Compositionality and Derivation

Edwin Williams

Nov. 2013; revised Sept. 2014

Abstract: Two case studies are presented to argue for a model of grammar in which semantics and syntax are both generative, but tied together by this principle: syntactic functions are paired with semantic functions, and a syntactic function can operate only on the arguments of the semantic function it is paired with. Both case studies address the problem of squaring the binarity of syntactic structures with the polyadicity of semantic functions. The usual solution—curried semantic functions—is shown to be unnecessary and awkward. The first case is the grammar of A-quantifiers vs. D-quantifiers, which are argued to differ only syntactically. The second case is the grammar of the complement clauses of such degree operators as *more, too, so,* etc., and relative clauses.

Keywords: Syntax-Semantics interface, compositionality, A-quantifiers, degree operators, unselective binding, currying, extraposition, conservativity

1. Two kinds of Compositionality

There are two ways to understand compositionality. A common formulation--"the meaning of a whole is a function of the meaning of its (immediate) parts"-- does not decide between them. In the tradition of logic, computer programming languages, and the some of the linguistic semantics that descends from the work of Richard Montague, compositionality arises from a pairing of syntactic rules with semantic rules in the grammar. The following is a rule of grammar in the format required for YACC, a computer language compiler generator:

(1) exp: exp exp '+' {$$ = $1 + $2}

The left side asserts that a well-formed expression of the language consists of two expressions followed by the plus sign. The right side asserts that the meaning of that expression ($$) is gotten by adding the meaning of the first exp ($1) to the meaning of the second exp ($2).

In linguistics, we might write (1) as the following:

(2) a. exp -> exp exp + (syntactic rule)
b. |[[exp$_1$ exp$_2$ +]$_{exp}$]| = |[exp$_1$]| + |[exp$_2$]| (rule to interpret structures)

(1) and (2) look different, but are they really? If we are talking about the linguistics of natural languages, there could be a question of which is right. In the theory in which (1) has a place, there is an explicit pairing of syntactic rule and semantic rule. In (2) there is no such pairing; rather, there are rules that interpret the structures that are built by syntax, and there might be, for example, diverse sources of input to a single interpretive rule. We might call (1) derivational compositionality, and (2) structural compositionality.

Are both always viable? Maybe not; it depends on the how syntax works. It is conceivable, for example, that syntax creates structures that cannot be easily interpreted. In Williams (2008), (2010), I have proposed that “affix hopping” of Tense onto the verb is built into, and occurs simultaneously with, the merge of Tense with its clause; so the derivation goes in a single step like this:

(3) { past, [Bill like Mary] }$_{\text{Workspace}}$ →
{ [Bill like+past Mary] }$_{\text{Workspace}}$

The more usual derivation has two steps:

(4) { past, [Bill like Mary] }$_{\text{Workspace}}$ →
{ [past [Bill like Mary]] }$_{\text{Workspace}}$ →
{ [Bill like+past Mary] }$_{\text{Workspace}}$[1]

In (4) *past* is first merged with the clause, and then in a subsequent step, attached to the verb of the clause. There are advantages to each. In (4), there is an appropriate structure for interpretation, namely the second line, which gets the scope of *past* right; in (3) the final structure is not appropriate for interpreting *past*, whose scope is the whole clause. On the other hand, (3) evades a problem that has been noted for (4); (4) does not comply with the “extension” condition, which requires that every derivational step “expand” the structure.

If the only compositionality is structural, then (3) is not viable, as it provides no structure appropriate for the interpretation of *past*. However, if compositionality is derivational, then we may entertain (3) after all. We need a syntactic rule paired with a semantic rule. The syntactic rule will do what (3) shows: it takes a Tense element and a clause, and attaches the Tense element to the head of the clause (a more careful formulation will follow). That rule will be paired with a semantic function which takes two arguments, the meaning of Tense and the meaning of the clause, and combines them. It will be useful to think of the items in the workspace as having two parts, a syntactic form and its meaning. Then the derivation in 3 would be:

(5) { (past,|[past]|) , ([Bill like Mary], |[Bill like Mary]|) } →
{ ([Bill like+past Mary], |[Bill like+past Mary]|) }

The derivational step is constituted of the simultaneous application of a syntactic function and a semantic function:

(6) f_{syn} (past, [Bill like Mary]) = Bill like+past Mary
f_{sem} (|[past]|, |[Bill like Mary]|) = |[Bill like+past Mary]|[2]

In what follows, I will explore the possibilities that arise when semantics is

[1] From now on, braces will enclose workspaces, and I will drop the label.

[2] Don’t be misled by the meaning brackets here; the meaning on the right is computed not from the structure, but from f_{sem} taking the meaning of *past* as the first argument and the meaning of the clause as the second argument. The meaning brackets here are a convenience to show what the meaning on the right is the meaning of.

compositionally related to syntax derivationally instead of structurally. I will be interested in particular in the following thesis about derivational compositionality:

(7) Syntax-Semantics Correspondence (SSC):
The arguments to a syntactic function can only be the expressions of the arguments to the semantic function it is paired with.

For many semantic functions, there is a variability in how the sign of the function and its arguments are expressed syntactically, both across languages and within languages. I will argue that the SSC is the appropriate bound on such variability.

1.2 The relation of form to meaning in some standard models

There are four ways in which surface form fails to be appropriate for semantic interpretation:

(8) a. Morphemes are displaced from where they should be for interpretation
b. There are syntactic operations that are irrelevant to meaning, but which create structures difficult to interpret compositionally.
c. Quantifier scopes are not marked.
d Syntax is binary, whereas semantic functions are polyadic

All of these “defects” in the surface form of language are acknowledged, and part of the job of analyzing natural language is to understand each of them in detail. Morphology (more specially, Morpho-syntax) is responsible for relating the surface position of morphemes to the position appropriate for structural interpretation; the derivation in (4) above is an example of that. The re-positioning of *past* onto the verb is not relevant to semantic interpretation, and so it takes place off of the Syntax-Semantics line. Likewise any Syntactic rules which do not affect meaning take place off that line as well; these are sometimes called “stylistic rules”. Quantifiers are assigned scope (“QR”) in the line of derivation from Syntax to Semantics, which simply means that quantifiers are relocated from their surface position to a position in “LF” in which they can be interpreted compositionally. These three considerations give rise to this familiar family of models:

(9) Morphology←Stylistic rules ←Syntax → LF → Semantics

Semantics depends on Syntax and LF, but not Stylistic Rules or Morphology.

The fourth discrepancy between form and meaning arises from the very puzzling binarity of syntax. The semantic functions that are instantiated in natural language are not binary; some take as many as four arguments. A quantifier, for example, takes two arguments a restrictor and scope argument. But natural language implements a syntax in which no function can take more than one argument at a time, as the syntactic structures are strictly binary. There is nothing inevitable about this; all programming languages implement functions with multiple arguments, but almost none of them has binary syntax, and none so thoroughly as natural language. To accommodate the discrepancy, semantic functions are “curried”--an n-place function is rewritten as an function that maps the first argument to an n-1 place function that takes the rest of the arguments in like manner, and so a semantic binarity is achieved to match syntactic binarity.

In sum then all of the defects in (8) are addressed in a model like the following:

(10) Morphology←Stylistic rules ←Syntax → LF → Semantics with curried functions

Syntax is the only generative component. All the others interpret its structures.

The model I will explore here looks more like this[3]:

(11) Syntax/ ------ SSC ------ Semantics
Morphology

The diagram in (11) is meant to indicate that both Syntax/Morphology and Semantics are generative, but tied together by the SSC. There is consequently no need for the parts of the model in (10) which address the inappropriateness of surface form for semantic interpretation, because no forms are interpreted in any case.

So, no QR, no off-line morphology or stylistic rule, no LF, no currying; just the SSC.

As an understood strengthening of the SSC, I will assume that the particularities of the syntactic expression of a given function and its arguments has no effect on the semantics of that function applied to those arguments. These particularities may differ from language to language, or from operator to operator within a language, or even from one occurrence of a given operator to another occurrence, in the same language. We will see cases of all three of these types of variation in evaluating the SSC. This strengthening of the SSC puts severe bounds on possible analyses that are not present in interpretative theories. Where semantics is done by interpreting structures, in principle any detail of the surface form could determine some aspect of the meaning.

Now, since the syntactic functions are binary, and the semantic functions in general are not binary, it would seem it would be difficult, under the SSC, to express all of the arguments of a polyadic semantic function:

(12) f_{sem} (A, B, C, D)
f_{syn} (X, Y)

If these two are paired in the grammar, then the SSC requires that form X be linked with A, B, C, or D, and that form Y be also; but there will be two semantic arguments that will not be represented. Now, while some arguments do remain “implicit” and fail to get syntactic expression, it does seem always possible to express all the arguments if one wants, and the SSC must not prevent that. We will accommodate polyadic semantic predicates by allowing a suite of syntactic functions, all binary, to be paired with a single semantic function:

[3] Williams (2003) chapter 4 has an earlier model of this sort, with something like the SSC implicit in the notion of “Representation”. In more recent work, I have extended the idea to the relation of Syntax to Morphology Williams (2010, 2011)

(13) f_{sem} (A, B, C, D)
f_{syn} (A, B)
f'_{syn} (C, D)
f''_{syn} (D, A)

Here of course "A" as an argument of f_{syn} refers to "the expression of" the semantic argument "A" of f_{sem}.

Importantly, f,f', and f'' in (13) are not alternatives to one another; all of them are coupled to f_{sem}, and so will all apply in a derivation in which f_{sem} applies. Allowing multiple f_{syn}'s to be coupled to a single f_{sem} is a departure from the practice in logic, programming languages, and the vein of semantics that stems from Montague's work, but it is in fact simply an accommodation of the binarity of syntax without curried definitions.

I will explore the potential of a model of grammar built on this basis. Obviously the main concern is, what are the f_{syn}'s. I will suggest in the next section that there is a finite parameterized family of binary f's which together constitute the theory of Morpho-syntax.

1.3 Combine

To make the discussion of the SSC concrete requires a sketch of a theory of Morphosyntax. By Morphosyntax I will mean phrasal syntax plus whatever part of morphology that is not done "in the lexicon"—that is, something like what is called "inflectional morphology". In both recent and traditional conceptions of morphology, for example, Distributed Morphology, Morphology is an interpretive component, in that it is a set of rules that applies to the structures that are the output of syntax. Pursuing the notion introduced above that all components are generative, we will want to fashion a theory of Morphosyntax in which both morphology and syntax are generative. What follows is one way to build such a theory, one derived from previous work (Williams (2008), (2010)).

