Question Embedding
and the Semantics of Answers

by

Benjamin Ross George

2011

For every student

who has ever heard the lie

that the ability to maintain a tidy notebook

is an indispensable prerequisite

for academic success.

Note on this Web Version, 6 July 2011

This document has been modified from the dissertation that I filed: I have adjusted the formatting and omitted some of the front matter.

Corrections, comments, questions, and suggestions are welcome, and should be directed to `seeking.unicorn@ucla.edu`.

dissertation without the help of Ivano Caponigro and Gennaro Chierchia (Italian), Benjamin Spector (French), Ed Keenan (Malagasy), and Lauren Winans and Maged Tawfelis (Egyptian Arabic). I am also indebted to Signe Söderlund for providing some Swedish data that turned out not to be very interesting, and to Daniel Büring and Thomas Graf for some German data that turned out to be *too* interesting.

In addition to the above contributions to the theoretical and empirical content of this document, I am grateful to Sarah Bergstrom for suggesting the quote at the beginning of Chapter 2, Chaos Golubitsky for helping me figure out how to word the dedication, Kate Duffy for an especially helpful editorial suggestion about the acknowledgments, and Robyn Orfitelli for her assistance in picking the names in the examples.

In the course of my work on this dissertation, a number of people have graciously tolerated me when I was not at my most composed or even-tempered. Special recognition in this area goes to Jessica Rett, Anya Essiounina, Melanie Levin, Chelsea Rosenthal, my long-suffering flatmates (Lisa Wu and Erin Gray) and office-mates (Heather Burnett, Isabelle Charnavel, and Melanie Bervoets), and the staff at Nature's Way Café (10917 Lindbrook Drive), Native Foods (1110 1/2 Gayley Avenue), Peet's Coffee and Tea (1154 Westwood Boulevard), Espresso Profeta (1129 Glendon Avenue), and Boba Loca (10946 Weyburn Avenue) in Westwood. The last three groups are thanked in particular for often playing music that was not too loud or too cognitively disruptive.

Misery loves company, and I am grateful to many fellow-sufferers, near and far, for making this process more bearable with their words of sympathy, their willingness to compare experiences, and their reminders that I was not alone. I am, in this area, especially indebted to Ariel Diertani, Robyn Orfitelli, and Kristine Yu.

My visit to Paris in the spring of 2010, funded by a Partner University Fund Grant from the FACE foundation, made possible many valuable conversations about questions in particular and language and meaning in general. I am grateful to Dominique Sportiche for suggesting and arranging this visit, and to Giorgio Magri, Vincent Homer, Emmanuel Chemla, Benjamin Spector, and Philippe Schlenker for their assistance in navigating the academic bureaucracy and various aspects of everyday life when I got there.

My move from the Computer and Information Science Department at the Uni-

versity of Pennsylvania to the Linguistics Department at UCLA involved a certain amount of anxiety, uncertainty, and disorganization on my part, and I am grateful to various people whose tolerance, flexibility, support, or insight made this change of course possible, and helped me to realize that it was the right choice. Without attempting any classification of the nature or significance of the various contributions, I would, in this connection, like to recognize Sharon Rose Alterman, Sam Crane, Aaron Dinkin, Mike Felker, Ted Fernald, Asia Furmanska, Melanie Goetz, Jack Hoeksema, Aravind Joshi, Kyra Jucovy, Ed Keenan, Tony Kroch, Steven Lindell, Sarah Alys Lindholm, Anoop Mahajan, Mitch Marcus, Donna Jo Napoli, Sven Olsen, Elliot Reed, Maribel Romero, Chelsea Rosenthal, Rachel Sapiro, Ken Shan, George Weaver, and Scott Weinstein.

Many of the people mentioned under other headings have made other salient contributions to my sanity, but there are a few people whose contributions didn't fit in neatly elsewhere. An exhaustive, or even fair, list of these would not be possible, but Jillian Waldman, Ben Newman, Byron Ahn, Emily Higgins, J'aime Roemer, and Jenn Fischer spring to mind as people who must be mentioned. Almost any of my not-otherwise-mentioned contemporaries in the graduate program in Linguistics might reasonably be added to this list.

I am also grateful for the generosity of those who have provided financial support for my studies and research, and especially the UCLA Linguistics Department and the UCLA Graduate Division. Both were incredibly important to the funding of my years as a graduate student, and, in particular, the writing of this document was supported by a Dissertation Year Fellowship from the latter.

The task of properly cataloging the contributions of my brother, my parents, and my late grandparents is daunting even by the standards of acknowledgments sections: they provided the love and support that made all of this possible. Without attempting to be thorough, I must mention two things that are especially relevant. First, my tendency to fixate on the technical details of seemingly insignificant things was essential to the completion of this dissertation, and I know that not every family would have appreciated, and often encouraged, it in the way that mine did. Second, I am grateful to my parents and brother for their acceptance of and support for my unexpected, and not entirely welcome, decision to move to Los Angeles.

Finally, I am indebted beyond my powers of description to Chelsea Rosenthal,

for undertaking the task of always being there for me while on the other side of a continent.

Abstract of the Dissertation

Question Embedding
and the Semantics of Answers

by

Benjamin Ross George

Doctor of Philosophy in Linguistics

University of California, Los Angeles, 2011

This dissertation is concerned with the semantics of questions and question embedding. Its investigative strategy is to set out the simplest theory (or, rather, the simplest one that I could think of) that accounts for what I take to be the core facts in the semantics of embedded *'wh'*-questions, and then to prod at this theory, and subject it to various stresses, in order to see where it breaks, and where it doesn't break, and especially *how* it breaks.

The theory that I take as a point of departure is presented in Chapter 2. In its formal details, it combines ideas found in a number of familiar theories. Its implementation of strong exhaustivity and its treatment of *'wh'*-phrases as λ-abstraction operators are heavily influenced by Groenendijk and Stokhof (1984) (and, as in Groenendijk and Stokhof (1984), this latter point reflects a hope that the semantics of relative clauses and of *'wh'*-questions can be situated in a general theory of *'wh'*-clauses). From Hamblin (1973), it takes its approach to mention-some answers, and also the idea that the semantic contribution of a *'wh'*-question is a set of possible answers for that question, the idea that this answer set is formed by existential quantification over possible values to 'fill in' for the *'wh'*-phrases, and the idea that the answer set should include both true and untrue answers. Its treatment of embedding is based in spirit, bur not in its technical details, on the approach to strong exhaustivity with non-veridical embedders in Egré and Spector (2007). In this baseline theory, the semantic contribution of a *'wh'*-question is taken to always be a set of possible answers, but questions are ambiguous between a mention-some reading (on which they contribute the set of all propositions identifying a value (or values) that could stand in for the *'wh'*-phrase(s)) and a strongly exhaustive reading (on which they contribute the set of all propositions identifying exactly which values could stand in for the *'wh'*-phrase(s)). On this account, when a question is embedded under a propositional attitude like *'know'*, the resulting sentence is true iff there exists an answer of which the attitude is true.

After developing this baseline theory, I begin to explore possible problems for it. In Chapter 3, I argue that, if domain restrictions and related effects are taken

seriously, the baseline theory can handle some well-known data that have historically been used to argue for kinds of answers beyond mention-some and strongly exhaustive answers. I also review some smaller complications that appear to require modest revisions.

In Chapter 4, I turn to a basic problem with my account of embedding, and for virtually all available accounts of embedding under *'know'* and similar embedders. I argue that contrary to standard assumptions, the truth-conditions of sentences embedding questions under *'know'* cannot be expressed solely in terms of which propositions the subject knows, and that the same issue arises for *'forget'*, and perhaps for other embedders as well. I then sketch a revised semantics for these kinds of predicates that offers a richer, but still constrained, picture of their embedding semantics.

The remaining chapters are devoted to various loose ends and unresolved issues. In Chapter 5, I briefly explore the applications of the baseline theory to other kinds of embedded questions. I conclude that it should be relatively easy to accommodate *'yes'*/*'no'* questions and (at least) simple concealed questions within the system, but that alternative questions will require some additional work. In Chapter 6 I revisit the issue of mention-some readings, and explore the strengths and weaknesses of various ways of explaining these readings away or limiting their availability. I conclude the dissertation in Chapter 7 by taking stock of the main results of previous chapters and briefly surveying some of the major issues not addressed in the preceding chapters.

CHAPTER 1

What is a Semantics of Questions and Answers?

> I promise nothing complete; because any human thing supposed to be
> complete must for that very reason infallibly be faulty.
>
> Melville (1851)

1.1   Data and Meaning in Semantic Theory

In the study of compositional semantics, our primary data are facts about the use
and communicative value of more or less 'free-standing' constituents in conversation.
Take for example (1):

(1)   Anne sought ten unicorns, and found all of them, except for one.

The act of uttering (1) will typically do a few different things. It will commit the
speaker to the position that the present state of things is such that Anne looked
for ten unicorns, nine of which she found. It will establish a discourse referent
associated with the unicorn that was not found. It may set up Anne as the 'center'
of the discourse. It will trigger an inference that the speaker thinks the audience
has some reason to care about Anne seeking or finding unicorns. If situated in
a longer narrative, it will probably advance the time frame of the story to some
point in time no earlier than the point at which the ninth unicorn is found. These
sorts of observations are the kinds of facts to which a semantic theory is ultimately
accountable.

To be successful, a theory of semantics should attribute to the sentence (1) a
semantic meaning from which all these effects can be derived. That does not mean
that all these effects need to be written into the meaning directly: the will in many
cases be derivable from the meaning conventions of discourse structure, pragmatics,

and so on, and many of these principles may be language-specific or particular to the cultural setting of the conversation. What a theory of semantics needs to do is make sure that the semantic contribution of the sentence gives the pragmatics enough to work with. Minimally, it needs to assign meanings to sentences in a way that ensures that, when two sentences have different effects in conversation, they also have different semantic meanings (unless there is good reason to attribute the different effects to some non-semantic difference.[1]). If the theory is to be of much interest, if should also assign meanings in a way that makes it feasible to describe how the pragmatics, and the rules of discourse read the conversational behavior of the sentence off of its meaning.

However, the various conversational/communicative effects of (1) are not the only data that bear on the semantics of (1), at least not if we are invested in having anything like a compositional semantics. We also have the conversational effects of sentences like (2) to consider:

(2)   Rupert told Red that Anne sought ten unicorns, and found all of them, except for one.

(2) has truth conditions, can advance a narrative, affects the center of discourse, and so on. These kinds of observations do not just bear on the meaning of (2), however: they also bear on the meaning of (1). If, for example, we note a difference between the truth-conditions of (2) and (3), this difference should naturally be attributed to a difference between the meaning of (1) and the meaning of (4):

(3)   Rupert told Red that Liam found the missing book.

(4)   Liam found the missing book.

The step of using observations about (2) to explore the semantics of (1) rests on a number of nontrivial assumptions. The most important of these is the assumption

---

[1]For example, we might wish to say that the pragmatics, but not the semantics, is sensitive to the amount of time or effort involved in producing or interpreting a sentence, or to the risk of error in interpretation. I do not mean to trivialize these important issues, but I also want to make it clear that the existence of these effects shouldn't excuse semanticists from their obligation to make sure that the semantic theory gives the pragmatics enough to work with.

that the constituent *'Anne sought ten unicorns, and found all of them, except for one'* that comes after *'that'* in (2) shares the meaning of the free-standing clause (1). Nothing forces this upon us. We could say that the sentence (1) is not embeddable, and that a parallel but distinct syntactic machinery builds the embedded clause in (2), which in turn is not permitted to be a free-standing sentence. If we take this step, then in principle the two could have very different semantic properties.

The main things standing in the way of such a move are its counterintuitiveness and the demands of parsimony. We must begin somewhere, and the identification of (1) with the embedded clause in (2) is a perfectly appropriate (and rarely questioned) starting assumption. We should, in principle, remain open to the possibility that the data may force us to abandon or weaken this assumption, but, for the purposes of this dissertation, I have no desire to reexamine it.

We have, then, two general kinds of ways of probing the semantics of clauses like (1). We have facts about the conversational effects of the clauses themselves, and facts about the conversational use of larger sentences in which those clauses are embedded. For sentences used declaratively, the most important facts to account for relate to truth-conditions.

When exploring the semantics of questions, we likewise have two kinds of data at our disposal. For a question like (5), we have judgments about (5) as a free-standing question, but we also have judgments about embedding sentences like (6) and (7):

(5)   Where is the book?

(6)   Liam knows where the book is.

(7)   Rupert asked where the book was.

The conversation-level effect of a question like (5) may have many facets. It may have presuppositions or implicatures. It may depend anaphorically on prior discourse, or deictically on the context. It may take part in various larger conversational structures. But, as truth-conditions have earned their role as the most important observable facet of the semantics of declarative sentences, so, for the semantics of questions, the primary concern is with answerhood conditions. The most

obvious use of a question like (5) is to seek an answer that identifies where the book is. In reply to (5), (8) is a good answer (if it is true). (9) is usually not as satisfying of an answer as (8). (10) seems to not quite be an answer, but is a reasonable reply if that's all the information one has. (11) is a cooperative non-answer. (12) is a slightly obfuscatory way of implicating an answer. (13) is a totally uncooperative response.

(8)   The book is on the shelves next to Rupert's desk at the leftmost edge of the third shelf from the top.

(9)   The book is on Rupert's desk or on the shelves next to it.

(10)   The book isn't on Rupert's desk.

(11)   Sorry, I don't have the faintest idea where the book is.

(12)   Have a look at the third shelf from the top on the shelves by Rupert's desk.

(13)   When the first platypus specimen reached England, scholars there expressed suspicion that it might be a hoax.

A complete theory of the semantics and pragmatics of questions should give us an account of why (8) is generally a satisfying answer to (5), and (13) is a completely inappropriate reply, and (9), (10), (11), and (12) occupy various positions between these extremes.[2] The best division of labor between the compositional semantics and other modules of the theory is not immediately clear, of course but, the semantics must at least give the other components enough of a picture of the question-meaning to 'hang on to', and most likely it must provide some semantic object that bears a close connection to the (intensional) property of being a place where the book in question is located.

In addition to the answerhood conditions of unembedded questions, a semantics of questions should also account for the semantic properties of sentences containing

---

[2]I do not promise a complete theory, and many of these examples will not be directly addressed.

embedded questions. For declarative sentences like (6) and (7), the most important data will be those pertaining to truth conditions. If Liam knows the proposition expressed by (8) (and knows that the book occupies a single location), then that suffices to make (6) true. If, on the other hand, he only knows the proposition expressed by (10), we are inclined to judge (6) untrue. If Liam does not know any proposition about the location of the book, then we confidently judge (6) untrue. A theory of the semantics of questions, of the verb *'know'*, and of question embedding should account for these facts (although there are many different imaginable ways that we might handle the division of labor between the meaning of the question, the meaning of the verb, and the combinatorics of the grammar).

Analogously with the case of declarative sentences, the claim in the previous paragraph rests on an identification of embedded clauses like those in (6) and (7) with unembedded questions like (5). For English, at least, this appears to be more of a leap than the corresponding case for declarative sentences and their corresponding embedded clauses, because there are conspicuous differences in overt structure between the embedded and unembedded clauses under consideration.[3] In spite of these differences, the association of embedded questions with unembedded questions is a natural starting point, because of obvious connections between the truth-conditions of embedded question sentences and facts about unembedded questions. A state of affairs in which Rupert has (deliberately and with understanding) uttered the question in (5) makes (7) true, and the sentence if (6) is true then Liam must know a proposition that, expressed with a sentence, would constitute a satisfying answer to (5). These connections suggest that the embedded questions in these sentences are contributing, if not the exact meaning of the question (5), then at least the meaning

---

[3]In English, we find some differences between embedded and unembedded forms with every kind of question, including *'yes'*/*'no'*-questions (e.g. (14)), *'wh'*-questions (e.g. (15)), and alternative questions (e.g. (16)).

(14)   a.   Does Alex remember the trip to the zoo?
        b.   Rupert won't tell me <u>whether Alex remembers the trip to the zoo</u>.

(15)   a.   Who did William kill?
        b.   Anne figured out <u>who William killed</u>.

(16)   a.   Does Anne love Liam, or Alex?
        b.   I don't know <u>whether Anne loves Liam or Alex</u>.

of some important shared constituent.

This dissertation will be concerned primarily with the study of question embedding as a tool for exploring the semantics of questions and question embedders, and the semantic aspects of the question-answer relationship. The emphasis on question embedding is motivated by a desire to limit the intrusion of the pragmatics into the facts observed, in order to get a better handle on the semantic issues. At the unembedded level, the semantic and pragmatic aspects of answerhood conditions are very difficult to untangle directly. In the case of declarative sentences, we have some familiar tools to help us get a handle on when a sentence is false and when it is literally true but unacceptably misleading or otherwise uncooperative. Assessing a corresponding difference between semantically specified answers and 'good enough' cooperative substitutes for answers (like, perhaps, (9), (11), and (12) as replies to (5)) appears more difficult. To the extent that answers are involved in the semantics of question-embedding declarative sentences, we can use those sentences to begin to understand the semantics of answers using truth-conditional judgments.

1.2 Two Kinds of Question-Embedders

Before going further, it will be helpful to distinguish between two major classes of question-embedders: one of these is exemplified by *'ask'*, and the other by *'know'*. The presumption is that the non-entity argument of *'ask'* is whatever sort of semantic object is contributed by a question. We understand *'ask'* as expressing a relation between entities and question-meanings, and an attribution of asking is true if and only if the entity has made an inquiry that is (in some suitable sense) synonymous with the embedded question: if we had perfect records of Anne's life, we could assess the truth of (17) by playing them back for the time period under discussion, and checking to see if she ever expressed a question equivalent to[4] *'Who murdered Jonathan?'*

(17)   Anne asked who murdered Jonathan.

---

[4]The exact notion of equivalence that is appropriate here is not clear, but is not immediately needed to cultivate the intuition of interest. I will assume that the notion of equivalence involved is identity of answerhood conditions, as formalized in Chapter 2, but I will not dwell on the issue, since the *'ask'* class of predicates is not my main concern.

Embedders like *'know'* behave rather differently from those like *'ask'*: they take proposition-expressing clauses as well as questions as arguments, and their question-oriented use is understood as related to their propositional use. Typically, accounts of question embedding handle predicates like *'know'* by reduction from the question-oriented use to the propositional use:[5] they define the truth-conditions of the question-oriented use (seen in (18)) in terms of the relation between entities and propositions that *'know'* expresses in sentences where it embed any *'that'*-clause (as in (19)), so that, if, say, Andrew murdered Jonathan, (18) is true iff (20) is:

(18)   Anne knows who murdered Jonathan.

(19)   Red knows that there will be a full moon Thursday night.

(20)   Anne knows that Andrew murdered Jonathan.

The details of how exactly we give the truth-conditions of (18) vary between theories, but, to pick one example, we might say that (18) is true if and only if Anne knows at least one proposition that counts as an answer to the embedded question *'who murdered Jonathan'*. This would then, of course, leave us with the problem of spelling out which props count as answers for these purposes.

The idea that we should distinguish between *'know'*-like embedders and *'ask'*-like embedders is appealing, but the exact boundary is disputed. One controversial case is *'agree'*: Groenendijk and Stokhof (1984) are committed to placing it in the *'ask'* class (in their terms, it is one of the 'intensional' predicates) rather than the *'know'* class (an 'extensional' predicate),[6] while Lahiri (2002) places *'agree'* in the *'know'* class ('responsive' predicates) rather than the *'ask'* class ('rogative' predicates).[7]

---

[5]The account of *'know'* in Spector (2005) is an important exception, which will be discussed in more detail in Chapter 4.

[6]Groenendijk and Stokhof's reason for leaving *'agree'* out of the *'know'* class is connected with their decision to characterize the *'know'* class as the class of embedders that are concerned only with the complete true answer to a question, which must exclude *'agree'* since it is non-veridical, even when embedding questions. Beck and Rullmann (1999) and Lahiri (2002) discuss some related issues with *'agree'*.

[7]The distinction between Groenendijk and Stokhof's definition of 'extensional' and Lahiri's and my notion of 'responsive' embedders, is, roughly, that the former are the subclass of the latter that are concerned only with true answers. Groenendijk and Stokhof (1984) do not devote much energy to classifying their 'intensional' embedders, and don't seem to recognize the responsive embedders

I will adopt the terminology and (roughly) the boundary of Lahiri (2002) because I think there is an intuition that the propositional and question-oriented uses of predicates like *'be certain'* and *'agree'* are connected in a way that deserves some kind of account, and I think that treating this connection as different from the connection between the two uses of *'know'* ignores some obvious parallelism. I will call 'responsive' any question embedder that also serves as a propositional attitude (that is, any one that can embed a 'indirect statement'-type propositional clause – in English, an indicative *'that'*-clause) and I will call other question embedders 'rogative'. Thus, the fact that *'know'* is responsive is shown by the acceptability of (22-a) (and its connection with (22-b)), and *'be certain'* is likewise responsive because we accept (23-a) as well as (23-b).[8] In contrast, *'ask'* is a rogative embedder, as is evidenced by the ungrammaticality of (24-a). (Note, though, that *'ask'* can embed 'indirect commands', as in (24-c).)

(22) a. Rupert knows that Liam found the book.
      b. Rupert knows who found the book.

(23) a. Alex is certain that William is the spy.
      b. Alex is certain of who is the spy.

(24) a. *Anne asked that Red found Janna's files.
      b. Anne asked what Red found.
      c. Anne asked that Red find Janna's files.

This classification, like most such divisions, has a certain number of ambiguous or borderline cases. In particular, the classification depends on when we recognize two

as a natural class.

[8]The use of 'responsive' adopted here is slightly different, and a bit fuzzier, than that found in Lahiri (2002). For Lahiri (2002), the distinguishing feature of the *'know'* class is the (alleged) paraphrasability of (21-a) by (21-b) (cf. example (1) of Lahiri's Chapter 3).

(21) a. Red knows what Anya bought.
      b. Red knows some proposition p that answers the question *'What did Anya buy?'*

I do not adopt Lahiri's definition because I consider the exact nature of the link between the propositional and question-oriented uses to be a thorny empirical problem, for reasons discussed in later chapters, especially Chapter 4.

uses as instances of the same predicate in a semantically relevant sense. For example, *'be certain'* requires the particle *'of'* in (23-b) but not in (23-a), which might lead us to conclude that there is an important difference between these, or that it is *of* that is the real question-embedder in (23-b). In most cases involving such prepositional particles, I will assume that the particle is a syntactic idiosyncrasy and that the verb or other open-class head attached to the particle is doing the real semantic work, so question-embedding and propositional uses of *'be certain'* will be treated as a pair to be explained, as will these two uses of *'agree'*.

A more interesting illustration of the difficulty of deciding when to treat two similar uses of the same word-shape as instances of the same embedder arises with *'wonder'*.[9] A somewhat archaic use of *'wonder'* can embed a *'that'*-clause and means roughly *'be amazed'*,[10] as illustrated by (25) and (26), reported in the Oxford English Dictionary:

(25)   I wonder that you will still be talking, signior Benedicke, no body markes you. (Shakespeare (c. 1600))

(26)   We wonder the parties did not take a patent for the discovery. (Greener (1846))

On the other hand, *'wonder'* with an embedded question means something like *'want to know'* or *'ponder'*:

(27)   I wonder why Cordelia is still talking.

These two uses of *'wonder'* are, of course, descended from the same historical source (for some discussion, see the Oxford English Dictionary), and propositional and question-oriented uses of *'wonder'* appear to have coexisted in the language for some time[11], although it is of course hard to tell what subtle semantic changes

---

[9]The issues with *'wonder'* described here are mainly associated with an earlier stage of the development of English. It appears that similar issues arise for at least some German speakers with *'wundern'*, but I have not yet explored the German data in detail.

[10]Compare the noun *'wonder'* in standard English, which still means something close to *'amazement'*.

[11]In the Oxford English Dictionary, the first quotation for the question-embedding use of *'wonder'* is from 1297, and the last for *'wonder that'* is in 1885.

the two uses may have undergone during this time. I will tentatively adopt the position that, at least on the meanings described informally above, they are not instances of the same *'wonder'* (at least not in the same way that the propositional and question-embedding uses of *'know'* are instances of the same predicate) and that the connection between them is therefore not something that a synchronic theory of question embedding needs to account for. There are a few reasons for treating the two *'wonder'*s as distinct for purposes of the semantics. First, appealing to the intuitions of the theorist, the states of being amazed at a fact (expressed by *'wonder that'*) and of wanting to know the answer to a question (expressed by question-embedding *'wonder'*) don't have any strong natural connection – at least not one that seems likely to admit any logical characterization. There is nothing like the intuitive link seen in the case of *'know'*. For example, being related to a question by *'know'* seems to unavoidably involve some kind of propositional knowledge, but there is no comparable link that I am aware of between *'wanting to know'* and *'being amazed'*, and it is hard to imagine that learning the meaning of one use of *'wonder'* would be much help in understanding the other use. Another issue is that some English speakers report that they are relatively tolerant of the *'wonder that'* usage, and these typically reject coordinated complement sentences that would ask the same instance of *'wonder'* to do both jobs. Even for people who find examples like (28) acceptable, (29) is unacceptable, in contrast with (30):[12]

(28)   I wonder that William disposed of the evidence.

(29)   *Anne wonders that William killed Holden, and what he did with the body.

(30)   Anne knows that William killed Holden, and what he did with the body.

In light of these considerations, I will follow standard practice and classify *'wonder'* as rogative. In classifying *'wonder'* as rogative and *'know'* as responsive, I am asserting that the explanation of the connection between the two *'wonder'*s is a job for historical semantics, but the connection between the two *'know'*s is something

---

[12]The reader is cautioned that my exploration of this issue has not been especially thorough, or at all statistically responsible, and that most of the speakers who describe (28) as acceptable do still characterize it as at least somewhat archaic-sounding.

that a complete model of the synchronic semantics should address. Of course, the possibility of homophony of question-embedded and propositional attitudes means that formally the theory can dispose of any ill-behaved responsive predicate by re-classifying it as a case of homophony or something similar, but some such excuses will be more credible than others. I find the claim that the two *'wonder'*s are, in the relevant sense, different words, palatable, but a corresponding claim for *'know'* is much harder to accept. I hope that the reader will share these impressions.

Going forward, I will focus almost exclusively on responsive embedders, because the semantics of responsive embedders seems to me more likely to shed light on the semantics of the question-answer relationship. The rogative embedder *'ask'* builds propositions about questions, not just about their answers: to assess the truth of a simple *'ask'* sentence like (24-b), we look at which questions the subject has asked, and see if the question in the object position (or perhaps some suitable equivalent of that question) is among them. Bringing in the notion of 'answer' doesn't seem to shed much further light on this. This does not, of course, mean that *'ask'* lacks semantic interest. The relationship between question-embedding uses (like (24-b)) and 'indirect command' uses (like (24-c)) deserves exploration, and test of interchange-ability under *'ask'* can serve as an interesting test to help study fine-grained issues related to when two questions are semantically equivalent. These and other issues deserve further attention, but I wish to focus on the question-answer relationship, and, for this, responsive predicates seem like the more natural place to direct my efforts.

1.3 Answers in the Semantics of Responsive Embedders

I have suggested that responsive embedders interact with a question through its answer or answers, and most theories of the semantics of question embedding follow this intuition. Although there is this much similarity, there is considerable variation in which propositions are considered as answers, and in how they are used in the semantics of embedding. Typical approaches handle responsive embedders more-or-less 'reductively' – we test whether a responsive predicate is true of a question by looking at which propositions it is true of, and in particular at how those propositions

11

answer the question.[13] Thus, for example, if the unicorn is in fact hiding behind the couch, the generalization is that (31-a) is true if and only if (31-b) is:

(31)   a.   Alex knows where the unicorn is hiding.
            b.   Alex knows that the unicorn is hiding behind the couch.

The key feature of the reductive picture is that knowing which proposition or propositions answer the question, and knowing the propositional extension of the responsive predicate, is supposed to be enough to evaluate the truth of the question-embedding sentence.

Some of these reductive accounts (like that of Groenendijk and Stokhof (1984), and, at least in its aspirations, that of Egré and Spector (2007)) are 'uniform' – they use the same notion or notions of 'answer' for all the responsive predicates they address, and, for all these predicate, the question-oriented case is reduced to the propositional case by the same rule or rules. For example, in handling what they call 'extensional' embedders, Groenendijk and Stokhof (1984) employ the uniform rule that the truth of a predicate applied to a question is tested in terms of the truth of that predicate applied to the (unique) answer to the question.

Restricting attention to 'wh'-questions, Groenendijk and Stokhof (1984) say that the only answer of interest for these embedding cases is the 'strongly exhaustive' answer. If a 'wh'-question is intuitively associated with a property or relation, then the strongly exhaustive answer to the question is a proposition that identifies exactly what the extension of that relation is – it is the proposition that identifies all the values of suitable type that could 'fill in for' the 'wh'-phrases, and also includes the information that the collection it identifies is complete. For example, if Anne, Red, and Alex were the only students at the party, (32-b) is the unique strongly exhaustive answer to (32-a):

(32)   a.   Which students were at the party?
            b.   The (only) students at the part were Anne, Red, and Alex.

---

[13]Historical exceptions to this generalization, and empirical problems for it, are discussed in detail in Chapter 4.

Many accounts consider multiple types of answer, and some of them do this non-uniformly. For example, Beck and Rullmann (1999) consider three kinds of answers. Strongly exhaustive answers (as in (32-b) above), 'weakly exhaustive' answers, and 'mention-some' answers. A weakly exhaustive answer is all the information in a strongly exhaustive answer except the 'that's all' component: it is the proposition that, for everything with the property picked out by the question (or every ordered tuple that stands in the relation picked out by the question), attributes that property (or relation) to that thing (or tuple), but doesn't exclude the possibility of there being more such things (or tuples). For (32-a), in the circumstances described, the unique weakly exhaustive answer is (33-a). A mention-some answer is an answer that identifies at least one (in some formalizations, exactly one) thing with the property picked out by the question – in the situation we have been discussing, (33-b) and (33-c) are among the mention-some answers to (32-a), and (on most formalizations) (33-d) and (33-a) are as well.

(33)   a.   Anne, Red, and Alex are students who were at the party.
       b.   Anne is a student who was at the party.
       c.   Red is a student who was at the party.
       d.   Red and Alex are students who were at the party.

Some authors, in recognizing multiple sorts of answers, admit some amount of non-uniformity in which answers are relevant to evaluating which predicates – in particular, *'surprise'* is supposed to select weak exhaustivity. To pick one concrete example, Guerzoni and Sharvit (2007) claim that (34) is true if and only if Rupert was surprised by the weakly exhaustive answer to (32-a),[14] but (35) is ambiguous between a reading where it asserts that Rupert knows the strongly exhaustive answer, and one where he need only know the weakly exhaustive answer.

(34)   Rupert was surprised by which students were at the party.

---

[14]Note that this doesn't mean that the sentence is necessarily untrue if Rupert was surprised by the strongly exhaustive answer: only that it is untrue if Rupert was surprised only by the strongly exhaustive answer and not the weakly exhaustive one. If Rupert became aware of both the weakly and strongly exhaustive answers, and was surprised by the weakly exhaustive answer, then he might very well have been surprised by the strongly exhaustive answer as well, but his surprise at the strongly exhaustive answer is, according to, for example, Guerzoni and Sharvit (2007), not directly relevant to the truth-conditions of (34).

(35)   Rupert knows which students were at the party.

Another dimension of non-uniformity in the ways that different responsive embedders handle questions, advocated by Lahiri (2002), is sensitivity to truth. For example the truth-conditions of (36) seem to include a requirement that the answer that Rupert told Liam is true, while (37) is not supposed to place any requirements on the accuracy of Rupert's beliefs. Likewise, as observed by Weatherson (2009), (38) denies that Liam is sure of any answer – not just that he is sure of a true one.

(36)   Rupert told Liam which students were at the party.

(37)   Rupert is certain of which students were at the party.

(38)   Liam isn't sure which students were at the party.

When talking about untrue answers, there does not seem to be any useful notion of *the* unique answer.[15] Our notions of unique answer were notions of (strongly or weakly) exhaustive answers, and both of these were defined in terms of picking out the maximal set of true instances. There is no interesting maximally false answer: in (37) there may only be one way for Rupert to be certain *and right*, but Rupert can be certain *and wrong* by adopting any of an infinite assortment of mutually incompatible false beliefs. In light of this, I think we will be better served by a theory that deals in truth-independent notions of *an* answer.

The notion of an untrue mention-some answer is straightforward enough – it is any proposition that identifies some value (or values) as having the property described. For example, if (32-b) is true, then (39) is a false mention-some answer to (32-a):

(39)   Billy is a student who was at the party.

The most useful notion of an untrue strongly exhaustive answer (used, for example, by Egré and Spector (2007) and Sharvit (2002)) is any proposition with the right 'shape' to be a strongly exhaustive answer – or, alternatively, any proposition that

---

[15]But see Lahiri (2002) for a different perspective on this point.

14

could be a strongly exhaustive answer in some possible world. So, in the scenario we've been discussing (40) is an example of an untrue strongly exhaustive answer to (32-a):

(40)   The (only) students at the party were Anne and Billy.

There does not appear to be any general notion of an untrue weakly exhaustive answer along these lines that is not equivalent to the notion of an untrue mention-some answer above.

Having introduced informally the general notion of *a* strongly exhaustive answer, I am now in a position to introduce one other notion of answerhood that has some intuitive interest: the notion of a 'partial answer'. Generalized partial answers do not play a direct role in any theory of embedding that I am aware of (although some of the answers used in van Rooij (2004) do not belong to any of the other answer classes in the present typology), but they are used for other purposes by Groenendijk and Stokhof (1984). Briefly, according toGroenendijk and Stokhof (1984), a partial answer is any proposition that can be derived by a (possibly infinite) disjunction of (true or untrue) strongly exhaustive answers. Generally we will want to exclude the trivial disjunctions of all the possible strongly exhaustive answers (yielding tautology) and none of them (yielding contradiction). Every strongly exhaustive answer is a partial answer, as is the weakly exhaustive answer and every mention-some answer. (41) is another example of a true partial answer to (32-a):[16]

(41)   Jonathan was not among the students at the party.

The discussion of the motivations for using these various sorts of answers (and, in some cases, the arguments against those motivations) will form a major part of this dissertation, but I want to make a few initial remarks so that the reader

---

[16]Since my examples of partial answers are not framed disjunctively, the reader may find them confusing. To see how, for example, (41) can be framed disjunctively, note that (41) is equivalent to the (infinite) disjunction of every possible strongly exhaustive answer that is consistent with it. Some strongly exhaustive answer must be true (ignoring presuppositional complication), and so (41) entails that at one of the strongly exhaustive answers consistent with (41) must be true, which is to say it entails the disjunction of all of them. Going the other way, this disjunction entails (41), since every strongly exhaustive answer that enters into the disjunction identifies some set as the set of all students at the party, and to be consistent with (41), all must exclude Jonathan from that set.

is not completely adrift. The motivation for mention-some answerhood is perhaps easiest: there are some question-embedding sentences that are naturally understood as making assertions about mention-some answers. For example (42-b) is naturally understood as asserting that William knows a mention-some answer to (42-a): the quasi-English expression (42-c) seems to convey the same information as (42-b), and (42-b) is intuitively true in a situation where (42-d) is the only thing William knows about the availability of Italian newspapers.[17]

(42)   a.   Where can Rupert buy an Italian newspaper?
            b.   William knows where Rupert can buy an Italian newspaper.
            c.   There is x such that William knows that Rupert can buy an Italian newspaper at x.
            d.   Rupert can buy an Italian newspaper at PaperWorld.

Interest in the weakly and strongly exhaustive answers is motivated by the observation that sometimes mention-some answerhood-conditions seem insufficient. For example, we are disinclined to accept (35) as true if Rupert knows of only one of the many students at the party: if Anne was not the only student at the party, it is clear that (43) does not suffice to validate (35).

(35)   Rupert knows which students were at the party.

(43)   Rupert knows that Anne is a student who was at the party.

In accounting for the truth-conditions of (35), we want to be dealing with some kind of 'mention all' answer, and this leads us to turn to exhaustive answers.

Starting with the 'mention all' idea, weak exhaustivity is a natural theoretical move: the weakly exhaustive answer is the proposition that combines *all* the true mention-some answers. This seems like a straightforward way to provide completeness. The problem that arises, which motivates the use of strongly exhaustive answers, is that the weakly exhaustive answer provides no *negative* information. Intuitively, if (35) is true, and I inform Rupert that Tucker is a student, Rupert should

---

[17]I will have cause to reevaluate this claim about (42-a) in 4.

be in a position to tell me whether Tucker was at the party, at least if we exclude cases where Rupert knows Tucker under another name or in another guise. Weak exhaustivity cannot account for this intuition: if Tucker was not among the students at the party, Rupert knowing the weakly exhaustive answer does not require that Rupert have any information to exclude the possibility that Tucker was a student at the party. Under these circumstances, the weakly exhaustive answer will be silent about Tucker, since it is merely a compilation of positive information about which students were at the party. The strongly exhaustive answer adds just enough information to license the inference we need. Once we have recognized that we sometimes need strongly exhaustive answers, we might naturally ask whether we need weakly exhaustive answers at all. This turns out to be a complex issue, which will be taken up in Sections 3.1 and 3.2.

1.4 Plan of the Dissertation

This dissertation is concerned with trying to develop an account of the semantics of embedding *'wh'*-questions under responsive embedders. Chapter 2 develops what I consider to be the natural baseline theory: a uniform theory that makes use of strongly exhaustive and mention-some answers, but no other kinds of answers. The remaining chapters are concerned with exploring the strengths, weaknesses, and possible refinements of this kind of theory. Chapter 3 presents a series of potential problems for the theory. I argue that most of these can either be explained away or handled with straightforward amendments. For a few, where the solution is not clear, I do my best to identify the nature an importance of the problem. Chapter 4 is devoted to a more serious, and I think ultimately successful, argument against all reductive accounts, including the one presented in Chapter 2. After presenting this argument, I explore the prospects for a modestly non-reductive account. Chapter 5 briefly discusses how the theory of *'wh'*-question meaning presented in Chapter 2 might be extended to handle other classes of questions, along with the 'concealed question' construction. Chapter 6 contains a speculative exploration of various unresolved issues related to mention-some readings. In Chapter 7, I conclude by reviewing the key features of my revised theory, and surveying the main open problems in the semantics of question embedding.

CHAPTER 2

The Baseline Theory

'I checked it very thoroughly,' said the computer, 'and that quite definitely is the answer. I think the problem, to be quite honest with you, is that you've never actually known what the question is.'

'But it was the Great Question! The Ultimate Question of Life, the Universe and Everything,' howled Loonquawl.

'Yes,' said Deep Thought with the air of one who suffers fools gladly, 'but what actually *is* it?'

Adams (1979)

In this chapter, I present what I consider to be a minimal theory of *'wh'*-question-embedding. I propose that a question's semantic contribution is the set of its possible answers, identifying the meaning of a question with its answerhood conditions in the same way that we identify the meaning of a declarative clause with its truth conditions. A rogative embedder is a function that takes the whole meaning of a question (i.e. an answer set) as an arguments, and a responsive embedder is a function that takes propositional arguments, and is applied to an answer set by existentially quantifying over the answer propositions (so, for example, *'know'* relates me to a question iff it relates me to one of the answers to that question). Questions are taken to be ambiguous between two different kinds of answers sets – mention-some answer sets and strongly exhaustive answer sets.

My preferred formalization of these ideas mixes elements from the theories of Groenendijk and Stokhof (1984) and Hamblin (1973) in a slightly different manner than is standard. My particular choice of formalization is not crucial at this stage. Beck and Rullmann (1999) and Lahiri (2002) provide systems that combine strongly exhaustive and mention-some embedding by layering strong exhaustivity on top of a Hamblin-type notion of question meaning, employing a formalization of strong

exhaustivity developed by Heim (1994), and Egré and Spector (2007) have shown one way to build the notion of possible (untrue) strongly exhaustive answers into this framework. My approach draws more heavily on Groenendijk and Stokhof (1984), but, for purposes of question embedding, most of what I do could be replicated within a Hamblin-style framework,[1] with the possible exception of some of the ideas I take up in Chapter 6. Nevertheless, having a concrete choice of formalism is useful, so I have chosen one with which I am comfortable.

2.1 Preliminary Remarks

Before launching into the details of my analysis, I want to spend a little time describing my general approach to the construction and presentation of formal semantic theories.

2.1.1 Information Content and Combinatorial Potential

As I discussed in Chapter 1, our theory of meaning is ultimately accountable to various kinds of data, such as judgments about the truth-conditions of declarative sentences, the answerhood conditions of questions, and so on. The semantics that we assign to each constituent is empirically adequate to the extent that it contributes to a theory that accounts for these data. To this I want to add the idea that we can, non-rigorously, think of formal theories of meaning as attributing two different kinds of semantic properties to constituents and classes of constituents: information content and combinatorial potential.

The distinction between these can be seen in the syntax and semantic of many kinds of constituents, but one simple example involves the syntax and semantics of names. A name can combine with a quantification DP by coordination, and theories of coordination typically define coordination on semantic objects of like type. Keeping to the extensional semantics for the moment, quantificational DPs are generally associated with semantic objects of type $\langle\langle e, t\rangle, t\rangle$, so it would appear that a name must have the potential to combine as a semantic object of type $\langle\langle e, t\rangle, t\rangle$. If we stop there, however, we ignore some important aspects of the semantics of

---

[1] In some cases, the Hamblin-style implementations strike me as less elegant, but much of this may be a matter of personal taste.

names. The extension of a name cannot be just *any* function of type $\langle\langle e, t\rangle, t\rangle$, or, to put it another way, if I know that an expression is a name, I don't need a complete description of the extension of such a function to know the name's extension. What I need – what distinguishes one name from another, extensionally – is just an entity (type e). The idea, shared by most semantic theories of names, is that an entity is all the extensional information that is specific to each name, but that, to allow the compositional semantics to run smoothly, there is a certain predictable way for a name to engage in various combinations that are more naturally characterized by associating it with a richer type.

A satisfying semantics of names should describe both facts: it should provide a mechanism for names to combine as type $\langle\langle e, t\rangle, t\rangle$ when needed, but it should also respect the idea that the possible extensional contributions of names do not over the full range of $\langle\langle e, t\rangle, t\rangle$ functions. There are many ways of doing this. Glossing over some details, we could say that Montague (1974) associates names with the higher type (or, rather, with a related intensional type), and includes semantic stipulations that constrain the space of possible name meanings to those that relate to an entity in a specific way, while Partee (1986) keeps names in the lower type, and provides a grammatical operation to shift them up to the higher type as needed, and Keenan and Faltz (1985) identify algebraic properties that distinguish name-meanings among possible DP meanings. We might debate the relative strengths and weaknesses of these and other approaches, but, what I am interested in here is what they all have in common: each of them recognizes, in its own way, the distinction between a name's combinatorial potential and its information content, and the need for the semantics to address this relationship.

A similar family of issues arises with the semantics of determiners. A determiner can combine with two properties to yield a truth value so, in a sense, every determiner's combinatorial potential allows it to behave like a function of type $\langle\langle s, \langle e, t\rangle\rangle, \langle\langle s, \langle e, t\rangle\rangle, t\rangle\rangle$. Although the semantics must explain how a determiner combines with two properties to yields a truth value, the class of functions of type $\langle\langle s, \langle e, t\rangle\rangle, \langle\langle s, \langle e, t\rangle\rangle, t\rangle\rangle$ contains many functions do not correspond to any plausible natural language determiner, and, if we pay some attention to distinctions between different classes of determiners, we may be able to restrict things further. Note, for example, that many kinds of determiners are extensional, in the sense that they do

not distinguish between properties with the same extension. Similarly, many determiners appear to be permutation-invariant. Finally, Keenan and Stavi (1986) have argued that *all* determiners are conservative. A good semantics of determiners will do its best to identify the range of possible determiner contents – to ask what sorts of functions that take two properties and yield a truth value are possible determiner meanings, and how much we need to know about a one determiner to distinguish it from another.

There is a sense in which most of this dissertation is concerned with exploring the combinatorial potential and information content of *'wh'*-questions and of responsive embedders, and this chapter presents a basic picture that implicitly addresses these issues.

For the broader workings of the semantics, it is convenient to associate *'wh'*-questions with proposition sets (encoding their answerhood conditions), but, if we stopped with that generalization, we would not have a very satisfying account of the information content of *'wh'*-questions. Like many other accounts of *'wh'*-questions, the one presented here adopts the position that the main information content of a *'wh'*-question is a property or relation, where the details of what kind of property or relation is involved depend on the number and kind of *'wh'*-phrases the question contains. An operator called Q will bridge the gap between the property or relation that serves as the main information content and the combinatorially convenient set of propositions. Although a property or relation is the main information content of a *'wh'*-question, we often seem to need to assume a little more information is involved, to account for various ambiguities. In this chapter, I adopt the position that the only additional information is a specification of the question as mention-some or exhaustive, encoded by the presence of absence of an operator called X. I will revisit this aspect of my account in Chapters 3 and 6.

On the side of the responsive predicates, it is normal to assume that their information content is enough to give us, directly or indirectly, a predicate of propositions. For this chapter, I assume that that is *all* there is to their information content, and that these combine with questions in a uniform way. This position strikes me as a natural baseline assumption, and I prefer to depart from it only when necessary. Setting aside some technical differences, this is the attitude found in Groenendijk and Stokhof (1984) and Egré and Spector (2007). In contrast, Beck and Rullmann

(1999), Lahiri (2002), and Heim (1994) all (implicitly or explicitly) help themselves to greater freedom to specify the question-embedding behavior of these embedders on a case-by-case basis. In subsequent chapters, especially Chapter 4, I will explore the issue of what additional information content we are compelled to attribute to responsive embedders.

2.1.2 A Note on (In-)Formalization

At this point, I want to give the reader a sense of how I plan to use logical formalism throughout this dissertation, and to warn the reader about certain abuses of notation. My general approach to presenting the compositional semantics involves translation into an intermediate logic, along the lines of Montague (1974). The particular logic assumed is a variant on a multi-sorted type-theory like that employed by Janssen (1984) and Groenendijk and Stokhof (1984). The idea is that each interpreted constituent will have a semantic value with respect to any given model, context, and assignment, and this will be given not directly by talking about models, assignments, and contexts, but by providing a translation of each constituent into an expression of the logic. The way that this expression receives its value in a given model, context, and assignment will be assumed to be more or less transparent to readers familiar with formal semantics, and a hasty overview of the key points can be found in Appendix A. When I step through the compositional steps for deriving the meaning of a particular sentence, I will give the translation of each interpreted constituent, assuming combination by rules of function application and variable binding, and some variants of these to be discussed below. The step of translation into the representation language is intended as a convenience for purposes of presentation, and not as any kind of substantial theoretical statement about the psychological representation of meaning.

In my expository prose, I will occasionally blend the English meta-language with the formal logic, committing such abuses as using the same variable and constant names in both. Many of the resulting statements will, from a technical point of view, be ill-defined, but the reaction I have gotten is that they are generally easier to read than the corresponding more rigorously satisfying metalinguistic statements would be. Concerned readers will, I hope, be able to work out the intended meaning for themselves.

2.2 Questions from Abstracts

2.2.1 The Idea of Abstracts

Following Groenendijk and Stokhof (1984), I assume that a *'wh'*-question is built up from something called an 'abstract', which we might think of as a 'general purpose' *'wh'*-clause. The idea of an abstract is, roughly, that it is what is structurally common to a *'wh'*-question and the analogous relative clause (although I will not explore the semantics of relative clauses here).[2] In particular, the intension of an abstract will be the property (or relation) that the *'wh'*-question is a question about. For example, the question (1-a) is understood a question about which things have the property of being a person that loves Anne, [3] expressed (1-c), and the relative clause in (1-b) is concerned with the same property.

(1)   a.   Who does Anne love?

        b.   They will try to murder somebody <u>who Anne loves</u>.

        c. $\lambda w_{\mathsf{s}} \lambda x_{\mathsf{e}}(person(w)(x) \wedge loves(w)(x)(Anne))$

The abstract is a constituent that has (1-c) as its intension, and occurs as a subpart of both the question (1-a) and the object of the declarative in (1-b), but lacks some operator or operators that make (1-a) a question. That is, assuming a question-formation operator Q,[4] the structural relationship between the *'wh'*-question (1-a)

---

[2]Throughout this dissertation, I will adopt the working assumption that the abstract is an actual syntactic constituent, and I think this is the right way to view things when situating things in a syntactic theory generative/transformational tradition. For theories that take other views of syntactic structure, syntactic derivation, or the syntax/semantics interface, this view may or may not be appropriate.

[3]Throughout this dissertation, I assume that *'who'* imposes a personhood restriction, but this is probably not quite right. For some speakers, *'who'* includes a weaker restriction, so that, for example, names of pets can be valid answers to a *'who'* question, and, of course, any entity that is metaphorically treated as a person can be an answer to a *'who'* question in a suitable context. That is, it may be that the right restriction is not personhood, but some slightly less restrictive requirement of high animacy, and entities that are considered persons metaphorically or 'by courtesy' should probably also be included. For convenience, I will continue to assume a personhood restriction, but the details deserve further exploration.

[4]I am, at this point, sneaking in an assumption that there is some part of (1-a) that makes it a question, beyond its *'wh'*-word (which is not enough to distinguish a question from a relative clause), and that this part is shared by (1-a) and the corresponding embedded question in (2):

(2)   Alex wonders <u>who Anne loves</u>.

and its abstract should be something like what we see in (3):

(3)

$$\text{question}$$

$$Q \quad \text{abstract}$$
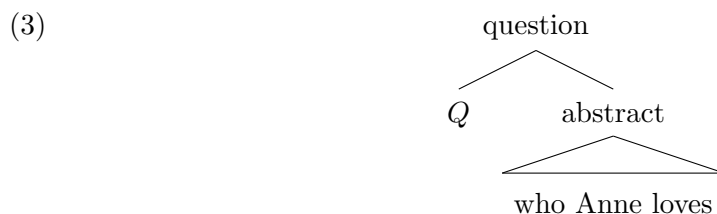
$$\text{who Anne loves}$$

Building an abstract, the idea is that *'wh'*-phrases do the work of restricted λ-operators, so the *'wh'*-word *'who'* contributes the $\lambda x_{\mathsf{e}}(person(w)(x) \land ...)$ part of (1-c), and the *'Anne loves [trace]'*[5] part that it scopes over contributes the $loves(w)(x)(Anne)$ part (the abstraction over w is just the what we do to take the intension). So, for example the abstract in (1-a) will have roughly the structure indicated by (4), where e_x is a trace associated with variable x:

(4)

$$\text{abstract}$$

$$\text{who}^x$$

$$\text{Anne loves } e_x$$
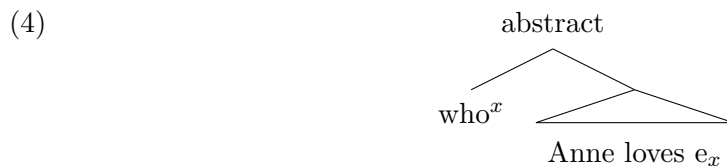
To handle multiple-*'wh'*-questions, we will need multiple-*'wh'* abstracts, so, for example, the abstract from which the grammar builds the question (5-a) will involve abstraction over one variable for each instance of *'who'*, and have the intension (5-b) with the structure schematized in (5-c).

(5)   a.   Who killed who?
       b. $\lambda w \lambda x_{\mathsf{e}} \lambda y_{\mathsf{e}}(person(w)(y) \land person(w)(x) \land killed(w)(y)(x))$

This is analogous to the idea in many mainstream syntactic accounts that (1-a) involves a complementizer that contributes a +Q feature. Since inversion occurs in (1-a) but not in (2), Q cannot, by itself, be the trigger for inversion, although it might be that it participates in the triggering of inversion in combination with some other grammatical phenomenon. I leave this as a matter for syntactic theory, noting only that any theory that uses a $C[+Q]$ in both (1-a) and (2) will have to assume that the presence of that $C[+Q]$ is not, by itself, enough to trigger inversion.

[5]I will tacitly assume that the syntax used includes *'wh'*-movement, and I will represent scope relations in the same trees I use for ordinary syntactic combination. This provides a convenient way of quasi-formally expressing my intent, but I don't foresee any particular difficulty in rendering the same ideas within any framework that lacks movement, provided that it has well-developed semantic tools for handling movement-like phenomena and scope interactions.

24

c.

$$\text{abstract}$$

who^y
   who^x
      e$_x$ killed e$_y$
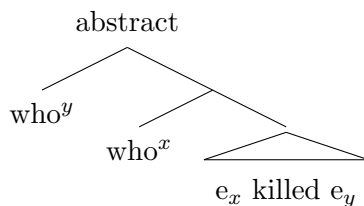
The basic derivation of abstracts is treated below in section 2.2.2, although many details are postponed until Section 2.4. The intuition that *'wh'*-phrases are λ-abstractors should be enough for most of the discussion of question-formation, at least when we're dealing with questions similar to (1-a) and (5-a).

2.2.2   Deriving Abstracts: 'Binding in' *'wh'*-Phrases

Having described the shape of abstracts informally above, I want to flesh out how abstracts are composed. This covers only the simple case, with complications postponed to Section 2.4. To begin, I will need an example of a *'wh'*-phrase meaning. Let's say, following Caponigro (2004), that *'who'* has an extension of type $\langle\langle \mathsf{e}, \mathsf{t}\rangle, \langle \mathsf{e}, \mathsf{t}\rangle\rangle$ (and so an intension of type $\langle \mathsf{s}, \langle\langle \mathsf{e}, \mathsf{t}\rangle, \langle \mathsf{e}, \mathsf{t}\rangle\rangle\rangle$), where the intension of *'who'* is given by (6).[6]

(6) $\lambda w_{\mathsf{s}} \lambda S_{\langle \mathsf{e}, \mathsf{t}\rangle} \lambda x_{\mathsf{e}} (person(w)(x) \wedge S(x))$

That is, *'who'* takes as its argument a set and returns the set of things that are people in that set. Importantly, *'who'* will be 'bound in' for a variable of type e, so the abstract for (1-a) will be built by the applying *'who'* to bind the variable (y) associated with its trace (e$_y$) as in (7), which is then evaluated as in (8).

(1-a)   Who does Anne love?

---

[6]This, along with the meanings given for other *'wh'*-words in later on, is the natural result of taking the account of *'wh'*-clauses in Groenendijk and Stokhof (1984) and making the meanings of the *'wh'*-words explicit, and is in the same spirit as the account of *'wh'*-words in Jacobson (1995).

(7)

$$\text{ii}$$

$$\text{who}^y \qquad \text{i}$$

$$\text{Anne loves e}_y$$
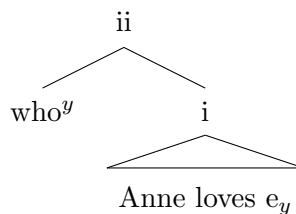
(8) $\text{i} : loves(w)(y)(Anne)$

'*who*' : $\lambda S_{\langle \mathsf{e},\mathsf{t}\rangle}\lambda x_{\mathsf{e}}(person(w)(x) \wedge S(x))$

$\text{ii} : (\lambda S_{\langle \mathsf{e},\mathsf{t}\rangle}\lambda x_{\mathsf{e}}(person(w)(x) \wedge S(x)))(\lambda y(loves(w)(y)(Anne)))$

$=$

$\lambda x_{\mathsf{e}}(person(w)(x) \wedge loves(w)(x)(Anne))$

At this point, an aside on my notation and compositional assumptions will be helpful: the step seen in (ii) of (8) is the result of binding '*who*' into node (i) for y: that is, we abstract over the variable associated with the trace of '*who*' (in this case y) to give our function $(\lambda S_{\langle \mathsf{e},\mathsf{t}\rangle}\lambda x_{\mathsf{e}}(person(w)(x) \wedge S(x)))$ an argument of the needed type ($\langle \mathsf{e},\mathsf{t}\rangle$). This is the same operation seen in 'quantifying in' (along the lines of the treatment of quantification in Montague (1974)) or 'quantifier raising' (e.g. Heim and Kratzer (1998)) analyses of quantification. This binding will always be annotated by superscripting the binding operator with the variable it binds. So binding constituent (OP) into constituent (SCOPE) for variable v will be written with the tree in (9):

(9)   ROOT

$$\text{OP}^v \quad \text{SCOPE}$$
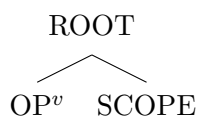
This is evaluable whenever there are types ρ, σ, and τ such that v is a variable of type ρ, OP has type $\langle\langle\rho,\sigma\rangle,\tau\rangle$, and SCOPE has type σ. In this case ROOT will have type τ, and, if we translate OP by a formula φ and SCOPE by a formula ψ, the translation of ROOT will be the formula $\varphi(\lambda v(\psi))$. In the case of '*who*', $\rho = \mathsf{e}$, $\sigma = \mathsf{t}$, and $\tau = \langle \mathsf{e},\mathsf{t}\rangle$. In the case of a typical nominal quantifier, $\rho = \mathsf{e}$, $\sigma = \mathsf{t}$, and $\tau = \mathsf{t}$. The grammar is assumed to provide complete guidelines for when we superscript a constituent with a variable, and I will not explore the details here, but

I assume one rule will be that a *'wh'*-phrase or 'raised'/'quantified-in' quantifier will always be superscripted with the variable of its trace. This presentation assumes variables and something like movement or the quantifying-in analysis of Montague (1974). In theories that handle things without variables, the formal details will of course look different, although something analogous should be possible.

Let's quickly review specific scheme for how this plays out in the case of abstracts: a *'wh'*-phrase is 'raised' to the beginning of the abstract, with a variable in its corresponding trace position. It is then combined with the rest of the clause by abstracting over its associated variable, and applying it to the result of this abstraction. The general scheme is thus that we have the structural relations given by (10). And the corresponding semantic relations given in (11):

(10)

$$
\begin{array}{c}
\text{ii} \\
\diagup\diagdown \\
\text{WH}^v \qquad \text{i} \\
\diagup\diagdown \qquad \diagup\diagdown \\
\ldots \qquad \ldots e_v
\end{array}
$$

(11)   WH : φ
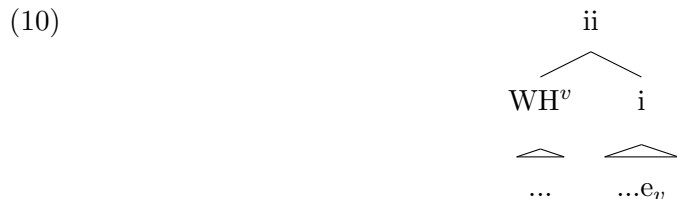
     i : $\psi(v)$

    ii : $\psi(\lambda v(\varphi(v))$

Different *'wh'*-phrases will have different types and different meanings, and will be associated with traces of different types, but (as far as single-*'wh'* questions are concerned) the scheme in (11) will work for all of them. The only important complication will come up when dealing with multiple-*'wh'* abstracts. These will be deferred until Section 2.4.2. For the time being, I will just assume that (5-a) receives the translation (12), leaving the compositional justification of this claim for later:

(5-a)   Who killed who?

(12) $\lambda x_\mathsf{e} \lambda y_\mathsf{e}(person(w)(y) \wedge person(w)(x) \wedge killed(w)(y)(x))$

2.2.3 Abstracts and Answer Sets

To get the answer sets we want for a theory of embedding, we will need operations to derive answer sets from abstracts. To begin, consider again the question (1-a):

(1-a)   Who does Anne love?

A mention-some answer to this question is any proposition that identifies an entity as a person who Anne loves – that is, it is any proposition with the property expressed by (13-a). A strongly exhaustive answer, on the other hand, is any proposition that identifies some set as the set of people that Anne loves – that is, any proposition with the property expressed by (13-b):[7]

(13)   a. $\lambda p_{\langle s,t \rangle} \exists y_e (p = \lambda w_s (person(w,y) \wedge loves(w, Anne, y)))$
       b. $\lambda p_{\langle s,t \rangle} \exists S_{\langle e,t \rangle} (p = \lambda w_s (S = \lambda x_e (person(w,x) \wedge loves(w, Anne, x))))$

Looking at (13-a) and (13-b), we see that they look quite similar: they are both sets of propositions all of which tell us something about which things have a certain property – the property associated with the 'abstract' in (1-a) – the abstract with the intension (1-c):

(1-c) $\lambda w_s \lambda x_e (person(w)(x) \wedge loves(w)(x)(Anne))$

Besides sharing (1-c), both (13-a) and (13-b) are sets of propositions p such that there is some value (x or S, in the above formulas) such that p is true if and only if that value stands in some relation to the property (1-c). The only differences are the kind of value under discussion, and the fact that in the mention some-case ((13-a)), p is true if and only if x is *in* the extension of the abstract, while in the strongly exhaustive case ((13-b)) p is true if and only if S *is* the extension of the abstract. In the mention-some case, we check membership, but in the strongly exhaustive case, we

---

[7]By defining the mention-some and strongly exhaustive answer sets for the question, I don't mean to suggest that both answer sets are salient or even semantically available for this question. Most questions seem to have a strong bias for either a mention-some use or a strongly exhaustive use, but I want to define, formally, the mention-some and strongly exhaustive answer sets for the same question, so that I can formally compare them directly.

check equality. This is really the only important difference between the two answer sets, as we will see in more detail below.

What's missing, at this point, is one or more operators to get us from the abstract (1-c) to the answer sets (13-a) and (13-b) (and do likewise for other *'wh'*-questions). Coming up with a function to get us from (1-c) to (13-a) (or (13-b)), and to do the analogous thing for other abstracts, is fairly straightforward, and the details will be fleshed out below.

2.2.3.1 Building Mention-Some Answer-Sets

For mention-some answers, the goal is to get from (1-c) to (13-a):

(1-c) $\lambda w_{\mathbf{s}} \lambda x_{\mathbf{e}}(person(w, x) \wedge loves(w, Anne, x))$

(13-a) $\lambda p_{\langle \mathbf{s}, \mathbf{t} \rangle} \exists y_{\mathbf{e}}(p = \lambda w_{\mathbf{s}}(person(w, y) \wedge loves(w, Anne, y)))$

To do this, intuitively, we just need to take (1-c) out of (13-a), and make what remains into our mention-some question formation operator. That is, we want our operator to contribute the meaning sketched intuitively in (14), where '(...)' is the 'hole' left behind when we pull the meaning of (1-c) out of (13-a):

(14) $\lambda p_{\langle \mathbf{s}, \mathbf{t} \rangle} \exists y_{\mathbf{e}}(p = \lambda w_{\mathbf{s}}((...)(w)(y)))$

Within the typed λ-calculus framework that I have been assuming, the way to rigorously give ourselves an operator that does the work of (14) is to replace (...) with a variable, and then λ-abstract over that variable to get ourselves a function, giving us (15):

(15) $\lambda \alpha_{\langle \mathbf{s}, \langle \mathbf{e}, \mathbf{t} \rangle \rangle} \lambda p_{\langle \mathbf{s}, \mathbf{t} \rangle} \exists y_{\mathbf{e}}(p = \lambda w_{\mathbf{s}}(\alpha(w)(y)))$

(15) will serve us well for many examples, but it cannot be fully general. The trouble is that it only works on properties of individuals. But *'wh'*-abstracts can express meanings of many other types – they can, for example, express properties of locations (as in (16)), of reasons, however we represent those (as in (17)), of times,

29

whether understood as intervals, events, or something else (as in (18)), and of other types besides.

(16)   Where can we buy an Italian newspaper?

(17)   Why did William kill that unicorn?

(18)   When were the bodies found?

Even if we restrict attention to nominal 'wh'-phrases, there are numerous arguments (see, for example, Groenendijk and Stokhof (1984), Aloni (2002), and Spector (2007)) that these should sometimes be associated with other types, and multiple 'wh'-questions will involve further type complications.

A general operator for forming mention-some questions will need to incorporate some form of type polymorphism. I will do this in an ad hoc manner: for every type τ, the corresponding question-formation operator Q_τ is the function given in (19):

(19) $Q_\tau = \lambda\alpha_{\langle \mathbf{s}, \langle \tau, \mathbf{t} \rangle \rangle} \lambda p_{\langle \mathbf{s}, \mathbf{t} \rangle} \exists \beta_\tau (p = \lambda w'_{\mathbf{s}} (\alpha(w')(\beta)))$

Within this general framework, (15) is just $Q_{\mathbf{e}}$. The general rule for the grammar will be that the choice of τ will be selected to match the abstract to which Q_τ is being applied: for every type τ, Q_τ will take as its arguments abstracts with intensions of type $\langle \mathbf{s}, \langle \tau, \mathbf{t} \rangle \rangle$.

Let's see how the theory works so far. The question (1-a) will be analyzed, on a mention-some reading, as the result of combining the operator Q with the abstract 'who Anne loves', as schematized in (20) (where the w superscript on the intensional operator Q indicates that it abstracts over the world variable w in its argument, 'binding in' for it in the manner discussed for other variables above.[8]), giving the sequence of semantic combinations in (21):

(1-a)   Who does Anne love?

_____

[8]This use of binding in corresponds to 'intensional function application' in Heim and Kratzer (1998).

(20)

$$
\begin{array}{c}
\text{ii} \\
\diagup\;\diagdown \\
Q_{\mathbf{e}}^{w} \qquad \text{i} \\
\diagup\quad\diagdown \\
\text{who Anne loves}
\end{array}
$$
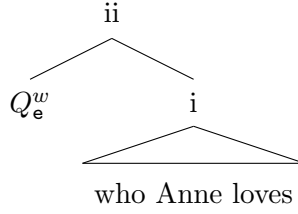
(21) $Q_{\mathbf{e}} : \lambda\alpha_{\langle\mathbf{s},\langle\mathbf{e},\mathbf{t}\rangle\rangle}\lambda p_{\langle\mathbf{s},\mathbf{t}\rangle}\exists\beta_{\mathbf{e}}(p = \lambda w_{\mathbf{s}}'(\alpha(w')(\beta)))$

$\quad$ i : $\lambda x_{\mathbf{e}}(person(w)(x) \wedge loves(w)(x)(A))$

$\quad$ ii : $(\lambda\alpha_{\langle\mathbf{s},\langle\mathbf{e},\mathbf{t}\rangle\rangle}\lambda p\exists\beta_{\mathbf{e}}(p = \lambda w'(\alpha(w')(\beta))))(\lambda w\lambda x\left(\begin{array}{c}person(w,x)\wedge \\ loves(w)(x)(A)\end{array}\right))$

$\qquad = \lambda p_{\langle\mathbf{s},\mathbf{t}\rangle}\exists\beta_{\mathbf{e}}(p = \lambda w_{\mathbf{s}}'(person(w',\beta) \wedge loves(w')(\beta)(A)))$

Here, $Q_{\mathbf{e}}$ is the version of Q chosen because the abstract *'who Anne loves'* in node (ii) is of extensional type $\langle\mathbf{e},\mathbf{t}\rangle$. To evaluate (ii) we bind the world variable w to match the needs of the intensional operator Q. The last line for (ii) in (21) shows how to simplify the expression in the first line.

This gives us a general tool for taking an abstract and building a set of mention-some answers.

2.2.3.2   Forming Multiple-*'wh'* Questions

To make Q fully general, one more issue needs to be addressed. With multiple *'wh'* questions, the abstract involved will not straightforwardly be of a type with the form $\langle\mathbf{s},\langle\tau,\mathbf{t}\rangle\rangle$. For example, the question (5-a) is built from an abstract with the intension given in (5-b), which has type $\langle\mathbf{s},\langle\mathbf{e},\langle\mathbf{e},\mathbf{t}\rangle\rangle\rangle$.

(5)   a.   Who killed who?
$\quad$ b. $\lambda w\lambda x\lambda y(person(w)(y) \wedge person(w)(x) \wedge killed(w)(y)(x))$

Simply put, there is no choice of τ such that Q_{τ} can combine with an argument like (5-b), since, on the standard type theory assumed in formal semantics, there is no value of τ such that $\langle\mathbf{s},\langle\tau,\mathbf{t}\rangle\rangle = \langle\mathbf{s},\langle\mathbf{e},\langle\mathbf{e},\mathbf{t}\rangle\rangle\rangle$. Beyond this type-theoretic issue, there is the fact that Q does not appear to give us the resources we want. The mention-some reading we want for (5-a) should be the set of propositions that

identify an ordered pair of entities as having the property that the first entity in the pair killed the second and both of the entities are people. That is, we want the answerhood conditions given by (22):

(22) $\quad \lambda p_{\langle \mathbf{s}, \mathbf{t} \rangle} \exists \beta_{\mathbf{e}} \beta'_{\mathbf{e}} (p = \lambda w'_{\mathbf{s}} (person(w)(\beta') \wedge person(w)(\beta) \wedge killed(w)(\beta')(\beta)))$

That is, we need to fill two argument slots in the abstract (5-b) by existential quantification over entities. But Q only gives us the resources to fill one argument position in this way:

(19) $\quad Q_\tau = \lambda \alpha_{\langle \mathbf{s}, \langle \tau, \mathbf{t} \rangle \rangle} \lambda p_{\langle \mathbf{s}, \mathbf{t} \rangle} \exists \beta_\tau (p = \lambda w'_{\mathbf{s}} (\alpha(w')(\beta)))$

What we would like to do, of course, is describe a general Q that can existentially quantify to fill arbitrarily many argument positions, allowing us to instantiate Q as in (23):

(23) $\quad Q = \lambda \alpha_{\langle \mathbf{s}, \langle \mathbf{e}, \langle \mathbf{e}, \mathbf{t} \rangle \rangle \rangle} \lambda p_{\langle \mathbf{s}, \mathbf{t} \rangle} \exists \beta'_{\mathbf{e}} \beta''_{\mathbf{e}} (p = \lambda w'_{\mathbf{s}} (\alpha(w')(\beta')(\beta'')))$

Unfortunately, as things have been set up so far, there is no way to make the Q in (23) an instance of the general schema given in (19).

This difficulty is not a deep theoretical problem, but only a side-effect of our relatively basic type theory. A natural paraphrase of existential quantification in (22) is could be 'for every (ordered) pair of entities...' If only we could set τ to in Q_τ to be the type of such ordered pairs, and if only our logic and our type theory gave us the tools to deal with such pairs, this problem would disappear. The idea of Q is that it builds a set of propositions by a process that includes existentially quantifying over whatever sort of things its argument is a property of, and, if we ask what kind of properties are expressed by multiple-'wh' abstracts, it seems natural to say they express properties of pairs. Everything that follows is just one way of expanding our type theory and associated machinery to reflect the (I hope) inoffensive idea that a function of two arguments can be applied to an ordered pair of two items, that a function of three arguments can be applied to an ordered triple, and so on, and that these pairs, triples, and so on can be included in the ontology for purposes of doing compositional semantics.

The straightforward solution is to leave the definition of Q in (19) exactly as it is, and augment the logic and type theory to deal in tuples and tuple types.[9] The idea will be that for any $n > 1$ and any types $\tau_1, ..., \tau_n$, there is a tuple type $(\tau_1; ...; \tau_n)$ of ordered n-tuples with first element drawn from D_{τ_1}, second element drawn from D_{τ_2}, and so on, where D_τ is the domain of values of a type τ. That is, $D_{(\tau_1; ...; \tau_n)} = D_{\tau_1} \times ... \times D_{\tau_n}$, which is to say the set of all n-tuples $(x_1, ..., x_n)$, such that $x_i \in D_{\tau_i}$ for all i such that $1 \leq i \leq n$.

Now all we have to do is let the τ in Q_τ range over singleton and tuple types, and define application of functions to tuple types in a way that handles multiple-'wh' questions. For example, since (5-b) expresses a property of pairs of individuals, we will form the mention-some answer set for (5-a) using $Q_{(e;e)}$. Since (5-b) has type $\langle s, \langle e, \langle e, t \rangle \rangle \rangle$, and $Q_{(e;e)}$ takes an argument of type $\langle s, \langle (e;e), t \rangle \rangle$, this does not immediately resolve things, unless we take the additional step of declaring that $\langle s, \langle e, \langle e, t \rangle \rangle \rangle$ and $\langle s, \langle (e;e), t \rangle \rangle$ are the same type (and describe how to apply a function of type $\langle (e;e), t \rangle$ to an argument of type e, and a function of type $\langle e, \langle e, t \rangle \rangle$ to an argument of type $(e;e)$).

The identification of these two types will be achieved by fiat. For present purposes, I declare that the rule for forming function types will now have two parts: if τ and σ are both non-tuple types,[10] then $D_{\langle \tau, \sigma \rangle} = D_\sigma^{D_\tau}$ as usual. If τ is a tuple type, on the other hand, let $\tau = (\tau_1; ...; \tau_n)$. In this case, $\langle \tau, \sigma \rangle$ is treated as another name for the type $\langle \tau_1, ... \langle \tau_n, \sigma \rangle ... \rangle$, so, considering as an example the case where $n = 3$, we'll have $\langle (\tau_1; \tau_2; \tau_3), \sigma \rangle = \langle \tau_1, \langle \tau_2, \langle \tau_3, \sigma \rangle \rangle \rangle$.[11] Since I only need to use tuples arguments for present purposes, I will assume that $\langle \tau, \sigma \rangle$ is undefined when σ is a tuple type.

Having defined the necessary type equivalences, we now need to see how to do the necessary function applications. That is, since $\langle (e;e), t \rangle$ is just another name for $\langle e, \langle e, t \rangle \rangle$, we need a way of applying a function of type $\langle e, \langle e, t \rangle \rangle$ to an argument of

---

[9]For more details of the way that I envision this working, the reader is referred to Appendix A.

[10]That is, types not of the form $(\tau_1; ...; \tau_n)$ for any $\tau_1, ..., \tau_n$.

[11]As a concrete example of the kinds of equivalences this produces, all the type expressions in (24) identify the same type:

(24) a. $\langle s, \langle e, \langle e, t \rangle \rangle \rangle$
   b. $\langle s, \langle (e;e), t \rangle \rangle$
   c. $\langle (s;e), \langle e, t \rangle \rangle$
   d. $\langle (s;e;e), t \rangle$

type $(e;e)$. Naturally, the way to do this is to apply the function to one coordinate of the pair at a time. That is, applying a function to a tuple argument will be defined by 'peeling off' one item from the tuple at a time. In the case of a pair (2-tuple), let φ is an expression of type $\langle \tau_1, \langle \tau_2, \sigma \rangle \rangle$ such that $[\![\varphi]\!] = f \in D_{\langle \tau_1, \langle \tau_2, \sigma \rangle \rangle}$, and let η is an expression of type $(\tau_1; \tau_2)$ such $[\![\eta]\!] = (a_1, a_2)$, where $a_1 \in D_{\tau_1}$ and $a_2 \in D_{\tau_2}$. The rule for evaluating $\varphi(\eta)$ will be that $[\![\varphi(\eta)]\!] = f(a_1)(a_2)$. A general recursive definition is possible, but, at least for now, the case of pairs will serve our purposes.

From the above, we may observe that, $Q_{(e;e)}$ is as in (25-a), which is equivalent to (25-b):

(25) a. $Q_{(e;e)} = \lambda \alpha_{\langle s, \langle (e;e), t \rangle \rangle} \lambda p_{\langle s, t \rangle} \exists \beta_{(e;e)} (p = \lambda w'_s (\alpha(w')(\beta)))$

  b. $Q_{(e;e)} = \lambda \alpha_{\langle s, \langle e, \langle e, t \rangle \rangle \rangle} \lambda p_{\langle s, t \rangle} \exists \beta'_e \beta''_e (p = \lambda w'_s (\alpha(w')(\beta')(\beta'')))$

With this in mind, We can build the mention-some answer set for (5-a) in terms of the composition expressed in (26).

(5-a) Who killed who?

(26)
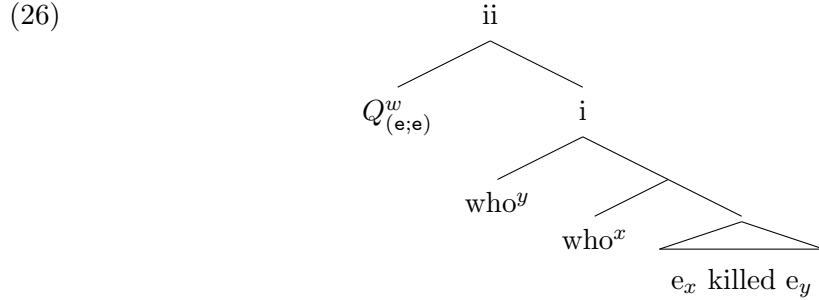
(27) i : $\lambda x \lambda y (person(w)(y) \wedge person(w)(x) \wedge killed(w)(y)(x))$

$Q_{(e;e)}$: $\lambda \alpha_{\langle s, \langle (e;e), t \rangle \rangle} \lambda p_{\langle s, t \rangle} \exists \beta_{(e;e)} (p = \lambda w'_s (\alpha(w')(\beta)))$

ii : $\lambda p_{\langle s, t \rangle} \exists \beta_{(e;e)} (p = \lambda w'_s ((\lambda x \lambda y \begin{pmatrix} person(w')(y) \wedge \\ person(w')(x) \wedge \\ killed(w')(y)(x) \end{pmatrix}) (\beta)))$

$= \lambda p_{\langle s, t \rangle} \exists \beta'_e \beta''_e (p = \lambda w'_s ((\lambda x \lambda y \begin{pmatrix} person(w')(y) \wedge \\ person(w')(x) \wedge \\ killed(w')(y)(x) \end{pmatrix}) (\beta')(\beta'')))$

$$= \lambda p_{\langle \mathsf{s,t} \rangle} \exists \beta'_{\mathsf{e}} \beta''_{\mathsf{e}} (p = \lambda w'_{\mathsf{s}} \left(\begin{array}{c} person(w')(\beta') \wedge person(w')(\beta) \\ \wedge killed(w')(\beta')(\beta) \end{array} \right))$$

That is, after paraphrasing out the type-$(\mathsf{e};\mathsf{e})$ variable and substitute in favor of two type-e variables, we are left with the set of propositions which identify some pair of entities as a pair of people such that the first coordinate of the pair killed the second. These propositions seem like the natural mention-some answers for (5-a).

It is important to see that our framework can generate multiple-'*wh*' questions, but in fact few of the issues that I discuss below will hinge on multiple-'*wh*' question data, so this machinery will not see much use. The formation of multiple-'*wh*' abstracts will be discussed below, however.

Having gone on at some length about how to define and use Q_τ for arbitrary types τ, I will for most of the rest of this document refrain from specifying the type, and speak simply of Q, since the choice of τ is determined completely by the type of Q's argument.

2.2.3.3   Building Strongly Exhaustive Answer Sets

Having shown how to derive mention-some answer sets, I now turn to the issue of strongly exhaustive answer sets. Recall that the strongly exhaustive answer set for (1-a) is (13-b):

(1-a)   Who does Anne love?

(13-b) $\lambda p_{\langle \mathsf{s,t} \rangle} \exists S_{\langle \mathsf{e,t} \rangle} (p = \lambda w_{\mathsf{s}} (S = \lambda x_{\mathsf{e}} (person(w,x) \wedge loves(w, Anne, x))))$

As with the mention-some case, our goal is to find an operator to derive (13-b) from the intension of the abstract for '*who anne loves*', given by (1-c):

(1-c) $\lambda w_{\mathsf{s}} \lambda x_{\mathsf{e}} (person(w)(x) \wedge loves(w)(x)(Anne))$

To do this, we could proceed as in the mention-some case: we would take (1-c) out of (13-b), see what was left, and write a function to add that in. This would work, but I want to try something else – (13-b) has a superficially almost identical shape

to the formula for the mention-some answer set (13-a):

(13-a) $\quad \lambda p_{\langle \mathbf{s}, \mathbf{t} \rangle} \exists y_{\mathbf{e}}(p = \lambda w_{\mathbf{s}}(person(w, y) \wedge loves(w, Anne, y)))$

In light of this, I propose that we declare the Q operator, which by itself serves to form mention-some questions, to be a general question-formation operator, and add an additional operator, called X, that adds exhaustivity, using the two operators in combination to produce strongly exhaustive readings of 'wh'-questions.

What must X do? the difference between (13-b) and (13-a) is that (13-b) includes an equality check. The mention-some answers say something about the membership of the set of people Anne loves – each of these propositions predicates (1-c) of some entity. Strongly exhaustive answers don't just say something about the membership of the set of people Anne loves: they tell us exactly what set is the set of people Anne loves: propositions in (13-b) are propositions that identify some set as the extension of (1-c).

My definition of X makes it the minimal equality-test-contributing function: for any type τ, let X_τ be as defined by (28):

(28) $\quad X_\tau = \lambda \gamma_\tau \lambda \delta_\tau (\gamma = \delta)$

The idea will be that X is applied to the extension of an abstract, and then Q is applied intensionally to the result. More fully, for an abstract with an extension of type τ (with free world variable w), the general pattern for the exhaustive reading of a question will involve the compositions schematized by (29), with the results given in (30):

(29)

36
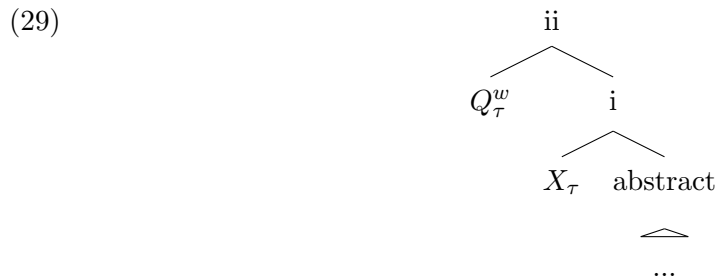
(30)   abs : $\varphi(w)$

$\quad X_\tau$: $\lambda\gamma_\tau\lambda\delta_\tau(\gamma = \delta)$

$\quad$ i : $(\lambda\gamma_\tau\lambda\delta_\tau(\gamma = \delta))(\varphi(w))$
$\qquad = \lambda\delta_\tau(\varphi(w) = \delta)$

$\quad Q_\tau$: $\lambda\alpha_{\langle \mathbf{s},\langle \tau,\mathbf{t}\rangle\rangle}\lambda p_{\langle \mathbf{s},\mathbf{t}\rangle}\exists\beta_\tau(p = \lambda w'_\mathbf{s}(\alpha(w')(\beta)))$

$\quad$ ii : $(\lambda\alpha_{\langle \mathbf{s},\langle \tau,\mathbf{t}\rangle\rangle}\lambda p_{\langle \mathbf{s},\mathbf{t}\rangle}\exists\beta_\tau(p = \lambda w'_\mathbf{s}(\alpha(w')(\beta))))(\lambda w\lambda\delta_\tau(\varphi(w) = \delta))$
$\qquad = \lambda p_{\langle \mathbf{s},\mathbf{t}\rangle}\exists\beta_\tau(p = \lambda w'_\mathbf{s}((\lambda w\lambda\delta_\tau(\varphi(w) = \delta))(w')(\beta)))$
$\qquad = \lambda p_{\langle \mathbf{s},\mathbf{t}\rangle}\exists\beta_\tau(p = \lambda w'_\mathbf{s}(\varphi(w') = \beta))$

Now lets see how this plays out for (1-a):

(1-a)   Who does Anne love?

(31)
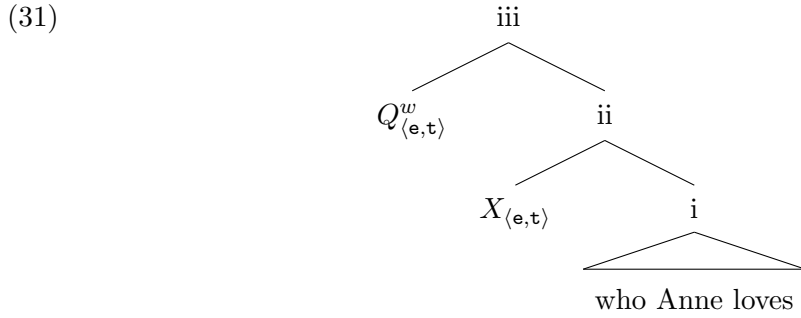
(32)   i : $\lambda x(person(w)(x) \wedge loves(w)(x)(Anne))$

$\quad X_{\langle \mathbf{e},\mathbf{t}\rangle}$: $\lambda\gamma_{\langle \mathbf{e},\mathbf{t}\rangle}\lambda\delta_{\langle \mathbf{e},\mathbf{t}\rangle}(\gamma = \delta)$

$\quad$ ii : $\lambda\delta_{\langle \mathbf{e},\mathbf{t}\rangle}(\lambda x(person(w)(x) \wedge loves(w)(x)(Anne)) = \delta)$

$\quad Q_{\langle \mathbf{e},\mathbf{t}\rangle}$: $\lambda\alpha_{\langle \mathbf{s},\langle\langle \mathbf{e},\mathbf{t}\rangle,\mathbf{t}\rangle\rangle}\lambda p_{\langle \mathbf{s},\mathbf{t}\rangle}\exists\beta_{\langle \mathbf{e},\mathbf{t}\rangle}(p = \lambda w'_\mathbf{s}(\alpha(w')(\beta)))$

$\quad$ iii : $\lambda p_{\langle \mathbf{s},\mathbf{t}\rangle}\exists\beta_{\langle \mathbf{e},\mathbf{t}\rangle}(p = \lambda w'_\mathbf{s}(\lambda x \begin{pmatrix} person(w')(x)\wedge \\ loves(w')(x)(Anne) \end{pmatrix} = \beta))$

That is, when we add X to the derivation of (1-a), we get the set of all propositions that identify some set β as the set of people Anne loves (so each answer proposition is, for some set β, the proposition that the entities in β are all people that Anne loves, and that everybody that Anne loves is in β). These are the strongly exhaustive answers we need.

As a brief aside, I want to now try to address the questions some readers may have about the possibility of scoping X over Q. I assume that this is ruled out on syntactic grounds (that is, roughly, that the kind of constituent formed by Q is not among permitted syntactic arguments of X, and that neither X nor Q can 'move' to generate scope ambiguities), and I won't attempt to provide any further principle to justify excluding this ordering. In particular, the order that scopes X over Q is not in any obvious way ruled out on type-theoretic grounds. Instead, I think of the prohibition against this order analogous to the way that default sentential structure in mainstream generative accounts of English syntax always has CP somewhere above TP, and TP somewhere above the VP, while permutations of these elements in the spine of a sentence are typically excluded.

For those who are curious, I will now briefly sketch the results of the X over Q order, to confirm that this order is not especially useful or interesting. Let's consider what happens when we perform this reversal on (1-a). This involves adopting the tree (33), with semantics as in (34):

(33)
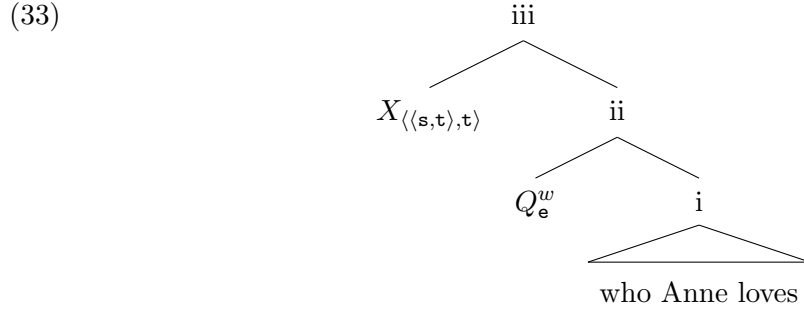
(34)   i : $\lambda x_{\mathbf{e}}(person(w)(x) \wedge loves(w)(x)(Anne))$

$Q_{\mathbf{e}} : \lambda\alpha_{\langle\mathbf{s},\langle\mathbf{e},\mathbf{t}\rangle\rangle}\lambda p_{\langle\mathbf{s},\mathbf{t}\rangle}\exists\beta_{\mathbf{e}}(p = \lambda w'_{\mathbf{s}}(\alpha(w')(\beta)))$

ii : $\lambda p_{\langle\mathbf{s},\mathbf{t}\rangle}\exists\beta_{\mathbf{e}}(p = \lambda w'_{\mathbf{s}}(person(w',\beta) \wedge loves(w')(\beta)(Anne)))$

$X_{\langle\langle\mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle} : \lambda\gamma_{\langle\langle\mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle}\lambda\delta_{\langle\langle\mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle}(\gamma = \delta)$

iii : $\lambda\delta_{\langle\langle\mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle}(\lambda p_{\langle\mathbf{s},\mathbf{t}\rangle}\exists\beta_{\mathbf{e}}(p = \lambda w'_{\mathbf{s}}\left(\begin{array}{c} person(w',\beta)\wedge \\ loves(w')(\beta)(Anne) \end{array}\right)) = \delta)$

That is, (ii) of (33) is interpreted as the mention-some answer set of (1-a), just like it was when we used Q without X to derive the mention-some reading. Adding X on

top of this packages this answer set inside a set – so for (iii) we get the set containing the mention-some answer-set of the question, and nothing else. I know of no reason why we would have any use of this object as a meaning for (1-a).

2.3 Question Embedders and Embedding

So far, I have sketched how questions contribute answer sets. Next, we need to ask how these answer sets combine with embedders to give us the truth-conditions of question-embedding sentences, and we need to address the related question of what sorts of extensions question-embedders have. I address this issue below, treating rogative and responsive embedders separately.

2.3.1 Rogative Embedders

Rogative embedders, as I discussed earlier, appear to simply be about questions, with no corresponding propositional use. Some, like *'ask'*, are typically used to report speech acts involving questions. In light of this, I will simply assume that the extension of a rogative embedder is a function that simply take an answer set (that is, the semantic contribution of a question) as one of its arguments. This means they will have an argument of type $\langle \langle \mathbf{s}, \mathbf{t} \rangle, \mathbf{t} \rangle$. Thus the extension of *'ask'* with used an indirect object (as in (35-a)) will be $\langle \mathbf{e}, \langle \langle \langle \mathbf{s}, \mathbf{t} \rangle, \mathbf{t} \rangle, \langle \mathbf{e}, \mathbf{t} \rangle \rangle \rangle$, and the type of *'ask'* with a direct object (as in (35-b)) will be $\langle \langle \langle \mathbf{s}, \mathbf{t} \rangle, \mathbf{t} \rangle, \langle \mathbf{e}, \mathbf{t} \rangle \rangle$.

(35)   a.   Anya asked Red who Alex admired.
       b.   Anya asked who alex admired.

There will be no further analysis of *'ask'*: (35-a) is true if and only if Anya asked Red a question with the same answerhood conditions as (36) (on either a strongly exhaustive or mention-some reading, depending on which reading the question is assigned), and that's all that we'll say about it.

(36)   Who does Alex admire?

This is surely not all there is to say about *'ask'* and other rogative embedders, but it is were I will stop – rogative embedders are not my focus, and this trivial analysis will serve well enough.

2.3.2 Responsive Embedders

2.3.2.1 The Idea of the Embedding Rule

The natural assumption for responsive embedders is that their basic meanings are functions of propositional arguments. So, for example, the extension of *'know'* would be a function of type $\langle\langle s,t\rangle,\langle e,t\rangle\rangle$ (that is, one combining first with a propositional (type $\langle s,t\rangle$) object and then with an entity-type subject). A ditransitive responsive embedder, like *'tell'* (as used in (37)) would have an extension of type $\langle e,\langle\langle s,t\rangle,\langle e,t\rangle\rangle\rangle$. *'surprise'*, which syntactically selects for a clausal subject (seen in (38)), might have an extension of type $\langle e,\langle\langle s,t\rangle,t\rangle\rangle$ (inverting the argument order in *'know'*).

(37)   Anne told Rupert that Liam was not who he said he was.

(38)   That Rupert had betrayed her surprised Anne.

Assuming that we are furnished with one of these functions (taking a propositional argument) as the contribution of a responsive predicate, the theory of question embedding needs to say how compositional semantics uses an answer set (type $\langle\langle s,t\rangle,t\rangle$) to fill a propositional argument slot. My answer is that the propositional predicate and the proposition-set are combined by existentially quantifying over propositions in the set. There are a few ways we might implement this, two of which will be discussed below, but first I want to cultivate the relevant intuition. The rule is simple: a responsive predicate is true of a question (contributing either an answer set, which may be either a set of mention-some answers or a set of strongly exhaustive answers) if it is true of at least one proposition in that answer set. As a concrete example, consider (39) on a strongly exhaustive reading of the embedded question. On this reading, the claim is that (39) is true if and only if there is at least one proposition p' in the answer set of the embedded question (1-a) (given in (40)),

such that Alex knows p'.

(39)   Alex knows who Anne loves.

(1-a)   Who does Anne love?

(40) $\lambda p_{\langle s,t \rangle} \exists \beta_{\langle e,t \rangle} (p = \lambda w'_s (\lambda x (person(w')(x) \wedge loves(w')(x)(Anne)) = \beta))$

Note that, in any given world, only one proposition in (40) is true. This is because the propositions in the answer set are the propositions that identify some set as the extension of the abstract. Since the abstract has only one extension, only one such proposition is true. In particular, (41) is the only true strongly exhaustive answer in the world w^*. Substituting any other set for the value of $\lambda x (person(w^*)(x) \wedge loves(w^*)(x)(Anne))$ will yield an untrue proposition.

(41) $\lambda w'_s (\lambda x \begin{pmatrix} person(w')(x) \\ \wedge loves(w')(x)(Anne) \end{pmatrix}) = \lambda x \begin{pmatrix} person(w^*)(x) \wedge \\ loves(w^*)(x)(Anne) \end{pmatrix})$

Now, since knowledge requires truth, any answer that Alex knows must be a true one, so, on this reading, (39) will be true iff Alex knows (41). All other answers in the set are untrue, and therefore cannot be part of Alex's knowledge. So, for example, if Anne loves Joyce and Liam and nobody else, (39) is true if and only if Alex knows that Joyce and Liam are the only people that Anne loves.

When we pick a non-veridical embedder, things get a little more interesting, even when we restrict attention to strongly exhaustive readings. For example, (42) will be true if and only if there is some proposition p' that belongs to the set (40) (that is, some strongly exhaustive answer to (1-a)) such that p' is among the things of which Red is certain:

(42)   Red is certain of who Anne loves.

To pick two examples, the propositions in the answer set include the propositions contributed by (43-a) and (43-b):

(43)   a.   The (only) people that Anne loves are Joyce and Alex.

          b.   The (only) person that Anne loves is Liam.

This means that Red's being certain of (43-a) or (43-b) (or any other strongly exhaustive answer) suffices to make (42) true. Because it is possible to be certain of a false proposition (while it is not possible to know a false proposition), the true answer is not privileged above the false ones.

As the last informal example, consider the case where the embedded question receives a mention-some reading (that is, where it is built by applying Q directly to the abstract, without first applying X). On a mention-some reading of (44), the answers for (44) are propositions that identify some place as a place where Rupert can buy an Italian newspaper – that is, the propositions in the set (45) (where l is the name of the type of locations), including the propositions given by (46-a) and (46-b).

(44)   Where can Rupert buy a newspaper?

$$(45)\quad \lambda p_{\langle s,t\rangle}\exists\beta_1(p=\lambda w'(\exists w''\left(\exists x\left(\begin{array}{c}can(w')(w'')\wedge\\ newspaper(w'')(x)\wedge\\ buysat(w'')(\beta)(x)(Rupert)\end{array}\right)\right)))$$

(46)   a.   Rupert can buy a newspaper at PaperWorld.

          b.   Rupert can buy a newspaper at Newstopia.

The prediction of the existential quantification story is that (47) is true if and only if Janna knows some proposition in the set (45). This seems to be a plausible statement of the truth-conditions of (47).

(47)   Janna knows where Rupert can buy an Italian newspaper.

The above provides everything that really needs to be said about my account of embedding. To see whether a responsive predicate R relates an entity x to the answer set of a question, we have to determine whether R relates x to any proposition in that answer set. If so, then R also relates x to the answer set. If not, then it does

not.

Two possible formal implementations are included below mainly in the interests of completeness, and to set up some modifications in future chapters. In particular, the lexical rule implementation will be the (mostly implicit) point of comparison for the revised account in Chapter 4, and the quantifier raising implementation will be relevant for the discussion of pair-list readings in Section 3.5.

2.3.2.2 Lexical Rule Implementation

One way to implement the embedding rule described above is to say that, as far as composition at the syntax-semantics interface is concerned, there are two *'know'*s (and likewise for other responsive predicates): one that takes a propositional argument, and one that takes an answer set as an argument. The latter will be derived from the former by a lexical rule. Propositional *'know'*, written $knows_{PROP}$, will be the familiar propositional predicate, with intensional type $\langle \mathbf{s}, \langle \langle \mathbf{s}, \mathbf{t} \rangle, \langle \mathbf{e}, \mathbf{t} \rangle \rangle \rangle$. Meanwhile, question-embedding *'know'*, written $knows_{QUES}$, will be the function of type $\langle \mathbf{s}, \langle \langle \langle \mathbf{s}, \mathbf{t} \rangle, \mathbf{t} \rangle, \langle \mathbf{e}, \mathbf{t} \rangle \rangle \rangle$ given by (48):

(48) $knows_{QUES} = \lambda w_{\mathbf{s}} \lambda P_{\langle \langle \mathbf{s}, \mathbf{t} \rangle, \mathbf{t} \rangle} \lambda x_{\mathbf{e}} \exists p'_{\langle \mathbf{s}, \mathbf{t} \rangle} (P(p') \wedge knows_{PROP}(w)(p')(x))$

That is, $knows_{QUES}$ relates an entity x to a set of propositions P in a world iff there is some proposition p' in P such that $knows_{PROP}$ relates x to p' in that world. This is just an encoding of the quantificational embedding rule already sketched. To see it in action, consider the derivation of (39), in (49) and (50):

(39)   Alex knows who Anne loves.

(49)

```
                          iii
                        /     \
                     Alex      ii
                             /    \
                          knows     i
                                  /    |    \
                             Q  X  who anne loves
```
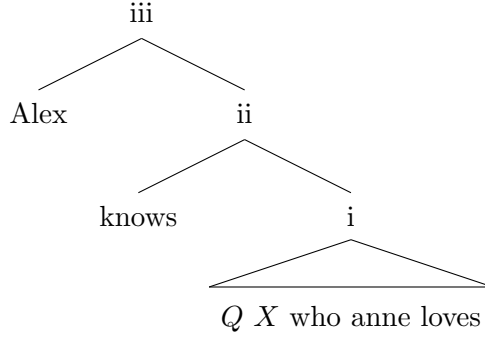
(50)　　i : $\lambda p_{\langle \mathbf{s},\mathbf{t}\rangle}\exists \beta_{\langle \mathbf{e},\mathbf{t}\rangle}(p = \lambda w'_{\mathbf{s}}(\lambda x \begin{pmatrix} person(w')(x)\wedge \\ loves(w')(x)(Anne) \end{pmatrix} = \beta))$

knows : $knows_{QUES}(w^*)$
$\qquad = \lambda P_{\langle\langle \mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle}\lambda y_{\mathbf{e}}\exists p'_{\langle \mathbf{s},\mathbf{t}\rangle}(P(p') \wedge knows_{PROP}(w^*)(p')(y))$

ii : $\lambda y \exists p' \begin{pmatrix} \exists \beta(p' = \lambda w'(\lambda x \begin{pmatrix} person(w')(x)\wedge \\ loves(w')(x)(Anne)\end{pmatrix} = \beta)) \\ \wedge knows_{PROP}(w^*)(p')(y) \end{pmatrix}$

Alex : $Alex$

iii : $\exists p' \begin{pmatrix} \exists \beta(p' = \lambda w'(\lambda x \begin{pmatrix} person(w')(x)\wedge \\ loves(w')(x)(Anne)\end{pmatrix} = \beta)) \\ \wedge knows_{PROP}(w^*)(p')(Alex) \end{pmatrix}$

That is, (39) is true iff Alex knows some proposition p' in the answer set of the embedded question *'who Anne loves'*. This is illustrated above for a strongly exhaustive reading, but the action of $knows_{QUES}$ on a set of mention-some answers is the same.

The theory will just be that the lexicon has a rule for deriving R_{QUES} from R_{PROP} for any responsive predicate R. In the case where R_{PROP} is of the same type as $knows_{PROP}$ (that is $\langle \mathbf{s}, \langle\langle \mathbf{s},\mathbf{t}\rangle, \langle \mathbf{e},\mathbf{t}\rangle\rangle\rangle$), the rule is just the result of substituting R_{QUES} and R_{PROP} for $knows_{QUES}$ and $knows_{PROP}$ in (48), so we have the following.

(51)　　If R_{PROP} is of type $\langle \mathbf{s}, \langle\langle \mathbf{s},\mathbf{t}\rangle, \langle \mathbf{e},\mathbf{t}\rangle\rangle\rangle$, then
$\qquad R_{QUES} = \lambda w_{\mathbf{s}}\lambda P_{\langle\langle \mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle}\lambda x_{\mathbf{e}}\exists p'_{\langle \mathbf{s},\mathbf{t}\rangle}(P(p') \wedge R_{PROP}(w)(p')(x))$

That is, this operation will derive a predicate of propositions from a predicate of propositions in such a way that the former is true of the question iff the latter is true of some answer to the question. To be fully general, this rule needs to be made applicable to functions of other types, to handle responsive predicates like *'surprise'* and *'tell'*. A fully general statement would only be distracting for present purposes, so below I simply note the cases for the types $\langle \mathbf{s}, \langle \mathbf{e}, \langle \langle \mathbf{s}, \mathbf{t} \rangle, \mathbf{t} \rangle \rangle \rangle$ and $\langle \mathbf{s}, \langle \mathbf{e}, \langle \langle \mathbf{s}, \mathbf{t} \rangle, \langle \mathbf{e}, \mathbf{t} \rangle \rangle \rangle \rangle$:

(52)   If R_{PROP} is of type $\langle \mathbf{s}, \langle \mathbf{e}, \langle \langle \mathbf{s}, \mathbf{t} \rangle, \mathbf{t} \rangle \rangle \rangle$, then
$$R_{QUES} = \lambda w_{\mathbf{s}} \lambda x_{\mathbf{e}} \lambda P_{\langle \langle \mathbf{s}, \mathbf{t} \rangle, \mathbf{t} \rangle} \exists p'_{\langle \mathbf{s}, \mathbf{t} \rangle} (P(p') \wedge R_{PROP}(w)(x)(p'))$$

(53)   If R_{PROP} is of type $\langle \mathbf{s}, \langle \mathbf{e}, \langle \langle \mathbf{s}, \mathbf{t} \rangle, \langle \mathbf{e}, \mathbf{t} \rangle \rangle \rangle \rangle$, then
$$R_{QUES} = \lambda w_{\mathbf{s}} \lambda x_{\mathbf{e}} \lambda P_{\langle \langle \mathbf{s}, \mathbf{t} \rangle, \mathbf{t} \rangle} \lambda y_{\mathbf{e}} \exists p'_{\langle \mathbf{s}, \mathbf{t} \rangle} (P(p') \wedge R_{PROP}(w)(x)(p')(y))$$

Since not all propositional predicates can embed questions (cf. *'hope'* in (54)), this lexical rule will only be available for certain propositional predicates: each propositional predicate will still need to be marked for whether it is allowed to embed questions, and some will not have this option available.

(54)   a.   I hope that Anne didn't kill Liam.
        b.   *I hope whether Anne killed Liam.
        c.   *I hope whether Anne or Kendra killed Liam.
        d.   *I hope who killed Liam.

For concreteness, the reader should assume that throughout the rest of the text that we are using the lexical rule implementation for embedding, except in those cases where I specifically say another approach is used. I would, however, like to briefly introduce an alternative 'quantifier raising' implementation before proceeding.

2.3.2.3   Quantifier Raising Implementation

The approach described above puts the existential quantification over answers into the semantics of the question-embedding variant of *'know'*. Another option would be to attach it to the question. That is, we could build a constituent that acts as an existential quantifier over propositions in the answer set, and combine this with the

meaning of propositional *'know'*, using whatever machinery we generally prefer to use for quantifying into an argument position of a predicate. This is the natural way of combining a 'lifted questions' account (see Szabolcsi (1997) for a good exposition of this general kind of account) with my assumption that the semantic value of every question is an answer set, and it also bears a strong resemblance to the 'interrogative raising' approach adopted by Lahiri (2002), although it does not address the issues (mostly related to quantificational variability) that Lahiri's approach was designed to handle.

For concreteness, let's assume that quantifying in is represented with a 'quantifier raising' analysis along the lines of Heim and Kratzer (1998). In this case, (39) would be analyzed with (55), where e_p is a proposition-type trace associated with variable p, and $\exists_{QUES}$ is the operator that contributes existential quantification. This is analogous to the analysis of (56) as (57).

(39)   Alex knows who Anne loves.

(55)

$$\text{lifted-question}^p$$

$$\exists_{QUES} \quad \text{who anne loves} \qquad \text{Alex knows } e_p$$
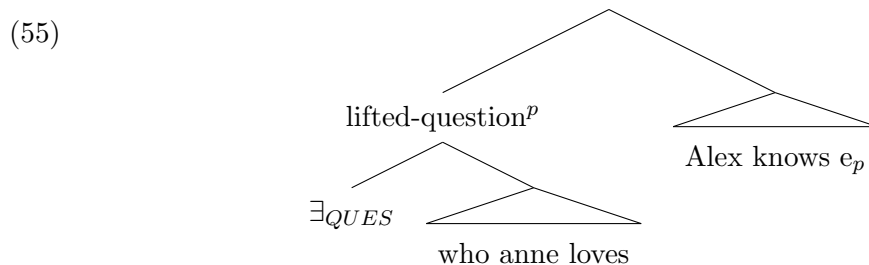
(56)   Anne loves some student.

(57)

$$DP^x \qquad \text{Anne loves } e_x$$

$$\text{some} \quad \text{student}$$
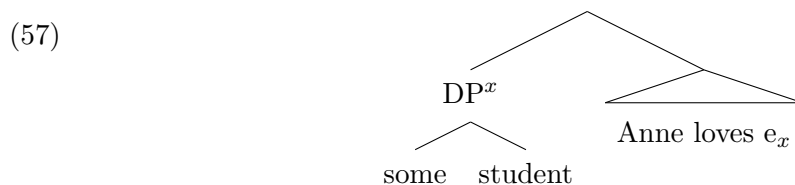
$\exists_{QUES}$, will be a function that takes two proposition-sets as arguments and yields truth if the intersection of these sets is nonempty, and falsehood otherwise – that is, it will be the function of type $\langle\langle\langle s, t\rangle, t\rangle, \langle\langle\langle s, t\rangle, t\rangle, t\rangle\rangle$ described in (58):

(58) $\exists_{QUES} = \lambda P_{\langle\langle s,t\rangle,t\rangle} \lambda P'_{\langle\langle s,t\rangle,t\rangle} \exists p''(P(p'') \wedge P'(p''))$

When we combine $\exists_{QUES}$ with a proposition set (the semantic contribution of a question), we will end up with a set of sets of propositions, or, in other words, a quantifier over propositions. In (55), this quantifier over propositions is the semantic value of the lifted question, just as an analogous quantifier over entities is the semantic contribution of the contribution of the quantifier over entities that is the value of the DP in (57).

Schematically, the general order of composition we get out of $\exists_{QUES}$ will be analogous to what we saw in (55). That is, it will have the general shape seen in (59):

(59)

For a concrete example, let's return to (39). On this account, its order of semantic composition is as in (60), giving rise to the meanings described in (61):

(39)   Alex knows who Anne loves.

(60)

47

(61) $\quad$ i : $\lambda p_{\langle \mathbf{s},\mathbf{t}\rangle}\exists \beta_{\langle \mathbf{e},\mathbf{t}\rangle}(p = \lambda w'_{\mathbf{s}}(\lambda x \left(\begin{array}{c} person(w')(x)\wedge \\ loves(w')(x)(Anne) \end{array} \right) = \beta))$

$\exists_{QUES} : \lambda P_{\langle \langle \mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle}\lambda P'_{\langle \langle \mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle}\exists p''(P(p'') \wedge P'(p''))$

ii : $\lambda P'_{\langle\langle \mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle}\exists p'' \left(\begin{array}{c} \exists \beta (p'' = \lambda w'(\lambda x \left(\begin{array}{c} person(w')(x)\wedge \\ loves(w')(x)(Anne) \end{array} \right) = \beta)) \\ \wedge P'(p'') \end{array} \right)$

$\mathbf{e}_{p'} : p'$

knows : $knows(w^*)$

iii : $knows(w^*)(p')$

Alex : $Alex$

iv : $knows(w^*)(p')(Alex)$

v : $\exists p'' \left(\begin{array}{c} \exists \beta (p'' = \lambda w'(\lambda x \left(\begin{array}{c} person(w')(x)\wedge \\ loves(w')(x)(Anne) \end{array} \right) = \beta)) \\ \wedge knows(w^*)(p'')(Alex) \end{array} \right)$

Note that what we derived as the value of node (v) is synonymous with the value we derived for (39) on the lexical rule approach.

On this approach, the inability of some attitudes, like *'believe'*, to embed questions will again have to be stipulated as a matter of their selectional properties, so, as with the lexical rule approach, the this treatment offers no semantic explanation of why certain propositional predicates cannot be question embedders.

I've framed the above in terms of quantifier raising, but we don't specifically need syntactic quantifier raising for this style of approach: any of the various mechanisms used to do the work that quantifier raising does in Heim and Kratzer (1998) should handle this case without much difficulty.

One possibility raised by this account is the possibility of the lifted question engaging in nontrivial scope interactions by raising to different positions. For the most part, we don't seem to see the relevant scope ambiguities,[12] but see Szabolcsi

---

[12]I will not explore this issue in depth, but I will note one example – in (62), negation always outscopes quantification over answers. That is, (62) can mean something like (63-a), corresponding to the scope relations in (63-b), and does not seem to have a reading like (64-a), corresponding to the scope relations in (64-b):

(1997) for some arguments in favor of scope ambiguities resulting from a similarly quantificational approach to embedded questions.

An appealing feature of the $\exists_{QUES}$ approach is that it makes it easy for our theory of coordination to handle cases like (65), in which a question and a propositional clause are coordinated in an embedded position:

(65)   Rupert knows where Anne went, and that Alex went with her.

Examples like (65) are used by Groenendijk and Stokhof (1984) to advocate for their account (on which the extension of a question is a proposition). On the lexical rule implementation discussed above, we will need some special complication of our theory of coordination to handle such examples, but, on the quantifier raising implementation of question embedding, we only need to borrow whatever grammatical device we use to coordinate names with quantificational DPs. That is because, on this approach, a propositional clause will contribute a proposition (type $\langle \mathbf{s}, \mathbf{t} \rangle$), and a ques-

---

(62)   Alex doesn't know who Anne loves.

(63)   a. $\neg \exists p''(\exists \beta(p'' = \lambda w'(\lambda x \left(\begin{array}{c} person(w')(x) \wedge \\ loves(w')(x)(Anne) \end{array} \right) = \beta)) \wedge knows(w^*)(p'')(Alex))$

     b.

(64)   a. $\exists p''(\exists \beta(p'' = \lambda w'(\lambda x \left(\begin{array}{c} person(w')(x) \wedge \\ loves(w')(x)(Anne) \end{array} \right) = \beta)) \wedge \neg knows(w^*)(p'')(Alex))$

     b.

49
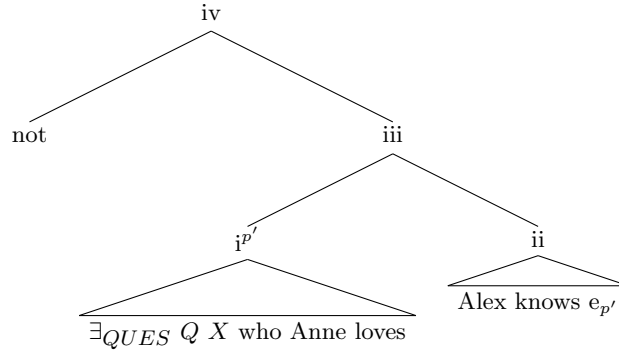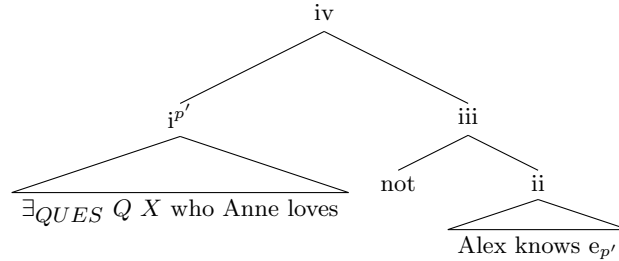
tion will contribute a generalized quantifier over propositions (type $\langle\langle\langle s,t\rangle,t\rangle,t\rangle$). Thus, all we need to do is appeal to a fairly standard type-lifting operator, along the lines of the one give in (66), that repackages something of type τ as something of type $\langle\langle\tau,t\rangle,t\rangle$.

(66) $\lambda\chi_\tau\lambda\xi_{\langle\tau,t\rangle}(\xi(\chi))$

Such a lifting operator, or something that does the work of one, is a reasonably normal assumption for a theory of nominal coordination, allowing a name (type e) to shift up to the type of generalized quantifiers over entities (type $\langle\langle e,t\rangle,t\rangle$) to allow coordination with a quantificational DP, allowing us to assign the correct interpretation to the coordinated DP in the object of (67):

(67) Maggie criticized Rupert and each of Rupert's students.

Thus, the quantifier raising interpretation allows us to analyze these coordinations by generalizing whatever device we assume to handle coordination of differently-typed DPs, without having to develop any new or specialized tools to handle this case.

2.4 Issues in the Semantics of Abstracts

We've already seen the semantics of abstracts in action, and briefly discussed how they are built, but a number of details have been left out. Below, I briefly discuss some other relevant issues.

2.4.1 'wh'-phrases

So far, I have looked at only one 'wh'-phrase: 'who'. I've said that 'who' is associated with type $\langle\langle e,t\rangle,\langle e,t\rangle\rangle$ and has an intension given by (6):

(6) $\lambda w_s\lambda S_{\langle e,t\rangle}\lambda x_e(person(w)(x) \wedge S(x))$

This is not the whole story for 'wh'-phrases. These phrases may have rich internal structure, and come in many different types. Even for a monomorphemic nominal

'wh'-phrases like 'who', things are complicated, and there is a lot more context sensitivity and ambiguity than is captured by (6) above. Some of the more important issues in the meaning of 'wh'-phrases are briefly discussed below.[13]

2.4.1.1 Some Other *'wh'*-Phrases

Recall the general scheme for binding a 'wh'-phrase into a clause, given by (10):

(10)

This scheme will produce a useable abstract meaning in any case where two conditions are met: the 'wh'-phrase must have type $\langle\langle\sigma, \mathtt{t}\rangle, \langle\tau, \mathtt{t}\rangle\rangle$, for some types σ and τ, and the 'wh'-phrase much bind a trace of type σ. (For the cases discussed here, it will be natural to set $\tau = \sigma$, but no formal consideration requires this.) For different 'wh'-words, the exact choice of type will vary. For 'where', for example, τ (and σ) could naturally be analyzed as the type of locations (the details will depend on our syntax and semantics of locative constructions). For single-word 'wh'-phrases, we can individually code meanings, using the analysis of 'who' in (6) as our model.

For complex 'wh'-phrases, such as 'which' phrases, we'll need to assign a semantics to a 'wh'-word in a way that allows it to form a 'wh'-phrase with suitable meaning. 'which', for example, will be a'wh'-determiner (in the sense that it takes a noun phrase argument, and gives rise a 'wh'-phrase associated with a DP-like trace), mapping a noun phrase extension (i.e. a set) to something of the same type as 'who', but with the noun phrase extension substituted for the personhood requirement. That is, 'which' will have the semantic value given in (68).[14]

---

[13]One important issue that I plan to mostly ignore is limitations on 'wh'-phrase meanings. the only major requirements for a 'wh'-phrase to play well with the rest of the theory are that it must be of type $\langle\langle\sigma, \mathtt{t}\rangle, \langle\tau, \mathtt{t}\rangle\rangle$, for some σ and τ, and must leave a trace of type σ. The 'wh'-phrase meanings given here mostly have relatively little information content, serving mainly to conjoin a restrictor onto a property, but I will not explicitly articulate any constraint on the lexical semantics of 'wh'-words. Some such constraint would be desirable, but the identification of a suitable one is left for another time.

[14]This is probably not the whole story of 'which', but it will suffice for now.

(68) $\lambda S_{\langle e,t \rangle} \lambda S'_{\langle e,t \rangle} \lambda x_e (S(x) \wedge S'(x))$

To see (68) in action, consider the abstract in (69), which will be structured as in (70), and interpreted as in (71)

(69)   Which teacher does Red admire?

(70)
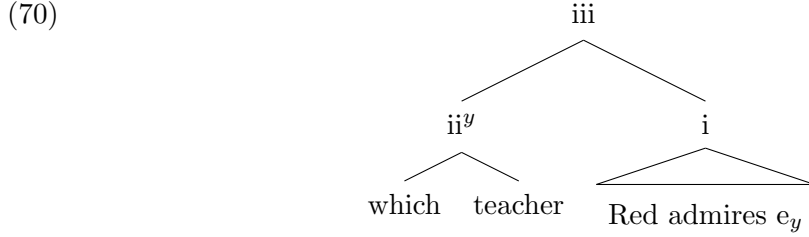
(71)   i : $admires(w)(y)(Red)$

which : $\lambda S_{\langle e,t \rangle} \lambda S'_{\langle e,t \rangle} \lambda x_e (S(x) \wedge S'(x))$

teacher : $teacher(w)$

ii : $\lambda S'_{\langle e,t \rangle} \lambda x_e (teacher(w)(x) \wedge S'(x))$

iii : $\lambda x_e (teacher(w)(x) \wedge admires(w)(x)(Red))$

That is, the extension of the abstract in a world will be the set of teachers that Red admires in that world.

Meanings for other 'wh'-words will be given when they are needed, but the examples of 'which' and 'who' should provide a sense of the general approach.

2.4.1.2   Domain Restrictions

Like a quantificational expression, 'who' (or any other 'wh'-phrase) may have a covert domain restriction. This can be seen in the example of question (72-a), which, in a typical context, is likely to be understood as something like (72-b), but not (72-c): the 'among our friends' is filled in by context.

(72)   a.   Who has a truck?
         b.   Who among our friends has a truck?

c.   Who <u>among all the people in the world</u> has a truck?

The workings of contextually determined domain restrictions raise many important issues for semantics, but these are not distinctive to the semantics of questions, so I will ignore them, hoping that any good theory of domain restriction in other areas will work for *'wh'*-phrases as well. When domain restrictions are important, I will just assume that context supplies a property to serve as the domain restriction, and represent the property with some suitable symbol (usually C) in the representation. Thus, the usual intension of *'who'* will be represented by (73)

(73) $\lambda w_\mathsf{s} \lambda S_{\langle \mathsf{e}, \mathsf{t} \rangle} \lambda x_\mathsf{e} (person(w)(x) \wedge C(w)(x) \wedge S(x))$

I assume C is a property (intensional) rather than a set (extensional) because I think it is possible to restrict the domain by a property the extension of which is not known to the conversational participants, and which may vary with an intensional context in the sentence. This can be seen with domain restrictions in ordinary nominal quantification. Consider, for example, (74): I claim that, in the case where *'everybody'* understood, with restriction, as *everybody in the course*, it is possible for (74) to be true, and that this possibility relies on the silent *'in the course'* restriction being read intensionally:

(74)   Alex can't name or identify a single person in the course, but he does know that everybody showed up for the final on time, and that only students in the course showed up for the final.

If the restriction of *everybody* to people enrolled in the course is understood extensionally, then it should be transparent with respect to *know*. So, if only Anne and Red are enrolled, (74) should allow substitution to give us (75):

(75)   Alex can't name or identify a single person in the course, but he does know that both Anne and Red showed up for the final on time, and that only students in the course showed up for the final.

In a case where Anne and Red are in fact enrolled in the course, (75) is problematic in a way that (74) is not. This is explained if the domain restriction is intensional, since in that case knowing that every person in the domain showed up for the final on time doesn't require the ability to identify even a single person who showed up for the final.

2.4.1.3   Variation in Types for Nominal *'wh'*-Questions

Beyond domain restrictions, there are variants of *'who'* that abstract over types other than entities. Numerous examples exist the need to extend *'who'* to various other types. First, as Aloni (2002) has noted, we need some freedom to let different kinds of guises of entities (implemented as individual concepts in Aloni's account) be sufficient to answer questions. For example, depending on circumstances, acceptable answers to (76) will often include (77-a), and sometimes include (77-b), but almost never include (77-c), while answers to (78) can include (79-c) or (79-b), but seldom (79-a).

(76)   Who is Rupert?

(77)   a.   [pointing] That's Rupert.
        b.   Rupert is the librarian.
        c.   Rupert is Rupert

(78)   [pointing] Who's that guy?

(79)   a.   [pointing] That's that guy.
        b.   That's the librarian.
        c.   That's Rupert.

If a name (like *'Rupert'*) is a rigid designator that directly identifies an individual, then the question (76) should be vacuous or nearly vacuous (if *'Rupert'* denotes the same entity in all worlds, propositions about which *entity* Rupert is will tend to be trivially true or trivially false, except to the extent that their domain restrictions are nontrivial). If *'Rupert'* is not rigid, then, on the assumption that *'who'* abstracts over

entities, it is not clear why (79-c) is a good answer to (78) (since, if (78) is interpreted as *'which entity is that?'*, then an identification of a particular entity ought to be required.). There are many possible replies to this, which I will not explore in depth, but Aloni (2002) argues, I think convincingly, that the right response to this is to build our questions from abstracts whose intensions are not properties of entities, but are instead properties of *guises* of entities. The intuition is that our semantics should allow (78) to be ambiguous between readings like *'which salient name denotes that?'* and *'which definite description of an appropriate form denotes that?'*. There will always be a restriction to some particular domain of guises, but there will be some freedom as to the choice.

It is not, of course, immediately clear how to implement the idea of a guise of an entity. Aloni uses individual concepts (functions of type $\langle \mathtt{s}, \mathtt{t} \rangle$), while Cumming (2006) instead uses discourse referents.[15] It is also not clear what kinds of restrictions we want to place on which guises are available as answers under which circumstances. These issues will not bear on my main concerns, so I will ignore them. The reader is referred to Aloni (2002) for one approach to these problems.

Another issue is functional readings of *'wh'*-questions. Such readings are exemplified by the way that (80-b) can serve as an answer to (80-a):

(80)   a.   What did every guest bring to the party?
        b.   Every guest brought their favorite dish.

According to our semantics as developed so far (80-b) is not a good answer to (80-a), at least on the assumption that *'what'* abstracts over entities (that is, the type of *'what'* is $\langle \langle \mathtt{e}, \mathtt{t} \rangle, \langle \mathtt{e}, \mathtt{t} \rangle \rangle$). If *'what'* abstracts over entities, the meaning of the abstract in (80-a) will be the property of being a thing that was brought by *every* guest. (80-b) doesn't identify even one thing as having been brought by every guest, nor does it provide the information that there is no such thing, so it furnishes neither a mention-some nor a strongly exhaustive answer to (80-a).

---

[15]These short characterizations of the two proposals are both inadequate and somewhat misleading. Both proposals are situated in research programs that seek to address many aspects of the way that the semantics deals with entities and reference to entities, and a thorough description of these projects will not be attempted here.

If *'what'* is considered as abstracting not over entities, but over some kind of relationships between entities, (80-b) might well be a suitable answer. In particular, consider the case where *'what'* abstracts over intensions for functions from entities to entities (type $\langle \mathsf{s}, \langle \mathsf{e}, \mathsf{e} \rangle \rangle$), and that its domain restriction limits us to functions drawn from a restricted class of salient, natural, and intuitively simple ones. In this case, (80-b) will be in the answer set, provided that the function that maps every world w and entity x to the favorite dish of x in w is among the functions in the restriction of *'what'*. I won't explore this issue further, but assume that nominal *'wh'*-phrases can abstract over some suitable type of functions or relations. The reader is referred to the discussion in Groenendijk and Stokhof (1984) and Chierchia (1993) for examples of more developed accounts along these lines.

Spector (2007) has noted yet another case of higher-type abstraction with *'who'* – one that can be seen by considering (81):

(81)   Which books does Anne have to read?

Suppose that Anne's assigned reading list gives some choice: she must read either *Moby Dick* or *Cat's Cradle*, but she may pick either one, and need not read both. In this case, there is not any book with the property that Anne must read that book, so we would expect the extension of the abstract to be empty. This would mean that the truth (82) would suffice to make (83) true.

(82)   Red knows that for no book is it the case that Anne has to read that book.

(83)   Red knows which books Anne has to read.

There might be a reading of (83) for which (82) makes (83) true under the circumstances described, but, if there is, it is not the typical reading. Instead, (83) seems to require that Red know which disjunction of books Anne has obligations about. Spector proposes a way of handling this by associating *'which books'* not with abstraction over entities, but instead with abstraction over generalized quantifiers over entities. Again, this does not bear directly on my concerns, so, having noted it, I will ignore it going forward.

2.4.1.4 'de re' and 'de dicto' Readings of a *'wh'*-Phrase

In the preceding examples, I have assumed that world arguments associated with a *'wh'*-phrase are filled with the same world variable associated with the clause that it is quantified into, giving rise to a 'de dicto' reading of the *'wh'*-phrase with respect to the embedding context. On this 'de dicto' reading, for (84) to be true, it is necessary that Alex know a proposition that identifies some set as the set of psychologists that Anne loves, or (on a mention-some reading) that identifies some entities as psychologists that Anne loves. In either case, the proposition that Alex knows must, loosely speaking, have the concept of psychologist-hood as one of its parts.

(84)   Alex knows which psychologists Anne loves.

Considering the exhaustive case, using the same world variable for *'psychologists'* and *'loves'*, we interpret (85) as in (86) (where w^* is the actual world):

(85)
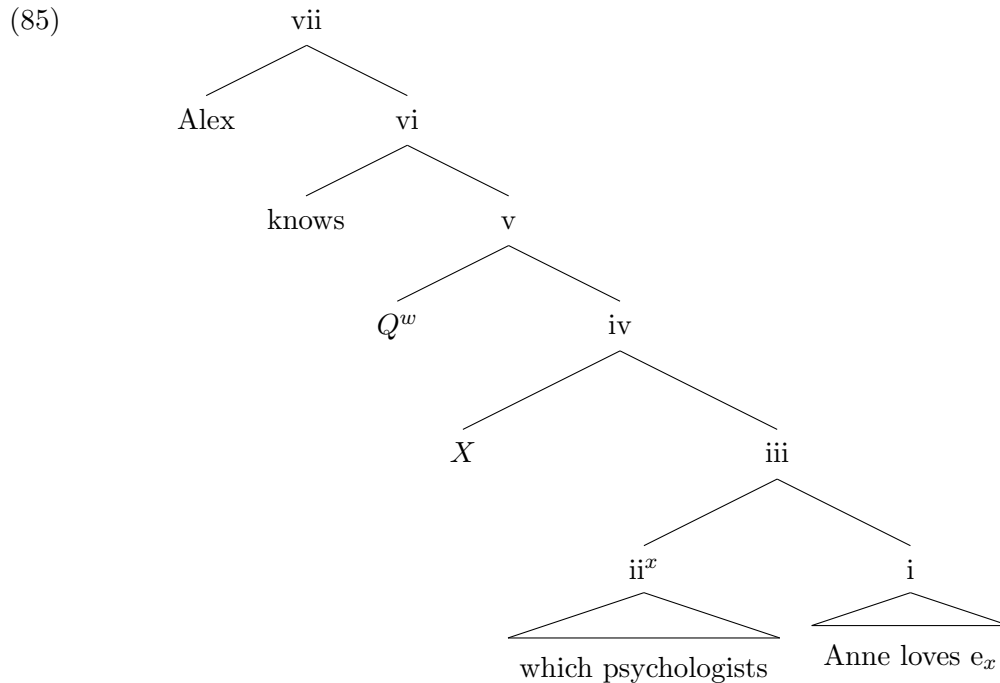
(86)   i : $loves(w)(x)(Anne)$

$$\text{ii} : \lambda S_{\langle \mathsf{e},\mathsf{t}\rangle}\lambda y_{\mathsf{e}}(psychologist(w)(y) \wedge S(y))$$

$$\text{iii} : \lambda y_{\mathsf{e}}(psychologist(w)(y) \wedge loves(w)(y)(Anne))$$

$$X_{\langle \mathsf{e},\mathsf{t}\rangle} : \lambda \gamma_{\langle \mathsf{e},\mathsf{t}\rangle}\lambda \delta_{\langle \mathsf{e},\mathsf{t}\rangle}(\gamma = \delta)$$

$$\text{iv} : \lambda \delta_{\langle \mathsf{e},\mathsf{t}\rangle}(\lambda y_{\mathsf{e}}(psychologist(w)(y) \wedge loves(w)(y)(Anne)) = \delta)$$

$$Q_{\langle \mathsf{e},\mathsf{t}\rangle} : \lambda \alpha_{\langle \mathsf{s},\langle\langle \mathsf{e},\mathsf{t}\rangle,\mathsf{t}\rangle\rangle}\lambda p_{\langle \mathsf{s},\mathsf{t}\rangle}\exists \beta_{\langle \mathsf{e},\mathsf{t}\rangle}(p = \lambda w'_{\mathsf{s}}(\alpha(w')(\beta)))$$

$$\text{v} : \lambda p \exists \beta_{\langle \mathsf{e},\mathsf{t}\rangle}\left(p = \lambda w'_{\mathsf{s}}\left(\lambda y \begin{pmatrix} psychologist(w')(y) \\ \wedge loves(w')(y)(Anne) \end{pmatrix} = \beta\right)\right)$$

$$\text{knows} : know_{QUES}(w^*)$$
$$= \lambda P_{\langle\langle \mathsf{s},\mathsf{t}\rangle,\mathsf{t}\rangle}\lambda x_{\mathsf{e}}\exists p'_{\langle \mathsf{s},\mathsf{t}\rangle}(P(p') \wedge knows_{PROP}(w^*)(p')(x))$$

$$\text{vi} : \lambda x \exists p' \left(\begin{array}{c} \exists \beta_{\langle \mathsf{e},\mathsf{t}\rangle}\left(p' = \lambda w'_{\mathsf{s}}\left(\lambda y \begin{pmatrix} psychologist(w')(y) \\ \wedge loves(w')(y)(Anne) \end{pmatrix} = \beta\right)\right) \\ \wedge knows_{PROP}(w^*)(p')(x) \end{array} \right)$$

$$\text{Alex} : Alex$$

$$\text{vii} : \exists p' \left(\begin{array}{c} \exists \beta_{\langle \mathsf{e},\mathsf{t}\rangle}\left(p' = \lambda w'_{\mathsf{s}}\left(\lambda y \begin{pmatrix} psychologist(w')(y) \\ \wedge loves(w')(y)(Anne) \end{pmatrix} = \beta\right)\right) \\ \wedge knows_{PROP}(w^*)(p')(Alex) \end{array} \right)$$

This reading is the most natural one for the above example, but, as has been argued elsewhere (see, e.g., Groenendijk and Stokhof (1984) and Beck and Rullmann (1999)), it is not the only one. Suppose, for example, that there two salient psychologists: Riley and Maggie, and that the only psychologist that Anne loves is Riley. On the reading given in (86), the true strongly exhaustive answer to (87) in this situation is (88):

(87)   Which psychologists does Anne love?

(88)   Riley is the only psychologist Anne loves.

Now suppose that Alex knows that Anne loves Riley and does not loves Maggie, but that Alex does not know that either Riley or Maggie is a psychologist. This will make (90) false, and, so, according to the truth conditions given in (86), it will make

(84) false as well.[16]

(90)   Alex knows Riley is the only psychologist Anne loves.

In this situation, where Alex knows that Anne loves Riley but not Maggie, but does not know who the psychologists are, we observe that (84) can still be judged true, but, as we just saw, it is untrue on the reading given by (vii) of (86).

To get the reading where Alex must know that Riley is a psychologist to make (84) true, we filled the world argument of *psychologist* with the world variable bound by the question operator Q, giving us a 'de dicto' interpretation of *'psychologist'*. We can also derive a 'de re' reading by instead evaluating *'psychologist'* with respect to the actual world w^*, giving (84) the truth-conditions in (91):

$$
(91) \quad \exists p' \left(\begin{array}{c} \exists \beta_{\langle \mathbf{e}, \mathbf{t} \rangle} (p' = \lambda w'_\mathbf{s} (\lambda y \left(\begin{array}{c} psychologist(w^*)(y) \\ \wedge loves(w')(y)(Anne) \end{array} \right) = \beta)) \\ \wedge knows_{PROP}(w^*)(p')(Alex) \end{array} \right)
$$

On the 'de dicto' reading, the answers to (87) are propositions that identify some set as the set of psychologists that Anne loves (i.e. (92) is the formula for the answerhood conditions). On the 'de re' reading, on the other hand, to figure out the answerhood conditions we first get the set of psychologists from the actual world, and then the answers are the propositions that tell us exactly which members of that set that Anne loves (i.e. (93) is the formula for the answerhood conditions).

$$
(92) \quad \lambda p \exists \beta_{\langle \mathbf{e}, \mathbf{t} \rangle} (p = \lambda w'_\mathbf{s} (\lambda y \left(\begin{array}{c} psychologist(w')(y) \\ \wedge loves(w')(y)(Anne) \end{array} \right) = \beta))
$$

$$
(93) \quad \lambda p \exists \beta_{\langle \mathbf{e}, \mathbf{t} \rangle} (p = \lambda w'_\mathbf{s} (\lambda y \left(\begin{array}{c} psychologist(w^*)(y) \\ \wedge loves(w')(y)(Anne) \end{array} \right) = \beta))
$$

---

[16]This scenario will also make (84) false on a mention-some reading, since the only true mention-some answer to (87) is (89), and Alex doesn't know that either:

(89)   Riley is a psychologist Anne loves.

That is, on the 'de re' reading, the only true strongly exhaustive answer to the question in the scenario we've been considering is the proposition given by (94) or, equivalently, (95):

(94)   Among Riley and Maggie, Anne only loves Riley.

(95)   Anne loves Riley and does not love Maggie.

For us, the crucial difference between the *'de dicto'* and *'de re'* answers is that the former, but not the latter, entails (96):

(96)   Riley is a psychologist.

In our scenario, Alex does know (95), so (84) will be true on a 'de re' reading, explaining the judgment that (84) is true on at least one reading.

This ambiguity is sometimes important, but it will not be my main focus. For this reason, I will generally just use a 'de dicto' reading of the restrictor of a *'wh'*-phrase, considering a 'de re' reading only when it is relevant to our other concerns. I will also completely ignore some thorny issues in how world variables get bound, and why it is that the Q operator (like other many intensional operators) seems to always abstract over the world variable associated with the main verb of the embedded sentence. I leave this to the general theory of world variable binding, hoping that the world variables in *'wh'*-phrases will turn out to behave similarly enough to the world variables in quantificational phrases. The reader wishing to learn more about these issues is referred to Groenendijk and Stokhof (1984), Beck and Rullmann (1999), and Sharvit (2002) for some discussion of this ambiguity in embedded questions.

2.4.2   Multiple-*'wh'* Abstracts

If we apply the scheme in (11) to the second *'who'* in (5-a), things go awry, for reasons related to the limitations of the binding in operation as developed so far.

(5-a)   Who killed who?

We should presumably analyze the compositional relations in (5-a) along the lines given in (97), but when we try to use the (11) scheme to compute the meaning of node (iii), we get type mismatch:

(97)

$$
\begin{array}{c}
\text{iii} \\
\diagup \quad \diagdown \\
\text{who}^z \qquad \text{ii} \\
\qquad \diagup \quad \diagdown \\
\qquad \text{who}^y \qquad \text{i} \\
\qquad\qquad \diagup \quad \diagdown \\
\qquad\qquad \text{e}_y \text{ killed e}_z
\end{array}
$$
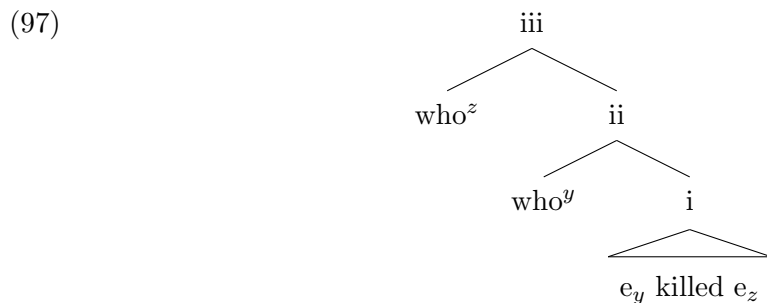
(98)   i : $killed(w)(z)(y)$

who : $\lambda S_{\langle e,t\rangle}\lambda x_e(person(w)(x) \wedge S(x))$

ii : $(\lambda S_{\langle e,t\rangle}\lambda x_e(person(w)(x) \wedge S(x)))(\lambda y_e(killed(w)(z)(y)))$

   =

   $\lambda x_e(person(w)(x) \wedge killed(w)(z)(x))$

iii : $*(\lambda S_{\langle e,t\rangle}\lambda x'_e\left(\begin{array}{c} person(w)(x') \\ \wedge S(x') \end{array}\right))(\lambda z_e\lambda x_e\left(\begin{array}{c} person(w)(x) \\ \wedge killed(w)(z)(x) \end{array}\right))$

   TYPE MISMATCH! Function of type $\langle\langle e,t\rangle,\langle e,t\rangle\rangle$ can't combine with argument of type $\langle e,\langle e,t\rangle\rangle$!

The problem here is that our rule for binding in 'who' for a variable of type e is designed to only work when the sister of 'who' has type t. This works fine for the first 'who', but for the second, we now have a sister of type $\langle e,t\rangle$ which won't work with our rule.

   This is a familiar problem: we want to bind an operator that takes an argument of type $\langle\sigma,\tau\rangle$ in for a variable of type σ when the sister of the operator is not of type τ, but is instead of some time that 'ends in' τ. We encounter the same issue when trying to quantify a nominal quantifier (type $\langle\langle e,t\rangle,t\rangle$) into an expression of type $\langle e,t\rangle$ (the typical extensional type of noun phrases, verb phrases, and, on some accounts, many adjective phrases and prespositional phrases as well) for a variable

of type e (a nominal trace).[17] We know that (iii) in (97) above ought to have an extension something like (99), but our compositional semantics, as developed up to this point, fails to derive it.

(99) $\quad \lambda x_\mathsf{e} \lambda x'_\mathsf{e}(person(w)(x') \wedge person(w)(x) \wedge killed(w)(x')(x))$

That is, when we have the configuration in (100), we want a rule that handles node (iii) as in (101):

(100)

(101) $\quad$ i : φ^{18}

$\qquad$ ii : ψ

$\qquad$ iii : $\lambda v'_\mathsf{e}(\psi(\lambda v \varphi(v')))$

In (97), we have (102-a) for φ and (102-b) for ψ:

(102) $\quad$ a. $\quad \varphi = \lambda x(person(w)(x) \wedge killed(w)(v)(x))$

$\qquad$ b. $\quad \psi = \lambda S_{\langle \mathsf{e},\mathsf{t} \rangle} \lambda x'(person(w)(x') \wedge S(x'))$

Plugging these into (iii) of (101), we get (103):

(103) $\quad \lambda v'_\mathsf{e}((\lambda S_{\langle \mathsf{e},\mathsf{t} \rangle} \lambda x' \begin{pmatrix} person(w)(x') \\ \wedge S(x') \end{pmatrix})(\lambda v(\lambda x \begin{pmatrix} person(w)(x) \wedge \\ killed(w)(v)(x) \end{pmatrix})(v')))$
$\qquad = \lambda v'_\mathsf{e} \lambda x'(person(w)(x') \wedge person(w)(v') \wedge killed(w)(x')(v'))$

Since we've seen that there is considerable variation in type among different *'wh'*-phrases and *'wh'*-abstracts, and because it seems desirable to use the same rule

---

[17]Montague (1974), for example, includes special rules (his rules S15, S16, T15 and T16) for quantifying a nominal quantifier into the categories that correspond to noun phrases ('CN') and verb phrases ('IV') in his system, where these expressions are assigned extensions of type $\langle \langle \mathsf{s}, \mathsf{e} \rangle, \mathsf{t} \rangle$.

[18]In linguistically natural cases, it is to be assumed that v is free in φ, but there is no reason to make this an assumption in defining the rule.

for quantifying into things that end in type $\mathtt{t}$ in non-question contexts, we should generalize the rule in (100) and (101) to any types that stand in a suitable relation, that is, assuming the type labels as in (104), and the meanings as in (i) and (ii) of (105), the semantic contribution of the top node of (104) will be as in (iii) of (105):[19]

(104)

$$\text{iii}_{\langle \sigma', \rho \rangle}$$
$$\text{ii}^{v_\sigma}_{\langle \langle \sigma, \tau \rangle, \rho \rangle} \quad \text{i}_{\langle \sigma', \tau \rangle}$$

(105)   i : φ

      ii : ψ

      iii : $\lambda v'_{\sigma'}(\psi(\lambda v_\sigma \varphi(v')))$

The case in (100) and (101) is just the special case where $\sigma = \sigma' = \mathtt{e}$, $\rho = \langle \mathtt{e}, \mathtt{t} \rangle$, and $\tau = \mathtt{t}$.

To generalize to arbitrary multiple-'*wh*' questions, we will have to allow σ' to range over singleton and tuple types. For example, consider the triple-'*wh*'-question (106):

(106)   Who gave what to who?

The abstract in (106) will be analyzed with (107), and interpreted as in (108):

---

[19]There are, of course, other ways of enriching the formalism to handle these cases. For example, Groenendijk and Stokhof (1984) modify their representation language introduce a general mechanism for marking λ-abstraction with a restriction. Although some of the details are different, their solution and the one presented here can be regarded equivalent for our purposes.

(107)

$$\text{iv}_{\langle(e;e),\langle e,t\rangle\rangle=\langle e,\langle e,\langle e,t\rangle\rangle\rangle}$$

$$\text{who}^{z_e}_{\langle\langle e,t\rangle,\langle e,t\rangle\rangle} \qquad \text{iii}_{\langle e,\langle e,t\rangle\rangle=\langle(e;e),t\rangle}$$

$$\text{what}^{y_e}_{\langle\langle e,t\rangle,\langle e,t\rangle\rangle} \qquad \text{ii}_{\langle e,t\rangle}$$

$$\text{who}^{x_e}_{\langle\langle e,t\rangle,\langle e,t\rangle\rangle} \qquad \text{i}_t$$

$$\text{e}_x \text{ gave e}_y \text{ to e}_z$$
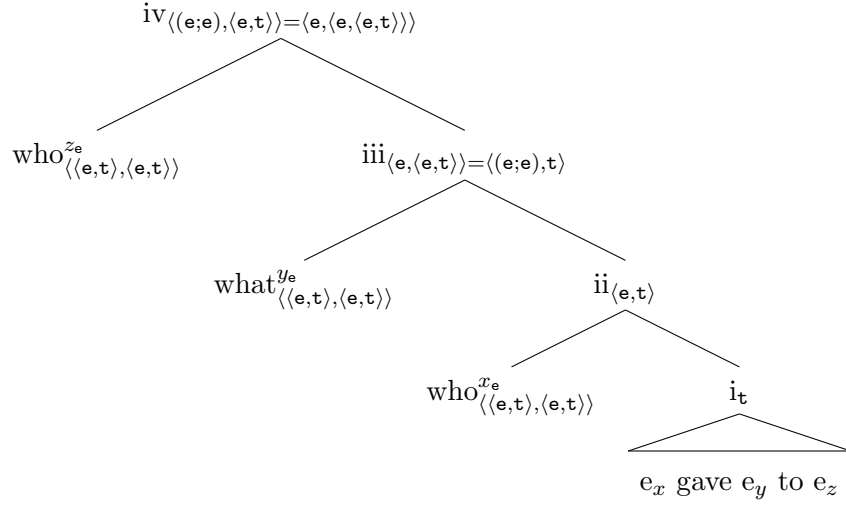
(108) $\text{who} : \lambda S_{\langle e,t\rangle}\lambda x_e(person(w)(x) \wedge S(x))$

$\text{what} : \lambda S_{\langle e,t\rangle}\lambda x_e(S(x))$

$\text{i} : gave(w)(z)(y)(x)$

$\text{ii} : (\lambda S_{\langle e,t\rangle}\lambda x'_e(person(w)(x') \wedge S(x')))(\lambda x(gave(w)(z)(y)(x)))$
$= \lambda x'_e(person(w)(x') \wedge gave(w)(z)(y)(x'))$

$\text{iii} : \lambda v_e((\lambda S_{\langle e,t\rangle}\lambda x''_e(S(x'')))(\lambda y(\lambda x'_e\begin{pmatrix} person(w)(x')\wedge \\ gave(w)(z)(y)(x') \end{pmatrix})(v)))$
$= \lambda v_e\lambda x''_e(person(w)(v) \wedge gave(w)(z)(x'')(v))$

$\text{iv} : \lambda v'_{(e;e)}\begin{pmatrix} (\lambda S_{\langle e,t\rangle}\lambda x'''_e\begin{pmatrix} person(w)(x''') \\ \wedge S(x''') \end{pmatrix}) \\ (\lambda z((\lambda v_e\lambda x''_e\begin{pmatrix} person(w)(v)\wedge \\ gave(w)(z)(x'')(v) \end{pmatrix})(v'))) \end{pmatrix}$

$= \lambda v'_{(e;e)}\lambda x'''_e\begin{pmatrix} person(w)(x''')\wedge \\ ((\lambda v_e\lambda x''_e\begin{pmatrix} person(w)(v)\wedge \\ gave(w)(x''')(x'')(v) \end{pmatrix})(v')) \end{pmatrix}$

$= \lambda v''_e\lambda v'''_e\lambda x'''_e\begin{pmatrix} person(w)(x''')\wedge \\ ((\lambda v_e\lambda x''_e\begin{pmatrix} person(w)(v)\wedge \\ gave(w)(x''')(x'')(v) \end{pmatrix})(v'')(v''')) \end{pmatrix}$

$= \lambda v''_e\lambda v'''_e\lambda x'''_e\begin{pmatrix} person(w)(x''') \wedge person(w)(v'') \\ \wedge gave(w)(x''')(v''')(v'') \end{pmatrix}$

64

That is, the abstract relates v'', v''' and x''' in w if and only if v'' and x''' are people in w, and v'' gave v''' to x''' in w. This works because of our ability in the last step to equivocate between the types $\langle \mathsf{e}, \langle \mathsf{e}, \mathsf{t} \rangle \rangle$ and $\langle (\mathsf{e}; \mathsf{e}), \mathsf{t} \rangle$.

2.5   Summary of the Theory

Having gone through the various details, I want to step back and quickly review the outline of the theory. I began by assuming that *'wh'*-phrases act as restricted λ-abstractors and that questions are built up from more generic *'wh'*-clauses, called 'abstracts', which are derived by binding in *'wh'*-phrases. For example, (109) contains an abstract with value (110-b), derived by the binding in both *'wh'*-phrases as illustrated in (110-a).

(109)   Which librarian betrayed which student?

(110)   a.
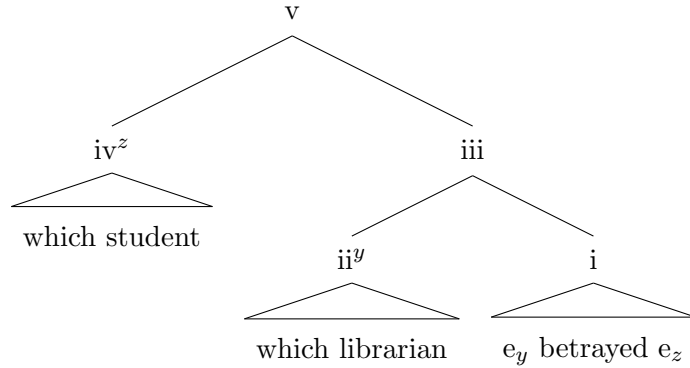
      b. $\lambda y \lambda z(student(w)(z) \wedge librarian(w)(y) \wedge betrayed(w)(z)(y))$

After this, we may optionally apply the X operator, given in (111), to get a strongly exhaustive reading, applying X to (110-b) yields (112).

(111) $X = \lambda \gamma \lambda \delta(\gamma = \delta)$

(112) $\lambda \delta_{\langle \mathsf{e}, \langle \mathsf{e}, \mathsf{t} \rangle \rangle}(\lambda y \lambda z \left(\begin{array}{c} student(w)(z) \wedge librarian(w)(y) \\ \wedge betrayed(w)(z)(y) \end{array} \right) = \delta)$

Next, we apply the Q operator in (113), abstracting over the main world variable of the abstract, to derive a set of answer propositions. This can be done with or without X. Q applied to (110-b) gives us the mention-some answer set (114-a), while Q applied to (112) gives us the strongly exhaustive answer-set (114-b).

(113) $Q = \lambda\alpha\lambda p_{\langle\mathbf{s},\mathbf{t}\rangle}\exists\beta(p = \lambda w'_{\mathbf{s}}(\alpha(w')(\beta)))$

(114)   a. $\lambda p_{\langle\mathbf{s},\mathbf{t}\rangle}\exists\beta_{\mathbf{e}}\beta'_{\mathbf{e}}(p = \lambda w'_{\mathbf{s}}\left(\begin{array}{c} student(w')(\beta') \wedge librarian(w')(\beta) \\ \wedge betrayed(w')(\beta')(\beta)\end{array}\right))$

     b. $\lambda p_{\langle\mathbf{s},\mathbf{t}\rangle}\exists\beta_{\langle\mathbf{e},\langle\mathbf{e},\mathbf{t}\rangle\rangle}(p = \lambda w'_{\mathbf{s}}(\lambda y\lambda z\left(\begin{array}{c} student(w')(z)\wedge \\ librarian(w')(y)\wedge \\ betrayed(w')(z)(y)\end{array}\right) = \beta))$

Formulas (114-a) and (114-b) represent the two main kinds of answerhood conditions for (109), although complications can arise from domain restrictions, and different handling of the world arguments can give rise to 'de re' readings.

Finally, we have a theory of embedding. For responsive embedders (which is the main case of interest for us), this theory says that a responsive predicate is true of a question if it is true of at least one element of its answer set – thus (115) is true if and only if (on a strongly exhaustive reading) William knows one of the propositions in (114-b), or (on a mention-some reading) William knows one of the propositions in (114-a).

(115)   William knows which librarian betrayed which student.

This theory assumes one main ambiguity: the ambiguity between mention-some and strongly exhaustive readings of a question. So far, its account of responsive embedders is uniform: both readings are assumed to always be available, and both are used in the same way when combining with any responsive embedder. The theory, like most accounts of question embedding, handles all responsive embedders reductively – we determine whether a responsive embedder is true of a question by looking at what propositions it is true of, so, for example, the truth (115) depends only on what propositions answer the embedded question, and what propositions William knows. In subsequent chapters, all of these aspects of the theory will be

re-evaluated, but it is helpful to have a starting point. The issue of generalizing this theory to non-'*wh*' questions will be taken up in chapter 5. Before turning to the empirical assessment of the theory, I want to briefly discuss some notions of answerhood not used by this theory, but potentially useful in extending the theory or comparing it with competitors.

2.6   Other Notions of Answerhood

The system developed here deals in two kinds of answers: mention-some answers and strongly exhaustive answers. In both cases it does not privilege true answers (unlike Groenendijk and Stokhof (1984) and Karttunen (1977), among others), so (except for 'de re' effects) the answer set for a question will be the same in all possible worlds.

As I have already indicated, some other notions of answerhood appear in the literature. Having concluded the main thread of this chapter above, I want to take the time to give definitions of these notions in terms of my framework, to make theoretical comparison easier, and to situate my approach in the literature. These alternatives do not form part of the baseline theory, but are provided as approximations of competing theories, and as possible extensions of the baseline theory.

2.6.1   Partial Answers

The notion of a 'partial answer' is discussed by Groenendijk and Stokhof (1984) and Lahiri (2002), among others. Below, I will give a definition of partial answer with respect to a particular answer set. The definition given here does not precisely coincide with definitions given elsewhere, but does respect the idea underlying those definitions, which is that a partial answer is a proposition that rules out some possible answers. I will use two notions of partial answer set: the 'generalized partial answers' and the 'strict partial answers'. The idea of the generalized partial answers is simple: the set of generalized partial answers is the closure of the answer set under (finite and infinite) disjunction. The strict partial answers are what we get when we cull certain pathological cases from the generalized partial answers: In particular, a (generalized) partial answer is 'nontrivial' if it is not the one derived by taking disjunction over the empty set (that would be a contradiction) or the one derived by taking disjunction

over the whole answer set.[20] The operator that maps an answer set onto the set of generalized partial answers for that answers set, will be called $\underline{\textsc{Part}}$ (compare **Part** for the strict partial answers), and is defined in (116).[21]

$$(116) \quad \underline{\textsc{Part}} = \lambda P_{1,\langle\langle\mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle} \lambda p' \exists P_{2,\langle\langle\mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle} \left(\begin{array}{c} P_2 \subseteq P_1 \wedge \\ \forall w_{\mathbf{s}} \left(\begin{array}{c} p'(w) \equiv \\ \exists p''(p''(w) \wedge P_2(p'')) \end{array} \right) \end{array} \right)$$

$\underline{\textsc{Part}}$ takes as its argument a set of propositions and returns the set of all propositions that can be derived by first taking a subset of that set, and then taking the disjunction of all propositions in that subset (metaphorically, erasing the boundaries between them to get one big set of worlds).

A partial answer set can be computed with respect to either a mention-some answer set or a strongly exhaustive answer set. The 'partial answers' of Groenendijk and Stokhof (1984) are partial answers with respect to the strongly exhaustive answer set, but there is no reason we can't apply the notion to mention-some answer sets as well.[22] A few formal observations about partial answers may help to cultivate an intuition about how the $\underline{\textsc{Part}}$ operator behaves:

(117) a. For all $P_{\langle\langle\mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle}$, $\underline{\textsc{Part}}(P) = \underline{\textsc{Part}}(\underline{\textsc{Part}}(P))$
           ($\underline{\textsc{Part}}$ is idempotent.)

       b. If $P \subseteq P'$ then $\underline{\textsc{Part}}(P) \subseteq \underline{\textsc{Part}}(P')$
           ($\underline{\textsc{Part}}$ is monotone.)

       c. For all P, $P \subseteq \underline{\textsc{Part}}(P)$

       d. For all types τ and all $\alpha_{\langle\mathbf{s},\langle\tau,\mathbf{t}\rangle\rangle}$, $Q(\alpha) \subseteq \underline{\textsc{Part}}(Q(\lambda w(X(\alpha(w)))))$
           (Every mention-some answer is a partial answer with respect to the strongly exhaustive answer set.)

---

[20]This latter disjunction will lead to a tautology under certain circumstances, such as when we start with a set of strongly exhaustive answers not laden with presuppositions, but not under all circumstances

[21]Here '$P_2 \subseteq P_1$' is shorthand for '$\forall p(P_2(p) \to P_1(p))$'.

[22]However, although this notion of partial answerhood is the most natural one in the strongly exhaustive case, it is not clear that it is the best definition in the mention-some case. This is, of course, hard to evaluate in the absence of a particular proposal for using partial answers in either the pragmatics or the compositional semantics.

As already noted, $\underline{\text{PART}}$ includes some pathological cases – including the contradictory proposition and the proposition that is true iff any proposition in the answer set is true. Most of the time, we will be interested in the nontrivial partial answers, so I will define the operator **Part** to derive the set of strict partial answers:

(118) $\mathbf{Part} = \lambda P_{\langle\langle\mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle} \lambda p' \left(\begin{array}{c} \underline{\text{PART}}(P)(p') \wedge \\ \exists w(p'(w)) \wedge \\ \exists w' \exists p''(P(p'') \wedge p''(w) \wedge \neg p'(w)) \end{array} \right)$

That is, **Part**(P) is the set of all non-contradictory propositions in $\underline{\text{PART}}(P)$ that rule out at least one possible world that would not be ruled out by taking the disjunction of all propositions in P. Among other things, this latter condition excludes tautologies from the set of nontrivial partial answers.

What sorts of partial answers does **Part** this operation give us? This depends considerably on what kind of answer set we start with. For a mention-some question like (119), the answer set consists of propositions of the shape informally given by (120):

(119)   Where can Rupert buy an Italian newspaper?

(120)   Rupert can buy an Italian newspaper at x.

The $\underline{\text{PART}}$ operator gives us the set of all disjunctions over sets of propositions with the shape given in (120), so it gives us an answer set that includes (121-a), (121-b), something like (121-c) (the disjunction of all mention-some answers besides (121-a)), and also the disjunction of *all* mention-some answers (at least roughly equivalent to (121-d)) and the disjunction over the empty set of answers (that is, the proposition that is false in every world, given by the expression (121-e)[23]):

(121)   a.   Rupert can buy an Italian newspaper at PaperWorld.
         b.   Rupert can buy an Italian newspaper at PaperWorld or Newstopia.

---

[23]Here, $\lambda w(0)$ means the constant function that takes one world argument and always returns the value 0 (false).

    c.   Rupert can buy an Italian newspaper at at least one place that is not PaperWorld (and maybe at PaperWorld as well).

    d.   Rupert can buy an Italian newspaper somewhere.

    e. $\lambda w(0)$

The **Part** operator adds to the $\underline{\text{PART}}$ operator a restriction that excludes the contradictory proposition (121-e) and the disjunction over all answers (121-d), but leaves in other partial answers, such as (121-a), (121-b), and (121-c).

    Turning to the strongly exhaustive case, things get a bit more complex. Consider for example the question (1-a). The answers to this question, on a strongly exhaustive reading, are the propositions of the form (122).

(1-a)   Who does Anne love?

(122) S is the set of people that Anne loves.

The strongly exhaustive answers famously partition the space of worlds (give or take some issues with presupposition): two distinct propositions of the shape given in (122) are mutually contradictory, and every world (at least, every world in which Anne exists) satisfies some proposition of this shape. For this reason, applying $\underline{\text{PART}}$ to the strongly exhaustive answer set of (1-a) gives us both the contradictory proposition (123-a) (the disjunction over the empty set of strongly exhaustive answers) and the tautologous proposition (123-b) (the disjunction over all strongly exhaustive answers). The nontrivial partial answers include any single strongly exhaustive answer, like (123-c), and any finite disjunction of strongly exhaustive answers, like (123-d), but also various propositions derivable as infinite disjunctions over strongly exhaustive answers, such as (123-e) (the disjunction over all strongly exhaustive answers except (123-c)), (123-f) (the disjunction over all strongly exhaustive answers that exclude Liam from the set of people loved by Anne), (123-g) (the disjunction over all strongly exhaustive answers that include Liam among the people loved by Anne), and (123-h) (the disjunction over all strongly exhaustive answers except those that include Alex in the set of people loved by Anne, but exclude Red).

(123)   a. $\lambda w(0)$

      b. $\lambda w(1)$

      c.   Liam is the only person Anne loves.

      d.   Liam is the only person Anne loves, or Red and Joyce are the only people Anne loves.

      e.   It is not the case that Liam is the only person Anne loves.

      f.   Anne does not love Liam.

      g.   Anne loves Liam.

      h.   If Anne loves Alex, then she also loves Red.

Applying **Part** instead of PART cuts out (123-a) and (123-b), but leaves in the other other answers in (123).

For the most part, theories of embedding don't do much with partial answers beyond the accustomed mention-some and strongly and weakly exhaustive answers (van Rooij (2004) and Ginzburg (1995a,b) are exceptions, although they frame the issue in different terms). However, partial answers are sometimes thought to be conceptually appealing, and are probably valuable for modeling the pragmatics of the conversational question-answer relationship (this is their primary purpose in Groenendijk and Stokhof (1984)). In real conversations, where time is short and uncertainty unavoidable, questions rarely receive complete answers, and so a notion of what counts as 'part of an answer' has many potential applications.

2.6.2   *The* Answer(s)

Many prominent accounts of question embedding, including Karttunen (1977), Groenendijk and Stokhof (1984), and Heim (1994), privilege true answers. In particular, the idea of strong exhaustivity is traditionally tied to truth, although Egré and Spector (2007), Sharvit (2002), and Lahiri (2002) have advocated for notions of strong exhaustivity that do not have a built-in truth requirement. It has been traditional to speak of *the* strongly exhaustive answer to a question – that is, the unique true proposition in the strongly exhaustive answer set, and, similarly, Karttunen's mention-some answer sets contained only true mention-some answers. I will not adopt this approach, but I want to briefly sketch how such an approach would be framed in terms of abstracts.

Since which propositions are true will vary from world to world, the notion 'the answer' will always be contingent on a world of evaluation. For any answer set P and any world w, the set of true answers for P in w is given by (124):

(124) $\lambda p'(P(p') \wedge p'(w))$

Since the true strongly exhaustive answer is unique, accounts that privilege the true strongly exhaustive answer (e.g. Groenendijk and Stokhof (1984) and Heim (1994)) have generally not defined a set of true strongly exhaustive answers, but directly defined *the* true strongly exhaustive answer. For purposes of illustratiion, I present Groenendijk and Stokhof's rule for defining the true strongly exhaustive answer directly from the abstract. The operator to do this, **Strong**, is given in (125). It can be defined for arguments of any intensional type $\langle \mathbf{s}, \tau \rangle$, and it requires us to provide a world argument to specify which world's strongly exhaustive answer it's supposed to provide.

(125)   **Strong** $= \lambda w \lambda \alpha_{\langle \mathbf{s}, \tau \rangle} \lambda w'(\alpha(w) = \alpha(w'))$

That is, for any world and any abstract, we can compute the strongly exhaustive answer in that world by taking the extension of the abstract in that world, and building the proposition that is true in exactly those worlds where that extension is the extension of the abstract under consideration.

Using (125), we would compute the truth-conditions of (39) with the series of combinations indicated in (126), and the interpretations in (127) (this is basically a notational variant on the approach seen in Groenendijk and Stokhof (1984)).

(39)   Alex knows who Anne loves.

(126)
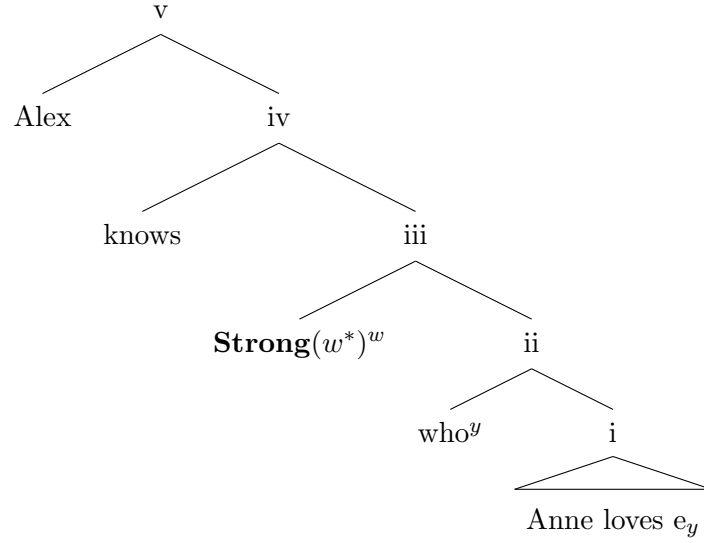
(127) $i : loves(w)(y)(Anne)$

$who : \lambda S_{\langle e,t\rangle}\lambda x_{e}(person(w)(x) \wedge S(x))$

$ii : \lambda x_{e}(person(w)(x) \wedge loves(w)(x)(Anne))$

$\mathbf{Strong}(w^*) : \lambda \alpha_{\langle s,\langle e,t\rangle\rangle}\lambda w'(\alpha(w^*) = \alpha(w'))$

$iii : \lambda w' \left(\begin{array}{l} \lambda x_{e}(person(w^*)(x) \wedge loves(w^*)(x)(Anne)) \\ = \lambda x_{e}(person(w')(x) \wedge loves(w')(x)(Anne)) \end{array} \right)$

$knows : knows(w^*)$

$iv : knows(w^*)(\lambda w' \left(\begin{array}{l} \lambda x_{e} \left(\begin{array}{c} person(w^*)(x)\wedge \\ loves(w^*)(x)(Anne) \end{array} \right) \\ = \lambda x_{e} \left(\begin{array}{c} person(w')(x)\wedge \\ loves(w')(x)(Anne) \end{array} \right) \end{array} \right))$

$Alex : Alex$

$v : knows(w^*)(\lambda w' \left(\begin{array}{l} \lambda x_{e} \left(\begin{array}{c} person(w^*)(x)\wedge \\ loves(w^*)(x)(Anne) \end{array} \right) \\ = \lambda x_{e} \left(\begin{array}{c} person(w')(x)\wedge \\ loves(w')(x)(Anne) \end{array} \right) \end{array} \right))(Alex)$

This works fine for *'know'*, but it will not produce the correct results for *'agree'* and *'be certain'*, since, for these, we need to consider all possible strongly exhaustive answers, and not just the true one. This is a problem for the embedding strategy

illustrated in (126). However, if we wanted to, we could use **Strong** to define the strongly exhaustive answer set, using the idea that a proposition is in the strongly exhaustive answer set iff it is the strongly exhaustive answer in *some* world.[24] Thus, given an abstract-intension α, the set of strongly exhaustive answers for α is given in (129) (Egré and Spector (2007) make good use of a similar approach, using quantification over worlds to quantify over possible answers).

(129) $\lambda p \exists w (p = \mathbf{Strong}(w)(\alpha))$

My preference for defining the answer set without appeal to the idea of the true strongly exhaustive answer, as I did above with Q and X, derives from several considerations. First, it allows us to apply the same embedding principle to mention-some and strongly exhaustive questions. Second, once we recognize that strongly exhaustive readings are possible with non-veridical embedders like *'be certain'*, we are compelled to use strongly exhaustive answer sets for embedding in some cases anyway, in which case it seems simpler to use the answer set all the time. Once we reach this point, defining the answer set by way of a true strongly exhaustive answer in a world seems like a somewhat roundabout way of doing things. Third, it allows me to maintain a shared question operator Q between strongly exhaustive and mention-some readings.

I'm happy with these reasons, but none of them is decisive, and in the end I suspect the choice of approaches will in large part turn out to be a matter of taste. Since we can trade off between the notion of a strongly exhaustive answer set, and the world-dependent notion of the unique true strongly exhaustive answer given in (125), we can take either as basic and derive the other from it as needed.

---

[24]This definition is not technically equivalent to my definition in all cases. The exception is that this definition of the strongly exhaustive answer set (reflected by (129)) guarantees that no strongly exhaustive answer is a contradiction, while my definition with Q and X will allow contradictory answers in the strongly exhaustive answer set in certain situations, such as question (128):

(128)   Which married bachelor provided a correct proof that there are finitely many prime numbers?

For most purposes, this difference doesn't really matter.

2.6.3 The Weakly Exhaustive Answer

The (true) weakly exhaustive answer to a question is the conjunction of all the true mention-some answers (this is, essentially, the definition found in Heim (1994)). We can also define the weakly exhaustive answer for an abstract directly with the **Weak** operator:

(130) $\mathbf{Weak} = \lambda w \lambda \alpha_{\langle \mathbf{s}, \langle \tau, \mathbf{t} \rangle \rangle} \lambda w' \forall \beta_\tau (\alpha(w)(\beta) \to \alpha(w')(\beta))$

That is, the weakly exhaustive answer for an abstract in a world w is the set of all worlds w' such that the extension of the abstract in w is a subset of the extension of the abstract in w' (as compared with **Strong**, we have replaced the equality of extensions requirement with a subset requirement on extensions).

Note that any notion of weakly exhaustive answer sets is likely to be at least a little messier than our notion of strongly exhaustive answer sets above. With strongly exhaustive answers, we could define *the* strongly exhaustive answer from the strongly exhaustive answer set by making it the unique true answer, and we could define the strongly exhaustive answer set from our notion of *the* strongly exhaustive answer by making the answer set the set of all possible strongly exhaustive answers. With weak exhaustivity, trading in an answer operator for an answer set complicates matters a bit. In particular, if we define the weakly exhaustive answer set in terms of **Weak**, if we start with an abstract-intension α we get (131) as our weakly exhaustive answer set:

(131) $\lambda p \exists w (p = \mathbf{Weak}(w)(\alpha))$

There will, in general, not be a unique member of this set that is true in a given world, so we cannot define the weakly exhaustive answer as the unique true member of the set in (131). To recover the weakly exhaustive answer from this set, we need to find the *maximal* true member of the set. This means, in turn, that we can't use the embedding rule that I used above for strongly exhaustive and mention-some answers. simply quantifying over possible weakly exhaustive answers, and trusting a factive predicate to narrow us down to the unique true one, will not produce the right results.

To see that (131) need not have a unique true member, consider again (1-a). There is, presumably, a possible world in which Anne loves only Liam. In this world, (123-g) is the weakly exhaustive answer to (1-a) so it will be in the weakly exhaustive answer set of (1-a). Now suppose that in the actual world, Anne loves Liam, Joyce, Red, Alex, and Rupert (but nobody else). In the actual world, then, (132) is the weakly exhaustive answer.

(1-a)   Who does Anne love?

(123-g) Anne loves Liam.

(132)   Anne loves Liam, Joyce, Red, Alex, and Rupert.

Since (123-g) and (132) are both weakly exhaustive answers for some world, they are both in the weakly exhaustive answer set. Further, since (132) entails (123-g), and (132) is the weakly exhaustive answer in the actual world, and so true, (123-g) will be true in the actual world. Thus, (123-g) will be in the set of possible weakly exhaustive answers, and will be true in the actual world, but will not be the weakly exhaustive answer in the actual world. We can, of course, still figure out that (132) is the weakly exhaustive answer by inspecting the answer set. To recover *the* weakly exhaustive answer, we simply take the maximal true member of the set – that is, the only true potential weakly exhaustive answer that entails all the others.

Because truth alone is not enough to does not suffice to distinguish the weakly exhaustive answer among the set of possible weakly exhaustive answers, the existential quantification rule for embedding will not produce the right results when combined with a weakly exhaustive answer set. The desire to use answer sets in a uniform way is one motivation for my leaving weak exhaustivity out. Another, discussed in Sections 3.1 and 3.2, is that I think the main empirical arguments for including weak exhaustivity in the theory of question semantics are not very convincing.

Although I assume that the weakly exhaustive answer doesn't have any role in the compositional semantics of questions or question-embedding, having it clearly defined may be useful background for discussions of comparison of theories in later chapters.

This concludes my discussion of alternative notions of answerhood. For more thorough and sympathetic discussion of some of these concepts, the directed to Heim (1994), Beck and Rullmann (1999), Groenendijk and Stokhof (1984), and especially Lahiri (2002).

CHAPTER 3

Some Challenges

JANE: [to Trent] How much money do you have?

[Trent is silent.]

JANE: Trent?

TRENT: I have none, so I said nothing.

<div align="right">Vebber (2001)</div>

This chapter is devoted to a survey of some arguments and data (mostly culled from the literature) that present problems for the baseline theory given in Chapter 2. These arguments and data have been presented for different purposes at different times, but all of them appear, or have appeared to somebody, to support some generalization about question embedding or answerhood conditions that runs contrary to the baseline theory presented above.

My goal in presenting these problems is to explore the strengths and weaknesses of the Chapter 2 theory and its near cousins, and to show that it fair somewhat better with these data than is generally assumed. For some problems, especially those related to weak exhaustivity, I will be able to show that the arguments are mistaken, or that the data has been misinterpreted, and that the baseline theory described above survives unamended. For others, I will sketch some ways in which we might revise the baseline theory. For some, I will only do my best to map out the issues, but will not offer a proper explanation. The list of problems presented here is, of course, not comprehensive, but I think that it covers many of the more important concerns, with an emphasis on major issues in the traditions exemplified by Hamblin (1973), Karttunen (1977), Higginbotham and May (1981), Groenendijk and Stokhof (1984), and Lahiri (2002).

The challenges presented here are, for the most part, reasonably well known, and in most cases a variety of solutions or partial solutions are already available. Imple-

menting these solutions may in some cases require us to sacrifice some uniformity in our theory of question-embedding, or require us to abandon certain technical details of the theory in Chapter 2, and there will no doubt be some difficulties in making solutions to all these problems work well together, but none of them threaten the basic ideas of accounts of question-embedding in this tradition. Some less widely appreciated challenges will be given more attention in Chapters 4 and 6.

3.1 Which Judgments Show Weak Exhaustivity? (And Which Don't?)

Weak exhaustivity has played, and continues to play, a major part in the literature on question-embedding. Since I left weak exhaustivity out of the system I presented in Chapter 2, and since introducing it into my system would require me to change both my semantics of questions and my embedding rule, any evidence for weak exhaustivity is a challenge that I need to take seriously. Recall the difference between the weakly exhaustive and strongly exhaustive answer to a question: if Alex and Red were the only students at the party, then the weakly exhaustive answer to (1) is (2), while the true strongly exhaustive answer is (3):

(1)   Which students were at the party?

(2)   Anne and Red were at the party and were students.

(3)   Anne and Red were the only students at the party.

That is, the weakly exhaustive answer is the proposition that identifies every student who was a the party as a student who was at the party, but (unlike the strongly exhaustive answer) doesn't convey that no more students were at the party.

The question before us is whether there is any case where the truth-conditions of a question-embedding sentence involve the weakly exhaustive answer to the embedded question, as distinct from a strongly exhaustive answer or a mention-some answer. There are a few different kind of data that are supposed to provide evidence for weak exhaustivity in question embedding, but one of the key ones is exemplified by (4)

(adapted from example 32 of Sharvit (2002)):

(4)   Maggie knows who was admitted to the program, but she doesn't know who wasn't admitted.

Some speakers (myself included) judge this sentence to be non-contradictory. For reasons that will be discussed below, the judgment of non-contradictoriness for (4) is incompatible with strong exhaustivity if we make certain (questionable) simplifying assumptions in our formal treatment (the key formal argument appears in Groenendijk and Stokhof (1984)). This formal result, together with the non-contradictoriness judgment, is taken to show that strong exhaustivity cannot be at work in (4), and, since there is an intuition that the truth-conditions of (4) typically involve Maggie knowing some kind of complete answer, the conclusion usually drawn is that she must know the weakly exhaustive answer. In the remainder of this section, I will show how this conclusion is not warranted. I will first sketch the argument for weak exhaustivity based on examples like (4), and then argue that the cases in which we get the troubling consistency judgments are all cases in which the simplifying assumptions of Groenendijk and Stokhof (1984), on which the weak exhaustivity argument depends, fail. This will involve two categories of issues: 'domain uncertainty' and 'complementation failure' (I do not mean to commit myself to a theoretical distinction between these categories, but only to note them as separate intuitive cases). Details appear below.

3.1.1   The Negation Generalization and Strong Exhaustivity

The relationship between strong exhaustivity and negation has played an important role in many arguments about the semantics of questions. The kind of reasoning involved is exemplified by Groenendijk and Stokhof (1984), who claim (while carefully noting certain important caveats) that each of the sentences in a pair (5) (cf. their argument X on p. 87) entails the other:

(5)   a.   Anne knows who walks.
        b.   Anne knows who doesn't walk.

The equivalence in (5) is not predicted if we assume the question *who walks?* gets a weakly exhaustive reading in (5), but (under some assumptions to be spelled out shortly), it is predicted if the embedded question receives a strongly exhaustive reading. The judgment of equivalence is taken by Groenendijk and Stokhof (1984) as evidence that these examples involve strong exhaustivity.

The argument that there is a non-strongly exhaustive reading at work in (4) starts with the assumption that the equivalence in (5) follows from strong exhaustivity. If this is generally true, then the equivalence should also apply for the two sentences in (6):

(6)   a.   Maggie knows who was admitted to the program.
        b.   Maggie knows who wasn't admitted to the program.

If these two sentences are equivalent, then any circumstances making one true must make the other true, meaning that (4) should be a contradiction. If there are speakers who don't find (4) self-contradictory or otherwise problematic, then, for those speakers, this equivalence must not hold, and, if strong exhaustivity predicts such an equivalence, then those speakers must be assigning a non-strongly-exhaustive reading to at least one of the embedded questions in (6).

What I want to do is argue that the simplifying assumptions used to motivate equivalence results for (5) and (6) are not as reliable as we might suppose, and that in particular they fail in the kinds of cases that make (4) true.

The first step is to look at the structure of the argument. Groenendijk and Stokhof (1984) associate the question (7-a) with the abstract (7-b), and the question (8-a) with the abstract (8-b):

(7)   a.   Who walks?
        b. $\lambda x(walks(w)(x))$

(8)   a.   Who doesn't walk?
        b. $\lambda x(\neg walks(w)(x))$

These abstracts leave out the personhood requirement of *'who'*, don't consider any covert domain restrictions, and treat English *'not'* as logical negation. Besides ignoring all domain restrictions in these sentences, Groenendijk and Stokhof (1984) further require that the same entities exist in all possible worlds. Their theory is situated in a classical bivalent logic, so that every assignment of values to the variables makes either $walks(w)(x)$ or $\neg walks(w)(x)$ true.

Under these assumptions, it can be shown that the strongly exhaustive answers associated with the two abstracts are the same, or, in the formalism of Chapter 2 that the equivalence in (9) must hold:

(9) $Q(\lambda w(X(\lambda x(walks(w)(x))))) = Q(\lambda w(X(\lambda x(\neg walks(w)(x)))))$.

The justification of of (9) proceeds by showing that a proposition is in the set (10-a) if and only if it is in the set (10-b).

(10)   a. $Q(\lambda w(X(\lambda x(walks(w)(x)))))$
        b. $Q(\lambda w(X(\lambda x(\neg walks(w)(x)))))$

The argument works the same way in both directions, so I show only one half of the biconditional. Let $D_{\mathbf{e}}$ be the domain of entities. Now, for every proposition p in the strongly exhaustive answer set for (7-b), there is some $S \subseteq D_{\mathbf{e}}$ such that p is true in a world u iff S is the set of entities a such that $walks(u)(a) = 1$. Since $D_{\mathbf{e}}$ is constant and bivalence is assumed, if S has this property for a given u, then $D_{\mathbf{e}} - S$ will be the set of entities a' such that $\neg walks(u)(a') = 1$. Thus, p is true in a world u iff $D_{\mathbf{e}} - S$ is the set of entities a' such that $\neg walks(u)(a') = 1$. Thus, there is $S' \subseteq D_{\mathbf{e}}$ (in particular $S' = D_{\mathbf{e}} - S$) such that p is true in a world u iff S' is the set of entities a' such that $\neg walks(u)(a') = 1$. This, though, is condition that is needed for p to be in the strongly exhaustive answer set of (8-b).

What we just saw is that, under the assumptions stated, (7-a) and (8-a) will contribute the same strongly exhaustive answer set. Since, in the system we've been assuming, the answer set of a question is its only contribution to the truth-conditions of a question-embedding sentence, this means that the two sentences in (5) (which contain embedded questions just shown to have the same strongly exhaustive answer

82

sets) must have the same truth-conditions when the embedded questions receive strongly exhaustive reading. This argument is not peculiar to these sentences, but will go through analogously for any pair of question-embedding sentences that differ only in the substitution of an embedded question for its negation, provided that no restricting material in the question outscopes the negation.

3.1.2 Domain Uncertainty

The assumptions used in to justify the equivalence in (5) are, taken literally, implausible. The set of existing entities does vary across worlds (at least on a traditional understanding of 'existing'), and *'who'* comes with some kind of personhood requirement, and the set of people certainly ought to vary across possible worlds.[1] As soon as we include the personhood requirement outside the scope of the negation, the extension of a *'who'*-abstract will no longer be completely determined by the extension of its negation, since there can be two worlds in which the set of people who walk is the same, but the set of people who don't walk is different because the set of (existing) people is different in the two worlds. Any covert domain restrictions will present a similar problem. (These points apply only to the 'de dicto' interpretations of *'wh'*-phrase restrictions: if the restrictions are understood 'de re', so that they don't vary with the world variable bound by Q, then the only possible issue will be variation in the domain of entities across worlds.)

The reason the equivalence in (5) sounds right is that usually it is: under typical conversational circumstances, we understand that there is a domain restriction, and we assume that its extension is known to Anne, and that it narrows things down to people Anne knows to be existing people. Under these circumstances, the issues just discussed disappear.

Groenendijk and Stokhof (1984) recognize these issues (on their page 87), acknowledging that there are certain potential counterexamples to the entailments involved in the kind of equivalence exemplified by (5). They note, in particular, that

---

[1]There are, we must assume, possible worlds in which you do not exist, and never have existed, and never will exist, in the sense that you have never been conceived (or built, or synthesized...), been born (or been activated, or been decanted, or hatched, or sprouted, or budded...), taken a breath, had an idea, paid taxes, died, or done any of the other things one might be tempted to say people essentially tend to do, and you never will. In such a world, your failure to be born, to die, and so on do not count as counterexamples to universal claims about all people in that world being born, dying, and so on, so in that world you seem not to be in the extension of *'person'*.

there are counterexamples that involve 'the type of situation in which the subject of the propositional attitude is not fully informed as to which set of individuals constitutes the domain of discourse'. They also recognize the need to fix knowledge of the restrictor in their discussion of a pair similar to (11-a) and (11-b) (cf. their argument XII, p. 90). The equivalent of two such sentences is, they observe, is invalid if *'man'* receives a 'de dicto' reading:

(11)   a.   Anne knows which men walk.
          b.   Anne knows which men don't walk.

For Groenendijk and Stokhof (1984), to validly infer (11-b) on the basis of (11-a), we must make the assumption that Anne knows who the men are. Taking into account the domain complexities, we will predict the contradictoriness of (4) only if Maggie's knowledge is sufficient to fix the set of existing people with whatever properties are picked out by the covert domain restriction. In the absence of further data, this assumption is not warranted. In particular, there is a highly natural situation in which Maggie will be uncertain about the domain restriction.

To see that Maggie's certainty about the extension of the domain restriction is not assured, suppose that Maggie is on a department's email list, but not on its admissions committee. Suppose that this department has just made the admissions decisions for its graduate program, and that somebody has just sent Maggie an email that begins *'Here's a list of the people we admitted...'*, and that she justifiably trusts the source of the list, and has committed the list to memory, and inferred, correctly, that the list is complete. Under these circumstances, Maggie knows a strongly exhaustive answer to the question, and (12) is judged true.

(12)   Maggie knows who was admitted to the program.

However, in the same scenario, (13) is judged false:

(13)   Maggie knows who wasn't admitted to the program.

If (12) is true on a strongly exhaustive reading (as it undeniably is in (12), in the situation described), then, under the assumptions used by Groenendijk and Stokhof

(1984), (13) should also be true on a strongly exhaustive reading (and also on a weakly exhaustive or mention-some reading). The falsehood judgment for (13) shows that one of these assumptions must not hold. Inspecting our intuitions, we see that the assumption of domain certainty fails, and that the domain does vary across worlds under consideration. If *'who'* contributes only a personhood requirement, then in this situation it is implausible that Maggie should be aware of a complete list of living people who weren't admitted. If, as is more plausible, *'who'* is further restricted, so that its contribution can be paraphrased as *'which applicants'*, then it seems likely that Maggie, not being on the admissions committee, doesn't know who the applicants were, and so can't identify the set of applicants who weren't admitted. Under these circumstances, (13) will be false on a strongly exhaustive reading (and possibly even a mention-some reading), even though (12) will be true on a strongly exhaustive reading, making (4) true on a strongly exhaustive reading. Since we have an example that makes (4) true on a strongly exhaustive reading, judgments of non-contradictoriness for (4) will not be evidence that another reading (such as a weakly exhaustive reading) is at work.

Before going further, I want to flesh out the verifying scenario in a bit more detail, to make sure that it works as I described. Suppose that there were three applicants to the program: Riley, Adam, and Robin. Suppose that Riley and Adam were admitted, and Robin were rejected. In this scenario, if (14-a) is understood as (14-b) and (15-a) is understood as (15-b), then the unique true strongly exhaustive answer to (14-a) is (14-c) and the unique true strongly exhaustive answer to (15-a) is (15-c).

(14) a. Who was admitted to the program?
   b. Which applicants where admitted to the program?
   c. Riley and Adam were the only applicants admitted to the program.

(15) a. Who wasn't admitted to the program?
   b. Which applicants weren't admitted to the program?
   c. Robin was the only applicant not admitted to the program.

As for Maggie's knowledge, suppose, as before, that Maggie was not on the ad-

missions committee or otherwise involved in admissions decisions, but that she has been notified of the list of accepted applicants, and that she had never heard of any of the applicants before. Thus she knows that Riley and Adam applied, and were admitted, and that they were the only people admitted, but she doesn't know that Robin applied (or even that Robin exists). Under these circumstances, Maggie knows the proposition given by (14-c), which is a strongly exhaustive answer to (14-a), making (12) true, but she doesn't know (15-c), since nothing in her knowledge excludes the possibility that Robin didn't apply, or, for that matter, the possibility that there were more applicants: that is, the various competing answers in (16) are all compatible with Maggie's knowledge:

(16) a. There were no applicants who weren't admitted to the program.
      b. Wesley was the only applicant not admitted to the program.
      c. Wesley and Robin were the only applicants not admitted to the program.

For this reason, Maggie doesn't know any strongly exhaustive answer (or, indeed any weakly exhaustive or mention-some answer) to (15-a) (since knowledge requires truth, and (15-c) is the only true strongly exhaustive answer), making (13) false. Thus, (4) is true on a strongly exhaustive reading in this scenario.

(12) Maggie knows who was admitted to the program.

(13) Maggie knows who wasn't admitted to the program.

(4)   Maggie knows who was admitted to the program, but she doesn't know who wasn't admitted.

Even if we somehow neutralize the restriction of the domain to applicants, Maggie may be uncertain about whether Robin and Wesley are people, or even whether they exist. In all cases, knowing the extension of a property in a domain is only equivalent to knowing the extension of that property's negation if one knows what is in the domain – whenever domain uncertainty of any kind arises, the (4) judgment will potentially be consistent on a strongly exhaustive reading, so these judgments are not, at least not by themselves, evidence of weak exhaustivity.

3.1.3 Complementation Failure

There is a natural way to try to rule out domain uncertainty: instead of using a very open-ended *'wh'*-phrase like *'who'*, we can use one with a very narrow, very explicit restriction, and take steps to make it clear that the subject of *'know'* knows the extension of this restriction. Such an attempt, adapted from an example in Guerzoni and Sharvit (2007), appears below:

(17)   Rupert knows which of his four students were admitted, but he doesn't know which weren't.

Suppose we first assure our informants that Rupert knows who his students are, and then we ask them to judge the contradictoriness or non-contradictoriness of (17). In this case, the explicit domain restriction is so narrow that an additional covert restriction seems unlikely. We have explicitly specified that the subject knows the extension of this restriction, and, even if we had not specified this, the small size of the restriction, and its close connection with the subject, makes it implausible that the subject would not know the extension of the restriction.

What are the judgments? For me, and for some other people that I've talked to, (17) sounds much weirder than (4). This is what we'd expect if a domain effect were the main culprit. However, I have encountered a number of people who report that a sentence like (17) sounds perfectly consistent to them. This is a more credible problem for strong exhaustivity.

In order to get a handle on the nature of the problem, I asked some of my informants who had judged something like (17) consistent to describe circumstances that would make (17) true. One person did describe a scenario that suggested a weakly exhaustive reading,[2] but most informants described other scenarios, along the lines of those described in (18):

(18)   a.   Rupert knows which students were admitted straight out, but he doesn't know which of the remainder were rejected and which were placed on

---

[2]I have no account of this one informant's response, but, as she is a professional linguist, I feel entitled to provisionally assume she was simply overthinking things. A survey of a larger number of naïve speakers would of course be desirable to confirm that this one informant sufficiently anomalous to justify ignoring her.

the waitlist.

b. It's entirely possible that some students were admitted, some students weren't, and some students didn't apply, and Rupert does not know who falls into which of the latter two categories.

One of the people who provided a scenario like (18-b) explicitly rejected a weakly exhaustive scenario when I described it to her. Informally, what is going on here is that our assumption that negation is a complementation operator is breaking down, so that *'were admitted'* and *'were not admitted'* do not divide the domain between them, but instead both exclude some marginal classes or borderline cases, because *'were not admitted'* is understood to mean *'were rejected'*. The formalization of the effect by which *'were not admitted'* can be treated as equivalent to *'were rejected'* is an interesting problem, but one beyond the scope of the present discussion. In the case of (18-b), a domain account probably offers a sufficient explanation, but for (18-a) the appropriateness of a domain account is less clear. It might be that *'not'* has a meaning that is really not much like truth-functional negation, and that is sensitive to degrees or intensionality in complex ways. It might be some kind of presupposition effect. It might be another domain effect, where *'which of his four students'* is understood as *'which of his four students that have received a final and definite decision'*.

Without taking a position on this question on why *'not'* is failing to act as a complementation operator, it is easy to show that, if *'were not admitted'* is understood as *'were rejected'*, then there are scenarios on which (17) is true on a strongly exhaustive reading. Consider the scenario in which Rupert's four students are Anne, Red, Alex, and Jonathan. Suppose that Anne and Red were accepted, Jonathan was rejected, and Alex falls into some other category, such as one of those outlined in (18). In this case, the strongly exhaustive answer to (19-a) is (19-b), and the strongly exhaustive answer to (20-a), when it is understood as (20-b), is (20-c):

(19)   a.   Which of Rupert's students were admitted?
       b.   Anne and Red were the only students of Rupert's who were admitted.

(20)   a.   Which of Rupert's students weren't admitted?

b. Which of Rupert's students were rejected?

c. Jonathan was the only student of Rupert's who was rejected.

Now suppose that Rupert knows that Anne and Red were admitted, and, further, that none of his other students were among those admitted (suppose, say, that he has access to a complete list of students who have been admitted so far), but that he doesn't know, for Jonathan and Alex, whether they were rejected or fall into some other category (perhaps he even suspects, erroneously, that Jonathan didn't apply, or that Alex applied and was rejected). In this case, Rupert knows (19-b) but not (20-c), so he knows a strongly exhaustive answer to (19-a) but not (20-a) (on the assumption that (20-a) is somehow understood as (20-b)). Thus, on strongly exhaustive readings of both questions, (17) will be true.

These problems illustrate the difficult of using negated questions to test for weak exhaustivity, and I think my observations show that the non-contradiction judgments reported by Sharvit (2002), Guerzoni and Sharvit (2007), and others do not harm an approach that handles all these examples with strong exhaustivity, but instead highlight the importance of domain restrictions in embedded questions, and perhaps also expose a curious fact about the way that some speakers interpret negation.

In principle, of course, the test could still be valid, if we could craft our examples and provide background information in a way that really guaranteed that the necessary conditions for the test were in effect. The complementation failure effect discussed above shows that this is more difficult than it at first appears, and that the needed basic assumptions can fail in unexpected ways. One important lesson is that, when using judgments of non-contradictoriness to show something, we should try, whenever possible, to find descriptions of scenarios in which the sentence under study is judged true – if possible, we should try to elicit descriptions of such scenarios from our informants, and when this is not possible, we should try to construct plausible scenarios ourselves, and then seek out judgments about the truth of the sentence in those scenarios. Doing this can help us to identify the source of unexpected non-contradiction judgments, and can expose unforeseen complications in the process.

3.2 'surprise'

Guerzoni and Sharvit (2007), and Heim (1994), among others, present arguments that *'surprise'* is a weakly exhaustive embedder.[3] That is, the claim is that (23) is true if and only if the weakly exhaustive answer to (24) surprised Anne.

(23)   Anne was surprised by who was at the party.

(24)   Who was at the party?

One piece of evidence presented in favor of this claim is the consistency of sentences like (25), which are interpreted as problematic for strong exhaustivity.

(25)   Anne was surprised by who was at the party, but she wasn't surprised by who wasn't.

For reasons discussed in section 3.1, I don't consider these kinds of judgments compelling in isolation, but, with *'surprise'*, there is reason to think that more is going

---

[3] *'surprise'* is the poster predicate of the 'emotive factive' predicates. These predicates are either verbs that associate their clausal argument with a subject-like syntactic position (like *'surprise'* or *'annoy'*) or adjectives (like *'happy'*):

(21)   a.   That Anne had survived surprised William.
        b.   William was surprised that Anne had survived.
        c.   That William had survived annoyed Anne.
        d.   Anne was annoyed that William had survived.
        e.   Anne was happy that Rupert had survived.

Emotive factives that embed questions typically embed *'wh'*-questions, but not *'yes'*/*'no'* questions or alternatives questions:

(22)   a.   Anne was surprised at who stole the documents.
        b.   Red was annoyed at who showed up for the party.
        c.   *Anne was surprised (at) whether William stole the documents.
        d.   *Red was annoyed at whether Faith showed up for the party.
        e.   *Anne was surprised (at) whether William or Faith stole the documents.
        f.   *Red was annoyed at whether William or Faith showed up for the party.

Many of the arguments given for *'surprise'* in this section are plausibly applicable to at least some other emotive factives (*'amaze'* and *'disappoint'*, in particular, are similar to *'surprise'*), but I will for the most part restrict attention to *'surprise'*, and not attempt an exhaustive survey of this fascinating class of embedders.

on. First, note that when we make domain restriction solutions difficult, *‘surprise’* fairs better than *‘know’*: (26) sounds a lot better than (27) (although both can be made consistent in a sufficiently contrived context).

(26)   Rupert was surprised by which of his students at the party, but he wasn’t surprised by which weren’t.

(27)   Rupert knows which of his students were at the party, but he doesn’t know which weren’t.

The contrast between (26) and (27) does not by itself establish that *‘surprise’* doesn’t allow a strongly exhaustive reading: an advocate of strongly exhaustive interpretations for both sentences could attempt to argue that (27) sounds odder than (26) because some pragmatic force that makes it easier to achieve a suitable domain restriction with *‘surprise’* than with *‘know’*. The possibility of such a force is, as usually, difficult to rule out conclusively, but the contrast between these two sentences should still give us pause.

More substantially (as discussed by Heim (1994), Sharvit (2002), and Guerzoni and Sharvit (2007)), we can find cases where Anne is surprised by a strongly exhaustive answer to (24), but (23) is nevertheless untrue, and many of these seem to correspond to the cases where, although the strongly exhaustive answer contradicts Anne’s expectations, the weakly exhaustive answer does not.

To see an example of this, suppose that Anne expected that *somebody* would be at the party, but it turns out that nobody showed up. In this case, the strongly exhaustive answer to (24) is (28), but the weakly exhaustive answer is the tautologous proposition (since it is the conjunction of all true mention-some answers, and there are no true mention-some answers).

(28)   Nobody was at the party.

If Anne becomes aware of (28), and is surprised by it, is (23) true? Many speakers report that it is not. On a standard strongly exhaustive account of *‘surprise’*, this is hard to explain. (28) is the strongly exhaustive answer, and Anne was surprised by

it, so (23) ought to be true on a strongly exhaustive reading. If, on the other hand, (23) is understood on a weakly exhaustive reading (so that it is true iff Anne was surprised by the weakly exhaustive answer to (24)), then this observation is easy to explain: in this case, the weakly exhaustive answer is transparently tautologous, so it should be unsurprising to Anne. Thus, in the case where nobody attended the party, weak exhaustivity is a better fit for the observed truth conditions than strong exhaustivity is.

If this were the only example, we might dismiss it as an exceptional case: the scenario where the extension of the abstract is empty might be an atypical one, and some special rule or pragmatic effect might be invoked for this particular case. However, we find the same effect in less trivial cases. Suppose, for example, that Anne expected Alex and Red would both show up for the party, and had no other expectation related to (24) (so in particular, neither Liam's attendance nor his absence was expected). Suppose further that, in fact, the only people who showed up for the party were Red and Liam. This surprises Anne, since she was expecting Alex. That is, (29) is the strongly exhaustive answer to (24), and it surprised Anne, making (30) true:

(29)   Red and Liam were the only ones at the party.

(30)   Anne was surprised that Red and Liam were the only ones at the party.

In tabular form, the situation looks roughly like this:

		Alex at party	Red at party	Liam at party
(31)	Anne's expectations:	Y	Y	?
	Facts:	N	Y	Y
	Anne surprised?	Y	N	N

In spite of (30) being true, (23) is judged untrue in this case. Thus, although Anne was surprised by a strongly exhaustive answer, the corresponding question-embedding sentence is untrue, indicating that strong exhaustivity is not involved here. Again, a weakly exhaustive reading would account for this. The weakly exhaustive answer to (24) is (32), and (33) is false, since neither Red's nor Liam's

presence upsets Anne's expectations. Anne is surprised by a strongly exhaustive but not a weakly exhaustive answer, and she is not related to the question by *'surprise'*.

(32)   Red and Liam were at the party.

(33)   Anne was surprised that Red and Liam were at the party.

These observations lead naturally to the conclusion that the truth-conditions of sentences like (23) should not be analyzed in terms of strongly exhaustive answers to the embedded question. This interpretation of the data seems to me to be correct. It does not, however, follow that we need a special weakly exhaustive reading. A mention-some reading will do just as well, since, in both of the scenarios above, (23) will be untrue on a mention-some reading. (In fact, a number of other authors, including Lahiri (2002) (and, according to Lahiri (2002), Hintikka (1976)), have already suggested (in different terms) that *'surprise'* is distinguished from many other embedders by a tendency to assign mention-some readings to the questions that it embeds.) First, consider the case where nobody was at the party. In this case, there are no true mention-some answers to (24), so (since *'surprise'* is factive) no mention-some answer surprised Anne, so (23) will be untrue on a mention-some reading.

In the case where Red and Liam were the only ones who showed up, the only true mention-some answers are (34-a), (34-b), and (34-c):

(34)   a.   Red was at the party.
        b.   Liam was at the party.
        c.   Red and Liam were at the party.

All of these mention-some answers are consistent with Anne's expectations, and none of them are surprising to her. What surprises her is Alex's absence, but that absence is not entailed by any mention-some answer. So, on a mention-some reading, we should expect (23) to be untrue in this situation, which is the judgment that is reported.

What we have just seen is that the judgments so far reported do not suffice to distinguish between a weakly exhaustive reading and a mention-some reading. To determine which reading is available, we will need to appeal to other intuitions. I want to argue that our truth-conditional judgments for sentences like (23) are in fact more naturally compatible with a mention-some reading than with a strongly exhaustive reading. To see this, first note that surprise entails (and indeed presupposes) awareness. So, for example, the inference from (33) to (35) is valid:

(35)   Anne came to know that Red and Liam were at the party.

This means that, for Anne to be surprised by a weakly exhaustive answer, she must come to know that weakly exhaustive answer.

It turns out to be relatively easy to find cases where (23) is true, and (32) is the weakly exhaustive answer to the embedded question, but (33) is not true because (35) is not true. In these cases, Anne *is* surprised by a mention-some answer. This makes a mention-some reading a natural fit for our judgments for (23), indicating that there is no need to include a separate weakly exhaustive reading.

(23)   Anne was surprised by who was at the party.

(32)   Red and Liam were at the party.

(33)   Anne was surprised that Red and Liam were at the party.

(35)   Anne came to know that Red and Liam were at the party.

The scenario is straightforward. Suppose Anne's awareness of who was at the party begins and ends with her knowledge that Liam was at the party, but that this surprised her, since she wasn't expecting Liam to show up. Now suppose that the weakly exhaustive answer is in fact (32). That is, only Red and Liam were at the party. In this situation, (35) is false, since Anne never learned that Red was at the party, so (33) is untrue. In spite of this, the judgment is that (23) is true in this situation. This is naturally accounted for by a mention-some reading: Anne became aware of, and was surprised by, the mention-some answer (34-b).

This indicates that (23) receives a mention-some reading, and that the judgments associated with it are not good evidence that weak exhaustivity is involved in question embedding. Our earlier observations also give us reason to suspect that it *does not have access* to a strongly exhaustive reading. This second point remains a real anomaly, and one that I do not know how to explain (except by stipulating that *'surprise'* has some means of selecting one reading of an embedded question). Thus, *'surprise'* is a real source of non-uniformity, although it is one that can still be handled with the embedding rule and answer sets given in Chapter 2.

A related anomaly with *'surprise'* is that it seems to bring out mention-some readings in embedded questions that otherwise favor strongly exhaustive readings. Although the cause of this effect is not clear, some embedded questions readily receive mention-some readings, while, in others, a mention-some reading is marginally available or unavailable. For example (36-a) naturally receives a strongly exhaustive reading (at least when embedded), while (36-b) receives a mention-some reading.

(36) a. Who was at the party?
      b. Where can we buy a newspaper?

This association of different questions with different kinds of answerhood conditions appears to be general across a variety of responsive predicates: (36-a) favors an exhaustive interpretation in (37-a), (38-a), and (39-a), but (36-b) favors a mention-some interpretation in (37-b), (38-b), and (39-b).

(37) a. Joyce knows who was at the party.
      b. Joyce knows where we can buy a newspaper.

(38) a. Alex forgot who was at the party.
      b. Alex forgot where we could buy a newspaper.

(39) a. William is certain of who was at the party.
      b. William is certain of where we can buy a newspaper.

In contrast with this, (23), as we saw above, receives only a mention-some reading and apparently *cannot* receive a strongly exhaustive reading. ((40) still receives a

mention-some reading.)

(23)   Anne was surprised by who was at the party.

(40)   Anne was surprised by where we could buy a newspaper.

I do not know how to address this issue (although, in Chapter 6, I offer some speculations on how it might related to other characteristics of *'surprise'*), but I feel compelled to flag this as an important problem for future work.

One final point is that *'surprise'* forces us to pay attention to the interaction between mention-some readings and plural semantics. This is because, as pointed out by Lahiri (2002), *'surprise'* fails to distribute across conjunction of propositions. That is, (41-a) does not entail (41-b) (indeed, it does not even entail (41-c)):

(41)   a.   I was surprised that Red and Alex showed up for the meeting.
         b.   I was surprised that Red showed up for the meeting and I was surprised
               that Alex showed up for the meeting.
         c.   I was surprised that Red showed up for the meeting or I was surprised
               that Alex showed up for the meeting.

To see that (41-a) does not entail (41-b) or (41-c), consider the scenario where I believe that Red and Alex are avoiding each other, and, for this reason, I expect that if one of them shows up for the meeting, then the other will make an excuse to avoid it. Neither Red's nor Alex's attendance would by itself be contrary to my expectations, but Red and Alex's joint attendance is surprising. In this case, if I became aware that Red and Alex both showed up for the meeting, and was surprised, then (41-a) will be true, but neither (42-a) nor (42-b) seems to be true in isolation, so neither (41-b) nor (41-c) is true:

(42)   a.   I was surprised that Red showed up for the meeting.
         b.   I was surprised that Alex showed up for the meeting.

This feature distinguishes *'surprise'* from many other propositional attitudes (both question-embedding and non-question-embedding): *'know'*, *'be certain'*, *'agree'*, *'re-*

member', and *'believe'* all allow for distribution (or something close to it).

What is important for us is that, in the scenario where (41-a) is true, but (42-a) and (42-b) are not, (43) is nevertheless true:

(43)   I was surprised by who was at the meeting.

If (44-a) and (44-b) were the only mention-some answers, this observation would be difficult to account for on a mention-some reading, since neither of those answers surprised me:

(44)   a.   Red was at the meeting.
         b.   Alex was at the meeting.

I think the solution is not to abandon a mention-some reading, but, drawing on Lahiri (2002) to recognize (45) as a mention-some answer:

(45)   Red and Alex were at the meeting.

This will (probably) fall out of the formalism in Chapter 2 naturally if, following Link (1997) and others, we allow the domain of entities to include plural entities, corresponding to groups of atomic entities.[4] That is, for every proposition p that is a mention-some answer to the embedded question in (43), there will still be some entity x such that p is the proposition that x was at the meeting, but x need not be an atomic entity. In the case of (45), x is the plural entity denoted by *'Red and*

---

[4]This of course depends on the details of our theory of plurality. Even if, for one reason or another, we are reluctant to handle this case with plural entities, closing the answer-set under conjunction would be an easy amendment to the theory. We would simply replace the standard Q with a conjunction-closed embellishment Q':

(46) $Q' = \lambda\alpha_{\langle s, \langle \tau, t \rangle \rangle} \lambda q_{\langle s, t \rangle} \exists P_{\langle \langle s, t \rangle, t \rangle} (P \subseteq Q(\alpha) \wedge \exists q'(P(q')) \wedge q = \lambda w \forall p (P(p) \to p(w)))$

Since any two strongly exhaustive answers are mutually inconsistent, Q and Q' will produce the same results when the X operator is present (except that Q' will always add the contradictory proposition $\lambda w(0)$ to the answer set). In the mention-some case, all that Q' does is guarantee that all conjunctions of basic mention-some answers are included in the mention-some answer set. The issues of plurality and answer conjunction are discussed extensively by Lahiri (2002), albeit with some different background assumptions, and the reader is encouraged to look there for some relevant observations and discussion.

Alex'. This is not a problematic move – plural entity theories are well-developed and motivated by many other considerations, but it is worth noting that, if we frame things as I have, we need to include plural entities in the system in order to get the right results for embedding under *'surprise'*.

3.3 *'tell'* and Veridicality

As I've mentioned, most treatments of strong exhaustivity (e.g. Groenendijk and Stokhof (1984)) evaluate question embedding under (most) responsive predicates not in terms of an answer set, but in terms of the unique true strongly exhaustive answer in the world of evaluation. The most important evidence for this given in Groenendijk and Stokhof (1984) involves *'tell'*. *'tell'* is not veridical with respect to an embedded *'that'*-clause ((47) doesn't entail (48)), but, when we embed a question under *'tell'*, it seems to become veridical: from (49) we typically infer that William told anne a *true* answer to the question.

(47)   William told Anne that the Illuminati stole the documents.

(48)   The Illuminati stole the documents.

(49)   William told Anne who stole the documents.

Groenendijk and Stokhof (1984) use this as an argument for privileging true answers across the board. This accounts for the data presented above, but leaves them ill-equipped to handle *'be certain'*. Lahiri (2002) handles the above observations for *'tell'*, along with *'be certain'*, by introducing some non-uniformity: some responsive predicates are marked as veridical question-embedders, and others are marked as non-veridical question-embedders. This is adequate, but somewhat unappealing. Egré and Spector (2007) attempt to overcome this non-uniformity by arguing that *'tell'* is not, in fact, veridical (at least not as a matter of its truth-conditional semantics). This, they claim, allows their theory (which, in the relevant respects, behaves the same as the one in Chapter 2) to handle all responsive predicates by an operation equivalent to existential quantification over true and untrue answers. I think that Egré and Spector (2007) are right, and that as a result we should keep the theory in

Chapter 2, along with its prediction of non-veridicality for question-embedding *'tell'*.

Egré and Spector (2007) point to examples like (50), (51), and (52) as evidence that question-embedding tell is non-veridical.

(50)   Every day the meteorologists tell us what the weather will be, but they are often wrong.

(51)   The pundit told us who we should blame, but she doesn't know what she's talking about.

(52)   The analyst told the investors what the market would do, but none of his predictions came to pass.

All three of these sentences seem readily interpretable, but all three assert that the answer being told is not a true one. This suggests that the truthiness inference for *'tell'* is not a truth-conditional or strong presuppositional effect, but is some kind of defeasible implicature.

This immediately raises the important question of how this truthiness implicature arises. I cannot answer this, but I note that the same truthiness implicature is seen, at least as strongly, with certain uses of *'tell'* with a propositional complement, especially when the subject of *'tell'* is inanimate. For example, from (53-a) we tend to infer (53-b), and from (54-a) we tend to infer (54-b):

(53)   a.   The distribution of the debris tells us that the bomb was placed under the stairs.
        b.   The bomb was placed under the stairs

(54)   a.   The spacing of the puncture wounds told Anne that the victims were killed in a vampire attack.
        b.   The victims were killed in a vampire attack.

This does not resolve the mystery of truthiness implicatures with *'tell'*, but it does suggest that saying that the implicature in question is triggered by question em-

bedding alone does not provide a completely satisfying account of the full range of data.

3.4 Granularity Effects

Ginzburg (1995a,b) and van Rooij (2004) argue for a considerably more flexible notion of answerhood conditions than I have considered so far. One important class of motivating examples involves variability in how much one has to know about one's location to know where one is. Consider, in for example, the relationship between examples (55-a) and (55-b):

(55)   a.   Rupert knows where Anne is.
        b.   Rupert knows that Anne is in Cleveland.

If Anne is in fact in Cleveland, is (55-b) sufficient information to infer the truth of (55-a)? That is, does (56-b) belong to the answer set of (56-a)?

(56)   a.   Where is Anne?
        b.   Anne is in Cleveland.

The adequacy of (56-a) as an answer to (56-b) appears to depend on context. If we're sitting someplace in southern California discussing Anne and her location, and what's been established so far is that Anne is not in our general region, but not much else, then (56-b) will be a good answer to (56-a). If we're sitting in Cleveland, and the background is that Anne went out for the evening, (56-b) will probably not be a good answer.

To see how this plays out with embedding, consider the following example (adapted loosely from examples in Lahiri (2002) and Ginzburg (1995a,b)). Suppose now that it is Rupert and Larry who are discussing Anne, and what they both know is that Anne is not especially near their location in southern California. In this case, if Larry asks (56-a), and Rupert answers with (56-b), and Larry trusts that Rupert is right, should Larry conclude that (55-a) is true? Apparently, (55-a) should be true in this situation, since Cleveland is far away from Larry and Rupert's location, and

little was perviously known about Anne's general location. If, on the other hand, the conversation is taking place in Cleveland, and it is known that Anne is in the area, (55-b) is not enough to establish the truth of (55-a), and what is needed is something more like (57).

(57)   Rupert knows that Anne is at the club down the street.

Ginzburg (1995a,b) and van Rooij (2004) see this (roughly) as evidence that the answerhood relation is inherently pragmatic in a special way, even in embedding contexts. Lahiri (2002) argues (his section 2.4, pages 55-61) that the phenomenon discussed here can be treated as a domain effect. I think that Lahiri is right, and that all the cases above involved a strongly exhaustive reading, and that (56-b) serves to convey an exhaustive answer to (56-a) with respect to some domain restrictions, but not others. In the remainder of this section, I will work through the domain argument explicitly, drawing heavily on the insights of Lahiri (2002).

We have already seen 'wh'-phrases carry a contextually specified domain restriction. This is presumably true with 'where' just as it is for other 'wh'-phrases: 'where' abstracts over spatial regions, and context can restrict us to certain regions. This can mean regions in a certain restricted spatial area, but it can also mean regions of a particular granularity. The idea is that in (58-a) 'where' will mean 'which city', and in (58-b) 'where' will mean something like 'which neighborhood' or 'which address' or 'which landmark'. Once we allow context to set a domain in this way, we will get the right answerhood conditions on either a strongly exhaustive or a mention-some reading:

(58)   a.   Where did Anne fly to?
        b.   Where did Anne walk to?

This is not different from any other domain restriction, and requires no special amendment to our theory of questions and question-embedding. Indeed, we see exactly the same variation with non-question-oriented quantification over places, as in the contrast between (59) (fine-grain parts of a smallish region) and (60) (coarse-grain parts of a large region):

(59)   There are posters for the mayor's re-election campaign everywhere.

(60)   It seems that there are planets of roughly Earth-like mass everywhere.

We also sometimes find conspicuous domain-restriction ambiguity with quantification over locations, just as we did with questions in the original example of (56-a). Consider the following examples:

(61)   a.   They sell mediocre American coffee everywhere nowadays.
        b.   They sell mediocre American coffee in most places nowadays.

If *'everywhere'* and *'in most places'* involve universal and majority quantification over arbitrarily small spatial regions, these sentences will have to be false in any conceivable non-fantastic world. However, I think it's easy to imagine a world and a context in which an utterance of either of these sentences is true, and another context for the same world in which both are false. To make both true, assuming we're talking about globalization and the international presence of American brands, so that the granularity of locations we're talking about is *nation-states* or *markets*, either at a global level or within some particular region. In that case, (61-a) or (61-b) is plausibly true provided that, a reasonable sampling of major cities in the region we're discussing contain a single representative of an American coffee chain. If, on the other hand, we keep the facts of the world the same, but are sitting in a particular city that contains only one place that sells mediocre American coffee, and are discussing the local business landscape, then (61-a) and (61-b) are intuitively false.

This variation doesn't mean that the semantics of *'everywhere'* isn't well-modeled by treating it as a universal quantifier over locations. It simply means that the domain restriction on this quantification can limit granularity, and that the context has considerable freedom to influence the selection of a domain restriction, so that, in different contexts, we could imagine the examples above being restricted to shops, to neighborhoods, to cities, to nation-states, or to even larger regions. Context will decide which, and the truth-value will be determined accordingly, but, given this contextual restriction, *everywhere* always contributes the same universal quantifier.

Likewise, in examples like (56-a), the embedded question can be understood as always exhaustive, with exactly the same variable granularity effect that we find with domain restrictions more generally.

I hope this is enough to convince the reader that these kinds of granularity examples are not a threat to the account in Chapter 2. For further discussion of this issue, the reader is referred to Section 2.4 of Lahiri (2002).

3.5   Pair-List Readings

Question (62) famously has three distinct readings, corresponding to three distinct kinds of answers: the ordinary (narrow scope) reading, the functional reading, and the pair-list reading.

(62)   What did every guest bring to the party?

(63)   Ordinary answers:
  a. Every guest brought a bottle of wine.
  b. Every guest brought a pie.

(64)   Functional answers:
  a. Every guest brought their favorite dish.
  b. Every guest brought the same thing they brought to the last party

(65)   Pair-list answers:
  a. Anne brought some fresh vegetables, Red brought some incense, Alex brought a box of donuts, Rupert brought a bottle of wine, and William brought some beer.
  b. Anne brought steaks, Red brought cookies, Rupert brought tea, and Alex and William didn't bring anything.

The ordinary reading was the main reading discussed in Chapter 2.[5] The functional reading was also discussed in Chapter 2, where I suggested that it fit within the theory presented there, provided that we allowed *'what'* to abstract over functions rather than entities, along the lines of what is done in Groenendijk and Stokhof (1984).

The pair-list reading is more troubling. A natural reaction is that it ought to involve the universal *outscoping* the question, but finding a suitable scope is difficult. In my framework, for example, scoping a universal over the Q operator does not produce the right results on either a strongly exhaustive or a mention-some reading.

To see why scoping the universal over Q consider what happens when we try to do this with (62). Let's try the mention-some variant:

(66)

$$\text{iv}$$
$$\text{iii}^x \qquad \text{ii}$$
$$\text{each guest} \qquad Q^w \qquad \text{i}$$
$$\text{what did e}_x \text{ bring}$$
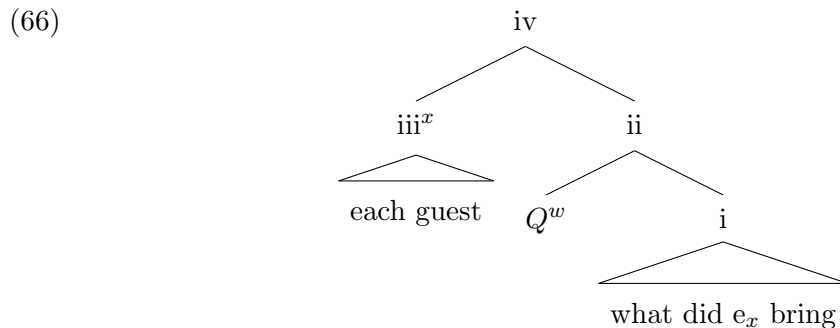
(67)   i : $\lambda y(brought(w)(y)(x))$

$Q : \lambda\alpha_{\langle\mathsf{s},\langle\mathsf{e},\mathsf{t}\rangle\rangle}\lambda p'_{\langle\mathsf{s},\mathsf{t}\rangle}\exists\beta_{\mathsf{e}}(p' = \lambda w'_{\mathsf{s}}(\alpha(w')(\beta)))$

ii : $\lambda p'_{\langle\mathsf{s},\mathsf{t}\rangle}\exists\beta_{\mathsf{e}}(p' = \lambda w'_{\mathsf{s}}(brought(w')(\beta)(x)))$

iii : $\lambda S_{\langle\mathsf{e},\mathsf{t}\rangle}\forall z(guest(w^*)(z) \rightarrow S(z))$

iv : $\lambda p'_{\langle\mathsf{s},\mathsf{t}\rangle}\forall z(guest(w^*)(z) \rightarrow \exists\beta_{\mathsf{e}}(p' = \lambda w'_{\mathsf{s}}(brought(w')(\beta)(z))))$

These answerhood conditions will almost certainly be unsatisfiable if there are two or more distinct guests, since a proposition p' satisfies these conditions iff, for each guest z, there is some β such that p' is the proposition that z brought β. If, for example, Rupert and Janna were both guests, then there must be some β such that

---

[5]Technically even the ordinary reading seen here introduces some issues, since it probably needs to be analyzed with abstraction over *kinds* rather than abstraction over *individuals*, but I will leave this issue aside.

p' is the proposition that Rupert brought β to the party and p' is also the proposition that Janna brought β to the party. Since these will typically be distinct propositions, they cannot both be equal to p', making the answerhood conditions unsatisfiable. An analogous issue arises in the strongly exhaustive case.

A further complication is that pair-list readings appear to be somewhat picky about which kinds of quantification they permit: such readings are quite natural with universal quantification, as seen in (62), but analogous wide-scope readings for most other quantifiers are, at best, controversial, so there is considerable debate (see, e.g. Szabolcsi (1997)) over whether an analogous wide scope is available even for (68):

(68)   What did two guests bring?

How should we handle the pair-list reading? We could attempt to explain away the pair-list reading as another guise of the functional reading. Solutions of this kind are explored by by Engdahl (1986) and Chierchia (1993). For reasons that Krifka (2001) (among others) discusses, I think this approach probably doesn't work. Other proposals involve finding a way to arrange a suitable wide scope for the universal, and several of these might be adapted to my framework.

Groenendijk and Stokhof (1984) explore more than one way of deriving pair-list readings. One of these, presented in Groenendijk and Stokhof (1982), derives the pair-list reading by a move that is, in the Groenendijk and Stokhof (1984) system, roughly analogous to scoping the universal above the X and below the Q. This approach, at least as given by Groenendijk and Stokhof (1984), will not work for my system, because it is dependent on a theory that privileges true answers, which I have argued against on the basis of data for *'be certain'*. This approach has the further difficulty that it is also not clear how to generalize it to derive a pair-list reading for a mention-some question. On the other hand, as discussed in Szabolcsi (1997) this approach does have the advantage that it naturally limits the availability of pair-list readings to universal quantifiers.

Krifka (2001) proposes a very different approach. For Krifka, pair-list readings arise not from universally quantifying into a proposition, but from universally quantifying into a speech act. Roughly, the idea is that the pair-list reading of (62) is to

be understood as *'For each guest x, I hereby ask you what x brought to the parry.'* I will not offer a detailed evaluation of this approach, but I will note that it does raise a couple of obvious concerns: one is that it is not clear how to achieve a sufficiently precise theoretical characterization of what it means to quantify into a speech act, and another is that it is not clear how to make this speech act approach work with embedded questions. Krifka (2001) engages with both these concerns, and the reader is directed there for a more detailed discussion.

For present purposes, two points are relevant. First, to the extent that this account can be made to work, it should be reasonably compatible with the theory of questions in Chapter 2. Nothing I do builds my question-meanings up to the level of speech acts, so assuming a speech act layer on top of my account will do no harm, and, once that layer exists, what happens below it should not be terribly relevant for quantifying into speech acts. (The theory of embedding, on the other hand, may need to be reframed in terms of speech acts, which raises a number of issues.) Second, the account in Krifka (2001) proposes to offer an explanation of the privileged status of universal quantifiers with respect to pair-list readings. Krifka notes that non-declarative speech acts can be targets for conjunction-like logical operations, but not for others. The reasoning is that universal quantification can be understood as multiple conjunction, while other forms of quantification cannot, so, if conjunction is the only logical operation that we can apply to non-declarative speech-acts, then universal quantification should be the only applicable kind of quantification.

Without resorting to the speech act level, the formal tools I have presented so far do present one way of quantifying into questions that should give rise to something like a pair-list reading. This idea is closely related to the approach of quantifying into 'lifted questions' discussed in Szabolcsi (1997). The idea is to use the quantifier raising account of question-embedding that I introduced earlier. The plan will be to actually scope the universal out of the question and above the $\exists_{QUES}$ operator. That is, the pair-list reading of (69) will be analyzed as in (70) and (71):

(69)   Joyce knows what each guest brought.

(70)

$$
\begin{array}{c}
\text{viii} \\
\diagdown \\
\text{vi}^p \qquad\qquad \text{vii} \\
\end{array}
$$

Tree (70):

- viii
  - vi^p
    - v^x — each guest
    - iv
      - $\exists_{QUES}$
      - iii
        - Q^w
        - ii
          - X
          - i — what e$_x$ brought
  - vii — Joyce knows e$_p$
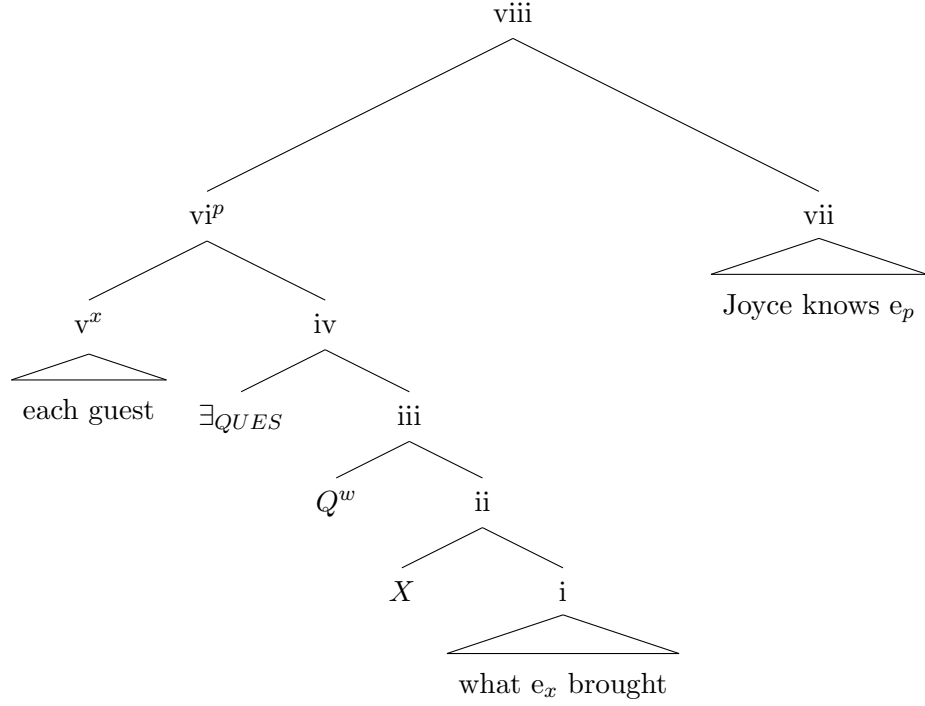
(71)

$$\text{i} : \lambda y(brought(w)(y)(x))$$

$$X_{\langle \mathsf{e,t}\rangle} : \lambda \gamma_{\langle \mathsf{e,t}\rangle} \lambda \delta_{\langle \mathsf{e,t}\rangle}(\gamma = \delta)$$

$$\text{ii} : \lambda \delta_{\langle \mathsf{e,t}\rangle}(\lambda y(brought(w)(y)(x)) = \delta)$$

$$Q_{\langle \mathsf{e,t}\rangle} : \lambda \alpha_{\langle \mathsf{s},\langle\langle \mathsf{e,t}\rangle,\mathsf{t}\rangle\rangle} \lambda p'_{\langle \mathsf{s,t}\rangle} \exists \beta_{\langle \mathsf{e,t}\rangle}(p' = \lambda w'_{\mathsf{s}}(\alpha(w')(\beta)))$$

$$\text{iii} : \lambda p'_{\langle \mathsf{s,t}\rangle} \exists \beta_{\langle \mathsf{e,t}\rangle}(p' = \lambda w'_{\mathsf{s}}(\lambda y(brought(w')(y)(x)) = \beta))$$

$$\exists_{QUES} : \lambda P_{\langle\langle \mathsf{s,t}\rangle,\mathsf{t}\rangle} \lambda P'_{\langle\langle \mathsf{s,t}\rangle,\mathsf{t}\rangle} \exists p''(P(p'') \wedge P'(p''))$$

$$\text{iv} : \lambda P'_{\langle\langle \mathsf{s,t}\rangle,\mathsf{t}\rangle} \exists p'' \left(\begin{array}{c} \exists \beta_{\langle \mathsf{e,t}\rangle}(p'' = \lambda w'_{\mathsf{s}}(\lambda y(brought(w')(y)(x)) = \beta)) \\ \wedge P'(p'') \end{array} \right)$$

$$\text{v} : \lambda S_{\langle \mathsf{e,t}\rangle} \forall z(guest(w^*)(z) \rightarrow S(z))$$

$$\text{vi} : \lambda P' \forall z \left(\exists p'' \left(\begin{array}{c} guest(w^*)(z) \rightarrow \\ \exists \beta(p'' = \lambda w'(\lambda y(brought(w')(y)(z)) = \beta)) \\ \wedge P'(p'') \end{array} \right) \right)$$

$$\text{vii} : knows(w^*)(p)(Joyce)$$

$$\text{viii} : \forall z \left(\exists p'' \left(\begin{array}{c} guest(w^*)(z) \rightarrow \\ \exists \beta(p'' = \lambda w'(\lambda y(brought(w')(y)(z)) = \beta)) \\ \wedge knows(w^*)(p'')(Joyce) \end{array} \right) \right)$$

That is, (69) is true iff, for every guest z, there is β such that Joyce knows that β is the set of things that z brought.

This account works pretty well for simple cases, but it suffers from a number of difficulties. The first of these is that it does not explain why universal quantification is special for pair-list readings. One possibility is to deny that it is (see Szabolcsi (1997) for some evidence of pair-list readings with non-universal quantifiers in embedded, but not unembedded, questions). A second difficulty is that it only generates a pair-list reading when we place the questions in a constituent that includes $\exists_{QUES}$. This would mean that, in order to use this account to derive pair-list readings for unembedded questions, we would need to claim that $\exists_{QUES}$ is always applied in question construction (or, equivalently, we would could write a new version Q that did both the work of Q and $\exists_{QUES}$). This would mean that, for the formalism, answerhood conditions would be conditions on *sets* of propositions, not on individual propositions. This change is not a deeply problematic one, but it does seem a bit like an arbitrary complication, and, beyond the case of pair-list readings, it is not clear how to motivate it.

A third potential issue for the approach to pair-list readings discussed above is that, at least as implemented here, it doesn't offer a clear way for the restrictor of the universal to be interpreted 'de dicto' with respect to the Q operator and the responsive-predicate. In (70), *'each guest'* is quantified in after the world variable associated with answer propositions has already been bound, so we have no way of making the truth-conditions in (viii) of (71) require that Joyce know anything about who is a guest. For each actual guest, she must know what the brought, but need not know that they were a guest. This is a consequence of our decision to bind the whole phrase *'each guest'* in. A more sophisticated account of quantification might allow *'each'* to assume wide scope while *'guest'* is interpreted in situ, but the nuanced exploration of the interaction between quantifier scope and intensional contexts that would be required to sort this out is beyond the scope of the present discussion. Of course, this is only an issue if the 'de dicto' interpretation is available. I don't know whether it is with respect to responsive predicates, but Groenendijk and Stokhof (1984) argue (I think convincingly) that it is available with respect to rogative predicates like *'ask'* and *'wonder'*, and getting it even there would require some refinement or clarification of this account.

I will not attempt to resolve these issues here. Instead, I will simply note that the pair-list reading seems to be plausibly analyzed in terms of some kind of wide-scope position for the universal, and that this wide scope, whatever it is, will be outside of the operators that form the core of my account of question-meaning. Further, I have argued that the general approach in Chapter 2 appears, tentatively, to be compatible with versions of both the lifted questions approach and the quantifying into speech-acts approach, making it reasonable to hope that a more sophisticated version of either approach may reasonably be adapted to something like the present account. I leave a more thorough exploration of pair-list readings for another time, and refer the reader to Krifka (2001), Szabolcsi (1997), and Chierchia (1993) for some discussion of the main issues involved.

3.6   Homogeneity Effects

Suppose that Anne knows that William or Alex was at the party, but she doesn't know which one, and she doesn't know anything about who else attended, so, in particular, she does not know a strongly exhaustive answer (or even a mention-some answer) to (24). Now consider the following dialog:[6]

(72)   Rupert: Who was at the party?
        Anne: I don't know who was at the party.

In this situation, Anne's response to Rupert's question is misleading. We might be tempted to call it a lie. If it is only misleading, this admits pragmatic explanation: if Rupert wants to know who was at the party, he would presumably prefer some information on this topic to none, even if the information provided is incomplete. If Anne has some information, and she's cooperative, she ought to provide it. If Anne's reply to Rupert consists entirely of a declaration of what she doesn't know, Rupert will infer that she has no information to contribute, making this response deceptive in a case where Anne does know something that bears on the question.

There is, however, a temptation to say that Anne's response is not merely misleading, but genuinely untrue. If this turns out to be the case (an issue that I won't

---

[6]This is a variation on an example suggested to me by Terence Parsons.

resolve here), I propose that we handle it by amending the embedding rules to include a presupposition of homogeneity.[7] That is, Anne knowing an answer to the question will make her reply true, and Anne knowing nothing that bears on the question will make her reply false, but Anne knowing only *part* of an answer will result in a presupposition failure.

The details of what exact notions of whole and part we should use here deserve more attention, but, as an illustrative example, I suggest simply using the partial answer set operator **Part** defined in Section 2.6.1. The first temptation might be to say that we should presuppose that, for every answer in the answer set, if Anne knows part of that answer then she must know the whole answer. This is not right: one partial answer can be a part of more than one answer, but, in general, it is sufficient to know one answer in the answer set, and not every answer that shares a part with that answer. In light of this, I suggest the following condition: if Anne knows a partial answer, then there is some proposition in the answer set that she knows and that entails that partial answer.

That is, to implement homogeneity presuppositions $know_{QUES}$ should be redefined as in (73), where $\mathbf{\Pi}$ is a presupposition triggering operator, read 'let us presuppose that...'

(73) $knows_{QUES} =$

$$\lambda w \lambda P_{\langle \langle \mathsf{s}, \mathsf{t} \rangle, \mathsf{t} \rangle} \lambda x_{\mathsf{e}} \left(\mathbf{\Pi} \left[\forall p' \left(\begin{array}{c} \left(\begin{array}{c} \mathbf{Part}(P)(p') \wedge \\ knows_{PROP}(w)(p')(x) \end{array} \right) \\ \rightarrow \\ \exists p'' \left(\begin{array}{c} P(p'') \wedge p'' \subseteq p' \wedge \\ knows_{PROP}(w)(p'')(x) \end{array} \right) \end{array} \right) \right] \\ \wedge \exists p'''_{\langle \mathsf{s}, \mathsf{t} \rangle}(P(p''') \wedge knows_{PROP}(w)(p''')(x)) \right)$$

More generally, for any responsive propositional predicate R_{PROP}, R_{QUES} will be defined as follows:

(74) $R_{QUES} =$

---

[7]I am indebted to Benjamin Spector for this suggestion.

$$\lambda w \lambda P_{\langle\langle \mathtt{s},\mathtt{t}\rangle,\mathtt{t}\rangle} \lambda x_\mathtt{e} \left(\Pi \left| \forall p' \left(\begin{array}{c} \left(\begin{array}{c} \mathbf{Part}(P)(p') \wedge \\ R_{PROP}(w)(p')(x) \end{array} \right) \\ \rightarrow \\ \exists p'' \left(\begin{array}{c} P(p'') \wedge p'' \subseteq p' \wedge \\ R_{PROP}(w)(p'')(x) \end{array} \right) \end{array} \right) \right. \\ \wedge \exists p'''_{\langle \mathtt{s},\mathtt{t}\rangle}(P(p''') \wedge R_{PROP}(w)(p''')(x)) \right)$$

That is, to our established existential quantification rule for question embedding, we add the presupposition that the question's answer set is homogeneous with respect to the responsive predicate.

The above is a bit hard to get a handle on. Perhaps providing some notation will help. Let us define the shorthand **Hom**, where $\mathbf{Hom}(\varphi)(\psi)$ is to be read 'φ is homogeneous in ψ':

(75) $\mathbf{Hom} = \lambda P \lambda P'(\forall p'' \left(\begin{array}{c} (\mathbf{Part}(P)(p'') \wedge P'(p'')) \\ \rightarrow \\ \exists p'''(P(p''') \wedge p''' \subseteq p'' \wedge P'(p''')) \end{array} \right))$

The intended case is the one where P is an answer set and P' is the set of propositions picked out by some responsive predicate. **Hom** is true of P and P' iff for every partial answer with respect to P that is in P', there is a member of P that entails that partial answer and is in P'.

Now we can restate the above definitions a bit more succinctly:[8]

(78) $knows_{QUES} =$

$\lambda w \lambda P_{\langle\langle \mathtt{s},\mathtt{t}\rangle,\mathtt{t}\rangle} \lambda x_\mathtt{e} \left(\begin{array}{c} \Pi(\mathbf{Hom}(P)(\lambda p'(knows_{PROP}(w)(p')(x)))) \\ \wedge \exists p''_{\langle \mathtt{s},\mathtt{t}\rangle}(P(p'') \wedge knows_{PROP}(w)(p'')(x)) \end{array} \right)$

(79) $R_{QUES} =$

---

[8]Everything said here is for the lexical rule implementation. The corresponding modification of $\exists_{QUES}$ is given by (77) (in contrast with (76), which is the definition given in Section 2.3.2).

(76) $\exists_{QUES} = \lambda P_{\langle\langle \mathtt{s},\mathtt{t}\rangle,\mathtt{t}\rangle} \lambda P'_{\langle\langle \mathtt{s},\mathtt{t}\rangle,\mathtt{t}\rangle} \exists p''(P(p'') \wedge P'(p''))$

(77) $\exists_{QUES} = \lambda P \lambda P'(\Pi(\mathbf{Hom}(P)(P')) \wedge \exists p''''_{\langle \mathtt{s},\mathtt{t}\rangle}(P(p'''') \wedge P'(p'''')))$

$$\lambda w \lambda P_{\langle\langle\mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle} \lambda x_{\mathbf{e}} \left(\begin{array}{c} \mathbf{\Pi}(\mathbf{Hom}(P)(R_{PROP}(w)(p')(x))) \\ \wedge \exists p'''_{\langle\mathbf{s},\mathbf{t}\rangle}(P(p''') \wedge R_{PROP}(w)(p''')(x)) \end{array} \right)$$

I will forgo an explicit tree using the above, since nothing about the order of composition has changed from the account in Chapter 2. I will also forgo an explicit production of a formula translating a sentence on this theory, since the complexity of the resulting formula would make it unnecessarily obscure, and it is easier to read the main semantic effects off of the definition of ref $know_{QUES}$.

The idea, again, is that $know_{QUES}$ is evaluated in a world (w), and its main arguments are an answer-set (P) contributed by the embedded question, and and an entity (x) contributed by the subject of 'know'. Filling these argument positions gives us a presupposition that P is homogeneous in x's knowledge (every partial answer that x knows is part of an answer that x knows), and an assertion that x knows at least one proposition in P (x knows an answer to the question). If we negate this, the assertion becomes an assertion that x does not know any answer to the question, and the presupposition remains the same. In this case, since the subject doesn't know any answers, and since we've presupposed that if the subject knows a partial answer then they know an answer, the sentence will only be true if the subject knows neither an answer nor a partial answer. Thus, we predict that, in a world where Anne knows a partial answer, but not a complete answer, to the embedded question (24), both (80) and (81) will suffer presupposition failure:

(24)   Who was at the party?

(80)   Anne knows who was at the party.

(81)   Anne doesn't know who was at the party.

The formalization of partial answerhood that I have adopted was not defined with this particular purpose in mind, and it may not be the right notion of partial answer for the analysis of homogeneity effects. More work in this area is needed.

3.7 Conclusion

In this chapter, I have explored a number of potential challenges to the baseline theory in Chapter 2. These have included arguments for weak exhaustivity, the non-exhaustivity effect with *'surprise'*, granularity effects, the truthiness effect with *'tell'*, pair-list readings of *'wh'*-questions, and homogeneity effects. I have argued that some of the major arguments for weak exhaustivity are unconvincing (although I have not thoroughly surveyed the space of possible arguments), and that granularity effects likewise pose no real threat, while recognizing that more needs to be said about the other challenges.

For *'surprise'*, the main problem is that this predicate *always* seems to select for mention-some readings, to the exclusion of exhaustive readings. This can, in a pinch, be handled by simply allowing predicates to select for the presence or absence of the X operator in embedded *'wh'*-questions. Some related issues are explored in more detail in Chapter 6. In any case, the threat seems modest – *'surprise'* uses a kind of answerhood conditions already included in the baseline theory, and the only mystery is why it favors these answerhood conditions even in cases where other responsive predicates do not.

For *'tell'*, we have another genuine puzzle: *'tell'* with an embedded question gives rise to an implicature of truthiness for the proposition communicated. Since this effect behaves like an implicature, and rises with certain other uses of *'tell'*, I think that an account of it will require us to look beyond the scope of our theory of question embedding, but an account is still needed, and I am unable to offer one.

For pair-list readings, I think that the necessary reading should be derived by somehow giving the universal quantifier that triggers the pair-list reading wide-scope with respect to the entire question meaning, and I have discussed two general ways that we might do this (scoping over speech acts, and scoping over lifted questions). I haven't resolved the issue, but I think my remarks at least suggest some possible directions forward.

For homogeneity effects, I have suggested a specific amendment to the theory, but have not engaged in a detailed exploration of the adequacy of this amendment or the necessity of putting homogeneity into the semantics. I leave a more detailed discussion of homogeneity for another time.

The main purpose of this chapter has been twofold: first, I have tried to convince the reader that weak exhaustivity can safely be excluded from the theory of question-embedding, or at least that it is not nearly as necessary as is widely believed. I am reasonably confident of this, and I think my arguments have been reasonably convincing. Second, I have presented some problem cases that I think can reasonably be regarded as minor, and sketched possible directions for dealing with them. Most of these, especially pair-list readings, have been discussed elsewhere, and I do not consider them urgent. In subsequent chapters, I will turn to problems that I consider more serious, and that are less well-studied within the family of frameworks to which the account in Chapter 2 belongs.

CHAPTER 4

(Non-)Reducibility

You keep using that word.

I do not think it means what you think it means.

<div align="right">Goldman (1987)</div>

4.1 Reducibility Assumptions

4.1.1 Introduction to the Reducibility Property

In earlier chapters, I made my best case for a relatively simple and near-uniform theory of question-embedding under responsive predicates. In my semantics of questions, *'wh'*-questions were ambiguous between two kinds of answerhood conditions: strongly exhaustive answerhood conditions (exemplified by (1-c), derived as (1-b), as an expression of the answerhood conditions for (1-a)) and mention-some answerhood conditions (exemplified by (2-c), derived as (2-b), as an expression of the answerhood conditions for (2-a))

(1)   a.   Which students passed?

      b. $Q(\lambda w_{\mathbf{s}}(X(\lambda x_{\mathbf{e}}(student(w)(x) \wedge passed(w)(x)))))$

      c. $\lambda p_{\langle \mathbf{s},\mathbf{t}\rangle}\exists \beta_{\langle \mathbf{e},\mathbf{t}\rangle}(p = \lambda w_{\mathbf{s}}(\lambda x_{\mathbf{e}}(student(w,x) \wedge passed(w,x) = \beta)))$

(2)   a.   Who can help?

      b. $Q(\lambda w\lambda x\exists w'(can(w,w') \wedge helps(x,w')))$

      c. $\lambda p_{\langle \mathbf{s},\mathbf{t}\rangle}\exists \beta_{\mathbf{e}}(p = \lambda w\exists w'(can(w,w') \wedge helps(\beta,w')))$

I adopted a uniform question-embedding rule, which stated that a responsive predication holds of a question if it holds of some answer to the question, so that, for example, the truth-conditions of (3-a) are given by (3-b), and the truth-conditions

of (4-a) are given by (4-b):

(3) a. Rupert knows which students passed.

   b. $\exists p_{\langle \mathbf{s},\mathbf{t} \rangle} \left(\begin{array}{c} Q(\lambda w_{\mathbf{s}}(X(\lambda x_{\mathbf{e}}(student(w)(x) \wedge passed(w)(x))))))(p) \\ \wedge knows(w^*)(p)(Rupert) \end{array} \right)$

(4) a. Red knows who can help.

   b. $\exists p_{\langle \mathbf{s},\mathbf{t} \rangle} \left(\begin{array}{c} Q(\lambda w \lambda x \exists w'(can(w,w') \wedge helps(x,w')))(p) \\ \wedge knows(w^*)(p)(Red) \end{array} \right)$

I have also described a variety of competing approaches to question embedding that involved other kinds of question meanings and other embedding rules. For example, most theories of strong exhaustivity (e.g. Groenendijk and Stokhof (1984); Heim (1994)) do not existentially quantify over strongly exhaustive answers as in (3-b) above, but instead compute the unique true strongly exhaustive answer to the question, and check the responsive predicate against that. So, for example, an account along the lines of Groenendijk and Stokhof (1984) says that the extension of any *'wh'*-question is its unique true strongly exhaustive answer, so that the extension of (1-a) will be (5-a), or more explicitly (5-b), and the truth-conditions of (6-a) will then be given by (6-b):

(5) a. $\mathbf{Strong}(w^*)(\lambda w_{\mathbf{s}} \lambda x_{\mathbf{e}}(student(w)(x) \wedge passed(w)(x)))$

   b. $\lambda w' \left(\begin{array}{c} \lambda x_{\mathbf{e}}(student(w^*)(x) \wedge passed(w^*)(x)) \\ = \lambda x_{\mathbf{e}}(student(w')(x) \wedge passed(w')(x)) \end{array} \right)$

(6) a. Rupert told Janna which students passed.

   b. $told(w^*)(J)(\mathbf{Strong}(w^*)(\lambda w \lambda x(student(w)(x) \wedge passed(w)(x))))(R)$

Similarly, one might, following the general approach advocated by Lahiri (2002), analyze (7-a) by existentially quantify over only the true mention-some answers, giving it the truth conditions in (7-b):

(7) a. Red told Anne who could help.

   b. $\exists p_{\langle \mathbf{s},\mathbf{t} \rangle} \left(\begin{array}{c} Q(\lambda w \lambda x \exists w'(can(w,w') \wedge helps(x,w')))(p) \wedge p(w^*) \\ \wedge told(w^*)(Anne)(p)(Red) \end{array} \right)$

116

I also noted that many theorists have suggested that in some cases we must check the weakly exhaustive answer (essentially the conjunction of all the true mention-some answers) against the predicate.

Within the tradition I've been discussing, these (or minor variations on them) are very nearly the only possibilities that are considered. The theorists I've been talking about often disagree about which answers to the embedded question are to be checked against a responsive predicate, taken as a predicate of propositions. They also differ about which of these alternatives are available for which predicates, but, with few exceptions, they all draw from this list of options. All of these options share a central characteristic. They reduce the question-embedding use of a responsive predicate to the meaning of its propositional use. That is, each account says that we can assess the truth of these question-embedding sentences by first computing the answer or answers to the embedded question and then checking some or all of them against the extension of the propositional predicate.[1] Each responsive predicate defines a relation between entities and propositions (or, as in the case of *'tell'*, a ternary relation between two entities and a proposition). If a theory uses this kind of reduction approach, then this relation will be entirely determined by the responsive predicate's behavior as a relation between entities and propositions.

This property is shared by some other, more often overlooked accounts of question embedding. For example, Karttunen (1977) gives treatment of embedding under *'know'* that defines the truth-conditions of a sentence like (3-a) entirely in terms of facts about which propositions Rupert knows, although the propositions tested are, in modern terms, a curious mix of strongly and weakly exhaustive answers.

Another way of saying this is to say that there is almost universal tacit agreement that (typical) responsive predicates share the following property:

---

[1]Indeed, Lahiri (2002) comes close to giving this as the definition of what it is to be a responsive predicate: for Lahiri, the alleged availability of (8-a) as a paraphrase of (8-b) is the key property that distinguishes the *'know'* class of embedders from the *'wonder'* class:

(8)   a.   Red knows what Anya bought.
           b.   Red knows some proposition p that answers the question *'What did Anya buy?'*

We will see, however, that the implementation in Lahiri (2002) hides some non-reducibility in the *'p* answers the question...' part of this.

(9)   Reducibility Property: a responsive predicate R has the 'reducibility property' if, for every two entities a and b, if R relates a to exactly the same propositions that R relates b to, then R relates a to exactly the same questions that R relates b to.[2]

All the accounts of embedding under responsive predicates described immediately above attribute the reducibility property to those predicates. Consider my own account of *'know'* as an example. Since I assume questions are ambiguous between strongly exhaustive and mention-some readings, the strongly exhaustive case and the mention-some case must be considered separately. For the strongly exhaustive case, my account of *'know'* says that an entity is related to a question by *'know'* if and only if the entity is related to one of the strongly exhaustive answers of that question by *'know'*. To prove that this *'know'* has the reducibility property, consider any two individuals a and b such that a and b are related to exactly the same propositions by *'know'*. For any question, and every p a strongly exhaustive answer to that question, a knows p if and only if b knows p, since a and b know exactly the same propositions. Thus, a and b know strongly exhaustive answers to exactly the same questions, which means that *'know'* relates a to exactly the same questions that b does, showing that *'know'* has the reducibility property with respect to the strongly exhaustive reading of a question.

The argument that *'know'* has the reducibility property with respect to embedded mention-some questions is essentially the same. Suppose a and b know the same propositions. This means that, for any question, any mention-some answer to that question is among the propositions that a knows if and only if it is among the propositions that b knows, so *'know'* relates a to the question if and only if it relates b to the question. So, with mention-some readings, we again see that if a and b are related to the same propositions by *'know'* then *'know'* will relate them to the same questions as well.

Since my account as developed so far treats all responsive predicates the same, nothing is special about *'know'* here: we could substitute any other responsive predicate. Thus, my account attributes the reducibility property to all responsive predi-

---

[2]This property is framed in terms of responsive predicates with exactly one entity-type argument and one clausal argument. The generalization to responsive predicates with more arguments, such as *'tell'*, is straightforward, but I will mostly ignore such predicates for the remainder of this chapter.

cates.

The alternative accounts of embedding under responsive predicates also have the reducibility property. Take, for example, an account like the account of 'extensional' predicates in Groenendijk and Stokhof (1984), on which a predicate R relates an entity a to a question if and only if R relates a to the unique strongly exhaustive answer to that question. Suppose that a given R relates individuals a and b to exactly the same propositions. Now consider any question, and let p be its unique true strongly exhaustive answer. Since R relates a and b to the same propositions, R relates a to p if and only if it relates b to p. This means that R relates a to a question if and only if it relates b to that question, showing that Groenendijk and Stokhof (1984) attribute the reducibility property to all the predicates that they call 'extensional' (including *'know'*).

If, we instead consider a weak exhaustivity account, the argument will look the same as the above: any question has a unique weakly exhaustive answer, so, if a predicate R relates a and b to the same propositions, it will relate a to a given question's answer if and only if it relates b to that same answer. Generally, any theory that says being related to a question by *'know'* (or any other responsive predicate) is the same as knowing the answer to that question will attribute the reducibility property to *'know'* – if two people know the same propositions, then it must be that both know the answer proposition, or neither does.

The assumption of some kind of reducibility is so intuitively appealing that we hardly think of it. It is the core intuition that underlies the whole project of these kinds of theories of embedding. Some of the more marginal and problematic responsive predicates (especially *'agree'*, as can be seen in the next section) are occasionally exempted from reducibility, but for core examples, like *'know'*, this kind of reduction is accepted almost universally.

In this chapter, I want to argue that the reducibility property is not satisfied by certain core examples of responsive predicates in English. In particular, I want to argue that *'know'* lacks this property. This is a problem for virtually all available theories, which, one way or another, define the semantics of question-embedding *'know'* in terms of propositional *'know'*.

If my argument is correct, and we cannot depend on the reducibility property, we need to re-evaluate the link between the question-oriented and propositional uses of

a responsive attitude. Traditionally, the link has been maintained by reduction, but, if we accept that reduction (at least in the traditional sense) is not always possible, then we need to say something new about how to constrain the link. I will explore some ways to do this while still maintaining a kind of uniformity.

4.1.2 A Note on my Definition of Reducibility

The formulation of the reducibility property in (9) is, I think, expansive enough to include all predicates that we would be inclined to call reducible, but it fails to rule out various predicates that we might want to characterize as being non-reducible. It is not as restrictive as it could be in a number of important ways.

One issue is that it places no restrictions on which propositions we look at, so, unlike the accounts of question embedding discussed so far, it does not enforce any requirement that we look only at propositions that have something to do with the question. To see an example of an unnatural and intuitively non-reductive question-embedding behavior allowed by this, consider the hypothetical adjective *'monocero-scient'*. *'monoceroscient'* relates an entity to a proposition if the entity knows the proposition (so (10-a) will be true if and only if Rupert knows that Anya can help), but relates an entity to a question if and only if that entity knows a proposition in the answer set of that question and knows that there are no unicorns (so, in a world where Red and Janna are the only ones who can help, (10-b) will be true (on a mention-some reading) if and only if Rupert knows that there are no unicorns, and either knows that Red can help or that Janna can help).

(10)   a.   Rupert is monoceroscient that Janna can help.
         b.   Rupert is monoceroscient of who can help.

To see that *'monoceroscient'* has the reducibility property, consider any two individuals a and b, and suppose that *'monoceroscient'* relates a and b to exactly the same propositions, which is to say a and b know exactly the same propositions. Let P be the set of propositions known by a (which is also the set of propositions known by b), and, for any question, let P' be the set of mention-some answers for that question. If P contains at least one proposition in P', and also contains the proposition that there are no unicorns, then *'monoceroscient'* will relate both a and b

to the question with answer set P'. If, on the other hand P does not overlap with P', or fails to contain the proposition that there are no unicorns, then *'monoceroscient'* relates neither a nor b to the question with answer set P'. Thus, *'monoceroscient'* relates a to a question if and only if it relates b to that question. Since we reached this conclusion from just the assumption that *'monoceroscient'* related a an b to the same propositions, it follows that *'monoceroscient'* has the reducibility property in the sense defined in (9). In spite of this, the semantics given for *'monoceroscient'* is intuitively non-reductive: the part about having to know that there are no unicorns seems to come out of nowhere, without any sensible relationship to the behavior of *'monoceroscient'* as a propositional attitude.

Another reason that the definition of reducibility in (9) is too inclusive is that it only involves comparison between subjects within worlds, and never involves comparison across worlds, meaning that it considers a question-oriented semantics for a predicate to be reducible even if the method of the reduction depends on unrelated facts about the world. For an example of a formally possible but unnatural predicate that exploits this weakness of my definition, consider the hypothetical verb *'wixplain'*. The stipulated semantics of *'wixplain'* are as follows: *'wixplain'* relates an entity a to a proposition p iff p is true and a explained that p. (so (11-a) is true if and only if Rupert explained that William stole the documents, and William in fact stole the documents). *'wixplain'* relates an entity a to a question with answer set P if and only if there is a proposition $p' \in P$ such that the following conditions are met: first, p' must be true, second, the a must explain p', and, third, a sentence expressing p' must appear somewhere in the text of the Anglo-Saxon edition of Wikipedia (so, in a world where William stole the documents, and was the only one to do so, (11-b) is true (on a mention-some reading) if and only if Rupert explained that William alone stole the documents, and a declaration that William alone stole the documents appears somewhere in the Anglo-Saxon edition of Wikipedia).

(11)   a.   Rupert wixplained that William stole the documents.
        b.   Rupert wixplained who stole the documents.

The reader can confirm that *'wixplain'* has the reducibility property as defined in (9) by reasoning similar to that used for *'monoceroscient'* above, but, again, the

question-oriented semantics that I provided for *'wixplain'* are not intuitively a simple reduction to its propositional semantics.

Other examples of perverse predicates that obey my formal reducibility are possible, including *'quunknow'*. *'quunknow'* relates an individual and a proposition if and only if the individual knows the proposition, and it relates an individual to a question if and only if the individual doesn't know the true strongly exhaustive answer to that question. That is, *'quunknow'* has exactly the same semantics as *'know'* does in Groenendijk and Stokhof (1984), except that it sneaks in an extra negation in the question-embedding case for no good reason.

For these reasons, the definition of reducibility given in (9) is clearly not constraining enough to be a good general-purpose definition of the reducibility property. Nevertheless, it is the definition that I will use for the rest of this chapter. The reason for this is that the major project of this chapter is to show that a number of responsive predicates, including the prototypical responsive predicate *'know'*, do not have the reducibility property. When trying to prove that something lacks a property, it is better to err on the side of being over-broad in one's definition of that property. The more narrowly I define reducibility, the greater the risk that my arguments against reducibility will fail to apply to some unforeseen account that is intuitively reductive, but ruled out by my definition. By defining reducibility broadly, I hope to reduce this risk, and maximize the formal applicability of the non-reducibility claims that appear below.

4.2   Some Non-Reductive Accounts

Reducibility is implicit in many treatments of responsive predicates, but some earlier accounts have attributed non-reductive meanings to particular predicates (generally on an ad hoc basis). I review the most prominent of these below. One reason for doing this is a simple obligation not to overstate my own contribution: although I don't know of any previous author who has explicitly articulated and rejected reducibility, I am not the first person to attribute a non-reducible semantics to a responsive predicate and in particular the non-reductive accounts found in Spector (2005) and Lahiri (2002), discussed below, anticipate many of the themes my treatment of non-reducibility. Another issue is that, if we are to reject reducibility, we should look at

how reducibility fails in trying to craft our response.

4.2.1 Beck and Rullmann's *'agree'*

Beck and Rullmann (1999) provide a semantics of *'agree'* that denies *'agree'* the reducibility property. Paraphrased slightly, their proposed semantics is that for any expression a denoting a plurality of entities x, and any embedded question φ with mention-some answer-set P, the sentence *'a agree on φ'* is true iff for all y and z that are atomic parts of x, and every p' in P, y believes p' iff z believes p', and y believes the negation of p' iff z believes the negation of p'.

These truth-conditions lack the reducibility property. To see this, consider the case where we have three people: Anne, William, and Joyce. Suppose that all three believe that Kendra was murdered, that Anne believes that Red was murdered, that neither William nor Joyce has any opinion about whether Red was murdered. Suppose further that none of Anne, William, and Joyce have any further beliefs about who was murdered, and that their beliefs on other matters correspond perfectly. People's individual beliefs can thus be represented by table (12):

(12)

Murder victim:	Kendra?	Red?
A's beliefs:	T	F
W's beliefs:	T	?
J's beliefs:	T	?

In this situation the judgment, which so far as I know is uncontested, is that Anne and William agree that Kendra was murdered, and to no other relevant proposition, since William is unopinionated about Red's status. Likewise, William and Joyce agree that Kendra was murdered, and to no other salient propositions:

(13)

	Kendra murdered	K not murdered	Red murdered	R not murdered
A and W agree?	Y	N	N	N
W and J agree?	Y	N	N	N

That is, the set of propositions agreed to by the plurality consisting of Anne and

William is exactly the same as the set of propositions agreed to by the plurality consisting of William and Joyce. Thus, if *'agree'* has the reducibility property, (14-a) and (14-b) should either both be true or both be false:

(14)   a.   Anne and William agree on who was murdered.

           b.   William and Joyce agree on who was murdered.

In the scenario just sketched, Beck and Rullmann's account predicts, not implausibly, that (14-a) is false while (14-b) is true. This contradicts reducibility, so Beck and Rullmann's *'agree'* lacks the reducibility property.

Before making some (unfortunately inconclusive) remarks on the empirical claim about the truth values of (14-a) and (14-b) in the scenario described above, I want to briefly review how Beck and Rullmann's account derives this prediction. The key feature of Beck and Rullmann's semantics for questions embedded under *'agree'* is that it is sensitive to the distinction between two different ways of failing to agree to a proposition. On the one hand, we have the lack of agreement between Anne and William regarding the proposition that Red was murdered. This lack of agreement involves a difference of beliefs, since the proposition that Red was not murdered is among Anne's beliefs, but not William's. According to Beck and Rullmann (1999), this kind of non-agreement falsifies question-oriented agreement: because Anne believes an answer that William does not believe, (14-a) comes out false. On the other hand, we have the lack of agreement between William and Joyce. This lack of agreement does not involve any *dis*agreement, but only a shared lack of opinion. In Beck and Rullmann's account, this kind of non-agreement, unlike the kind that involves belief mismatch, does not falsify question-oriented agreement, so (14-b) comes out true.

If Beck and Rullmann's account of *'agree'* is right, then *'agree'* is an example of a non-reducible responsive predicate. Unfortunately, it is difficult to know how to proceed, since it is not clear how empirically adequate Beck and Rullmann's *'agree'* is. This raises two different complications. First, if Beck and Rullmann are wrong about the judgment for pairs like (14-a) and (14-b) above, then this particular problem for reducibility disappears. If these judgments are right (and if domain effects, presupposition accommodation, and similar confounds can be controlled adequately),

then we must recognize *'agree'* as non-reducible, but that doesn't by itself tell us what the general question-embedding semantics of *'agree'* is, making it difficult to decide how to handle the non-reducibility.

Claims about the truth-conditions of sentences like (14-b) vary, with Egré and Spector (2007) offering a characterization on which this sentence will be untrue in the scenario described.[3] My own intuitions are uncertain with regard to cases like the one under consideration: my initial reaction is to characterize them as untrue, but I have to admit that (14-b) sounds a lot better as a description of the scenario described above than a false (or strongly presupposition-failing) sentence usually does. Some informants express unease with these examples, saying that in the case described above, William and Joyce don't seem to exactly agree, but don't disagree either. Egré and Spector (2007) and Lahiri (2002) suggest that *'agree'* presupposes opinionatedness, which would account for this unease with (14-b). The judgments of truth that other informants report for (14-b) might then be the result of some narrowing of the domain restriction to avoid presupposition failure (something like this is discussed in Lahiri (2002)), but it is not clear how plausible this is. On the face of it, though, many people seem to get the contrast between (14-a) and (14-b), which is enough to make this a serious challenge for reducibility.

Even though this data point seems right, Beck and Rullmann's more general picture of the semantics of *'agree'* is empirically less adequate on some other points. The most conspicuous of these is that it predicts that a sentence like (14-b) should be true in a case where neither William nor Joyce has any beliefs whatsoever about who was murdered, which is contrary to our intuitions. This indicates that Beck and Rullmann's *'agree'* is not quite right as formulated, but it don't do anything to address the judgments for (14-a) and (14-b), which, if true, are enough to establish non-reducibility. In light of the uncertain judgments, and the difficulties for Beck and Rullmann's empirical generalization, even the description of the facts for *'agree'* remains a substantial unresolved problem. I have nothing to add to this issue at this

---

[3]Egré and Spector (2007) offer an account of *'agree'* similar to the one derived by the baseline theory in Chapter 2: *'agree'* relates a plurality of entities to a question if and only if it relates that plurality to a strongly exhaustive answer to the question. This would make (14-b) untrue in the scenario described. Neither William nor Joyce believes a strongly exhaustive answer: they both believe that Kendra was murdered, but this this is the limit of their beliefs on the matter, so neither has any belief that identifies some set as the set of all murdered people. Since neither believes a strongly exhaustive answer individually, such an answer is not among the propositions they agree to, so (14-b) will be untrue on Egré and Spector's account.

time, and so I leave the matter of *'agree'* aside as a problem for future investigations.

4.2.2 Heim's *'tell'*

Heim (1994) proposes a semantics for question-embedding *'tell'* that is not reducible to propositional *'tell'*. For Heim, (15) is true iff Alex spoke the true weakly exhaustive answer to the embedded question to Rupert, and, by speaking the true weakly exhaustive answer, he succeeded in communicating the true strongly exhaustive answer.

(15)   Alex told Rupert who Anne loved.

That is, suppose that Anne loves only Liam. In this case, the claim is that (15) is true iff both Alex told Rupert the weakly exhaustive answer (16), and, by so doing, succeeded in communicating the strongly exhaustive answer (17) to Rupert.

(16)   Anne loves Liam.

(17)   Liam is the only one Anne loves

To see that these proposed truth-conditions for (15) are non-reductive, consider the case where both Alex and Red spoke the true weakly exhaustive answer to Rupert (both told rupert (16)), and neither told Rupert anything else, but, for one reason or another, Alex only succeeded in communicating the true weakly exhaustive answer (that is, Alex communicated (16) but not (17)), while Red managed to communicate the weakly and strongly exhaustive answers (that is, Red communicated (16) and (17)), as shown in (18):

(18)

	told Rupert:	communicated to Rupert:
Red:	(16)	(16), (17)
Alex:	(16)	(16) only

In this case, assuming neither Alex nor Red told Rupert anything else, Alex and

Red arguably told Rupert the same propositions, but (15) is predicted to be untrue, while (19) is predicted to be true.

(19)   Red told Rupert who Anne loved

In this scenario, both Red and Alex told Rupert exactly the same propositions (both told Rupert (16) and nothing else), but, according to Heim's analysis of *'tell'*, *'tell'* relates Red and Alex to different questions, since the claim is that (15) is false while (19) is true. Thus, this proposed semantics for *'tell'* lacks the reducibility property.

Although this is an interesting example of a non-reductive proposal for the semantics of a responsive predicate, I do not think it is an empirically adequate proposal. In Section 3.3, I presented arguments (due, for the most part, to Egré and Spector (2007)) that *'tell'* is non-veridical (at least in its truth-conditions, if not in its implicatures), making the emphasis on true answers inappropriate, and I also suspect that the requirement of speaking the weakly exhaustive answer is not a strict truth-conditional requirement, but rather an inference arising from the fact that speaking the weakly exhaustive answer is a common conventional way to communicate the strongly exhaustive answer.[4] For these reasons, I am not convinced that *'tell'* is non-reducible in the way suggested by Heim. I acknowledge, however, that the question-embedding behavior of tell is mysterious in several ways, and that further exploration is needed.

4.2.3   Spector's Weakly Exhaustive *'know'*

In the previous chapter, I gave my reasons for rejecting the use of weakly exhaustive answers in the semantics of question embedding. As weak exhaustivity is usually articulated (e.g., as it is articulated by Sharvit (2002)), one knows a question on a weakly exhaustive reading iff one knows the weakly exhaustive answer to the question. This formulation of weak exhaustivity has another serious empirical flaw. Consider the situation where Alex believes, with equal confidence, and on evidence that to him seems comparable, that, among his colleagues, Faith, William, Andrew, and Robin are spies. Suppose that in fact Faith and William are spies, but Andrew

---

[4]This effect, discussed at length by Groenendijk and Stokhof (1984), is interesting and important, but won't be explored here.

and Robin are not. Suppose further that Alex's belief that Faith and William are spies has whatever characteristics it needs to constitute knowledge. That is, Alex's beliefs and the facts of the world are as in (20):

(20)

	Faith	William	Andrew	Robin
Alex's beliefs:	spy	spy	spy	spy
the facts:	spy	spy	non-spy	non-spy

In this case, the weakly exhaustive answer to the question (21-a) is (21-b). Since Alex knows (21-b) (i.e. (21-c) is true), and (21-b) is the weakly exhaustive answer to (21-a), Alex knows the weakly exhaustive answer to (21-a). If a standard weakly exhaustive reading is available for (21-d), then it will be true in this situation.

(21) a. Which of Alex's colleagues are spies?
      b. Faith and William are spies.
      c. Alex knows that Faith and William are spies.
      d. Alex knows which of his colleagues are spies.

This prediction of weak exhaustivity is not supported by intuition: (21-d) seems pretty clearly false in the scenario described by (20), since Alex falsely believes that Andrew and Robin are spies, and they are not. Alex's knowing the weakly exhaustive answer (21-b) is simply not enough to make (21-d) true.

For an account, like the mine, that does without weak exhaustivity, this is not a problem. If (21-d) involves strongly exhaustive answerhood conditions, then (21-d) will not be true, since the strongly exhaustive answer (22) is not among Alex's beliefs (and in fact directly contradicts his belief that Andrew and Robin are spies):

(22)   Faith and William are the only spies among Alex's colleagues.

If one does believe that a weakly exhaustive reading is available for (21-d), the judgment of untruth for (21-d) is a serious problem. One response is to say that this sentence has only a strongly exhaustive, and not a weakly exhaustive reading, or to say that the weakly exhaustive reading is dispreferred. This is the usual interpretation of these kinds of examples, as discussed by Lahiri (2002), among others.

Another reply is to augment the weakly exhaustive semantics for *'know'* to rule out this problematic case. Spector (2005) takes the second approach to this problem. In essence, Spector (2005) proposes that there is a question-embedding use of *'know'* that makes (21-d) true if and only if Alex knows the weakly exhaustive answer to (21-a) *and* every mention-some answer to (21-a) that Alex believes is true. In the scenario under consideration, this means that for (21-d) to be true, Alex must know (21-b), and no proposition attributing spyhood to anybody else may be among Alex's beliefs. On Spector's semantics for *'know'*, (21-d) is false because Alex believes that Andrew and Robin are spies, and these beliefs are false.

Spector's semantics of *'know'* is non-reductive: in it, the truth conditions of (21-d) depend not just on Alex's propositional knowledge, but also on his beliefs. Spector does not connect his semantics for *'know'* with any notion of reducibility, and so does not point out that this represents a major departure from our normal, tacit assumptions about responsive predicates. Nevertheless, this is a major departure, and one that we should take seriously. Although I think that Spector's account is mistaken in one respect (its use of weak exhaustivity), I think that its reasons for being non-reductive are in essence correct. If one does believe that a weakly exhaustive reading is available, then the judgment reported above for (21-d) makes the abandonment of the reducibility assumption unavoidable.

If we reject weak exhaustivity, the (21-d) judgment is not itself problematic, since (21-a) seems to favor an exhaustive reading, and, with strong exhaustivity, our predictions match the judgment. However, an analogous argument can, with slight changes, be applied to cases involve mention-some readings. The argument, which is fleshed out in Section 4.3.1, makes what I think is a reasonably compelling case for abandoning reducibility. This issue is also not limited to *'know'*: in Section 4.3.2, I argue that a similar issue arises with *'forget'*. After going through these arguments, I will explore some possible replies to the problem.

4.2.4   Lahiri's Restricted Answer Quantification

Lahiri (2002) allows one major source of non-reducibility.[5] In Lahiri's system, quantification over mention-some answers is restricted by a covert domain restriction.

---

[5]This discussion is based on Lahiri's sections 3.2.2.5, 3.2.2.8, and 3.5.3, along with Appendix 4 of Lahiri's Chapter 3.

This restriction is supplied in part by context, but is partly selected by the responsive predicate under consideration. The possibility of attributing this restrictor selection to presupposition accommodation is entertained, but is ultimately rejected for reasons related to the idea that question-embedding *'tell'* is veridical. For Lahiri (2002), the basic analysis of (23) would be something like (24), where $\varphi(p)$ is my stand-in for Lahiri's formula saying that p is a mention-some answer for the embedded question *'who came to the party?'*.

(23)   Alex told Rupert who came to the party.

(24) $\forall p((\varphi(p) \wedge C(p)) \rightarrow told(w^*)(Rupert)(p)(Alex))$

For Lahiri, *'tell'* imposes the following constraint on C (when evaluating in w^*):

(25) $C \subseteq \lambda p(p(w^*))$

That is, (23) is supposed to be true iff Alex told Rupert all the contextually salient true answers to *'who came to the party?'*. This universal quantification over mention-some answers achieves a version of weak exhaustivity. Existential quantification (with the same restriction to C) is also permitted to derive mention-some readings, and a variant strong exhaustivity operator that incorporates C is defined in Appendix 4 of Chapter 3 of Lahiri (2002). The condition imposed by (25) is, according to Lahiri, a relatively common one, applying to *'know'* and *'communicate'* as well as tell.

As I've defined reducibility, (25) is not a source of non-reducibility, but the constraints on C selected by some other predicates in Lahiri's system do give rise to non-reducibility. In particular, in Lahiri's account of embedding under *'be certain'*, C must be a subset of the propositions that the subject considers possible (the possibility of a further restriction to propositions the subject considers likely is entertained), so (26) is supposed to be true iff, for every p that is a salient mention-some answer that Rupert considers possible, Rupert is certain that p.

(26)   Rupert is certain of who came to the party.

On the variant where C is only restricted to be a subset of the propositions that rupert considers possible, it is not clear whether we have non-reducibility.[6] The variant where we restrict to propositions the subject considers likely does appear to demonstrate non-reducibility, however. To see this, consider the case where there are only two salient answers to (27): (28) and (29).

(27)   Who came to the party?

(28)   Janna came to the party.

(29)   William came to the party.

Now suppose that Alex and Red are both certain that Janna came to the party, but that Alex considers William's attendance likely, while Red considers it possible but not especially likely. Schematically, we have the state of affairs in (30):

	J attended	W attended
A thinks...	definitely	probably
R thinks...	definitely	maybe

(30)

   Assume that Alex and Red are in complete agreement about all other subjects. In this situation, Alex and Red are certain of exactly the same propositions. However, if (28) and (29) are the only salient answers, then, on Lahiri's account seems to predict that (31) is true while (32) is untrue:

(31)   Red is certain of who came to the party.

(32)   Alex is certain of who came to the party.

The reason for this prediction is that (28) is the only answer that Red thinks likely, and Red is certain of that answer, but there are two salient answers that Alex thinks likely ((28) and (29)), and Alex is not certain of all of these answers (in

---

[6]The issue here is that considering p possible is arguably equivalent to not being certain that the p is false, and this possible logical connection between certainty and the consideration of possibility makes it difficult to construct an argument that the proposed semantics is non-reducible.

particular, he is not certain that William came to the party). I don't want to evaluate the empirical merits of this analysis of *'be certain'* here, but only to note that it lacks the reducibility property, and so illustrates how Lahiri's approach of letting each predicate impose lexically specified constraints on its restrictor allows us to derive a question-embedding semantics that lacks the reducibility property. Similar arguments could be made for Lahiri's treatment of *'agree'*, but I will not explore the details here.

Lahiri's approach is in some respects very similar to the approach that I eventually develop to handle some non-reducibility cases, discussed in Section 4.5.3 below. The reader will note that both approaches allow the lexical entry for a responsive predicate to hold a little bit of extra information, above and beyond the propositional predicate that we normally associate with it. However, the details are rather different, especially with respect to *'know'*, where Lahiri restricts quantification to true answers while I, following Spector (2005), make a point of considering untrue answers that are among the subject's beliefs.

4.3 Examples Illustrating Non-Reducibility

In my introduction of reducibility, I defined the reducibility property broadly, so that a responsive predicate is considered reducible if and only if, whenever it relates two entities to the same propositions, it relates them to the same questions as well. My assertion that not all responsive predicates have the reducibility property is founded on a construction of examples where a predicate relates two entities to the same propositions but to different questions. I will sketch this argument for two predicates: *'know'* and *'forget'*.

4.3.1 Non-Reducibility of *'know'*

For *'know'*, the scenario described will be one in which Red and Janna know the same propositions, but (33) is true while (34) is false:

(33)   Janna knows where Rupert can buy a newspaper.

(34)   Red knows where Rupert can buy a newspaper.

The scenario will be designed around the intuition that for (33) to be true, two conditions must be satisfied. First, Janna must know, for some (salient, reasonably priced, reasonably located) business x, that Rupert can buy a newspaper at x. Second, for every y such that Janna believes Rupert can buy a newspaper at y, it must in fact be the case that Rupert can buy a newspaper at y. This dependency not just on knowledge, but on belief, is the key to constructing a counterexample to reducibility: we will give Janna and Red the same propositional knowledge, but different beliefs.

The scenario that provides our counterexample is summarized in (35):

(35)   scenario for *'know'*:

    a.   Red and Janna both know Rupert can buy a newspaper at PaperWorld.

    b.   Neither Red nor Janna has any other knowledge about the availability or non-availability of newspapers.

    c.   On all other topics, Red and Janna know exactly the same propositions.

    d.   Red falsely believes that Rupert can buy a newspaper at Newstopia, and is under the impression that this belief constitutes knowledge.

    e.   Janna is unopinionated about the availability of newspapers at Newstopia.

    f.   Neither Red nor Janna has any beliefs about the availability of newspapers besides those attributed to them above.

    g.   With the exception noted in (35-d) and (35-e), Red and Janna believe exactly the same propositions.

The key facts can be visualized as (36):

Newspaper available at ...	PaperWorld?	Newstopia?
R's beliefs:	Y	Y
J's beliefs:	Y	?
Facts:	Y	N

(36)

My judgement is that under the circumstances described in (35) (and more briefly in (36)), (33) is true while (34) is untrue.

The truth of (33) seems clear enough – Janna knows that Rupert can buy a newspaper at PaperWorld, and this is the only belief she has about the availability of newspapers. This is the simplest case of knowing an answer to a question on a mention-some reading.

(34), on the other hand, doesn't seem to be true.[7] Red's beliefs about where Rupert can buy a newspaper are at odds with the facts, and Red is not a reliable source of information about the availability of newspapers.

In the scenario described in (35), Janna and Red know exactly the same propositions, but they *believe* different propositions. This difference in beliefs is enough to make it so that *'know'* relates Janna, but not Red, to the question *'where can Rupert buy a newspaper?'*. Because we cannot ignore facts about propositional belief when assessing question-knowledge, question-oriented *'know'* is not reducible to propositional *'know'*.

The nature of the counterexample gives us some idea of what the meaning for the question-embedding use of *'know'* should look like. Where P is the set of possible (true or untrue) mention-some answer propositions for (37), a typical mention-some account would give (37) the truth-conditions sketched in (38). On the basis of the observations above, (39) appears closer to the correct truth conditions:

(37)   Where can Rupert buy a newspaper?

(33)   Janna knows where Rupert can buy a newspaper.

(38)   There is $p' \in P$ such that Janna knows p'.

(39)   There is $p' \in P$ such that Janna knows p', and there is no $p'' \in P$ such that p'' is false and Janna believes p''.

Stated in this way, the truth-conditions for the problematic uses of *'know'* give us at least a ray of hope that we may be able to come up with a nontrivial theory of embedding for responsive predicates without the reducibility property. Even if the

---

[7]For the moment, I wish to leave aside the issue of whether it is simply false, or suffers a presupposition failure or some other more exotic kind of untruth.

question-oriented use of *'know'* depends on relations between individuals and propositions other than propositional knowledge, it still depends only on relations between its subject and propositions. This puts it in intuitive contrast with a paradigm rogative predicate like *'ask'*, which doesn't appear to have a natural paraphrase in terms of any combination of propositional relations.

A natural question is whether a comparable non-reducibility effect arises with strongly exhaustive readings. I am not aware of any such effect, and I don't thing that one is likely to be found for *'know'*, but I cannot exclude the possibility that some example of non-reducibility with a strongly exhaustive reading will be found in the future. A straightforward attempt to adapt the argument presented here to the strongly exhaustive case fails, in any case, so the arguments presented *'know'* above, and for *'forget'* below, only show that non-reducibility arises in the mention-some case.

To see why the argument is not readily adapted to the strongly exhaustive case, consider the structure of the argument. We begin with a question φ and an embedding sentence of the form *'Janna knows φ'*. Next, we find some proposition p such that Janna knowing p is enough to make *'Janna knows φ'* true. Finally, we find some other proposition p' that bears on the question, but is independent of p. We show that if p' is false, and Janna believes p', then this makes *'Janna knows φ'* untrue even if Janna knows p. What allows us to construct the counterexample is the fact that we can freely vary whether somebody believes the false answer p' without without affecting which propositions they know. Because of this, we can set things up so that two people (Red and Janna) know the same relevant propositions (that is, they both know p and no other relevant proposition), but only one of them is related to the question by *'know'*, because they have different beliefs regarding the false p'.

With strong exhaustivity, the strategy that I've been using to find a suitable false-but-relevant proposition (p' above) won't work. My strategy in the mention-some case was to just pick another possible answer, which could generally be relied upon to be independent of the other proposition. This won't work in the strongly exhaustive case because any two strongly exhaustive answers are mutually inconsistent. Consider, for example, (40):

(40)   Janna knows who was at the party.

Relative to a particular domain restriction and set of facts, there is a single proposition that Janna *must* know for (40) to be true: the true strongly exhaustive answers to (41):

(41)   Who was at the party?

(41), like any other *'wh'*-question, has only one true strongly exhaustive answer. If, for example, Anne and William were the only ones at the party, it is given by (42):

(42)   Anne and William were the only ones at the party.

If (42) is the only true strongly exhaustive answer, then anybody who is related to (41) by *'know'* has to know (42). Thus, in trying to construct a non-reducibility argument, we'll have to assume that both of our potential knowers (Janna and Red) know (42). The next step in the argument is to pick a proposition that bears on the question and is independent of (42), so that we can say that it is false, and that it is believed by one person but not the other.

This is where things break down. (42) is a strongly exhaustive answer, after all. It provides a maximal, or very nearly maximal, amount of information bearing on the question (41). Finding a suitable proposition that bears on the question, can be false while (42) is true, but that can be consistently believed or not believed independent of (42) is thus rather difficult, and, without such a proposition, the needed scenario cannot be created. In particular, no other strongly exhaustive answer will work (since any strongly exhaustive answer besides (42) will contradict (42)), and no mention-some answer will work, since no mention-some answer is independent of (42) (the various propositions that some combination of Anne and William were among the people at the party will be entailed by (42), and any proposition that identifies anybody else as a person at the party will be inconsistent with (42)). This doesn't rule out the possibility of finding *some* suitable proposition, but it does mean that my strategy for finding the needed proposition to build the argument won't work. The key feature of my argument is that it exploits the incompleteness of mention-

some answers, so, when we move to examples where complete answers are needed, it no longer works.

4.3.2 (Near) Non-Reducibility of *'forget'*

The argument for *'forget'* proceeds along similar lines to the argument for *'know'*. Just as (43-a) requires not just that Janna know one mention-some answer to the embedded question, but also that she not believe any false mention-some answers, so (43-b) requires not just that Janna forgot one answer, but also that she not remember any of the answers she once knew.

(43) a. Janna knows where Rupert can buy a newspaper.
     b. Janna forgot where Rupert could buy a newspaper.

For a formal reason that will be discussed below, this argument may not show strict non-reducibility, but the kind of reduction that it potentially leaves open is one that typical theories of embedding are not well equipped to exploit, and gives us reason to think that a reductive approach here, even if possible, is not a natural or appropriate way to proceed.

As with the discussion of *'know'*, the scenario for *'forget'* provides a situation where Red and Janna have forgotten the same propositions, but (44-b) is untrue while (44-a) is true. The scenario needed is sketched in (45) and summarized in (46):

(44) a. Janna forgot where Rupert could buy a newspaper.
     b. Red forgot where Rupert could buy a newspaper.

(45) Scenario for *'forget'*:
     a. Red and Janna both once knew, and have forgotten, that Rupert can buy a newspaper at PaperWorld.
     b. For the entire time period under consideration, up to the present, Red has known that Rupert can buy a newspaper at Cellulose City.
     c. Janna has always been unopinionated about whether Rupert can buy a newspaper at Cellulose City.

    d.   Neither Red nor Janna has ever had any other knowledge or beliefs about the availability of newspapers.

    e.   With respect to all topics except where Rupert can buy a newspaper, Red and Janna have forgotten exactly the same propositions at exactly the same times.

    f.   With the exception noted in (45-b) and (45-c), Red and Janna have always possessed exactly the same knowledge.

(46)

Newspaper available at ...	PaperWorld	Cellulose City
Red:	forgot	still knows
Janna:	forgot	never knew

As with the discussion of *'know'*, we seem to have a case of non-reducibility. It appears that Janna and Red have forgotten exactly the same propositions. However, they had, and retain, different knowledge – Janna never knew and never forgot about the availability of newspapers at Cellulose City, while Red did know about this, and still does. Neither has forgotten this fact, but the fact that Red persists in knowing it makes us judge (44-b) untrue.

Even if this judgment is right, it is not a rigorous counterexample to reducibility in the same way as the *'know'* example is, because it rests on the assumption that Red knew both (47-a) and (47-b), and forgot (47-a), without knowing and then forgetting any other propositions about the availability of newspapers. This is a problem because, under the circumstances described, it is natural to infer that Red knew and forgot (47-c). If it is impossible to know and then forget a proposition without knowing and then forgetting its conjunction with every other proposition one knows, then technically the set of propositions that Red forgot and the set of propositions that Janna forgot will be different (in particular, Red, but not Janna, will have forgotten the proposition expressed by (47-c)).

(47)   a.   Rupert can buy a newspaper at PaperWorld.

        b.   Rupert can buy a newspaper at Cellulose City.

        c.   Rupert can buy a newspaper at PaperWorld and Rupert can buy a newspaper at Cellulose City.

If we accept that the semantics of *'know'* and *'forget'* constrains things in a way that guarantees that the situation described in (45) can't be true in the form presented, but must be amended to include the fact that Red forgot (47-c), then *'forget'* does appear to still have the reducibility property. We can even state the semantics of question-embedding *'forget'* reductively as follows: Where P is the answer-set of a mention-some question φ, let P' be the closure of P under conjunction (intersection of world-sets). Now let F be the set of propositions forgotten by Red, we say that *'Red forgot φ'* is true iff the following conditions hold: first, $F \cap P' \neq \emptyset$ and, second, for all q, $q' \in P'$, if $q \subseteq q'$, and $q \in F \cap P'$, then $q' \in F \cap P'$. If it is necessarily true that one cannot forget any proposition q'' without also forgetting the conjunction of q'' with every other proposition one knows, then the reductive truth conditions just given will be sufficient to make (44-a) true but (44-b) untrue in the situation described above.

Note, though, that even if we grant this (nontrivial) conjunction closure property for forgetfulness, the semantics for *'forget'* proposed above is completely ad hoc. It doesn't appear to correspond to any standard reductive scheme, and, as far as I know, it has never been proposed as the semantics for embedding questions under *'forget'*. This solution thus represents a serious problem for attempts to keep question embedding relatively uniform across responsive predicates, and goes beyond the kinds of predicate-dependent flexibility (like selection for strongly or weakly exhaustive answers in Guerzoni and Sharvit (2007)) that have generally been advocated. Thus, even if *'forget'* has the reducibility property, the task of turning this reducibility property into a useable reduction will require some serious reevaluation of the way we build our reductive theories (at least if we cannot explain it away).

Although this argument does not provide a case of formal non-reducibility, the most natural formulation of the generalization we observe is non-reductive. If the most straightforward reductive account would analyze (44-a) as (48), it would appear that something like the non-reductive (49) is a better fit.

(48)   There is $p' \in P$ such that Janna forgot p'.

(49)   There is $p' \in P$ such that Janna forgot p', and, for every $p'' \in P$, if p'' Jana knew p'' then Janna forgot p''.

4.3.3 Nonstandard Reducibility of *'ignorare'*

A number of languages have verbs meaning *'not know'*. For example, descendants of the Latin *'ignorare'*, which has this meaning, are found in a number of Romance languages.[8] These embed mention-some questions in a manner inconsistent with the kind of existential-quantification-over-answers model that seems to work in most other cases: the truth-conditions are not that there is a mention-some answer (or a true mention-some answer) to which the subject is related by the *'not know'* predicate, but rather that there is not a mention some-answer that the subject knows. The relevant examples for the Italian *'ignorare'* and the French *'ignore'* can be found in (50) and (51), respectively:

(50)   Gianni ignora   dove si   possono comprare libri in inglese a
           Gianni not-know.3sg where one can.3pl buy.INF   books in English in
           Milano.
           Milan
           Gianni doesn't know where one can buy books in English in Milan.

(51)   Jean ignore   où   l'on peut   acheter des   cigarettes dans
           Jean not-know.3sg where one can.3sg buy.INF INDEF cigarettes in
           cette ville.
           this   city
           Jean doesn't know where one can buy cigarettes in this city.

The judgment reported for (50) is that the sentence can only mean that there does not exist x such that Gianni knows that x is a place in Milan where one can buy books in English. It does not have a reading that corresponds to existential quantification over mention-some answers taking scope of *'not know'*.[9] That is, in a scenario where Gianni knows of many places to buy books in English in Milan, but doesn't know

---

[8]English *'ignore'* is derived Latin *'ignorare'* through French *'ignorer'*, but, at least in modern English, it lacks this *'not know'* sense. English *'be ignorant'* does have the *'not know'* meaning, but can only embed propositional clauses for some speakers. For most English speakers, including those who can embed propositional clauses under *'be ignorant'*, the truth-conditional judgments with embedded mention-some questions appear to be the same as those given for the *'not know'* verbs in (50) and (51).

[9]The interest of this judgment depends on the assumption that the embedded question *'dove si possono comprare libri in inglese a Milano'* has a mention-some reading available. The availability of such a reading is suggested by judgments for the truth-conditions of (52), which, like the English translation I provide, does not require Luca to know an exhaustive list of places in Milan where books in English can be bought.

about some others, (50) is not true on any available reading. The judgment for (51) is analogous.

This judgment does not show that *'ignorare'* and *'ignorer'* lack the reducibility property. There is, in fact, a relatively straightforward reductive account that will handle the facts presented above. The reduction rule will be that *'ignorare'* relates an entity to a question if it relates that entity to *every* true mention-some answer to the question. That is, question-embedding *'ignorare'* would be have as its intension the function given in (53-b), while propositional *'ignorare'* would mean the function in (53-a):

(53) a. $\lambda w \lambda p_{\langle \mathbf{s}, \mathbf{t} \rangle} \lambda x (\neg knows(w)(p)(x))$

b. $\lambda w \lambda P_{\langle \langle \mathbf{s}, \mathbf{t} \rangle, \mathbf{t} \rangle} \lambda x \forall p' ((P(p') \wedge p'(w)) \rightarrow \neg knows(w)(p')(x))$

This semantics for *'ignorare'* makes the right predictions for (50), since it says that (50) is true if and only if Gianni doesn't know even a single true mention-some answer, which is to say if and only if Gianni doesn't know of a single place in Milan where books in English can be bought. This formalization of *'ignorare'* has the reducibility property – if two people are related to the same propositions by (53-a), they must be related to the same proposition-sets by (53-b).

Although the semantics in (53) is adequate to account for these data, and satisfies the reducibility property, it is nevertheless disconcerting, at least if we aspire to a theory of question embedding that handles all responsive predicates in a reasonably uniform way. Whenever we have seen embedding of mention-some questions in the past, we've *existentially* quantified over mention-some answers. For example, (54) can be true even if Rupert didn't find out every place he could buy an Italian newspaper – if he found out about a single place, that is enough information.

(54) Rupert found out where he could buy an Italian newspaper.

---

(52) Luca sa   dove si   possono comprare libri   in inglese a Milano.
      Luca know.3sg where one can.3pl buy.INF   books in English in Milan
      Luca knows where one can buy books in English in Milan.

The existential quantification over these answers is the characteristic that makes embedded mention-*some* readings recognizable: this weakness of only asserting that there is some positive instance of which the propositional predicate holds is the common characteristic of these kinds of examples. The sudden introduction of universal quantification is thus an anomaly.

Note that the *'ignorare'/'ignorer'* examples above appear to have identical truth-conditions to their English translations with *'not know'*. The English sentences are not anomalous because we apply the question-embedding rule to *'know'*, and then negate the result. That is, we scope the negation over the existential quantification over answers. This gives us the same results as in (53-b) because of the standard negation-duality between universal and existential quantification, which introductory logic students learn as the general equivalence of $\neg\exists v\varphi$ and $\forall v\neg\varphi$. One natural way of handling the *'ignorare'/'ignorer'* data, then, is by decomposition of the verb into two separate parts – one equivalent to English *'not'*, and the other equivalent to English *'know'*. The *'know'* part will then embed questions the same way that any other responsive predicate embeds questions (that is, on my scheme, by existential quantification over possible answers), and then the resulting proposition will be negated by the *'not'* part. This solution, discussed at greater length in Section 4.4.2, is a reasonably natural one for *'ignorare'/'ignorer'* (although it is not completely free of complications), so defenders of reductive accounts like the one I have been using probably won't find the *'ignorare'/'ignorer'* data presented here especially troubling. I have included these examples primarily for completeness, but they will not be my main focus in what follows.

4.4   Conservative Replies

If the judgments I describe for the scenarios sketched above are right, then the simple reductive approach won't work, at least for cases of *'know'* and *'forget'* embedding questions that receive mention-some interpretations. If one wishes to avoid abandoning reducibility, there are a number of conservative replies available. One natural reply is to dispute the data. Another approach is to situate a reductive account in some more combinatorially complex semantics or syntax-semantics interface, so that there is still reduction, but the relations to which the reductive embedding rule is

applied are not *'know'* and *'forget'*, as it first appears, but something else instead. Still another reply would be to try to attribute these judgments to a contextually-influenced specificity effect in the choice of a mention-some answer to evaluate. I explore some possible developments of these alternatives below.

4.4.1 Contesting the Data and Pragmatic Explanations

The most direct way to defend reducibility is to deny the truth-conditional judgments reported above. The extreme here is a flat-out denial of the intuitions I report: one could simply insist that (34) is a perfectly fine description of the state of affairs in (36) (or one could insist that (33) is also untrue in this situation), and deny the analogous judgments for the corresponding *'forget'* examples.

(33)   Janna knows where Rupert can buy a newspaper.

(34)   Red knows where Rupert can buy a newspaper.

(36)

Newspaper available at ...	PaperWorld?	Newstopia?
R's beliefs:	Y	Y
J's beliefs:	Y	?
Facts:	Y	N

I don't find this extreme plausible, at least not for my own judgments and the judgments of many informants, but others may find it better.

Of course, this extreme isn't the only alternative, and there are subtler ways to deny my claims about the truth-conditions of (33) and (34). The defender of a reductive account could also place the oddness in the postcompositional pragmatics. The claim would be that, in the situation described in (36), the assertion that Red knows where Rupert can buy a newspaper is technically true, but is an uncooperative thing to say in typical circumstances, and that it is in fact so uncooperative that we have difficulty imagining it ever being thought of as accurate as a description of that scenario.

In a typical situation where the fact expressed by (34) is conversationally relevant, uttering the embedded question sentences under discussion would indeed be

uncooperative, at least for a speaker who was well-informed about the situation. This is because the most easily imagined reason to say that Red knows where somebody can buy a newspaper is to convey that Red is a reliable source of guidance on this point. In (36), Red is not a reliable source of information, since her beliefs include a false answer to the question: for all we know, if Rupert were to ask Red where he could buy an newspaper, she would be as likely as not to give him a false answer, even if she replied without malice. Reporting Red's knowledge without noting her unreliability conjures up an image of Red as a reliable source of information on this subject, and, since she is not, asserting (34) in this situation is rather misleading. Likewise, pointing out that Red has forgotten where we can buy a newspaper usually serves to communicate that Red is no longer a useful information source, so asserting such forgetfulness in a situation where she is still able to direct us to a seller would be misleading, even if it were technically true.

It is difficult to conclusively refute this line of argument, especially in the vague terms in which I have sketched it above, but, to the extent that I have intuitions about such things, for me the key sentences in the scenarios seem to go well beyond being simply misleading – in my judgment, though perhaps not in those of all speakers, they do not even seem to be true 'on a technicality'. Further, if we put the entire situation in the past, and substitute (55) for (33) and (56) for (34), and if we further imagine that neither Janna nor Red was ever accessible to Rupert as an information source, I think that the judgments do not change much.

(55)   Janna knew where Rupert could buy a newspaper.

(56)   Red knew where Rupert could buy a newspaper.

Another intuition that goes against the pragmatic approach is that other sources of unreliability in combination with knowledge do not yield the same kinds of intuitions. (33) is true in the scenario sketched in (36), and it remains an intuitively accurate description of events even if Janna is otherwise unreliable. If, for example, Janna wishes to maliciously waste Rupert's time, and so is inclined to lie, she will cease to be a reliable information source for Rupert, but (33) will still sound like a much more accurate description of the situation than (34). Similarly, if we assume
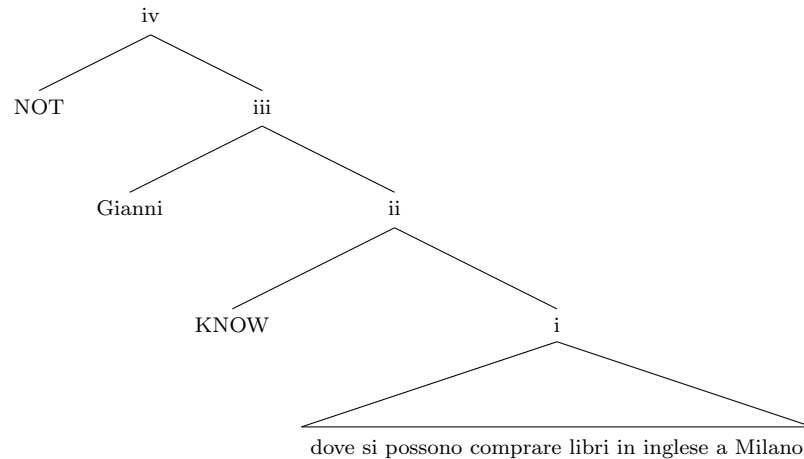
Janna suffers from a neurological condition that causes her to randomly substitute one proper name for another, or to randomly insert and remove negation in her sentences, she will be quite unreliable, but (33) remains perfectly fine. This all suggest that, if there is a pragmatic inference of reliability associated with (33) and (34), that inference is a much weaker effect than the effect responsible for the judgment that (34) is inaccurate in (36). This observation doesn't rule out every pragmatic account along these lines, but it suggests that the advocate of such an account would probably have to appeal to some notion that is subtler than the 'reliable information source' inference suggested above.

The above is not by any means an exhaustive survey or systematic refutation of all possible pragmatic replies, but I think it does indicate some of the more important issues that a pragmatic reply would have to overcome.

4.4.2 Decomposition

Another possible reply concedes non-reducibility, but avoids any special complications with the embedding rule by decomposing the problematic predicates into multiple layers, and claiming that a reductive embedding rule applies to the propositional relation at the lowest layer. This approach is most natural for the *'ignorare'* verbs: the effect we encountered with these was, in essence, that the *'not'* part of the meaning seems to outscope the existential quantification over mention-some answers, which in turn outscopes the *'know'* part. We can simply turn this observation into our theory of these verbs, decomposing *'ignorare'* into separate, grammatically recognized, *'not'* and *'know'* parts, as schematized in (57) as an analysis of (50):

(57)

```
                            iv
                          /    \
                       NOT      iii
                              /     \
                          Gianni     ii
                                   /    \
                               KNOW      i
                                        /|\
                                       / | \
                                      /  |  \
                        dove si possono comprare libri in inglese a Milano
```

(50) Gianni ignora   dove si   possono comprare libri   in inglese a
      Gianni not-know.3sg where one can.3pl buy.INF   books in English in
      Milano.
      Milan
      Gianni doesn't know where one can buy books in English in Milan.

We can then evaluate node (iii) as we would evaluate (58), so the intension of node
(iii) will, on our reductive account, be the proposition that Gianni knows of at least
one place in Milan where books in English can be bought.[10]

(58)   Gianni knows where one can buy books in English in Milan.

The truth-conditions computed for node (iv) will then be the negation of the truth-
conditions of (58) – that is, we will predict that (50) is true if and only if Gianni
doesn't know any mention-some answer to the embedded question. The judgments
for (50) are compatible with these truth-conditions.

For *'ignorare'*, this approach is relatively plausible. Latin *'ignorare'* was formed
with the common negative prefix *'in-'*, which is also a prefix in French and Italian
(although it is not clear whether it is readily identifiable as a part of the modern
'ignorare' and *'ignorer'*), so we should not be completely surprised if speakers of
these languages recognize a distinct negation in the shape of their *'ignorare'* verbs.

---

[10]Of course, these reductively derived truth-conditions for (58) make use of a reductive account
of *'know'*, which is probably not exactly right, but let's ignore this for the moment while we focus
on the distinctive difficulties presented by *'not know'* verbs.

Even if they do not recognize a distinct negation in the morphology, they might still adopt a decomposition at some suitable level of semantic analysis. Furthermore, separating out negation in negative words is a widely argued-for form of lexical decomposition (cf. Jacobs (1980) and Potts (2000)). If, as seems to be the case, available judgments indicate that embedding questions under *'ignorare'* produces truth-conditions identical to embedding them under *'not know'*, then this looks like a pretty reasonable response.[11]

For *'forget'*, the decompositions *'cease to know'* and *'not remember'* look promising. For *'know'*, however, there isn't an obvious decomposition, although, if we try to decompose *'know'* into something like *'can say'* or *'can tell you'*, as with the idiomatic use of (63-b) to indicate (63-a), we perhaps get a hint of a possible line of attack.

---

[11]This response does have some difficulties to overcome, related to differences in grammaticality facts between Italian *'ignorare'* and *'non sapere'* (which renders negation and knowledge with separate words). Notably, *'non sapere'* allows certain kinds of modification that *'ignorare'* does not:

(59) Paolo non sapeva   affatto che   cosa fosse successo   quella notte
      Paolo not knew.PAST at-all   what thing was   happened that   night
      Paolo didn't know at all what happened that night.

(60) *Paolo ignorava   affatto che   cosa fosse successo   quella notte
      Paolo ignorare.PAST at-all   what thing was   happened that   night

This suggests that *'ignorare'* or *'non sapere'* differ in which syntactic attachment points they provide, or in their licensing of negative polarity items, or both. If the difference relates to attachment possibilities, then the decomposition theorist ought to have something to say about what the structural difference between the two is. I suspect that a satisfying account of this would be possible, but it's worth flagging as something that would need to be worked out. If *'ignorare'* and *'non sapere'* differ in their NPI-licensing behavior, this will cast doubt on any decomposition of *'ignorare'* that involves a negation.

In order to test for differences in NPI-licensing behavior, independent of local syntactic properities, it would be good to see if NPIs in an embedded clause behave differently with respect to *'non sapere'* and *'ignorare'*. Unfortunately, NPIs in the embedded clause appear not to be very good with either, making it hard to get a distinction. It is worth noting, though, that speakers report that (61), although questionable, is noticeably better than (62), indicating that the NPI *'mai'* may treat *'ignorare'* and *'non sapere'* differently:

(61) ?Luca non sapeva   che Andrea fosse   mai stato a Los Angeles.
       Luca not knew.PAST that Andrea were.SUBJ ever been to Los Angeles

(62) *?Luca ignorava   che Andrea fosse   mai stato a Los Angeles.
       Luca ignorare.PAST that Andrea were.SUBJ ever been to Los Angeles

(63)   a.   I don't know what happened next.

           b.   I can't say what happened next.

As a matter of formal possibility, it seems to me quite likely some kind of decomposition account can be made to work for all the problematic examples discussed above. General decomposition is an extremely powerful tool. If we can pick some formal predicate of propositions to apply our general mention-some embedding procedure to, and then combine that after the fact with arbitrary other semantic operators, we have a great deal of flexibility. For this reason, I think we should be wary of these decompositional accounts – if we are forced to entertain every possible decomposition, then even a strict uniform theory of embedding will make only very weak predictions. Of course, there are many cases where the kind of decomposition might have some plausible independent motivation (and, in particular, it has some appeal for *'ignorare'* and *'forget'*), but it is still worth seeing if we really need such decomposition, especially for predicates like *'know'* where it is harder to independently motivate the deocmposition that we would need. If one were to pursue a general decompositional approach, it would be desirable to put some constrains of the kinds of decompositions the grammar considers.

4.4.3   Specific Answers

The kinds of judgments that I have been cultivating to show to argue for non-reducibility can be strengthened or weakened by making a particular mention-some answer more situationally salient.[12] If Janna actually did furnish Rupert with a particular answer, for example, then the truth of (64) seems to be naturally evaluated in terms of the truth of that answer, without the need to inspect all the answers among Janna's beliefs. So in (65), we conclude that (64) is untrue based on Janna's believing and offering a single false answer, and in (66) we conclude that (64) is true base on Janna's believing and offering a single true answer.

(64)   Janna knows where Rupert can buy an Italian newspaper.

---

[12]I am indebted to Sam Cumming for raising the issue of specificity, and for suggesting examples (66) and (67).

(65)   I thought Janna knew where Rupert could buy an Italian newspaper, but he says he went where she told him to, and they didn't have any (and I know Janna wouldn't lie to Rupert deliberately), so it turns out I was wrong, and Janna didn't know.

(66)   I thought Janna knew where Rupert could buy an Italian newspaper, and he says he went where she told him to and they sold him one, so I was right and Janna did know.

The problematic judgment, for our purposes, is that the truth of (66) does not seem to be called into question by the possibility that, besides the true answer that she gave Rupert, Janna may have had false beliefs about the availability of Italian newspapers.

To pick a more extreme example, suppose that I have just lost a bike race against Janna, and lost rather badly. In this case, (67) is not really a good reaction – Janna's false beliefs about how to ride a bike, no matter how extreme, simply do not make it any less true that she knows how to ride a bike, when we are speaking in a context where Janna has just demonstrated her bike-riding proficiency.[13]

(67)   Well, at least I know how to ride a bike – Janna thinks that it's possible to ride a bike is by pushing on the pedals telekinetically.

This suggests a possible reply to claims of non-reducibility: perhaps context makes a certain answering proposition salient, and, to evaluate the truth of (64), we simply ask if that answer was known by Janna.

The judgment for (66) needs to be accounted for, but I don't think that the reply to non-reducibility built around it is very convincing. The key issue is that we generally can't evaluate Janna's knowledge of just any salient proposition: the salient proposition must be salient among Janna's beliefs.

---

[13]It is hard to say exactly what this example tells us without having a precise characterization of the answerhood conditions of embedded infinitive questions, but *'it's possible to ride a bike is by pushing on the pedals telekinetically'* does seem to be an answer (or close enough to an answer) for *'how to ride a bike'*, so this example is at least cause for concern.

Thus, to revisit a theme from 4.4.1, if there is a salient answering proposition that is not among Janna's beliefs, but is not know buy Janna (either because it is false or for some other reason), that is not enough to make (64) untrue. In (65), the inference that (64) is untrue is judged acceptable because the story described leads us to assume that Janna sincerely believed the answer she gave Rupert. If this belief is removed, then the inference that (64) was untrue in the situation at hand is undermined, as in (68)

(68)   I thought Janna knew where Rupert could buy an Italian newspaper, but he says he went where she told him to, and they didn't have any. I asked around about Janna, and it turns out she has a habit of maliciously giving inaccurate advice. #So it turns out I was wrong, and Janna didn't know.

The point here is that the answer furnished by Janna is presumably salient in both (65) and (68), and in neither case is it know by Janna (since it's false), but only (65), and not (68), supports the inference that (64) was false. Salience of a specific answer alone cannot account for this. The grammar needs to distinguish between propositions believed by Janna (which are, in some sense, candidates for being known by Janna), and propositions not believed by Janna (which are not). Thus, non-reducibility endures: even controlling for salience effects, Janna's failing to know exactly the same answer may or may not suffice to make (64) untrue. Further, the fact that belief is the requirement here is specific to 'know' – it would not be the right requirement for 'forget' or for 'tell', indicating that this kind of effect can be special to different responsive predicates.

Having argued that specificity effects are not the source of the non-reducibility phenomena under consideration, I will not pursue the precise analysis of these effects here. One natural kind analysis is that the quantification over answers involved in embedding (including that involved in the checking for false beliefs) is naturally restricted to the most salient answers, but many details would need be worked out. Some possibly related issues are briefly and inconclusively discussed in Section 6.2.

4.5 Radical Replies

The replies discussed in the previous section were conservative in the sense that they try to keep a familiar architecture for responsive predicates embedding mention-some questions: the predication is true of the question iff it is true of some mention-some answer. The first response I discussed maintained this in its simplest form, and tried to explain away all contrary intuitions as the results of pragmatic effects, while the second accepted this but added layers of structure to suggest that we have not been assessing this rule with respect to the responsive predicates that are really in play, and the third tried to solve the problem to allowed specificity effects to intrude on the choice of which answers to check.

I now want to turn to more radical approaches – approaches that concede that the reductive picture that we've been assuming so far is incapable of dealing with the data discussed above, and try to replace it with something that makes the question-oriented behavior of a responsive attitude depend on something beyond its propositional behavior.

4.5.1 The Nuclear Option

The most extreme rejection of the reductive picture is a theory that treats the question-oriented and proposition-oriented uses of each problematic responsive predicate as completely distinct. On such an account, the lexicon would simply contain two complete entries for the semantics of *'know'*: one for propositional uses, and one for question-embedding. Nothing in the synchronic grammar would enforce any link between these.

This approach may seem like a straw man, but we must concede that it has the great advantage of being undeniably capable of getting the truth conditions right for all our tricky examples. Furthermore, we have already seen at least one sign (*'wonder'*) that seems to be best analyzed as expressing an exclusively question-oriented predicate while (in an archaic usage) expressing a distinct propositional predicate, where it seems that the two predicates must simply be learned separately without any productive synchronic link, so this kind of homophony or polysemy is not wholly implausible. We might say that *'wonder'* is not the exception, but the norm, or, more cautiously, we might say that the normal reductive pattern holds for

most responsive predicates, but that *'know'* is not truly a responsive predicate, but instead an impostor similar to *'wonder'*.[14]

It is, of course, extremely difficulty to falsify this kind of account. Given suitable lexical entries for its two *'know'*s, this reply can mirror the results of any more constrained account. But it seems fair to say that we should regard this solution as a last resort. It would say that there is no connection between the two *'know'*s, or, at least, it would leave that connection out of our theory of question embedding completely. Taken generally, this kind of solution represents an abandonment of any kind of nontrivial theory of question-embedding under responsive predicates (at least for the predicates to which we apply it). We should, of course, remain open to the possibility that this is the best we can do, but we should also try to see if we can do better.

4.5.2   Inverted Reduction

If reducing the question-oriented use of a predicate to its propositional use won't work, one natural reaction is to try to run the reduction the other way. That is, we could say that the lexicon provides for each responsive predicate a question-oriented semantics, and some general rule derives the proposition-oriented semantics from this.

I will attempt to sketch an approach that does this below. I believe that the approach that I sketch is adequate to account for the data that we've seen, but that it is insufficiently constraining. I cannot claim to have exhaustively explored the full space of theories that could be said to employ some kind of inverted reduction, however, so I think we should remain open to the possibility that some approach along these lines may do significantly better than the one that I develop below.

For purposes of this account, I will continue to associate questions with the type

---

[14]If the judgments for (69) and (70) reported in Chapter 1 are right, we will probably want to attribute them to *some* relevant difference between *'know'* and *'wonder'*, so a bit more will need to be said.

(69)   *Anne wonders that William killed Holden, and what he did with the body.

(70)   Anne knows that William killed Holden, and what he did with the body.

$\langle\langle \mathtt{s}, \mathtt{t}\rangle, \mathtt{t}\rangle$. That is, I assume each question contributes a proposition-set identifying the set of all its possible answers. Thus, a responsive predicate like *'know'* will contribute a relation between entities and sets of propositions. To apply such a predicate to a proposition, we will simply package that proposition in a singleton set (that is, we apply the operator $\lambda p\lambda p'(p = p')$ to the proposition[15]), and then feed in the resulting proposition-set as an argument.[16]

For *'know'*, we can then get by with just the question-oriented semantics in (71):

(71) $know_{QUES} = \lambda w\lambda P\lambda x \left(\begin{array}{c} \forall p'((P(p') \wedge believes(w)(p')(x)) \rightarrow p'(w)) \\ \wedge \exists p''(P(p'') \wedge knows(w)(p'')(x)) \end{array} \right)$

That is, *'know'* relates an entity x to a proposition set P if and only if at least one proposition in P is part of x's knowledge, and every proposition in P that x believes is true.

When P is a mention-some answer set, this handles our non-reducibility data and gives us plausible truth conditions. Here *'know'* relates x to a mention-some question iff x knows one mention-some answer (the $\exists p''$... part of (71)) and x doesn't falsely believe any other mention-some answers (the $\forall p'$... part of (71)). That is, for (34) to be true, Red must know at least one mention-some answer (that is, at least one proposition with the shape exemplified by (72-a) and (72-b)), and every mention-some answer that Red believes must be true.

(34)   Red knows where Rupert can buy a newspaper.

(72)   a.   Rupert can buy a newspaper at PaperWorld.

---

[15]It may be of interest that the $Q_{()}$, the zero-ary variant of the Q operator that is discussed in Chapter 5, is equivalent to $\lambda p\lambda p'(p = p')$. This means that, on the inverted reduction account, we can force propositional clauses to embed as proposition sets by saying that an appropriately-typed Q is present in *all* embedded clauses. If we adopt this approach, we should probably rename Q, since this formal move would mean that Q was no longer associated specifically with questionhood.

[16]A formally similar approach, in which *'know'* selects exclusively for sets of propositions, and interacts with individual propositions only by an operator that puts them in singleton sets, is used by Uegaki (2011) as part of an analysis of some very different data, related to the fact that only an acquaintance-like reading is available for *'know the rumor that ...'*, while a reading that treats the rumor as a proposition is available for *'believe the rumor that...'* (related issues are discussed, within a very different theoretical framework, by Ginzburg (1995a,b)). A discussion and evaluation of Uegaki's work on this subject would take us too far afield, but it is interesting that the inverted reduction account, or something like it, has been advocated for other reasons.

b.   Rupert can buy a newspaper at Newstopia.

The requirement of knowing one mention-some answer is just the picture of embedded mention-some questions from Chapter 2 (when our uniform embedding rule was existential quantification over possible answers). What is new is requirement that every mention-some answer that the subject believes must be true. This requirement is neatly tailored to handle the data that led me to conclude that *'know'* was non-reducible. Recall that the problematic scenario was the one given in (36):

(36)

Newspaper available at ...	PaperWorld?	Newstopia?
R's beliefs:	Y	Y
J's beliefs:	Y	?
Facts:	Y	N

On the old account, with only the existential half of the story, we predicted that, in this scenario, both (34) and (33) should be true, since both Red and Janna knew themention-some answer (72-a). This prediction went against the judgment that the situation in (36) made (33) true and (34) untrue.

(33)   Janna knows where Rupert can buy a newspaper.

The revised question-embedding semantics for *'know'*, given by (71), solve this problem. Under the circumstances in (36), (33) is still true, because Janna knows the true answer (72-a), and also doesn't believe any untrue answers. Meanwhile, (34) is untrue under our new semantics for question-embedding *'know'*, since for Red believes the untrue mention-some answer (72-b), and, under universal clause of $know_{QUES}$, all the mention-some answers that Red believes must be true if (34) is to be true. Thus, the $know_{QUES}$ of (71) handles the problematic data.

When P is a strongly exhaustive answer set, we again derive a reasonable prediction. The $\exists p''$ part of (71) requires that the subject must know at least one strongly exhaustive answer (and, since *'know'* is factive, that answer must be the true one). This part gives us the standard truth-conditions for question-embedding under *'know'* with a strongly exhaustive reading. In the $\forall p'$... part, we require that every strongly exhaustive answer believed by the subject be true. Since knowledge of

one strongly exhaustive answer is already established in the other part, all this adds is that the subject can't believe any other, false, strongly exhaustive answer (which will, of course, be inconsistent with the true one). This prohibition on a certain narrow class of extremely inconsistent belief states is consistent with all my intuitions for *'know'*, and doesn't add much of interest to the observable truth-conditions.

Finally, let us turn to the case where we have an embedded propositional clause, rather than an embedded question. In this case, P is a singleton set, so the existential part of the above requires that the subject know the only proposition in the set, and the universal part (redundantly) requires that the proposition be true if the subject believes it. Since the universal part is redundant, this gives us the ordinary propositional *'know'*: the reduction has been successful in the sense that our procedure for applying a predicate of questions to a proposition has given us the propositional behavior we observed to begin with.

A similar approach can work for *'forget'* and *'ignorare'*, using either the decompositions discussed above or the approach discussed below in Section 4.5.3. I have no doubt that this approach can address all known cases of non-reducibility. My concern is that, as stated, it is so powerful that it is not clear that it imposes any real constraint on the relation between question-oriented and proposition-oriented uses of a responsive predicate. If the inverted reduction approach can give one predicate any combination of question-oriented and propositional behavior, without any requirement for a relationship between the two, then it is no improvement over the nuclear option discussed above.

The reason that the inverted reduction approach, as formulated here, is probably no more constraining than the nuclear option is that proposition-sets derived from propositions apparently have a mathematical property that easily distinguishes them from those that are naturally derived as the meanings of questions: the former are always singleton sets, while it appears that the latter never are.[17] This means that the

---

[17]Formal exceptions to this generalization may exist. Jessica Rett suggests (73) as a possible cause for concern:

(73)   Which Tigger bounces?

Assume that *'Tigger'* is interpreted 'de re' with respect to the Q operator, and (following Milne (1928)) that there is exactly one Tigger. In this case, the mention-some answer set of (73) will be as in (74-a), and the strongly exhaustive answer set will be as in (74-b), where $Tigger(w^*)$ denotes the set of Tiggers in the actual world, and so is rigid with respect to the other intensional operations at

distinction between question-derived sets and proposition-derived sets is something

work:

(74)   a. $\lambda p_{\langle \mathsf{s,t} \rangle} \exists \beta_{\mathsf{e}}(p = \lambda w'_{\mathsf{s}}(Tigger(w^*)(\beta) \wedge bounces(w')(\beta)))$
        b. $\lambda p_{\langle \mathsf{s,t} \rangle} \exists \beta_{\langle \mathsf{e,t} \rangle}(p = \lambda w'_{\mathsf{s}}(\lambda y(Tigger(w^*)(y) \wedge bounces(w')(y)) = \beta))$

Neither (74-a) nor (74-b) is a singleton set, but both are alarmingly close. Call the actual Tigger t. (74-a) contains exactly two propositions: the proposition that the actual t is a member of the set $\{t\}$ and bounces (i.e., the proposition that t bounces), and the unique contradictory proposition (since choosing any $x \neq t$ as a value for β will get us a proposition that can't be true, since its truth-conditions will include a requirement that $x \in \{t\}$).

Meanwhile, (74-b) will contain only three propositions: the proposition that t doesn't bounce (that is, the proposition $\emptyset = \{x \in \{t\} \mid x \text{ bounces}\}$), the proposition that t bounces (that is, the proposition $\{t\} = \{x \in \{t\} \mid x \text{ bounces}\}$), and the contradictory proposition. ((73) might turn out to contain only two propositions: If, as has sometimes been suggested, the question (73) carries a presupposition that at least one Tigger bounces, then the proposition $\emptyset = \{x \in \{t\} \mid x \text{ bounces}\}$ will be excluded from the strongly exhaustive answer set, leaving only the proposition that t bounces and the contradictory proposition.)

Since both (74-a) and (74-b) contain at least two propositions, neither is a singleton set, so the generalization that singleton sets never arise from questions is not threatened. If we were to decide to amend our definition of answer sets to exclude contradictory propositions, then one or both of these answer sets *would* contain only a single proposition, violating my generalization. Since I think the inclusion of contradictory propositions in the answer set is basically harmless, however, I am not too troubled by this case.

There is a more troubling case, where we really do seem to have a singleton answer set. Consider the case of mention-some answers for an inherently contradictory question, like (75):

(75)   What married bachelor can help?

The only mention-some answer to (75) is the contradictory proposition, so, on a mention-some reading, the answer set of (75) will contain only a single proposition. This will mean that the mention-some answer set of (75) will be a singleton set. This means that we would derive the minimal constraint that, for example, (76-a) and (76-b) must have the same truth conditions on an inverted reduction account:

(76)   a.   Rupert is certain of which married bachelor can help.
        b.   Rupert is certain that Alex is the tallest student and Jonathan is the shortest student and Alex is strictly shorter than Jonathan.

I am not too worried about the case of inherently contradictory questions for two reasons – the first is that, even if we cannot differentiate a contradictory question from a contradictory proposition, the inability of predicates to distinguish between proposition sets in this case will be a minor and somewhat marginal point of overlap, so the point about the inverted reduction approach being insufficiently constraining will still hold. The second reason that I am not concerned is that the inference that (75) has only one mention-some answer rests on our association of propositions with world-sets. This has been adequate for my purposes in this dissertation, but it is almost certainly wrong: (77) and (78) are both contradictions, but it is not clear that they express the same proposition:

(77)   Alex is a married bachelor who can help.

(78)   William is a married bachelor who can help.

that a predicate of sets of propositions can be made sensitive to. Given an arbitrary two predicates R and R', of types $\langle \mathsf{s}, \langle \langle \langle \mathsf{s}, \mathsf{t} \rangle, \mathsf{t} \rangle, \langle \mathsf{e}, \mathsf{t} \rangle \rangle \rangle$ and $\langle \mathsf{s}, \langle \langle \mathsf{s}, \mathsf{t} \rangle, \langle \mathsf{e}, \mathsf{t} \rangle \rangle \rangle$, respectively, we can define a new predicate (given by (79), where $singleton(P)$ is shorthand for $\exists p' \forall p''(P(p'') \equiv (p'' = p')))$ that, on the inverted reduction account sketched above, will handle embedded propositions like R' and embedded questions like R.

(79) $\quad \lambda w \lambda P \lambda x \begin{pmatrix} singleton(P) \wedge \exists p'(P(p') \wedge R'(w)(p')(x)) \\ \vee \\ \neg singleton(P) \wedge R(w)(P)(x) \end{pmatrix}$

What this says is that every combination of question-oriented and propositional behavior that can be derived by the nuclear option can also be derived by this implementation of embedded reduction. Suppose we start out with a nuclear option analysis that says that it embeds questions as the relation R and embeds propositions as the relation R'. Given these R and R', (79) describes a single predicate of proposition-sets that, under the inverted reduction account, will behave exactly like R with respect to embedded questions (which always contribute answer sets with more than one element) but will behave exactly like R' with respect to embedded propositional clauses (which always contribute singleton sets). That is, there is *nothing* that the nuclear option can do that this implementation of the inverted reduction account can't do, which means that, in this implementation, inverted reduction provides no improvement over the nuclear option in the search for a constraining theory.

This is, of course, only a problem with the particular account sketched above. If we were to identify question-oriented predicate meanings with something more nuanced than arbitrary predicates over sets of propositions, we might be able to do better, and, likewise, a reduction other than the one that packages each proposition as a kind of trivial question might give us something more interesting. I think that some more constrained version of an inverted reductive account is worth pursuing, but, except for a few remarks in Section 4.5.3.6, I have no further thoughts on how we might pursue it, so I will put it aside and move on to another approach.

If (77) and (78) express different propositions, those propositions will both be mention-some answers to (75), so the answer set of (75) will contain more than one (contradictory) proposition.

4.5.3 A Selectively Enriched Lexicon

4.5.3.1 General Remarks

I have said that the option of treating the question-oriented and propositional uses of a given responsive predicate as independent should be seen as a last resort. In the preceding chapters, I assumed that the question-oriented use was to be reduced to the propositional use, but in this chapter we have found reasons to doubt that this is always possible. We have briefly looked at the alternative of reducing the propositional use to the question-oriented use, but it is not clear that this can be made significantly more constraining. That is, we have explored both alternatives of deriving one use from the other use, and we have explored the possibility that both uses are unlinked. One natural option that we have not yet explored is the option of deriving both uses from some lexical content that is not directly equivalent to either use. If this lexical content is suitably constrained, and if the derivation of both uses from the lexical content is sufficiently uniform, this kind of approach has the potential to constrain the link between the question-oriented and propositional uses of a predicate, without reducing one to the other. There are many kinds of lexical contents we could employ for this purpose. Below, I sketch one possibility.

4.5.3.2 Introduction to the Twin Relations Theory

Recall that, in the reduction from the propositional case to the question-oriented case above, the entries I proposed for *'know'* and *'forget'* could both be framed as a conjunction of an existential quantification over propositions and a universal quantification over propositions. (For *'know'*, for example, I conjoined a requirement that there be at least one answer known by the subject with a requirement that every answer believed by the subject had to be true.) My proposal for our lexical entries is that we write this format into the grammar. That is, as the meaning of each responsive predicate, the lexicon will record an ordered pair of two relations between propositions and individuals: one for the existential half, and one for the universal half. For a predicate written R I will write these halves $R_\exists$ and $R_\forall$. For most of our responsive predicates, each of these will have type $\langle \mathbf{s}, \langle \langle \mathbf{s}, \mathbf{t} \rangle, \langle \mathbf{e}, \mathbf{t} \rangle \rangle \rangle$. From these, the grammar will have standard rules for deriving the question-oriented and propositional uses, written R_{QUES} and R_{PROP}, respectively.

To see how this works, let's revisit the entry I proposed earlier for question-oriented *'know'*. A successful theory should derive this for $know_{QUES}$:

(71) $\quad know_{QUES} = \lambda w \lambda P \lambda x \left(\begin{array}{c} \forall p'((P(p') \wedge believes(w)(p')(x)) \to p'(w)) \\ \wedge \exists p''(P(p'') \wedge knows(w)(p'')(x)) \end{array} \right)$

Now, to derive the lexicons $know_\exists$ and $know_\forall$, I propose that we isolate the components of each quantification that do not depend on the answer set P. In short, I propose the following:

(80)   a. $know_\exists = \lambda w \lambda p \lambda x (knows(w)(p)(x))$

        b. $know_\forall = \lambda w \lambda p \lambda x (believes(w)(p)(x) \to p(w))$

The scheme for deriving R_{QUES} from $R_\exists$ and $R_\forall$ will essentially be the skeleton of $know_{QUES}$ that's left over when we remove the $know_\exists$ and $know_\forall$ components. So, in general R_{QUES} will be the following:

(81) $\quad R_{QUES} = \lambda w \lambda P_{\langle\langle \mathsf{s,t}\rangle,\mathsf{t}\rangle} \lambda x_\mathsf{e} \left(\begin{array}{c} \forall p'_{\langle \mathsf{s,t}\rangle}(P(p') \to R_\forall(w)(p')(x)) \\ \wedge \exists p''_{\langle \mathsf{s,t}\rangle}(P(p'') \wedge R_\exists(w)(p'')(x)) \end{array} \right)$

Inserting $know_\exists$ and $know_\forall$ into this framework gives us something equivalent to the $know_{QUES}$ we started with.

Given what we know so far, the derivation of R_{PROP} is underspecified, but I propose conjunction as a natural approach.

(82) $\quad R_{PROP} = \lambda w \lambda p \lambda x (R_\exists(w)(p)(x) \wedge R_\forall(w)(p)(x))$

This means that we have:

(83) $\quad know_{PROP} = \lambda w \lambda p \lambda x (knows(w)(p)(x) \wedge (believes(w)(p)(x) \to p(w)))$

Since knowledge entails both belief and truth, this is equivalent to (84), which is of course the ordinary propositional *'know'*:

(84) $know_{PROP} = \lambda w \lambda p \lambda x (knows(w)(p)(x))$

In light of the observation in (84), it is tempting to adopt a model where $R_{PROP} = R_\exists$. This would work fine for the case at hand, and is a viable option. I favor the approach in (82), which conjoins in $R_\forall$, because it is more constraining. The rule in (82) imposes the constraint that R_{PROP} must entail $R_\forall$. The approach in (82) eliminates the possibility of an embedder placing a requirement on all answers in the question-embedding case, without imposing that requirement at all in the propositional case. If we only required that $R_{PROP} = R_\exists$, the grammar would not impose such a constraint.

Concretely, consider the hypothetical predicate *'uncertimes'*. The proposed semantics for this predicate is that $uncertimes_{PROP}$ relates an entity a to a proposition p iff it is not the case that a is certain that p, and $uncertimes_{QUES}$ relates an entity a to a proposition set P iff there is $p' \in P$ such that it is not the case that a is certain that p', and, for every $p'' \in P$, p'' has never been asserted on the front page of the *Los Angeles Times*.

We will see (in result (103) below) that *'uncertimes'* is excluded by the twin relations theory as implemented in (81) and (82), but if we replaced the rule in (82) with the rule $R_{PROP} = R_\exists$, then *'uncertimes'* would be allowed: all we would have to do would be to say that $uncertimes_\exists = \lambda w \lambda p \lambda x (\neg certain(w)(p)(x)$ and $uncertimes_\forall(w)(p)(x)$ was true iff p had never appeared on the front page of the *Los Angeles Times* in w.

To summarize, for any responsive predicate R, the propositional R_{PROP} and and the question-oriented R_{QUES} are derived from the two lexically specified relations $R_\exists$ and $R_\forall$ according to the uniform rules described in (85). For the example of *'know'*, (86) illustrates my proposed choice of $know_\exists$ an $know_\forall$ and the $know_{PROP}$ and $know_{QUES}$ that they derive.

(85)   a. $R_{PROP} = \lambda w \lambda p \lambda x (R_\exists(w)(p)(x) \wedge R_\forall(w)(p)(x))$

       b. $R_{QUES} = \lambda w \lambda P_{\langle\langle s,t\rangle,t\rangle} \lambda x_e \left(\begin{array}{l} \forall p'_{\langle s,t\rangle}(P(p') \rightarrow R_\forall(w)(p')(x)) \\ \wedge \exists p''_{\langle s,t\rangle}(P(p'') \wedge R_\exists(w)(p'')(x)) \end{array} \right)$

(86)   a. $know_\exists = \lambda w \lambda p \lambda x (knows(w)(p)(x)))$
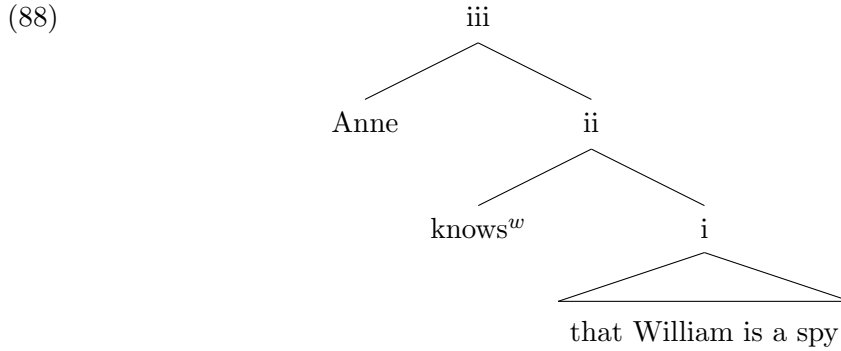
b. $know_\forall = \lambda w \lambda p \lambda x (believes(w)(p)(x) \rightarrow p(w))$

c. $know_{PROP} = \lambda w \lambda p \lambda \left(\begin{array}{c} knows(w)(p)(x) \wedge \\ (believes(w)(p)(x) \rightarrow p(w)) \end{array} \right)$

d. $know_{QUES} =$

$$\lambda w \lambda P_{\langle\langle s,t\rangle,t\rangle} \lambda x_e \left(\begin{array}{c} \forall p'(P(p') \rightarrow (believes(w)(p')(x) \rightarrow p'(w))) \\ \wedge \exists p''(P(p'') \wedge knows(w)(p'')(x)) \end{array} \right)$$

4.5.3.3   Examples of $know_{PROP}$ and $know_{QUES}$ in Action

I'll consider some more general issues for the twin relations account in a moment, but first I want to quickly illustrate the behavior of $know_{PROP}$ and $know_{QUES}$, as worked out above, in some embedding sentences. I will illustrate one example for an embedded *'that'* clause, one for an embedded strongly exhaustive question, and one for an embedded mention-some question.

Starting with an embedded *'that'* clause, consider sentence (87), analyzed with (88) and (89).

(87)   Anne knows that William is a spy.

(88)

(89)   i : $spy(w)(W)$

knows : $know_{PROP}(w^*)$

       $= \lambda p \lambda x (knows(w^*)(p)(x) \wedge (believes(w^*)(p)(x) \rightarrow p(w^*)))$

   ii : $\lambda x \left(\begin{array}{c} knows(w^*)(\lambda w(spy(w)(W)))(x) \wedge \\ (believes(w^*)(\lambda w(spy(w)(William)))(x) \rightarrow spy(w^*)(W)) \end{array} \right)$

Anne : A

$$\text{iii} : \left(\begin{array}{c} knows(w^*)(\lambda w(spy(w)(W)))(A) \wedge \\ believes(w^*)(\lambda w(spy(w)(W)))(A) \rightarrow spy(w^*)(W)) \\ = knows(w^*)(\lambda w(spy(w)(W)))(A) \end{array} \right)$$

The simplification in the last line of (iii) above rests on the fact that both truth and belief follow from knowledge, so that conjoining the truth and belief requirements with the knowledge requirement is strictly redundant. What we see in (89) is that the twin relations proposed for *'know'* give us our accustomed behavior for $know_{PROP}$.

Turning to a strongly exhaustive question, consider (90), analyzed with (91) and (92).

(90)   Rupert knows who Anne loves.

(91)
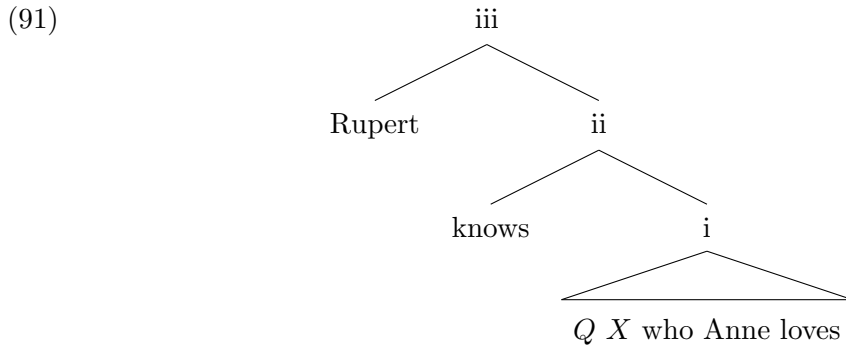
(92)   i : $\lambda p_{\langle \mathbf{s}, \mathbf{t} \rangle} \exists \beta_{\langle \mathbf{e}, \mathbf{t} \rangle}(p = \lambda w'_{\mathbf{s}}(\lambda x(person(w')(x) \wedge loves(w')(x)(A)) = \beta))$

knows : $knows_{QUES}(w^*)$

$$= \lambda P_{\langle \langle \mathbf{s}, \mathbf{t} \rangle, \mathbf{t} \rangle} \lambda y \left(\begin{array}{c} \forall p'(P(p') \rightarrow (believes(w^*)(p')(y) \rightarrow p'(w^*))) \\ \wedge \exists p''(P(p'') \wedge knows(w^*)(p'')(y)) \end{array} \right)$$

ii : $\lambda y \left(\begin{array}{c} \forall p' \left(\begin{array}{c} \exists \beta(p' = \lambda w'(\lambda x \left(\begin{array}{c} person(w')(x) \wedge \\ loves(w')(x)(A) \end{array} \right) = \beta)) \\ \rightarrow (believes(w^*)(p')(y) \rightarrow p'(w^*)) \end{array} \right) \\ \wedge \\ \exists p'' \left(\begin{array}{c} \exists \beta'(p'' = \lambda w'(\lambda x \left(\begin{array}{c} person(w')(x) \wedge \\ loves(w')(x)(A) \end{array} \right) = \beta')) \\ \wedge knows(w^*)(p'')(y) \end{array} \right) \end{array} \right)$

Rupert : R

162

$$\text{iii}: \begin{pmatrix} \forall p' \begin{pmatrix} \exists \beta (p' = \lambda w'(\lambda x \begin{pmatrix} person(w')(x) \wedge \\ loves(w')(x)(A) \end{pmatrix} = \beta)) \\ \rightarrow (believes(w^*)(p')(R) \rightarrow p'(w^*)) \end{pmatrix} \\ \wedge \\ \exists p'' \begin{pmatrix} \exists \beta' (p'' = \lambda w'(\lambda x \begin{pmatrix} person(w')(x) \wedge \\ loves(w')(x)(A) \end{pmatrix} = \beta')) \\ \wedge knows(w^*)(p'')(R) \end{pmatrix} \end{pmatrix}$$

That is, (90) is true iff every strongly exhaustive answer to the question that Rupert believes is true, and Rupert knows at least one strongly exhaustive answer to the question. The second part here will make the semantic contribution of the first part negligible in most cases. Since strongly exhaustive answers are mutually inconsistent, it is hard to imagine, and harder to cultivate intuitions about, a scenario in which Rupert knows one strongly exhaustive answer while believing another, inconsistent, strongly exhaustive answer. In particular, whenever Rupert knows the true strongly exhaustive answer to the question (90) will be true *unless* Rupert has a certain kind of extremely inconsistent beliefs about the question. As far as I know, this is compatible with our intuitions for (90).

Finally, consider the case of the embedded mention-some question in (93), as analyzed in (94) and (95).

(93)   Rupert knows who can help.

(94)
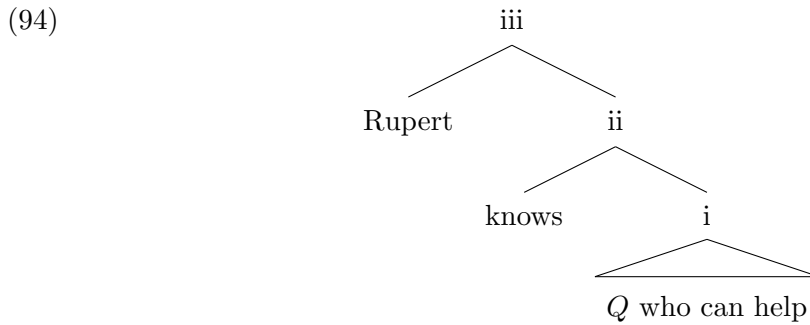
(95)   i : $\lambda p_{\langle \mathbf{s},\mathbf{t} \rangle} \exists \beta_\mathbf{e} (p = \lambda w \exists w'(can(w, w') \wedge helps(\beta, w')))$

       knows : $knows_{QUES}(w^*)$

$$= \lambda P_{\langle\langle s,t\rangle,t\rangle} \lambda y \left(\begin{array}{c} \forall p'(P(p') \rightarrow (believes(w^*)(p')(y) \rightarrow p'(w^*))) \\ \wedge \exists p''(P(p'') \wedge knows(w^*)(p'')(y)) \end{array} \right)$$

ii : $\lambda y \left(\begin{array}{c} \forall p' \left(\begin{array}{c} \exists \beta(p' = \lambda w \exists w'(can(w,w') \wedge helps(\beta,w'))) \\ \rightarrow (believes(w^*)(p')(y) \rightarrow p'(w^*)) \end{array} \right) \\ \wedge \exists p'' \left(\begin{array}{c} \exists \beta(p'' = \lambda w \exists w'(can(w,w') \wedge helps(\beta,w'))) \\ \wedge knows(w^*)(p'')(y) \end{array} \right) \end{array} \right)$

Rupert : R

iii : $\left(\begin{array}{c} \forall p' \left(\begin{array}{c} \exists \beta(p' = \lambda w \exists w'(can(w,w') \wedge helps(\beta,w'))) \\ \rightarrow (believes(w^*)(p')(R) \rightarrow p'(w^*)) \end{array} \right) \\ \wedge \exists p'' \left(\begin{array}{c} \exists \beta(p'' = \lambda w \exists w'(can(w,w') \wedge helps(\beta,w'))) \\ \wedge knows(w^*)(p'')(R) \end{array} \right) \end{array} \right)$

That is, (93) is true iff Rupert knows one mention some answer, and believes only true mention-some answers. This is what we want. In particular, this distinguishes the case where Rupert knows a true mention-some answer and is otherwise unopinionated ((93) is intuitively true) from the case where he knows a true mention-some answer and also believes one or more independent false mention-some answers ((93) is intuitively false). This behavior in the mention-some case is the main payoff of the Twin Relations account.

4.5.3.4 Twin Relations for *'forget'* and Other Predicates

For *'forgot'*, I propose the following pair of relations (I've placed things in the past tense to avoid some issues with the event structure of forgetting):

(96)   a. $forgot_{\exists} = \lambda w \lambda p \lambda x(forgot(w)(p)(x))$

          b. $forgot_{\forall} = \lambda w \lambda p \lambda x(knew(w)(p')(x) \rightarrow forgot(w)(p')(x))$

These give us the following embedding behaviors:

(97)   a. $forgot_{PROP} =$
              $\lambda w \lambda p \lambda x(forgot(w)(p)(x) \wedge (knew(w)(p)(x) \rightarrow forgot(w)(p)(x)))$
              $= \lambda w \lambda p \lambda x(forgot(w)(p)(x))$

          b. $forgot_{QUES} =$

$$\lambda w \lambda P \lambda x \left(\begin{array}{c} \forall p'(P(p') \to (knew(w)(p')(x) \to forgot(w)(p')(x))) \\ \wedge \exists p''(P(p'') \wedge forgot(w)(p'')(x)) \end{array} \right)$$

Thus, on a mention-some-reading, we derive truth-condition on which an individual and a question stand in the relation expressed by *'forgot'* if and only if the individual forgot at least one mention-some answer, and forgot every mention-some answer that he or she knew. This seems appropriate.

For responsive attitudes that do not show problems for the reductive picture sketched so far, we simply put everything in the existential part, giving us the same reduction we had in Chapter 2. For example, if we think that *'be certain'* is well served by the reductive account sketched in the previous chapters, we can give it the following lexical components:

(98)   a. $certain_\exists = \lambda w \lambda p \lambda x (certain(w)(p)(x))$

      b. $certain_\forall = \lambda w \lambda p \lambda x (1)^{18}$

This will be equivalent to our old reductive account, since this effectively undoes all the universal components written into the twin relations theory:

(99)   a. $certain_{PROP} = \lambda w \lambda p \lambda x (certain(w)(p)(x))$

      b. $certain_{QUES} = \lambda w \lambda P \lambda x \exists p''(P(p'') \wedge certain(w)(p)(x))$

Recall that, wile *'know'* required knowledge of *one* true mention-some answer, *'ignorare'* required ignorance of *all* mention-some answers. For the issues with *'ignorare'*, decomposition is of course an option, but, if we wish to handle it in the twin relations scheme, we can do so by putting all the content into the universal component:

(100)   a. $ignorare_\exists = \lambda w \lambda p \lambda x (1)$

      b. $ignorare_\forall = \lambda w \lambda p \lambda x (\neg knows(w)(p)(x))$

---

[18]This is the constant-valued function that maps every triple of a world, a proposition, and an entity to 1 – corresponding to the set of all such triples. This means that the $certain_\forall$ component will always come out true no matter what arguments we give it, so conjoining it in or universally quantifying into it will never add further restrictions.

This gives us an $ignorare_{PROP}$ that is the negation of $know_{PROP}$, while the resulting $ignorare_{QUES}$ denies knowledge of *any* of the answers:

(101)   a. $ignorare_{PROP} = \lambda w \lambda p \lambda x (\neg knows(w)(p)(x))$

   b. $ignorare_{QUES} = \lambda w \lambda P \lambda x \forall p'(P(p') \rightarrow \neg knows(w)(p')(x))$

When describing things decompositionally, it is natural to say that *'ignorare'* scopes existential quantification of answers *under* the negation in its meaning, but we can achieve the same result by trading in existential quantification for universal quantification, and then scoping that over the negation, taking advantage of the standard interchangeability of '$\neg \exists$' and '$\forall \neg$'.

4.5.3.5   How Constraining is the Twin Relations Account?

The twin relations theory is only valuable to the extent that it imposes more substantial constraints on the connection between propositional and question-oriented uses of a responsive predicate than a 'nuclear option' theory which stipulates these independently. We saw earlier that my implementation of the inverted reduction approach, which at first appeared to closely link the propositional use to the question-oriented use, was in fact not a substantial improvement over the nuclear option.

The twin relations theory may not be as constraining as we might like, but it does impose at least two major constraints, both related to the link between the question-oriented and propositional uses, given below.

(102)   For any twin relations pair $(R_\forall, R_\exists)$, if R_{PROP} and R_{QUES} are derived from $(R_\forall, R_\exists)$ by the twin relations theory, then for any world w, entity a, and proposition set P, if $R_{QUES}(w)(P)(a) = 1$ then there is $p' \in P$ such that $R_{PROP}(w)(p')(a) = 1$. (That is, if R relates an entity to an answer set, then it relates that entity to at least one proposition in that answer set.)

The proof of (102) is relatively simple. Assuming $R_{QUES}(w)(P)(a) = 1$, note that, by the question-embedding rule, there must be $p' \in P$ such that $R_\exists(w)(p')(a) = 1$, and that for all $p'' \in P$, $R_\forall(w)(p'')(a) = 1$, since, $p' \in P$, the universal half of this implies that $R_\forall(w)(p')(a) = 1$, so $R_\exists(w)(p')(a) = 1$ and $R_\forall(w)(p')(a) = 1$, which (by

the propositional rule) means that $R_{PROP}(w)(p')(a) = 1$.

(103)   For any twin relations pair $(R_\forall, R_\exists)$, if R_{PROP} and R_{QUES} are derived from $(R_\forall, R_\exists)$ by the twin relations theory, then for any world w, entity a, and nonempty proposition set P, if, for every $p' \in P$, $R_{PROP}(w)(p')(a) = 1$, then $R_{QUES}(w)(P)(a) = 1$. (That is, if R relates an entity to every proposition in a nonempty answer set, then it also relates the entity to that answer set.)

The proof of (103) is also straightforward. If, for every $p' \in P$, $R_{PROP}(w)(p')(a) = 1$, then for every $p' \in P$, $R_\forall(w)(p')(a) = 1$. Since P is nonempty, let p'' be a member of P. We know that $R_{PROP}(w)(p'')(a) = 1$, so $R_\exists(w)(p'')(a) = 1$. So we find that there is $p'' \in P$ such that $R_\exists(w)(p'')(a) = 1$. Between this and the observation about $R_\forall$, we have all the requirements for $R_{QUES}(w)(P)(a) = 1$.

Of these two results, (102) is the more linguistically relevant, for the simple reason that there are few natural propositional attitudes that we can imagine somebody holding of *every* proposition in the answer set to a question, meaning that the conditions for (103) are rarely satisfied.

To see a concrete prediction of (102), consider the made-up predicate '*unicornfident*'. *unicornfident*$_{PROP}$ relates an entity a to a proposition p iff a is confident that p because a unicorn told a that p, and *unicornfident*$_{QUES}$ relates an entity a to a proposition set P iff a is confident of some $p' \in P$. '*unicornfident*' violates (102) because it is possible to be confident of an answer to a question (and so be related to the answer set by *unicornfident*$_{QUES}$) without ever having conversed with a unicorn. The nuclear option (and perhaps inverted reduction) do not exclude '*unicornfident*', so '*unicornfident*' serves as an example of how the twin relations theory constrains the pairing of question-oriented and propositional uses.

To see a concrete prediction of (103), consider '*uncertimes*', already discussed above. *uncertimes*$_{PROP}$ relates an entity a to a proposition p iff it is not the case that a is certain that p, and *uncertimes*$_{QUES}$ relates an entity a to a proposition set P iff there is $p' \in P$ such that it is not the case that a is certain that p', and, for every $p'' \in P$, p'' has never been asserted on the front page of the *Los Angeles Times*. '*uncertimes*' is ruled out by (103) because it is possible to not be certain of

any answer to a question even in a world where at least one answer to that question has been asserted on the front page of the *Los Angeles Times*.

Both (102) and (103) are linking constraints. They are examples how, if we start out picking R_{QUES}, we cannot then pick an arbitrary R_{PROP} and have a pair of R_{QUES} and R_{PROP} that are allowed by the twin relations theory (and likewise for picking R_{PROP} and then choosing an arbitrary R_{QUES}). The entire premise of the nuclear option was that we could choose any R_{PROP} and R_{QUES} we wanted and combine them freely, and we saw that, under certain plausible assumptions, the inverted reduction approach allowed us to do something that was equivalent to this in practice. Results like (102) and (103) allow say that certain properties of R_{QUES} will be unavoidable given a particular R_{PROP}, and certain properties of R_{PROP} will be unavoidable given a particular R_{QUES}.

After a more thorough survey of non-reducibility effects, (102) or (103) might turn out to be too constraining (for example, they seem to rule out Beck and Rullmann's *'agree'*, although, as I said in Section 4.2.1, I am not sure that the empirical claims of Beck and Rullmann (1999) are correct), and they certainly do not rule out every combination of propositional and question-oriented behavior that we would deem implausible, but they at least show that the twin relations theory constrains the connection between R_{PROP} and R_{QUES} in some way, and so illustrate how such a constraint can be achieved with an enriched meaning theory.

4.5.3.6 Twin Relations as Constrained Inverted Reduction

For reasons of presentation, I have framed the Twin Relations account as an account that derives a separate R_{QUES} and R_{PROP} for every responsive predicate R, but we can also do away with R_{PROP} and regard the Twin Relations account as a special case of the inverted reduction account. On this view, it will be a generalization about responsive predicates that every responsive predicate R is derivable from some $R_\forall$ and $R_\exists$ by the R_{QUES} construction, but our theory of embedding will simply be the inverted reduction theory.

Recall the definitions of R_{PROP} and R_{QUES} under the Twin Relations account:

(81) $R_{QUES} = \lambda w \lambda P_{\langle\langle \mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle} \lambda x_{\mathbf{e}} \left(\begin{array}{l} \forall p'_{\langle \mathbf{s},\mathbf{t}\rangle}(P(p') \rightarrow R_\forall(w)(p')(x)) \\ \wedge \exists p''_{\langle \mathbf{s},\mathbf{t}\rangle}(P(p'') \wedge R_\exists(w)(p'')(x)) \end{array} \right)$

(82) $R_{PROP} = \lambda w \lambda p \lambda x (R_\exists(w)(p)(x) \wedge R_\forall(w)(p)(x))$

The definition of R_{QUES} will remain unchanged, but now, instead of using R_{PROP}, we will embed a propositional clause by packaging the proposition in a singleton set, and feeding it into R_{QUES}. The key observation here is that, for any proposition q, feeding $\lambda q'(q' = q)$ into R_{QUES} will always produce the same truth value as feeding q into R_{PROP}. This is because with a singleton set, the distinction between universal and existential quantification in the definition of R_{QUES} is vacuous. If to say that $R_\forall$ relates x to every proposition in $\lambda q'(q' = q)$ is just to say that $R_\forall$ relates x to q, and saying that $R_\exists$ relates x to at least one proposition in $\lambda q'(q' = q)$ is again just saying that $R_\exists$ relates x to q. Before, we evaluated R with respect to a proposition q a world w' and a subject y by evaluating $R_{PROP}(w')(q)(y)$, but it turns out that we will always get exactly the same results by evaluating $R_{QUES}(w')(\lambda q'(q' = q))(y)$ instead. We already saw that the $know_{QUES}$ that I've been advocating will work this way, in the initial discussion of the inverted reduction account, but it turns out that this will work for *any* R_{QUES} and R_{PROP} derivable by the twin relations theory.

4.6   Open Issues

4.6.1   Other Cases of Non-Reducibility

I developed the Twin Relations account as a response to particular observations about the non-reducibility of *'know'* and *'forget'*. These are almost certainly not the only non-reducible responsive predicates, and it is likely that, after a more thorough exploration of the non-reducibility effects we find with responsive predicates, some revisions will be in order.

One case where further work seems especially important *'agree'*. As I mentioned in Section 4.2.1, Beck and Rullmann (1999) make claims about the question embedding behavior of *'agree'* that would make it non-reducible. In particular, for Beck and Rullmann (1999), two different kinds of non-agreement about a mention-some answer have different implications for the truth of a sentence. If there is a mention-

some answer p such that two people fail to agree that p because both of them are unopinionated about the truth value of p, this, according to Beck and Rullmann (1999) does not falsify their agreement to the question that p answers, but, if there is p such that one person believes p to be true, and the other believes it to be false, then this failure of propositional agreement *does* falsify question-agreement. This failure to treat all cases of propositional non-agreement the same means that propositional agreement facts are not enough to determine question-agreement, so Beck and Rullmann's *'agree'* is non-reducible.

What is more striking is that, if we take the Beck and Rullmann (1999) account literally, Beck and Rullmann's *'agree'* cannot be modeled with the Twin Relations approach. This is a consequence of the result in (102).

(102)   For any twin relations pair $(R_\forall, R_\exists)$, if R_{PROP} and R_{QUES} are derived from $(R_\forall, R_\exists)$ by the twin relations theory, then for any world w, entity a, and proposition set P, if $R_{QUES}(w)(P)(a) = 1$ then there is $p' \in P$ such that $R_{PROP}(w)(p')(a) = 1$. (That is, if R relates an entity to an answer set, then it relates that entity to at least one proposition in that answer set.)

If *'agree'* is to be modeled by the twin relations theory, then it must satisfy (102): that is, it must be the case that if some people agree to a question, they agree to at least one answer to that question. Recall, though that for Beck and Rullmann (1999), this need not be the case. In particular, recall that, on Beck and Rullmann's account, if Anne and William are both totally unopinionated about who was murdered, then (14-a) is predicted to be true, since in this case Anne and William believe exactly the same mention-some answers (neither believes any) and exactly the same negated mention-some answers (neither believes any), and are both unopinionated about exactly the same mention-some answers (all of them).

(14-a)   Anne and William agree on who was murdered.

But if Anne and William are both totally unopinionated about who was murdered then there is no (mention-some or strongly exhaustive) answer p to the embedded question such that Anne and William agree that p. Thus, Beck and Rullmann

(1999) predict that *'agree'* can related Anne and William to the question *'who was murdered?'* without relating them to any of its answers, meaning that (102) excludes Beck and Rullmann's agree from being modeled by the twin relations theory.

Note, though, that the prediction that (14-a) will be true when Anne and William are both totally unopinionated about who was murdered is among the less palatable predictions of the Beck and Rullmann (1999) analysis of *'agree'* (my own judgment is that it is untrue in this situation, and this seems to be fairly standard). In order to assess the implications of *'agree'* for the Twin Relations account, and the appropriate handling of *'agree'* more generally, we need to get a better handle on the shape of the data. Once we determine whether *'agree'* displays non-reducibility (it seems likely, but far from certain, that it does), and what kind of non-reducibility it displays (this part is far less certain), we can proceed to see how handle it within a well-behaved theory of embedding, but, since so many of the empirical issues remain unresolved, I have ignored *'agree'* in the development of the Twin Relations account.

4.6.2   Other Lines of Attack

The approaches sketched here represent only a small sample of the wide array of formally possible responses to the non-reducibility observations for *'know'* and *'forget'*, and many of the others deserve further exploration. In particular, a more involved search for a genuinely constraining inverted reduction account would be of interest, since inverted reduction is among the most obvious formal alternatives to standard reduction, and my arguments against inverted reduction make use of various technical details of my proposed implementation of inverted reduction, meaning that other implementations might avoid the problem.

Within the realm of accounts that use a shared enriched lexical meaning, the Twin Relations account is not the only alternative, but only the first approach that comes to mind (for me) when considering the problematic data against the background of the framework that I adopted in Chapter 2. There are no doubt many other possible ways of building enriched lexical meanings for responsive predicates, and of defining question-oriented and propositional behavior from these enriched meanings, and the space of possibilities deserves to be explored in greater detail.

4.6.3 Aspectual/Event Structure

I have provided only two good examples of predicates that present serious problems for reductive approaches: *'know'* and *'forget'*. Both of these have in common that they can be used statively. At the other extreme, a clearly eventive predicate like *'tell'* clearly is only concerned with one answer on a mention-some reading:

(104)   Anne told us where we could buy a newspaper.

Uttered without any context to fix a particular time in the past, (104) is true if and only if there was any salient past episode of Anne furnishing us with a suitable answer. The judgment seems to be that only a single act of telling is needed to validate (104), that that act will generally involve a single proposition-like unit of communicated information, and that what happened at other moments will not be relevant. That is, the eventive nature of *'tell'* seems to be connected with the way that it naturally deals with one proposition at a time. This is very different from the stative *'know'*, which comfortably relates an entity to arbitrarily many unrelated propositions at once. This is enough to make one suspect that event structure is connected with the kinds of non-reducibility effects that we have observed.

This is certainly possible, and it deserves further investigation, but I have nothing to add at this time. I would note that another possibility is simply that, if non-reducibility arises from checking of different relations between an entity and a variety of predicates in the situation of evaluation, then, if we assume eventive predicates are evaluated against very 'small' situations, in which few relations and propositions participate, we would expect it to be hard to derive any non-reducibility effect even if we started them out with a non-reductive lexical semantics. If something like this were right, eventiveness would simply render non-reductive meanings irrelevant, and we might not need any deeper account.

A more thorough survey of responsive predicates and their aspectual/event structure is needed before we can proceed. At present, I can only mark this as a problem for the future.

4.6.4 Presupposition

Another issue that I have not directly addressed is the question of which parts of the semantics of R_{QUES} and R_{PROP} should be treated as assertions, and which as presuppositions, and whether the presuppositional effects should be written into the rules for deriving R_{QUES} and R_{PROP}, or coded into the lexical specification of $R_\exists$ and $R_\forall$, or whether we should say that presupposition triggering is found in both places.

This issue is of interest for a number of reasons. One is that it points to some empirical issues that I have not adequately addressed. In particular, some speakers report that a sense of presupposition failure seems to accompany the untrue examples in my non-reducibility arguments. For instance, returning to my scenario for *'know'* (schematized in (36)), we seem to judge (34) untrue, but many speakers are also inclined to judge (105) untrue, or at least reluctant to deem it true.

(36)

Newspaper available at ...	PaperWorld?	Newstopia?
R's beliefs:	Y	Y
J's beliefs:	Y	?
Facts:	Y	N

(34)   Red knows where Rupert can buy a newspaper.

(105)   Red doesn't know where Rupert can buy a newspaper.

The judgment that neither (34) nor (105) is true is the signature of presupposition failure, and, if this judgment turns out to be robust, then the details of either the Twin Relations account or my particular choices of $know_\exists$ and $know_\forall$ will need to be fleshed out to account for this presupposition. I leave the further investigation of these issues for another time.

Another curious point relating to presupposition relates to the observation that the material in $R_\forall$ often seems, informally, to split out certain things that we are used to counting among the presuppositions of R_{PROP}. For example, with *'know'*, we found that $know_\forall$ imposed a conditional relationship between belief (an important part of the assertion of *'know'*) and truth (the main presupposition of *'know'*). That

is, $know_\forall$ seems to *almost* be placing the assertion of our traditional *'know'* on one side of the →, and the presupposition on the other:

(80-b) $know_\forall = \lambda w \lambda p \lambda x (believes(w)(p)(x) \to p(w))$

A similar issue is seen with *'forget'*, where $forgot_\forall$ introduces a constraint related to knowledge, and *'forget'* generally presupposes prior knowledge. Again, we seem to find something like the presupposition of *'forget'* on one side of the conditional, and something like the assertion on the other:

(96-b) $forgot_\forall = \lambda w \lambda p \lambda x (knew(w)(p')(x) \to forgot(w)(p')(x))$

That is, there is some reason to be suspicious that the division between presupposition and assertion is somehow connected with the structure of lexical meanings in the Twin Relations account.[19]

This idea definitely deserves further exploration, but I admit to not being certain of how to proceed. It would be tempting to try to derive our Twin Relations meanings entirely from information about the presupposition/assertion divide, but, at least as I have articulated things, there is not a straightforward way to do this. Although (96-b) and (80-b) both split an assertion-like and presupposition-like component around the conditional, they do it in different ways: in (80-b), the assertion-like component *believes* appears in the antecedent of the conditional, while in (96-b), it is the presupposition-like component *knew* that appears in the antecedent. In addition, the sense in which *forgot* can be considered the assertion of *'forgot'* is different from the (harder to make precise) sense in which *believes* can be considered the main assertive part of *'know'*, so again we seem to have mismatch between these two cases. However, it is not clear whether a different articulation of $know_\forall$ and $forget_\forall$, or a different formulation of the Twin Relations theory, might bring out a clearer pattern here. In any case, the possibility that the decomposition into $R_\forall$ and $R_\exists$ is somehow connected with presupposition should be noted as a possible area for further study.

---

[19]Compare Lahiri (2002), where the possibility that presupposition accommodation us used to set the restrictor of answer quantification is explored.

4.6.5 Homogeneity and Twin Relations

The reader may have noticed that the R_{QUES} meanings derived by the Twin Relations theory bear a certain resemblance to the augmented R_{QUES} meanings that I proposed in Section 3.6 to deal with homogeneity effects. We might be inclined to try to unify the two, especially in light of the apparently presuppositional nature of both the universal quantification in the Twin Relations theory (discussed in Section 4.6.4) and the universal quantification used to define homogeneity.

Reformatting things slightly to bring out the similarity, the homogeneity presupposition modification gave us $know_{QUES}$ as in (106), and the Twin Relations theory gave us $know_{QUES}$ as in (107).

(106)   Homogeneity presupposition:
$$knows_{QUES} =$$
$$\lambda w \lambda P \lambda x \left(\Pi \left[\forall p' \left(\begin{array}{c} \left(\begin{array}{c} \mathbf{Part}(P)(p') \wedge \\ knows_{PROP}(w)(p')(x) \end{array} \right) \\ \rightarrow \\ \exists p'' \left(\begin{array}{c} P(p'') \wedge p'' \subseteq p' \wedge \\ knows_{PROP}(w)(p'')(x) \end{array} \right) \end{array} \right) \\ \wedge \exists p'''_{\langle \mathbf{s,t} \rangle}(P(p''') \wedge knows_{PROP}(w)(p''')(x)) \right] \right)$$

(107)   Twin relations:
$$know_{QUES} =$$
$$\lambda w \lambda P \lambda x \left(\begin{array}{c} \forall p'(P(p') \rightarrow (believes(w)(p')(x) \rightarrow p'(w))) \\ \wedge \exists p'''(P(p''') \wedge knows(w)(p''')(x)) \end{array} \right)$$

In my Twin Relations account I said $know_\forall$ imposed the constraint that if the subject believed an answer, then that answer had to be true, but, given our data, I might just as well have said that $know_\forall$ required that if the subject believed an answer, then the subject had to know that answer. That is, by changing the lexical entry for 'know', I could just as easily have derived the following $know_{QUES}$, which would have been just as compatible with our data:

(108)   Twin relations (alternative $know_\forall$):

$$know_{QUES} =$$

$$\lambda w \lambda P \lambda x \left(\begin{array}{c} \forall p'(P(p') \to (believes(w)(p')(x) \to knows(w)(p')(x))) \\ \wedge \exists p'''(P(p''') \wedge knows(w)(p''')(x)) \end{array} \right)$$

Note that the resemblance between (108) and (106) is even closer than the resemblance between (107) and (106). It is natural to try to merge them into something like (109):

(109)   Possible hybrid account for *'know'*:

$$know_{QUES} =$$

$$\lambda w \lambda P \lambda x \left(\boldsymbol{\Pi} \left(\forall p' \left(\begin{array}{c} (\mathbf{Part}(P)(p') \wedge believes(w)(p')(x)) \\ \to \\ \exists p'' \left(\begin{array}{c} P(p'') \wedge p'' \subseteq p' \wedge \\ knows(w)(p'')(x) \end{array} \right) \end{array} \right) \right) \right. $$
$$\left. \wedge \exists p'''_{\langle \mathbf{s,t} \rangle}(P(p''') \wedge knows(w)(p''')(x)) \right)$$

I leave the problem of deriving (109) with a constraining enriched meaning theory along the lines of the Twin Relations theory, and of making this account compatible with *'forget'*, for another time.

Until then, if we wish to integrate homogeneity presuppositions into the Twin Relations theory, another option is to leave the truth-conditions in the Twin Relations theory as they stand, and then, for each responsive predicate R attach a homogeneity presuppositions (everything in the scope of $\boldsymbol{\Pi}$ in (106)) given entirely in terms of R_{PROP}. Again, I leave the details for another time, but I think it is important to flag this family of issues as something requiring further attention.

4.7   Concluding Remarks

In this chapter, I have suggested that most approaches to embedding under responsive predicates have relied, explicitly or implicitly, on the assumption that the question-oriented use of such a predicate is to be reduced to its propositional use. I have argued that this assumption is incompatible with our judgments for *'know'*, and runs into some difficulties for our judgments for *'forget'* as well.

After arguing for the non-reducibility of *'know'* and *'forget'*, I explored a number of possible responses. I first discussed responses that try to deny the data or marginalize the problem, either by attributing the issue to pragmatic or other contextual effects, or by combining a reductive approach with a theory that decomposes the problematic predicates into multiple layers, and I (tentatively) rejected both approaches.

Next, I turned to the possibility of a non-reductive semantics for responsive predicates. The first approach, the 'nuclear option', simply abandoned any attempt to connect the propositional and question-embedding uses, essentially treating propositional *'know'* and question-embedding *'know'* as two distinct words, each with its own, distinct, lexically specified semantics. I argued that this was capable of accounting for the data at hand, but should be treated as a last resort. I then explored the possibility of inverting standard reduction approaches by reducing the propositional case to the question-oriented case. I developed an implementation of this account, but showed that the implementation I had devised was no more constraining than the 'nuclear option'. Finally, I proposed the Twin Relations account, which gave a format of enriched lexical meanings for responsive predicates, and a pair of uniform rules for deriving the propositional and question-oriented uses from these enriched meanings. I showed that this account could handle the problematic data for *'know'* and *'forget'*, and that, unlike the 'nuclear option', it did impose linking conditions on the semantics of the propositional and question-oriented uses of the predicate, and provided examples of hypothetical responsive predicate meanings that the Twin Relations account excluded, but that the 'nuclear option' allowed.

I did not attempt any systematic exploration of the space of possible enriched meaning accounts, so I cannot offer any remarks on the comparative strengths and weaknesses of the Twin Relations account and its (as yet undeveloped) nearest competitors. As a minimum, I think that any competing account should handle the data that I presented for *'know'* and *'forget'*, and should give examples of constraints it places on the semantic connection between the question-oriented and propositional uses that are roughly comparable in strength to those derived for the Twin Relations account.

In the remaining chapters, I will, for the most part, ignore the observations made in this chapter, and stick with the theory from Chapter 2. This is because I want

to turn to issues that seem to be (mostly) independent of non-reducibility, and the theory for Chapter 2 is substantially easier to work with.

CHAPTER 5

Speculations on non-*'wh'* Questions

'Do you advocate the overthrow of the United States government by force
or violence?' After some thought she wrote, 'Force.'

Dillard (1987)

This dissertation is concerned primarily with *'wh'*-questions, but of course *'wh'*-questions ought to fit into a general picture of question semantics. In this chapter, I want to quickly look at how the account in Chapter 2 might be expanded to handle other kinds of questions and question embedding. I show that *'yes'/'no'* questions and 'concealed questions' are easily accommodated, and discuss the nature of the difficulties that arise for alternative questions. Throughout this chapter, I will be assuming the system presented in Chapter 2, ignoring the issues raised in Chapter 4. These issues are mostly independent, and would only complicate the presentation of the other ideas.

5.1 *'yes'/'no'* Questions

Consider the *'yes'/'no'* question (1):

(1)   Is Rupert alive?

There are two answer propositions for (1): (2-a) and (2-b):

(2)   a.   Rupert is alive.
   b.   Rupert is not alive.

These are the relevant answers for embedding under a responsive predicate: (3-a) is true if and only if Alex knows one of these answers – that is, if either (3-b) or (3-c)

179

is true.

(3) a. Alex knows whether Rupert is alive.

   b. Alex knows that Rupert is alive.

   c. Alex knows that Rupert is not alive.

Except for question-specific material (*'whether'* in the embedded case, and inversion in the unembedded case), (1) appears to built in the same way as (2-a), and it is customary to try to derive one as a variant of the other. I will assume that (1) is built by stacking the X and Q operators on top of (2-a), and, more generally, that a *'yes'/'no'* question is formed from something like the corresponding declarative by the application of these operators.

In more detail, my analysis of (1) is given in (4).

(4)
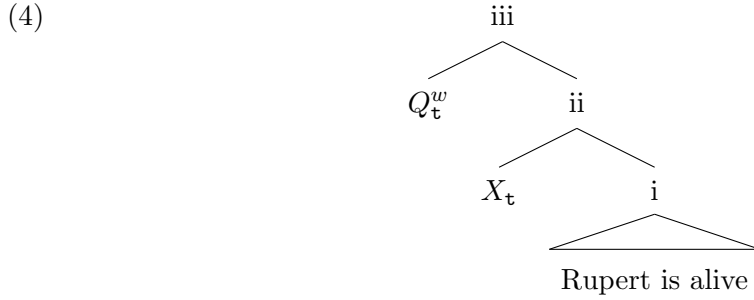
That is, we take the clause (2-a), and treat it exactly like we treated an abstract when deriving an exhaustive reading of a *'wh'*-question. I briefly work through the interpretation of (4) in (5):

(5) i : $alive(w)(Rupert)$

   $X_{\mathbf{t}}$: $\lambda\gamma_{\mathbf{t}}\lambda\delta_{\mathbf{t}}(\gamma = \delta)$

   ii : $\lambda\delta_{\mathbf{t}}(alive(w)(Rupert) = \delta)$

   $Q_{\mathbf{t}}$: $\lambda\alpha_{\langle\mathbf{s},\langle\mathbf{t},\mathbf{t}\rangle\rangle}\lambda p_{\langle\mathbf{s},\mathbf{t}\rangle}\exists\beta_{\mathbf{t}}(p = \lambda w'_{\mathbf{s}}(\alpha(w')(\beta)))$

   iii : $(\lambda\alpha\lambda p\exists\beta_{\mathbf{t}}(p = \lambda w'(\alpha(w')(\beta))))(\lambda w\lambda\delta_{\mathbf{t}}(alive(w)(Rupert) = \delta))$
    $= \lambda p\exists\beta_{\mathbf{t}}(p = \lambda w'(alive(w')(Rupert) = \beta))$

Assuming that there are only two truth values (or at least that standard quantification over truth values is restricted to only two), there are only two propositions satisfying the answerhood conditions given in (iii) of (5). These are the propositions in (6):

(6)   a. $\lambda w'_{\mathbf{s}}(alive(w')(Rupert) = 1)$
      b. $\lambda w'_{\mathbf{s}}(alive(w')(Rupert) = 0)$

Formula (6-a) gives the proposition that Rupert is alive (i.e. the meaning of (2-a)), and formula (6-b) gives the proposition that Rupert is not alive (i.e. the meaning of (2-b)), so the answer set we derive contains of the affirmative and negative answers to the question, and nothing else.

What we just saw is that our X and Q operators derive the desired answer set for a *'yes'/'no'* question without any modification.

This result is not a distinctive feature of my system, but is inherited from its predecessors: most operators for forming strongly exhaustive answers or answer sets will do something similar when applied to propositional clauses. In particular, the strong exhaustivity approaches in Groenendijk and Stokhof (1984) and, with slight modification, Heim (1994) are capable of producing similar results.

It would be natural to ask what happens when we apply just the Q operator to a propositional clause. The answer, on the face of things, is type mismatch: Q_τ takes an argument of type $\langle \mathbf{s}, \langle \tau, \mathbf{t} \rangle \rangle$, and the intensional type of a propositional clause is of course $\langle \mathbf{s}, \mathbf{t} \rangle$, so there is no conventional choice of τ that allows us to apply Q_τ to a propositional clause.

Type mismatch explanations are usually not very satisfying. What happens if we try to find some way to generalize Q operator to avoid type mismatch? To see one reasonably natural way of generalizing Q, recall that, to handle multiple-*'wh'* questions, I already allowed τ in the definition of Q_τ to be a tuple type. I had assumed that we were only concerned n-tuples for $n \geq 1$, but I haven't provided any reason to exclude 0-tuples.

How could we extend the type theory with 0-tuples? Well, to start with, we would have to include a type of for 0-tuples, written (). There is only one 0-tuple, so () will not be especially interesting. I further assume that, for all types τ, $\langle (), \tau \rangle = \tau$,

and, for every expression η of type τ, and every expression μ of type $()$, $\eta(\mu)$ is an epression of type τ with the same semantic value as η. That is, everything can act as a function on 0-tuples, but applying it to a 0-tuple doesn't do anything. If we allow this, then we can instantiate Q and $Q_{()}$:

(7) $Q_{()} = \lambda\alpha_{\langle \mathbf{s},\langle (),\mathbf{t}\rangle\rangle}\lambda p_{\langle \mathbf{s},\mathbf{t}\rangle}\exists\beta_{()}(p = \lambda w'_{\mathbf{s}}(\alpha(w')(\beta)))$

But 0-tuples never do any work, so (7) is equivalent to (8):

(8) $Q_{()} = \lambda\alpha_{\langle \mathbf{s},\mathbf{t}\rangle}\lambda p_{\langle \mathbf{s},\mathbf{t}\rangle}(p = \lambda w'_{\mathbf{s}}(\alpha(w')))$

From (8), we can infer that, for all propositions p', $Q_{()}(p') = \lambda p(p = \lambda w'_{\mathbf{s}}(p'(w'))) = \lambda p(p = p')$. That is, $Q_{()}$ takes a propositional argument, and returns the set containing only that proposition. This doesn't seem to correspond to any natural notion of question-formation.[1] For example, trying to repeat our analysis of (1) without X, illustrated in (9), we derive an answer set as described in (10):

(9)
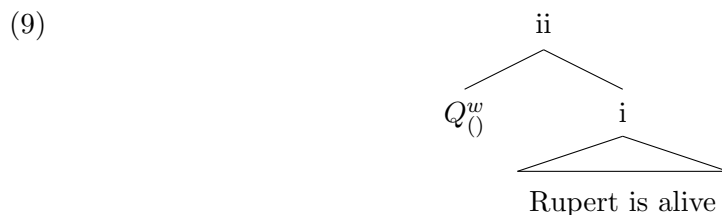
(10)   i : $alive(w)(Rupert)$

   $Q_{()}$: $\lambda\alpha_{\langle \mathbf{s},\mathbf{t}\rangle}\lambda p_{\langle \mathbf{s},\mathbf{t}\rangle}(p = \lambda w'_{\mathbf{s}}(\alpha(w')))$

   ii : $\lambda p_{\langle \mathbf{s},\mathbf{t}\rangle}(p = \lambda w'_{\mathbf{s}}(alive(w')(Rupert)))$

That is, we derive an answer set for (1) that contains only one proposition: the one expressed by (2-a):

(1)   Is Rupert alive?

---

[1]It does, however, provide a convenient way of packaging a single proposition in a proposition set, which might be relevant for something like the inverted reduction approach discussed in Sections 4.5.2 and 4.5.3.6.

(2-a)   Rupert is alive.

There is, as far as I know, no reason to think that (1) has such a reading, so I propose that we continue to exclude 0-tuples, or else that we declare that trivial answers sets (those that contain fewer than 2 propositions) are excluded by some other constraint.

5.2   Alternative Questions

The account in Chapter 2 does not do a good job with alternative questions like (11).

(11)   Is William a traitor or is Andrew a traitor?

The basic difficulty is that no scoping of disjunction with respect to the Q and X operators does what we want. One of the options produces the *'yes'/'no'* reading of (11), but none of them produce the alternative question reading. There are five combinations of the components to consider as possible accounts: if X is present, then *'or'* might conceivably scope under X and Q, or between X and Q, or over both X and Q. Without X, *'or'* can scope over or under Q.

I will go through all five combinations below, but I want to quickly sketch the issue informally. The basic problem is this: we already have a way of getting from a proposition to a *'yes'/'no'* answer set (applying $X_\mathtt{t}$ and then $Q_\mathtt{t}$). If we start by taking the disjunction (12), and then apply the *'yes'/'no'* question-formation process to that, then we get an answer set for (11) that contains two propositions: (12) and its negation. This is an available reading for (11), so we need to derive it somehow, and using the normal steps that form a *'yes'/'no'* question on a disjunctive proposition will naturally produce it.

(12)   William is a traitor or Andrew is a traitor.

On the other hand, suppose we take a disjunction of the *'yes'/'no'* questions (13) and (14). This will give us the union of their answer sets. The answer set of (13) contains exactly propositions in (15), and the answer set of (13) contains exactly

the propositions in (16), so the membership of the answer set of this disjunction of questions will contain all four propositions.

(13)   Is William a traitor?

(14)   Is Andrew a traitor?

(15)   a.   William is a traitor.
       b.   William is not a traitor.

(16)   a.   Andrew is a traitor.
       b.   Andrew is not a traitor.

So scoping disjunction high relative to formation of *'yes'*/*'no'* questions will give us answerhood conditions on which (15-b) is an answer to (12), meaning that if (17) is true (and Robin knows nothing else about William or Andrew) then (18) will be true.

(17)   Robin knows that William is not a traitor.

(18)   Robin knows whether William is a traitor or Andrew is a traitor.

There might be a 'choice' reading of (11) where these answerhood conditions are right, but it is not a very natural reading, and in any case these answerhood conditions still do not capture the alternative question reading that is our main concern here.

There are a number of other options presented by the possibility of leaving out X or assigning the disjunction intermediate scope, but the above points illustrated the basic difficulty. I now turn to the derivation of all five possibilities.

5.2.1 Q over *'or'* (no X)

If we scope disjunction low, and have no X, then (11) can be analyzed as follows:

(19)
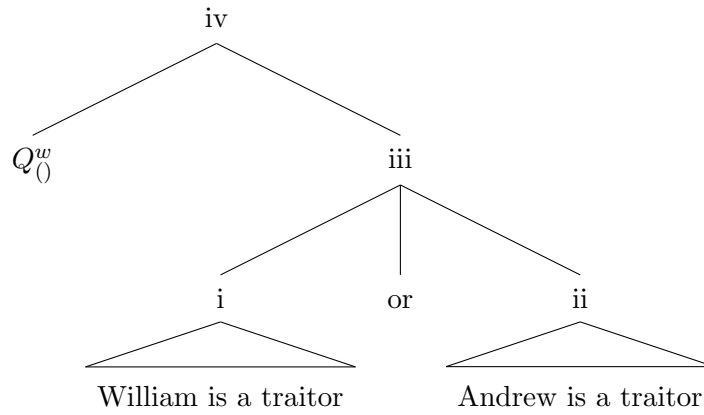
(20)   i : $traitor(w)(W)$

        ii : $traitor(w)(A)$

        iii : $traitor(w)(W) \vee traitor(w)(A)$

        $Q_{()}$: $\lambda\alpha_{\langle \mathbf{s},\mathbf{t}\rangle}\lambda p_{\langle \mathbf{s},\mathbf{t}\rangle}(p = \lambda w'_{\mathbf{s}}(\alpha(w')))$

        iv : $\lambda p_{\langle \mathbf{s},\mathbf{t}\rangle}(p = \lambda w'(traitor(w')(W) \vee traitor(w')(A)))$

This gives us the answer set containing only the proposition $\lambda w'(traitor(w')(W) \vee traitor(w')(A))$. This is not a plausible reading for (11), and certainly is not a good alternative question reading.

5.2.2   *'or'* over Q (no X)

If we leave out X, but scope disjunction over Q, we get the following:

(21)

(22)   i : $traitor(w)(W)$

ii : $traitor(w)(A)$

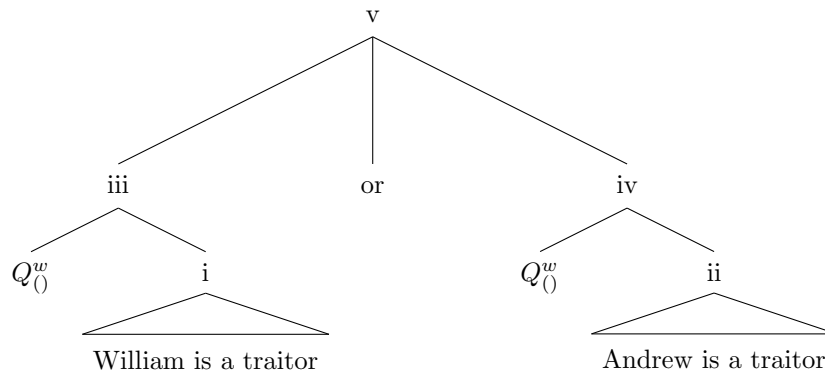$Q_{()} : \lambda\alpha_{\langle s,t\rangle}\lambda p_{\langle s,t\rangle}(p = \lambda w'_s(\alpha(w')))$

iii : $\lambda p_{\langle s,t\rangle}(p = \lambda w'(traitor(w')(W)))$

iv : $\lambda p_{\langle s,t\rangle}(p = \lambda w'(traitor(w')(A)))$

v : $\lambda p_{\langle s,t\rangle}(p = \lambda w'(traitor(w')(W)) \vee p = \lambda w'(traitor(w')(A)))$

This gives us an answer set containing just two propositions: the proposition that William is a traitor and the proposition that Andrew is a traitor. This doesn't sound like an entirely unreasonable alternative question reading. The typical way to answer the alternative question (11) is by uttering either (15-a) or (16-a), so, at first glance, it doesn't seem unreasonable to suggest that the answer set for (11) contains just these two propositions.

(11)   Is William a traitor or is Andrew a traitor?

(15-a)   William is a traitor.

(16-a)   Andrew is a traitor.

Unfortunately, a closer look a these results reveals a number of empirical problems for this answer set as an account of alternative questions. First, it predicts that (23)should entail (18), even when Andrew is also a traitor and Robin doesn't know it, since on this analysis (15-a) is in the answer set of (11), and, under our embedding rule, Robin only needs to know one proposition in the answer set to make (18) true.

(23)   Robin knows that William is a traitor.

(18)   Robin knows whether William is a traitor or Andrew is a traitor.

This seems like it is probably wrong: on the alternative question reading of (18), if both William and Andrew are traitors, then I think the intuition is that Robin must know both.

If we adopt this as our analysis of alternative questions, we also predict that (18) and (24) are incompatible, which, again, doesn't seem quite right:

(24)   Robin knows that neither William nor Andrew is a traitor.

There is some intuition that (18) presupposes that exactly one of William and Andrew is a traitor, which would make concerns about the cases where both of them are traitors or neither is a traitor irrelevant. However, this doesn't really make things better for this account, which does not do anything to derive such a presupposition.

Beyond these empirical considerations, this account suffers from the theoretical difficulty that it makes use of $Q_{()}$, which is not motivated for any other type of question. This is not a serious drawback, but, even if the empirical issues mentioned immediately above can be overcome, we should remember that adopting this account will require us to find something to say about the availability or unavailability of $Q_{()}$ in other settings.

5.2.3 Q over X over *'or'*

This is is the configuration that would give us a *'yes'/'no'* reading for the whole question:

(25)

William is a traitor or Andrew is a traitor
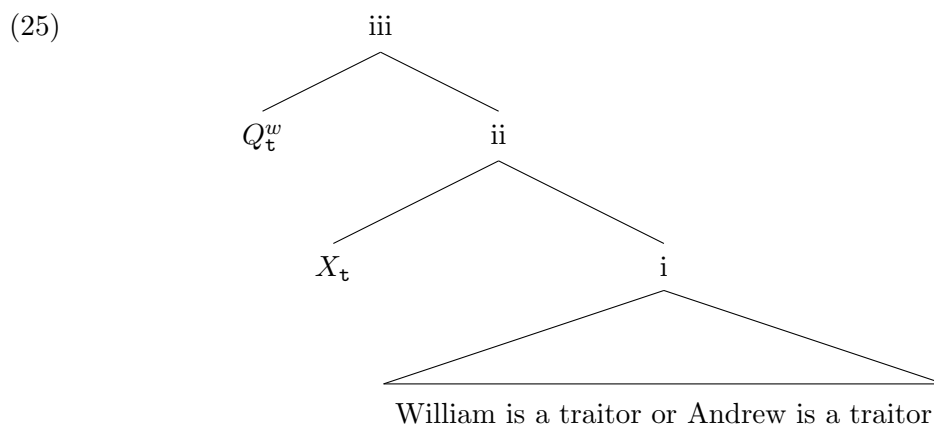
(26)   i : $traitor(w)(W) \vee traitor(w)(A)$

        $X_{\mathsf{t}} : \lambda\gamma_{\mathsf{t}}\lambda\delta_{\mathsf{t}}(\gamma = \delta)$

$$\text{ii} : \lambda\delta_\mathtt{t}((traitor(w)(W) \lor traitor(w)(A)) = \delta)$$

$$Q_\mathtt{t} : \lambda\alpha_{\langle\mathtt{s},\langle\mathtt{t},\mathtt{t}\rangle\rangle}\lambda p_{\langle\mathtt{s},\mathtt{t}\rangle}\exists\beta_\mathtt{t}(p = \lambda w'_\mathtt{s}(\alpha(w')(\beta)))$$

$$\text{iii} : \lambda p_{\langle\mathtt{s},\mathtt{t}\rangle}\exists\beta_\mathtt{t}(p = \lambda w'_\mathtt{s}((traitor(w')(W) \lor traitor(w')(A)) = \beta))$$

That is, in this case we derive the answer set containing (12) and its negation:

(12)   William is a traitor or Andrew is a traitor.

This is the reading that makes the question a *'yes'/'no'* question. Such a reading is probably available (at least if we apply the disjunction at a suitable syntactic level), but it is not the alternative question reading that we are looking for, which is only answered by a proposition that identifies who among William and Andrew is a traitor.

5.2.4 Q over *'or'* over X

This is the only intermediate-scope option provided by the framework adopted:

(27)
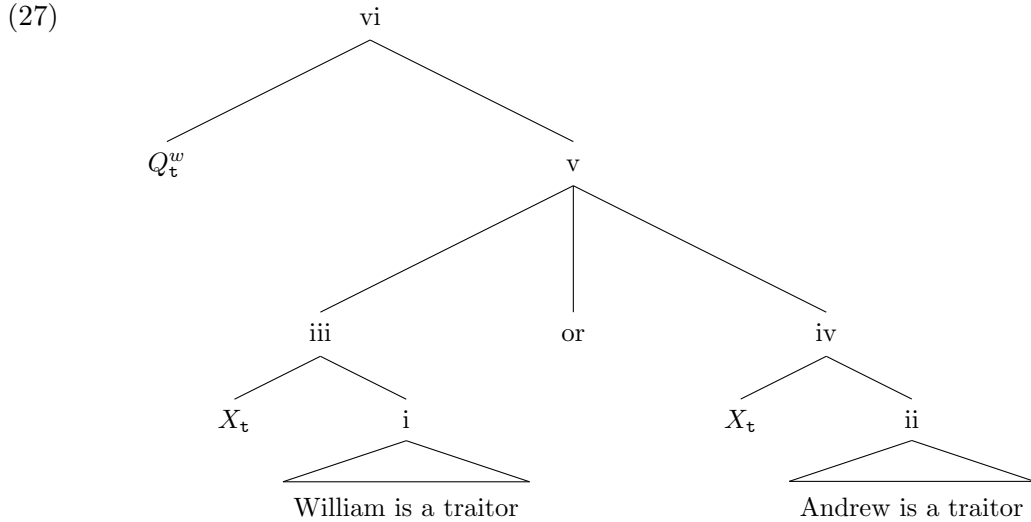
(28) $\text{i} : traitor(w)(W)$

$\text{ii} : traitor(w)(A)$

$X_\mathtt{t} : \lambda\gamma_\mathtt{t}\lambda\delta_\mathtt{t}(\gamma = \delta)$

$$\text{iii} : \lambda\delta_{\mathsf{t}}(traitor(w)(W) = \delta)$$

$$\text{iv} : \lambda\delta_{\mathsf{t}}(traitor(w)(A) = \delta)$$

$$\text{v} : \lambda\delta_{\mathsf{t}}(traitor(w)(W) = \delta \vee traitor(w)(A) = \delta)$$

$$Q_{\mathsf{t}} : \lambda\alpha_{\langle\mathsf{s},\langle\mathsf{t},\mathsf{t}\rangle\rangle}\lambda p_{\langle\mathsf{s},\mathsf{t}\rangle}\exists\beta_{\mathsf{t}}(p = \lambda w'_{\mathsf{s}}(\alpha(w')(\beta)))$$

$$\text{vi} : \lambda p_{\langle\mathsf{s},\mathsf{t}\rangle}\exists\beta_{\mathsf{t}}(p = \lambda w'_{\mathsf{s}}(traitor(w')(W) = \beta \vee traitor(w')(A) = \beta))$$

That is, this derives the answer set consisting of two propositions: (12) and (29):
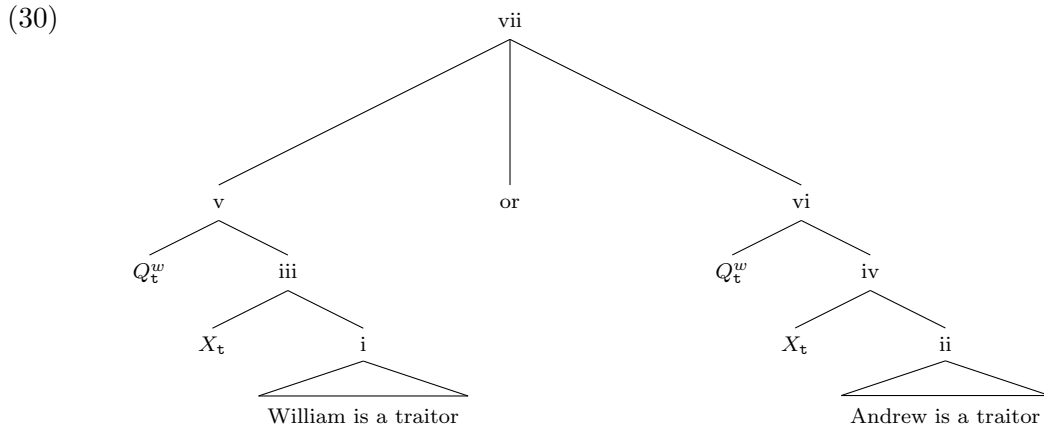
(12)   William is a traitor or Andrew is a traitor.

(29)   William is not a traitor or Andrew is not a traitor.

This does not seem to be a well-motivated reading, and it is certainly not the reading we're looking for.

5.2.5   'or' over Q over X

This is the case where we take the union of two 'yes'/'no' answer sets:

(30)

(31) $\text{i} : traitor(w)(W)$

$\quad\quad\text{ii} : traitor(w)(A)$

$\quad X_{\mathsf{t}} : \lambda\gamma_{\mathsf{t}}\lambda\delta_{\mathsf{t}}(\gamma = \delta)$

$$\text{iii} : \lambda\delta_{\mathtt{t}}(traitor(w)(W) = \delta)$$

$$\text{iv} : \lambda\delta_{\mathtt{t}}(traitor(w)(A) = \delta)$$

$$Q_{\mathtt{t}} : \lambda\alpha_{\langle \mathtt{s},\langle \mathtt{t},\mathtt{t}\rangle\rangle}\lambda p_{\langle \mathtt{s},\mathtt{t}\rangle}\exists\beta_{\mathtt{t}}(p = \lambda w'_{\mathtt{s}}(\alpha(w')(\beta)))$$

$$\text{v} : \lambda p_{\langle \mathtt{s},\mathtt{t}\rangle}\exists\beta_{\mathtt{t}}(p = \lambda w'_{\mathtt{s}}(traitor(w')(W) = \beta))$$

$$\text{vi} : \lambda p_{\langle \mathtt{s},\mathtt{t}\rangle}\exists\beta_{\mathtt{t}}(p = \lambda w'_{\mathtt{s}}(traitor(w')(A) = \beta))$$

$$\text{vii} : \lambda p\exists\beta_{\mathtt{t}}\left(\begin{array}{l} p = \lambda w'_{\mathtt{s}}(traitor(w')(W) = \beta) \\ \vee p = \lambda w'_{\mathtt{s}}(traitor(w')(A) = \beta) \end{array}\right)$$

That is, we have the answer set consisting of four propositions: the proposition that William is a traitor, the proposition that William is not a traitor, the proposition that Andrew is a traitor, and the proposition that Andrew is not a traitor.

As already discussed before this exhaustive survey of possible scope combinations, this is not what we want. The biggest problem is the inclusion of the two negative answers (the proposition that William is not a traitor and the proposition that Andrew is not a traitor). For purposes of embedding, these are just not sufficient answers to the question. If Andrew is a traitor, and William is not, and I know that William is not a traitor, but not that Andrew is, then it does not seem right to say that I know whether William or Andrew is a traitor. The derivation above, however, predicts that I do know whether William or Andrew is a traitor in this case.

5.2.6 Alternative Questions: General Remarks

The core problem here is not specific to my account. Any theory of question-formation in English that tries to build both *'yes'/'no'* and alternative questions with the same tools of question formation, and that deals in answer sets, is likely to have a similar problem, at least if it adopts a classical Boolean disjunction[2] and associates propositional clauses with something like world-sets. The most common response, seen in Groenendijk and Stokhof (1984) and Karttunen (1977), is simply to stipulate separate processes of question-formation for the two types of questions.

---

[2]Disjunction is famously complicated, so the adoption of a classical Boolean disjunction is almost certainly inappropriate, although it is not clear whether any independently motivated disjunction will help us here. For examples of a family of recent approaches that try to make sense of some issues in question semantics by using a richer disjunction, the reader is referred Groenendijk (2009) and Mascarenhas (2009).

Given normal assumptions about disjunction and propositions, one of the only potential ways to avoid this is to adopt a extra structural level at which to apply the disjunction.[3]

If our question formation strategy works for *'yes'/'no'* questions, then it is difficult to make it interact with disjunction in the needed way. Possible lines of attack stipulating a special rule for one kind of question or the other, complicating the semantics of *'or'* (perhaps along the lines of what is done in Mascarenhas (2009) and Groenendijk (2009)), complicating our notion of propositions, and finding some level at which to apply disjunction that does not behave the same as the levels that I've considered. Of course, this list does not exhaustively cover the possible alternatives, and I do not mean to exclude the possibility that there is some other way of addressing this difficulty within the kind of system that I have been describing, but this is how I see the state of the problem. Approaches that adopt a very different account of question meaning, or of the meanings of any of the components of the alternative question, may behave very differently, in ways about which it is hard to offer even tentative generalizations.

If we have a theory that handily derives alternative questions, one other option is to treat *'yes'/'no'* questions as elliptical alternative questions, so that, for example, (32-a) is understood as a syntactic variant on (32-b):

(32)   a.   Robin knows whether Andrew is a traitor.
        b.   Robin knows whether Andrew is a traitor or Andrew is not a traitor.

This approach has some appeal, but of course it does not solve my other problem, which is my difficulty in integrating alternative questions with my account of *'wh'*-questions.

I hope that this section has done a reasonable job of outlining the difficulties that the framework adopted here faces. I am, as I have said before, concerned mainly with *'wh'*-questions, so I will not commit myself to a particular solution, although one stipulative treatment of alternative questions is sketched below in order to illustrate

---

[3]For example, Spector (2010) presents an elegant account that handles this by applying disjunction to intermediate objects from the account Karttunen (1977), called 'proto-questions'. These proto-questions don't cleanly correspond to anything in my account, so this approach is not available to me.

a possible way of integrating them into this system.

5.2.7 A Brute-Force Solution for Alternative Questions

One rather inelegant way of resolving the issue of alternative questions, while leaving my Q and X framework intact, would be to simply to declare that the *'or'* of English alternative questions is not really the same word as any other *'or'* found in English.

This concludes my discussion of alternative questions. I intend to leave this issue unresolved, but I hope that the above does a reasonable job of outlining the difficulties that the framework adopted here faces. I am, as I have said before, concerned mainly with *'wh'*-questions, so I leave alternative questions for another time.

5.2.7.1 Plausibility of a Brute-Force Solution

This approach is not inherently unreasonable, since there are a number of languages where the word used to separate alternatives in the alternative question construction is different from the ordinary disjunction in the language. One well-known example of this are classical Latin,[4] and Finnish provides a well-document example of something like this in a living language, with *'tai'* serving as the default disjunction, and *'vai'* reserved for the formation of alternative questions, with the complication that *'tai'* can apparently also form alternative questions.[5]

Malagasy provides another example: here *'sa'* is found only in alternative ques-

---

[4]In Latin, *'an'* is used to separate clauses in an alternative question, but disjunction in other contexts is usually expressed by other words, such as *'aut'* and *'vel'*. A responsible survey of the intricacies of disjunction in Latin would take us too far afield.

[5]Kaiser (2004) reports that (33) allows either a *'yes'*/*'no'* or alternative question reading, while (34) allows only an alternative question reading:

(33)   Huomasiko   Pekka   miehen   tai naisen?
        Noticed-QUEST Pekka-NOM man-ACC tai woman-ACC
        Did Pekka notice man or woman? [*'y'*/*'n'* or alt. reading]

(34)   Huomasiko   Pekka   miehen   vai naisen?
        Noticed-QUEST Pekka-NOM man-ACC vai woman-ACC
        Did Pekka notice man or woman? [alt. reading only]

tions and with an alternative-like meaning in a tag question construction,[6] while *'na'* is use for disjunction in other contexts, so that, for example, (36) and (37) cannot receive a *'yes'/'no'* answer, even as a joke.

(36)   Hijanona ianao sa handeha?
        Fut+stay you.sg or fut+go
        Are you staying or leaving?

(37)   Te-hisotro   dite na kafe   ianao?
        Want+fut+drink tea or coffee you.sg
        Do you want to drink tea or coffee?

Unfortunately, the distribution of *'na'* in Malagasy is considerably more limited than, for example, the distribution of *'or'* in English, making it difficult to get a handle on the precise contrast.

   Another example of this general sort of division of labor is found in Egyptian Arabic. Here, the two *'or'* words are *'aw'* and *'wala'*. Both can be used with a meaning similar to ordinary disjunction, although *'wala'* is a negative polarity item. In questions, however, *'wala'* forces an alternative reading,[7] so (38) allows a reply of *'Iowa'* (*'yes'*), but (39) does not:

(38)   Ayza   teroohy teshoofy film aw teshtery hagaat?
        Want(2.sg.f) go   see   film or buy   something?
        Do you want to go see a film or shop?

(39)   Ayza   teroohy teshoofy film wala teshtery hagaat?
        Want(2.sg.f) go   see   film or   buy   something?
        Do you want to go see a film or shop?

The data available to me at this time make it difficult to be certain of whether an alternative question readings is available with *'aw'*, but *'wala'* does appear to be

---

[6](35) provides an example of *'sa'* in a tag question:

(35)   Vizaka ve   ianao? Sa tsy izany?
        Tired   QUES you.sg or not that
        Are you tired, or is that not the case?

[7]*'wala'* can also be used to form tag questions.

preferred for translating English sentences were an alternative question reading is more salient.

As the incomplete and heavily hedged nature of these notes suggests, more work is needed to get a reasonable cross-linguistic picture of the relationship between alternative questions and ordinary disjunction, but I think the above observations at least suggest that assigning *'or'* a different meaning in alternative questions is not completely absurd.

5.2.7.2 An Example of a Brute-Force Solution

If we are willing to take the step of declaring that the *'or'* of English alternative questions is not really the same word as any other *'or'* found in English, things become *too* easy, since there are many possible ways to write a lexical entry for it that will produce the desired predictions within the current framework. One example would interpret the *'or'* of alternative questions as the binary function on propositions given in (40)

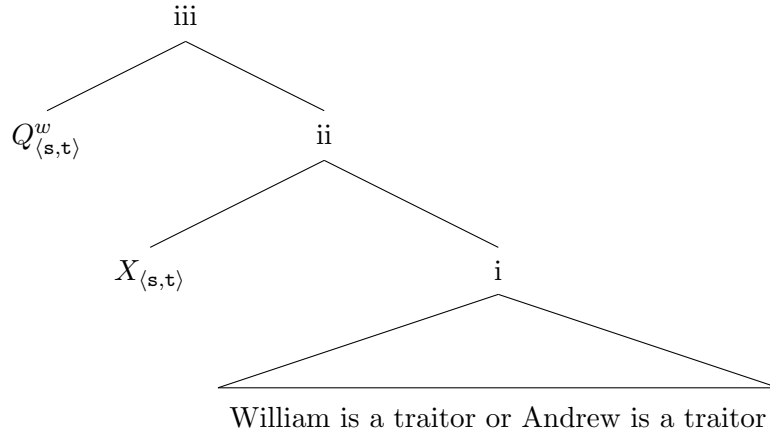(40) $\quad \lambda p \lambda q \lambda w' \lambda p'((p(w') \wedge p = p') \vee (q(w') \wedge q = p'))$

For any p and q, the function given in (40) maps a world w' to the proposition-set containing whichever of p and q are true in w'. I should note that the only important property of (40) is that its value in a world w' encodes in some way the information of which of p and q are true in w'. Any function that does this will serve our purpose equally well.[8]

Given (40), applying $X_{\langle \mathbf{s}, \mathbf{t} \rangle}$ and then $Q_{\langle \mathbf{s}, \mathbf{t} \rangle}$ will produce a plausible alternative question interpretation, as set out in (42) and (43):

---

[8]For example, where $\mathbf{n}$ is the type of natural numbers, (41) would also do the job:

(41) $\quad \lambda p \lambda q \lambda w' \lambda m_{\mathbf{n}}((p(w') \wedge m = 1) \vee (q(w') \wedge m = 2))$

194

(42)

$$
\begin{array}{c}
\text{iii} \\
\diagup \quad \diagdown \\
Q^w_{\langle \mathbf{s},\mathbf{t}\rangle} \qquad \text{ii} \\
\diagup \quad \diagdown \\
X_{\langle \mathbf{s},\mathbf{t}\rangle} \qquad \text{i} \\
\diagup \qquad \diagdown \\
\text{William is a traitor or Andrew is a traitor}
\end{array}
$$

(43)

$$
\text{i} : \lambda p' \left(\begin{array}{c} (traitor(w)(W) \wedge \lambda w'(traitor(w')(W)) = p') \\ \vee (traitor(w)(A) \wedge \lambda w'(traitor(w')(A)) = p') \end{array} \right)
$$

$$
X_{\langle\langle \mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle} : \lambda \gamma_{\langle\langle \mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle} \lambda \delta_{\langle\langle \mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle}(\gamma = \delta)
$$

$$
\text{ii} : \lambda \delta (\lambda p' \left(\begin{array}{c} (traitor(w)(W) \wedge \lambda w'(traitor(w')(W)) = p') \\ \vee (traitor(w)(A) \wedge \lambda w'(traitor(w')(A)) = p') \end{array} \right) = \delta)
$$

$$
Q_{\langle\langle \mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle} : \lambda \alpha_{\langle \mathbf{s},\langle\langle\langle \mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle,\mathbf{t}\rangle\rangle} \lambda p_{\langle \mathbf{s},\mathbf{t}\rangle} \exists \beta_{\langle\langle \mathbf{s},\mathbf{t}\rangle,\mathbf{t}\rangle}(p = \lambda w''_{\mathbf{s}}(\alpha(w'')(\beta)))
$$

$$
\text{iii} : \lambda p \exists \beta (p =
$$
$$
\lambda w''_{\mathbf{s}}(\lambda p' \left(\begin{array}{c} (traitor(w'')(W) \wedge \lambda w'(traitor(w')(W)) = p') \\ \vee (traitor(w'')(A) \wedge \lambda w'(traitor(w')(A)) = p') \end{array} \right) = \beta))
$$

That is, p is in the answer set for node (iii) iff p identifies some proposition-set β as the set of all true propositions p' such that p' is either the proposition that Andrew is a traitor or the proposition that William is a traitor. That is, an answer p is a proposition that says whether which of William and Andrew are traitors, and which aren't.[9] These appear the right answerhood conditions for the alternative question.

The above brute force theory was sketched briefly for illustrative purposes, and I will not explore its strengths and weaknesses further. There is, of course, a great deal more to be said about alternative questions, but much of it must wait for a more detailed investigation of disjunction in general. For present purposes, the two points to remember are that Chapter 2 does not get alternative questions 'for free' (in contrast

---

[9]On this implementation, the answer set also includes the contradictory proposition $\lambda w(0)$, which is what we get whenever we pick a β that includes any proposition besides the proposition that William is a traitor and the proposition that Andrew is a traitor.

with the situation for *'yes'/'no'* questions), and that it can be amended to handle alternative questions, if we are willing to make some fairly arbitrary stipulations about the semantic contribution of *'or'* in these questions.

5.3 Concealed Questions

The term 'concealed question' is used to describe the question-like behavior of certain nominal phrases when embedded under responsive predicates.[10] The concealed question reading is exemplified by (44):

(44)　　William knows the capital of Moldova.

This sentence has an 'acquaintance reading', on which it conveys William's familiarity with some city (in particular, the one that happens to be the capital of Moldova). This corresponds to the only reading of (45).

(45)　　William knows Chişinău.

We are interested in the other reading of (44): the reading that seems to be naturally paraphrased by (46-a). On this reading, (44), together with the fact that Chişinău is the capital of Moldova, allows us to infer the truth of (46-b).

(46)　　a.　William knows what the capital of Moldova is.
　　　　b.　William knows that Chişinău is the capital of Moldova.

The availability of paraphrases like the one in (46-a) is what makes the term 'concealed question' a natural way to describe this phenomenon – although there is no overt question in (44), there seems to be a *'wh'*-question, or at any rate something with a similar semantics, hiding someplace in (44).

　　There are many important semantic and syntactic problems in the study of concealed questions that I will not address here. One important distributional issue in English is that concealed question readings are readily available with responsive

---

[10]A few rogative embedders, like *'depend on'*, may also allow concealed question readings.

embedders like *'know'*, *'agree'*, and *'tell'*, and probably available with the rogative embedder *'depend on'*, but unavailable or only marginally available with most rogative predicates, including *'wonder'*, as illustrated in (47).

(47)   a.   Rupert knows the capital of Moldova.

         b.   Red and Janna agree on the capital of Moldova.

         c.   Rupert told Anne the capital of Moldova.

         d.   The capital of Bougainville depends on which government you recognize.

         e.   The price of milk depends on the price of fuel.

         f.   The police asked my name.

         g.   *Anne asked Rupert the capital of Moldova.

         h.   *Anne wonders the capital of Moldova.

A good overview of the distributional issues can be found in Nathan (2006), and Romero (2010) provides some examples of how *'ask'*-type embedders can embed concealed questions in Spanish, indicating that the cross-linguistic picture of how concealed questions are distributed is even more complex.

Another interesting pattern (discussed by, for example Caponigro and Heller (2007)) is that concealed questions typically involve complex phrases prepositional prepositional phrases or relative clauses – concealed question readings are apparently unavailable for names or simple definite descriptions:[11]

(48)   a. #Red knows the president.

           ($\neq$ Red knows who the president is.)

      b. #Anne doesn't know William.

           ($\neq$ Anne doesn't know which person William is.)

The above issues were mentioned in recognition of the fact that concealed questions are a complex phenomenon, and that I do not know how to account for them completely. Having recognized this, I want to point out that my approach to forming strongly exhaustive answer sets by applying the X and Q operators gives us a theory that derives a plausible answer set meaning for concealed questions. Like my obser-

---

[11]The example of 'Liam asked the time' violates both these generalizations, but this seems to be a non-productive, perhaps idiomatic, special case.

vations about *'yes'/'no'* questions above, this is not peculiar to my theory, but falls out in a similar manner from most ways of defining strongly exhaustive question-formation operators, as has been noted by others. (For example, Lahiri (2002) gives a treatment of concealed questions involving a version of Heim's strongly exhaustive answer operator, Cumming (2006) points out that the (strongly exhaustive) question-formation rule from Groenendijk and Stokhof (1984) derives plausible concealed question meanings, and Romero (2005) also describes an analysis that handles concealed questions by applying an exhaustive answer operator to an individual concept.)

Assume, for the sake of exposition, that *capital* is a function of type $\langle \mathsf{s}, \langle \mathsf{e}, \mathsf{e} \rangle \rangle$ that maps a world w and an entity x to the capital of x in w (and that raises a presupposition failure if x has no capital in w). The concealed question *'the capital of Moldova'* can be analyzed by applying X and Q to the definite description as in (49), yielding a concealed question meaning as in (50)

(49)
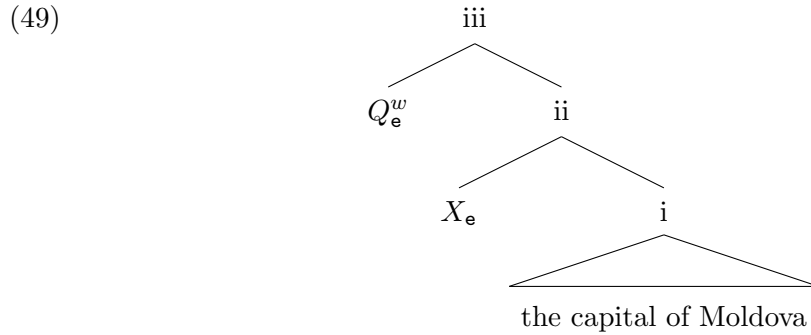
(50)   i : $capital(w)(Moldova)$

   X_{e} : $\lambda\gamma_{\mathsf{e}}\lambda\delta_{\mathsf{e}}(\gamma = \delta)$

   ii : $\lambda\delta_{\mathsf{e}}(capital(w)(Moldova) = \delta)$

   Q_{e} : $\lambda\alpha_{\langle \mathsf{s}, \langle \mathsf{e}, \mathsf{t} \rangle \rangle}\lambda p_{\langle \mathsf{s}, \mathsf{t} \rangle}\exists\beta_{\mathsf{e}}(p = \lambda w'_{\mathsf{s}}(\alpha(w')(\beta)))$

   iii : $\lambda p_{\langle \mathsf{s}, \mathsf{t} \rangle}\exists\beta_{\mathsf{e}}(p = \lambda w'_{\mathsf{s}}(capital(w')(Moldova) = \beta))$

That is, if the intension of the definite description *'the capital of Moldova'* is a function that maps every world to the capital of Moldova in that world, then the concealed question represented by (49) is the set of all propositions that identify some entity as the capital of Moldova. In other words, applying the same operations that

198

we would usually apply to get the strongly exhaustive reading of a *'wh'*-question, or to build a *'yes'/'no'* question, to an expression of type e gives us the set of propositions that identify some entity as the extension of the expression we started with. In the case illustrated in (49), these propositions are plausibly answers to the question *'what is the capital of Moldova?'*, giving the concealed question the needed question-like meaning.

When we apply the question-embedding rule from Chapter 2 to this answer set, we get the reading we want for (44) – we predict that this sentence should be true if and only if William knows at least one of the propositions in the set identified by (iii) of (50), which is to say if and only if he knows which entity is the capital of Moldova.

(44)   William knows the capital of Moldova.

(51)

$Q\ X$ the capital of Moldova
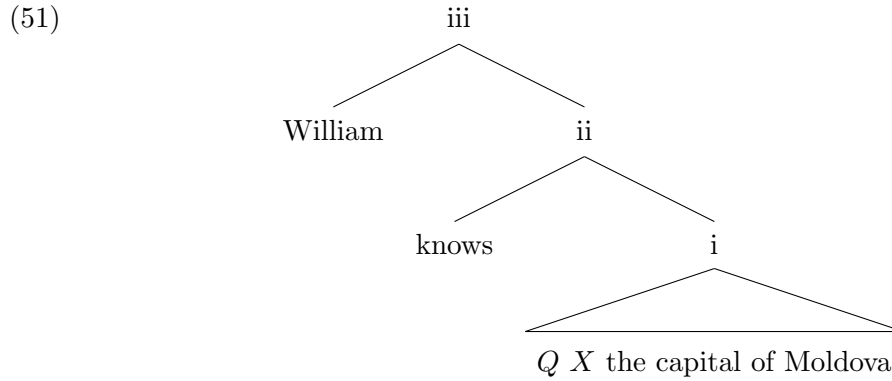
(52)   i : $\lambda p_{\langle\mathsf{s},\mathsf{t}\rangle}\exists\beta_{\mathsf{e}}(p = \lambda w'_{\mathsf{s}}(capital(w')(Moldova) = \beta))$

knows : $\lambda P_{\langle\langle\mathsf{s},\mathsf{t}\rangle,\mathsf{t}\rangle}\lambda z\exists p'(P(p') \wedge knows(w^*)(p')(z))^{12}$

ii : $\lambda z\exists p'\left(\begin{array}{c}\exists\beta_{\mathsf{e}}(p' = \lambda w'_{\mathsf{s}}(capital(w')(Moldova) = \beta)) \\ \wedge knows(w^*)(p')(z)\end{array}\right)$

William : $William$

iii : $\exists p'\left(\begin{array}{c}\exists\beta_{\mathsf{e}}(p' = \lambda w'_{\mathsf{s}}(capital(w')(Moldova) = \beta)) \\ \wedge knows(w^*)(p')(William)\end{array}\right)$

---

[12]This is the reductive version of question-embedding *'know'* based on the account from Chapter 2. