition. Formally, assignments are functions from natural numbers into individuals, so that $\mathbf{love}(x_2, x_1)$ is interpreted as the set of assignments that satisfy the relation R = **love**, which is the set $\{g \in D^N : \langle g(2), g(1)\rangle \in R\}$, with "“love” as the meaning of R.

Expressions of the form $g(n)$ will be called **pseudo-variables.** There is an obvious correspondence between pseudo-variables and ordinary variables x_n, but there is also a crucial contrast: pseudo variables do not yield open propositions in the traditional sense of the term; this is so because assignments are always bound as in (46), which in fact contains no free variable whatsoever. On the other hand, as in the classical framework, (46) shares with open proposition the property that its interpretation still depends on the choice of indexes. That is, (46) denotes a different object than (47):

(47) $\lambda g.\mathbf{love}(g(4), g(5))$

(46) and (47) may be called pseudo-meanings; in order to get to the real meaning we should free ourselves from the particular indexes or pointers. There are several ways of doing so, cf. in particular Kracht (2007); for simplicity, I will assume here that the real meaning is something like a function from pairs of indeces into open propositions such that for each pair of indeces n,m we assign the functions that yield the pairs of things a for n and b for m such that a loves b:

(48) $\lambda\langle n, m\rangle\lambda g.\mathbf{love}(g(n), g(m))$

This means effectively that the syntactic indexing of theta grids provide arguments for (48); applying (48) to the indexes of the theta grid yields a pseudo-meaning.

The next step is to describe the binding of pseudo-variables by a quantifier. The result of binding a variable with index (or pointer) 1 by a universal quantifier should be something like (49) (allowing for alphabetic variants for x).

(49) $\lambda g.\forall x.\mathbf{love}(g(1), x)$

The question is how this can be achieved in a compositional way. The answer is that we simply translate the truth conditions we already know from the meta-language into the object language and see how these can be expressed as formulas in the object language. This way we do not add any complexity, but simply map the complexity of the meta-language into the object language. As everyone knows, part of the meta-linguistic description of binding involves modified assignments: an assigment $g[n/x]$ is the same as g apart from the possible difference that $g(n) = x$. This can be defined directly in the object language as shown in (50), where α and β are variables for assignments and ι (the jota-operator) reads as the one and only α such that...

(50) $\beta[k/u] := \iota\alpha((\alpha(k) = u) \wedge \forall n(n \neq k \rightarrow \alpha(n) = \beta(n))$[3]

Translating the truth conditions for quantification from the meta-language into the object language is easy and allows us to derive the following equivalences:

(51) $\lambda g.\forall x.\mathbf{love}(g(1), x)$
$= \lambda g.\forall x.\mathbf{love}(g(1), g[2/x](1))$
$= \lambda g.\forall x.\mathbf{love}(g[2/x](1), g[2/x](2))$
$= \lambda g.\forall x[\lambda g.\mathbf{love}(g(1), g(2))](g[2/x])$

[3]See Bennett (1974), where assignments are functions of type $\langle n, e\rangle$ from numbers into individuals. A more general theory assumes that assignments can have an arbitrary range $\langle n, y\rangle$; see Sternefeld (2001) for such a generalization.

The second line follows from the equivalence of x and $g[n/x](n)$; the third line follows from (50) and the fact that $g[2/x](1) = g(1)$. The forth line is Lambda-exportation. From the last line we can extract the compositional meaning of the universal quantifier; assuming that the pseude-variable to be bound is x_2, we get:

(52) $(\forall x_2) = \lambda p.\lambda g.\forall x.p(g[2/x])$

In a sorted universe with n as the logical type of natural numbers, p has the type $\langle\langle n, e\rangle, t\rangle$. Applying (52) to $\lambda g.\mathbf{love}(g(1), g(2))$ yields (51). Accordingly, the general meaning of $\forall$ is independent of an index, the index is supplied by the theta indexing and an argument of (53) (with n being a variable of type n):

(53) $\forall = \lambda n.\lambda p.\lambda g.\forall x.p(g[n/x])$

It is now easy to see how to build in the restriction of a quantifier; I will not go into details (cf. section 10), the result is of course something like:

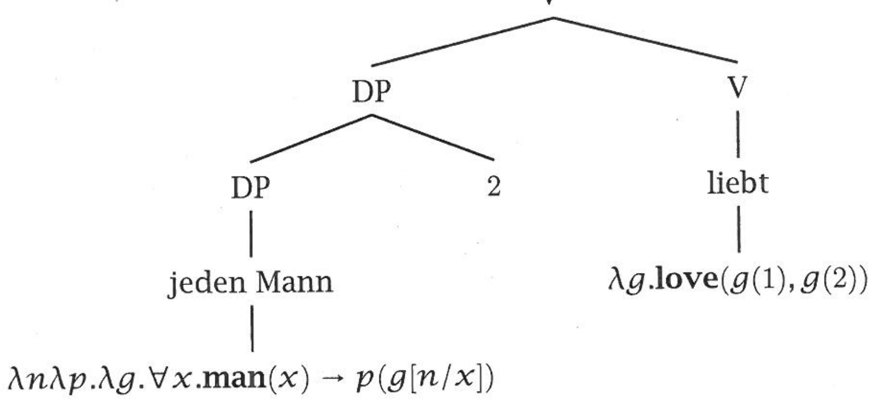
(54) $\lambda n.\lambda p.\lambda g.\forall x.\mathbf{man}(x) \rightarrow p(g[n/x])$

In what follows, I assume that two-place operators like $\rightarrow$ are "stronger" than the quantifiers in the sense that (54) abbreviates (55).

(55) $\lambda n.\lambda p.\lambda g.\forall x(\mathbf{man}(x) \rightarrow p(g[n/x]))$

Now we are in a position to interpret a sentence like *(dass) eine Frau jeden Mann liebt* in a completely compositional way. The only additional device which is not strictly lexical, is theta-indexing. Let us start with (56) and assume that the theta-index (the binding index) has been adjoined to the DP:

(56)



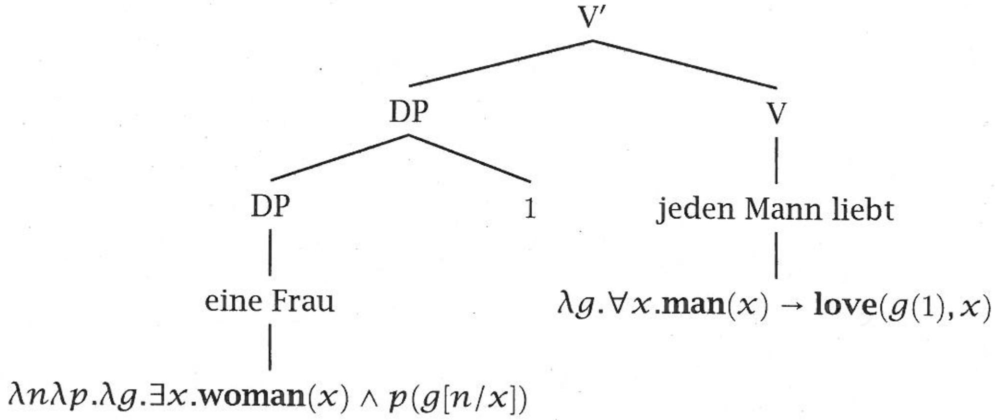
By lambda conversion we successively reach the result in (57-f):

(57) a. $\lambda n.\lambda p.\lambda g.\forall x.\mathbf{man}(x) \rightarrow p(g[n/x])(2)(\lambda g.\mathbf{love}(g(1), g(2)))$
b. $= \lambda p.\lambda g.\forall x.\mathbf{man}(x) \rightarrow p(g[2/x])(\lambda g.\mathbf{love}(g(1), g(2)))$

c. $= \lambda p.\lambda g.\forall x.\mathbf{man}(x) \rightarrow p(g[2/x])(\lambda g.\mathbf{love}(g(1), g(2)))$
d. $= \lambda g.\forall x.\mathbf{man}(x) \rightarrow \lambda g.\mathbf{love}(g(1), g(2))(g[2/x])$
e. $= \lambda g.\forall x.\mathbf{man}(x) \rightarrow \mathbf{love}(g[2/x](1), g[2/x](2))$
f. $= \lambda g.\forall x.\mathbf{man}(x) \rightarrow \mathbf{love}(g(1), x)$

In the same manner, we interpret the subject, which yields the scoping that corresponds to c-command in surface structure.

(58)



(59) a. $\lambda p.\lambda g.\exists x.\mathbf{woman}(x) \wedge p(g[1/x])(\lambda g.\forall x.\mathbf{man}(x) \rightarrow \mathbf{love}(g(1), x))$
b. $= \lambda p.\lambda g.\exists x.\mathbf{woman}(x) \wedge p(g[1/x])\lambda g.\forall y.\mathbf{man}(y) \rightarrow \mathbf{love}(g(1), y)$
c. $= \lambda g.\exists x.\mathbf{woman}(x) \wedge \lambda g.\forall y.\mathbf{man}(y) \rightarrow \mathbf{love}(g(1), y)(g[1/x])$
d. $= \lambda g.\exists x.\mathbf{woman}(x) \wedge \forall y.\mathbf{man}(y) \rightarrow \mathbf{love}(g[1/x](1), y)$
e. $= \lambda g.\exists x.\mathbf{woman}(x) \wedge \forall y.\mathbf{man}(y) \rightarrow \mathbf{love}(x, y)$

One might think that there should also be the reversed scope reading, with the object having wide scope over the subject. I claim that in German this reading does not exist, it is an invention of logicians. I will come back to the point in section 9.

4.2 Further consequences

For the moment, let us look at a number of consequences of this kind of analysis. One concerns the position of negation. Above, in our discussion of coherence, we presupposed that negation is an adjunct that can adjoin to any projection of V, thereby taking scope (and possibly also focus) at the surface position. This flexibility of scope taking has always been a problem for traditional approaches, whereas in the present theory the scoping possibilities are still consistent with having only one type of negation: since all V-projections are "open propositions", the logical type of negation can be kept constant, regardless of the position it takes in the structure. I consider this a major advantage in comparison with the traditional Schönfinkeled types which would not allow for a uniform treatment of negation.