Consider the two derivations of Affix hopping in (3) and (4). (4) clearly has an interpretive morphology, in that the rule that suffixes the verb with *past* is applying to a structure created by syntax. The desirable feature of (4) is that the two operations are simple: in the first step, *past* is merged with a clause; in the second step, *past* is moved onto the verb. By contrast, the single operation in (3) seems complicated: *past* is merged with the clause, but it is not concatenated with the clause, it is attached to the verb of the clause. But there is a way of thinking of (3) in which the simplicity of (4) is recovered. There really are two operations in the derivation of (3); it is just that they happen *simultaneously.* They can happen simultaneously, because they happen in different dimensions. One of the dimensions is phrasal syntax, and the other is morphology. Two separate questions must be answered by the grammar. First, what does Tense attach to, morphologically? French and English make the same choice here: Tense attaches to the verb. Other languages, say Chinese, make a different choice: Tense is a free form, not affixed to anything. The second choice, for English and French, is, where is the result of the first choice ("V+Tense") to be positioned in phrasal syntax? Here, French and English make different choices: French prefixes that result to a clause, whereas English puts that result in the head position of the clause:

(14) a. John see+present Mary
b. voyer+present [Jean Marie]
see+present Jean Marie[4]

So two generative steps are taken, but in different dimensions: in the morphological dimension, Tense is affixed to V; in the syntactic dimension, the result is prefixed to a clause, or put in the head position of the clause.

As the two operations are coupled, we expect there to be a relation between their arguments. Since both morphology and syntax are binary, we expect one syntactic operation to be coupled with one morphological operation. Below is a first stab at what the arguments could be:

(15) Grammar of Verb Raising (and lowering)
f_{syn} (Tense, Clause) = Tense+Clause
f_{mor} (Tense, Head-of-the-Clause) = Tense+Head-of-the-clause

The second arguments look different but are surely related—the attachment effected by f_{mor} must be to the head of the clause that serves as the second argument of f_{syn}. We could in fact simply put "Clause" as the second argument of f_{mor}, with the understanding that a morphological attachment of a morpheme to a phrase will always be to the head of the phrase. Likewise, we could make the first argument of f_{syn} simply "Tense", with the understanding that f_{syn} will position whatever f_{mor} attaches Tense to. Let us suppose that f_{syn} has two possibilities (corresponding to English/French options discussed above: either T+head can be prefixed to Clause, or T+head can be put in head position of the clause, and that there is a parameter governing this choice (Syn=X/XP). Looking beyond Verb Raising (and lowering), a second parameter is whether or not there is a morphological accompaniment to a particular f_{syn}; this presumably can be governed by whether one of the arguments is a free or bound morpheme (Mor=yes/no).

Note that Syntax is coupled to Morphology in the same manner in which Syntax is coupled to Semantics under the SSC: the arguments to the morphological function are directly related to the arguments of the syntactic function. It is in this sense that Morphology is generative, but constrained by this relation to Syntax, in a way similar to the way that the SSC constrains the relation of Semantics to Syntax.

Now, we can then fully specify what the pair of rules will do by specifying the two arguments and the value of the two parameters. Since the two functions now take identical arguments, we don't need to write them out separately. We can write a single function, "Combine", specifying the two arguments and the value of the two parameters. For Tense, we get the following:
(16) a. Combine(T, Clause)

[4] I ignore here the positioning of the subject in French, which, as in all theories that adopt the VP internal subject hypothesis, is handled by later steps in derivation,

Syn = X, Mor = yes (English)

b. Combine(T, Clause)
Syn = XP, Mor = yes (French)

c. Combine(T, Clause)
Syn = XP, Mor = no (Chinese)

Now, Combine with its two arguments and two parameters is not just for head movement, it is for all of syntax plus morphosyntax; and its parameters are meant to define the space of syntactic variation. So, for example, movement must arise from Combine, and it would not be obvious from the discussion so far how that would be done. For a fuller discussion see Williams (2010), (2012).

With Combine in place as a candidate for the theory of Morphosyntax, we can return to the relation of Semantics to Morphosyntax.

1.4 Wh movement and quantification

If Combine is the only Morphosyntactic operation, then long-distance movement must be implemented as an instance of Combine. This is done in the following way. Only top-level members of the workspace are eligible for operations. When NP$_i$ is embedded as the complement to a verb, [V i] is added to the Workspace, where "i" is a "pointer" to NP$_i$, and the NP itself remains in the workspace available for further operations. So, at the relevant point in the derivation, the workspace contains for example the operator "which" and two expressions that can serve as its Restrictor and Scope arguments. The semantic side of the derivational step is to apply *which* to the (meanings of) the selected Restrictor and Scope arguments. The Morphosyntactic side of the derivation is to apply a set of Combine operations to *the same selection from the workspace.* The two arguments to *which* are its Restrictor and its Scope; so the possibilities for Combine under the SSC are: Combine (*which*, Restrictor), Combine (*which*, Scope), and Combine (Restrictor, Scope). Different languages make different choices. The following illustrates the English choice:

(17) **WH movement:**

Semantics: which (|[man$_j$]|, |[[you think John saw j]]|)

\|

{ which, man$_j$, [you think John saw j] **}** → **{** [which man]$_j$, [**j** [do you think John saw **j**]] **}**

\|

Combine$_{Syn=XP}$ (which, man$_j$) = [which man$_j$]
Combine$_{Syn=\_XP}$ (j, [do you think John saw j]) =
[**j** [do you think John saw **j**]]

First, Combine(which, Restrictor) gives [which man], and that goes into the workspace. Second, Combine(Restrictor, Scope) applies. By convention, when the first argument is already "in" the second argument (the pointer "j" object of *saw*), a pointer to that object is attached to the second argument. As a result, the final Morphosyntactic structure has two pointers "j" to the embedded NP "[which man]"; a "read-out" rule says, pronounce the NP in the position of the highest pointer, thus giving the effect of movement. See Williams (2003), (2010) for an account of the locality of movement in such a theory.

Quantification structures arise similarly; a quantifier takes the same two arguments as WH--Restrictor and Scope--and in principle the same possibilities of expression arise. The following is the English style:

(18) **Quantification:**

Semantics: Q(Restrictor, Scope)
every(man(j) , laughs(j))

{ every**,** man$_j$ **,** [j laughs] **}** → **{** [every man]$_j$, [j laughs] **}**

Combine$_{Syn=\_NP}$(every, man$_j$) = [every man$_j$]

Quantification in English, unlike WH, does not involve Combining a second pointer with the scope, and so the expression “every man” is read out *in situ*. There is no LF or QR in this theory; the scope is determined by the semantic Scope argument, and the “size” of that argument is determined by what point in the derivation the quantifier comes into play, as in classical Montague grammar, except that the timing of events is different here, as will be explained .[5]

The difference between WH-movement and Quantification is minimal; there is no semantic difference, the difference lies solely in the fact that English-style Quantification lacks the Combine(Restrictor, Scope) operation that gives rise to movement.

In the case of both Wh and other quantifiers, the quantifiers themselves are in the workspace, and so complex quantifiers can be derived, such as “exactly how” and “almost every”. We must assume that the Combine parameters of the complex quantifier are inherited from the simple quantifier in it.

In both of these constructions, the arguments to Combine are exactly the (expression of) the arguments to the semantic function that is applied in parallel; so both comply with the SSC. A consequence here is that “Every man” does not get an interpretation; “Every man laughs” gets an interpretation (Every(man(j) , laughs(j))), but “Every man” does not play a role in forming that meaning, as it is purely a Morphosyntactic object with no semantic analog.

[5] The derivation of D-quantification structures outlined here has struck some as similar to derivations using "Cooper storage", as first proposed in Cooper (1983). In Cooper's theory, when a quantified NP is encountered in the derivation, say in direct object position, a variable is merged with the verb, and the quantified NP meaning is put in a special "store" to be accessed later; later in the derivation it is removed from the store and combined with a clause that has derivationally inherited the store, thereby effectively assigning scope to the NP. The theory here differs on many points. First, in the theory here there is no such thing as a quantified NP meaning, so obviously no such thing is stored. Second, there is no special storage mechanism, there is only the workspace, which stores everything indifferently. Third, Cooper's theory gives rise to "semantic ambiguity" in the sense that there is more than one semantic derivation (resulting in distinguishable meanings) for a give syntactic derivation; here, as in classical theories, including Montague’s, different meanings can arise only if there are different syntactic derivations.

A consequence of this treatment of quantification is that the disjunctive subject in the following case does not get a meaning[6]:

(19) Every man or no woman left

The subject does not get a meaning because its constituent parts, “every man” and “no woman”, do not get meanings. And yet the entire sentence does get a meaning. Importantly, the prominent reading of (19) does not have the logical structure in (a) below, but rather (b), where the quantifiers are subordinate to the disjunction:

(20) a. Every man x [no woman y [[x or y] left]
b. [Every man left] or [no woman left]

This situation presents a potential problem for the theory of quantification put forward here; “not getting a meaning” now must be understood to be a property of phrases capable of indefinitely deep recursion: [NP or [NP or [NP …...]. So the challenge of finding a meaning for (19) without positing meanings for “[Every N]” must be met.

In the present theory, (19) must be derived by syntactic conjunction reduction, from something like (21),

(21) { [Every man]$_i$, [i left] , [no woman]$_j$ $_,$ [j left] }

To sketch the analysis: the rule for the syntax of *or* must be formulated so that it can combine two Quantified NPs if their corresponding scopes are identical up to re-indexing, in which case one of the scopes is eliminated from the workspace, yielding something like the following:

(22) { [Every man$_i$ or no woman$_j$]$_i$, [i left] , ~~[j left]~~ }[7]

In other words, cases like (19) must result from syntactic conjunction reduction. That this analysis is forced shows that the theory is restrictive, not that it is impossible, or even implausible.

Such an analysis must be grounded on a real theory of ellipsis, which I will not attempt here, except to point out one obvious property. We must allow predicates like [i left] and [j left] to count as “the same” for purposes of coordinate ellipsis; on the other hand, we cannot allow [most boys]$_i$ to count as the same as [most boys]$_j$, otherwise we would derive the notorious conjunction reduction here:

(23) a. Most boys danced and most boys left **=/=**
b. Most boys danced and left[8]

[6] Thanks to Manfred Krifka for pointing out this consequence to me.

[7] It will not matter that the indices (i and j) are not identical here, as the meaning is not computed from the derived structure.

[8] cf. Partee (1970).

In the best case the difference between (22) and (23) would be reduced to something independent of coordinate structures, perhaps something like a notion of “anaphoric equivalence”. We know from VP ellipsis that two predicates are anaphorically equivalent if they are identical modulo systematic re-indexing:

(24) John$_i$ [t_i left] because he thought that Bill$_j$ had ~~[t_j left]~~

On the other hand, two Quantified NPs are never anaphorically equivalent[9]; e.g. in a pro-drop language like Italian, one would not expect the following pair to be equivalent:

(25) a. Ogne studente pensa che e un genio
Every student thinks that pro is a genius
b. Ogne student pensa che ogne student e un genio
Every student thinks that every student is a genius

Although there is much to be filled in here there is a suggestion that a general theory of ellipsis can account for the difference between (22) and (23), and other like cases.

In the remainder of this essay I will pursue the SSC as an explanatory principle. It becomes especially powerful when Morphosyntax is limited to the single function Combine, according to which there are only a limited number of ways of forming the expressions that correspond to the meanings produced by the semantic functions. I will present two case studies, one concerning Adverbs of quantification, and the other, the placement of clauses related to operators like *too, so, more, etc.,* In both cases I think that the SSC leads to some surprising, and correct, conclusions. In section 2, I will suggest that all quantifiers are essentially adverbs of quantification, and that the differences between A-quantifiers and D-quantifiers is purely morphosyntactic. I will show further that the behavior of focus-marking adverbs is limited by the SSC and Combine. In section 3 I will show how the particularities of apparent clause movement are fixed by the SSC and Combine.

