Semantic Typology and Composition (EP-version)[1]

Paul M. Pietroski, Rutgers University

Let's assume that the nouns 'cat' and 'horse' have meanings of the same kind, even though cats and horses are animals of different kinds. Likewise, let's assume that 'above' and 'from' have meanings of the same sort, even though the corresponding relations are importantly different. By contrast, 'horse' and 'from' seem to have meanings that differ in kind. Of course, instances of a category can belong to distinct subcategories. Perhaps 'horse' and 'house' are instances of a common sort that also differ along some semantic dimension. Or perhaps not. The relevant categorical judgments may reflect our concepts, or various beliefs about the things we think and talk about, as opposed to kinds of meanings. As discussed below, there is no guarantee that natural linguistic expressions have meanings that exhibit *any* typological diversity. But let's suppose, tentatively, that expressions of spoken English instantiate some "semantic types."

That leaves all of the more specific questions unanswered. For example, does the meaning of (1) differ in kind from the meaning of (2)?

(1) Socrates (2) die in Athens

Initially, one might think that names have designative meanings that are essentially unlike the predicative meanings of verb phrases. Though on second thought, names can be—and perhaps should be—described as special cases of predicates. A third thought is that (2) might be a predicate of events, raising the question of how verb meanings are related to noun meanings. Such thoughts invite further questions that turn out to be systematically related.

Does the meaning of (3) differ in kind from that of (4)?

(3) every (4) every philosopher

If so, could (1) be an instance of the same semantic type as (4)? What about (5) and (6)?

(5) in Athens (6) see Athens

Do 'in' and 'see' exhibit a common semantic type? Do (6) and (7)?

(7) gave Socrates the cat

Does it matter that 'gave' is tensed? Are (8) and (9) examples of a common semantic kind?

(8) saw Socrates die (9) Socrates die

Is (8) an instance of the same type as (6), (7), both, or neither? What about (9) and (10)?

(10) Socrates died.

Are sentences *sui generis*? How many phrasal meaning types are there? How many lexical meaning types are there? Are some semantic types more basic than others?

One familiar proposal is that natural languages generate expressions of many semantic kinds: entity denoters of a basic type <e>, truth-evaluable sentences of a basic type <**t**>, and expressions of other nonbasic types characterized by the recursive principle (R);

(R) if $<\alpha>$ and $<\beta>$ are types, so is $<\alpha, \beta>$

where expressions of type $<\alpha, \beta>$ signify *functions* from things of the sort signified with expressions of type $<\alpha>$ to things of the sort signified with expressions of type $<\beta>$. On this view, the spoken/signed languages that children regularly acquire—let's call them Slangs—are relevantly like the language that Frege (1879) invented to study the logical foundations of arithmetic. Many insights have been framed in these terms. But as with the study of Slang syntax, old theoretical vocabulary may need to be replaced as insights emerge.

---

[1] For an NYU seminar (2/13/18). The original paper, Pietroski (2018b), had to be shorter. Thanks to the editors, Norbert Hornstein, Valentine Hacquard, Terje Lohndal, Alexander Williams, and many attenders of colloquia.

I think the tripartite proposal is wrong in three ways. In my view, Slangs do not generate expressions of type <e>; they do not generate expressions of type <**t**>; and (R) overgenerates wildly, in a way that is not plausibly constrainable. In the course of defending these claims, I sketch an alternative proposal that is developed in Pietroski (2018a): Slangs generate expressions of *exactly two* semantic types, <M> and <D>, corresponding to monadic representations like 'Horse(_)' and dyadic representations like 'Above(_, _)'; and while many lexical items exhibit the relational type <D>, all complex expressions are of the non-relational type <M>, despite the diversity suggested by (1-10). There are some subcategories, but as we'll see, they reflect minor distinctions that must also be drawn within the Fregean categories <e, **t**> and <e, <e, **t**>>.

1. Some Background and Terminology

For these purposes, let's adopt a broad conception of languages as "things"—generative procedures, sets of expressions, clusters of dispositions, or whatever—that somehow connect certain signals with certain interpretations. Then we can describe Slangs as special cases that connect pronunciations with meanings, leaving it open what these distinctive signals and interpretations are. Put another way, meanings are the interpretations that Slangs connect with certain vocal or manual gestures. The important point will be that Slangs link these meanings to pronunciations in constrained ways that are not happily described in terms of Fregean typology. Though as usual, any talk of meanings calls for comment.

1.1 Unbounded yet Limited

Chomsky (1957, 1964, 1965) offers a helpful starting point. Slangs allow for ambiguity, even bracketing lexical homophony of the sort illustrated with 'bank'. English provides endlessly many examples like 'saw a man with a telescope', which illustrates the phenomenon of structural homophony. This string of words corresponds to more than one phrase, since 'with a telescope' can modify 'man' or 'saw a man'. The corresponding phrasal meanings share a pronunciation despite having different implications. But this kind of ambiguity is constrained. For example, 'eager to eat' connects a certain meaning—roughly, that of 'eager to be an eater'—with a pronunciation that is *not* connected with the meaning of 'eager to be eaten'. In this sense, 'eager to eat' is interestingly *un*ambiguous; it has one meaning as opposed to two. But 'ready to eat' is ambiguous, while 'easy to eat' has only the meaning of 'easily eaten'.

Similarly, even with 'saw' understood as the past tense form of 'see', the relatively simple string (11) is three ways ambiguous, as indicated below.

(11) we saw the spy walking towards the store

(11a) We saw the spy while we were walking towards the store.

(11b) We saw the spy who was walking towards the store.

(11c) We saw the spy walk towards the store.

Note that (11b) can be used to describe a situation in which we saw the *spy* without seeing her *walk*; see Chomsky (1964, p.73). But (12) has only the interpretation corresponding to (11c),

(12) this is the store that we saw the spy walking towards

with the implication that we saw the spy walk. So (12) is interestingly unambiguous. By contrast, (13) is as ambiguous as (11).

(13) this is the store such that we saw the spy walking towards it

In this respect, (12) is unlike (13) and more like (14),

(14) this is the store such that we saw the spy walk towards it

even though (12) is superficially more similar to (13), which retains 'walking' instead of 'walk'.

String (15) is also like (14) and (12) in being unambiguous.

(15) what did we see the spy walking towards

We can use (15), with rising intonation, to ask which thing is such that we saw the spy *walk* towards it—but not which thing is such that we saw the spy while we were walking towards it, or such that we that we saw the spy who was walking towards it. As Chomsky notes, this suggests a common constraint on the meanings of relative clauses and questions. Though one can imagine languages in which (15) is unambiguous in some other way, or as ambiguous as (11).

Examples of this ilk provide a relatively neutral way of indicating what I'm talking about when I talk about meanings: (11) has three meanings, but (12) has only one; 'ready to eat' has two, but 'eager to eat' has only one; etc. In general, the expressions of a Slang connect each pronunciation with *n* but not *n+1* meanings for some *n*. In cases of word salad, like (16), *n* = 0.

(16) *I have been might there

Note that (16) is not understood as a defective variant of 'I might have been there'. However, (17) is understood as a defective variant of 'The child seems to be sleeping'.

(17) *the child seems sleeping.

Indeed, (17) is interestingly unambiguous, since it cannot be understood as a defective variant of 'The child seems sleepy'. So even an ungrammatical string can have one but not two meanings. In other cases, a string of words can be pronounced in a way that supports two but not three meanings, or three but not four, and so on; see Higginbotham (1985), Berwick et. al. (2011).

With this point in mind, let's return to 'saw a man with a telescope'. While this string of words is often used to illustrate structural homophony, it is even more interesting as an illustration of how ambiguity is constrained in Slangs. Note that (18) can be understood in the two ways that are indicated with (18a) and (18b), but not in the third way indicated with (18c).

(18) a boy saw a man with a telescope

(18a) A boy saw a man who had a telescope.

(18b) A boy saw a man by using a telescope.

(18c) #A boy saw a man and had a telescope.

Given (18a), it seems that 'with a telescope' can be understood as a conjunctive modifier, as indicated in (18α). Yet (18) *cannot* be understood as shown in (18γ).

(18α) ∃x[Boy(x) & ∃y[Saw(x, y) & Man(y) & ∃z[With(y, z) & Telescope(z)]]]

(18γ) #∃x[Boy(x) & ∃y[Saw(x, y) & Man(y)] & ∃z[With(x, z) & Telescope(z)]]

So if the words in (18) can be grammatically structured as in (18-G) or as in (18-GG),

(18-G) [[a boy] [saw [a [man [with a telescope]]]]]

(18-GG) [[a boy] [[saw [a man]] [with a telescope]]]

and (18-G) corresponds to (18α), one wants to know why (18-GG) *doesn't* correspond to (18γ).

If the phrase [man [with a telescope]] connects its pronunciation with an interpretation like that of the invented predicate 'Man(y) & ∃z[With(y, z) & Telescope(z)]', then why doesn't [[saw a man] [with a telescope]] connect its pronunciation with an interpretation like that of '∃y[Saw(x, y) & Man(y)] & ∃z[With(x, z) & Telescope(z)]'? We can add event variables, replace (18α) with (18α′), and then accommodate (18b) with (18β′);

(18α′) ∃e∃x[Boy(x) & ∃y[Saw(e, x, y) & Man(y) & ∃z[With(y, z) & Telescope(z)]]]

(18β′) ∃e∃x[Boy(x) & ∃y[Saw(e, x, y) & Man(y) & ∃z[With(e, z) & Telescope(z)]]]

where 'Saw(e, x, y)' is satisfied by an ordered triple <ε_1, ε_2, ε_3> if and only if ε_1 was an event of ε_2 seeing ε_3. This is progress. But one still wants to know why (18) *cannot* be understood as in (18γ′).

(18γ′) #∃e∃x[Boy(x) & ∃y[Saw(e, x, y) & Man(y) & ∃z[With(x, z) & Telescope(z)]]]

More specifically, one wants to know why (18-GG) corresponds to (18β′) *and not* to (18γ′). The fact that string (18) has two meanings, and not three, turns out to be quite interesting.