Another welcome consequence is that we do not need QR for the purpose of variable binding. In traditional frameworks, there is no direct way to interpret sentences like (60) directly without additional operations.

(60) everyone$_i$ adores himself$_i$

Suppose that the standard meaning of *everyone*$_i$ is represented as in (61-a), and assume that *adore himself*$_i$ is translated as in (61-b):

(61) a. $\lambda P.\forall x_i(\mathbf{human}(x_i) \rightarrow P(x_i))$
b. $\lambda y.\mathbf{adore}(y, x_i)$

Although the binding variable in (61-a) and the free variable in (61-b) are coindexed, there is no way of taking advantage of this coindexation, because lambda conversion is not allowed: Applying (a) to (b) and converting into

(62) $\forall x_i(\mathbf{human}(x_i) \rightarrow \mathbf{adore}(x_i, x_i))$

is blocked by the fact that conversion would bring a free variable into the scope of a binder. But now think of these formulas as abreviations for the corresponding formulas that use pseudo-variables instead. Then conversion is allowed, because there are no free variables that could be bound by conversion. To be more precise, let us start with the assumption that pronouns are translated as pseudo-variables. A pronoun with n as its so-called referential index naturally corresponds to the pseudo-variable $\lambda g.g(n)$. By type shifting we must assure that the pronoun gets the logical type of a DP. Furthermore, we have to account for theta indexing which means that the pronoun must know which argument position it will replace within a theta grid. Assuming that the theta index of a pronominal DP is i, then DP$_i$ translates as (63):

(63) $\lambda p \lambda g.p(g[i/g(n)])$

Without the index, the general format of a pronoun is of course:

(64) $\lambda j.\lambda p.\lambda g.p(g[j/g(n)])$

Now, applying (63) to the open proposition (65-a) yields (65-b) which finally converts into (65-e):

(65) a. $\lambda g.\mathbf{adore}(g(n), g(i))$
b. $\lambda p.\lambda g.p(g[i/g(n)])(\lambda g.\mathbf{adore}(g(n), g(i)))$
c. $\lambda g.(\lambda g.\mathbf{adore}(g(n), g(i)))(g[i/g(n)])$
d. $\lambda g.\mathbf{adore}(g[i/n](n), g[i/g(n)](i))$

e. $\lambda g.\mathbf{adore}(g(n), g(n))$

But this is already the reflexive meaning we wanted to get. The fact that variable binding can be achieved without QR is a major advantage of the system; since QR massively overgenerates, be get rid of an unsolved problem if we can do without QR altogether. In fact, this is what I claim to be the case in the grammar of German.

Heim and Kratzer in their textbooks cite two further arguments for QR: Inversed linking and antecedent contained deletion. The latter does not exist in German, and I will return to the former further below.

In Sternefeld (2001) I have shown that the system can also cope with reconstruction in a purely semantic way. So examples like (66) can be interpreted directly at the surface, without undoing movement at LF.

(66) a. Himself$_i$ everyone adores t$_i$
b. dass sich$_i$ jeder t$_i$ rasiert
that himself everyone shaves
c. [Seine$_i$ Mutter]$_j$ liebt jeder$_i$ t$_j$
his mother loves everyone
'everyone loves his mother'

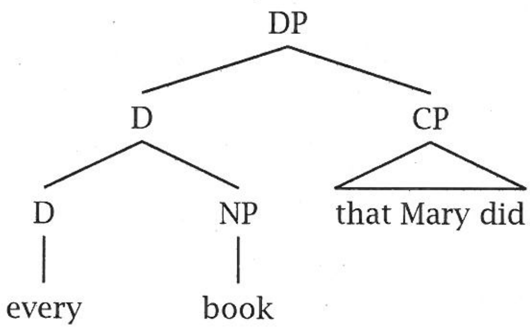
The reason for this is again that the material to be reconstructed does not contain real variables, but only pseudo-variables. Therefore reconstruction can simply be achieved by lambda conversion.

Heim and Kratzer's textbook gives only two further motivations for QR, namely inversed linking and antecedent contained deletion. As for the latter, consider their standard example:

(67) Bill read every book that Mary did

Let us first look at the quantified DP. As I have argued elsewhere, the syntactic structure of DPs should, for totally independent syntactic and phonological reasons, be as shown in (68):

(68)



This means that the restrictions of the quantifier are arguments of the D which can take any number of arguments that restrict the domain of the quantifier. Let us not think of these restrictions as of open propositions. Given that the DP has a binding index j, the open proposition that restrict the quantifier must have the same index as a pseudo variable; in traditional notation the first restriction is $\mathbf{book}(x_j)$, and the second restriction is something like $\mathbf{Mary\text{-}read}(x_j)$. Two issues arise at this point: one concerns the compositional combination, I will address this in section 10. The second issue is indexing of predicates: it is obvious that the usual VP-elipsis theories requires identity of indexing, contrary to what we assumed above, namely that the indeces are chosen diffently in the numeration.

The isssue is complicated and has many repercussions already in the traditional QR-theory of ellipsis. Let ús assume here that certain violations of the new index requirement must be tolerated. Assuming so, VP-deletion seems to imply that the VP in *Mary did VP* is represented as

(69) $\text{Mary}_i\ \text{read}(x_i,x_j) = \text{Mary}\ \lambda x.\text{read}(x,x_j)$

where i is the binding index of *Mary*. Likewise, in order to get identity of indexes, *Bill* must also have the binding index i, so that we get

(70) $\text{Bill}_i\ [\ \text{read}(x_i,x_j)\ [_{DP}\ \text{every}\ x_j\text{: book}(x_j)\ \text{and Mary}_i\ \text{read}(x_i,x_j)\ ]]$

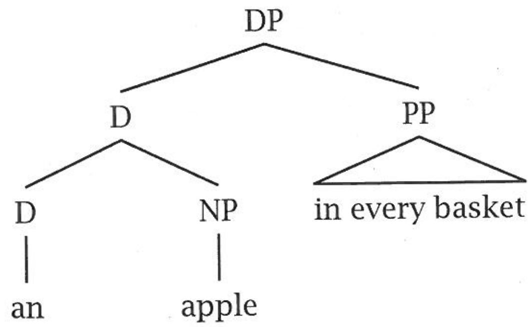
It should by now be obvious how (70) can be interpreted by applying the DP-object to the open proposition and that the identity of the predicats should not be a problem at all. So there is not motivation for movement in these constructions.

Let us finally consider inversed linking:

(71) An apple in every basket is rotten

The problem here is that *every basket* must gain wide scope without QR. We now apply techniques that have been developed in continuation semantics, cf. Barker (2002). Again, the syntactic structure is as in (67):

(72)



Now, *in* is a two-place relation which by our conventions can be represented as $\lambda g \mathbf{in}(g(1), g(2))$. By type-shifting, which is the usual way to handle continuations, we get (73):

(73) $[\![\text{in}_{12}]\!] = \lambda\overline{p}.\overline{p}(\lambda g.\mathbf{in}(g(1), g(2)))$

The next entity we encounter is the complement *every basket*$_2$:

(74) $\lambda p \lambda g \forall x.\mathbf{basket}(x) \rightarrow p(g[x/2])$

At this point there are two ways to combine (74) with (73). One results in (75-a) with the composition rule (75-b):

(75) a. $\lambda\overline{p}.\overline{p}(\lambda g \forall x.\mathbf{basket}(x) \rightarrow \mathbf{in}(g(1), x))$
b. $\lambda\overline{p}.\overline{p}.[\![in]\!]([\![NP]\!])$

The second one results in (76-a) with the composition rule (76-b):

(76) a. $\lambda\overline{p}\lambda g \forall x.\mathbf{basket}(x) \rightarrow \overline{p}(\lambda g'.\mathbf{in}(g'(1), g'(2)))(g[x/2])$
b. $\lambda\overline{q}.[\![NP]\!]([\![in]\!](\overline{q}))$

This yields of course the inversed linking reading. Let us continue (76-a) with *an apple*$_1$.

(77) $\lambda p \lambda g \exists y.\mathbf{apple}(y) \wedge p(g[y/1])$

If we would do this simply by functional application, we would get something like *in every basket there is an apple*:

(78) $\lambda\overline{p}\lambda g \forall x.\mathbf{basket}(x) \rightarrow \overline{p}(\lambda g'.\mathbf{in}(g'(1), g'(2)))(g[x/2])(\lambda p \lambda g \exists y.\mathbf{apple}(y) \wedge p(g[y/1]))$

$=$

$\lambda g \forall x.\mathbf{basket}(x) \rightarrow (\lambda p \lambda g \exists y.\mathbf{apple}(y) \wedge p(g[y/1]))(\lambda g'.\mathbf{in}(g'(1), g'(2)))(g[x/2])$

$=$

$\lambda g \forall x.\mathbf{basket}(x) \rightarrow \lambda g \exists y.\mathbf{apple}(y) \wedge p(g[y/1]))(\lambda g'.\mathbf{in}(g'(1), g'(2)))(g[x/2])$

$=$

$\lambda g \forall x.\mathbf{basket}(x) \rightarrow \lambda g \exists y.\mathbf{apple}(y) \wedge \lambda g'.\mathbf{in}(g'(1), g'(2))(g[y/1]))(g[x/2])$

$=$

$\lambda g \forall x.\mathbf{basket}(x) \rightarrow \lambda g \exists y.\mathbf{apple}(y) \wedge \mathbf{in}(g[y/1](1), g[y/1](2)))(g[x/2])$

$=$

$\lambda g \forall x.\mathbf{basket}(x) \rightarrow \lambda g \exists y.\mathbf{apple}(y) \wedge \mathbf{in}(y, g(2)))(g[x/2])$

$=$

$\lambda g \forall x.\mathbf{basket}(x) \rightarrow \exists y.\mathbf{apple}(y) \wedge \mathbf{in}(y, g[x/2](2)))$

$=$

$\lambda g \forall x.\mathbf{basket}(x) \rightarrow \exists y.\mathbf{apple}(y) \wedge \mathbf{in}(y,x))$

This is of course not what we want to get, therefore the combination rule must be a bit more complicated in order to arrive at a decent NP meaning. What we would like to get as a result is (79):

(79) $\lambda p \lambda g \forall x.\mathbf{basket}(x) \rightarrow \exists y.\mathbf{apple}(y) \wedge p(g[y/1][x/2]) \wedge \mathbf{in}(y,x)$