2. Case 1: Scope Markers, Adverbs, and Quantifiers

In this section I will argue that the variability in the realization of quantificational functions (with their arguments) is well captured by the SSC and the parameterized Combine function. Semantically all of the quantifiers discussed here have semantics of the form Q(Restrictor, Scope), but see section 3 for another species. In the first section, I will make a necessarily superficial cross-linguistic survey of that variability. In the second section, I will argue that the difference between A- and D-quantifiers is purely morpho-syntactic, and that the properties that have been suggested to distinguish them—selectivity and conservativity—do not do so. In the third section, I will take a close look at the interaction of A-quantifiers and Focus, arguing that the SSC makes sense of the phenomenon called “attachment to Focus”.

2.1 Typology induced by Combine parameters

[9] Indeed, the fact that there is no pronoun that can resume the meaning of a quantified NP is no mystery if quantified NPs do not have meanings.

The Combine Parameters provide a first approximation to a typology of quantificational elements of all sorts. The semantics is always the quantifier applied to its two arguments, the Restrictor and the Scope. The permissible Combine operations can take as their arguments the Q, the Restrictor, and the Scope, in all combinations.

(1) Semantics: Q (Restrictor, Scope)
Morphosyntactic Possibilities under the SSC:
1. D-quantifiers: Combine (Q, Restrictor)
2. A-quantifiers: Combine (Q, Scope)
3. D-quantifiers with scope markers:
Combine (Q, Restrictor), Combine (Q, Scope)
4. Scope-motivated movement: Combine(Restrictor, Scope)
5. Clitic scope marker Combine($Q_{\text{clitic settings}}$ Scope)
6. Affixal quantifier: Combine (($Q_{M=root}$, Scope)

I will briefly sketch the empirical basis for the typology so defined, showing that each is instantiated in some quantificational structure.

French negation (D-Quantifier with scope marker; multiple restrictors):

French two-part negation (*ne-personne, ne...rien*, …) is an instance of a quantifier with both a Restrictor marking and Scope marking. The scope-marker is morphologically a clitic:

(2) Semantics ne-personne(Restrictor, Scope)
Morphosyntax:
Combine(ne_{clitic}, Scope)
Combine(personne, restrictor)

ne..personne is an operator than can have multiple restrictors, a phenomenon that will show up frequently in this section ("&" means "ambiguous", and "~&" "not ambiguous"):

(3) a. Paul accepte de ne recevoir personne
Paul agrees not to see anybody
b. Paul n'accepte de recevoir personne
Paul does not agree to see anybody
c. ~&Personne n'accepte de ne rencontrer personne (Double Negation)
Nobody agrees to see nobody
d. &Personne n'accepte de rencontrer personne (Double Negation *or* Neg Concord)
Nobody agrees to see nobody / to see anybody
(Godard (2004))

(a) and (b) are simple cases in which both Combine operations apply. The derivation procedes by Combine (ne, de recevoir) and Combine(personne, $[]_{NPi}$):

(4) **{** ... [de recevoir i], $[]_{NPi}$ …**}** ➔ **{** ... [de ne recevoir i], $[\text{personne}]_{NPi}$ ...**}**

The attachment of *ne* is done with parameters appropriate for clitics (perhaps Syn=V^0). The scope is only approximately marked; the scope is the whole of [de recevoir i], even though the scope-marker ends up inside of that expression by virtue of its Syn parameter value.

(c) illustrates a case in which the same quantifier is applied twice, with different scopes, giving a double negation reading.

(d) illustrates two different derivations, one for each of the two readings. They both require special comment. The Double Negation reading is derived by applying *ne..personne* twice, with the same scope but with different restrictors. What is not explained is why we do not get two *ne*'s in that reading, and I have no special insight about that. The other reading arises from a single application of *ne..personne,* but with two restrictor arguments:

(5) Semantics: ne..personne(Scope, (Restrictor1, Restrictor2))

Here I have represented the second argument as a sequence of Restrictor arguments, and they are each eligible for a Combine operation. So the Negative Concord derivation goes this way:

(6) {ne..personne,personne, $[]_{NP1}$, $[]_{NP2}$, [1 accepte de rencontrer 2] } →
{$[\text{personne}]_{NP1}$, $[\text{personne}]_{NP2}$, [1 n'accepte de rencontrer 2] }

We must countenance multiple restrictors for operators other than negation. For example (the Restrictors are bolded).:

(7) a. Which **man** bought which **boat**?
b. So many **people** had so much **fun** that we will do it again
c. So many **people** ate so much **cake** that they got sick that we won't do that again
d. More **people** ate more **cake** than we expected.

So, alongside negation, *WH, so,* and *more* take multiple Restrictor arguments. In the case of *so* we get both "*so* concord" (b) and "double *so*" (c). So its seems a general thing for quantifiers to do this. It is arguable that D-quantifiers like *few* and *many* take multiple restrictors, if sentences like the following are ambiguous:

(8) a. Many people ate many things.
b. Few customers bought few books

These clearly have double-Q readings. But I think they also have "Q-concord" readings, indicating multiple restrictors. For (a): there were many people eating and there were many things eaten, but perhaps no single person ate many things. For (b): there were few customers and few books were bought, a reading sharply different from its double-Q reading. The phenomenon of Q-concord seems common enough that perhaps what needs explaining is its occasional absence. This is an important conclusion, as it undermines any special connection between unselective binding and A-quantification.

Hindi WH (clitic scope marker alternating with Wh movement):

Sometimes we find different versions of Combine competing for the same Combine (X, Scope) operation. The following is an illustration:

(9) a. raam **kyaa** maantaa hai ki siitaa kisko pyaar karti hai?
Ram what believes be that Sita who-acc love does be
Matrix Question: 'Who does Ram believe that Sita loves?'
*Embedded question: ‘Ram believes who Sita loves’
b. raam maantaa hai ki siitaa **kisko** pyaar karti hai
Ram believes be that Sita who-acc love does be
*Matrix Question: 'Who does Ram believe that Sita loves?'
*Embedded question: ‘Ram believes who Sita loves’
c. **kisko** raam-ne (*kyaa) sochaa ki bill-ne dekhaa
*who-Acc Ram-erg (*what) thought that Bill-erg saw*
‘Who did Ram think Bill saw?’ Malhotra and Chandra (2007)

(b), with Wh in-situ, is ungrammatical as a matrix question because that requires a matrix scope marker; it is anomalous as an embedded question because the governing verb is not interrogative. As Malhotra and Chandra explain, Wh must be marked either by movement or by a scope-marker, and they go on to say,

> “This is a rather interesting set of puzzles. On the one hand, we find wh-scope markers behaving like X^0’s blocking other clitics from adjoining to the verbal head. On the other hand, these elements also seem to be sensitive to the presence of other A-bar elements in the same clause, indicating that they are XPs undergoing A-bar movement.”

From the point of view of usual theories, it is indeed puzzling how an X^0 element could be incompatible with XP movement. But from the point of view furnished by Combine and the SSC, it is more understandable. Some languages don't mark the scope of Wh at all (Chinese; no Combine(X, Scope) operation). Some languages mark Wh scope with a fixed marker (Japanese, Combine(ka, Scope)), Some languages mark Wh scope by prefixing the Restrictor to the Scope (English, Combine (Restrictor, Scope)). There are no other possibilities consistent with SSC. From that perspective, it is not surprising that some languages allow alternatives from this limited set. For example, French allows some in-situ question words, in addition to having movement (French *quoi,* doesn't move, others do move). So the Hindi situation is almost expected: of the small number of possibilities allowed, it allows two of them, and they are of different character. In section 3 we will several other such cases in an entirely different domain.

In our terms, Hindi allows two options for Combine(X, Scope):

(10) a. Combine(kyaa, Scope) with clitic Values (Syn = Tense)
or

b. Combine(Restrictor, Scope) with _XP (Syn = XP)

(the (b) option is movement, as the Restrictor argument is prefixed to the scope.)

The SSC does not explain why there cannot be both a scope-marker and movement in the same clause. It could be that the incompatibility is language-particular. But if it is not, then why not? We could strengthen the SSC in the following way: for each argument A of the semantic function, there can be at most one Combine(X, A) operation.

Mayali Quantificational Prefixes (scope-marker as verbal affix):

If a quantifier has values for the parameter Mor, it will combine with the head of its Scope argument or of its Restrictor argument (there is an example of the later in section 3), there are no other possibilities. Head-of-scope attachment is found with Mayali quantificational prefixes:

(11) a. Garri-**djaark**-dulubom duruk
we.plural-together-shootPP dog
We all shot the dog(s) (ex 43)

b. Gunj barri-**bebbe**-yeame-ng
kangaroo 3sP-DISTR-spear-PP
They each killed a kangaroo (ex. 53)
(Evans (1995) p. 218)

(12) Combine ($Q_{Mor=root}$, Scope)

Again, the surface position of the affix only approximately marks the scope—the scope is the entire clause, as the the subject and the object are eligible restrictors. The deviation from accurate marking is the (perhaps entire) subject-matter of Morpho-syntax.

2.2 A-quantifiers are not semantically different from D-quantifiers

A-quantifiers are supposedly different from D-quantifiers in two ways: only A-quantifiers are typically non-selective, and only D-quantifiers are necessarily conservative.

2.2.1 Selectivity

In regards to selectivity, I will suggest that it is matter of Morphosyntax, and selectivity does not semantically distinguish the two patterns of quantification.

Partee (1991), for example, suggests the following about the difference between A- and D- quantification: "...D-quantification and individual-level predication favor quantifying over individuals, while A-quantification and stage-level predication favor quantifying over episodes or cases"; she cites among others the following cases:

(13) a. Almost every woman who owns a dog talks to it
b. Almost always, if a woman sees a dog she talks to it.
Partee (1991) p. 443

pointing out that (a) is most easily understood as a quantification over women, and (b) as a quantification over woman-dog pairs.

Partee did not intend this as a “deep” difference between A- and D-quantifiers (note the weak “favor”)--but many have taken it to be. If Partee is at all correct, it may show nothing more than the fact that A-quantifiers can take multiple restrictors, as we have already observed in the case of French negation. But D-quantifiers do as well, as we have also seen. Heim's (1982) analysis of Donkey sentences in fact amounts to exactly that; in her analysis, “Every” binds of course “farmer”, but also a second variable with a different restrictor (“a donkey”) in “Every farmer who owns a donkey beats it”; The second restrictor is not marked as such, and in that respect it resembles some cases, but not all cases, of A-quantification (again, French negation marks all restrictors).

But there are many D-quantifiers that can take multiple restrictors, all marked; we have already seen French negation, *few*, *many*, *more*, and *so.* Here are some more:

(14) a. Too many people bought too many refrigerators for the economy to fail[10]
b. Enough people bought enough refrigerators that the economy can't fail
c. How many people bought how many refrigerators?

