1 Preliminaries

1.1 TWO PROBLEMS

There are at least two significant problems with writing a book about the semantics of modification. The first is that it's not at all clear what modification is, precisely. The second is that it's not at all clear **whether** it is – that is, whether it exists as a single coherent grammatical phenomenon.

'Modification' and 'modifier' are the sorts of terms that we routinely use as though they had agreed-upon theoretical content. Yet they're useful in part precisely because, as McNally (forthcoming) observes, they lack a generally accepted, formally explicit theoretical definition. In the absence of a theoretical definition, it wouldn't be unreasonable to expect a clear descriptive one. Even here, though, we may need to set aside the 'clear' and, for that matter, the 'one'. In most contexts, to say that something is a modifier, or that it modifies something else, is not to make a falsifiable claim. Of course, that doesn't mean such claims are inherently suspect, but it's best not to have any illusions about how much weight they can bear.

That's the first problem, the terminological one. The second problem is more profound: to solve the first problem and provide a solid definition of modification, it would really help if it were a single phenomenon or natural class of phenomena. But it may be that 'modification' is merely a cover term for a motley assortment of constructions, facts, and puzzles that may, in various combinations, have some features in common.

Of course, it's not necessary to solve these problems in order to talk about them. Perhaps it's only in talking about them at some length that one can begin to address them. It might be an interesting journey, even if it turns out that modification isn't really a useful notion semantically. Nevertheless, the term appeals to us for some reason. Surely we should ask whether it does so because there is, in fact, a genuine grammatical insight behind it, something in the real world to which it refers?

Before we can address this, there is some practical business to attend to.

1.2 WHAT THIS BOOK IS AND ISN'T

This book is about formal linguistic semantics. That said, I really hope it might prove useful to people approaching it from other theoretical and methodological perspectives as well – if nothing else, in its characterization of the facts and of various particular puzzles. It has two primary target audiences. One is graduate students and advanced undergraduates who have undergone the initial rites of passage into formal semantics and have (at least) survived with their will to continue intact. Another is researchers in related fields, who sometimes find themselves in a distinct though not entirely dissimilar situation. They may have a longstanding familiarity with work in semantics, but a passive one, as spectators but not practitioners. If they would like to play a more active role, neither general introductory texts nor handbook articles are ideally suited to their needs.

This is intended as something between an advanced textbook and a topical survey of research in a broad area, a bridge between the basic orderly framework-building of textbooks and the sophisticated, cacophonous, and often formally challenging to-and-fro of the primary literature. The aim is to present some analytical tools and concepts that can serve as a starting point for the reader's own research. It's to provide a way of thinking about a particular set of problems and a sense of where to look to find out more.

A number of things follow from that. First, I have tried to emphasize problems over particular solutions and analytical strategies over particular instances of them. That said, the most interesting problems often emerge only against the background of some theoretical assumptions. It's impossible to be surprised if you have no expectations.

Second, there is no attempt here to be comprehensive. 'Modification' is a topic so broad that it could encompass virtually all of semantics. There may be no area of the field in which some class of modifiers hasn't been a major concern. So, in the interests of keeping the book a reasonable length – in fact, finite – there are many interesting topics of potential discussion that I will forgo. Discussion of adverbials other than adverbs in the strict sense will be conspicuously absent, as will discussion of relevant work in psycholinguistics and language acquisition. The focus will be on the grammar of adjectives, adverbs, and degrees.

Third, I have tried to maintain a consistent theoretical framework throughout. When encountering the literature for the first time, people are sometimes struck with a kind of intellectual vertigo. They have a few hard-won analytical tools in hand, but soon discover that work in semantics varies widely in formalism, style of analysis, and theoretical assumptions. It's as though they had just learned Italian only to find, upon visiting Italy, that people freely switch between Italian, French, Portuguese, Latin, and for some reason Japanese – and a handful of people seem to be saying really interesting things in Klingon. There is no solution to this in the long term other than to learn to deal with it. Nevertheless, I have enforced an artificial consistency on the discussion, translating various ideas into a single analytical and representational language. (Italian, one is tempted to say, taking the analogy too far.) This, of course, entails making many small adjustments to the original proposals, and a few larger ones. I call attention to the latter.