If ‘saw’ connects its pronunciation with a triadic representation like ‘Saw(e, x, y)’, why can ‘with a telescope’ modify the ‘y’-position as in (18α′) or the ‘e’-position as in (18β′), but not the ‘x’-position (18γ′)? I will return to this question, which reminds us that *absent* readings can be relevant when evaluating hypotheses about lexical meanings and how they determine phrasal meanings given relevant modes of grammatical combination.[2] My suggestion will be that while Slangs tolerate a little dyadicity, they abhor polyadicity. But for now, the important point is that Slangs connect pronunciations with meanings in limited ways. We can invent languages in which string (18) has the interpretation indicated with (18c), or (18) fails to have the interpretation indicated with (18b), or both. In general, given any Slang, there will be many possible languages that connect the same pronunciations with *additional* interpretations. This raises questions about why children don’t acquire such languages, which would exhibit more homophonies than the corresponding Slangs that children do acquire; see, e.g., Crain and Pietroski (2001, 2011). Answers may well include a posited typology for the possible meanings of Slang expressions.

1.2 Discovery and Definition

Let me stress that these are matters for empirical enquiry. Unlike invented languages whose semantic character is stipulated, Slangs are languages that certain primates can naturally acquire. These distinctively human languages have whatever character they have. If language *L* is a Slang that one or more speakers have acquired, theorists don’t get to decree which meanings *L* does or does not connect with any particular pronunciations. Likewise, theorists don’t get to stipulate a typology for the meanings that *L* connects with pronunciations. There is no guarantee that these meanings exhibit any typology, much less one that conforms to the Fregean principle (R).

(R) if <α> and <β> are types, so is <α, β>

I don’t deny that one can introduce a technical notion of Meaning such that by hypothesis, Meanings exhibit the familiar types <e>, <**t**>, <e, **t**>, etc. But then it isn’t obvious—or truistic, or analytic—that Slang expressions have Meanings, given any natural sense for ‘have’. Slangs connect pronunciations with certain interpretations in constrained ways that mirror aspects of the human psychology in virtue of which an ordinary child can acquire a capacity to understand strings like (12) and (18) in certain limited ways.

(12) this is the store that we saw the spy walking towards

(18) a boy saw a man with a telescope

One can hypothesize that these interpretations come in endlessly many flavors that conform to (R). But this is a very bold conjecture, not a platitude. In case the boldness of this conjecture isn’t obvious, let me offer three quick remarks that are elaborated in the pages that follow.

First, constraints on homophony bolster the independently plausible idea that each Slang is an I-language in Chomsky’s (1986, 1995) sense—a biologically implementable *procedure* that generates, in a human way, expressions that are meaningful and pronounceable. From this perspective, questions about semantic typology are questions about the potential outputs of a natural generative system that emerged long ago in some of our hominid ancestors. The issues are not about invented formulae that might be used to “regiment” natural expressions for various purposes. In particular, while it can sometimes be useful to view Slang expressions as analogs of Fregean inventions, natural procedures may not generate instances of Fregean types.[3]

---

[2] Perhaps (18α′) and (18β′) reflect possessive—as opposed to instrumental—uses of ‘with’. But then the question is why (18γ′) is not an available interpretation for (18) given the possessive use of ‘with’.

[3] If a Slang expression of some syntactic type Σ has a meaning of type μ and a pronunciation of type π, then the expression is an instance of all three types; and in my view, its syntactic type (e.g., NounPhrase) is not more essential than its semantic type. Chomsky (1957) invited this perspective on semantics; see also Chomsky (1995).

Second, Frege (1879, 1884, 1892) thought he needed to invent a language that generated expressions of certain "fruitful" types—e.g., <<e <e, **t**>>, <<e <e, **t**>>, **t**>>—that seem to be unattested in the languages that children naturally acquire. In my view, it would be amazing if Frege's focus on the foundations of arithmetic yielded correct descriptions of Slangs, as opposed to a new language whose invented expressions exhibit some scientifically useful types.

Third, Tarski (1944) showed how to provide a truth theory for a first-order predicate calculus whose boundlessly many expressions exhibit *no* semantic typology. The only expressions (a.k.a, well-formed formulae) of such a calculus are *sentences* like (19) and (20).[4]

(19) Nx & Pxy (20) ∀x[Nx ⊃ ∃y[Pxy & Ny]]

Closed sentences like (20) are not special by virtue of having truth values. Tarskian sentences do not differ in semantic type; they all have satisfaction conditions, recursively specified in terms of sequences of domain entities, with truth characterized as satisfaction by all sequences. There is no upper bound on the number of variables that a sentence can have; compare (19) with (21).

(21) Nx & [Pyz & Nw]

So the sentences are not instances of any Fregean type, or types, and neither are the ampersands.

In providing stipulations according to which (20) is true if and only if every natural number is the predecessor of another, one need not assume that (20) has a truth value, much less that closed sentences denote truth values. Church (1941) showed how to *extend* a Tarskian calculus with denoters of types <e> and <**t**>, along with function-denoters like (22) and (23),

(22) λx. ⊤ if Nx, and ⊥ otherwise

(23) λw.λz.λy.λx. ⊤ if Nx & [Pyz & Nw], and ⊥ otherwise

of the types <e, **t**> and <e, <e, <e, <e, **t**>>>>. But while (21) is a constituent of (23), (21) is itself an expression of a Tarskian language that doesn't generate instances of Fregean types.

If the expressions of a language exhibit no semantic categories, then every principle governing the interpretation of complex expressions is syncategorematic. Tarski's treatment of sentences like (19-21), provides a model. I don't think Slangs are Tarskian in this sense. But natural linguistic categories have to be discovered, and they may not include any Fregean types.[5]

1.3 Procedures Matter

Following Chomsky (1957), we can ask which kinds of recursion are characteristic of Slangs. Some bundles of operations—e.g., those that could be used to define any recursively specifiable set of strings assembled from a finite stock of "formatives," or bundles that don't employ any transformational operations—seem quite different than our natural capacity to build linguistic expressions. With regard to semantic typology, we can likewise try to steer between the Scylla of overgeneration and the Charybdis of undergeneration, keeping Frege and Tarski in mind. If we think of Slangs as (child-acquirable) *procedures that generate* expressions in certain ways—rather than *sets* of expressions that might be generated in various ways—we can compare these natural procedures to invented analogs that generate expressions via specific operations. And if

---

[4] To ease reading, I use 'x', 'y', and 'z' rather than 'x', 'x′', and 'x″'. Constants can be added as special variables. But however Tarskian variables are spelled, they are elements of an alphabet, not expressions of a special type.

[5] If we want to find out what meanings *are*, we must be willing to *find out* what meanings are. And facts concerning homophony can be relevant. For example, 'Some odd number precedes every prime number', seems ambiguous in the way indicated with '∃x:Odd(x)[∀y:Prime(y)[Precedes(x, y)]]' and '∀y:Prime(y)[∃x:Odd(x)[Precedes(x, y)]]'. But both readings correspond to the set of all possible worlds, since in every world, the number one precedes every prime number. So one might conclude that identifying sentence meanings with sets of possible worlds—thereby abstracting away from relevant structural distinctions—leads to theories that fail to mark important semantic distinctions. The flip side of this point is that other assumptions about meanings might posit too many distinctions.

certain procedures generate expressions that exhibit types not attested in Slangs, that is some reason for not identifying Slangs with such procedures.

The procedure/set distinction is clear enough, at least since Church (1941). Consider the set of ordered pairs <x, y> such that x is a whole number, and y is the absolute value of x – 1. This set can be described in many ways. Given the notion of absolute value, we can talk about the set of pairs <x, y> such that $y = |x - 1|$. We can also employ the notion of a non-negative square root to talk about $\{\langle x, y\rangle: y = {}^{+}\sqrt{(x^2 - 2x + 1)}\}$. These descriptions correspond to different procedures for computing a value given an argument. A given machine might be able to execute one algorithm but not the other. Church contrasted functions *in intension* with functions *in extension*, and he offered corresponding interpretations of his lambda expressions: on the intensional construal, $\lambda x.|x - 1|$ and $\lambda x.{}^{+}\sqrt{(x^2 - 2x + 1)}$ are distinct but extensionally equivalent procedures; on the extensional construal, $\lambda x.|x - 1|$ is the same set as $\lambda x.{}^{+}\sqrt{(x^2 - 2x + 1)}$. But as Church noted, if the goal is to talk about *computability*, then one wants the intensional construal.[6]

Chomsky (1986) echoed this point, describing I-languages as expression-generating procedures. And since the ordinary word 'language' is polysemous, he used 'E-language' to cover languages of any other sort. So if certain "dispositions to verbal behavior" count as languages, and so do certain sets of formulae, they are E-languages in Chomsky's sense.[7] More importantly, however we count languages, it seems clear that *acquiring* a Slang is a matter of acquiring a biologically implementable procedure that generates expressions in certain ways.

I find it bizarre to say that children acquire sets that have endlessly many elements; cp. Lewis (1975). But *if* children regularly acquire such sets, they presumably do so by acquiring finitely specifiable generative procedures. And in any case, 'English' is not a label for a specific Slang—much less a particular set—if only because of lexical variation across communities, and within communities across individuals. Speakers *of English* have acquired procedures that we can describe as the many English I-languages, which are historically related in ways that are often reflected by similarities across these I-languages. Speakers of English are also like speakers of Norwegian in having acquired *a Germanic language*, much as horses are like cows in being ungulates (and hence mammals). But it's not that Germanic is a language shared by people in London and Oslo. And we can say that English Slangs are closely related Germanic Slangs.

From this perspective, (R) is part of a hypothesis about the expressions generated by Slangs.

(R) if $\langle\alpha\rangle$ and $\langle\beta\rangle$ are types, so is $\langle\alpha, \beta\rangle$

The familiar idea is that Slang expressions exhibit types that correspond to "semantic values" of various kinds: entities, truth values, functions from entities to truth values, etc. I think this hypothesis is false, and that we should adopt a more limited semantic typology.