The capacity to take multiple restrictors is certainly not limited to A-quantifiers; in fact, in light of these examples, it seems typical of quantification *in general.*

If it is a given that A- and D- quantifiers are both quantifiers, then the question is, what is the theory common to them both, and how, precisely, do they differ. It is my contention that they are not different semantically, but only morphosyntactically; and furthermore, the morphosyntactic difference is exactly what falls under the scope of the Combine options. Semantically, they are both of the form

(15) Q (Restrictor, Scope) or Q (Restrictor*, Scope),
where Restrictor* allows for multiple restrictors

You cannot tell from (15) whether Q is an A- or a D-quantifier, or for that matter, a hybrid, which marks both scope and restrictor. The difference comes in what syntactic Combine operations are associated with a given quantifier; if Combine(X, scope) then it is a A-quantifier, if Combine(X, Restrictor) then it is a D-quantifier, and if it has both operations it is hyprid.

[10] I am treating “too many” as a complex D-quantifier. One grounds to do so is that in many languages a single morpheme suffices to express the combination (e.g. French *trop*). But the assumption is not necessary to the point that I am making; we could take *too* as an operator on its own marking two restrictors (the two *many*'s), and it would then itself be an instance of a restrictor-marking degree operator taking multiple restrictors.

To repeat the point in different terms, the term “unselective” should be applied to *both* arguments of a quantifier. An A-quantifier is unselective with respect to its Restrictor argument(s)[11], but conversely, a D-quantifier is unselective with respect to its Scope argument, which is why D-quantifiers give rise to scope ambiguity that does not arise with A-quantifiers. Since D-quantifiers are ambiguous in their scope, LF is required in standard accounts, so that Semantics can be deterministic—LF disambiguates syntactic structures with respect to quantifier scope. Symmetrically, LF is also needed for A-quantifiers, as it must be determined which restrictors they will bind, else again semantics will not be deterministic.

In the scheme I am proposing, LF is needed for neither type of quantifier. Once the selection of items in the workspace is made (including the quantifier itself, the scope argument, and the restrictor arguments) the semantic side of the derivation is determined. The syntactic component of the derivation, which proceeds separately but with the same choice of items from the workspace, will be shaped by the Combine operations and parameters associated with the given quantifier. It is in this sense that I claim that the difference between A- and D-quantifiers is purely syntactic, and that the term “unselectivity” as it is currently used gives rise to a misleading picture of how form and meaning are related to each other in quantificational structures.

2.2.2 Conservativity

Turning to Conservativity, we may again conclude that there is no real difference between A-quantifiers and D-quantifiers, despite the usual claim that Conservativity is a property that characterizes D-quantifiers (e.g. Keenan and Stavi (1986)). To do so though we have to be careful about what the arguments to the A-quantifiers are. We say a quantifier is conservative if $Q(R,S)=Q(R,S\cap R)$.

Since A-quantifiers do not mark their Restrictor argument, there is some ambiguity about what it is, and the role of Focus in fixing the Restrictor argument is well-known:

(16) John only likes PIE

We may define *only* in terms of Rooth's (1992) “Focus Value”, as Rooth in fact did himself. The Scope argument of *only* is marked (it is the whole clause), and the Restrictor argument is obtained by abstracting over the Focus (PIE in this case):

[11] Some readers have raised the question of whether it right to speak of unselective binding as leading to ambiguity, at least the kind of ambiguity that is on a par with scope ambiguity. If the surface configuration fully fixes the question of which unselective binder binds what free variables, then it is truly different from scope ambiguity. Heim (1982, 147 ff.) at first blush seems to propose that there is no ambiguity: "it is the *lowest* quantifier that picks up the subscript of an indefinite" (p. 147, emphasis in the original). This makes a strong but patently false prediction; however, Heim's actual implementation of unselective binding nullifies the prediction. In fact, every unselectively bound restrictor is moved (she calls it "NP prefixing") to a position immediately subordinate to its unselective binder; so, while the unselective binding itself is indeed unambiguous, the movement that precedes it has the same capacity to introduce ambiguity as the rule assigning scope to D-quantifiers (if it is not that very rule itself, "QR" in standard formulations). If this is the right implementation then there is no question that unselective binding involves the same kind of ambiguity as D-quantification, and in fact Heim's implementation suggests that the ambiguities have a common source.

(17) Scope = |[John likes pie]|
Restrictor = {|[John likes X]|, for all X} (Rooth's (1993) Focus Value)
Definition: only(Restrictor, Scope) = Restrictor - {Scope}= {}
for (16), only ({|[John likes X]|, for all X}, |[John likes pie]|) = {} is True

But if this is the correct meaning for *only*, and the correct identification of the arguments of *only,* then *only* is trivially conservative; since the Scope argument is always at best an instance of the Restrictor argument; if R-{S} = {}, then R-{S}∩R = {} as well. So *only* is conservative. One could arrive at a different conclusion only by identifying the arguments of *only* in some other way, and that would get the meaning wrong. What is said here about *only* applies to all two-place focus-sensitive adverbs, so they are all conservative. And they differ from two-place D-quantifiers only morphosyntactically. As Bill Hawkins (p.c.) puts it, *every* is an adverb.

Keenan (2003) has suggested that Conservativity is relevant to *only* in a different way: *only* is conservative on its *second* argument (compared to the usual case where a quantifier is conservative on its first (Restrictor) argument). If I am right, he has simply misidentified the "first" and "second" arguments, or, as I would put it, the Restrictor and the Scope arguments, and once that is untangled, there is no need for the notion "conservative on the second argument".

Keenan's treatment of the Conservativity of *only* really only covers cases in which it is prefixed to a noun, as in the following:

(18) Only dogs bark

and he defines *only* as

(19) Only A B = all B A

Given the definition, only will be conservative on its second argument ("CONS$_2$") because *all* is conservative on its first argument ("CONS$_1$").

Keenan suggests that CONS$_2$ is the distinguishing property of NPs that can appear in "there is/are NP". I think that his finding is accidental in the sense that he has identified the wrong first and second arguments for *only*, treating it as a D-quantifier for the small fraction of cases in which that is feasible. In the first place, Keenan's principle, that only Qs conservative on their second argument can be "there" associates, is arbitrary—it could have been the opposite. In the second place, except for *only* as used in (18) (and *mostly*, which for present purposes acts the same), all of the quantifiers that are CONS$_2$ are actually conservative on their first argument as well as their second, as they are symmetric, so maybe the property relevant to "there is" is symmetry, rather than CONS$_2$. Finally, given the resources for defining quantifiers, it is hard, but not impossible, to define a quantifier that is not conservative on either argument, and so admitting CONS$_2$ as an admissible category of quantifier considerably weakens the claims that can be made about quantifiers in general. In my view the analysis presented here is better than Keenan's because it is fully general in that it enforces the original conservativity for *all* quantifiers in the same way.

Keenan's treatment of *only* is not general enough even for the pre-nominal cases he treats once the role of Focus is taken into consideration; e.g.

(20) a. Only dogs with striated vocal chords bark =/=
b. Every barker is a dog with striated vocal chords.

(a) is not equivalent to (b), because it is consistent with (a) that there could be non-dogs that bark. Once Keenan's definition is revised to take Focus into account, it will be essentially equivalent to mine, and so fall under CONS_1 and not CONS_2**.**

The one thing that Keenan gives that I cannot supply a substitute for here is a specification of what can be an associate in the "there is" construction, since I am suggesting that the CONS_2 property is not linguistically significant. I think though that Keenan's most basic assumption is faulty, the assumption that *only* is a syntactic alternative to *every, some, the, 3, etc.* in the sorts of sentences he discusses. In fact, pre-nominal *only* co-occurs with all of these other quantifiers, and it is always the quantifier that *only* occurs with that determines whether the DP can occur as a *there* associate:

(21) There were only 3/a few/some/*every/*the/*most dogs in the yard.

That is, if the quantifier is OK with *only*, it is also OK without *only*. *Only* only appears to play a role in licensing an associate when there is no overt second quantifier:

(22) There were only dogs in the yard

But again, whatever the analysis of bare plurals is, surely the grammaticality of (22) reduces to the grammaticality of (23):

(23) There were dogs in the yard

and so (22) is really just a special case of (21) (perhaps with a covert second quantifier), and so *only* plays no role in licensing *there* associates.

In addition, Keenan never makes that awkward step of saying that a quantifier must be either CONS1 or CONS2, presumably because he is aware of the awkwardness. But this is a must, as there are no quantifiers that are neither. E.g., there are no quantifiers like the following:

(24) Q (A, B) = |A-B| = |B-A|

Finally, I think that Keenan's theory fails to make sense out of the semantic asymmetry between the arguments of a quantifier. The R argument, however it is expressed (Focus Value, N'), *restricts the universe* quantified over; the S argument then asserts something relative to that restriction. Conservativity simply asserts that every quantifier works this way, regardless of how the arguments are expressed.

2.3 Focus Marking

I want to return to the question of what the arguments to *only* are. In the previous section, we took the arguments of *only* to be its Scope, and as Restrictor, the Focus Value, where the Focus Value is determined by the Scope and the Focus, as in (a,b). But given the relation of Focus and Scope to Focus Value, we really should take the Scope and the Focus to be the arguments to *only,* and rewrite the definition as (c), rather than (b):

(24) a. Focus Value = FV(Scope, Focus)
b. Def: *only* (FV(Scope, Focus), Scope) = FV(Scope, Focus) – {Scope} = {}
c. Def: *only* (Scope*,* Focus) = FV(Scope, Focus) – {Scope} = {}

In other words, if we adopt (c), the Focus Value is not directly an argument of *only,* but is a component of its definition. The Focus argument is serving as the Restrictor argument, and can do so because of the special relation of the Focus to the Focus Value and the Scope: the Focus and the Focus Value are "duals" of one another with respect to the Scope—given the Scope, either can be derived from the other.

I think that there is a good a-priori reason to take the Focus and the Scope to be the arguments of Focus-sensitive adverbs: the Focus and the Scope correspond to different Morphosyntactic objects that are in the Workspace, whereas the Focus Value does not correspond to anything in the Workspace. In other words, the SSC suggests that the Scope and Focus are the arguments to *only.* But in fact there is a further benefit to rewrite the definition as in (24c): we can derive some further facts about focus-sensitive adverbs under the SSC. The SSC tells us that there can be Combine operations involving only expressions of the arguments of a quantifier; if the arguments were the Scope and the Focus Value, we would get only one possibility, because the syntactic expression of the Scope cannot be distinguished from the syntactic expression of the Focus Value. But if the Restrictor argument is really the Focus itself, then we get two attachment possibilities, and in fact it seems that they are both realized:

(25) a. John only eats [gummy bears]$_F$
Combine(*only*, Scope)
b. John eats only [gummy bears]$_F$
Combine(only, Focus)

In (a), *only* is attached to the Scope (the Focus is marked, and *only* is not attached to it); but in (b), which shares a reading with (a), *only* is attached to the Focus. These are the only two possibilities allowed under the SSC, and *only* exploits both, apparently. The second possibility arises only if the Restrictor argument is the Focus itself.

It follows that some sentences will be ambiguous exactly on the point of which argument *only* is attached to:

(26) John only eats gummy bears

The two possibilities are:

(27) a. only (gummy bears, John eats gummy bears) =
it is only gummy bears that John eats
Combine(only, John eats gummy bears) (attach *only* to the Scope argument)
b. only (eat gummy bears, John eats gummy bears) =
it is only eating gummy bears that John does
Combine(only, eats gummy bears) (attach *only* to the Restrictor argument)

These are exactly the two possible readings available to (26) if it has the nuclear accent pattern. With that pattern, the only possible Focuses are “gummy gears”, leading to (a) and “eats gummy bears”, leading to (b).