The book presupposes familiarity with the essential tools of formal semantics. I've tried to keep things relatively accessible, but engaging most of the content fully will require some previous background. Having absorbed the first few chapters of Heim and Kratzer (1998), Chierchia and McConnell-Ginet (1990) – or the relevant parts of volume two of Gamut (1991) or certain other semantics textbooks – should be sufficient. That should include a general understanding of quantification, lambda abstraction, and semantic types.

I have attempted to make the chapters of the book as independent of each other as possible. There are some dependencies that are difficult to avoid, though – you will get more out of Chapter 4 (on comparatives) if you have first read Chapter 3 (on vagueness, degrees, and the lexical semantics of gradable predicates). Chapter 6 (on crosscategorial phenomena) is best read in light of all preceding ones. But, on the other hand, if you wanted to skip past further preliminaries now and dive right into Chapter 2, you would not suffer unduly for having done so.

1.3 BACKGROUND ASSUMPTIONS

1.3.1 Glossing logical notation

Some introductory courses and textbooks develop a sophisticated semantics without recourse to logical notation other than lambdas, so I should briefly gloss the symbols I'll rely on. Many readers will want to skip this section. Obviously, one shouldn't mistake it for the shortest introduction to logic ever written. It just provides a way of mapping symbols onto familiar concepts or natural-language paraphrases.

First, some connectives, which make new propositions out of old ones:

(1)	$\neg p$	'it's not the case that p' or 'p is false'
	$p \wedge q$	'p and q'
	$p \vee q$	'p or q (or both)'
	$p \rightarrow q$	'if p, then q' or 'p is false or q is true'

Only the last of these is tricky. It's customary to paraphrase it as a conditional, but the second, more unwieldy paraphrase is more accurate. For our purposes, remembering the intuitive version will suffice.

Next, quantifiers:

(2)	$\exists x[\ldots]$	'there is an x such that …'
	$\forall x[\ldots]$	'for every x, …'
	$\forall x \in S[\ldots]$	'for every x in the set S, …'

Combining these elements, an existentially quantified sentence like (3) can be represented with conjunction:

(3) a. A dog is furry.
b. $\exists x[\mathbf{dog}(x) \wedge \mathbf{furry}(x)]$
'there is an x such that x is a dog and x is furry'

But for universal quantification, the conjunction strategy won't fly. *Every dog is furry* doesn't mean that for every x, x is a dog and furry – that would require that everything be a dog. So we need another connective:

(4) a. Every dog is furry.
b. $\forall x[\mathbf{dog}(x) \rightarrow \mathbf{furry}(x)]$
'for every x, if x is a dog, then x is furry'

This correctly restricts our attention to dogs.

1.3.2 Theoretical framework

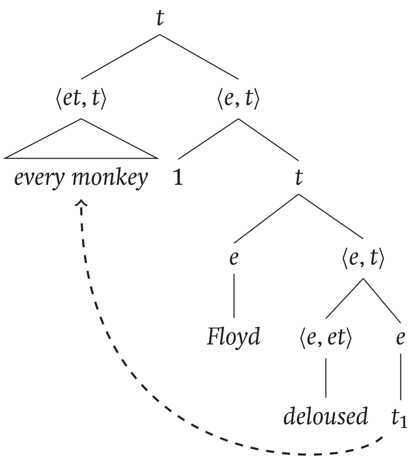
The question 'what theoretical framework are you using?' has two answers, one short and the other long. The short one is 'Heim and Kratzer (1998), more or less, with variations'. For many readers, this will be sufficient, and they need not bother with the rest of section 1.3.2. For the rest, here's the long answer.

As stated, the book adopts the Heim and Kratzer framework in most things. One departure is that it will use less English and more logic as a metalanguage in stating denotations. Even so, although most of the denotations will be well-formed logical expressions of an appropriate logic, I will follow Heim and Kratzer in treating them as components of a metalanguage that might at times include bits of English as well. I won't adopt an indirect interpretation system of the classically Montagovian sort, in which much of the semantics resides chiefly in how expressions of natural language are translated into expressions of a logic. One additional peculiarity is that I've systematically curried/schönfinkeled all logical predicates for consistency – that is, I will write $\mathbf{eat}(y)(x)$ rather than $\mathbf{eat}(x, y)$.

As for the syntactic assumptions, they are conventionally generative but with a minimum of theoretical commitments. For the most part, only the shape of trees – the constituency, not syntactic category – will matter, and I'll often omit syntactic category labels entirely. Where a neutral term like 'nominal' becomes inappropriate, I assume DP is the category of, for example, *the monkey from Cleveland* (Abney 1987), and NP as the category of the next maximal projection down (*monkey from Cleveland*).

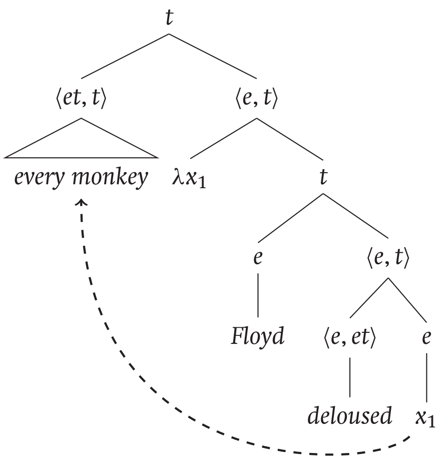
I assume that the syntax has movement, and that quantified nominals usually take scope by undergoing Quantifier Raising (QR). The way I'll represent movement will diverge in a notational way from Heim and Kratzer's. In their standard treatment, a moved expression such as a generalized quantifier leaves behind an individual-denoting trace in the position it previously occupied. This trace receives a numerical index. By moving, the quantifier creates next to its landing site a binder for this index. This is represented as a number that occupies a node in the tree, which branches from the node to which the displaced quantifier is attached. Thus, for them, QR looks like this:

(5) a. Floyd deloused every monkey.

b.


The trace is then interpreted as a variable, over which the binding node triggers lambda abstraction. In contrast, I'll represent movement in this way:

(6)



This simply replaces the trace with the corresponding variable and the binder with the corresponding lambda. (It's a little easier to read than the original when there are both individual- and degree-denoting expressions moving.) In a somewhat more unusual move, I will use variables with numerical indexes whenever they are associated with movement, as a subtle reminder of the more standard indexed-trace representation and of their connection to movement. The purist is free to disregard the non-subscripted material, which will render the representation virtually identical to the original.

As (6) reflects, I will occasionally place variables directly into the object language – that is, hang them from trees or from expressions in trees – in, again, a relatively standard fashion. Variables introduced this way and left free are assumed to get their value from the context(ually supplied assignment function).

Here is an example of how a computation might run (I'll generally skip more steps than I do here):

(7) a. $[\![\textit{every}]\!] = \lambda P_{\langle e,t\rangle}\lambda Q_{\langle e,t\rangle} . \forall x[P(x) \rightarrow Q(x)]$

b. $[\![\textit{every monkey}]\!] = [\![\textit{every}]\!]([\![\textit{monkey}]\!])$
$= \lambda Q_{\langle e,t\rangle} . \forall x[[\![\textit{monkey}]\!](x) \rightarrow Q(x)]$
$= \lambda Q_{\langle e,t\rangle} . \forall x[\mathbf{monkey}(x) \rightarrow Q(x)]$

c. $[\![\textit{deloused}]\!] = \lambda x \lambda y . \mathbf{deloused}(x)(y)$

d. $[\![\textit{Floyd deloused } x_1]\!] = [\![\textit{deloused}]\!]([\![x_1]\!])([\![\textit{Floyd}]\!])$
$= [\lambda x \lambda y\,.\,\mathbf{deloused}(x)(y)](x_1)(\mathbf{Floyd})$
$= \mathbf{deloused}(x_1)(\mathbf{Floyd})$

e. $[\![\lambda x_1 \textit{ Floyd deloused } x_1]\!] = \lambda x_1\,.\,\mathbf{deloused}(x_1)(\mathbf{Floyd})$

f. $[\![\textit{every monkey}]\!]([\![\lambda x_1 \textit{ Floyd deloused } x_1]\!])$
$$= \left[\lambda Q_{\langle e,t\rangle}\,.\,\forall x\left[\begin{array}{c}\mathbf{monkey}(x) \rightarrow \\ Q(x)\end{array}\right]\right]\left(\left[\!\!\left[\begin{array}{l}\lambda x_1 \text{ Floyd} \\ \text{deloused } x_1\end{array}\right]\!\!\right]\right)$$
$$= \forall x\left[\begin{array}{l}\mathbf{monkey}(x) \rightarrow \\ \quad [\![\lambda x_1 \textit{ Floyd deloused } x_1]\!](x)\end{array}\right]$$
$= \forall x[\mathbf{monkey}(x) \rightarrow [\lambda x_1\,.\,\mathbf{deloused}(x_1)(\mathbf{Floyd})](x)]$
$= \forall x[\mathbf{monkey}(x) \rightarrow \mathbf{deloused}(x)(\mathbf{Floyd})]$

I have not represented the assignment function explicitly. Again, the purist can reconstruct how things would look if I had.[1]

The type system I assume is standard except where otherwise noted. On occasion, I will switch into an intensional system with overt quantification over possible worlds.

1.3.3 Notational and typographical conventions

The conventions I'll observe, notational and typographic, are relatively self-explanatory, but for the sake of explicitness, I'll list them:

- I will omit '1 iff' in, e.g., '$[\![\textit{Floyd exploded.}]\!] = 1$ iff **exploded**(**Floyd**)' and write, e.g., $f(x)$ in place of $f(x) = 1$
- constants will be in **boldface**, variables in *italics*
- the types of variables for functions will be indicated as subscripts next to the lambdas that introduce them
- words used in a technical sense for the first time will be in SMALL CAPS (I'll adhere to this practice consistently even at the cost of making a few pages look like comments on a blog post, full of DERANGED ANGRY YELLING)
- emphasis is indicated with **boldface**
- outside of examples, the object language is in *italics*
- 'iff' abbreviates 'if and only if'

[1] The relevant steps are the move from the denotation of a pronoun-like unpronounced element in the syntactic tree, $[\![x_1]\!]$, to the logical variable x_1 (really, there should be an assignment function that maps from one to the other); and the application of a Predicate Abstraction Rule like Heim and Kratzer's to interpret the floating object-language lambda.

The conventions about variable names are as follows:

- P, Q for properties of individuals or events
- R for relations
- G for gradable degree predicates, any type with both a d and an e in it: $\langle e, d\rangle$, $\langle e, dt\rangle$, $\langle d, et\rangle$
- D for properties of degrees, type $\langle d, t\rangle$
- p, q for propositions, type $\langle s, t\rangle$
- $f, g, \ldots$ for other functional types
- $e, e', \ldots$ for events, type v (not s, as is common; I'll reserve that for worlds)
- $d, d', \ldots$ for degrees, type d
- $w, w', \ldots$ for possible worlds, type s

1.4 WHAT, IF ANYTHING, IS MODIFICATION?

With that out of the way, we can return to the substantive question at hand: what precisely is modification? Does it constitute a single grammatical phenomenon?

The easiest answer to give – and, after some reflection, simultaneously the more obvious and more surprising one – is no. We think of the grammar largely in terms of predicates and their arguments. 'Modifier' is simply a term for linguistic expressions that don't fit neatly into either conceptual box. If this is right, construing modification as a unified phenomenon is doubly mistaken. First, it's uselessly broad. Writing a book about modification would be like writing a book about arguments: essentially an impossibility. One can talk coherently of argument **structure**, of course, but this isn't evidence that all expressions that happen to be arguments have something essential in common. Second, on this understanding, modifiers would be the complement of a natural class – that is, a meaningless set defined in reverse, like non-Bolivian non-dermatologists. If you had encountered a class like this in a phonology problem set, you would be justified in suspecting you had taken a wrong turn somewhere.

But there is another way of looking at the question, even if it's harder to perceive. One place to start is consulting one's intuitions about the use of the term, however inconsistent or precarious they may be. An adjective is a modifier, except for when it isn't. An adverb is almost always a modifier, though adverbs might really be just glorified adjectives in any case. A prepositional phrase is sometimes a modifier and sometimes it isn't, depending perhaps on whether it's an adjunct.

A noun or noun phrase isn't a modifier, but what about in, say, *died last night*? Functional elements like tense morphemes, modal auxiliaries, and most determiners clearly aren't modifiers. Clauses are modifiers in various adjoined positions, but not elsewhere.

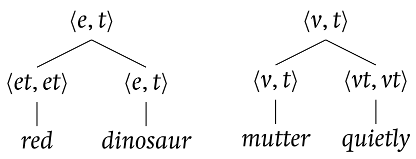
In this meandering litany, one can discern something about the nature of the conceptual struggle. The categories most readily at hand are syntactic, but we seem to be groping for something semantic. The references to syntactic category seem to be a clumsy proxy for an adequate language to talk about the lexical semantics of expressions, one that might ultimately express an intuition about their distribution too. Clearly, all this will need to be firmed up to make progress on the broader question.

In one respect, that can be done immediately. There is behind the whole thing a kind of equivocation that needs to be corrected. It's between two ways of characterizing a phrase. There is a difference between labels for the **internal** characteristics of phrases and for the **external** role they play in the constructions they enter into. Terms like 'subject', 'complement', 'adjunct', 'resultative', or 'purpose clause' all unambiguously characterize constituents by the role they play as part of larger ones, their external role. Terms like 'noun' unambiguously name lexical categories, and no one is inclined to use them to mean, say, 'complement to a verb' (setting aside sloppy talk of 'acting as a noun' in first-semester undergraduate assignments and prescriptivist harangues). They're characterizations of an internal property of a word and of the phrases it heads, not of their relation to larger expressions. The term 'modifier' is uncomfortably perched astride this fence. It characterizes both a family of (internal) lexical semantic characteristics and a family of (external) distributional ones. That, I think, may account for some of the conceptual muddle.

The internal sense of 'modifier', then, to a very crude first approximation, may amount to just this: you're a modifier if you're an adjective or an adverb. That probably makes you pretty good at gradability. The external sense of 'modifier' has to do with crosscategorial parallels in the role an expression plays. You're a modifier if you're adjoined to something that you're not a semantic argument to. You very well might have a semantics that can be expressed with *and*: a *red dinosaur* is red and a dinosaur.

Obviously, the distinction doesn't instantly cut through the haze. However, it is useful because, for the external sense, it's possible to provide a straightforward and rigorous (if imperfect) definition of modification in terms of semantic type. As we'll see in subsequent chapters, on one classical way of thinking, a modifier is any expression that maps a type to the same type: that is, anything whose denotation is type $\langle \tau, \tau \rangle$, where τ is a type. When τ is a predicate type, anything with this kind of meaning is called a PREDICATE MODIFIER. An example:

(8)



The meaning of *red* is a function that maps dinosaurs to red dinosaurs, the meaning of *quietly* is a function from mutterings to quiet mutterings:

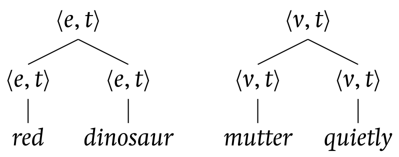
(9) a. $[\![\textit{red dinosaur}]\!] = [\![\textit{red}]\!] ([\![\textit{dinosaur}]\!])$

b. $[\![\textit{mutter quietly}]\!] = [\![\textit{quietly}]\!] ([\![\textit{mutter}]\!])$

All the elements combine by function application.

Much of the time, there's an even simpler option – indeed, one that is often preferable, as we'll see. That's INTERSECTIVE MODIFICATION, in which an element denotes a property (of individuals or events or anything, in principle) and combines with something else that denotes the same kind of property:

(10)



The idea here is that *red dinosaur* should denote a property of individuals that are red and dinosaurs, and *mutter quietly* a property of events that are mutterings and quiet. This is 'intersective' in the sense that, in set-theoretic terms, it involves intersecting the set of red things with the set of dinosaurs (or the set of muttering events with the set of quiet events). Function application can't achieve this for (10), of course. The types don't fit. But a rule of intersective interpretation such as Heim and Kratzer (1998)'s Predicate Modification (suitably generalized to events) can. It allows the modifiers in (9) to combine as in (11):

(11) a. $[\![\textit{red dinosaur}]\!] = \lambda x \,.\, [\![\textit{red}]\!] (x) \wedge [\![\textit{dinosaur}]\!] (x)$

b. $[\![\textit{mutter quietly}]\!] = \lambda e \,.\, [\![\textit{quietly}]\!] (e) \wedge [\![\textit{mutter}]\!] (e)$