The required combination rule is

(80) $\lambda p \lambda g [\![PP]\!](\lambda q [\![DP]\!](\lambda g.p(g) \wedge q(g)))$

Now, when applying the DP-meaning to $\lambda g.p(g) \wedge q(g))$ we replace the continuation in DP by a more complex continuation:

(81) $\lambda q \lambda p \lambda g \exists y.\mathbf{apple}(y) \wedge p(g[y/1])(\lambda g.p(g) \wedge q(g)))$
$=$
$\lambda p \lambda g \exists y.\mathbf{apple}(y) \wedge \lambda g.p(g) \wedge q(g)(g[y/1])$
$=$
$\lambda p \lambda g \exists y.\mathbf{apple}(y) \wedge p(g[y/1]) \wedge q(g[y/1])$

According to (81), we now have:

(82) $\lambda p \lambda g [\![PP]\!](\lambda p \lambda g \exists y.\mathbf{apple}(y) \wedge p(g[y/1]) \wedge q(g[y/1]))$

And continuing analogously to (77) above, it is clear that we now arrive at (78), and applying this to $\lambda g\mathbf{rotten}(g(1))$ yields:

(83) $\lambda g \forall x.\mathbf{basket}(x) \rightarrow \exists y.\mathbf{apple}(y) \wedge \lambda g.\mathbf{rotten}(g(1))(g[y/1][x/2]) \wedge \mathbf{in}(y,x)$
$=$
$\lambda g \forall x.\mathbf{basket}(x) \rightarrow \exists y.\mathbf{apple}(y) \wedge \mathbf{rotten}(g[y/1][x/2](1)) \wedge \mathbf{in}(y,x)$
$=$
$\lambda g \forall x.\mathbf{basket}(x) \rightarrow \exists y.\mathbf{apple}(y) \wedge \mathbf{rotten}(y) \wedge \mathbf{in}(y,x)$

This is exactly what we wanted to derive.

4.3 Comparatives

(84) John_1 is larger than every man_2

(85) er = λA λg dim=A ∧ dim(g(1)) ⟩ dim(g(2))

Conclusion: Incorporating assignments into the language of semantic interpretation provides us with a mild and uniform way of type shifting with just the right sort of complexity needed in order to do surface true interpretation. Or so it seem. . . We will see in the next sections that there is still a problem that cannot be solved by type shifting, neither of the sort advertized here nor by any other sort of type shifting used in Categorial Grammar.

5 Complex coherent constructions

Complex so-called coherent constructions have already been exemplified in (39), repeated as (86):

(86) der Mann dem_D Hund einen Knochen gegeben_P haben muss
the man the dog a bone given have must
V_1 AUX_2 AUX_3

The main characteristics of coherence is that the structure contains any number of auxiliaries in a single sentence. The syntax I will be assuming is that developed in Sternefeld (2006*a,b*) and I will only very briefly illustrate the main points by way of an example. Consider an Exceptional Case Marking construction like

(87) dass ihr den Wagen kommen hören *werdet*
that you the car come hear will
that you will hear the car come

In traditional grammar, it is assumed the verbs are structured as a cluster in a so-called verbal complex; ignoring complications of various sorts, the structure should thus be something like

(88) $[_{CP}$ dass $[_{VP}$ ihr den Wagen $[_{V}$ kommen hören werdet]]]

Internally to V, we must assume that the finite auxiliary verb *werdet* is the the head which projects its agreement features to the subject *ihr*. At the same time it selects an infinitive complement, let us call this X:

(89) $[_{CP}$ dass $[_{VP}$ ihr den Wagen $[_{V}$ $[_{X}$ kommen hören] werdet]]]

Clearly, X is a verbal projection which can be considered a VP. The head of this VP must be hören, a VP again:

(90) $[_{CP}$ dass $[_{VP}$ ihr den Wagen $[_{V}$ $[_{VP}$ $[_{VP}$ kommen] hören] werdet]]]

Note that both VPs have external arguments: den Wagen is the external argument of *kommen*, and *ihr* is the external argument of *hören*.

When replacing the future auxiliary werden by the aspectual perfect auxiliary haben, something strange happens. For most speakers of German, a structure like (90) would no more be well-formed:

(91)??[$_{CP}$ dass [$_{VP}$ ihr den Wagen [$_{V}$ [$_{VP}$ [$_{VP}$ kommen] hören] habt]]]

One reason for this might be, that *haben* normally requires a participle form, so that *hören* should be replaced by *gehört*:

(92)??[$_{CP}$ dass [$_{VP}$ ihr den Wagen [$_{V}$ [$_{VP}$ [$_{VP}$ kommen] gehört] habt]]]

But unfortunately this construction is again rather marked; the best way to express (92) is by chosing another order of the auxiliaries, namely:

(93) [$_{CP}$ dass [$_{VP}$ ihr den Wagen [$_{V}$ habt [$_{VP}$ [$_{VP}$ kommen] hören]]]

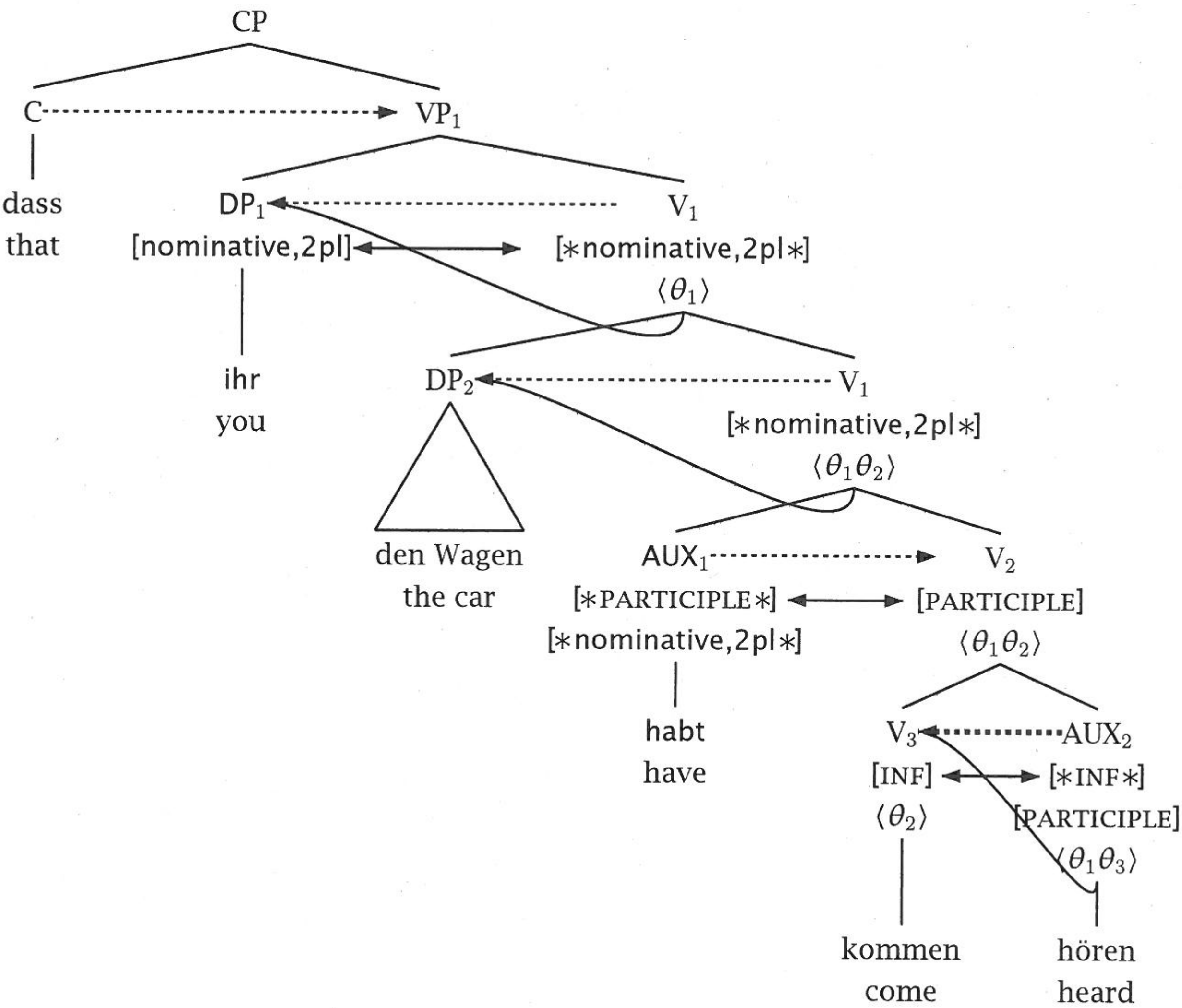
This change of order is one motivation for a syntactic analysis that unites all verbs under a single constituent, I will come back to the issue later. A second point to be made here is that the change of word order also goes along with an obligatory realisation of the participle as an infinitive; this is known as the IPP (*infinitivus pro participio*).

(94) *[$_{CP}$ dass [$_{VP}$ ihr den Wagen [$_{V}$ habt [$_{VP}$ [$_{VP}$ kommen] gehört]]]

The IPP-effect occurs (a) in some varieties of West-Germanic that have left-branching verb clusters; (b) in all varieties that have right-branching verb clusters; (c) only in those varieties of West-Germanic in which the past participle has a *ge*-prefix; (d) where there is variation with respect to the type of verb introducing the verb cluster in which the IPP-effect occurs, there is an implicational hierarchy: causatives ≪ modals ≪ perception verbs ≪ benefactives ≪ duratives ≪ inchoatives ≪ control verbs. There is much more to say about the IPP-effect (see Schmid et al. (2005)); here I will completely ignore the phenomenon.