Beaver and Clark (2008) claim that *always* is fundamentally different from *only*, in that with *only*, the Restrictor is always the Focus Value, but with *always*, the Restrictor can be anaphorically filled in as something different from the Focus Value. They give the following in support:

(28) a. #Mary only completed her tests, and Mary only completed her homeworks
b. Mary always completed her tests, and Mary always completed her homeworks

(a) is perceived as contradictory, but (b) not.

If we assume that “tests” and “homeworks” are the two focuses, and that the Restrictors are “{Mary completes X, for all X}” for both conjuncts in both examples, then (a) will be contradictory, but so will (b). To account for the fact that (b) is not perceived as contradictory, Beaver and Clark suggest that the Restrictor is not always the Focus Value for *always*, as it is for *only*. Rather, *always* has the option to fix the Restrictor anaphorically in a way that is not related to the Focus Value.

I agree with the judgments, but think that Beaver and Clark have misinterpreted these examples in a rather subtle way. Once that misinterpretation is corrected, then there is no reason to distinguish *only* from *always* in the way that they propose, and in fact the Restrictor is the Focus Value in all cases.

To put it in a nutshell, I think Beaver and Clark misapprehend what the Focus Value is for these two cases, and this is because they misapprehend what the “Local Focus” of the two conjuncts is, particularly for (28b). If we put either of the conjuncts in (28b) by itself, we see that the natural focus goes on the verb, rather than the direct object:

(29) a. Mary always COMPLETED her tests
b. ?Mary always completed her TESTS

(b) is in fact narrowly contrastive on the direct object.

This is probably because *complete* itself has a presupposition that one is taking a test, and that presupposition makes “taking a test” or “starting a test” into alternatives to “completing a test” on a scale, and those alternatives naturally bring the verb into focus, since the objects are the same. The same goes for the second conjunct as well, so the “true” focus structure of (28b) is,

(30) Mary always COMPLETED her tests and Mary always COMPLETED her homeworks

Given this accent pattern, there is no contradiction for (28b), even if we take the Restrictors to be the Focus Values, as the Focus Values of the two conjuncts are {Mary Xed her tests, for all X} and {Mary Xed her homeworks, for all X}, respectively. So we do not need Restrictors other than the Focus Values to account for (28b).

The only problem is, you cannot pronounce (28b) in the way indicated in (30). The repetition of *complete* with different direct objects across the two conjuncts demands the de-accenting of *complete* and the consequent accenting of the direct objects; if that accent pattern is applied, then we do indeed arrive back at the pronunciation appropriate for (28b).

The problem is that we have competing claims on the accent pattern; a local claim based on the presupposition of *complete* and the scale that that gives rise to, and a global claim based on the contrastive environment that arises across the two conjuncts. The question is, which claim wins. Clearly, given the ungrammaticality of the pronunciation indicated in (30), the more global claim wins. It is proposed in Williams (1996) that this is always the case. This allows us to posit local focuses, and Focus Values, for each of the conjuncts even though on the surface those Focuses are not marked intonationally.

So the Focus in each of the two conjuncts is actually on the verb, even though that Focus does not register phonologically because of the contrast across the conjuncts. Nevertheless, the local Focus generates a Focus Value in each of the conjuncts. Then, given that the conjuncts have different Focus Values ({X her tests}, for all X, and {X her homeworks} for all X)), no contradiction arises.

The reason that the same thing does not arise with *only* is that locally, *only* is not compatible with the local presupposition:

(31) ??Mary only COMPLETED the homework

If “complete” is at the top of a scale that includes “start”, “do some of”, “do most of” and “complete”, then it will be incompatible with *only,* as *only* is incompatible with the top of a scale (e.g. **John has only everything*). However, it is in fact possible to take *complete* to be part of a different scale, a scale in which *complete* is not at the top--consider the scale (“start”, “complete”, “complete and submit”, “complete, submit, and get credit for”). With that scale in mind, *only* IS compatible with *complete:*

(32) Mary only COMPLETED the homework (she did not submit it)

And, using the same scale, and putting the verb in a contrastive environment that overrides the claim of the local Focus to set the intonation pattern, we can construct an example with *only* that is *not* contradictory, because each conjunct has its own local Focus Value:

(33) Mary only completed her TESTS, and she only completed her HOMEWORKS.

This is in fact identical to (26a), but now interpreted with respect to a scale that makes the local Focus Values similar to those in (26b).

If this line is correct, then there is no fundamental difference between *always* and *only*—in both cases, if valid local Focus Values based on focusing the verb exist, then contradiction will be evaded. It is simply *easier* to do that with *always* that with *only*.

If quantificational adverbs can attach to either their Restrictor or their Scope, then the question arises, how do you know which argument is the Restrictor and which the Scope? I think there are at least two independent means of telling which is which. First, the Restrictor must occur within the Scope, to avoid "Vacuous Quantification". Second, there is a semantic asymmetry between the two arguments discussed in the previous section. Some quantifiers express a "proper fraction" where the Restrictor provides the denominator and the Scope provides the numerator. For example, "John usually takes Mary to the MOVIES" says that | John takes Mary to the moves | / |John takes Mary to X| > 0.5. It is important to have an independent semantic means of distinguishing Restrictor and Scope, as we cannot rely on the position of the quantifier. In fact, as we will see in the next section, some cases are ambiguous as to whether the attachment site is the (syntactic expression of) the Restrictor or the (syntactic expression of) the Scope.

However, and this is important for my overall argument, there are uses of these adverbs that cannot be handled this way:

(34) John takes Mary mainly to the MOVIES.

We cannot take *mainly* to be attached to its Scope in these examples; clearly, the subject and the object are in the Scope but are not c-commanded by *mainly*. The SSC tells us that the only attachment possibilities are to the Scope or to the Restrictor, so we must conclude that "to the movies" is the Restrictor argument, and so the Restrictor argument does in fact have syntactic expression (as opposed to being anaphoric) and the relevant Morphosyntax is the following:

(35) Combine$_{\text{Syn=XP}}$(*mainly*, Restrictor)

That is, the adverb is prefixed to the Restrictor. We assume, as in the case of *only,* that the Restrictor argument is the Focus. Attaching adverbs like *mainly*, *usually*, etc., to the Focus is not as free as it is with *only,* but I think it is justified to think of *mainly* as attached to the Focus in (34), because (34) is well-formed only if "to the MOVIES" is construed as the narrow focus of the sentence, a restriction that does not follow from Focus projection alone.

Since the Restrictor argument is syntactically pinned down by the attachment of *mainly* to it, there will not be the same way to evade contradiction with silent local focuses as in the

Beaver and Clark cases; and in fact, when *mainly* attaches to the Focus, the contradiction again arises, as the following contrast shows:

(36) a. John mainly takes Mary to the movies and John mainly takes Mary to the doctor
b. #John takes Mary mainly to the movies and John takes Mary mainly to the doctor

The SSC forces this conclusion: Combine(*mainly*, X) forces X to be the syntactic expression of a semantic argument of *mainly*; since X cannot be the Scope argument in (33b), it must be the Restrictor, and so the Restrictor cannot be the (silently focused) verb for such cases, and so contradiction arises, just as it would with with *only*. (a) escapes this because the placement of *mainly* makes it possible to take always as marking Scope, leaving the Restrictor free to be anaphoric.

As before, the difference between Focus attachment and Scope attachment is morphosyntactic, not semantic. Cinque (1999) excluded Focus-attached adverbs from f-structure ordering, largely because they do not appear in the expected F-structure positions, which in our terms, is always the Scope-marking position. This exclusion is mistaken on several grounds. Consider the observation (Williams (1974)) that certain adverbs are excluded from clauses that are "too small"; for example, the complement of *want* appears too small to house *probably,* whereas *that*-clause complements and ECM complements to epistemic verbs are "big enough":

(37) a. *I want John to probably win the prize
b. I think that John will probably win the prize
c. I believe John to have probably won the prize

But now consider *probably* in a Focus-attached use; here attached to "to the movies":

(38) Bill wants John to take Mary probably to the movies

At first blush, it looks like *probably* can occur in the complement of *want* just in case it is Focus-attached, thus bolstering Cinque's decision to exclude such attachment from F-structure ordering. However, close attention to the meaning of the example suggests otherwise. First, although *probably* occurs in the complement of *want,* it is not interpreted there; its scope is necessarily the matrix clause. Furthermore, the Focus-Value is computed by taking "to the movies" to be a *matrix* focus, not the focus of the complement clause, and the Focus-Value is

(39) { |[Bill wants John to take Mary to X]|, for all X }

The assertion is that, given (39), "X is probably to the movies", which is adequately paraphrased by the following:

(40) It is probably to the movies that Bill wants John to take Mary

So in fact, even in focus-attached uses, *probably* is excluded from the complement of *want, in that it cannot take its scope there.* In Cinque's (1999) terms, this means that even Focus-attached adverbs are subject to f-structure ordering, although it is not clear how to implement that under his hypothesis about adverb genesis.

In our terms, the conclusion is clear—the difference between Scope marking uses and Focus marking uses is purely morphosyntactic; it is the difference between (a) and (b) below:

(41) Semantics: probably (Focus, Scope) = |Scope| / FV(Focus, Scope)| > 0.5
Moprhosyntax:
a. Combine(*probably* Scope)
or
b. Combine *(probably,* Focus)

Importantly, for adverbs that do not take two arguments, Focus-attachment is not a possibility—the adverb must attach to its Scope, again as dictated by SSC. Supposing that *suddenly* is a monadic adverb, we will be unable to create examples parallel to (38)[12]:

(42) John wants Bill to take Mary suddenly to the movies

The only interpretation is one where *suddenly* modifies *take;* there is no interpretation like the following:

(43) It suddenly is to the movies that John wants Bill to take Mary

Again, the SSC determines this; *suddenly* does not take a Focus argument, and so must attach to its Scope; that fixes its scope in (43) to be the lower clause.

3. Case 2: Clausal Complements to Degree Operators

In section 2, we considered cases in which an operator takes two arguments, and showed, in principle, that it could attach to either one of them, or sometimes even to both of them. In this section we will consider operators that take as many as four arguments, and here we will find that there are as many as four Combine operations enabled under the SSC (leaving aside the already discussed possibility of multiple Restrictors). A principle suggests itself: there is one Combine operation per argument.

It is a curious thing that semantic functions are polyadic while the syntactic function, Combine, is binary. Of course, it is possible to make the semantic functions binary as well, through currying, and then Syntax and Semantics are parallel. If one is interpreting structures, and structures are binary, and Semantics is completely compositional, then currying is necessary. However, we are not interpreting structures here; rather, we are building structures with Combine in parallel (bounded by the SSC) with semantic functions

[12] The grammaticality of (42) must be attributed to a “late” purely syntactic reordering function that does not engage the semantic functions in question. The only defense for taking such a dodge is that it is a familiar one.

that are building meaning. So in the present scheme currying is not a necessity. It might simply be an artifact of interpreting *structures*, which we have given up. In any case, the binarity of Syntax remains a mysterious thing, and I have no insight into it.