---

[6] The key proof in Church (1941) employs what he calls "the calculus of λ-δ-conversion." He invents this formal language in stages. But at the outset, he introduces the intensional/extensional contrast and says where he is heading:

> In the calculus of L-conversion and the calculus of restricted λ-*K*-conversion, as developed below, *it is possible, if desired*, to interpret the expressions of the calculus as denoting functions in extension. However, in the calculus of λ-δ-conversion, where the notion of identity of functions is introduced into the system by the symbol δ, *it is necessary, in order to preserve the finitary character of the transformation rules,* so to formulate these rules that an *interpretation by functions in extension becomes impossible*. The expressions which appear in the calculus of λ-δ-conversion are interpretable as denoting functions in intension of an appropriate kind. (p.3, my italics).

Frege (1892) also took the procedural notion of a mapping as basic, for purposes of logic, distinguishing Functions in his sense from their extensionally individuated "courses of values." See Horty (2007) for related discussion.

[7] Chomsky (1986) also suggested that I-languages are individuated *internalistically* and *individualistically*, as opposed *externalistically*; cp. Ludlow (2011). But these connotations of 'I-' are not definitional.

Others can reject this psychologistic perspective on semantic theories, and suggest another way of thinking about the role that (R) plays in accounts of linguistic meaning. Perhaps there is a non-stipulative sense in which Slang expressions are related (e.g., via canonical uses) to semantic properties that exhibit an (R)-ish typology, even if the expressions do not themselves exhibit this typology. My contrary suspicion is that if (R) does not figure in a good description of the natural generative capacity, it has no important place in theories of Slangs, apart from its role in letting us describe a space of conceivable interpretations for symbols. In any case, I want to focus on—and object to—the following idea: Slangs generate expressions of Fregean types, in accord with (R), which is like other principles that linguists formulate in attempts to characterize procedures that children can naturally acquire; see Heim and Kratzer (1998), Jacobson (2014).

Thanks to Frege and the other founders of modern logic, we can imagine a mind that has the following capacities: it can generate expressions of two basic types, <**t**> and <e>, perhaps initially by assigning special labels to the sentences and constants of a Tarskian calculus; it can create expressions of type <e, **t**> by a kind of abstraction on type <e> constituents of type <**t**> expressions; and it can likewise create expressions of the other Fregean types, modulo various performance limitations. I grant that adults, who already have a Slang, can invent and use procedures that generate expressions of types like <<e, **t**>, **t**> and <<e, <e, **t**>>, <<e <e, **t**>>, **t**>>. But my question is whether the Slangs we naturally acquire, as children, are Fregean languages.[8]

2. Unwanted Recursion

Given at least one semantic type, (R) implies that there are boundlessly many.

(R) if <α> and <β> are types, so is <α, β>

Initially, this might seem innocuous, since a grammar for a Slang can harmlessly imply that the language has—i.e., generates—boundlessly many expressions. But while any expression of English can be part of another, even if there are limits on the size of expressions that can actually be constructed, it doesn't follow there are endlessly many *types* of expressions or meanings.

2.1 Rapid Overgeneration

Moreover, given *two* basic types, just a few iterations of (R) yields many, many more.

Consider a basic domain consisting of some entities (e.g., the natural numbers) and two truth values, ⊤ and ⊥. We can view <e> and <**t**> as types that constitute Level Zero of a hierarchy whose next level includes four types: <e, e>; <e, **t**>; <**t**, e>; and <**t**, **t**>; where each of these types corresponds to a class of functions from things of some Level Zero sort to things of some Level Zero sort. Put another way, Level Zero is exhausted by the two basic Fregean types, which can be described as <0> types. Level One is exhausted by the four <0, 0> types. At the next level, there are the new types that can be formed from those at the two lower levels:

eight <0, 1> types, including <e, <e, **t**>> and <**t**, <**t**, e>>;
eight <1, 0> types, including <<e, e>, e>> and <<**t**, **t**>, **t**>; and
sixteen <1, 1> types, including <<e, e>, <e, e>> and <<e, **t**>, <**t**, **t**>>.

So at Level Two, there are thirty-two types, each corresponding to a certain class of functions. At Level Three, there are 1408 new types that can be formed given those at the three lower levels:

---

[8] We can talk about sets that connect pronunciations with "ideal" interpretations that rational creatures could agree on without acquiring human I-languages; see, e.g., Lewis (1975). Imagine the pronunciation of 'Bert' being used to signify Bertrand Russell, perhaps via various definite descriptions and Tarskian constants. If such uses have a shared singular content, the public expression 'Bert' may be of the ideal type <e>. But my question concerns meanings, not ideal communicative contents. If the distinction is denied, we can say that Slangs connect meenings with pronunciations, and that as meeningful expressions are used, they *may* acquire meanings/contents. Then the question is whether meenings exhibit a Fregean typology. For extended discussion, see chapter one of Pietroski (2018a).

sixty-four <0, 2> types, including <e, <e, <e, **t**>>>;
sixty-four <2, 0> types, including <<e, <e, **t**>>, **t**>;
one-hundred-and-twenty-eight <1, 2> types, including <<e, **t**>, <<e, t>, **t**>>;
one-hundred-and-twenty-eight <2, 1> types, including <<e, <e, **t**>>, <e, **t**>>; and
one-thousand-and-twenty-four <2, 2> types, including <<e, <e, **t**>>, <e, <e, **t**>>>.

At Level Four, there are more than two million types: <e, <e, <e, <e, **t**>>> and 5631 more <0, 3> or <3, 0> types; 11,264 <1, 3> or <3, 1> types; 90,112 <2, 3> or <3, 2> types; and 1,982,464 <3, 3> types. (Compare the "iterative conception" of the Zermelo-Frankl sets, as discussed by Boolos 1998.) Let's not worry about Level Five, at which there are more than 5 x 10^{12} types. But it is worth thinking about Levels Three and Four. I'll return to the lower levels in section three.

Let 'e**t**' abbreviate '<e, **t**>', and consider the Level Three type <<e, e**t**>, **t**>. Functions of this type map functions like λy.λx.**Predecessor**(x, y)—i.e., λy.λx. ⊤ if Predecessor(x, y), and otherwise ⊥—onto to truth values.[9] So we can invent Fregean expressions of type <<e, e**t**>, **t**> to encode judgments concerning properties of first-order dyadic relations. For example, we can use (24) and (25) to say that while λy.λx.**Predecessor**(x, y) is not transitive, λy.λx.**Precedes**(x, y) is.

(24) ~**TRANSITIVE**[λy.λx.**Predecessor**(x, y)]
(25) **TRANSITIVE**[λy.λx.**Precedes**(x, y)]

Fregean languages also support abstraction over the dyadic relations. For example, the function λ**D**.**TRANSITIVE**(**D**) maps λy.λx.**Precedes**(x, y) to ⊤, while mapping λy.λx.**Predecessor**(x, y) to ⊥. One can also encode thoughts concerning relations among relations—and in particular, the thought that precedence is the transitive closure or "ancestral" of the predecessor relation—in a logically perspicuous way. Indeed, as Frege (1879, 1884) showed, the real power of his logic is revealed with expressions of the Level Four type <<e, e**t**>, <<e, e**t**>, **t**>> as in (26).

(26) **ANCESTRAL-OF**[λy.λx.**Precedes**(x, y), λy.λx.**Predecessor**(x, y)]

Note that λ**D′**.λ**D**.**ANCESTRAL-OF**(**D**, **D′**) is like λ**D**.**TRANSITIVE**(**D**) in being second-order, and like λy.λx.**Predecessor**(x, y) in being dyadic; compare λ**D**.**ANCESTRAL**(**D**), which maps λy.λx.**Predecessor**(x, y) to λy.λx.**Precedes**(x, y). In providing this kind of symbolism, Frege thought he was offering a *new* way of representing relations among relations. He thought he needed to invent languages that generated sentences, of a basic type <**t**>, whose constituents could include expressions of the abstract type <<e, e**t**>, <<e, e**t**>, **t**>>. One can use a Slang to say that the relation of precedence is the transitive closure of the smallest relation that one number bears to another if and only if the first is the predecessor of the other. In this sense, we can gesture at the content of (26) with naturally generable locutions. But this requires circumlocution in which we talk about relations as things; and prima facie, the nominalizations fail to reflect certain logical relations among the relations and relata. By contrast, we can imagine minds that naturally generate (24-26) and pronounce them as we would pronounce (24a-26a);

(24a) Predecessor doesn't transit.
(25a) Precede transits.
(26a) Precede ancests predecessor.

where 'transit' and 'ancest' would exhibit, respectively, the unSlangy types <<e, e**t**>, **t**> and <<e, e**t**>, <<e, e**t**>, **t**>>.[10]

---

[9] Crucially, **Predecessor**(2, 3) is a truth value, but Predecessor(3) is a number. Even "monadic" functions of type <e**t**> are relational, since they map entities to truth values. But let's ignore Frege's talk of Functions/Concepts being *unsaturated*, and use notation like 'λx.**Prime**(x)' to talk about denotable functions as in Church (1941); see note 6.

[10] Such words would not be verbs—perhaps corresponding to abstract states—that combine with *names* of relations.

Frege showed how a possible mind—one that enjoys a competence partly characterized with (R)—could generate expressions of much higher types.

(R) if <α> and <β> are types, so is <α, β>

Though one can also imagine such a thinker being subject to performance limitations. Perhaps she cannot naturally employ expressions of types above Level Four, given the memory needed to abstract and store such expressions. Such a thinker might, qua theorist, know that she has an I-language that generates expressions of type <<e, <e, e**t**>>, <<e, <e, e**t**>>, **t**>. But this might be akin to our theoretical appreciation of the fact that (27) is a sentence.

(27) The rats the cats the dogs chased chased fight mice mice mice fight fight.

So the mere unavailability of Level Five expressions doesn't show that a thinker doesn't have a competence characterized with (R); and many expressions of lower types like <**t**, <e, <**t**, e>>> might correspond to functions (i.e., intensions) that thinkers find unnatural, even if such expressions are available in principle. But my concern is not merely that endlessly many Fregean types, including the vast majority of "lower" types, are unattested in human languages. My initial concern is that humans can, and with a little help *do*, grasp the thoughts indicated with formalism like (24-26). So why can't we pronounce these thoughts by simply introducing words like 'ancest', and using expressions like (24a-26a), if Slangs permit expressions of Fregean types?

