to appear in R. Oehrle and G-J. Kruiff (eds.), *Binding and Resource Sensitivity*, Kluwer Academic Press.

Binding without pronouns (and pronouns without binding)*

Pauline Jacobson
Department of (Cognitive and) Linguistic Science(s)
Brown University
pauline_jacobson@brown.edu

1. Setting the stage

1.1. Variable-free semantics and the hypothesis of direct compositionality

This paper is rooted in the hypothesis of "direct compositionality" (cf., Montague, 1973). In its broadest formulation, this hypothesis is that the syntax and semantics work in tandem in such a way that the syntactic combinatory operations specify a set of well-formed expressions and each such operation is coupled with a semantic operation which provides a model-theoretic interpretation for the expression. I also assume a general Categorial Grammar (CG) implementation of this, whereby the syntactic category of an expression encodes its combinatory possibilities and its semantic type and where the semantic operation associated with any syntactic operation is predictable. (Moreover, as in most modern versions of CG, I assume that there is a small and very general set of syntactic / semantic operations - although the question at issue here is just how small.) Of crucial importance is the claim that each syntactic expression is <u>directly</u> assigned a model-theoretic interpretation, and hence no expression needs to be mapped into some other level such as LF in order to be interpreted.

* A version of this paper was presented at the symposium on Grammatical Inference and Resource Sensitivity at the Formal Grammar Conference, Utrecht, 1999. I would like the thank the participants in the symposium and especially the organizers, Geert-Jan Kruijff and Dick Oehrle. I would also like to especially thank David Dowty and Anna Szabolcsi for discussion over the years of direct relevance to the question in this paper. This research was supported by NSF grant SBR 98-50552.

In a series of papers, I have argued that this program is most easily implemented if one also adopts a "variable-free semantics"; the variable-free view has its roots in Combinatory Logic (Curry and Feys, 1958); it was also proposed for natural language in Quine (1966), and has been explored recently in much of the Categorial Grammar literature. By this, I mean a semantics which makes no essential use of variables - variables can (and here, will) be used to represent meanings but this is for notational convenience only. (One can instead use combinators in the notation, but such notation is not very reader-friendly.) To elaborate on this: the "standard" view is that the meaning of any linguistic expression is relative to some way to assign values to variables (i.e., it is relative to an assignment function). Put slightly differently, the meaning of any expression is a function from assignment functions to something else, where each assignment function in turn is a function from variable names to "normal" model-theoretic objects (such as individuals). An "open" expression - such as an expression C which contains a pronoun unbound within C - can be a non-constant function from assignment functions, while a "closed" expression will always be a constant function from assignment functions.

In the variable-free view, there is no role for assignment functions and hence also, of course, no role for variables. The meaning of any linguistic expression is simply some normal, healthy model-theoretic object - something constructed only out of the "stuff" that any theory presumably needs: individuals, worlds, times, perhaps events, etc. In my previous papers I have argued that such a view not only simplifies the model-theoretic apparatus, but is also a great boon to the hypothesis of direct compositionality. Space precludes detailing this point here (see Sec. 3 where one relevant example is fleshed out in full detail), but the basic observation comes from the fact that many of the traditional arguments for LF center on binding phenomena. More specifically, many such arguments rely on the assumption that "binding" is a relationship between two actual syntactic objects - a binder and a bindee - which must find themselves in a particular configurational relationship to each other. But on the surface the relevant configurational relationship is often absent - hence an LF is posited in which things are moved around and/or extra elements added in order to get the right binder/bindee relationships. However, I have argued in, e.g., Jacobson (1999, 2000b) that this kind of motivation for LF is actually an artifact of a particular view of binding based on variables and binders. Once we move to a variable-free semantics, many of these threats to direct compositionality disappear.

1.2. But where is the binding effect located?

But there are several possible ways to implement the variable-free program, and so the present paper is devoted to the question of just how pronominal binding is accomplished. Does the grammar contain a type-shift rule for this, or is pronominal binding instead built into the lexical meaning of the pronoun? The first is the approach that I have used in my work (see, e.g., Jacobson 1991, 1994, 1999, 2000b etc.), while the second has been proposed in Szabolcsi (1987, 1992), Dowty (1999), Morrill (2000) and others. (But see Szabolcsi (this volume) for an approach which instead makes use of the first tack.) The second tack - building binding into the pronoun meaning - is at first glance a tempting one, especially under any kind of "radical lexicalist" view of grammar which seeks to keep the syntactic and semantic combinatory operations as minimal as possible and put much of the work into the lexical meanings and types. The tradition which has come to be known as Type Logical grammar, for example, has generally opted for this approach; thus Dowty's (1999) proposal is rooted within this tradition. (Although Jaeger (1997, this volume) presents a type-logical analysis which does not do this and which is much closer to my analysis.) And even in a theory which does allow type-shift rules, Szabolcsi points out to me that there is a certain appeal in building binding into pronoun meaning. After all, any theory must assign some meaning to a pronoun, and so if binding can be accomplished in this way then nothing more needs to be said.

Nonetheless, this present paper will argue for a type-shift view of binding, and will show that the required machinery for this is actually quite simple. As is implied by the title of this paper, the arguments for divorcing binding from pronoun meaning center on cases where there are binding effects but with no pronoun on which to hinge the responsibility (nor any other material whose lexical meaning can in any obvious way be held responsible for the binding effect). The reverse is true as well: there are pronouns without binding - that is, free pronouns. In the approach here nothing extra needs to be said about these, while these are actually somewhat of a nuisance for the view of binding as pronominal meaning.

Of course if the empirical facts discussed in this paper forced us into an overly complex or baroque set of type-shift rules, then we certainly might question the force of the evidence. But I hope to convince the reader that the apparatus needed to do binding by type-shift is extremely simple, elegant, and natural - and this small amount of apparatus leads to simple accounts of a wide array of phenomena (in addition to the ones discussed in this paper, see Jacobson 1994, 1998, 1999, 2000a, 2000b for several others).

The remainder of this paper is thus structured as follows. In Sec. 2.1 I briefly outline the system that I am assuming, and Sec. 2.2 compares this to what we can call the binding-as-pronoun-meaning view. Secs. 3 and 4 detail two cases which support the type-shift view: one centers on Antecedent Contained Deletion and the other on certain exceptions to i-within-i effects. Sec. 5 briefly mentions a few additional cases which look like they would be problematic for the binding-as-pronoun-meaning view, although I do not have space to thoroughly demonstrate that here.

2. Type-shift vs. Binding-as-pronoun-meaning

2.1. The type-shift approach

We give here a brief sketch of the system as put forth in Jacobson (1999); the interested reader can consult that paper for further details. First, assume that any expression containing a pronoun which is unbound within that expression (including a pronoun itself) is actually a function from individuals to something else. (If an expression contains two unbound pronouns then it is a function from two individuals to something else.) Thus, for example, the italicized expression in (1) is a function from individuals to propositions (and is essentially the function lost') while the italicized expression in (2) is a function from individuals to individuals:

(1) Every man$_i$ believes that *he$_i$ lost*. he-lost' = λx[lost'(x)] (= lost')

(2) Every man_i loves *his_i mother*. his-mother' = λx[the-mother-of'(x)]

This raises several questions. What does the pronoun itself mean? How does it - or material containing it - combine with other expressions? And, how do we ultimately derive the binding effect?

First, a word about the syntax. I assume the general CG view which tightly couples the syntax and the semantics and according to which the semantic type is predictable from the syntactic category. Since an expression like *his mother* does not denote an individual but rather a function of type <e,e>, its syntactic category cannot be NP. (I assume that NPs are individual-denoting expressions and that expressions of type <<e,t>,t> are of category S/(S/NP). To this end, I adopt the following convention: For any two categories A and B, there is a category A^B, and the semantic type of A^B is a function from B-type meanings to A-type meanings. Thus A^B is of the same semantic type as the normal Categorial Grammar type A/B but its syntax is somewhat different. It is not an expression which actually wants to combine in the syntax with an expression of category B to give an expression of category A; rather the superscript feature in essence records the fact that there is an unbound pro-form within the expression. This will all become clearer momentarily.

Thus a pronoun itself such as *he* has a meaning of type <e,e> - and in particular its lexical meaning is the identity function on individuals. (This is oversimplified: presumably gender also plays a role in the semantics and so *he* is actually the identity function on male individuals. But for simplicity I will ignore the role of gender throughout.) Similarly, its syntactic category is not NP, but rather NP^{NP}.

What allows this to combine with normal expressions wanting NP arguments is a type-shift rule which I have dubbed **g** (since its semantics is what is often referred to as the "Geach" rule; Geach (1972)). This allows any expression wanting an argument of any category A to instead take an argument of category A^{NP} and it "passes up" the information that there is an unbound pronoun within it. A few notational points before spelling this out: I use $A/_RB$ to mean an expression which wants to combine with a B to its right to give an A and $A/_L B$ to mean an expression wanting a B to its left. (In addition, I assume that there are wrap operations; $A/_W B$ will be an expression which wants to "wrap around' its argument while $A/_I B$ is an infix, but these will play no role in the material here.) I will generally suppress the directional subscript feature whenever it is obvious or unnecessary. I will, moreover, represent any linguistic expression as a triple of the form <[(roughly) phonological form]; syntactic category; meaning>.

Thus we assume that the grammar contains the following rule shifting the syntactic category and meaning of expressions:

(3) The **g** rule: Let α be an expression of the form <[α]; A / B; α'>. Then there is an expression β of the form <[α]; A^C/B^C; λf[λc[α'(f(c))]] (for f of type <C',B'> and c of type C').

Actually a very slight generalization of (3) is needed to cover cases of expressions containing more than one pronoun; the interested reader can consult Jacobson (1999) for details.

The derivation of *he lost* is now straightforward:

(4) he; NP^{NP}; λx[x]

lost; S / NP; lost' --->_g lost; S^{NP}/NP^{NP}; λf[λy[lost'(f(y))]]

he lost; S; λf[λy[lost'(f(y))]](λx[x]) = lost'

The derivation of *his mother* is somewhat more complex to document since it requires spelling out the semantics of genitives. We will not do this here, but will take this simply to be NP^{NP} with meaning <u>λx[the-mother-of'(x)]</u> or, more simply <u>the-mother-of'</u> function.

Binding is accomplished by a second type-shift rule, which I call **z** and which is spelled out in (5):

(5) The **z** rule: Let α be an expression of the form <[α]; (A / NP) / B; α'>. Then there is an expression β of the form <[α]; (A / NP) / (B^{NP}); λf[λx[α'(f(x))(x)]]>.

(This rule is actually also given in somewhat more general form in Jacobson (1999); the more general form will, for example, allow for binding "across" an argument position and is needed for the case of 3-place verbs. One may also wish to generalize this in such a way that NP above is replaced by a variable over categories; this will allow binding of other types of proforms.) Thus this rule takes something which wants as argument an expression of category B and a higher NP argument, and it shifts it into a new expression which wants a pronoun-containing B-like expression as argument. Moreover, since the expression B^{NP} which it will take as argument is a function from individuals to B-type meanings, it binds that individual-argument slot to its higher NP-argument slot. So, for example, take the derivation of (2) *(Every man_i loves his_i mother)*. loves' is an ordinary relation between two individuals. But applying (5) yields **z**(loves') which is a relation between individuals and functions of type <e,e>, such that to **z**(love') a function f is to be an x who loves' f(x). Thus when this takes the-mother-of function as object, the meaning of *loves his mother* is λx[love'(the-mother-of'(x))(x)] and this then is argument of the subject generalized quantifier.

The full system requires one more rule which is commonly assumed in much other work: I assume that expressions can freely type-lift (over any category). I will make little use of this here, except to note that subject NPs like John will often be lifted into their generalized quantifier counterparts. I will notate the lift operation as **l**.

This is all that is needed to handle all cases of binding that I know of. In Jacobson (1999) I document that this system (with only slight generalizations of **g** and **z**) handles cases of any number of pronouns bound by any number of binders and in any order. For example, we can get both nested and crossed patterns of binding - the differences come just in the order of applications of **g** and **z**. We also easily get cases of more than one pronoun bound by the same thing, as in:

(6) Every $woman_i$ thought that the man who fed her_i dog should love her_i dog.

In a variable-free system one might wonder how the two pronouns here can in the end correspond to the same thing - after all, they do not have variable names and so cannot be the same variable. Indeed, they themselves mean only the identity function and therefore they have no way of knowing that they will be "co-bound". But nonetheless the appropriate meaning is easily derived. The two pronouns are, in effect, semantically "merged" via two applications of **z** on *thought*.