In this section we will be particularly interested in the disposition of clausal complements to polyadic operators, and the role of the SSC in it. The operators in question are *more, so, too, etc*., and the clauses are the *than*-clause, the *that*-clause complement to *so*, the infinitival complement to *too*, etc; relative clauses will be treated as well. Classically, the relation of the clause to its associated operator was understood in terms of a syntactic operation of extraposition. But, in the model assumed here (which stems from Williams (1998) and (2003)), extraposition is not a possible syntactic operation, because, simply put, clauses cannot move. The consequence of that for semantics is that clauses must be intepreted in-situ, and that causes problems for compositionality using the usual curried semantic definitions designed for interpreting binary structures. Because of the importance of the syntactic assumption here, I will review the syntactic evidence relevant to clausal movement in 3.1. The consequences for semantics are explored in section 3.2. Section 3.2 can stand alone, and you can skip straight to it, but I think that section 3.1 makes the issue all the more pressing.

3.1 Do Clauses Move?

There are both theoretical reasons and empirical reasons for supposing that clauses do not move. I will summarize the arguments here, but the reader is referred to Williams (2003) for full discussion.

First, the theoretical reason. Clauses don't move because they are embedded under a different regime from that under which DPs are embedded. DPs are embedded before they are fully formed, and grow in the course of the derivation of the sentence in which they occur, as suggested in section 1 of this essay. In 2003 I called this “Co-generation”. Clauses, on the other hand, are embedded all at once, exactly when they are built up to the size that the verb to which they are complement to requires. At that point, they are *removed from the Workspace as top-level entities.* I called this regime “Level Embedding (LE)”. The point of LE was to derive some general consequences that have to do with Locality, Improper Movement, Remnant Movement, and Reconstruction. These things follow from LE under the assumption that only members of the workspace are eligible for operations. All clauses are derived in unison, in the same Workspace; and at any given time they are the same “size”, as size is determined by how far the workspace has advanced in Functional terms. This is summarized as:

(1) Level Embedding derivation:

A. F-structure is a "clock" F_0....F_n that only sweeps once; F_i things happen at F_i time; e.g. Wh movement happens at F_{wh}.

B. Clauses come in different sizes, (F_0P, F^nP, and everything in-between)

C. A clause is embedded at the point in the derivation that it is the right size (LE) and the clause is removed from the Workspace.

The following is a derivation of a biclausal sentence under the regime in (1):

(2) a. F_0: **{** John, say, he, laugh **}** initial workspace
b. $F_{i.}$: **{** [John say] **,** [he laugh] **}** form both clauses
c. $F_{T.}$: **{** [John said]$_{TP}$ **,** [he laughed]$_{TP}$ **}** ...grow them into TPs
d. F_C: **{** [John said]$_{CP}$ **,** [that [he laughed]$_{TP}$]$_{CP}$ **}** => ...grow them into CPs
e. F_C: **{** [John said [that [he laughed]$_{TP}$]$_{CP}$]$_{CP}$ **}** ...embed

Once a clause is embedded it is removed from the Workspace, as in (e), and is no longer available as the argument of further operations (semantic or morphosyntactic).

In Williams (2003) I showed that this regime has the following consequences:

(3) 1. Generalized "Improper Movement constraint": X cannot move to F_i in the matrix from F_j in the embedded clause unless i>j.
2. Generalized Reconstruction Relation: X reconstructs for Y if X targets F_i and Y targets F_j and i>j. "X reconstructs for Y" in the model is "Y occurs before X"
3. Generalized Remnant Movement Constraint: if X targeting F_i creates a remnant moved by Y targeting F_j, then i>j.

Recall that an NP is embedded early, and is *not* removed from the Workspace, and so remains available for further operations, such as movement. So clauses do not move, because when they are embedded, they are removed from the Workspace. See Williams (2003), (2010), (2010a) for an implementation of movement in these terms, and an account of islandhood. These constitute the theoretical side of the argument that clauses do not move.

Now we turn to the empirical side of the claim. There are several constructions in which it has been supposed that clauses do move: Topicalization, Passive, Clausal pied-piping, Subject extraposition, and other extrapositions. I will dispense quickly with all but the last of these, as it is in the account of “other extrapositions” that the SSC does some interesting work.

Clausal Topicalization?

It has been known since Koster (1978) (see also Rizzi (1997)) that clauses are not topicalized in the same way that NPs are, as the following suggest:

(4) a. *That John is here I was not aware.
b. That John is here I was not aware of.
c. *I was not aware of that John is here.
d. [That John is here]$_{Topic}$ [NP [I was not aware of t]].

It appears that the supposedly topicalized clause can only originate in an NP position, but the clause cannot actually appear in that position. For this and other reasons, Koster suggested that the clause was base-generated in the Topic position, and the relation between the clause and some NP position was mediated by (covert) Wh movement from that NP position to the front of the matrix clause, as indicated in (d). (a) is ungrammatical because there is no Wh word for a purely clausal position. If this is correct, then clauses can be Topics, but they cannot move.

Likewise, raising and passive do not move clauses:

(5) a. That John was here was not spoken of t
b. That John is here seems [t to be known]
c. That John was here [NP [t was not spoken of t]]

As before, the raised or passivized clauses appear to come from pure NP positions in which they could not occur on the surface. We assume as before that the clausal Topic is generated in-situ; the oly difference between (4) and (5) is that the relation between the clausal Topic and the NP position is mediated by a chain of Wh- and NP- movements.

Clausal Pied-piping?

A more challenging case is clausal pied-piping. Here, the movement seems to be a side-effect of Wh-movement. But again, it shows the same signs of 'base generation' as does clausal topicalization. In fact, there are reasons to think that clausal topicalization underpins clausal pied piping:

(6) a. *That is a party to be invited to which I would be very happy
b. That is a party to be invited to which I would be very happy about
c. *To be invited to which party are you very happy (about)?

We see again that the supposedly moved clause comes from a pure NP position, and cannot come from a pure clause position. This suggests that the failure of clauses to move does not reduce to the absence of a Wh-word for clauses, because in fact the Wh word in a pied-piped clause is nominal.

We not only can treat clausal pied-piping in the same way that we treated clausal topicalization; it is argued in Williams (2003), (2010) that relativization targets a position different from question formation, and that the position targeted is in fact the Topic position[13]; so in fact the very mechanism that works for clausal topicalization works for clausal pied-piping: the “pied-piped” clause is base-generated, and Wh-movement of an (invisible) NP mediates the relation of the clause to the trace position. This conclusion is bolstered by the fact that clausal pied-piping is impossible in questions in both English ((c) above) and German[14], as the topicalization mechanism does not underlie the movement relation in questions.

[13] Schachter (1973) shows that relativization and Focusing, as in "It was John who left", have a common syntax (islands, etc.,). He does not show that relativization and Topicalization do not share a common syntax, or are less closely related relativization and Focus, and so in fact his findings are compatible with the conclusion drawn here.

[14] See van Riemsdijk (1985).

So, the most obvious cases of clausal movement have been put aside. What remains is the case of the extraposition of clausal complements to quantifiers like *more*, *so*, *etc.*

3.2. Q-Complement Extrapositions explained as Combine (Clause, Scope)

I will treat here the clausal complements to degree operators like *more, too, enough, so, and such,* which are simply arguments of those operators. At the end, there will be some remarks about assimilating relative clauses to the picture that emerges. The goal will be to show that the positioning of those clausal complements is finely in accord with the SSC. I will look at how operators that take several arguments, as many as 4, deploy their arguments, and I will argue that the findings strongly discourage the idea that "currying" is the right means of reconciling the binarity of syntax with the polyadicity of semantic functions.

3.2.1 The SSC and Extraposition

It was shown in Williams (1974) that such degree clauses "optionally extrapose" to the end of the scope of the operator they are complement of; thus we get the following:

(7) a. &John has to win more races **than I do** in order to get a prize
b. ~&John has to win more races in order to get a prize **than I do**

The conclusion I came to in 1974 was correct for (b)--the *than*-clause is at the end of the scope of *more*, and includes both "has to" and the "in order to" clause; but I failed to understand the significance of (a), which I attributed to the optionality of extraposition. I think now a more principled analysis is available under the SSC.

To begin with, *more* takes 3 arguments:[15]

(8) more (gradable predicate, Scope, than-clause)

It is slightly more complicated than an ordinary quantifier; if we understand the graded predicate as the Restrictor, then it takes a R(estrictor) and a S(cope) argument and then one more, the *than*-clause.

We could of course give *more* a "curried" definition, one that allowed *more* to take its arguments one at a time, giving a binary semantic structure, but why bother? Furthermore, a curried definition would make impossible the analysis that is to follow.

Now, the Morphosyntax of *more* must be a suite of Combine operations, at most one for each argument. The following addresses the cases in (7):

(9) Morphosyntax:

[15] In this section I am treating *more* as a D-quantifiier, whereas a more plausible analysis of "more races" would be "more many races", where *more* is parallel to *so,* a degree operator. In the next section I argue that degree operators take a fourth argument, and "races" is that fourth argument in "more many races", and so the argument made here is preserved: the than-clause can only attach to arguments of *more.*

Combine (*more*, R)
Combine (*than*-clause, R) ***or*** Combine (*than*-clause, S)

There are two possible alternatives for Combine (than-clause, X); under SSC, X=R or X=S, no other choice is allowed. The two possibilities are illustrated here:

(10) a. More people than we thought were arrested
b. More people were arrested than we thought.

Now, we may understand the examples in (7) to arise from these possibilities. (7a) arises in two ways, corresponding to the two options for the *than*-clause. If the *than*-clause is attached to the Scope argument, then the scope is small, consisting of “win more races”, and so *more* is understood as subordinate to “have to”. But (7a) can arise from the other option, in which the *than*-clause is attached to its restrictor, “races”, in which case the scope is not marked, and can be construed to be the entire sentence. That is why (7a) is ambiguous. (7b) on the other hand is unambiguous—the *than*-clause cannot be understood as attached to the Restrictor, and so must be understood as attached to its Scope, and that Scope must therefore be construed as containing the “in order to” clause and “have to”, and so we get only the wide-scope interpretation for *more.*

So we have gotten the appearance of extraposition, even the appearance of optional exptraposition, without extraposition. Optionality is present, in the choice of X for Combine (than-clause, X), but that falls within the narrow limits of the SSC.

In fact, we can empirically distinguish the theory based on the SSC from a theory based on syntactic extraposition. If extraposition is a syntactic rule it should be governed by some kind of locality. That would suggest that if a given target of extraposition is impossible, a more distant target should also be impossible. But that is disconfirmed by the following facts:

(11) a. *More people's health **than I expected** was compromised – no derivation
b. More people's health was compromised **than I expected** --Combine(than, S)
c. *More people's **than I expected** health was compromised --Combine(than, R)
d. The health of more people **than I expected** was compromised --Combine(than, R)
e. ?More people than I expected’s health was compromised.—also Combine(than, R)

Since a than-clause can appear at the end of an NP (cf. (10), and (11d)), (11a) is surprising from the point of view of the extraposition theory, especially since (b) shows that a more distant target is grammatical. So a theory with extraposition cannot explain the (a/b) difference.

The Combine theory with the SSC though explains everything except (c). (a) is ungrammatical because the *than-clause* is attached to “more people's health”, which is neither the S nor the R argument of *more.* (b) is grammatical because the *than-clause* is attached to the S argument of *more.* (d) is grammatical because the *than-clause* is attached to “people”, which is the R argument of *more*. (c) remains unexplained—here, the *than-*

clause is attached to the R argument, but something still not understood about the complexity of pre-nominal modifiers rules it out independently, and so it is not relevant to the theory comparison at hand; see also (e).

The difference between (10) and (11) show that there is a correspondence between where the (supposedly) extraposed clause goes and the arguments of the quantifier which the clause is itself an argument of; that in turns strongly suggests that there is no rule of purely syntactic extraposition. But this poses a problem for "curried" definitions of quantifiers. For example, it is not possible to give a curried definition for *more* that is adequate for both of the cases in (10), which, schematically, have the following structure:

(12) a. [[[more restrictor] than-clause] scope]
b. [[[more restrictor] scope] than-clause]

We would need two different definitions of *more*, with the following types:

(13) a. <restrictor <than-clause <scope t>>>
b. <restrictor <scope <than-clause t>>>

If the syntax were slightly more complicated, even these alternative definitions would not allow a binary curried semantics to interpret a surface structure; suppose that *more* attached morphologically to the Restrictor in the surface form (as it in fact does in English); then a more abstract syntax would be needed, one in which *more* appeared in a semantically relevant position. But if we have alternative curried binary definitions, QR, *and* abstract syntax, we must raise again the question of whether this richer system would rule out examples that we have supposed to be ruled out for principled reasons in the SSC theory, say (11a). Only a fully specified theory can be evaluated with respect to this question.

3.2.2 Degree operators take a fourth argument

If we look at other operators (*so, too, enough)*, we find evidence for a fourth argument, and further evidence for the SSC.

(14) a. Your car is too big to put in our canoe
b. #Your car is too big **a car** to put in our canoe.

(a) may express an obvious truth, but it is not anomalous in the way that (b) is. (b) suggests that your car is too big *as a car* to be put in our canoe, and that there are cars which *as cars* would not be too big to put in our canoe. (a) does not have this implication. It seems that "a car" in (b) is forcing the context-set to be the set of cars, whereas in (a), the context set is implicit, and can be understood to be something weaker, like "thing". Clearly, *so* takes a further argument, the context-set argument. Note that the gradable-predicate must be contained inside of the context-set, which itself must be contained inside of the scope. These containment relations seem to be necessary for these operators, perhaps part of a generalized prohibition against vacuous quantification.

The grammar of *too* (and, we will assume, of all the operators under discussion) will then be the following:

(15) too:

Semantics:

too (gradable-predicate, Scope, context-set, infinitive-clause)
The only obligatorily explicit argument is gradable-predicate.

Morphosyntax:

Combine (*too*, gradable-predicate)
Combine (gradable-predicate, context-set)
Combine (infinitive, scope)

It should not be surprising that these operators take a fourth argument, the context-set argument. Kennedy (2007), for example, has argued that the "positive" form of the adjective, which in Kennedy's terms consists of the adjective with a null "pos" determiner "[*pos* A]" is subject to truth-condition-altering effects that depend on the modified noun, or on the presence of a "for a NP" expression:

(16) a. Kyle's car is expensive for a Honda (Kennedy ex. 7a)
b. Kyle's car is an expensive Honda (Kennedy ex. 8a)

He suggests in fact that the context-set (or, in his terminology, the *comparison class*), is figured into the meaning of the positive form compositionally, and in fact that the comparison class is an argument (k below) of the *pos* operator:

(17) a. |[pos]| = \g\k: k in $D_{<et>}$. \x. g(x) > norm(k,g) (Kennedy ex. 10)

Given Kennedy's finding about *pos*, it would be surprising if the alternatives to *pos*, namely *more*, *so*, *too*, *enough*, and *how*[16], did not have the same set of arguments. The cases cited in (14) clearly show that *too* has the context-set argument, and I will assume that the rest do as well, although for some the effect is more subtle. Bresnan (1973) has noted the following difference:[17]

(18) a. a woman taller than my father
b. #a taller woman than my father

Why the difference? Suppose, as Kennedy (at least for *pos*) and I both do, that a pre-nominal [deg A] expression will always take the modified N to constrain[18] the context-set; then (18b) anomalously puts my father in the context-set defined by "woman". (a) on the other hand, not being prenominal, can have an implicit context-set, which can be taken to be defined by any predicate into which both comparees can be sensibly included, such as "person" or "thing".

[16] *How* not only takes a context set ("how big a fool"), but also a complement clause ("how foolish is he that he can't see how they manipulate him").

[17] Brought to my attention by Michelle Sheehan

[18] See Kamp and Partee (1995)

So, I am going to assume that all of the operators under discussion in this section have the semantics that *too* has in (14), in the sense that they have the same number and same kind of arguments. This includes *more* in “more races”, which now is analyzed as “more many races”, with *many* as the Restrictor of more, and “races” as the context-set.

In what follows, I want to systematically look at the syntax of two of these operators, *so* and *more,* with an eye to determining how much of that syntax can be explained by the SSC, and what cannot be.

As the empirical base, I will consider 16 examples, systematically varying 4 relevant binary distinctions. The first distinction is between *so* and *more;* they do not behave identically, even though their semantics is essentially the same, or at least is based on the same number and type of arguments. The second distinction has to do with the scope of the operator, which can be either internal to the NP containing the [deg A] or external to it. A third distinction is between whether the clausal argument is in extraposition or in-situ. The final distinction is between whether the [deg A] is pre-nominal or post-nominal. This gives 16 different examples:

(19) so/too **X** wide-scope/narrow-scope **X** extraposition/in-situ **X** pre/post-nominal

In order to make the extrapositions evident, all examples will have the [[deg A] N] constituent as subject.

The content of the clausal argument is manipulated in order to control the scope. This is quite easy to do for the *that-clause* that goes with *so;* it is a subtler matter for *more,* as the discussion which follows will show.

Below are the sixteen examples. I will explain the status of the examples (good, bad, ambiguous, etc.,) first, and then assess to what extent they conform to SSC. Finally, I will highlight features of the paradigm that remain unexplained.

More:
than S = “than we had expected” (favors wide scope)
than NP = “than Bill” (favors narrow scope)

(20) in-situ:
a. A **wealthier** man **than S** joined the company
b. A man **wealthier than S** joined the company
c. A **wealthier** man **than NP** joined the company
d. A man **wealthier than NP** joined the company

(21) Extraposed:
a. A **wealthier** man joined the company **than S**
b. *A man **wealthier** joined the company **than S**
c. ?A **wealthier** man joined the company **than NP**
d *A man **wealthier** joined the company **than NP**

So:

that Sw = that our average net worth went up 50% (favors wide scope)
that Sn = that he doesn't pay any taxes (favors narrow scope)

(22) in situ:
a. ?So **wealthy** a man **that Sw** joined the company
b. ?A man so **wealthy that Sw** joined the company
c. ?So **wealthy** a man **that Sn** joined the company
d. A man so **wealthy that Sn** joined the company

(23) Extraposed:
a. So **wealthy** a man joined the company **that Sw**
b. A man so **wealthy** joined our company **that Sw**
c. *So **wealthy** a man joined the company **that Sn X**
d. *A man so **wealthy** joined our company **that Sn X**

A note on the clausal arguments. For *more:* “than we had expected” (on the right reading) forces wide-scope for *more*, as the antecedent for the ellipsis must be included in the scope. “than NP” will normally take narrow scope (but (21c) will require special comment). Similarly, for *so: “that he doesn't pay any taxes”* is compatible with *so* taking scope just over “wealthy”; that is, “so wealthy that he doesn't pay any taxes” is a “closed” modifier that doesn't depend on whether he joined the company or not. On the other hand, “that our average net worth went up 50%” *does* depend on whether he joined the company or not, and so favors the wide scope reading.

A remark on the difference between (20b) and (20d) is in order. I will claim below that *than S* is attached to *wealthier* in (20b) because it is the Scope argument, but that the *than NP* is attached to *wealthier* in (20d) because it is the Restrictor argument. The reader may rightfully wonder how the same word in the same position could be both Restrictor and Scope. There is no special problem here. The Restrictor is always contained in the Scope anyway. In the case of small clauses, which I will take APs to be, the difference between the Scope and the Restrictor might be phonologically null. It doesn't have to be; in ”probably wealthier than NP”, *wealthier* is the Restrictor, and (part of) “probably wealthier” is the Scope[19]. We could call “wealthier” the AP core of the small clause, and “probably wealthier” the clausal projection of that core. But in the simplest cases, they are not distinguished by phonologically present material.

Now we may survey the facts to see how the SSC fares. For each case the part of the derivation(s) concerning the clausal argument is given, if there is one, and conformity to the SSC will be noted.

(24)
[20] a. Combine (than-S, context-set) possibility of wide scope predicted
b. Combine(than-S, Restrictor) possibility of wide scope predicted

[19] *more* does not out-scope *probably,* but that is true generally, not just in small clauses

	c.	Combine(than NP, context-set)	possibility of narrow scope predicted
	d.	1. Combine(than NP, scope)	
		2. Combine(than NP, restrictor)	both derivations compatible with narrow scope.
[21]	a.	Combine(than-S, scope)	obligatory wide scope predicted
	b.	Combine(than-S, scope)	ungrammaticality **not predicted**
	c.	No derivation	posibility of *than* NP **not predicted**
	d.	No derivation	ungrammaticality predicted.
[22]	a.	Combine(that-S, context-set)	possibility of wide scope predicted
	b.	Combine(that-S, restrictor)	predicts wide scope possibility; oddness **not predicted**
	c.	Combine(that-S, context-set)	narrow-scope possibility predicted
	d.	1. Combine(that-S scope)	
		2. Combine(that-S, restrictor)	both derivations compatible with narrow scope
[23]	a.	Combine(that-S, scope)	necessity of wide scope predicted
	b.	Combine(that-S, scope)	necessity fo wide scope predicted
	c.	No derivation	ungrammaticality predicted
	d.	No derivation	ungrammaticality predicted

Of the three failures of prediction, 2 are cases where Combine(clause, X) over-generates a possibility (21b, 22b); only one is a case where an apparent reading is not generated (21c).

The under-generation is perhaps the acute problem, as it directly challenges the SSC, so I will comment on it first. In (21c) a narrow-scope *more* has an extraposed *than-NP*, exactly what the SSC would forbid, as the entire clause is not an argument of *more* when *more* is narrow in scope. I think in fact that what has happened here is that the scope has been insufficiently controlled in the examples, and that there is, after all, a possibility of a wide-scope reading for a *than* NP. Consider the following:

(25) a. A better candidate than Bill has emerged
b. A better candidate has emerged than Bill

(26) a. A better candidate than Bill is in the lead
b. *A better candidate is in the lead than Bill

In all cases, Bill is understood to be a candidate, because of the obligatory context-set argument. In (25) there is a subtle distinction between the two cases—in (a), Bill might not have "emerged" yet, but in (b), Bill has emerged as a candidate. This is perhaps because "than Bill" truly does have wide scope, including the matrix clause. (26) confirms this conclusion. If we assume that only one candidate can be in the lead, we can explain why extraposition is ungrammatical here: it entails that Bill is in the lead as well, which is impossible.

The conclusion is that "than NP" does take wide scope after all, as some further manipulation of the scope possibilities has shown, and so (21c) should not count against the SSC.

Turning to the under-generation cases, it is useful to note that *more* and *so* differ here. Consider (21b) and (23b) directly contrasted:

(27) [21b]. *A man **wealthier** joined the company **than S**
[23b] A man so **wealthy** joined our company **that Sw**

One might think at first that it had something to do with the appearance of postnominal AP without a complement. Adding a complement to the *more* case does seem to improve things:

(28) ?A man prouder of his country came to the party than we had expected.

But if that is the correct account of (21b), then (23b) becomes mysterious. But perhaps it has nothing to do with extraposition at all; the operator "that", which takes no clausal argument, can appear in a complement-less post-nominal modifier as well:

(29) A man that wealthy should be taxed.

The remaining over-generation case is (22b). I don't understand what is at play here, but it is instructive to compare the whole of (22) with the corresponding paradigm with *more,* (21). Both paradigms conform to the SSC, but *more* goes further—it exemplifies every possibility that the SSC allows. A simple way to distinguish *so* from *more* would be to say that *so* strongly favors Combine(that-clause, scope) over the other possibilities. I don't know why this should be so.

The pattern with *so* seems to be replicated with other of "Deg A a N" operators:

(30) (wide scope favored)
a. Too big a gash was carved into the tree for it to heal
b. ?Too big a gash for {it, the tree} to heal was carved into {the tree, it}
c. ??A gash too big for {it, the tree} to heal was carved into {the tree, it}

(narrow scope favored)
d. ?Too big a boat to clean in one day sank in the harbor
e. A boat too big to clean in one day sank in the harbor
f. *Too big a boat sank in the harbor to clean in one day
g. *A boat too big sank in the harbor to be cleaned in one day

As with *so*, pre-nominal positioning in the [deg A a N] context seems to favor wide scope, and *too* seems to favor Com bine(clause, scope) over other possibilities.

In conclusion, all of the facts considered are in conformity with the SSC, as we have argued away the one case of under-generation as based on a misapprehension that *than NP* could not have wide scope. Closely similar quantifiers differ in what options afforded by the SSC are exploited, and those differences are unexplained.

3.2.4 A bit more on the syntax of the degree operators

I have pretended in this section that *more* and *so* have the same syntax, which they clearly don't ("so obvious a mistake" vs. *"more obvious a mistake"), and so a word is owed about how to account for the difference. I think the difference is small, and involves an extra Combine operation for *so* that *more* does not have. At the relevant point in the workspace for either we will have:

(31) { [a j mistake], obvious$_j$, more/so }

Note that the AP is embedded in the NP by a pointer, as it must remain in the workspace to be subject to further operations. For both *more* and *so* we have Combine(so/more, obvious), where obvious is the restrictor. But for *so*, there is a further Combine operation, Combine(restrictor, context-set), where the context-set is "a j mistake"; this operation is allowed, since both the restrictor and the context-set are arguments of *so*; applying the operation will give the following workspace:

(32) { [j [a j mistake]], [so obvious], }

which, by rules discussed earlier, will be pronounced "so obvious a mistake".

The sort of representation in (32) highlights another feature of the theory which one anonymous reader found hard to swallow—in the following, "a car" serves as both an argument of *bought* (the direct object) and as an argument of *too* (the context-set argument):

(33) John bought too expensive a car to park on the street

In the derivation, "an expensive car" enters the derivation in the following way:

(34) { [an j car]$_k$, expensive$_j$, [bought k] }

Here, "an expensive car" is registered as argument of *bought* through pointers. Since these elements remain in the workspace, they are eligible to be arguments to later syntactic and semantic functions, This must of course be made compatible with the theta criterion, which says than an NP can be the argument of at most one verb.

3.2.5 Relative Clauses

Finally, a word about how relative clauses fit into this picture. The following is suggestive:

(35) a. *Anyone's health that was in Chernobyl was compromised.
b. Anyone's health was compromised that was in Chernobyl

This replicates the pattern in (11), and suggests that the relative clause is attaching to the scope of something, and the likely candidate is *any.* This suggests that a relative clause can serve as a third argument to a quantifier like *any.*

Furthermore, the following reproduces the pattern found with “have to..in order to” and “more” in (7):

(36) a. *Everyone has to leave in order for peace to be restored who was in Chernobyl
b. Everyone has to leave who was in Chernobyl in order for peace to be restored

Given the “in order to” clause, the sensible reading is the one where “has to” has scope over *every;* but that suggests that the relative, which is related to *every,* should occur inside of the “in order to” clause; hence the oddness of (a).

The above examples show the parallelism between relatives and the complement clauses of *more*, *too*, etc., But the usual view of relatives is that they are adjuncts to NP, not arguments to Det. But it may be that there are relative clauses of two sorts, D-related relatives and NP adjunct relatives. Carlson (1977) has proposed that “amount relatives” were D-related, and brought out their relation to comparative clauses. On the other hand, there are predicative NPs lacking D which allow relative clauses, and these must be adjunct to NP. A question that arises from the findings above is whether these non-D-related relatives can appear in exptraposition, given that they are not arguments of operators with scope:

(37) a. ??John became a cook in that period who cannot boil water. (NP-adjunct relative)
b. I met many cooks in that period who cannot boil water.[20] (D-related relative)

I think there is a difference. The striking parallels between relative clauses and the clausal complements to degree operators in (35) and (36) suggest many further questions, but they will not be taken up here.

Conclusion

In this essay I have reviewed some of the variability in the syntactic expression of semantic functions, quantifiers in particular. There is variability--within and across languages, and sometimes between different instances of the same quantifier in the same language-- in what a quantifier marks--scope, restrictor, or both. For an A-quantifier, there is variability in whether it marks focus or scope. There is variability in the positioning of degree operator complements--sometimes adjoined to the restrictor, sometimes adjoined to the scope. There is variability in the marking of the scope of Wh operators; sometimes the scope is not marked; sometimes the Wh+restrictor is prefixed to the scope (i.e. Wh movement) sometimes not; sometimes the scope is marked with a fixed morpheme, and in some languages (recalling Hindi) sometimes there is a choice between two of these.

I have put forward the binary Combine operation with its parameters as a story of this variation. Given a semantic function, there is variation in what Combine operations are

[20] (b) would not count as an amount relative for Carlson (1977), as *many* is ungrammatical in “_ men there were on the life-raft died” (p. 521 ex. 6). I would not accept this as a diagnostic of D-quantifiers but must leave discussion of the point for another time.

enabled, but the SSC requires that for a given function, nothing but that function and its arguments can be arguments of Combine. The other source of variation is in the Combine parameters. No recourse is made to QR, LF, extraposition, or an interpretive morphology in accounting for the variation.

I have suggested that Syntax and Semantics operate "independently", and that both are generative, in that semantics is not a direct function of syntactic structure. Obviously they are not completely independent, otherwise the system would not achieve the minimum goal of connecting sound and meaning. But I have suggested that the connection is as light as it can be, consistent with that minimum goal and certain known findings. I have suggested that each semantic function is paired with a suite of binary operations that generate the syntactic expression of that function applied to its arguments, and that the only general bound on that pairing is the SSC. As a consequence, the meaning of a function applied to its arguments cannot depend on the surface details of the expression of that function or its arguments. In this, I think the model is a principled challenge to theories in which structure is interpreted, in which those details are available by default.

Bibliography

Carlson, G. 1977 “Amount Relatives” Language 53.3 520-542.

Cinque, G. 1999 *Adverbs and Functional Heads: A Cross-Linguistic Perspective*.
Oxford: Oxford University Press.

Cooper, R. 1983 *Quantification and Syntactic Theory* Reidel, Dordrecht.

Evans, N 1995 “A-Quantifiers and Scope in Mayali” in Bach, Emmon, Eloise Jelinek, Angelika Kratzer, and Barbara H. Partee, eds, *Quantification in Natural Languages*, Kluwer

Godard, D. 2004 “French negative dependency” in F. Corblin and H. de Swart (eds). *Handbook of French semantics* Stanford, CA: CSLI Publications, 351–390.

Heim, I. 1982 *The Semantics of Definite and Indefinite Noun Phrases* Umass Dissertation.

Kamp, H. and B. Partee (1995) “Prototype Theory and Compositionality” *Cognition* 57, 127-191.

Keenan, E. and Stavi, J., 1986 “A semantic characterization of natural language determiners,” *Linguistics and Philosophy*, 9: 253–326.

Keenan, E 2003 “The Definiteness Effect: Semantics or Pragmatic?” in *Natural Language Semantics* 11, 187–216.

Kennedy C. 2007 “Vagueness and grammar: the semantics of relative and absolute gradable adjectives” in *Linguistics and Philosophy 30,* 1-45.
Koster, J. 1978 “Why Subject Sentences Don’t Exist.” In *Recent Transformational*

Studies in European Languages, *Linguistic Inquiry Monograph 3,* ed. S. J. Keyser, 53–64. Cambridge, MA: MIT Press.

Malhotra, S. and P. Chandra 2007 “A short note on Wh-scope marking” *University of Maryland Working Papers* Volume 16.

Montague 1973 "The Proper Treatment of Quantification in Ordinary English". In: Jaakko Hintikka, Julius Moravcsik, Patrick Suppes (eds.), *Approaches to Natural Language*. Dordrecht 1973, 221–242.

Partee, B. 1970 “On the Requirement that Transformations Preserve Meaning”
Indiana University Linguistics Club.

Partee, B 1991 “Adverbial Quantification and Event Structures” in *Proceedings of the Seventeenth Annual Meeting of the Berkeley Linguistics Society: General Session and Parasession on The Grammar of Event Structure* 1991, pp. 439-456.

Rizzi, L. 1997 “The Fine Structure of the Left Periphery.” In *Elements of Grammar*, ed. L. Haegeman, 281–337. Dordrecht: Kluwer.

Rooth, M 1992 ”A Theory Of Focus Interpretation.” *Natural Language Semantics* 1:75-116

Schachter, P 1973 “Focus and Relativization” *Language*, 49.1 19-49.

van Riemsdijk 1985 “On pied-piped infinitives in German relative clauses” in Studies In German Grammar, Jindrich Toman, ed., Dordrecht:Foris.

Williams, E. 1974 *Rule Ordering in Syntax* MIT dissertation.

Williams, E. 1998 "Economy as Shape Conservation", MIT Press's "N. Chomsky's 70th Birthday Celebration" Web Page, J. Keyser and K. Hale, eds.

Williams, E. 2003 *Representation Theory* MIT Press.

Williams, E. 2008 "Merge and Mirrors" e-published on Lingbuzz.

Williams, E. 2010 *Regimes of Derivation in Syntax and Morphology,* Routledge.

Williams, E. 2010a “Islands Regained” in *Jan Koster Festschrift*, Jan-Wouter Zwart, ed. Benjamins; also in Williams 2010.