One can always posit limits on how competences get used. So one can still claim that our semantic competence is basically Fregean. Perhaps Slangs allow for (24a-26a), but other aspects of human nature keep us from naturally constructing and using such expressions, even though we can invent and use (24-26). There may be barriers to introducing lexical items of certain Fregean types. But the coherence of this position is not an argument that it is correct. There are independent reasons for thinking that Slangs generate expressions like (27), and plausible suggestions about why this kind of embedding makes expressions hard to parse; see Chomsky (1957, 1965). So one wants to see reasons for thinking that Slangs generate expressions of Level Three and Level Four types, along with accounts of why expressions of types <<e, e**t**>, **t**> and <<e, e**t**>, <<e, e**t**>, **t**>> are unattested, even though we can comprehend (24-26).

Here is another way of indicating the concern; cp Chierchia (1984). Abstraction on the subject or object position of (28), as in (28a) and (28b), is easy. So why isn't (28c) available,

(28) The plate outweighs the knife.

(28a) The plate is something *which outweighs the knife*.

(28b) The knife is something *which the plate outweighs*.

(28c) *Outweighs is something *which the plate the knife*.

with the underlined phrase understood as a relative clause of type <<e, e**t**>, **t**>? (If 'something' or 'which' imposes a type restriction, why can't we have 'somerel whonk the plate the knife'?) And why can't we use (29c) to say that λy.λx.**Precedes**(x, y) is a relation that three bears to four?

(29) Three precedes four.

(29a) Three is something *that precedes four*.

(29b) Four is something *that three precedes*.

(29c) *Precedes is something *that three four*.

2.2 Unrelational Quantifiers

It is often said that words like 'every' and 'most', as in (30) and (31),

(30) Every cat ran quickly.

(31) Most cats ran quickly.

are instances of the Level Three type <e**t**, <e**t**, **t**>>. The idea is that modulo niceties regarding tense and agreement, a determiner combines with an "internal" argument of type <e**t**> and an “external” argument of the same type, much as ‘precedes’ can combine with two arguments of type <e>. On this view, ‘every’ and ‘most’ indicate relations that are exhibited by functions of type <e**t**>. Correlatively, the types <e, <e**t**>> and <e**t**, <e**t**, **t**>> instantiate the abstract pattern <α, <α, **t**>>. But <<e, e**t**>, <<e, e**t**>, **t**>> exhibits this same pattern. So if some human words indicate functions of type <e, e**t**>, and our I-languages are characterized by (R),

(R) if <α> and <β> are types, so is <α, β>

we should be able to generate expressions of type <<e, e**t**>, <<e, e**t**>, **t**>>. Even if *verbs* cannot have meanings of this type, one wants to know why humans can’t naturally use Slangs to form expressions like (32); where ‘Ancestral predecessor’ is a constituent of type <<e, e**t**>, **t**>.

(32) Ancestral predecessor precede.

What precludes (32) if ‘Every cat’ is a constituent of type <e**t**, **t**> in (30)?

(30) Every cat ran quickly.

This bolsters other reasons for suspecting that determiners do not have meanings of type <e**t**, <e**t**, **t**>>. One obvious question is why (33) *fails* to have a sentential reading.

(33) every cat which ran quickly

If the relative clause ‘which ran quickly’ exhibits any Fregean type, it is presumably <e**t**>. But then why can’t (33) be understood as a sentence roughly synonymous with (30)?

One can speculate that for some syntactic reason, ‘every’ cannot take a relative clause as its external argument, and must instead take a smaller clause of the same semantic type. But the issue runs deep. We can specify the meaning of (30) as follows: for every cat, there was an event of it running quickly. And I am happy to say that ‘Every cat’ raises—leaving a trace of displacement—so that the determiner’s external argument is a sentential expression akin to (34).

(34) It ran quickly.

But if ‘every’ takes an external argument that is *sentential*, then one needs an ancillary assumption to maintain that ‘every’ is of type <e**t**, <e**t**, **t**>>.

Heim and Kratzer (1998) are admirably explicit about this. On their view, (30) has the form shown in (30a), with the indexed trace interpreted like the pronoun in (34).

(30a) [[every cat]$_{\langle et,\, t\rangle}$ [1 [t_1 ran quickly]$_{\langle t\rangle}$]$_{\langle et\rangle}$]$_{\langle t\rangle}$

The index is allegedly reproduced in a way that converts a sentential expression of type <**t**> into an expression of type <e**t**>. But the index is not posited as an expression of type <**t**, e**t**>; it is said to be more like ‘which$_1$’ in [which$_1$ [t_1 ran quickly]$_{\langle t\rangle}$]$_{\langle et\rangle}$. Heim and Kratzer thus appeal to a syncategorematic operation of abstraction, corresponding to Tarski-style quantification over ways of assigning values to indices.[11] In my view, such appeal is unavoidable. So my concern is *not* that (30a) has an element that is not an instance of any Fregean type. My concern is that this element is posited as a device that converts the external/sentential argument of ‘every’ into a relative clause—thereby effacing the apparent contrast with the internal/nominal argument—even though quantificational determiners cannot take relative clauses as external arguments.

As noted above, (33) cannot be understood as a sentence that is roughly synonymous with (30). So it seems odd to say that in (30), ‘Every cat’ combines with an expression whose meaning is that of ‘which ran quickly’. Absent the *stipulation* that in (30), the sentential clause [t_1 ran quickly]$_{\langle t\rangle}$ combines with a syncategorematic element to yield an expression of type <e**t**>, why describe ‘every cat’ as an instance type <e**t**, **t**>—i.e., as an expression that can combine

[11] They posit a rule according which: if a sentence S contains a trace with index *i* and combines with a copy of *i*, the result is an expression of type <e**t**>; and relative to any assignment $\mathcal{A}$, *i*^S indicates a function that maps each entity e to ⊤ iff S denotes ⊤ relative to the minimally different assignment $\mathcal{A}$* that assigns e to *i*.

with an expression of the same type as a relative clause? If (33) cannot be understood as a sentence in which ‘every cat’ combines with a relative clause, perhaps that is because ‘every cat’ is not of type <e**t**, **t**>. One can speculate that ‘every’ abhors relative clauses, even though it indicates a dyadic relation exhibited by functions of type <e**t**>, and its external argument gets converted into something that looks like a relative clause. But even if this proposal is coherent, that does not make it plausible. On the contrary, the proposal seems desperate.

Moreover, if words like ‘every’ indicate second-order dyadic *relations*, one wants to know why these relations are invariably *conservative* in Barwise and Cooper’s (1981) sense; see also Higginbotham and May (1981). My suggestion is that the antecedent of this conditional is false. In my view, words like ‘every’ are not instances of the Fregean type <e**t**, <e**t**, **t**>>.

In a domain of sets, a first-order relation **R** of type <e, e**t**> is conservative if and only if $\forall s'\forall s[\mathbf{R}(s', s) \equiv \mathbf{R}(s \cap s', s)]$. For example, *inclusion* is a conservative relation: s′ includes s—i.e., s′ is a (perhaps improper) superset of s—if and only if the intersection of s and s′ includes s. The set {1, 2, 3} includes {1, 2}; and their intersection includes/is {1, 2}. The set {1, 2, 3} doesn’t include {3, 4, 5}; and their intersection, {3}, also fails to include {3, 4, 5}. Likewise, *overlap* is a conservative relation: s′ overlaps s—i.e., s′ intersects with s—if and only if the intersection of s and s′ overlaps s. The set {1, 2, 3} overlaps {3, 4, 5}; their intersection, {3}, also overlaps {3, 4, 5}; but neither {1, 2, 3} nor ∅ overlaps {4, 5, 6}. So we can say that λY.λX.**Overlap***(X, Y) is a conservative function of type <e**t**, <e**t**, **t**>>; where functions of type <e**t**> overlap* if and only if their extensions overlap. Likewise, λY.λX.**Superset***(X, Y) is conservative. But λY.λX.**Subset***(X, Y) is not conservative. While {1, 2, 3} *isn’t* a subset of {3, 4, 5}, their intersection *is* a subset of {3, 4, 5}.

I take it as given that the first word in (35), which can be added to (36) at many points,

(35) Only cats ran.

(36) Felix thinks that Fido admires him.

is not a determiner whose meaning is λY.λX.**Subset***(X, Y); see, e.g., Herburger (2001). But this raises the question of why kids *don’t* acquire a determiner ‘Ryev’ that is the semantic converse of ‘Every’. More generally, for each asymmetric conservative relation, there is a corresponding nonconservative relation.[12] The symmetric relations **Identical**(s′ s) and **Equinumerous**(s′ s) are also nonconservative: {1, 2, 3} isn’t identical, or even equinumerous, with {1, 2}; but the intersection of these sets is identical, and hence equinumerous, with {1, 2}. So if ‘every’ is of the same semantic type as ‘λY.λX.**Superset***(X, Y)’, we need some explanation for the absence of determiners—call them ‘ident’ and ‘equi’—that indicate the functions λY.λX.**Identical***(X, Y) and λY.λX.**Equinumerous***(X, Y).

Putting the point another way, a sentence like (37) can be glossed as in (37a),

(37) Every/Some/No cat ran.

(37a) Every/Some/No cat is one which ran.

with ‘one’ serving as kind of anaphoric restriction. Likewise, all/none/most of the cats ran if and only if all/none/most of the cats are cats which ran. But if there were sentences like (38),

(38) Ryev/Ident/Equi cat ran.

they wouldn’t be paraphrasable with analogs like (38a).

(38a) Ryev/Ident/Equi cat is a cat which ran.

This asymmetry, between *internal/nominal/restrictor* and *external/sentential/scope* arguments of determiners, can be described in many ways. But why should there be any such asymmetry if

[12] Given λY.λX.**Overlap***(X, Y) & **Singleton***(Y) for ‘the’, consider λY.λX.**Overlap***(X, Y) & **Singleton***(X).

determiners are of type <e**t**, <e**t**, **t**>>?[13] Eliminating the grammatical distinction between the internal and external arguments, in order to treat them all as instances of the Fregean type <e**t**>, seems a like bad idea. One can posit a filter on otherwise admissible meanings. But prima facie, this is a manufactured solution to a deeper problem, endemic to the idea that Slang determiners instantiate a second-order relational type. If a determiner displaces from its original position in some sentential expression S, one would expect the determiner to recombine with S itself, as opposed to a more complex expression that is more like a relative clause.

2.3 Sidebar: Abstracting on Abstractions is Math (not Grammar)

The point of the last few paragraphs is not merely that words like ‘every’ are not examples of the Level Three type <e**t**, <e**t**, **t**>>. Similarly, the point about transitivity and ancestrals was not merely that Slangs seem unlike Frege’s *Begriffsschrift*, in that the natural human languages seem to abhor expressions of the Level Four type <<e, e**t**>, <<e, e**t**>, **t**>>. I think that whatever the basic types are, Slangs don’t generate expressions that exhibit types of the form <α, <α, β>>.

Initially, (39) might be described in classical Subject-Predicate terms, as in (39a);

(39) Felix is a cat.

(39a) [Felix$_{\langle Subj\rangle}$ [is cat$_{\langle Pred\rangle}$]$_{\langle Pred\rangle}$]$_{\langle Sent\rangle}$

where ‘a’ is treated as a syncategorematic mark of a singular count noun, and variants of the copular verb ‘be’ are treated as syncategoremata that are used to create tensed predicates. One can add that ‘every’ is another syncategorematic expression that combines with suitable predicates to form subjects, as shown below,

(40) Every cat ran.

(40a) [[every cat$_{\langle Pred\rangle}$]$_{\langle Subj\rangle}$ ran$_{\langle Pred\rangle}$]$_{\langle Sent\rangle}$

while a transitive verb like ‘chased’ is a syncategorematic device that can combine with suitable subjects to form predicates, as in [chased Felix$_{\langle Subject\rangle}$]$_{\langle Predicate\rangle}$]. But instead of positing so much syncategorematicity, one might say that ‘every’ is an instance of <Pred, Subj> as in (40b).

(40b) [[every$_{\langle Pred,\ Subj\rangle}$ cat$_{\langle Pred\rangle}$]$_{\langle Subj\rangle}$ ran$_{\langle Pred\rangle}$]$_{\langle Sent\rangle}$

Similarly, one might say that copulas and transitive verbs instantiate the nonbasic types <Pred, Pred> and <Subj, Pred>. Then (41) might be described as in (41a).

(41) Fido chased every cat

(41a) [Fido$_{\langle Subj\rangle}$ [chased$_{\langle Subj,\ Pred\rangle}$ [every$_{\langle Pred,\ Subj\rangle}$ cat$_{\langle Pred\rangle}$]$_{\langle Subj\rangle}$]$_{\langle Pred\rangle}$]$_{\langle Sent\rangle}$

Though given boundlessly many examples like (42) and (43),

(42) Every cat chased every cat.

(43) Felix chased a dog that every cat chased.

it seems that a transitive verb combines asymmetrically with two arguments. This invites further abstraction, perhaps with ‘chased’ redescribed as an instance of <Subj, <Subj, Sent>> and the category <Subj> analyzed as <Pred, Sent>. This way of describing Slangs characterizes predicates and sentences as expressions of basic semantic types, while the classical category of Subject is redescribed as non-basic. In my view, this hypothesis about Slang typology is wrong, but so are more familiar neo-Fregean hypotheses.[14]

---

[13] One can say that ‘Every cat ran.’ has the structure [[every cat]$_{\langle et,\ t\rangle}$ [~~one [which~~$_1$ [t_1 ran]$_{\langle t\rangle}$]$_{\langle et\rangle}$]$_{\langle et\rangle}$]$_{\langle t\rangle}$. But then it is even harder to explain why ‘every cat which ran’ cannot mean that every cat ran.

[14] One can describe instances of <Pred, Subj> as instances of <Pred, <Pred, Sent>> without taking <Pred> as a basic type; cp. <e**t**, **<**e**t**, **t**>>. And one might describe transitive verbs as instances of the type <<e**t**, **t**>, <<e**t**, **t**>, **t**>>. These steps in the direction of a categorial grammar raise many interesting questions about how to spell out the details; see, e.g., Jacobson (1999), Steedman (1996). They also highlight questions about which types are basic. Frege argued that logicians don’t need the traditional notion of “Subject;” and Chomsky (1957, 1965) argued that

Given the type <Sent>, along with either <Pred> or <Subj>, the recursive principle (R) characterizes a space of types that correspond to possible abstractions.

(R) if <α> and <β> are types, so is <α, β>

Replacing <Sent> and <Subj> with <**t**> and <e> yields a hypothesis about the space of meaning types that Slang expressions can exhibit. But in evaluating any such hypothesis, we need to think about what Slangs *don't* generate, and not just possible ways of describing what Slangs do generate. If 'every' can combine with two predicates to form a sentence, then for some purposes, it does no harm to describe the meaning of 'every' (*a la* Frege) in terms of abstractions from sentence meanings. But positing meanings of type <Pred, <Pred, Sent>> or <e**t**, <e**t**, **t**>> raises the question of why the meanings that are allegedly of this type respect the constraints they do. Similarly, if 'predecessor' and 'precedes' are described as instances of some relational type <τ>, then one wants to know why the type <τ, <τ, **t**>> is *un*attested if <**t**> is a basic type. In short, describing Slangs in terms of (R) turns many absences into puzzles.

2.4 Ubiquitous Overgeneration

The more than two million types at Level Four include <<e, e**t**>, <<e, e**t**>, **t**>>. This type was especially important to Frege, who wanted to talk about how the predecessor relation is related to the more inclusive relation of precedence. But perhaps some natural limitation blocks dyadic abstractions across dyadic relations—like **λD′.λD.ANCESTRAL-OF(D, D′)**—while still allowing for semantic types like <e**t**, <e**t**, **t**>> and <e, e**t**>. Yet even if this ancillary hypothesis is correct, the Level Four types also include <e**t**, <e**t**, <e**t**, t>>>> and <e, <e, <e, <e**t**>>>>. So if these types are unattested in Slangs, one wants to know why.

It isn't hard to imagine "ditransitive" quantificational determiners like 'Glonk' in (44).

(44) Glonk dogs cats are brown.

Such words would combine with three predicates, yielding a meaning like that of (44a) or (44b).

(44a) The brown dogs outnumbered the brown cats.

(44b) There are some brown dogs or some brown cats.

Nor is it hard to imagine a "tritransitive" verb that appears in sentences like (45),

(45) A dog sald a cat a car a dollar.

with the meaning indicated in (45a/b).

(45a) A dog sold a cat a car for a dollar.

(45b) A dog sold a car to a cat for a dollar.

For these purposes, let's not worry about the indefinite descriptions. Suppose that 'sald' would be of type <e, <e, <e, <e**t**>>>> and not the Level Five type <e**t**, <e**t**, <e**t**, <e**t**, **t**>>>>. For now, let's also ignore adverbial modification as in (46) and the need for an event variable.

(46) Yesterday, a dog happily sald a cat a dollar in Boston.

The important point is that we seem to have a concept of selling that differs in adicity from the corresponding concept of giving. A seller gets something back, as part of the exchange. So why don't we have a verb whose type matches that of the tetradric Fregean concept SOLD(X, Y, Z, W)? String (47) is anomalous, not a simpler way of saying that Fido sold a car to Felix for a dollar.

(47) *Fido sold Felix a car a dollar.

But why use the prepositional phrase '*for* a dollar' if verbs can be of type <e, <e, <e, <e**t**>>>>? And note that (48) has the meaning indicated in (48a), with no implication that Fido sold a car.

(48) Felix bought Fido a car for a dollar.

---

linguists should replace the traditional notion with "NP of S," corresponding to a rule like "S ➔ NP aux VP." Though one wants to know why the Subject-Predicate distinction is intuitive if <Sent> is not a *basic* category that corresponds to the "Start" symbol in a phrase structure grammar. For discussion, see Pietroski (2018a).

(48a) Felix bought a car for (the benefit of) Fido (in exchange) for a dollar.

One can speculate that some limitation on the use of Slangs precludes expressions of any semantic types from above Level Three. But this is to admit that given (R),

(R) if <α> and <β> are types, so is <α, β>

one needs to posit further constraints, and not just a "conservativity filter" on meanings of type <e**t**, <e**t**, **t**>>. Recall that λ**D.TRANSITIVE(D)** is of the Level Three type <<e, e**t**>, **t**>. One can say that while <e, e**t**> and <**t**> are attested types, some performance limitation blocks further abstraction. But at this point, the question is why appeal to (R) and the requisite ancillary assumptions is better than appeal to a *list* of attested types. Moreover, a cluster of facts suggest that <e, <e, <e**t**>>>—arguably the simplest Level Three type—is not attested, not even by lexical items that would seem to be prime candidates; see Pietroski (2018a).

I assume that humans, along with many other animals, can have the triadic concept BETWEEN(X, Y, Z). So one might have expected 'between' to indicate such a concept, and appear in sentences like (49), if human words can be of type <e, <e, e**t**>>;

(49) Fido between Garfield Harriet.

where 'between Garfield' would be a constituent of type <e, e**t**>. But instead, we have the circumlocutory (50), as if Slangs abhor lexical analogs of triadic concepts.

(50) Fido is between Garfield and Harriet.

Similarly, we seem to have a triadic concept of jimmying that relates a jimmier to both a thing jimmied and an instrument. Yet we resort to prepositional phrases, as in (51),

(51) A thief jimmied a lock with a screwdriver.

instead of introducing a verb of higher adicity as in (52); see Williams (2005, 2007, 2015).

(52) *A thief jimmied a lock a screwdriver.

Examples like (53) initially suggest that some human words are of type <e, <e, e**t**>>.

(53) A dog gave a cat a car.

But the verbs in ditransitive *constructions* may not be semantically triadic; cp. (54).

(54) A woman kicked a dog a bone.

With regard to meaning, and perhaps with regard to syntax, the "indirect objects" are plausibly more like the prepositional phrases in (53a) and (54a); see Larson (1988) and Baker (1997).

(53a) A dog gave a car to a cat.

(54a) A woman kicked a bone to a dog.

Though for present purposes, it doesn't matter which of the arguments in sentence (53) is not an argument of the *verb* 'give'. Perhaps it will turn out that some verbs really do have Level Three meanings. But the absence of evidence for such verbs is striking.

3. Events and Predicates

One might reply that while 'give' and 'kick' are of the same semantic type, along with 'sell' and 'chase', this type is at Level Three. So given independent motivation for event variables, perhaps many verbs are of type <e, <e, *e***t**>>; where the italicization is simply a reminder that 'e' ranges over a domain that includes *events* in which entities may participate. As a defense of (R),

(R) if <α> and <β> are types, so is <α, β>

this seems a bit lame. For the idea is not that '*e*' corresponds to yet another abstracted position, with 'sell' and 'give' exemplifying Level Five and Level Four types. Rather, limitations to Level Three are assumed, and paradigmatically transitive verbs are said to be instances of a Level Three type. Moreover, the questions about λ**D**.**TRANSITIVE**(**D**) and λz.λy.λx.**Between**(x, y, z) remain. But in any case, appealing to event variables casts further doubt on the idea that verb meanings are fundamentally relational.

3.1 Severing Agents

Drawing on Davidson (1967), one might describe the meaning of (55) with (55a);

(55) A dog chased a cat.

(55a) ∃e:**Past**(e)[∃x**:Dog**(x)[∃y:**Cat**(y)[**ChaseByOf**(e, x, y)]]

where boldface indicates Fregean mappings to truth values. But even if predicates like ‘dog’ and ‘cat’ are semantically relational in this sense—see note 9—the verb ‘chase’ may not be an instance of type <e, <e, *e***t**>>. There are other options. Consider (55b),

(55b) ∃e:**Past**(e)[∃x**:Dog**(x)[**DoneBy**(e, x)] **&** ∃y:**Cat**(y)[**ChaseOf**(e, y)]]

in which the ampersand conjoins expressions of type <e**t**>. And ‘**ChaseOf**(e, y)’ might be replaced with ‘**Chase**(e) **& DoneTo**(e, y)’. Kratzer (1996) severs Agent-representations from verb meanings. Schein (1993, 2002) urges more general thematic separation; see also Pietroski (2005), Williams (2007). Here, let me note one cluster of considerations that motivate at least Kratzer-style severing, and so *not* specifying verb meanings as instances of type <e, <e, *e***t**>>.

The passive (56) presents difficulties if ‘chase’ is semantically triadic.

(56) A cat was chased.

There is no independent grammatical evidence that (56) includes a covert representation of a chaser, akin to the overt subject in (57).

(57) Something chased a cat.

While (56) seems to imply (57), this may be due to our knowledge that chases have chasers, not that ‘chase’ requires a chaser-argument; cp. (58), which in my view, doesn’t imply (59).

(58) An icecube melted.

(59) Something melted an icecube.

One might try to accommodate (56) with a “passivizing” morpheme of the Level Four type <<e, <e, *e***t**>>, <e, *e***t**>>. But we can say instead that ‘chase’ is of type <e, *e***t**>. Then combining ‘chase’ with one argument yields a phrase of type <*e***t**>; and (57) can be accommodated with a Kratzerian “voice head” of the Level Three type <*e***t**, <e, *e***t**>>. On this view, (57) is like (59) in having two verbal heads, each supporting one argument: [something [*v* [melted [an icecube]]]]; see also Chomsky (1995), Pietroski (2005), Borer (2005), Lohndal (2014).

This suggests an explanation for why (18), *doesn’t* have the reading indicated with (18c).

(18) a boy saw a man with a telescope

(18a) A boy saw a man that had a telescope.

(18b) A boy saw a man by using a telescope.

(18c) #A boy saw a man and had a telescope.

If ‘see’ is eventish but semantically *dyadic*, with no variable for an experiencer, then (60)

(60) see a man with a telescope

can be understood as (60a) or (60b), but not as (60c).

(60a) ∃y:**Man**(y)[**SeeingOf**(e, y)] **&** ∃z:**Telescope**(z)[**With**(y, z)]

(60b) ∃y:**Man**(y)[**SeeingOf**(e, y)] **&** ∃z:**Telescope**(z)[**With**(e, z)]

(60c) #∃y:**Man**(y)[**SeeingByOf**(e, x, y)] **&** ∃z:**Telescope**(z)[**With**(x, z)]

In particular, the verb phrases (60-G) and (60-GG) correspond to (60a) and (60b), respectively.

(60-G) [[see [a [man [with a telescope]]]]]

(60-GG) [[[see [a man]] [with a telescope]]]

By contrast, we face difficulties if ‘see a man’ is an expression of type <e, *e***t**> that can be adverbially modified by ‘with a telescope’. This would require either a covert element of type <*e***t**, <<e, *e***t**>, <e, *e***t**>>>—appended to the prepositional phrase of type <*e***t**>—or an equivalent rule of “restriction” for dyadic predicates, as posited by Chung and Ladusaw (2003). Moreover, blocking the unwanted reading would require a notion of restriction according to which the modifier *can and must* skip over the first open position in the allegedly dyadic predicate. One can insist that this is how adverbial modification works. But then the question is why only adverbial modification is possible. In a Fregean language, (60) could have all three readings.

Appeal to Kratzer-style severing presupposes that a verb phrase can be *conjoined* with a predicate like '∃x**:Dog**(x)[**DoneBy**(e, x)]'. But positing a rule of Predicate Conjunction, as in Heim and Kratzer (1998)—or a covert element of the Level Three type <et, <et, et>>—is arguably no worse than appealing to quantificational meanings of type <et, <et, t>> and a voice head of type <et, <e, et>>. Though if verbs do not exhibit semantic types from above Level Two, one might wonder if appeal to Level Three *functional* vocabulary reflects imposition of Fregean typology on Slangs, as opposed to an independently attractive hypothesis. So one might look for alternatives to positing any morphemes of type <et, <e, et>> or <et, <et, t>>.

3.2 Minimal Relationality

Consider (61), whose meaning might initially be specified with the Tarskian (61a) or (61b).

(61) A cat arrived.

(61a) ∃e[Past(e) & ∃x[Cat(x) & ArrivalOf(e, x)]]

(61b) ∃e:Past(e)[∃x:Cat(x)[ArrivalOf(e, x)]]

Neither specification seems quite right, given the permissive characters of '&' and '∃'. Recall that the Tarski-style ampersand can connect open sentences of any adicity, and the adicity of the resulting conjunction can be higher than that of either conjunct, as in (14).

(14) Nx & [Pyz & Nw]

To be sure, there are also respects in which Slangs go beyond a Tarski-style predicate calculus. But in searching for models of Slangs, one might look for a modest extension of *some fragment* of a Tarski-style language, instead of supplementing a language that already generates (14) with still fancier expressions that denote truth values and higher-order functions.

Given lambda expressions, (61b) might be replaced with (61c),

(61c) ∃:**Past**[λe.∃:**Cat**[λx.**ArrivalOf**(e, x)]]

using 'e' and 'x' as indices of abstraction. Relative to each assignment 𝒜 of values to indices: **ArrivalOf**(e, x) is a truth value— ⊤ or ⊥, depending on whether or not 𝒜(e) is an arrival of 𝒜(x); and λx.**ArrivalOf**(e, x) is a function that maps each entity to ⊤ if and only if 𝒜(e) is an arrival of that entity. This function is mapped to ⊤ by the quantificational function ∃:**Cat** if and only if some cat is such that 𝒜(e) is an arrival of that cat. So one can say that (61c) denotes ⊤ if and only if some event is an arrival of some cat. But the same effect can be achieved with less power.

In (61c), 'λx' plays a dual role: it targets an index in the dyadic predicate; it also converts an expression of type <t> into an expression of type <et>. In this latter respect, 'λx' is unlike '∃x' in (61a). But 'λx' is also unlike '∃' and '∃:**Dog**', neither of which can target either position in a dyadic predicate. With this in mind, let's imagine a language that allows for atomic sentences with one or two open positions—e.g., 'Cat(_)' and 'ArrivalOf(_, _)'—but no more.

One-placers are instances of a basic type <M>; two-placers, which permit expression of dyadic relations, are instances of a basic type <D>. There are no other types. And every complex expression of this language, Monadish, is an instance of type <M>. All instances of type <D> are atomic. This language includes no lambda expressions. Strings like (61c) cannot be generated. Indeed, Monadish has no variables. So adding lambdas wouldn't help, since strings like (14) are also ungenerable. Still, boundlessly many Monadish expressions have dyadic consitutents.

There are a few combinatorial operations. The simplest is “M-junction,” via which two expressions of type <M> can be combined to form a third: ‘Grey(_)^Cat(_)’ applies to something if and only if both ‘Grey(_)’ and ‘Cat(_)’ apply to it. Faced with speakers of Monadish, Fregean field semanticists might adopt the following hypothesis: ‘Cat(_)’ indicates a function **C** of type <e**t**>; ‘Grey(_)’ indicates a function **G** of the same type; and ‘Grey(_)^Cat(_)’ indicates λx. ⊤ if **G**(x) = ⊤ **&** **C**(x) = ⊤, and otherwise ⊥. But this overintellectualizes. Instances of type <M> are non-relational. Like the Tarskian sentence ‘Cat(x)’, an expression of type <M> does not indicate a function. Put another way, the Fregean apparatus is an overly powerful tool for describing the I-language that speakers of Monadish acquire. That apparatus could also be used to describe a procedure according to which joining ‘Grey(_)’ with ‘Cat(_)’ forms a phrase that indicates a dyadic function: λy.λx. ⊤ if **G**(x) = ⊤ & **C**(y) = ⊤, and otherwise ⊥. But by hypothesis, speakers of Monadish cannot understand ‘grey cat’ as an expression of type <D>.[15]

On the contrary, these speakers understand ‘arrive cat’ as an expression of type <M>, akin to the English ‘arrival of a cat’. More generally, an atomic Monadish expression of type <D> can combine with an expression of type <M> via the operation “D-junction,” which yields another expression of type <M>. Combining ‘ArrivalOf(_, _)’ with ‘Cat(_)’ with yields ‘∃[ArrivalOf(_, _)^Cat(_)]’, which is understood in accord with the following constraints: the existential closure applies to the *monadic* predicate, which must be linked to the *second* position of the dyadic predicate. The mandatory construal is indicated in (62),

(62) ∃[ArrivalOf(_, _)^Cat(_)]

which is an analog of (62a) in a Tarski-style language that also generates (62b) and (62c).

(62a) ∃x[ArrivalOf(e, x) & Cat(x)]
(62b) #∃e[ArrivalOf(e, x) & Cat(x)]
(62c) #∃x[ArrivalOf(e, x) & Cat(y)]

But ‘∃[ArrivalOf(_, _)^Cat(_)]’ is not disambiguated by inserting variables that reflect a choice of how to link the unsaturated positions in (62); the D-junction applies, unambiguously, to e if and only if e is an arrival of a cat.

If it helps, think of ‘∃’ as skipping over ‘Arrival(_, _)’—since neither position can be targeted by the variable-free closer—and think of the second position in ‘Arrival(_, _)’ as one that needs some link to independent content, since the first position is for events in which something arrives. In general, an instance of ‘∃[D(_, _)^M(_)]’ applies to e if and only if e bears the dyadic relation to *something* that has the monadic property. So an expression of type <M> can have an atomic constituent of type <D> whose second position was restricted and closed via D-junction. This operation is a little more complex than M-junction. But it is still severely restricted to inputs of certain types, and so less powerful than the rather sophisticated operation of function-application that is associated with (R).[16]

(R) if <α> and <β> are types, so is <α, β>

[15] Likewise, humans cannot understand ‘grey cat’ as applying to pairs <*x*, *y*> such that *x* is grey and *y* is a cat.

[16] We can view D-junction as a special case of Higginbotham’s (1985) posited operation of *theta-marking*, which allowed for polyadicity, while M-junction is the analog of what Higginbotham called *modification*.

In this context, it is worth remembering that ‘a cat’ may be of the same type as ‘cat’; cp. Kamp (1981), Heim (1982). From a Fregean perspective, this seems odd. Why say that ‘a cat’ combines with ‘arrive’ in a conjunctive way that involves covert existential closure, instead of treating ‘a cat’ as a quantifier of type <e**t**, **t**>? But in languages without indefinite articles, the closure is covert either way. Moreover, (63) is understood as in (63a); cp. (64) and (65).

(63) see a cat arrive

(63a) ∃[SeeingOf(_, _)^∃[ArrivalOf(_, _)^Cat(_)]]

(64) Dogs saw cats arrive.

(65) Cats fell on cats.

So it isn’t obvious that in (63), ‘a’ signifies existential closure; see also Higginbotham (1987).

Slangs may exhibit more semantic typology than Monadish. But whatever the facts about Slangs, Monadish at least provides a model for how a language can allow for a minimal degree of relationality—*viz.*, locally bounded dyadicity—without generating expressions of Fregean types. I have argued elsewhere that the usual range of textbook examples, along with various puzzling cases, is plausibly accommodated in these terms; see Pietroski (2005, 2014, 2018a). But there is ample room for other models. Consider Dyadish, which admits atomic expressions of a third type—exhibited by atomic open sentences like ‘ChaseByOf(_, _, _)’—and generates complex expressions of type <D>. Though if children acquire languages that are more like Monadish than Frege’s *Begriffsschrift*, perhaps semanticists should try to supplement spare models, instead of trying to limit powerful ones. If plausible analyses of specific constructions cannot be plausibly recoded in terms of the types <M> and <D>, that is evidence that Slangs generate expressions of other types. But we shouldn’t insist on Fregean types.

3.3 Eschewing Level Zero

Obviously, <M> and <D> are analogs of <e**t**> and <e, e**t**>. But <M> and <D> are *basic* types, not abstract types from Levels One and Two of a hierarchy that starts with <e> and <**t**>. Let’s stipulate that Monadish does not permit denoters—of entities or truth values. It can still allow for predicates like ‘Felix(_)’, which may apply to at most one thing. Monadish also allows for (66),

(66) Demonstrated(_)^∃[CalledWith(_, _)^ Sound-of‘Felix’(_)]

which would apply to an entity e in context C if and only if e is both demonstrated in C and (properly) called with the sound of ‘Felix’. These expressions have Fregean analogs of type <e**t**>. But the question is not whether Monadish has expressions of type <e**t**>. It doesn’t. The question is whether Slangs have expressions of the basic Fregean types <e> and <**t**>.

I can’t review here the many reasons for thinking proper nouns are *not* expressions of type <e>. But the history of this topic is one of considering alternatives. A quantificational analysis, *a la* Russell/Montague, would be attractive if phrases like ‘a/the/every cat’ are of type <e**t**, **t**>. Though as Quine (1963) suggested, in discussing ‘Pegasus’, predicative analyses have their own attractions. Burge (1973) and Katz (1994) offered insightful proposals that can be formulated in Monadish-friendly terms. Deictic pronouns and traces of displacement can also be described as instances of type <M>; see Pietroski (2014, 2018a). One can easily imagine languages that forbid expressions like (67).

(67) The three Tylers at the party included that nice Professor Tyler Burge.

So if kids *could* become adults who treat ‘Tyler’ as an instance of type <e>, why don’t they?

Of course, describing ‘Tyler’ as an instance of type <e**t**> does not avoid appeal to the Fregean types <e> and <**t**>. But if we don’t need the Fregean hierarchy of types, perhaps we can eschew <e> in favor of <M>. Eschewing appeal to <**t**>, as a semantic type, is also pretty easy.

Tarski showed how to describe true/false sentences as those satisfied by all/no sequences. And if we only have to worry about monadic predicates, we can posit a pair of operators—⇑ and ⇓—that combine with expressions of type <M> to form "polarized" predicates of the same type. For any expression Σ of type <M> and any entity e: the result of prefixing Σ with '⇑' (⌜⇑Σ⌝) applies to e if and only if Σ applies to something; the result of prefixing Σ with '⇓' (⌜⇓Σ⌝) applies to e if and only if Σ applies to nothing. If there is at least one cat, then '⇑Cat(_)' applies to each thing in the domain, and '⇓Cat(_)' applies to nothing. If the domain is catless, then '⇑Cat(_)' applies to nothing, and '⇓Cat(_)' applies to each thing. Thus, '⇑Cat(_)' and '⇓Cat(_)' are like 'such that there is a cat' and 'such that there is no cat'. Truth tables can be reconstructed in these terms. So we can do without <**t, t**> and <**t**, <**t, t**>>.

For each thing, the polarized predicate '⇑[Past(_)^∃[ArrivalOf(_, _)^Cat(_)]]' applies to it if and only if there was an arrival of cat; cp. 'such that a cat arrived'. And we can say that relative to any assignment 𝒜 of values to indices '⇑[Past(_)^∃[ArrivalOf(_, _)^1(_)]]' applies to each thing or to nothing, depending on whether or not there was an arrival of whatever 𝒜 assigns to the first index; cp. 'such that 𝒜(1) arrived'. If indices can be devices of abstraction, as discussed above, then it is a small step to allowing for (68);

(68) 1-⇑[Past(_)^∃[ArrivalOf(_, _)^1(_)]]

where this clausal expression of type <M> applies to an entity e if and only if e is such that there was an arrival of e; cp. 'thing that arrived' and the earlier discussion of Heim and Kratzer (1998).

This doesn't show that tensed expressions are *not* instances of type <**t**>. One can hypothesize that tense morphemes are restricted existential quantifiers of type <*e***t**, **t**>; cp. 'λΦ.∃*e*:Past(*e*)[**Φ**(*e*)]'. But on this view, a single morpheme does two very different jobs: it makes its own contribution as a temporal *restrictor* of an event predicate; and it *closes* that predicate, creating an expression of type <**t**>. But if 'a cat' need not be a restricted existential quantifier, why think that a tense (or aspect) morpheme is a quantifier? One can treat tensed expressions as instances of type <e**t**> and posit a covert quantifier of type <*e***t**, **t**>. But then why not treat tensed expressions are instances of type <M> and posit a covert polarizer?

Partee (2006)—prompted by Carstairs-McCarthy's (1999) discussion of sentences and noun phrases—asked whether we to need to posit *two* basic semantic types. I share Partee's view that "tradition" explains much of the current reliance on appeals to type <**t**> in theories of meaning for Slangs. But I think that much the same can be said about reliance on appeals to type <e>. Frege's reasons for taking his two "saturating" types as basic, and describing expressions of nonbasic types as "unsaturated," do not show that Slangs fit this mold. Tarski showed that a language can have name-like expressions that are not of type <e> and sentences that are not of type <**t**>. So in offering theories of Slangs, we should not assume that they generate expressions of boundlessly many types, as opposed to a few basic types like <M> and <D>.

3.4 Level Three Revisited

Still, one might wonder: what's the alternative to describing overt quantifiers, like 'every' and 'most', as instances of type <e**t**, <e**t**, **t**>>? One suggestion is that they are expressions, of type <D>, which apply to ordered pairs of *sets of* things in the relevant domain. This may be no worse than invoking higher-order functions, and at least we could avoid specifying the meaning of 'every' in terms of truth values. But quantificational words can also be described as expressions that apply more directly to ordered pairs of the things quantified over, given plural quantification as discussed by Boolos (1998) and others. I conclude by indicating how; for details, see Pietroski (2018a).

Let's stipulate that for any entities e and e′, e is the "inner participant" of the ordered pair <e′, e>—a.k.a. {e′, {e′, e}}—whose "outer participant" is e′. Ordered pairs can meet various conditions, distributive and collective. For example, some pairs might be such that each of their inner participants is a cat, and their outer participants are three dogs. Given ordered pairs of things in a domain that includes the cats, every cat arrived if and only if some pairs meet the following three conditions: every one of their inner participants is one of their outer participants; their inner participants are the cats; and their outer participants are the things that arrived.

Recalling the discussion of conservativity, 'every cat' lets us ignore the noncats. We can encode the restricted character of this quantification by assuming, for Slangs, ordered pairs of a very limited sort: outer participants must be inner participants, as if determiners indicate nested second-order *classifications*, as opposed to genuine second-order *relations*; the inner participants reflect an initial selection from the domain via some predicate like 'cat', while outer participants reflect a second selection from the restricted domain (i.e., the initial selection) via some predicate like 'arrived'. Echoing Barwise and Cooper's (1981) idea that determiners live on their internal arguments, we can say that Slang determiners are limited to "inner-focused" ordered pairs. But the suggestion is not that this generalization about Slangs reflects a special constraint on which *relations* determiners can indicate. The suggestion is rather that Slang determiners are analogs of some Fregean quantifiers that can be mimicked with nested second-order classification.

Given a general "outers must be inners" constraint on Slang determiner meanings, the meaning of 'every' can be specified with an *identity* condition. Every cat arrived if and only if some pairs are such that: their outer participants *are* their inner participants; their inner participants are the cats; and their outer participants arrived. There are various ways of converting this observation into a compositional semantics; see Pietroski (2005, 2018a). But given a maximizing operator that converts 'Cat(_)' into the plural predicate 'TheCats(_)', Monadish permits '∃[Internals(_, _)^TheCats(_)]', which applies to ordered pairs whose internal participants are the cats. Encoding the condition that corresponds to a determiner's external argument is also easy if that argument is a sentential/polarized predicate of type <M>.

Relative to any assignment $\mathcal{A}$ of values to indices, the predicate 'such that t_1 arrived' applies to Felix if and only if $\mathcal{A}(1)$ arrived; so relative to a minimal variant of $\mathcal{A}$ that assigns Felix to the indexed trace, 'such that t_1 arrived' applies to Felix if and only if Felix arrived. So given Tarski-style quantification over the minimal variants of $\mathcal{A}$ that correspond to a restricted domain of cats, the open sentence 't_1 arrived' can be used to specify the cats that arrived. This can be viewed as a special case of lambda abstraction. But appealing to this very special case—needed to accommodate relative clauses, whatever we say about quantification—is not tantamount to invoking Level Three abstraction in a Fregean language.

Recall that neither equinumerosity nor the second-order identity relation is conservative. But if determiners don't indicate second-order relations, we don't need to say that (61) and (62)

(61) $\lambda\Psi.\lambda\Phi.\ \Phi = \Psi$

(62) $\lambda\Psi.\lambda\Phi.\#\{x: \Phi(x)\} = \#\{x: \Psi(x)\}$

are less conceptually natural than (63), which specifies the function corresponding to 'most'.[17]

(63) $\lambda\Psi.\lambda\Phi.\#\{x: \Phi(x)\ \&\ \Psi(x)\} > \#\{x: \Phi(x)\} - \#\{x: \Phi(x)\ \&\ \Psi(x)\}$

[17] If seven of ten cats arrived, then most of the cats arrived; and correlatively, the number of cats that arrived exceeds the result of subtracting that number from ten—i.e., 7 > (10 –7). See Lidz. et. al. (2011) for evidence that speakers do understand 'most' in terms of cardinalities and subtraction.

Neither (61) nor (62) corresponds to a condition on inner-focused ordered pairs, which reflect predications-within-predications, as opposed to genuine second-order *relations*. But while (63) is intuitively more complex, it does correspond to a condition on inner-focused ordered pairs. Given some pairs of this sort, they meet the ‘most’-condition if and only if the number of their external participants exceeds the result of subtracting that number from the number of their internal participants. The complexity can be seen as a way of mimicking a Fregean quantifier.

Appealing to meanings of type <e**t**, <e**t**, **t**>> is not the only game in town. On the contrary, it raises the question of why so many meanings of this type are not attested. More generally, appealing to (R) and the two basic Fregean types is neither inevitable nor desirable.

(R) if <α> and <β> are types, so is <α, β>

Positing meanings of the Monadish types <M> and <D> may be enough.

References

Baker, M. (1997). Thematic Roles and Grammatical Categories. In L. Haegeman (ed.), *Elements of Grammar*, Dordrecht: Kluwer, pp. 73-137.

Barwise, J. and Cooper, R. 1981: Generalized Quantifiers and Natural Language. *Linguistics and Philosophy* 4:159-219.

Berwick, R. et.al. (2011). Poverty of the Stimulus Revisited. *Cognitive Science* 35: 1207-42.

Boolos, G. (1998). *Logic, Logic, and Logic*. Cambridge, MA: Harvard University Press.

Borer, H. (2005). *Structuring Sense* (volumes I and II) Oxford: Oxford University Press.

Burge, T. (1973). Reference and Proper Names. *Journal of Philosophy* 70:425-39.

Carstairs-McCarthy, A. (1999). *The Origins of Complex Language: An Inquiry into the Evolutionary Beginnings of Sentences, Syllables, and Truth*. Oxford: OUP.

Chierchia, G. (1984). *Topics in the Syntax and Semantics of Infinitives and Gerunds*. UMASS Dissertation.

Chomsky, N. (1957). *Syntactic Structures.* The Hague: Mouton.

—(1964). *Current Issues in Linguistic Theory.* The Hague: Mouton.

—(1965). *Aspects of the Theory of Syntax*. Cambridge, MA: MIT Press.

—(1986). *Knowledge of Language*. New York: Praeger.

—(1995). *The Minimalist Program*. Cambridge, MA: MIT Press.

Chung, S. & Ladusaw, W. (2003). *Restriction and Saturation*. MIT Press.

Church, A. (1941). *The Calculi of Lambda Conversion*. Princeton: Princeton University Press.

Crain, S. and Pietroski, P. (2001). Nature, Nurture, and Universal Grammar. *Linguistics and Philosophy* 24:139-186.

—(2011). The Language Faculty. In *The Handbook of Philosophy of Cognitive Science* (E. Margolis, R. Samuels, and S. Stich, eds.), Oxford University Press.

Davidson, D. (1967). The Logical Form of Action Sentences. *Essays on Actions and Events*. Oxford: Oxford University Press.

Frege, G. (1879). *Begriffsschrift*. Halle: Louis Nebert. English translation in *From Frege to Gödel: A Source Book in Mathematical Logic, 1879-1931,* J. vanHeijenoort (ed). Cambridge, MA: Harvard University Press, 1967.

— (1884). *Die Grundlagen der Arithmetik*. Breslau: Wilhelm Koebner. English version in *The Foundations of Arithmetic*, J. L. Austin (trans). Oxford: Basil Blackwell, 1974.

— (1892). Function and Concept. In Geach & M. Black (trans). *Translations from the Philosophical Writings of Gottlob Frege*. Oxford: Blackwell, 1980.

Heim, I. (1982). The Semantics of Definite and Indefinite Noun Phrases, University of Massachusetts: Ph.D. dissertation; published 1989, New York: Garland.

Heim, I. & Kratzer, A. 1998. *Semantics in Generative Grammar*. Oxford: Blackwell.
Herburger, E. (2000). What counts: focus and quantification. Cambridge, MA: MIT Press.
Higginbotham, J. (1983). The Logical form of Perceptual Reports. *J. Phil.* 80:100-27.
—(1985). On Semantics. *Linguistic Inquiry* 16:547-93
Higginbotham, J. and May, R. (1981). Questions, Quantifiers, and Crossing. *The Linguistic Review* 1: 47-80.
—(1987). Indefiniteness and Predication. In *The Representation of (In)definiteness*. E. Reuland & A. ter Meulen (eds). Cambridge, MA: MIT Press.
Horty, J. (2007): *Frege on Definitions: A Case Study of Semantic Content*. Oxford: OUP.
Jacobson, P. (1999). Variable Free Semantics. *Linguistics and Philosophy* 22:117-84.
—(2014). Compositional Semantics. Oxford: OUP.
Kamp, Hans. 1981. A theory of truth and semantic representation. Reprinted in Portner and Partee, eds., 2002.
Katz, J. (1994). Names Without Bearers. *Philosophical Review* 103: 1-39.
Kratzer, A. (1996). Severing the External Argument from its Verb. In J. Rooryck and L. Zaring (eds.), Phrase Structure and the Lexicon. Dordrecht (Kluwer Academic Publishers).
Larson, R. (1988). On the Double Object Construction. *Linguistic Inquiry* 19: 335-391.
Lidz, J. et.al. (2011). Interface Transparency and the Psychosemantics of 'Most'. *Natural Language Semantics* 19: 227-56.
Lohndal, T. (2014). *Phrase Structure and Argument Structure.* Oxford: Oxford University Press.
Lewis, D. (1975). Languages and Language. In K. Gunderson (ed.), *Minnesota Studies in the Philosophy of Science*, Volume 7, Minneapolis: University of Minnesota Press, pp. 3–35.
Ludlow, P. (2011). *The Philosophy of Generative Grammar*. New York: OUP.
Montague, R. (1974). *Formal Philosophy*. New Haven: Yale University Press.
Partee, B. (2006). Do we need two basic types? In *40-60 puzzles for Manfred Krifka* http://www.zas.gwz-berlin.de/40-60-puzzles-for-krifka.
Pietroski, P. (2005). *Events and Semantic Architecture*. Oxford: Oxford University Press.
—(2014). Lexicalizing and Combining. In *Verb Concepts: Cognitive Science Perspectives on Verb Representation and Processing,* R. de Almeida & C. Manouilidou (eds).
—(2018a, in press). Conjoining Meanings: Semantics Without Truth Values. Oxford: OUP.
—(2018b, in press). Semantic Typology and Composition. In *The Science of Meaning* (B. Rabern and D. Ball, eds.), Oxford University Press.
Quine, W. (1963). On What There Is. In *From a Logical Point of View*, New York: Harper and Row.
Schein, B. (1993). *Events and Plurals*. Cambridge, MA: MIT Press.
—(2002). Events and the Semantic Content of Thematic Relations. In *Logical Form and Language* (G. Preyer and G. Peters, eds.), 91-117, Oxford University Press.
— forthcoming. Conjunction Reduction Redux. Cambridge, MA: MIT Press.
Steedman, M. (1996). Surface Structure and Interpretation. Cambridge, MA: MIT Press.
Tarski, A. (1944). The Semantic Conception of Truth. *Philosophy and Phenomenological Research* 4: 341-75.
Williams, A. (2005). *Complex Causatives and Verbal Valence*. Doctoral Dissertation: University of Pennsylvania.
—(2007). Patients in Igbo and Mandarin. In J. Dölling and T. Heye-Zybatow (eds.) *Event Structures in Linguistic Form and Interpretation* (Berlin: Mouton de Gruyter).
—(2015). *Arguments in Syntax and Semantics*. Cambridge: CUP.