While space precludes spelling out the particulars here, there is one related point worth detailing. This is that nothing extra needs to be said for the case of the meaning of a free pronoun. Consider, for example, the case of a sentence with an unbound pronoun, such as

(7) He lost

Under the system here, this is actually an expression of category S^{NP} and its meaning is of type <e,t> rather than of type t. But we can assume that in order to extract propositional information from this, a listener will apply this function to some contextually salient individual. While it may seem counterintuitive at first to think of (7) as denoting a function from individuals to propositions rather than a proposition itself, it should be pointed out that the standard view has no real advantage here. After all, under that view the meaning of (7) is also not a proposition, but rather a function from assignment functions to propositions, and so the listener presumably applies this to some contextually salient(?) assignment function in order to extract propositional information. (In fact, all sentences are actually functions from assignment functions, but in the case of a closed sentence it makes no difference which assignment function is chosen.) There is, then, no sense in which the standard view has any advantage.

The important point is that we do not need to posit any kind of lexical ambiguity between free and bound pronouns - a happy result since the full set of such pronouns are morphologically identical, and it would thus be quite suspicious to treat free and bound pronouns as cases of accidental homophony. Moreover, as far as I know this fact is true cross-linguistically, although I do not have the cross-linguistic expertise to assert this with full confidence (I thus leave this as a question for further research). Additional evidence against the accidental homophony view is discussed in fn. 13 below. This, then, is what motivates the subtitle to this paper: *pronouns without binding.* Under the system proposed here, the existence of free pronouns (and the fact that they are identical to bound pronouns) is completely unsurprising. Because a pronoun itself does not ask to be bound, the free reading comes "for free".

One further related point. In Jacobson (2000a, 2000b), I document that the paycheck reading of a pronoun also comes for free under the system here. Space precludes details, but the basic idea is that the ordinary meaning of a pronoun (the identity function on individuals) can undergo the **g** rule which will map it into the identity function on functions of type <e,e>, and this is what will yield paycheck meanings for pronouns. (I am grateful to Mark Hepple for pointing this out to me.) And Szabolcsi (this volume) shows that the view of pronouns as identity functions can be extended to cover cases of cross-sentential binding.

2.2. Binding as pronoun meaning

2.2.1. The analysis

The major alternative to this that has appeared in the literature is to build the binding effect into the pronoun meaning.[1] I believe that accounts of this type will be roughly similar. - the pronoun will take one or more argument(s), and one of these will be a function of type <e,X> where the e-argument slot here is the "binder". The pronoun meaning will thus "merge" that slot with the argument slot that it occupies. To make this more concrete, I will illustrate with the account of Szabolcsi (1992) since it is cast within the same general framework as is assumed here. On her account, the category and meaning of a pronoun is:

(8) (VP/(VP/S))/(S/NP); $\lambda g[\lambda f[\lambda x[f(g(x))(x)]]]$ (Note that crucially the slash directionality is unspecified here.)

Take something like

(9) Every man_i thinks that he_i left.

[1] There is at least one other logical possibility as to how to accomplish binding (a possibility which was brought to my attention by Matthew Stone). This is to treat pronouns as I have done here, but to have a type-shift rule which does not type-shift the verb but rather the binder itself. I have not explored in any detail, though far as I can see, most of my results will work equally well under this view. A problem with this view, however, will come from the kind of binding found in examples like (41) which are discussed in Sec.4.1, see fn. 11. In any case, though, I know of no actual advantage to this approach, and it still requires a type-shift rule.

Here *he* combines with a VP (an $S/_L NP$) to give something which takes as argument *think* and then "merges" the subject position of *think* with the argument of *left:*

(10) he left; VP / (VP / S); λf[λx[f(left'(x))(x)]]

thinks (that) he left; VP; λx[thinks'(left'(x))(x)]

Hence the "duplicator" aspect is built into the meaning of *he* rather than into the operation on the meaning of *thinks*. Things work similarly for the case of an object pronoun as in (11):

(11) Every man_i thinks that Mary likes him_i.

The difference is simply that here Szabolcsi assumes that *Mary likes* can compose first to give an $S/_R NP$; the rest of the derivation will be as above. In cases of longer distance binding, the idea is presumably that function composition will allow a derivation analogous to (10). Take for example

(12) Every man_i thinks that Mary knows that he_i left.

If we compose *thinks that Mary knows* into a complex transitive verb then this will work exactly as in (10). If one does not like to make use of function composition in these derivations, the entire proposal can be translated into a view in which strings are put together in the more "normal" way and then pronouns infixed in at appropriate points. See, for example, Dowty (1999) for an implementation along these lines within a Type-Logical framework. Dowty's proposal is slightly different from that of Szabolcsi, but the major points of relevance in this paper carry over to his as well.

2.2.2. Principle B

There is one apparent advantage to Szabolcsi's original approach over mine which should be mentioned before moving on. As detailed in Szabolcsi (1992), it gives a way to incorporate Principle B effects. (See also Morrill, 2000.) The key here is that the occurrence of S in the category above means that the pronoun will get bound only if it is within an expression which occurs with some other expression that wants an S as argument. This builds in the "anti-locality" effect of pronouns - the binder will have to be at least one clause up from the pronoun.

(This is of course too strong - Szabolcsi's use of S in the pronoun category as a way to give principle B effects incorrectly rules out cases like *Every woman$_i$ loves her$_i$ mother*. But this can be trivially solved by revising the category to (VP / (VP / Y)) / (Y / NP)), for Y a variable over S or NP. Thus the occurrence of *mother* which takes a genitive argument is presumably of category NP /$_L$ NP [GEN]. *Her* can thus take this as argument and return a VP / (VP / NP) which in turn takes the transitive verb *loves* as argument.)

In my account, on the other hand, there is no provision made for Principle B effects. I will nonetheless argue that the bulk of the evidence overwhelmingly favors the type-shift approach. But what to say about Principle B? I am unsure. Indeed it could be built in to the type-shift approach with a clever enough use of features (I leave it to the interested reader to work this out), but I am not convinced that building it into the grammar is really the right move. The problem stems from the well-known and much discussed fact that Principle B effects are not only effects blocking "binding" relationships, but that the effect must also block the "accidental coreference" reading in something like (13):

(13) *He$_i$ / John$_i$ loves him$_i$.

No account of Principle B in terms of a constraint on binding will rule out the relevant reading of (13). (See, e.g., discussion in Grodzinsky and Reinhart, 1993.) One would hope, then, that whatever rules out the relevant reading of (13) will also rule out the "bound" reading of (14), and thus we would not need any grammatical condition on binding here:

(14) *Every man$_i$ loves him$_i$.

There is much more to say, for Szabolcsi (1992) is well aware of this point and in fact argues that (13) and (14) should not be collapsed. Her reasons (and that of other researchers, such as, e.g., Grodzinsky and Reinhart, 1993) center on evidence from Wexler and Chien (1991) to the effect that children at some stage allow (13) while disallowing (14). But since we don't know just what the children are doing (despite the massive amount of ink which has been dedicated to this point), I don't see that we can draw any confident conclusions from the child evidence. On the other hand, if there is a completely separate principle ruling out (13) and (14) it seems somewhat surprising that the two principles are concerned with exactly the same configuration(s). (Reinhart (1999) attempts to explain just this by having one principle piggy-back off the other. But her explanation is highly transderivational, and attributes an extreme degree of intelligence and complexity to the processing mechanism - i.e., to that mechanism which tries to compute the accidental coreferential reading in (13). Under her explanation, this mechanism ultimately knows not to allow accidental coreference here in virtue of computing that the grammar blocks the bound reading. Such an explanation seems to involve an extremely complex view of processing. For the processor doesn't just chug along and compute meanings in accordance with a basic set of rules. Rather, it must also consult whether these meanings are equivalent to meanings that result from alternative derivations, and indeed it must even check on alternative <u>bad</u> derivations.) The upshot, then, is that Principle B effects seem to be quite poorly understood.

2.2.3. <u>Pronouns without Binding</u>

Leaving Principle B effects as an unsolved mystery, the bulk of evidence seems instead to favor the type shift approach. Our first point of comparison concerns the case of free pronouns. As shown above, the type-shift approach handles these effortlessly, and does not need to posit accidental homophony here. Can the same point be made for the binding-as-pronoun-meaning approach? Perhaps, but only with some difficulty. Note first that Szabolcsi (1992) (and Morrill, 2000) simply treats free pronouns as simply being distinct from bound pronouns (in particular, as free variables). Sec. 2.1. has already argued against any tack like this, pointing out that since free and bound pronouns always look the same, it is suspicious to treat them as merely accidental homophones. Thus, the reader should bear in mind that the discussion below is not addressed to Szabolcsi's actual proposal but rather to a "straw man" based on an attempt to extend her account of bound pronouns to free pronouns.

The question, then, is whether the binding-as-pronoun-meaning approch could in fact treat free pronouns as the same creatures as "bound" pronouns where the free pronouns simply happen never to get their binder. Again for concreteness, I will use a Szabolcsi (1992) style approach to illustrate (in fact, the other possible approaches that I can think of which build binding into the pronoun meaning simply have an even harder time accommodating free pronouns). Under that approach, the answer is to the above question is yes, but the story is very complicated and not very satisfying. Take the case of a free pronoun as in a sentence like *He lost*. Here *he* can indeed take *lost* as argument, and this expression is of category VP / (VP / S) with meaning of type <<t,<e,t>>,<e,t>>. Recall that in the type-shift view *he lost* also does not denote a proposition, but rather a function from individuals to propositions and so the listener must apply this to a contextually salient individual in order to arrive at propositional information. On both accounts, then, this is "incomplete". But under the binding-as-pronoun-meaning approach, the meaning of *he lost* is far more complex, and much more needs to be supplied. For here the listener needs to supply as argument a contextually salient verb-type meaning (such as <u>think'</u>) as well as a contextually salient individual. I see no way to really motivate the former step. In a discourse context where, for example, John is salient, it is fine and quite normal to utter *He left*. There is no mystery about how we infer a proposition from this and no mystery about how we supply the contextually salient individual. But the binding-as-pronoun-meaning approach requires us not only to find an individual <u>j</u> but also some 2-place relation (<u>think'</u>, <u>say'</u>) or whatever - a step which does not seem to be motivated.

3. <u>The interaction of Antecedent Contained Deletion (ACD) and Binding</u>[2]

We will now take a detailed look at two cases arguing for binding as a type-shift; a few others will be mentioned in Sec. 5 (see also the discussion in Szabolcsi, this volume). The first detailed case study centers on the interaction of Antecedent Contained Deletion (hereafter, ACD) and binding. In Sec. 3.1, I provide a thumbnail sketch of how this interaction argues for

---

[2]I want to thank Anna Szabolcsi for pointing out to me the relevance of my analysis of binding and ACD for the question addressed in this paper. The gist of the analysis of ACD and binding was originally developed in Jacobson (1992b), but the present analysis improves upon the original one in a number of the details.

locating the binding effect in a type-shift rule rather than in the meaning of the pronoun. I will, however, suppress certain details in this section, and so Sec. 3.2, comes back to these details.

3.1. A sketch of the analysis

3.1.1. Background: Ordinary ACD

3.1.1.1. The standard wisdom on ordinary ACD: how this argues for a level of LF

The central question in this paper is whether or not binding in a variable-free semantics should be located in a type-shift rule or in the lexical meaning of the pronoun. But before turning to this, I would like to digress to walk step-by-step through the standard analysis of VP Ellipsis, relative clause semantics, and ACD. The reason is that a considerable amount is at stake here: this whole domain has played a very critical role in the "received view" that the hypothesis of direct compositionality cannot be correct and that a level of LF is required. But a careful look at the assumptions underlying the standard analysis reveals that some of these assumptions have little foundation - and once these are abandoned we can see that this domain does not threaten direct compositionality. It turns out, though, that the evidence against direct compositionality completely vanishes only if we adopt a version of variable-free semantics which locates binding in a type-shift rule, and this fact will then return us to the main point of the paper.

We thus begin with a consideration of the semantic composition of an ordinary relative clause, as in (15):

(15) John read every book which Bill will read.

First, we need to make some decision about how to handle the auxiliary. In much standard work, it is actually assumed that the auxiliary is "raised" at the level which inputs the semantics and is therefore actually a sentence modifier. However, in order to keep the differences between the standard account and the CG account to a minimum and to focus only on those differences which are crucial for the point at hand, let me assume that will' is a VP modifier. Since for convenience I am extensionalizing throughout this paper, we will represent the meaning of *will* as <u>λP[λx[**F**(P(x))]]</u> (for **F** the future operator). (With the caveat that obviously the argument of **F** should be an intensional object.) (It is also quite possible that the type is right here, but that the meaning of *will* is considerably more complex than the above - which is nothing more than a fancy packaging of ordinary **F** - but this too is orthogonal to the points at hand.)

Continuing with (15), the conventional wisdom regarding its semantic composition is that the meaning of the material following *which* must at some point be an open proposition (a proposition with an unbound variable within it), and the open variable is then λ-abstracted over. In other words, the meaning of *Bill will read* is - at some point in the semantic composition - <u>will'(read'(x))(b)</u>. This is then mapped into the property <u>λx[will'(read'(x))(b)]</u>. And finally, this property is intersected with the meaning of *book*. [3] Now let us telescope in, and look in detail at the composition of the meaning of the material following *will*. The usual view here is that the meaning of this expression is <u>read'(x)</u> and not just <u>read'</u>. (Of course the usual thinking on this is that the material following *will* in the syntax is not just an ordinary transitive verb *read* but is rather the VP *read* t_x, where the meaning of the trace t_x is what supplies the variable in the semantics.)

---

[3]There is another analysis available, in which *every book* first combines and this then combines with the meaning of the relative clause. (Following Stockwell, Schachter, and Partee (1968) we can refer to this as the "NP-S" analysis, and the analysis above as the "Det-Nom" analysis.) Indeed there was considerable discussion in the late 70's and early 80's as to how to compositionally maintain a "NP-S" analysis; see in particular Partee (1976) (who argued against the NP-S analysis on the grounds that it could not be done in a straightforward compositional way) and the subsequent replies in Bach and Cooper (1979) and Jacobson (1982) (both of whom supply compositional ways to implement the NP-S analysis. Mine, incidentally, turns out to be a variable-free version of Bach/Cooper analysis (they crucially use a distinguished variable), though at the time this was not rooted in the variable-free program in general. In any case, should the NP-S analysis turn out to be correct this will not affect any of the rest of the discussion in this section: it still is the case that the meaning of *which Bill will read* is ultimately of type <e,t> and so what happens in the semantic composition "higher" than this is not relevant here.

From whence comes this conventional wisdom about the semantic composition? As far as I can tell, the motivation behind this is the (sometimes unstated) assumption that the only (or at least the usual) method of semantic composition when two expressions combine is functional application. Thus since will' is of type <<e,t>,<e,t>>, it follows that the material with which it combines must be of type <e,t> in order to be of the right type to serve as argument. (As noted above, many standard practitioners do not actually assume that will' is of type <<e,t>, <e,t>>. Rather, many assume that *will* is "raised" at LF and thus has a meaning of type <t,t>. But even if we take this view we ultimately come to the same conclusion. In this case, it follows that the meaning of *Bill read* is of type t in order to be the argument of will', and this in turn means that the meaning of the material following *Bill* must be of type <e,t> and thus must contain a variable in the object position.)

Before continuing, there is one further assumption which should be spelled out and which is made in almost all work on VP Ellipsis (and on ACD). This is that the "missing" meaning in an ellipsis construction must be the meaning of some other linguistic expression (call this the "antecedent"). It follows from this that VP Ellipsis in general and ACD in particular can be used as a diagnostic for the meaning of the other linguistic expression: in general ACD will require us to find some expression which can serve as "antecedent" and which can thus supply the appropriate meaning. Indeed, I will discuss this in somewhat more detail below; for now simply assume that it is correct.

We are now in a position to consider the implications of ACD, a phenomenon noted originally in Bouton (1970) and exemplified in (16):

(16) John read every book which Bill will Ø.

The traditional tale runs as follows. Since will' must find some meaning of type <e,t> as its argument, then something somewhere will have to supply a VP-type meaning. (Again, the argument can be made in a slightly different form: if *will* raises and has a meaning of type <t,t> then the details run somewhat differently but the story at the end of the day is the same. That is, if follows that the meaning of *Bill Ø* must be of type t (where Ø represents the ellipsis site) and so *Bill* will have to combine with a VP-type meaning). But then, what supplies the "missing" meaning? Obviously it cannot be the surface matrix VP, since this is exactly what leads to the Antecedent Containment paradox - an attempt to supply the "missing" meaning by considering the meaning of *read every book which Bill will* leads to an infinite regress. What saves the day is the assumption that there is a level of representation at which the object NP is scoped out leaving a variable (and/or trace which translates as a variable) in object position. Thus this means that there is a level of representation at which there is an expression whose meaning is read'(x), and so this can be supplied as the missing meaning. The rest of the semantic composition proceeds in essentially the same way as the composition of (15) above. The key conclusion: there must be a level of representation (LF) at which the object NP is scoped out and this in turn dooms the hypothesis of direct compositionality. Or so it seems.

3.1.1.2. The variable-free analysis of relative clauses in CG

But of course this entire tale depended on the initial assumption that the semantic composition of the expressions in question necessarily involved functional application. Within the Categorial Grammar and related literature, however, it has long been known that certain pieces of the above story can be abandoned. For starters, Steedman (1987) and others point out that we could compose the meaning of (15) by using function composition instead of functional application. Thus assume that there is no trace and/or variable in the object position following *read* in (15), and take *read* to have as its meaning the 1-place relation read' of type <e,<e,t>>. Then this meaning can function compose with will' to give λx[will'(read'(x))] which can in turn function compose with the type-lifted meaning of the subject. The meaning of *Bill will read* thus ends up as the property λx[will'(read'(x))(b)] - just as it does in the usual view. Notice, then, that this differs from the standard account in being variable-free: we never supply an unbound variable as object of read' and then later λ-abstract over this variable.[4]

But once we recast the semantic composition of (15) in this way, then there also is no reason to analyze the ordinary case of ACD as involving a level of representation at which the object is scoped out; this point was noted first by Cormack (1985) and later in Evans (1988) and Jacobson (1992a). Thus, Cormack points out that ACD is perfectly compatible with direct interpretation: the "missing" material can simply be the two-place relation read' and this combines with the meaning of *will* in (16) just as it does in the semantic composition of the full case in (15). Since will' combines with a meaning of type <e,<e,t>> and not a VP-type meaning, the argument for scoping out the object disappears. We will call this the ACD-as-TVP (transitive verb phrase)-ellipsis hypothesis.[5] Of course there is more to say: it remains to show how this will account for many of the known generalizations concerning ACD such as those discussed in Sag (1976) and Larson and May (1990). We do not have space for a discussion of this here, but see Cormack (1984) and Jacobson (1992a) for a demonstration of the fact that many of the known generalizations plus some other interesting ones do in fact follow from the CG account.

---

[4]Incidentally, it is not even crucial to discard the assumption that all binary combinations involve functional application, since function composition can always be broken down into the two steps of the "Geach" rule followed by application. In fact, I will do just this in Sec. 3,2, For convenience, I will continue to analyze extraction constructions as directly involving function composition in the semantics rather than **g** plus application, but this is for expository convenience only.

[5]A related proposal is made in Lappin (1996) who treats ACD as an instance of pseudo-gapping (where a trace remains in object position). There are, incidentally, some differences between pseudo-gapping - which is also a kind of TVP ellipsis - and ACD, and an anonymous reviewer for SALT 8 pointed out to me that this fact is potentially problematic for my analysis. I am hopeful that a full analysis of the syntax of both ACD and of pseudo-gapping will account for these differences, but this is only a promissory note at this time.

3.1.2. ACD-like phenomena with pronouns

3.1.2.1. The standard story

But the story does not end here: we still have an obstacle to surmount in order to maintain direct compositionally for this entire domain. For if one considers the full range of ACD cases discussed by Bouton, one immediately notices an apparent problem for the ACD-as-TVP-ellipsis hypothesis. This derives from cases where it appears that the "missing" meaning must indeed be some kind of full VP meaning, and where a paraphrase in which the ellipsis site is spelled out will involve a VP that contains a pronoun. A typical example is (17); note the relevant (and most natural) meaning is one whose non-elliptical paraphrase (shown in (18)) contains a full syntactic VP with a pronoun in object position:

(17) John kissed every girl who thought that he would.

(18) John kissed every girl who thought that he would *kiss her*.

The standard view of the meaning of *kiss her* is that it is kiss'(x) - and so the conclusion is that the "missing" material here is a full VP meaning (which contains a variable) and not just a TVP meaning. More specifically, it appears that the missing meaning is kiss'(x) and that the variable x ultimately gets bound in the same way that pronouns ultimately get bound. However the variable corresponding to the meaning of *her* is bound in (18), the same process will bind x in the VP meaning which is supplied in the ellipsis site in (17).

But alas, we are then led back to the conclusion that there must be a level of LF at which the object of the matrix is scoped out. The logic is exactly the same as the logic documented earlier. Here there is no overt VP whose meaning could possibly be kiss'(x) - we can get that to be the meaning of some linguistic expression only by positing a level of representation at which the object is scoped out leaving a VP whose representation at LF is something like *kiss* t_x and whose meaning is kiss'(x).

3.1.2.2. Variable-Free to the rescue

Fortunately, it is an illusion that a case like (17) requires a level of LF; the illusion derives from the illusion that pronouns have the meaning of unbound variables which get bound somewhere along the line. As discussed in Jacobson (1992b), the entire problem surrounding (17) evaporates once one moves to a variable-free semantics - at least one of the sort where the binding of the object argument position of kiss' can be accomplished by a type-shift rule which operates on some material which is overt in (17). For example, there is no problem with this case given the existence of the **z** rule.

To demonstrate, notice that under the variable-free semantics sketched in Sec. 2.1, the meaning of the paraphrased VP in (18) *kiss her* is exactly the same as the meaning of *kiss'* - and so here too this is a case of TVP ellipsis. Thus the missing meaning is just kiss', and this is supplied by the meaning of the matrix transitive verb. The object position of kiss' is bound by the application of **z** on think' (this is spelled out below in (22)). It is true that the non-elliptical paraphrase syntactically contains a pronoun in object position rather than a bare transitive verb - but this is for syntactic and not semantic reasons. It is simply the case that nothing in the syntax sanctions the syntactic composition of (19) as opposed to (18), and we will discuss this more fully in Sec. 3.2:

(19) *John kissed every girl who thought that he would kiss.

But nothing in the semantics goes awry if we simply supply the 2-place relation kiss' as the missing meaning, since this meaning is exactly the same as the meaning of *kiss her* (modulo the contribution of gender).

To more fully spell out the details, let me first remove one potential confound. Notice that (17) also contains a pronoun as subject of *would kiss;* a fact which would complicate the exposition but which is irrelevant to the points here. The fact that examples like (17) are generally bad when there is a full NP instead of a pronoun in this position has received some discussion in the literature (cf., Haik, 1987)[6], but I suspect that it derives from focus

---

[6]I cannot resist the temptation to note that an unpublished squib by Jacobson, Schmerling, and Wall in 1980 also pointed out that examples of this type are bad unless there is a pronoun in subject position. Thus we discussed the following contrast :

considerations (for discussion of a different but probably related case, see Heim, 1997). Note that - modulo a very mild "Principle C violation" effect - (20) is also good with an unstressed NP in subject position, and so I will change the example accordingly:

(20) ?John will kiss every girl who thinks that John will Ø.

As further evidence for the fact that such examples do not actually <u>require</u> a pronoun in subject position but that rather focus is playing a crucial role, we can construct the following scenario. Imagine a t.v. game show, which we will call Pick Your Date. The rules are as follows. In secret, each woman picks a man who she wants to date. At the same time, each man writes down the name of one woman who he thinks will pick him. If A picks B and B correctly anticipates that A picks B, then they get to go out on a wonderful date, courtesy of the show. John, Bill, Mary and Sue are the contestants this week, but alas, no one wins. For:

(21) ??Unfortunately, Mary picked the man who thought that SUE would, and Sue picked the man who thought that MARY would.

While not perfect, (21) does not seem to be out of the question (several informants have confirmed my judment on this).

Hence to avoid the extra complication engendered by the use of a pronoun in subject position, I will modify the original example to the slightly more awkward one in (20). The full semantic combinatorics are thus spelled out in (22):

(i) a. *John kissed every girl who thought that Bill would.
b. John kissed every girl who thought that Bill would.

The squib was rejected by *Linguistic Inquiry* on the grounds that the facts were incorrect; I now agree with this assessment, so (21) shows that cases like (ia) can be ameliorated with appropriate focus considerations.

(22) will Ø; $\mathbf{g}(will')(kiss') = \lambda R[\lambda x[will'(R(x))]](kiss') = \lambda x[will'(kiss'(x))]$

John; $\mathbf{g}(\mathbf{l}(j))$ - $\mathbf{g}(\lambda P[P(j)]) = \lambda R[\lambda y[\lambda P[P(j)](R(y))]] = \lambda R[\lambda y[R(y)(j)]]$[7]

John will Ø; $\lambda R[\lambda y[R(y)(j)]](\lambda x[will'(kiss'(x))]) =$

$$\lambda y[\lambda x[will'(kiss'(x))](y)(j)] = \lambda y[will'(kiss'(y))(j)]$$

thinks; $\mathbf{z}(thinks') = \lambda P[\lambda x[thinks'(P(x))(x)]]$

thinks (that) John will Ø; $\lambda P[\lambda x[thinks'(P(x))(x)]](\lambda y[will'(kiss'(y))(j)]) =$

$$\lambda x[thinks'(\lambda y[will'(kiss'(y))(j)](x))(x)] =$$

$$\lambda x[thinks'(will'(kiss'(x))(j))(x)]$$

The rest of the composition depends on one's precise analysis of relative pronouns. One can, for example, assume that a relative pronoun takes an <e,t> denoting expression to its right and then an <e,t> denoting expression to its left and intersects them; the result is then straightforward.

3.1.2.3. <u>The punchline</u>

The punchline should now be evident: the ACD-as-TVP-ellipsis analysis of (17) and (20) goes through because all that needs to be supplied is a missing TVP meaning - and there of course is an overt TVP *(kiss)* which can supply the meaning. The object position of this 2-place relation is bound by **z** on <u>think'</u>. Could the analysis be mimicked under a view in which binding is built into the meaning of pronouns? I don't see how, at least not in any straightforward way. Consider first how the combinatorics works in the case of (18), which is the non-elliptical paraphrase. Under Szabolcsi's (1992) particular analysis, there actually is no VP meaning here: *John (will) kiss* first combines up and then occurs as argument of the pronoun. This, though, is partly an accident of her particular details, so let us assume that we can indeed combine *kiss her* to give the necessary meaning in which to accomplish binding; this would be:

[7]One might wonder what stops unlifted j from just occurring as argument of the above, which would ultimately yield the wrong meaning in which *John* is the kissee rather than the kisser. This will be blocked because of the syntax; the interested reader can verify this once the syntactic categories are fully spelled out in Sec. 3.2.

λx[λg[λy[g(kiss'(y)(x))(y)]]]. Thus in order for (17) (the corresponding ellipsis case) to have the required meaning, this meaning will somehow have to be supplied to combine with the meaning of *will*. But what can supply this? There is no overt linguistic expression anywhere else in the sentence which can reasonably supply it - for there is no pronoun to do the binding work. (Nor is there any other linguistic expression on whose lexical meaning we can pin the responsibility.)

Before leaving this, I should forestall one possible objection. It is well known that in the case of VP Ellipsis, there are occasional examples where the missing meaning is not the meaning of any actual linguistic expression but is just some pragmatically salient property. I will point out later that the same is true for the case of TVP ellipsis - with enough work one can construct examples in which the context supplies the missing 2-place relation. The fact that VP ellipsis is occasionally an instance of "deep anaphora" has been noted sporadically throughout the history of the literature, beginning with Schachter (1977) (relevant examples are also found in Webber (1978); Dalrymple, Shieber and Pereira (1991) and Hardt (1992). Just when the missing meaning can be inferred from the context rather than being supplied linguistically is not well-understood. But since this can sometimes happen, one might argue that the case at hand shows nothing, and that the meaning λx[λg[λy[g(kiss'(y)(x))(y)]]] is just inferred from the context. While I cannot prove that this is incorrect - inasmuch as I don't have a story to tell about just when we can supply meanings contextually - I doubt that anyone would really be comfortable with this way out. The fact is that cases where the meaning is contextually supplied are somewhat difficult to construct; they tend to either be rather conventionalized cases or they require a goodly amount of contextual "fiddling". But this doesn't seem to be the case here - we don't need to be terribly imaginative in our contexts in order to get the relevant meaning for (17) and (20). Moreover, the meaning that would need to be made salient by the context is an extremely complex one - we are not supplying a simple 2-place relation here but a VP meaning with a duplicator relationship built in, and it is hard to imagine how this would be made salient by the context. I thus think it is reasonable to assume that this is not a case of "deep anaphora".

3. 2. Spelling out the details

This is the argument in a nutshell. However, without a more precise analysis of ellipsis in general one might wonder just how firm the analysis is. That is, I have left open a number of questions. First, what does it mean to "pick up" a meaning? How can this notion be folded in compositionally? What does the syntax look like? For example, what ensures that the paraphrase for (17) requires *kiss her* rather than just the bare transitive verb *kiss?* (As noted above, this cannot be a semantic fact, since the two have the same meaning.) Since the analysis is plausible only if these points can all be spelled out in some satisfactory way, we now turn to doing just this.

The first question which needs to be dealt with is the following: just what does it mean to "supply a missing meaning" in VP Ellipsis in general? A variety of answers have been given here; let me discuss three possibilities. The first - and most common one - is that a full copy of the antecedent VP is actually represented in the ellipsis site at the level of representation which inputs the semantics. This analysis in turn comes in two varieties. In one, the relevant structure is interpreted and then a deletion takes place. This was the original view put forth first in Ross (1967) and in somewhat different form in Sag (1976), and it has also been resurrected in much current work such as Chomsky (1993), Fox (1993), etc. (Note that it is often assumed that deletion takes place "close to the surface" but requires identity of LF of the antecedent and the ellipsis site; cf. Sag, 1976.) In the other variety, (cf. Williams, 1977), the missing material is copied in from the representation of the antecedent. (The usual view is that the antecedent is first mapped into an LF representation, and this LF representation is then copied in.)

A second broad class of solutions here is that the missing material is interpreted in the way that bound pronouns are interpreted in general. This type of solution has been put forth from time to time, as for example in Rooth (1981), Szabolcsi (1991), Jaeger (1997) and Schwarz

(2000). In the view that I will ultimately adopt, nothing would stop some instances of VP ellipsis from being analyzed in this way as I will indeed be treating the auxiliary that precedes the "ellipsis site" as a kind of pro-form. (Which in turn means that there is no real ellipsis site - but I will sometimes continue to use this terminology for convenience.) However, it seems unlikely that all cases of VP Ellipsis involve binding. For, unlike the situation with normal pronominal binding, VP Ellipsis allows for backwards anaphora (and other kinds of Weak Crossover violations); the antecedent can be in another sentence, etc. In other words, there seem to be no structural constraints on the relationship of the anaphor to its antecedent, which is of course not the case for an ordinary bound pronoun. Thus even if some cases of VP Ellipsis do involve a kind of binding of the ellipsis site (or, more accurately, of the VP anaphor), it seems difficult to analyze all such cases this way.

A third type of solution is one which is much less popular, but will nonetheless be adopted here. This is to assume that there is no actual linguistic connection between the "antecedent' and the "ellipsis site" - there is no copying, nor any deletion under identity (nor, in general, is there binding). Rather, missing material is supplied in roughly the same way as for the case of free pronouns - the expression containing an ellipsis site is, at the end of the day, a function from VP (or TVP) type meanings to propositions, and it is applied by the listener to some contextually salient proposition. But of course we cannot simply leave the story here: we have argued above (and it has been known since Hankamer and Sag (1976)) that in general the relevant property (or 2-place relation) is one which really wants to be made salient by being the meaning of a linguistically overt expression rather than just being something inferred from the discourse context. I will comment directly on just why this might be so. For now, though, I want to point out that as long as we assume that the "missing" meaning is one which likes to be supplied linguistically, then much of the traditional wisdom about VP Ellipsis will carry directly over to the view here - even though we are not positing any actual grammatical relationship (copying or deletion) between the antecedent and the ellipsis site (or, the anaphor). Thus despite the lack of this kind of relationship, VP Ellipsis (and TVP Ellipsis) is

- as noted earlier - nonetheless a valid diagnostic for the meaning of the antecedent. (Again, all bets are, of course, off in the rare cases where we supply the missing meaning pragmatically, but we can see that such cases are rare and require either a good deal of contextual support or else tend to be somewhat conventionalized.) Incidentally, nothing in my overall analysis really hinges on this particular view of VP Ellipsis. Should it turn out that it involves deletion under identity of meaning then the argument in Sec. 3.1 still goes through - we still must adopt a type-shift view of binding rather than building it in to pronoun meaning.

Before continuing, then, let us look a bit more at the original observations of Hankamer and Sag (1976) to the effect that there is a difference between Ellipsis cases and clear "deep anaphors" like regular pronouns (such as *it)* as in:

(23) The garbage needed to be taken out,

(a) so John did it

(b) *so John did.

That is, VP Ellipsis likes to have its property made extremely salient by being the meaning of an overt linguistic expression, while this is not the case for ordinary free pronouns.[8] I have only a very preliminary speculation to offer as to why this is so. Assume, following Chierchia (1984) that there is a difference between a pure function of type <e,t> and its "individual correlate". Assume further that pure functions of type <e,t> are rather fragile objects - they are not easy to access and even once they are set up (by being the meaning of a linguistic expression) they do not hang around the discourse context for very long - unlike their individual correlates. (What remains to be explained is why this is so, since it is not so obvious just what is the real conceptual difference between a function and its individual correlate. Nor is it obvious why one

[8] Note that in much of the literature, *did it* in (23a) is called a "VP anaphor". This is misleading; the *do* in (23a) is presumably simply the main verb *do* - it has all the properties of an ordinary main verb, and *it* is its object, which picks up the contextually salient property take out the garbage. Thus the actual anaphor here is *it*..

cannot easily access the function given the individual correlate.) But if something like this is correct, it can help explain why VP Ellipsis likes (but does not absolutely require) an overt linguistic antecedent. In (23b) what is missing is an actual function of type <e,t>, while in (23a) what is missing is the individual correlate of such a function; we thus hypothesize that the latter is easier to retrieve from context. In a nutshell, then, the difference in the two types of anaphors is a consequence of their different semantic types Thus since "pure" functions of type <e,t> (as opposed to their individual correlates) are fragile objects, they are most easily accessed when made salient by being the meaning of some linguistic expression. Moreover, the expression which supplies the meaning will have to be close in the discourse context - this is consistent with the observation in Tanenhaus, Carlson and Seidenberg (1985) that the overt antecedent must be quite recent.

We can now spell out the full syntactic and semantic details. Assume that auxiliaries (and *to)* are listed in the lexicon as being of category $VP/_{R}VP$ - that is, they are VP modifiers. (Presumably they are specified as taking complements with certain features; this is needed to get the morphology right.) Thus for example, the auxiliary *will* is of this category and has the meaning (roughly) $\underline{\lambda P[\lambda x[\mathbf{F}(P(x))]]}$. Assume further that there is a rule mapping all auxiliary items of category VP / VP into members of category VP^{VP} with the same meaning. In other words, such auxiliaries now are not functions which syntactically want a VP complement, but are rather essentially pro-VPs. (Unlike pronouns they have more complex meanings than just the identity function - although this may well be the correct meaning for *do* - but they have the same syntax.) They thus combine in the same way that pronouns do. (As mentioned above, then, it is somewhat misleading to talk of this as "VP Ellipsis" and to refer to an "ellipsis site"; it would be more felicitous to use the term "VP Anaphora". I will however continue with the more traditional "ellipsis" terminology.) The information that these are proforms is passed up in the syntax and - just like the case of free pronouns - the final expression can contain an unbound pro-VP and hence be of category S^{VP} with meaning of type <<e,t>,t>. It will thus be applied to some contextually salient property (again, with the caveat that the relevant types of properties are difficult to access by inference and hence like to be made very available by being the meaning of an overt and a close VP). There are, moreover, focus considerations which need to be taken into account in a full analysis (see, e.g., Rooth (1992) for discussion), but we leave these out of consideration here.

Consider for example a 2-sentence discourse as in (24)

(24) Bill will leave. John will too.

Our concern is just with the second clause, and I will ignore the contribution of *too*. The basic idea, however, is quite simple and straightforward, and the composition proceeds as follows:

(25) John; $S/_R VP$; $\lambda P[P(j)]$ --->_g

John; $S^{VP}/_R VP^{VP}$; $\lambda T[\lambda Q[\lambda P[P(j)]](T(Q))]] = \lambda T[\lambda Q[T(Q)(j)]]$ (for T of type <<e,t>,<e,t>>

will; VP^{VP}; $\lambda P[\lambda x[\mathbf{F}(P(x))]]$

John will; S^{VP}; $\lambda Q[\lambda P[\lambda x[\mathbf{F}(P(x))]](Q)(j)]] = \lambda Q[\mathbf{F}(Q(j))]$

(thus this denotes the set of properties that in the future John will have)

In the end, then, this is applied to some function of type <e,t> which is salient in the discourse context; usually some property made salient by being the meaning of an overt VP.

The account generalizes to the case of ACD with very little effort. This will be just like the case of ordinary VP Ellipsis, except that here what is missing is a 2-place relation which is contextually supplied (and again, this prefers to be the meaning of overt linguistic material). Before proceeding with the actual generalization, it is worth pointing out that TVP ellipsis does indeed have the same basic distribution as VP ellipsis. It can occur across a conjunction as in (26), across sentences as in (27), and it can precede its antecedent as in (28):

(26) I know which author Mary likes, and I know which author Sue doesn't. (Evans, 1988)

(27) Bagels, I like. Donuts, I don't. (Evans, 1988)

(28) ??Although I know which book Sue will, I don't know which book Mary will read.

Since (26) - (28) are exactly the type of sentence which Sag's (1976) definition of "alphabetic variants" was designed to preclude, a brief digression is in order here. Thus Sag noted the ungrammaticality of (29):

(29) Which author does Mary like? I don't know, *which does Sue?

In order to rule this out, he gives a somewhat complex and non-standard definition of "alphabetic variance" (and a definition which cannot easily be recast in purely semantic terms; see Partee and Bach (1981) for discussion). Under Sag's treatment, both the antecedent and ellipsis site in (26) - (28) contain variables in the object position, and in both cases the variable is unbound within the lowest VP. Sag required that any unbound variables within each of the VPs (the antecedent VP and the elided VP) must ultimately be bound by the same thing. This is not a truly compositional definition, for one must look "up the tree" to see whether or not two VPs are alphabetic variants. Put differently, the usual definition of "alphabetic variants" is set up in such a way as to ensure that two formulas with different variable names in bound positions can nonetheless be alphabetic variants. This will happen just when the two formulas end up meaning the same thing. Thus the rationale for giving a definition of "alphabetic variants" is actually set up to correct a problem which is arguably an artifact of using variable names - it is a way to ensure equivalence of formulas containing different variable names but where the two formulas nonetheless have the same meaning. (A side advertisement: under the variable-free view no such confound arises.) Now take the case of two identical formulas both of which contain unbound variables. If the name of the unbound variable is the same then these too have the same meaning. Thus <u>like x</u> and <u>like x</u> necessarily have the same semantic value under a compositional treatment of the meaning of such formulas 0 even if ultimately the two occurrences of <u>x</u> are bound by different material. The usual definition thus treats these two formulas as alphabetic variants. But Sag's definition of alphabetic variance was designed to rule out alphabetic variance here, just in case the two occurrences of <u>x</u> have different binders. This is thus non-compositional as it does not respect the actual meaning of these formulas, and this non-compositional complication in turn was stipulated in order to account for cases like (29). (It was also designed to account for cases of the general form in (30) below which Sag mistakenly thought were impossible.)

Thus, under just about anyone's view of how VP ellipsis works, Sag's condition (or its analogue in other frameworks) is actually a rather complex and unfortunate one. Fortunately (for everyone), his basic observations about where VP ellipsis was possible were simply incorrect. Thus while (29) is indeed ungrammatical, Evans (1988) points out that this undoubtedly has something to do with the presence of subject-aux inversion here, since (26) is perfect. Jacobson (1992b) moreover cites cases like (30) which are fine but which run counter to Sag's basic generalization:

(30) John asked Mary to water his plants, and Bill asked Sue to.

I believe that most researchers now agree that such cases are fine provided the stress and focus considerations are set up properly. Notice that the TVP ellipsis analysis - being a completely compositional one - correctly predicts that (26) - (28) and (30) are all good - in all cases all that is supplied is a missing 2-place relation. In (30), for example, the missing meaning supplied will be λx[water'(x's plants)]; this is the type required here because *ask* undergoes **z** in such a way that its subject slot will bind the x-slot. (*Sue* will undergo the **g** rule so as to combine with a 2-place relation rather than an ordinary meaning of type <e,t>>.)

Returning to the basic distribution of TVP Ellipsis, we should ask whether we also get "deep anaphora" cases- that is cases where the antecedent is supplied contextually. These are rather difficult, but with enough context it does seem to be possible, as in (31) and (32):

(31) Context: I see you about to grab some cookies: Not those, you don't.

(32) Same context; I point to one batch and say: "These, you may". Pointing to a second batch I say "Those you can't " - at least not until they cool down.

It is unquestionably much more difficult to construct "deep anaphora" situations for TVP Ellipsis than for regular VP Ellipsis, and this ultimately needs to be explained. A tentative suggestion is that this is because 2-place relations are even more fragile and more difficult to access from context than are functions of type <e,t>. Note that in general the constraints on TVP

Ellipsis are heavier - it is harder to get this when the ellipsis site precedes the antecedent and it is somewhat harder to get it across sentences.

Now let us return to showing that the analysis generalizes to ACD with little further ado. To work this out, we need some concrete account of extraction. There are a variety of possibilities available in the CG marketplace, and so to be neutral I will use the notation A|B to mean an A with a B-type extraction gap. (Thus I mean this to be neutral between the account of extraction in, e.g., Steedman (1987) where we would have an A/B (with the normal CG "slash") and the account in which what we have is A with an extraction feature (as done in, e.g., GPSG or Jacobson, 1989) or the variant of this A↑B of Moortgat, 1988.) Let me further assume that the Geach rule is generalized so that any thing of category A/B can undergo this to become (A|C)/(B|C) and that expressions of category A^{B} can also undergo **g** to become $(A|C)^{(B|C)}$. Notice that the first mentioned generalization means that any expression which wants to take as argument something of category B can instead take as argument something which would be of category B but contains an extraction gap (of category C), and the result inherits the information that there is an extraction gap here. This is basically the convention needed to "pass up" extraction gaps - any theory will need some such conventions (this is essentially the slash-passing convention of GPSG.) The second generalization means that any expression which contains an "anaphor" of category B (which could be a pronoun, or an ellipsis site) can shift to contain an "anaphor" of category B but with an extraction gap of category C; again the entire constituent also contains the information that there is an extraction gap here. Thus a VP^{VP} can map into a $VP|NP^{VP|NP}$ - this means that a pro-VP can map into a proform over VPs with NP extraction gaps.[9]

---

[9] Szabolcsi (personal communication) raises the question of how to account for the fact that (i) has the reading in (iia) as well as the reading in (iib):

(i) Every man_i saw his_i mother before Bill did.

(ii) a. Every man_i saw his_i mother before $Bill_j$ saw his_i mother.
b. Every man_i saw his_i mother before $Bill_j$ saw his_j mother.

Indeed this appears puzzling at first glance: in order for the first *his* to be bound by the subject, one might think that *saw* must undergo **z**. But if **z**(saw'(his-mother')) is supplied as the "missing" meaning here then we would get the meaning in (iib) and not the meaning in (iia). (More needs to be said, in fact, to get the (iib) meaning - in particular *do* has to undergo a shift so that its missing argument is not just a VP but a VP^{NP}, but the interested reader can easily construct the details which will lead to the meaning in (iib). The more subtle question is how to get the meaning in (iia).)

In fact, though, the system does correctly supply this meaning as well. Note first that - in almost any theory - we must conclude that the *before* clause is a VP modifier, since in general the subject of the whole S can "bind" into this clause and hence must be a higher argument:

(iii) a. Every man_i saw his_i mother before Bill saw his_i mother.
b. Every man_i ran before Mary saw his_i mother.

In CG terms, then, *before* must have as its category $((S/_LNP)/_L(S/_LNP))/_RS$ and have a meaning of type <t,<<e,t>,<e,t>>>. The literature on temporal semantics often defines before' in terms of a relation between two propositions - but it is trivial to recast this "core" meaning into a meaning of the above type, by the usual mappings of a function f into the "Geach" of that function. (It has also been pointed out from time to time that objects can also bind into *before* clauses, as in: *John spoke to each/every man_i before Mary spoke to him_i.* The most straightforward account of this fact in the framework here is to assume that *before* is polymorphic and can combine with an S to yield a TVP modifier as well as a VP modifier, where the object wraps in. It seems to me that any theory will need to tell the analogous story here. We willnot, however, pursue this further as it is tangential to the main point.)

Once we note see the category and type of *before,* we can see that binding in (iiib) involves the application of **z** on *before.* Note that it cannot be accounted for by binding on *ran,* but there is no difficulty with allowing *before* to undergo **z** - indeed the ability of *before* to undergo **z** is an automatic consequence of the whole system. Note then that **z**(before') is of type <<e,t>,<<e,t>,<e,t>> and it will merge the highest e-argument slot (the ultimate subject argument slot) with - to put this informally - the argument slot which is marked by the pronoun within the sentential complement of *before.* (It should be noted that this involves the generalization of the **z** rule discussed in Jacobson (1999) whereby an argument position is "skipped" - such a generalization is needed independently for the case of 3-place verbs.) Thus the meaning of ***z****(before)* is: $\lambda P[\lambda Q[\lambda x[\text{before}'(P(x))(Q)(x)]]]$.

The derivation if (iiia) is a bit more complex, but only because we have two pronouns here. As discussed in Jacobson (1999), one way to have two

With this apparatus, we can spell out the full derivation of (33), a run-of-the-mill ACD case:

---

pronouns bound by the same argument slot is to have two applications of **z**. Probably the simplest way to think about this is as follows: apply **z** to *before* as done above, and combine this with the meaning of *Bill saw his mother,* which is λy[saw'(the-mother-of'(y)(b)]. We then have:

(iv) before Mary saw his mother; $(S/_L NP)/_L(S/_L NP)$;
λP[λQ[λx[before'(P(x))(Q)(x)]]](λy[saw'(the-mother-of'(y))(b)]) =
λQ[λx[before'(saw'(the-mother-of'(x))(b))(Q)(x)]]

This ensures that the subject of the entire S will "bind" the pronoun *his* within the complement of *before.* This phrase itself is now waiting to combine with a VP to give a VP, and thus is of type <<e,t>,<e,t>>. But it too can then undergo **z** such that it wants as argument a pronoun-containing VP, where the ultimate subject slot will "bind" that pronoun. In other words, it will shift by **z** into the following meaning:

(v) λR[λx[before'(saw'(the-mother-of'(x))(b))(R(x))(x)]]

It will then combine with the "matrix" VP *saw his mother* - and ultimately both occurrences of *his* will be "bound" by the subject slot. Although this might look complex, note that no new apparatus is involved here - this derivation is straightforwardly licensed by the system.

From here it should be easy to see that the reading of (i) in (iia) is also straightforward, although the technical details are space-consuming and so will be skipped here. The point, though, is simply that *do* is "missing" a 2-place relation as its argument, and so what is ultimately picked up as argument here is the meaning of the "matrix" VP *saw his mother* (without any application of **z** on *saw)*. In other words, what is picked up is the two-place relation λx[saw'(the-mother-of'(x))]. The binding of the outermost argument slot here happens in the same way as does the binding of the second *his* in (iiia).

Much of the relevant literature has also noted the fact that while (i) is ambiguous, there is no corresponding ambiguity in (vi):

(vi) Every man_i called his_i mother. Bill did too.

But this is perfectly unsurprising in the system here (and, indeed, in any reasonable system) - since (modulo the well-known case of indefinites and donkey pronouns) we don't have binding across clauses. (But, see Szabolcsi (this volume) for an extension of the basic system here to cross-sentential donkey binding.)

(33) John saw every man who Bill will.

will; VP^{VP}; $\lambda P[\lambda x[\mathbf{F}(P(x))]]$ $\text{--->}_{\mathbf{g}}$

will; $(VP \mid NP)^{(VP \mid NP)}$; $\lambda R[\lambda y[will'(R(y))]]$ =

$\lambda R[\lambda y[\lambda P[\lambda x[\mathbf{F}(P(x))]](R(y))]]$ =

$\lambda R[\lambda y[\lambda x[\mathbf{F}(R(y)(x))]]]$

Bill; (in its type lifted category and meaning) $S/_{R}VP$; $\lambda P[P(b)]$ $\text{--->}_{\mathbf{g}}$

Bill; $(S \mid NP)/_{R}(VP \mid NP)$; $\lambda R[\lambda z[Bill'(R(z))]]$ = $\lambda R[\lambda z[R(z)(b)]]$ $\text{--->}_{\mathbf{g}}$

Bill; $(S \mid NP)^{(VP \mid NP)}/_{R}(VP \mid NP)^{(VP \mid NP)}$: $\lambda W[\lambda T[\lambda R[\lambda z[R(z)(b)]](W(T))]]$ =

$\lambda W[\lambda T[\lambda z[W(T)(z)(b)]]]$

Bill will; $(S \mid NP)^{(VP \mid NP)}$; $\lambda W[\lambda T[\lambda z[W(T)(z)(b)]]](\lambda R[\lambda y[\lambda x[\mathbf{F}(R(y)(x))]]])$ =

$\lambda T[\lambda z[\lambda R[\lambda y[\lambda x[\mathbf{F}(R(y)(x))]]](T)(z)(b)]]$ = $\lambda T[\lambda z[\lambda y[\lambda x[\mathbf{F}(T(y)(x))]](z)(b)]]$ =

$\lambda T[\lambda z[\lambda x[\mathbf{F}(T(z)(x))](b)]]$ = $\lambda T[\lambda z[\mathbf{F}(T(z)(b))]]$

This is thus a meaning which is looking, first, for a TVP type of meaning as its outermost argument slot. As in the case of any other anaphoric containing expression, this open slot will keep getting passed up and - since this is the case of a free anaphor - it will eventually be supplied by the context. The expression is also missing an object argument - and this is bound by the meaning of the relative pronoun (or however we put relative clauses together in general). Thus the relative pronoun in turn also undergoes the **g** rule on 2-place relations to pass up a "missing" 2-place relation.

Now consider the crucial case (20):

(20) John will kiss every girl who thinks (that) John will.

Again we can show that this is indeed an instance of the same basic phenomenon; note for example that this can apply across sentences in a discourse as in the following case from Jacobson (1992b):

(34) John kissed every girl. But Mary is the only one who wanted him/John to.

Recall that the relevance of (20) for the punchline of this paper was as follows. The standard wisdom on (20) is that the "missing" meaning is a VP meaning - i.e., the meaning of *kiss her* which we can represent (in the standard account) as kiss x. The problem then is that there appears to be no linguistic expression with this meaning, and hence we are forced to the conclusion that the object NP is scoped out, leaving kiss x as the meaning / representation of the antecedent VP which can thus serve as antecedent to the missing complement of *will*. However, in the variable-free theory there is no problem - provided that the binding effect of a pronoun is located in the **z** rule rather than in the meaning of the pronoun. The point is that we do just pick up the meaning kiss' and the object "slot" of this is bound via the **z** rule. Put differently, we can think of this as picking up a meaning which is the same as the meaning of *kiss her* - but this is no problem because *kiss her* does in fact still just mean kiss'. Thus the semantic composition of (20) will be exactly like that of (18); the point then is that the open object slot will be bound by the application of the **z** rule on *thinks*.. We can thus spell out the full composition:

(20) John will kiss every girl who thinks John will.

(35) will; VP^{VP}; $\lambda P[\lambda x[\mathbf{F}(P(x)]]$ $\text{--->}_{\mathbf{g}}$ $(VP^{NP})^{(VP^{NP})}$; $\lambda R[\lambda y[\lambda P[\lambda x[\mathbf{F}(P(x))]]\,(R(y))]]$

$$= \lambda R[\lambda y[\lambda x[\mathbf{F}(R(y)(x))]]]$$

John; S/VP $\text{==>}_{\mathbf{g}}$ S^{NP}/VP^{NP}; $\lambda R[\lambda x[\lambda P[P(j)](R(x)]] = \lambda R[\lambda x[R(x)(j)]]$

$\text{==>}_{\mathbf{g}}$ $(S^{NP})^{(VP^{NP})}/(VP^{NP})^{(VP^{NP})}$;

$$\lambda W[\lambda T[\lambda R[\lambda x[R(x)(j)]](W(T))]] = \lambda W[\lambda T[\lambda x[W(T)(x)(j)]]]$$

where W is a variable over functions of type <<e,<e,t>>,<e,<e,t>>>

and T is a variable over functions of type <e,<e,t>>

John will; $(S^{NP})^{(VP^{NP})}$;

$$\lambda W[\lambda T[\lambda x[W(T)(x)(j)]]](\lambda R[\lambda y[\lambda z[\mathbf{F}(R(y)(z))]]]) =$$

$$\lambda T[\lambda x[\lambda R[\lambda y[\lambda[\mathbf{F}(R(y)(z))]]](T)(x)(j)]] = \lambda T[\lambda x[\lambda y[\lambda z[\mathbf{F}(T(y)(z))]](x)(j)]] =$$

$$\lambda T[\lambda x[\lambda z[\mathbf{F}(T(x)(z))](j)]] = \lambda T[\lambda x[\mathbf{F}(T(x)(j))]]$$

thinks: VP/S; thinks' $\text{--->}\mathbf{z}$: VP/S^{NP}; $\lambda P[\lambda y[think'(P(y))(y)]]$ $\text{--->}_{\mathbf{g}}$

$VP^{(VP^{NP})}/(S^{NP})^{(VP^{NP})}$; $\lambda W[\lambda R[\lambda P\lambda y[think'(P(y))(y)]]\,(W(R))]] =$

$$\lambda W[\lambda R[\lambda y[think'(W(R)(y))(y)]]]$$

thinks John will; $VP^{(VP^{NP})}$; $\lambda W[\lambda R[\lambda y[think'(W(R)(y))(y)]]]\ (\lambda T[\lambda x[\mathbf{F}(T(x)(j))]]\ )$ =

$$\lambda R[\lambda y[think'(\lambda T[\lambda x[\mathbf{F}(T(x)(j))]](R)(y))(y)]] =$$

$$\lambda R[\lambda y[think'(\lambda x[\mathbf{F}(R(x)(j))](y))(y)]] =$$

$$\lambda R[\lambda y[think'(\mathbf{F}(R(y)(j)))(y)]]$$

Hence the subject slot of *thinks* (the y-slot) binds the object slot of the missing 2-place relation. This material will ultimately combine with the relative pronoun *who* in such a way that the "missing" 2-place relation continues to be held off (as is always the case when a proform is passed up) and indeed this passing up of the open 2-place relation will happen up to the top. At the end, then, the meaning of the final S will be:

(36) $\lambda R[\quad x[girl'(x) \text{ and } think'(x)(\mathbf{F}[R(x)(j)]) \dashrightarrow \mathbf{F}[kiss'(x)(j)]]]$

and this will be applied to the contextually salient <u>kiss'</u> relation. (The interested reader can verify that this will come out as expected).

We can now make good on an earlier promise: answering the question as to why the full paraphrase of something like (20) contains a transitive verb with an overt pronominal object and not just a simple intransitive verb (the paraphrase is in (18)). The answer is actually analogous to the one which would be given in a classical account: if there is a missing object or extraction gap in the syntax after *kiss* then ultimately there must be something higher up which expects to co-occur with material containing an extraction gap. But there is no such expression. (We do find the pronoun *who* which is the subject of *thinks,* but this of course has its desires filled by combining with the VP headed by *thinks.)* To detail this a bit more: note that *will* comes in two forms. It is a VP/VP and a VP^{VP}. The former takes a full VP complement; the latter occurs in normal "VP Ellipsis" cases. The former can also map into something taking a VP-like complement but one which contains a pronoun - i.e., it can map into VP^{NP}/VP^{NP}. This is the item which occurs in the full paraphrase in (19). The question at hand here is whether there is any way that *will* can occur with bare *kiss* in this kind of construction? Consider the possible categories for *kiss*. It can be the ordinary transitive verb

VP / NP or it can be the item occurring with an extraction gap: VP | NP. We have defined no conventions which allow *will* to combine with a VP / NP. What about trying to combine *will* with a VP | NP? Indeed this is possible, but in this case the result will also be of category VP | NP (this will involve *will* of category (VP | NP) / (VP | NP). In fact the composition can continue - but the information that there is an extraction gap will keep being passed up. Ultimately then we need to find some item (such as a relative pronoun) which subcategorizes for S | NP, but here there is none. [10]

4. Apparent Exceptions to the i-within-i effect

A second case which supports the view that binding is effected by **z** rather than being built into the meaning of the pronoun centers on the so-called i-within-i condition, and in particular on some interesting exceptions to this condition. Sec. 4.1 will first elucidate the phenomenon and the account of it which was developed in Jacobson (1993, 2000b) and Sec. 4.2 will show how some apparent exceptions to this effect provide the key evidence here.

4.1. The i-within-i effect

By the i-within-i effect, I mean essentially the generalization spelled out in (37). A typical example is given in (38); much more detailed data can be found in Jacobson (2000b):

(37) The complement of a relational noun cannot be "bound" by the head

(38) *The / Every wife$_i$ of her$_i$ childhood sweetheart came to the party.

---

[10] There is admittedly more that needs to be said here. Ultimately we need to explain just what kinds of expressions can stand alone - i.e., what categories are root categories. Thus we can end up with a sentence-like expression which contains a free pronoun, but its category is not S but rather S^{NP}. However, we generally do not find stand-alone expressions of the form S | NP - that is, expressions which contain extraction gaps. The semantics is the same, so the answer will have to come from the syntax. While I have no answer to this here, I believe that similar questions arise in any theory. Certain kinds of expressions are quite natural as standalone expressions (not just Ss but, for example, NPs) while other kinds (e.g., transitive verbs) are not.

The mystery surrounding this is that there is no obvious semantic and / or pragmatic problem here. Notice that wife' is a 2-place relation and thus of type <e,<e,t>>, and so there is no obvious reason why a phrase such as *wife of her childhood sweetheart* cannot mean:

(39) λx[wife-of'(the-childhood-sweetheart-of'(x))(x)]

In fact, rough paraphrases of (38) are just fine. Thus a similar NP containing a relative clause is good, and even a reduced relative and other kinds of participles are good here:

(40) The / Every woman$_i$ who$_i$ is married to her$_i$ childhood sweetheart came to the party.

(41) The / Every woman$_i$ married to her$_i$ childhood sweetheart came to the party.

This also argues against any kind of pragmatic account of these (for additional discussion, see Jacobson, 2000b).[11]

If we consider just the semantics of the system developed here, then indeed there is no explanation for the ungrammaticality of (38). Note that the meaning of *her childhood sweetheart* is of type <e,e> and is λx[the-childhood-sweetheart-of'(x)] - or, more simply the-childhood-sweetheart-of'. (Again, I am ignoring details of the semantics of the genitive; for further discussion see Jacobson (2000b).) wife' wants an argument of type e, but should it undergo **z** it will want an <e,e> argument, as shown in (42). Application of this to the-childhood-sweetheart-of' will yield the offending meaning given above in (39):

(42) wife' <e,<e,t>> ==> **z**(wife') = λf[λx[wife-of'(f(x))(x)]]

But the key to the i-within-i effect resides in the tight coupling of the syntax and the semantics. As we will see momentarily, common nouns have a syntactic peculiarity - a peculiarity which can be demonstrated on grounds quite separate from the domain here. This - combined with the program which couples the syntax and the semantics in the way that CG does - will automatically account for the i-within-i violations.

---

[11] For example, these have sometimes gone under the rubric of involving "circular reference". But such rubric begs the question of why cases like (40) and (41) are good while (38) is not; note moreover that there is nothing truly circular about the meaning in (38).

To elaborate, note first that the formulation of **z** in (5) applies not only to expressions which are semantically of type <B',<e,A'>> but which also have (at least) two syntactic argument slots: the input expression must have a category of the form (A/NP)/B. The restriction to items of this syntactic category follows from the general program in which the semantic and syntactic rules will be matched. There is, moreover, a technical reason for providing the **z** rule with a syntactic as well as a semantic side - the input expression needs to shift its syntactic category so that it now wants complements of category B^{NP} rather than just of category B.

But now consider the syntax of ordinary common nouns such as *table*, etc. Although these presumably have meanings of type <e,t> - just as do adjectives and intransitive verbs - they appear not to syntactically be two-place. In other words, N seems to be a primitive category in the syntax - not a shorthand for some category of the general form (X/NP) (contra Montague, 1974). What motivates this claim is the fact that common nouns never actually occurs with overt subjects - even in "small-clause" environments. This contrasts sharply with the case of adjectives, participles, etc. - the latter do directly take subjects in certain environments, as shown in (43) and (44):

(43) a. With Zelda chairing the committee, they will never authorize another theoretical linguistics position.

b. With Mitka so disobedient, I don't trust him off leash.

c. With Tom married to Sally, he'll get to do a lot of traveling.

(44) a. What I would prefer is a true liberal running for Senate.

b. What I would prefer is Mitka more obedient.

c. What I would prefer is Tom married to Sally.

Thus note that participles (such as *married (to)*) certainly contain a syntactic subject slot - these are presumably of some category such as S/NP (with some kind of feature on the result S category), or, perhaps SC/NP (for SC = "small clause"). It is thus not surprising that *married to* in (41) can undergo **z** in such a way that its subject slot "binds" the pronoun slot in its object argument. Similarly in (40) we have a full relative clause with an ordinary VP, and so **z** applies here too.[12]

Ordinary common nouns, on the other hand, never combine with subjects - even in these small clause environments:[13]

(45) *With that piece of wood table, we'll have a place to eat.

(46) *I would prefer that piece of wood table.

This fact motivates the claim that N is a primitive category, not an abbreviation for something of the form X/NP (for some category X). And this in turn means that relational nouns are of category N/PP[OF} - they are thus syntactically only one place. And from this it follows that they do not meet the structural description to undergo **z** . Hence even though the i-within-i effect is mysterious on purely semantic grounds, it becomes completely unsurprising under this view of binding given the peculiar syntax of common nouns.

4.2. <u>The exceptions to i-within-i</u>

So far, the same basic observations carry over directly to the binding-as-pronoun-meaning account. As discussed in Sec. 2.2, there is more than one such account in the literature and others are probably possible. But they all work in such a way that the pronoun eventually

---

[12] Note that in the account here we do not need any actual subject to bind the pronoun - only a subject slot. Thus no silent subject needs to be provided. Cases like (41) are problematic for the suggestion in fn. 1 where binding is built into a type-shift rule on the binder, since there is no actual material here to be the binder.

[13] There is a well-known class of exceptions to this; nouns like *president, Chair,* etc. can occur predicatively (basically they are nouns which generally denote singleton sets):

(i) With Andy Chair again, you can be sure that discussion of that matter will be suppressed.

However, I assume (with Partee, 1985) that these nouns actually shift into NPs (a not surprisingly shift from <e,t> to e since as nouns they denote singleton sets), and it is well-known that NPs can then shift into predicative expressions.

takes as argument something of type <e,X> (where the e-slot corresponds to the "binder" slot) - and in a CG framework we would thus expect the syntax to be parallel. That is, we would expect the relevant argument to be syntactically of category X/NP. And the fact that relational nouns do not have the relevant NP argument slot in the syntax will thus ultimately account for the i-within-i effect in a way parallel to the story here. Consider for example Szabolcsi's (1992) account. Recall that we revised her pronoun category to: (VP/(VP/Y))/(Y/NP)); this was necessary to get things like *Every woman* $_i$ *loves her* $_i$ *mother*. Moreover, take "VP" here to be an abbreviation for anything which has a subject slot - it thus abbreviates S/NP, SC/NP, etc. (This would be needed to allow for binding by the subject slots in participles and adjective phrases, for example.) To simplify the exposition, I also need to ignore the difference between NPs - the most convenient way to do this will be to take *wife of* to be a single lexical item. Since in reality N is a primitive category this would be of category N/NP, but in order to trace through the point we will for the moment assume - counterfactually - that *wife of* has a subject slot. This means it is of category (SC/NP)/NP. Under this counterfactual scenario, we would get i-within-i violations. One instantiation of the variables in the category for *her* is ((SC/NP)/((SC/NP)/NP))/(NP/NP[GEN]). Thus *her* first combines with *best friend* (of category NP/$_L$ NP[GEN] to give as result (SC/NP)/((SC/NP)/NP)). Note that *wife of* is (under this mistaken category assignment) (SC/NP)/NP - it has the same basic category as an ordinary transitive verb, with the only difference being that once a subject is supplied the result is a SC (small clause) rather than an ordinary subject. Then *wife of her best friend* will be an ordinary common noun - an SC/NP - and the offending meaning will result; I leave it to the interested reader to trace through the semantic details.

But of course this account can explain the i-within-i effect in exactly the way we have done in the type-shift approach. The mistake in the above paragraph was to assign ordinary common nouns the category SC/NP and hence relational nouns the category (SC/NP)/NP. In reality, an ordinary noun is N, and a relational noun is N/NP. This in turn means that pronouns which want to be bound cannot combine in the way shown above, and i-within-i effects are thus expected.

But there is more to the story. The crucial evidence for the type-shift approach centers on a class of systematic exceptions to the i-within-i effect. Thus Lauri Karttunen and Geoffrey Nunberg both pointed out to me that the i-within-i effect disappears (or is at least greatly ameliorated) with nouns which are transparent agentive nominalizations. Thus, for example, compare (47) to (48):

(47) a. ?*Every / The $builder_i$ of her_i house left.

b. ??Every / The $builder_i$ of her_i father's house left.

c. ?Every / The $builder_i$ of her_i own house left.

(48) a. *Every / The $wife_i$ of her_i / her_i own childhood sweetheart left.

b. *Every / The $wife_i$ of her_i sister's childhood sweetheart left.

c. *Every / The $wife_i$ of the author of her_i biography left.

Although (47a) remains somewhat bad (for reasons which I do not understand but which are probably orthogonal to the points here, (47b) improves dramatically and (47c) is just about impeccable. (Thus compare (47c) to (48a) where the addition of *own* does not help at all.) There is incidentally a bit of speaker and lexical variation on this (a point to which I will return momentarily), but all speakers with whom I have consulted do find the improvement effect. The effect can also be documented by contrasts like (49) and (50):

(49) a. *Every / The $author_i$ of her_i own / her_i mother's biography left.

b. ??Every / The $writer_i$ of her_i own / her_i mother's biography left.

(50) a. *Every / The $cousin_i$ / $lover_i$ / $partner_i$ of her_i (own) childhood sweetheart left the party.

b. ??Every / The future $owner_i$ of her_i father's yacht came to the party.

And an especially striking pair supporting the claim that transparent agentive nominals show a marked improvement is provided by the contrast in (51). In (51a) *lover* is used in its transparent, compositional sense, while in (51b) it is used in its fully lexicalized sense:

(51) a. ?Every / The lover$_i$ of his$_i$ mother's art collection will get to inherit it.

b. *Every / The lover$_i$ of his$_i$ mother's hairdresser will get many wigs.

Under the type-shift approach to binding, these exceptions are unsurprising. We can account for them by assuming that what is nominalized here is **z**(*build)*, **z**(*own)*, **z***(write)* etc. Since these verbs are listed as 2-place in the lexicon, **z** can first apply, followed by the nominalization process. But relational nouns like *wife* , *author, partner,* and *lover* (in its idiomatic sense) are of course not derived from verbs and so are never 2-place. Moreover, it is not entirely surprising to find speaker variability on these. Morphological processes like these are well-known to be slippery and variable in terms of how productive they are and to vary across speakers and across lexical items. Since agentive nominalization is not a completely productive process, those speakers who don't allow the above may allow nominalization only of the basic and not the shifted form.

It might at first glance appear that the same story carries over to the binding-as-pronoun-meaning view, but in fact it does not. The trouble is that the pronoun combines with the already nominalized form - and so it should not have any way of "seeing" that the nominalized form is 2-place in the lexicon. Notice that once these 2-place verbs nominalize they do in fact lose their subject argument slot and behave in the syntax just like any other relational noun:

(52) a. *With Bill lover of fine art, he'll spend a lot of money collecting it.

b. *I would prefer Sally builder of my next house.

This means that the agentive nominalizations have lost their subject slot by the time they get to combine with the pronoun which is within their complement in (47), (49b), (50b), and (51a). Thus the pronoun cannot be sensitive to the fact that in these cases there once was a subject slot while in the relational nouns in (48), (49a), (50a), and (51b) there was not. The only way that I can to allow the binding-as-pronoun-meaning analysis to cash in on the fact that the relevant verbs are 2-place in the lexicon is to have them first combine with the verbs and then to have nominalization apply to the entire phrase (where the verb already has its object). This of course is reminiscent of early transformational treatments such as that of Lees (1960) and may be possible, but it certainly requires some rather non-standard views of the interaction of morphology and syntax.

5. Additional Cases

The material dealt with in my previous work - especially in Jacobson (1999, 2000b) contains a number of additional cases which show binding effects but where there is no overt pronoun. In fact, this point is discussed in some detail in Jacobson (2000b, Sec. 5) - where I use this fact to argue against the standard view of binding in which it is a relationship between a binder and an overt pronoun / variable. But the point carries over directly here: such cases also indicate that binding is not a fact about the lexical meaning of a pronoun. And - at least in many of these cases - there also is no other overt expression whose lexical meaning can give rise to the binding effect.[14]

---

[14] Certain cases of binding without overt pronouns could be handled by lexical meaning. This is true for the case of what I refer to as "Mitchell / Partee expressions" (Mitchell, 1986; Partee, 1991) which includes things like *local,* as in:

(i) Every Red Sox fan $_i$ watched the game at a local$_i$ bar.

Here the binding-as-lexical-meaning view has no problem; binding can be built into the meaning of *local.* In the cases discussed in the text, however, there is no obvious lexical item whose basic lexical meaning can give the binding effect.

The existence of expressions like *local* does, though, have an interesting consequence for the treatment of free pronouns. In Sec. 2.1 I argued against the view that free pronouns are just accidentally homophonous to bound pronouns, and the Mitchell / Partee expressions provide further evidence of this. This is because items like *local* can be free as well as being bound (this fact is noted in Mitchell, 1986 and Partee, 1991):

(ii) Every Red Sox fan$_i$ visited a local$_j$ bar

where the value of the *local* function is provided contextually (in this case, for example, it can be interpreted as local to speaker). As far as I know, this is true for every single expression in this class - and so we certainly would not want to posit accidental homophony.

Obviously I do not have space to detail these cases here, but it is worth a quick mention of some of them. Perhaps the most dramatic concerns the case of NPs with functional readings but where the argument slot of the function is "bound" in the way that binding normally takes place. This type of phenomenon is found, for example, in the case of the inferences shown in (52) (Reinhart, 1990; Jacobson, 1992b, 1994, 1999); these are unexpected under standard views of binding but quite unsurprising here:

(52) a. John (always) buys the (very) thing that Bill always buys.

b. $Bill_i$ always buys his_i favorite car.

c. Therefore, $John_j$ always buys his_j favorite car.

In Jacobson (1992b, 1999) I documented that the inference in (c) comes almost for free here - we need only one additional step. This is a step allowing the head noun *thing* to shift into a set of functions whose range is of type <e,t> (and as far as I know, any account of these inferences within any other view of binding will need this piece, too, so there is no extra apparatus here). Informally, the semantic composition of (a) and (b) is as follows:

(53) a. John **z**- buys the function f with range thing' such that Bill **z**-buys f.

b. Bill **z**-buys the-favorite-car-of function (i.e., the function <u>λx[the-favorite-car-of'(x)]</u>

And so from (a) and (b) it follows that John also stands in the **z**-buy relation to <u>the-favorite-car-of</u> function.

This works in virtue of two key facts. The first is that *the thing that Bill buys* is allowed to shift into a function f with range <u>thing'</u>. (For full details of the internal composition along with details of the interpretation of *the* here, see Jacobson (1994; 1999); see also Sharvit (1997) for related discussion.) The second point is that the <u>argument position</u> of that function is bound by the subject of *buy* - even though there is no overt pronoun which is attached to the function. This is because that binding happens in virtue of the application of **z** on <u>buy'</u>.

What would one need to say under a view in which binding normally is accomplished via the lexical meaning of a pronoun? It is unclear to me - but the best I can think of is to allow a type-shift rule or series of such rules which somehow shifts the meaning of the object NP *the thing that Bill buys* into a complex meaning which does the binding work. I leave it to others to work out the relevant details, but I can see little advantage to pursuing this course. It too, as far as I can see, will have to have one or more type-shift rules applying here, and so there is no obvious way to build everything into lexical meaning.

A second a related case concerns my analysis of functional questions (Jacobson, 1992b, 1999, etc.). Thus I have argued that under the **z** approach to binding the functional reading of a question like (54) also comes (almost) for free:

(54) Who does every Englishman love? (His mother.)

The "almost" caveat here is necessitated by the fact that *who* will have to be polymorphic under this approach, but again I believe that this is required under any other approach to functional questions, too and so does not represent any real complication here. But the rest of the composition of (54) automatically gives the functional reading. The key concerns the composition of *every Englishman love*. Here love' type-shifts by **z** so as to expect an object of type <e,e> rather than an ordinary individual object. It then combines with every Englishman' in the way that verbs with extraction gaps normally do. (Either the two directly function compose, or every Englishman' first undergoes **g** and then takes the verb as argument.) To illustrate in the simplest way, assume that every-Englishman' directly function composes with the verb; we then get the functional reading as follows:

(55) every-Englishman' o **z**(love') =

$\lambda f[\text{every-Englishman}'(\lambda x[\text{love}'(f(x))(x)])$

The functional "gap" is a missing argument like any other - love' expects a functional object in virtue of having undergone **z**. Likewise the argument of this function is bound by the subject slot of love' in the way that binding takes place in general.

But here too there is no overt pronoun on which to hinge the responsibility of binding the argument position of the missing functional object. Under the binding-as-pronoun-meaning approach, the most obvious solution would be to posit a silent pronoun in the gap position which is doing the binding. This is possible - but a theory which allows silent pronouns like this should be equally happy with the **z** type-shift approach. After all, one can always recast **z** as some silent prefix or suffix whose lexical meaning is **z** and which operates on verbs in the way that the type-shift does here.

There are, then, several cases of "binding without pronouns". Were the type-shift approach to require a series of complex and/or baroque rules we might reject it nonetheless - but the requisite type-shift rule (**z**) seems to be quite simple and natural and this - combined with **g** and **l** - allows for a simple analysis of numerous cases of binding-like phenomena. Under the "radical lexical" approach, which insists on building the binding effect into lexical meaning, it is not clear how to deal with cases of binding without any overt pronoun (or similar lexical item). Nor is it obvious how to deal straightforwardly with the case of free pronouns. The conclusion seems to be that while we usually think of pronouns when we think of binding, this is only one case of a much more general phenomena.

References

Bach, E. and R. Cooper (1978). "The NP-S analysis of relative clauses and compsitional semantics". *Linguistics and Philosophy* ***2***, 145-150.

Bouton, L, (1970). "Antecedent contained pro-forms", in M. Campbell et al., (eds.), *Proceedings of the 6th Regional Meeting of the Chicago Linguistic Society*. Chicago: University of Chicago.

Chierchia, G. (1984). *Topics in the Syntax and Semantics of Infinitives and Gerunds*. Ph.D. Dissertation, University of Massachussetts.

Chomsky, N. (1993). "A minimalist program for linguistic theory", in K. Hale and S.J. Keyser (eds.), *The view from Building 20: Essays on linguistics in honor of Sylvain Bromberger*, MIT Press.

Cormack, A., (1985). "VP Anaphora: Variables and Scope", in F. Landman and F. Veltman (eds.), *Varieties of Formal Semantics*. Dordrecht: Foris.

Curry, H. and R. Feys (1958). *Combinatory Logic*. Amsterdam: North Holland.

Dalrymple, M., S. Shieber and F. Pereira (1991). "Ellipsis and Higher Order Unification", *Linguistics and Philosophy* ***14***.

Dowty, D. (1999). "Natural Language Anaphora and Type Logical Syntax", presented at the *Formal Grammar 1999* Conference, Utrecht.

Evans, F. (1988). "Binding into Anaphoric Verb Phrases", in J. Powers and K. de Jong (eds.), *Proceedings of the Fifth Eastern States Conference on Linguistics*, Ohio State University Working Papers.

Fox, D. (1995). "Economy and Scope". *Natural Language Semantics* ***3***, 283-341.

Geach, P. (1972). "A Program for Syntax", in D. Davidson and G. Harman (eds.), *Semantics of Natural Language*. Dordrecht: Reidel.

Grodzinsky, Y. and T. Reinhart (1993). "The innateness of binding and coreference", *Linguistic Inquiry* ***24***, 69-102.

Haik, I. (1987). "Bound VP's that Need to Be", *Linguistics and Philosophy* ***10***, 503-10.

Hankamer, J, and I. Sag (1976). "Deep and Surface Anaphora", *Linguistic Inquiry* ***7***, 391-428.

Hardt, D. (1992).. "VP Ellipsis and Semantic Identity", in C. Barker and D. Dowty (eds.), *Proceedings of the Second Conference on Semantics and Linguistic Theory*. Ohio State Working Papers.

Heim, I. (1997). "Semantic Types for Syntactic Categories: Evidence from Ellipsis", in *Proceedings from the Seventh Conference on Semantics and Linguitic Theory*. Cornell: Cornell Working Papers in Linguistics.

Jacobson, P. (1982). *On the Syntax and Semantics of Multiple Relatives in English*. Bloomington: Indiana University Linguistics Club.

Jacobson, P. (1989). "A(nother) Categorial Grammar Account of Extraction". Paper presented at the Conference on Categorial Grammar, LSA Summer Institute, Tucson, Arizona.

Jacobson, P. (1991). "Bach-Peters Sentences in a Variable-Free Semantics", in P. Dekker and M. Stokhof (eds.), *Proceedings of the Eighth Amsterdam Colloquium*. Amsterdam: University of Amsterdam ILLC.

Jacobson, P. (1992a). "Flexible Categorial Grammars: Questions and Prospects" in R. Levine (ed.), *Formal Grammar: Theory and Implementation*. Oxford: Oxford University Press (Vancouver Studies in Cognitive Science).

Jacobson, P. (1992b). "Antecedent Contained Deletion in a Variable-Free Semantics", in C. Barker and D. Dowty (eds.), *Proceedings of the Second Conference on Semantics and Linguistic Theory*. Columbus: Ohio State University.

Jacobson, P. (1993). "i-within-i Effects in a Variable-Free Semantics and a Categorial Syntax", in P. Dekker et al. (eds.), *Proceedings of the 9th Amsterdam Colloquium*. Amsterdam: University of Amsterdam ILLC.

Jacobson, P. (1994). "On Copular Connectivity", in *Proceedings of the Fourth Conference on Semantics and Linguistic Theory*. Cornell: Cornell Working Papers.

Jacobson, P, (1998). "Antecedent Contained Deletion and Pied Piping: Evidence for a Variable-Free Semantics", in D. Strolovitch and A. Lawson (eds.), *Proceedings of the Eight Conference on Semantics and Linguistic Theory*. Cornell: Cornell Working Papers.

Jacobson, P. (1999). "Towards a Variable-Free Semantics" *Linguistics and Philosophy* ***22***, 117-184.

Jacobson, P. (2000a). "Paychecks and Stress", in *Proceedings of the Tenth Conference on Semantics and Linguistic Theory*. Cornell: Cornell Working Papers.

Jacobson, P. (2000b). "Paycheck Pronouns, Bach-Peters Sentences, and Variable-Free Semantics", *Natural Language Semantics* ***8***, 77-155.

Jaeger, G. (1997). "Anaphora and Ellipsis in Type-Logical Grammar", in P. Dekker, M. Stokhof, and Y. Venema (eds.), *Proceedings of the 11th Amsterdam Colloquium*. Amsterdam: University of Amsterdam ILLC.

Jaeger, G. (this volume). "Resource sharing in type logical grammar".

Lappin, S. (1996). "The Interpretation of Ellipsis", in S. Lappin (ed.), *The Handbook of Contemporary Semantic Theory*. Oxford: Blackwell, Inc.

Larson, R. and R. May (1990). "Antecedent containment or vacuous movement: Reply to Baltin", *Linguistic Inquiry* ***21***.

Less, R. (1960). *The Grammar of English Nominalizations*. The Hague: Mouton.

Mitchell, J. (1986). *The Formal Semantics of Point of View*. Ph.D. Dissertation, University of Massachussetts. Amherst, MA: UMass GLSA Publication.

Montague, R. (1974). "The Proper Treatment of Quantification in Ordinary English", in R. Thomason (ed.), *Formal Philosophy: Selected Papers of Richard Montague,* New Haven: Yale University Press.

Moortgat, M. (1988). *Categorial Investigations*. Mouton de Gruyter.

Morrill, G. (2000). *Type-Logical Anaphor*. Report de Recerca LSI-00-77-R, Departament de Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya.

Partee, B. (1976). "Some Transformational Extensions of Montague Grammar", in B. Partee (ed.), *Montague Grammar*. New York: Academic Press.

Partee, B. (1985). "Ambiguous Pseudoclefts with Unambiguous Be", in S. Berman, J.W. Choe, and J. McDonough (eds.), *Proceedings of NELS* ***16***, GLSA Publication, University of Massachusetts, Amherst.

Partee, B. (1991). "Anaphoric and Deictic Pieces of Meaning", paper presented at the 8th Amsterdam Colloquium. Amsterdam: ITLI.

Partee, B. and E. Bach (1981). "Quantification, pronouns, and VP anaphora", in J. Groenendijk, T. Jannsen, and M. Stokhof (eds.), *Formal methods in the study of language: Proceedings of the Third Amsterdam Colloquium*. Amsterdam: Mathematisch Centrum.

Quine, W.V. (1966). "Variables Explained Away", in W.V. Quine, *Selected Logic Papers*. New York: Random House.

Reinhart, T. (1990). "Self Representation", paper presented at the Princeton Conference on Anaphora, Princeton.

Reinhart, T. (1999). "Strategies of Anaphora Resolution", in H. Bennis, M. Everaert and E. Reuland (eds.), *Interface Strategies*. Amsterdam: North Holland.

Rooth, M. (1981). "A Comparison of Three Theories of Verb Phrase Ellipsis", in W. Chao and D. Wheeler (eds.), *University of Massachusetts Occasional Papers in Linguistics,* GLSA, University of Massachussetts, Amherst.

Rooth, M., (1992). "Ellipsis Redundancy and Reduction Redundancy", in S. Berman and A. Hestvik (eds.), *Proceedings of the Stuttgart Ellipsis Workshop*. Heidelberg: Arbeitspapiere des Sonderforschungsbereichs 340.

Ross, J.R. (1967). *Constraints on Variables in Syntax*. Ph.D. Dissertation, MIT.

Sag, I. (1976). *Deletion and Logical Form*. Ph.D. Dissertation, MIT. (Distributed by Indiana University Linguistics Club)

Schwarz, B. (2000). *Topics in Ellipsis*. Ph.D. Dissertation, University of Massachussetts.

Steedman, M. (1987). "Combinatory Grammars and Parasitic Gaps", *Natural Language and Linguistic Theory* ***5***, 403-440.

Schachter, P. (1977). "Does She or Doesn't She?", *Linguistic Inquiry* ***8,*** 763=767.

Stockwell, R., P. Schachter, and B. Partee, 1968. *Integration of Transformational Theories of English Syntax*. UCLA.

Szabolcsi, A. (1987). "Bound Variables in Syntax: Are There Any?", in J. Groenendijk et al. (eds.), *Proceedings of the 6th Amsterdam Colloquium*. Amsterdam: University of Amsterdam ITLI.

Szabolcsi, A. (1992). "Combinatory Categorial Grammar and Projection from the Lexicon", in I. Sag and A. Szabolcsi (eds.), *Lexical Matters*. Stanford: CSLI Publications.

Szabolcsi, A. (this volume). "Binding on the Fly".

Tanenhaus, M., G. Calrson and M. Seidenberg (1985). "Do listeners compute linguistic representations?" in D. Dowty, L. Karttunen and A. Zwicky (eds.), *Natural Language Parsing*, Cambridge: Cambridge University Press.

Webber, B. (1978). *A Formal Approach to Discourse Anaphora*. Ph.D. Dissertation, Harvard University. (Published by Garland Press, Outstanding Dissertations in Linguistics Series.)

Wexler, K. and Y.C. Chien (1991). "Children's knowledge of locality conditions in binding as evidence for the modularity of syntax and pragmatics", *Language Acquisition* ***1***, 225-295.

Williams, E., 1977. "Discourse and Logical Form", *Linguistic Inquiry* ***8***.