For reasons to become clear later, I will use the label AUX for the temporal auxiliary and for the ECM-verb. Furthermore, I will use the label VP only for maximal projections that are complete in the sense of not having an external argument. Then the analysis of this structure can be summarized in the following tree that contains syntactic features and theta-grids:

(95)



I distinguish five components of grammatical description: (1) rules for theta assignment; (2) rules for morphological selection; (3) percolation conditions; (4) linearity conditions; and (5) adjacency conditions. Let me briefly comment on these conditions one by one:

(1) is the layer of theta assignment represented by the thin curved arrows starting from the theta positions of the θ-grids. As usual, discharging of theta-roles requires sisterhood. Only the rightmost theta role of the θ-grid can be discharged. Having discharged or assigned a theta role implies erasure from the list, the reduce list is then passed on to the mother node. In a well formed structure, all theta roles must be discharged (or otherwise removed from the list, cf. below). Moreover, all theta roles must bear different indeces.

Next there is the morphological level (2): a feature [$*\alpha*$] has to be **checked** against a matching feature [α]. In Sternefeld (2006*a*) I assumed that checking also requires sisterhood; this is represented by the solid double arrows. Due to this sisterhood requirement we also need a mechanism for feature projection.

Features may project only up to a maximal projection; within the theory of Bare Phrase Structure maximal projections are defined as non-heads. In the structure shown in (95), the node bearing the subcategorizing feature [$*\alpha*$] is the head (by definition); this means that V1 and VP_1 are projections of AUX_1, V_2 a projection of AUX_2. By definition, then, V_3 and V_2 are maximal projections. The agreement features of AUX_1 project to V_1 and further to the V_1-daughter of VP_1, where they are checked against the features of the subject. The precise mechanism of morphological selection has been worked out in Sternefeld (2006*a*). In a well formed structure, each feature [$*\alpha*$] must be checked; the counterpart features [α] must not be checked only if they are 'interpretable', otherwise they must also enter into a checking relation with [$*\alpha*$]. The details will not be of concern in what follows.

Turning next to the percolation conditions (3), observe that the projection of the θ-grids does not necessarily stop at maximal projections: As a defining characteristics of coherence we assume that maximal projections in the sense of morphological selection can still be transparent for theta marking. Observe that some kind of transparency is inherent to the traditional notion of an external argument; the innovation adopted here is that all arguments can in some sense be "external". This is a language specific assumption that is both natural and necessary in order to deal with complex predicates. The analysis is therefore **two-dimensional**, although both dimensions "live on" the same tree: one dimension is m-selection, the other is "administration of theta-roles" (a term I boroughed from Manfred Bierwisch). I will come back to an elaboration of point (3) later under the label of percolation conditions.[4]

Finally, the structure has to satisfy conditions on linearization (4) and adjacency (5). The former is represented by the dotted arrows between major categories; theta assignment is always to the left of a head, m-selection follows the rules discussed above. Adjacency must hold between V_3 and AUX_2 only and is represented by a thick dotted arrow.

As regards the semantics, the composition of theta roles is the most problematic issue: All previous theories of coherence were based on functional composition: two predicates must combine into a new predicate. The predicates as such were functions, but since the arities of the functions were not the same, there was no direct and compositional method of doing functional composition. This fact makes an ugly and very complicated theory (cf. Jacobs (1992) or Ste-

[4]Note that the top most VP is a natural boundary for projection of θ-grids, although perhaps not one that is absolute: if movement of V into C turns C into a verbal category (thereby turning the CP into yet another new VP), one might assume that the former topmost VP has become transparent for the projection of the subject, implying that the subject can be base generated in SpecC in situ, ie. without movement. This possibility has been discussed at some length in Sternefeld (2006*b*), Chapter 5, Section 3.

chow (1992)) which let me refrain from persuing the these proposals in previous attempts to design a simple and surface-true theory.

6 Scope reversal: A first analysis

Let us now return to a surface oriented analysis of coherence. Consider:

(96) dass bald ein Taxi [muss kommen]
that soon a cab must come

Th order of modal and main verb is grammatical only in Swiss German. The problem here is a purely semantic one, namely that *ein Taxi* has syntactic wide surface scope with respect to the modal *muss*, but the intended semantic interpretation should give wide scope to the modal, leading to a unspecific reading of *ein Taxi*. Syntactic reconstruction is out of question because nothing has moved.

Of course there are numerous ways to get the intended interpretation via more complicated types and various ways of type shifting as discussed in categorial grammar. In more traditional terms, we could assume, following Montague (1970) that all verbs are type shifted, so that *kommen* has type $\langle\langle et, t\rangle t\rangle$ rather than $\langle et\rangle$, together with a meaning postulate that ensures the subject transparency of *kommen*. We could then apply semantic reconstruction, and thus get the indended truth conditions. This way, however, is ruled out because just like QR, would overgenerate massively in allowing for any departure of surface scope, generating scope relations between arguments that do not exist.

As another (non-)solution to the problem we might assume a more local sort of type shifting. To illustrate the sort of calculation that inverts scope I have in mind, assume the following translations:

(97) ein Taxi$_k$
$\lambda p \lambda g \exists x[\mathbf{Taxi'}(g(@), x) \wedge p(g[k/x])]$

The convention is that g(@) denotes the actual world.

(98) kommen
$\lambda g.\mathbf{kommen'}(g(@), g(k))$

We assume here that the syntax has assigned the index k to the one theta role of the verb kommen. This implies that the meaning of kommen, namely $\lambda n \lambda g.\mathbf{kommen'}(g(@), g(n))$ has been applied to k. Now applying (97) to (98) yields an acceptable outcome, namely

(99) $\lambda p \lambda g \exists x[\mathbf{Taxi'}(g(@), x) \wedge p(g[k/x])](\lambda g.\mathbf{kommen'}(g(@), g(k)))$

$= \lambda g \exists x[\textbf{Taxi'}(g(@), x) \wedge \lambda g.\textbf{kommen'}(g(@), g(k))(g[k/x])]$
$= \lambda g \exists x[\textbf{Taxi'}(g(@), x) \wedge \textbf{kommen'}(g(@), x)]$

Now the modal *müssen* universally quantifies over a set of (contextually restricted) possible words, so that the meaning could be

(100) $\lambda p \lambda g \forall w[\textbf{müssen'}(g(@), w) \rightarrow p(g[(@)/w])]$

Appling (100) to (99) will give us:

(101) $\lambda p \lambda g \forall w[\textbf{müssen'}(g(@), w) \rightarrow p(g[(@)/w])](\lambda g \exists x[\textbf{Taxi'}(g(@), x) \wedge$
$\textbf{kommen'}(g(@), x)])$
$= \lambda g \forall w[\textbf{müssen'}(g(@), w) \rightarrow (\lambda g \exists x[\textbf{Taxi'}(g(@), x) \wedge$
$\textbf{kommen'}(g(@), x))(g[(@)/w])]]$
$= \lambda g \forall w[\textbf{müssen'}(g(@), w) \rightarrow \exists x[\textbf{Taxi'}(w, x) \wedge \textbf{kommen'}(w, x)]]$

The modal is thus represented by a relation between the actual world and the worlds w where everything that must be the case hold. But now consider the following variant.

(102) müssen
$\lambda p \lambda \overline{q} \lambda g \forall w[\textbf{müssen'}(g(@), w) \rightarrow (\overline{q}(p))(g[(@)/w])]$

I added an additional variable $\overline{q}$ which introduces a certain sort of delayed syntactic context, so that functional application between *ein Taxi* and *müssen kommen* will reverse, as seen below in the computation of our example. The logical type of $\overline{q}$ is a function which applied to a proposition yields a proposition again.

Combining these translations compositionally, we successively get the following:

(103) muss kommen
$\lambda \overline{q} \lambda g \forall w[\textbf{müssen'}(g(@), w) \rightarrow (\overline{q}(\lambda g.\textbf{kommen'}(g(@), g(k))))(g[(@)/w])]$

(104) ein Taxi muss kommen
$\lambda g \forall w[\textbf{müssen'}(g(@), w) \rightarrow$
$(\lambda p \lambda g \exists x[\textbf{Taxi'}(g(@), x) \wedge p(g[k/x])](\lambda g.\textbf{kommen'}(g(@), g(k))))(g[(@)/w])]$
$=$
$\lambda g \forall w[\textbf{müssen'}(g(@), w) \rightarrow$
$(\lambda g \exists x[\textbf{Taxi'}(g(@), x) \wedge \lambda g.\textbf{kommen'}(g(@), g(k))(g[k/x])])(g[(@)/w])]$
$=$
$\lambda g \forall w[\textbf{müssen'}(g(@), w) \rightarrow$
$(\lambda g \exists x[\textbf{Taxi'}(g(@), x) \wedge \textbf{kommen'}(g(@), x)])(g[(@)/w])]$
$=$

$$\lambda g \forall w[\textbf{müssen'}(g(@), w) \rightarrow \exists x[\textbf{Taxi'}(w, x) \wedge \textbf{kommen'}(w, x)]]$$

This is precisely the intended translation.

As should be obvious, however, the analysis described above is still very much ad hoc. In fact, it turns out that the method illustrated does not work in full generality, for instance when there are two quantifiers to be crossed. The most general case can be described as follows:

Assume we have a sequence of quantifiers Q and a sequence of modal operators M. The surface order S is composed out of both sequences, so that substracting M from S yields Q and substracting M from S yields Q. The semantic interpretation of S is scope ambiguous in the following way:

(105) a. Any Q_i that asymmetrically c-commands Q_j has scope over Q_j.
b. Any M_i that asymmetrically c-commands Q_j/M_j has scope over Q_j/M_j.
c. Any M_i may have scope over an Q_i that c-commands M_i.

In Sternefeld (2006*b*) this situation has been descibed as a syntactic difference between surface and Logical Form. LF can be genererated by head movement of the modals in an acyclical way that satisfies the head movement constraint. It is easy to see that this mechanism satisfies the conditions in (105). On the other hand, LF-movement is a device that should be dispensed with in a theory that generates surface structures as we did above: Above we took much pains to avoid any abstract additional structure in the syntactic analysis, and it would be counter-productive to reduce structure in one part of the syntax but re-introduce additional structure at another part of syntax, namely at LF. Furthermore, as discussed in Sternefeld (2006*b*), this kind of head movement is in fact QR, and this is precisely the operation we wanted to get rid of.

Moreover, the solution above suffers from undergeneration, because it does not allow us to generate all the readings that satisfy the requirements laid down in (105). The reason for this can be traced back to a property that has been called "syntactic integrity" in Barker (2002). Barker's theory is surface true in the sense that surface compositionality interprets only (binary) branchings in a tree, using rich forms of type shifting as proposed by many proponents of categorial grammar. By various sorts of type shifting, each branching [A B] offers two possibilities: either A applies to B or vice versa. Chosing appropriate types this yields either scope; on the other hand, if B gains scope over A, material contained in B will also outspope A, and this is a sort of Kaynean result that gives us the wrong scope relations for the case at hand. This result is not accidental, but systematic; the method yields systematically wrong truth conditions, because it does not allow us to keep the relative order of M's and Q's, while at the same time varying the order of Ms with respect to the Qs.

Some reflection will reveal that the problem is indeed one that is beyond a purely semantic treatment that obeys any form of syntactic integrity; what is needed is an additional syntactic device which has traditionally been called scope indexing. I will show how this mechanism can be employed to solve the current problem in the next section.

7 Incremental interpretation

Before we tackle the problem of scope reversal, we have to introduce a new device which allows us to take a new perspective on the problem at hand. The general aim is to arrive at a semantic interpretation that operates on the syntactic surface in a way that is consistent with semantic parsing. This implies that we interpret sentences not in a bottom up manner but from left to right. Under this pespective, scope reversal can be analysed as first introducing a variable for the scope taking operator which is bound by the operator as soon as this operator is encountered on the way from left to right.

Let us start with a number of simplifcations. Above we took pains to explain that landba conversion is illegitimate in certain contexts, and that taking assignments into the representation language can solve the problem. We still adhere to this technique, but simplify notation by not mentioning assignments at all in our formulas. This means that in a formula like

(106) $\lambda y \exists x.P(y,x)(x)$

lambda conversion is possible, so that (106) is in fact equivalent to $P(x,x)$.

Another simplification concern the lexical material I will consider: I will mostly disregard the restricting property of quantifiers, the notation I adopt for a quantifier like *everyone* is this:

(107) $\lambda p \forall x.p$

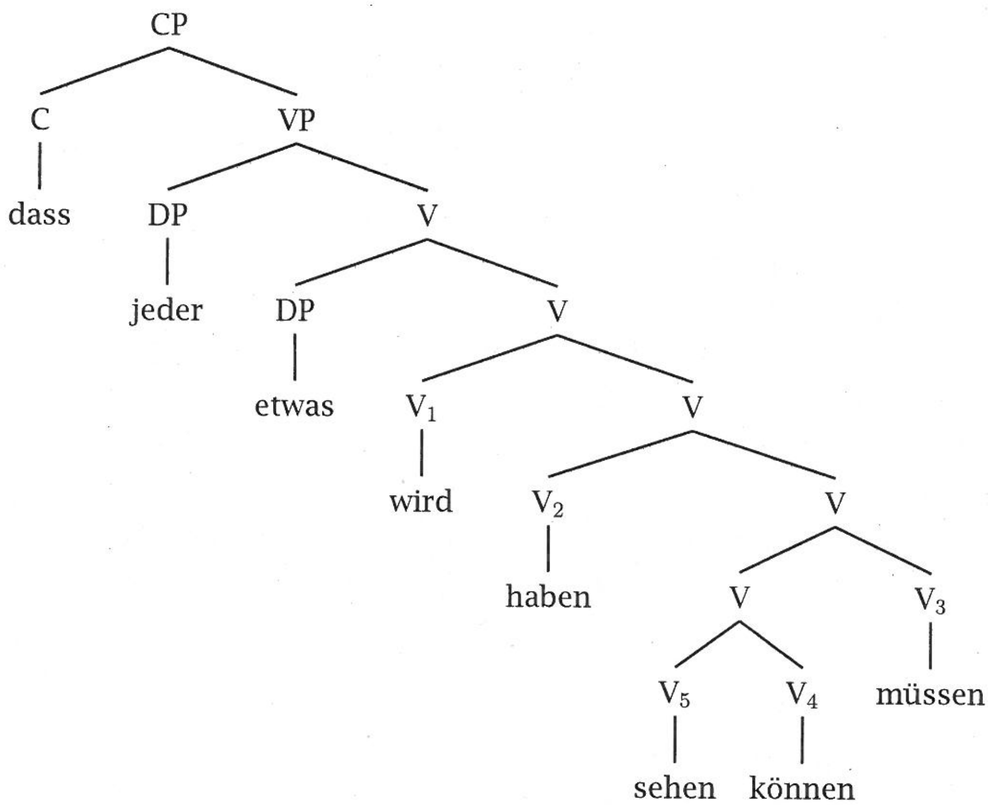
Building in the restriction is a trivial matter built in in section 10.

I will also completely ignore the complexity of the meaning of auxiliaries: In what follows I pretend that auxiliars are simply sentential operators like $\Diamond$ (can), $\Box$ (must), $\mathcal{F}$ (= future), or $\mathcal{P}$ (= past). I will also ignore control verbs or other types of verbs, thus concentrating on simple sentential operators.

I will now examine the semantics of coherent sentences from an incremental left to right point of view. Lets take the following example:

(108)	(dass)	jeder	etwas	wird	haben	sehen	können	müssen
	(that)	everyone	something	will	have	seen	can	must

'that it will have been the case that everyone must have been able to see something'



Ignoring the complementizer, we start from left to right with the VP. The structure to be interpreted is [$_{\text{VP}}$ jeder V]. The lexical entry for *jeder* is given in (107). We assume that *jeder* is theta indexed by a verb, having a theta role θ_i. By comparing V with VP we know that this θ-role has been discharged, which leads to a semantic operation on (107): recall that (107) is short for an expression that applies to a number, namely the numeral i of the discharged theta role. So applying (107) to i yields a new representation, simplified again in (109):

(109) $\lambda p \forall x_i.p$

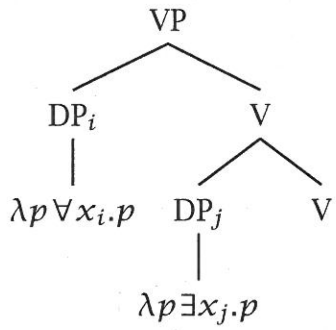
It is this quantifier that will ultimately apply to **see'**(x_i, x_j).

Let us move on to the next lexical item *etwas*. This is not a sister of (109), but introduces its own syntactic context [$_{\text{V}}$ *etwas* V]. As before, this expression will be interpreted in a syntactic context determined by the θ-grid of the lower V, which is $\langle \theta_i \theta_j \rangle$ which leads to the index j and the representation (110):

(110) $\lambda p \exists x_j.p$

What we get then is:

(111)



Since both *jeder* and *etwas* form a connected subtree, the second DP must be interpreted in the immediate context of the first, so that the resulting interpretation should be something like:

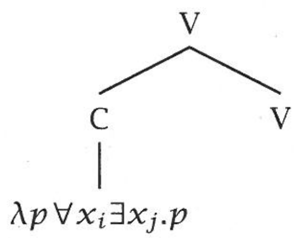
(112) $\lambda p \forall x_i \exists x_j.p$

From this we can calculate the more general semantic rule that interprets the insertion of the subtree of *etwas* into the subtree of *jeder*. The rule combines the meanings of two DPs in the following way:

(113) $[_V \text{DP}_i\ [_V \text{DP}_j \text{ V }]] \rightsquigarrow \lambda p.[\![DP_i]\!]([\![DP_j]\!](p))$

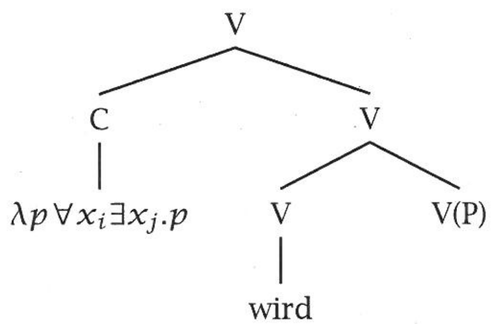
What we get in semantic representation is (114), where C represents the left context calculated so far.

(114)



Next, we encounter the future tense expressed by *wird*.

(115)



As before, we know that the result should be something like (116):

(116) $\lambda p \forall x_i \exists x_j . \mathcal{F} p$

At this point, we run into difficulties because we have ignored intensionality. Although in Montague's intensional logic, the formulas in (117) are wellformed, they are not equivalent:

(117) a. $\mathcal{F} A$
 b. $[\lambda p \mathcal{F} p](A)$

The reason is that A is a constant of type t, and constants are interpreted relative to possible worlds. In (117-b), however, p is a variable of type t, and variables are never interpreted relative to worlds. What we need is a variable of type $\langle st \rangle$. So, with p of type $\langle st \rangle$ the correct formula euivalent to (117-a) is:

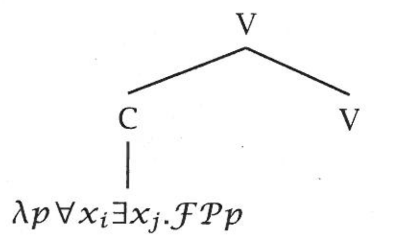
(118) $[\lambda p \mathcal{F}\,{}^{\vee}p]({}^{\wedge}A)$

In what follows I will deliberately ignore such complications, they will not in any way influence the architecture of the theory, except that if taking it seriously it would somewhat complicate the notation. We therefore accept (115); accordingly, the general operation that interprets generates (115) is

(119) $\lambda p . [\![C]\!](\mathcal{F}(p))$

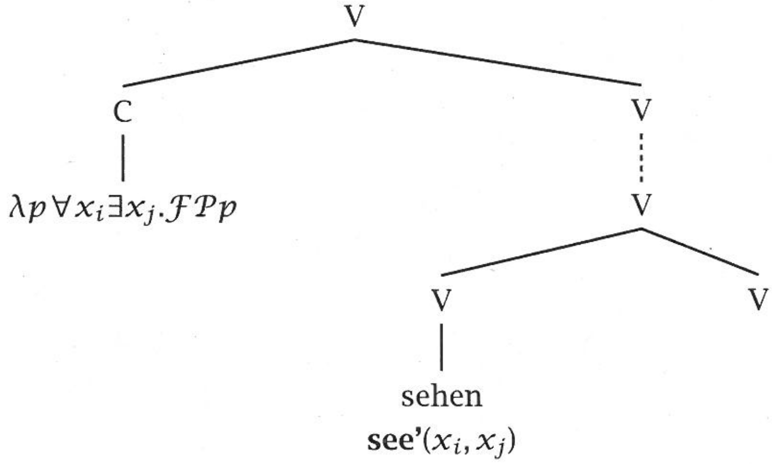
The next operation interpreting *haben* doesn't introduce anything new, so by the simplifying assumption that *haben* expresses Past, and by applying the same operations as before, we now get:

(120)



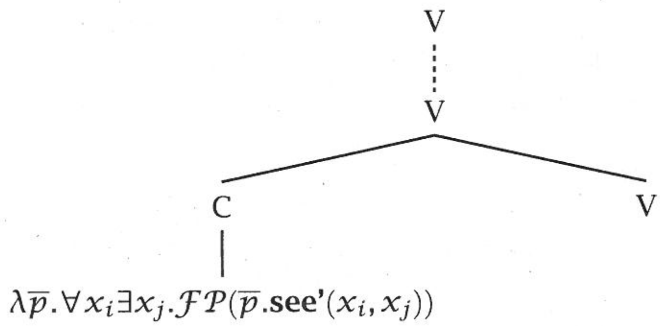
Next we encounter the main verb *sehen*. The syntax of Standard German predicts that *sehen* must be syntactically unconnected with the context C, therefore something must be missing here, and the structure is something like:

(121)



Now, if we want to combine C and V, we must somehow take account for the missing V which we represent as a variable $\overline{p}$ which expresses a property of proposition, also referred to as a possible continuation. The result is (122):

(122)

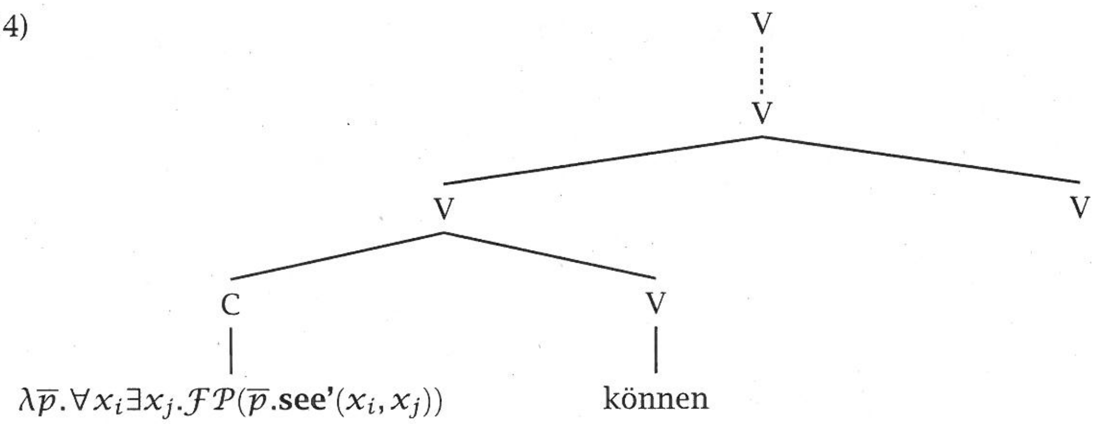


The general rule for α as the main verb to combine with C is:

(123) $\lambda p \lambda \overline{p}.[\![C]\!](\overline{p}p)(\alpha)$

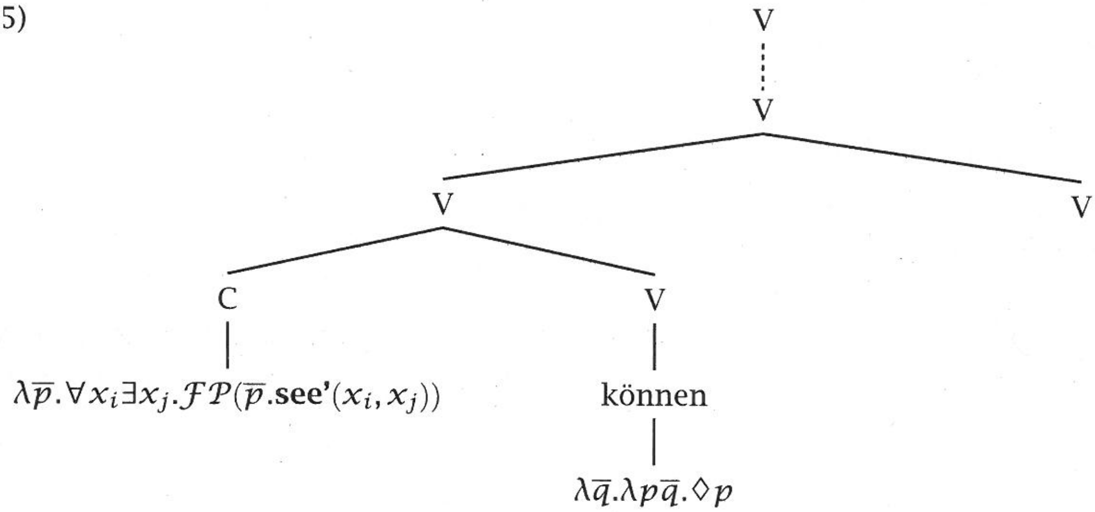
Next, we encounter the modal *können*, which is directly attached to *sehen*.

(124)



Now, directly applying C to V would yield the wrong result, because there is no place left for the remaining modals. Let us therefore type shift the meaning of *können* so as to account for possible continuations; we then get the following:

(125)



The general rule to combine these nodes is

(126) $\lambda\overline{r}.C(\mathbf{Modal}(\overline{r}))$

Let us calculate this in detail.

(127)
a. $\lambda\overline{r}.C(\lambda\overline{q}\lambda p.\overline{q}\Diamond p(\overline{r}))$ (insert modal)
b. $\lambda\overline{r}.C(\lambda p.\overline{r}\Diamond p)$ (lambda conversion)
c. $\lambda\overline{r}.\lambda\overline{p}.\forall x_i \exists x_j.\mathcal{FP}(\overline{p}.\mathbf{see'}(x_i, x_j))(\lambda p.\overline{r}\Diamond p)$ (insert C)
d. $\lambda\overline{r}.\forall x_i \exists x_j.\mathcal{FP}(\lambda p.\overline{r}\Diamond p).\mathbf{see'}(x_i, x_j))$ (lambda conversion)
e. $\lambda\overline{r}.\forall x_i \exists x_j.\mathcal{FP}(\overline{r}\Diamond \mathbf{see'}(x_i, x_j))$ (lambda conversion)

This is exactly what we want to get. Note that the type shifting needed to achieve this result ensures that the modal has narrow scope with respect to what follows, it therefore respects c-command. In theory, we could also have the inverse type of shifting, namely:

(128) $\lambda\overline{q}.\lambda p.\Diamond\overline{q}p$

This is the continuation we would expect if the modal is on a right branch c-commanding its scope. Thus, the type of shifting depends on syntactic context. If (128) were chosen in the stucture

(129) [[V Modal_1] Modal_2]

We would have predicted the wrong scope relations, with Modal 1 having scope over Modal_2. The question arises whether in a syntactic context as the one displayed in the tree above such a scope reversal should ever be permitted. I regard this as an open question, with an eye on the many problems that arise in structures like

(130) $[_{VP}\ [_{V'}\ \mathrm{V}\ \mathrm{DP}_1\ ]\ \mathrm{DP}_2\ ]$

where in fact DP_1 should have scope over DP_2. It seems to me that the two-dimensional analyses that have been proposed were motivated mainly by semantic facts, so that if these can be treated completely within semantics, there is no need for multi-dimensional syntactic representations, as proposed eg. by Pesetsky (1995).

Now the next modal we encounter is *müssen*:

(131)

V
⋮
V
├── C
│   └── $\lambda\overline{r}.\forall x_i \exists x_j.\mathcal{FP}(\overline{r}.\Diamond\mathbf{see'}(x_i, x_j))$
└── V
    └── müssen

Continuing in the same manner would of course yield:

(132) a. $\lambda\overline{p}.\forall x_i \exists x_j.\mathcal{FP}(\overline{p}.\Box\Diamond\mathbf{see'}(x_i, x_j))$

But now we have to account for the fact, that there is no further syntactic continuation: Realising that the tree is now connected, there cannot be any continuation, so it now suffices to apply C to the non-type shifted modal, which directly leads to (133):

(133) $\forall x_i \exists x_j.\mathcal{FP}(\Box\Diamond\mathbf{see'}(x_i, x_j))$

This completes our sample derivation.

8 Modal displacement

Let us now return to the problem of "displaced" operators. Above we have shown that something like "delayed interpretation" can be captured by variables for possible continuations, and this in turn suggests that we should introduce some variable $\overline{p}$ for the same purpose. However, as observed above, the relation between anticipated variables and related operators is not nesting but crossing. This is the main reason why a purely semantic handling of the dependency in terms of the lambda calculus is not feasable. We therefore have to invoke some sort of syntactic mechanism, alongside and similar to argument indexing; this syntactic mechanism will tell us which variable to bind and in which order. In a sense, it is the mirror image of argument indexing: with the latter, the argument index percolates down from binder to the variable to be bound, the scope index percolates from avariable to its binder. Let us see how this can be worked out.

By way of example, assume that the scopal distribution of operators should be this:

(134) $\mathcal{F}\forall x_i\mathcal{P}\Box\exists x_j\Diamond\mathbf{see'}(x_i, x_j)$

Going from left to right, the first problem is to account for the future operator outscoping the universal quantifier. Gaining scope can as usual be represented by:

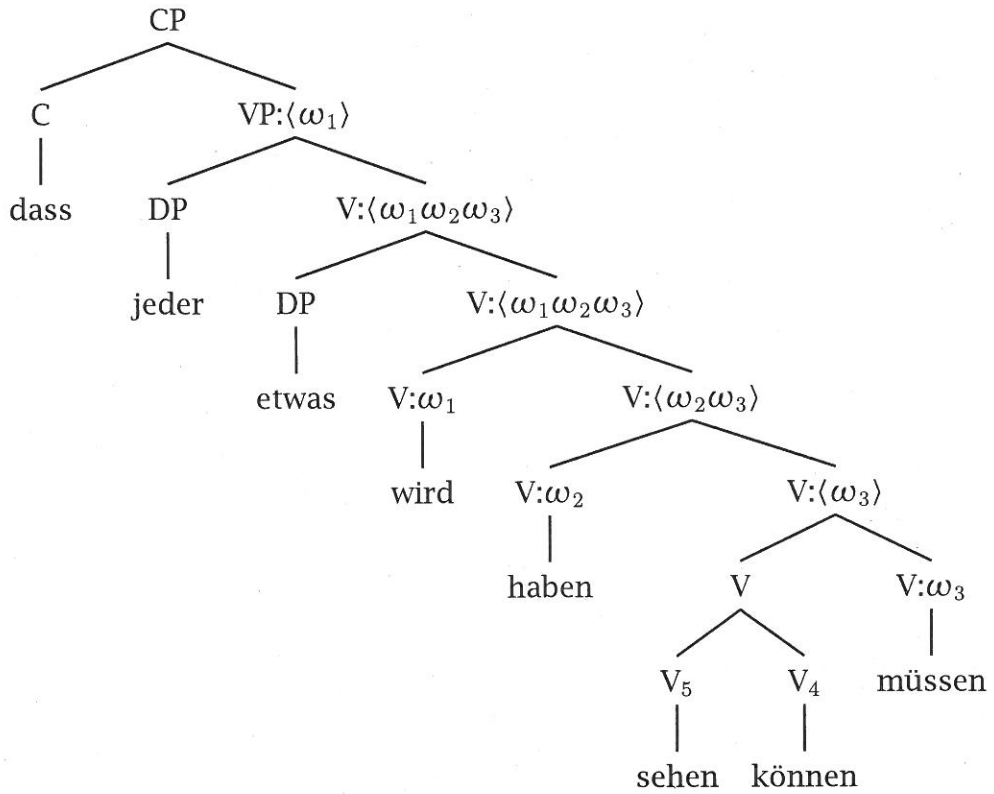
(135) $\lambda p\overline{p}\forall x_i.p$

(135) must do two things: first, it must syntactically identify the variable to be bound by the quantifier; this is done be the index i, which is actually the result of the syntactic configuration of theta indexing. Which is to say that the sister node of (135) contains θ_i as the right most element of theta indexing which identifies the argument of the predicate. Concerning the variable $\overline{p}$, it must in turn identify the operator whose application is procrastinated.

Identification has to bee encoded by indexing, as discussed above, indexing should be a means not only to identify arguments, but also to identify prospective operators. In the present context, this means that syntax must include some kind of scope indexing, which operates alongside with theta-indexing. That is, besides theta-grids as sequences of argument identifying indexes, we also assume sequences of operator identifying indexes, which identify the place of the interpretation of an operator to come. Using numbers again to identify variable operator relations, we assume that an indexed syntactic structure that is generated as the source of (135) is this:

(136) jeder [$_1$ etwas [$_{123}$ wird$_1$ [$_{23}$ haben$_2$ [$_3$ sehen können$_4$]
everyone something will have see can
müssen$_3$]]]
must

For convenience, we may assume that the structure to be interpreted is



Now, interpreting this structure from left to right, the first branching contains an additional operator index. This is a sequence containing one index we write as ω_1. Whereas θ_i introduces an identifying argument, ω_1 introduces a new variable $\overline{p_1}$ that will become bound later by the coindexed operator. The result is shown in (137).

(137) $\lambda p \overline{p}_1 \forall x_i.p$

Again we assume that this is possible only if the mother node does not contain ω_1, which means that this operator index has been discharged at the current position. The general operation assumed here is

(138) $\lambda p.\overline{p}_i \mathrm{DP}(p)$

Going one step further to the right we now have to interpret the existential quantifier in the context of the sequence $\langle \omega_1 \omega_2 \omega_3 \rangle$. As with theta role discharge (with respect to the θ-grid $\langle \theta_i \theta_j \rangle$ we first look at the rightmost index which by (138) yields:

(139) $\lambda p \overline{p}_3 \exists x_j.p$

Inspecting the syntactic context now reveals that the result of removing the last ω would not be identical to the mother node; we still have to account for ω_2.

Proceeding as before, this yields

(140) $\lambda p\overline{p}_2\overline{p}_3\exists x_j.p$

Since removal of $\omega_2\omega_3$ leaves ω_1 as the index of the mother node we are done and can combine this with the preceding context as discussed above, which yields:

(141) $\lambda p\overline{p}_1\forall x_i.\overline{p}_2\overline{p}_3\exists x_j.p$

Turning to the next branching we see that ω_1 is projected from the head *wird*$_1$, which means that the interpretation must bind $\overline{p}_1$ as shown in (142):

(142) a. $\lambda\overline{p}_1(141)(\mathcal{F})$
b. $\lambda p\mathcal{F}\forall x_i.\overline{p}_2\overline{p}_3\exists x_j.p$

By the same reasoning we interpret *haben*$_2$:

(143) a. $\lambda\overline{p}_2(142)(\mathcal{P})$
b. $\lambda p\mathcal{F}\forall x_i.\mathcal{P}\overline{p}_3\exists x_j.p$

Next, we reach the point where right branching switches to left branching, the interpretation of which has already been dealt with above: We must, at this stage, introduce a new variable $\overline{p}$ that accounts for the operators to come.

(144) $\lambda\overline{p}\mathcal{F}\forall x_i.\mathcal{P}\overline{p}_3\exists x_j.\overline{p}\mathbf{see'}(x_i,x_j)$

The next operator *können* is interpreted in situ as usual:

(145) $\lambda\overline{p}\mathcal{F}\forall x_i.\mathcal{P}\overline{p}_3\exists x_j.\overline{p}\Diamond\mathbf{see'}(x_i,x_j)$

In contrast, the modal *müssen*$_3$ projects its scope index, which means that the structure must be interpreted by replacing $\overline{p}_3$, yielding:

(146) $\lambda\overline{p}\mathcal{F}\forall x_i.\mathcal{P}\Box\exists x_j.\overline{p}\Diamond\mathbf{see'}(x_i,x_j)$

Attaching the left branching part to the remainder of the tree, thereby completing the structure, also completes the semantic derivation; if nothing more can be added, we apply the continuation representation to the trivial identity function; we thereby get (147), which is (133) repeated:

(147) $\mathcal{F}\forall x_i.\mathcal{P}\Box\exists x_j.\Diamond\mathbf{see'}(x_i,x_j)$

This also completes our admittedly somewhat sketchy discussion of the necessary semantic calculations.

It remains to account for the syntactic mechanism of scope or omega indexing. Discharging of ωs is immediately obvious: as already discussed above it precedes from right to left. Conversely, given a branching $[_C \text{ A B }]$ or $[_C \text{ B A }]$ with A as a modal head. Then the following rules apply:

(148) a. If B is omega indexed, A must have an omega index and C contains B's omega sequence plus A's index to the sequence at the left, i.e. at the beginning of the sequence.

b. If C does not contain an ω-indexing, we may or may not add B's omega index to C.

The two options in (148-b) say that we either interpret the operator in situ, or try to give it wide scope by passing its index on to C. The rule in (148-b) ensures to things: (1) once we know from A that an operator tries to get wide scope, adding the index of B is obligatory; this is analogous to the head movement constraint at LF: we cannot simply skip an operator and thereby change c-command among operators on the way to LF. (2) Expanding the sequence at the left corresponds to something like shape conservation or crossing (instead of nesting). This is the crucial aspect of the analysis which hindered us to give a purely semantic account: it seems to me that there is no other way to deal with crossing other than syntactic.

There are aLso additional constrains that should have some semantic explanation I am still unable to state in purely semantic terms. A case in point concerns the scope negation. Regardless of how negation is analysed, as an adjunct or as a head, the scope of negation seems to be surface true in German: it is impossible for an operator to outscope negation at LF. Semantically this would imply that negation replaces all free scope variable of the context C by the identity function. When it then comes to the interpretation of the modal, we must ensure that lambda abstraction cannot be vacous. But this is surely a purely syntactic constraint. Alternatively, we could block scope percolation right away in syntax: in a context like $[_A \text{ NEG B}]$ omega indexing cannot be projected from B to A, which means that it is not permitted in this context, and the same might also be the case for other adverbials whose scope is completely determined by their surface position.

9 Some general remarks on scope taking

The head movement constraint is one of the most solid axioms of Generative Grammar, so the fact that it is also implemented in the present semantic account of operator scope seems to introduce a redundancy which tends to make an account in terms of LF movement somewhat more attractive than I would like to have it. Observe, however, that the validity of the constraint, although seemingly uncontroversial in overt syntax, is not so undisputable in semantics. For example, it has been argued the the relation between tense and modals in English is not as straightforward as one might expect. The reason for this is that modals cannot be emdedded in English, in particular they cannot be embedded under tense. Nonetheless, has been argued that in sentences like

(149) a. John might have won
b. When I was a school boy I would have gone to Paris if I had won in the lottery

the tense operator can receive a wide scope interpretation with respect to the modal (cf. Condoravdi (2002)). This seems to suggest that, in syntactic terms, the HMC is perhaps not as solid as should be. In terms of scope indexing, the situation is described very simply by the assumption that *have* can project a scope index, whereas *must* permits the index to project without adding its own index; this precisely allows for scope inversion.

Another question concerns our assumption that quantifier scope in the Middle Field is determined by surface position. This claim seems to be blatantly wrong, since it is quite obvious that a sequence like $\forall\exists$ can be interpreted as $\exists\forall$. I would not deny the fact as such, although my reaction is that this does not in itself justify a rule like QR to apply. What we see here is rather a case of scope independent interpretation in the sense of Liu (1990), i.e. a specific *in situ* interpretation that ignores the previous interpretation of *every*. I briefly indicate how this can be implemented in some standard semantics in the Appendix ??. The reason for rejecting QR is an asymmetry between the availabilty of scope independent reading (which is always easy to get) and an interpretation of the surface sequence $\exists\forall$ as $\forall\exists$ which I claim to be impossible. This is against much of the literature on scope in German which has been demonstrated in extenso that a wide scope reading should be easily available. Let us see how this is possible.

Looking at the standard examples

(150) dass ein Student jedes Buch liest, ist (un)wahrscheinlich
that a student every book reads, is (un)likely

(151) dass ein Mann jede Frau heiratet, ist (un)wahrscheinlich
that a man every woman marries, is (un)likeky

I simply do not get the inverse reading, although it is well known as a universal of quantifier scope, that *every* is the quantifier most eligable for wide scope, and secondly that in most situations the surface reading with wide scope of the existential is much more unlikely than the reverse reading. Still I find the reverse reading so hard to get, that it should be blocked by principles of grammar.

On the other hand there are numerous examples where wide scope seems unproblematic. A famous example, due to Fanselow in early liter

(152) a. dass in Spanien ein Polizist vor jeder Bank steht
that in Spain a policeman in-front-of every bank stands
b. dass in Spanien vor jeder Bank ein Polizist steht

Although I find the surface order in (152-b) slightly more natural, I have no difficulty at all in interpreting (152-a) exactly the same as (152-b) with every bank having wide scope. Why is this?

First of all one might claim that this reading is the only one that makes sense, because it is physically impossible for a policeman to be in front of more than one bank. But I think this is not the explanation; the decisive difference is that the relation expressed is not one between indiviuals directly, it is the PP which introduces different places (each one containing a bank) and that it is the place parameter that gets wide scope: in front of a bank defines a place much like other operators define a world or a time. The wide scope of *every* means that it gets wide scope with respect to places. In other words, what we might paraphrase (152) as: For every place o such that it is in front of a bank, it holds that a policeman stands at o.

Assume now that in-front-of-every-bank is an operator defining places alongside with the usual operators on times and worlds. Then the structure is something like (153) with the PP having a scope indexing as discussed earlier for modal operators.

(153) dass $\overline{p}_i \exists x.$Polizist($x$) [(vor jeder Bank)$_i$ steht(x)]

More explicitly, assume that *vor jeder Bank* means:

(154) $\lambda p \forall y.\textbf{bank'}(y) \rightarrow \exists o.\textbf{in-front-of'}(o,y) \wedge p(o)$

We can now process (152) as follows:

(155) a. $\lambda p \exists x.\textbf{policeman'}(x) \wedge p$ (ein Polizist)
b. $\lambda p \overline{p}_i \exists x.\textbf{policeman'}(x) \wedge p$ (ein Polizist, i)
c. $\lambda \overline{p}_i [\lambda p \overline{p}_i \exists x.\textbf{policeman'}(x) \wedge p](\lambda p \forall y.\textbf{bank'}(y) \rightarrow \exists o.\textbf{in-front-of'}(o,y) \wedge p(o)))$ (ein Polizist vor jeder Bank)
d. $[\lambda p(\lambda p \forall y.\textbf{bank'}(y) \rightarrow \exists o.\textbf{in-front-of'}(o,y) \wedge p(o))) (\exists x.\textbf{policeman'}(x) \wedge p)]$ (λ-conversion)
e. $\lambda p \forall y.\textbf{bank'}(y) \rightarrow \exists o.\textbf{in-front-of'}(o,y) \wedge \exists x.\textbf{policeman'}(x)$

$\wedge\ p(o))$ (λ-conversion)

f. $\lambda p \forall y.\mathbf{bank'}(y) \rightarrow \exists o.\mathbf{in\text{-}front\text{-}of'}(o,y) \wedge \exists x.\mathbf{policeman'}(x)$
$\wedge\ p(o))(\lambda o.\mathbf{stand'}(x,o))$ (ein Polizist vor jeder Bank steht)

g. $\forall y.\mathbf{bank'}(y) \rightarrow \exists o.\mathbf{in\text{-}front\text{-}of'}(o,y) \wedge \exists x.\mathbf{policeman'}(x)$
$\wedge\ \lambda o.\mathbf{stand'}(x,o)(o))$ (λ-conversion)

h. $\forall y.\mathbf{bank'}(y) \rightarrow \exists o.\mathbf{in\text{-}front\text{-}of'}(o,y) \wedge \exists x.\mathbf{policeman'}(x)$
$\wedge\ \mathbf{stand'}(x,o))$ (λ-conversion)

Note also that we get a nice account of inversed linking, as exemplified in (156).

(156) Genau ein Polizist vor jeder Bank schläft
exatly one policeman in front of every bank sleeps

(156) is somewhat marginal, meaning the same as

(157) Vor jeder Bank schläft genau ein Polizist

But the structure of (156) is radically different from (157) in that *genau ein Polizist vor jeder Bank* is a constituent in (156) but not in (157). Analysing the incremental generation of this constituent we first find the DP *genau ein Polizist* being a projection of the quanifier, and I have argued elsewhere in Sternefeld (2006*a*) that modifications of DPs are attached to DP as a further projection of D. This implies that an ordinary DP meaning has to be combined with the PP in a structure like $[_{DP}\ [_{D'}$ genau ein Polizist $][_{PP}$ vor jeder Bank$]]$. Returning to (153) we see that the logical type of the PP is identical to that of the DP; moreover, the branching is not asymmetrical, rather, there seems to be a choice between $\lambda p.\mathrm{DP}(\mathrm{PP}(p))$ and $\lambda p.\mathrm{PP}(\mathrm{DP}(p))$.

(158) Weiss Karl, dass in Spanien genau ein Polizist jede Bank bewacht?

An extremely costly operation,

10 Interpreting the restriction of a quantifier

References

Barker, Chris (2002): 'Continuations and the Nature of Quantification', *Natural Language Semantics* **10**, 211–242.

Beck, Sigrid (1996): Wh-constructions and transparent Logical Form. PhD thesis, University of Tübingen.

Bennett, Michael (1974): Some Extensions of a Montague Fragment of English. PhD thesis, University of California, Los Angeles.

Bennett, Michael (1979): Questions in Montague Grammar. Indiana University Linguistics Club.

Bouma, Gosse (2003): Verb Clusters and the Scope of Adjuncts in Dutch. *In:* P. Seuren and G. Kempen, eds, *Verb Constructions in German and Dutch.* Benjamins, Amsterdam, pp. 5–42.

Condoravdi, Cleo (2002): Temporal Interpretation of Modals: Modals for the Present and for the Past. *In:* D. Beaver, ed., *The Construction of Meaning.* CSLI, Stanford, pp. 59–88.

Dowty, David (1991): 'Thematic Proto-Roles and Argument Selection', *Language* **67**, 547–619.

Heim, Irene and Angelika Kratzer (1998): *Semantics in Generative Grammar.* Blackwell, Malden, Mass.

Hodges, Wilfried (1998): 'Compositionality is not the Problem', *Logic and Logical Philosophy* **6**, 7–33.

Jacobs, Joachim (1992): 'Bewegung als Valenzvererbung — Teil I', *Linguistische Berichte* **138**, 85–122.

Kamp, Hans (1971): 'Formal Properties of "now"', *Theoria* **37**, 227–273.

Kaplan, Ronald and Annie Zaenen (2003): West-Germanic Verb Clusters in LFG. *In:* P. Seuren and G. Kempen, eds, *Verb Constructions in German and Dutch.* Benjamins, Amsterdam, pp. 127–150.

Kempen, Gerard and Karin Harbusch (2003): Dutch and German Verb Constructions in Performance Grammar. *In:* P. Seuren and G. Kempen, eds, *Verb Constructions in German and Dutch.* Benjamins, Amsterdam, pp. 185–222.

Kracht, Marcus (2007): 'The Emergence of Syntactic Structure', *Linguistics and Philosophy* **30**, 47–95.

Kratzer, Angelika (1995): Scope or Pseudoscope? Are ther Wide-Scope Indefinites. Ms. , University of Massachusetts at Amherst.

Krifka, Manfred (2001): 'Quantifying into Speach Acts', *Natural Language Semantics* **9**, 1–40.

Liu, F. (1990): Scope Dependency in English and Chinese. PhD thesis, UCLA.

Montague, Richard (1970): 'Universal Grammar', *Theoria* **36**, 373–398.

Pesetsky, David (1995): *Zero Syntax. Experiencers and Cascades.* The MIT Press, Cambridge, Mass.

Rambow, Owen (2003): Coherent Constructions in German: Lexicon or Syntax. *In:* P. Seuren and G. Kempen, eds, *Verb Constructions in German and Dutch.* Benjamins, Amsterdam, pp. 223–246.

Saarinen, Esa (1979): *Essays in honour of Jaakko Hintikka.* Reidel, Dordrecht.

Schmid, Tanja, Markus Bader and Josef Bayer (2005): Coherence — an Experimental Approach. *In:* S. Kepser and M. Reis, eds, *Linguistic Evidence. Empirical, Theoretical and Computational Perspectives.* Mouton de Gruyter, Berlin,

pp. 435–456.
Stechow, Arnim von (1991): Syntax und Semantik. *In:* A. v. Stechow and D. Wunder lich, eds, *Semantik. Ein internationales Handbuch zeitgenössischer Forschung*. Walter de Gruyter, Berlin, pp. 90–148.
Stechow, Arnim von (1992): Kompositionsprinzipien und grammatische Struktur. *In:* P. Suchsland, ed., *Biologische und soziale Grundlagen der Sprache*. Niemeyer, Tübingen, pp. 175–248.
Stechow, Arnim von (1993): Reconstruction and Logical Form. Ms. Universität Tübingen.
Sternefeld, Wolfgang (2001): Semantic vs. Syntactic Reconstruction. *In:* C. Rohrer, A. Rossdeutscher and H. Kamp, eds, *Linguistic Form and its Computation*. CSLI Publications, Stanford, CA, pp. 145–182.
Sternefeld, Wolfgang (2006*a*): *Syntax. Eine morphologisch motivierte generative Beschreibung des Deutschen. Band 1*. Stauffenburg Verlag, Tübingen.
Sternefeld, Wolfgang (2006*b*): *Syntax. Eine morphologisch motivierte generative Beschreibung des Deutschen. Band 2*. Stauffenburg Verlag, Tübingen.
Tarski, Alfred and Robert Lawson Vaught (1957): 'Arithmetical Extensions of Relational Systems', *Compositio Mathematica* **12**, 81–102.
Villalta, Elisabeth (2003): 'The Role of Context in the Resolution of Quantifier Scope Ambiguities', *Journal of Semantics* **20**, 115–162.
Villalta, Elisabeth (2006): Context Dependence in the Interpretation of Questions and Subjunctives. PhD thesis, Universität Tübingen.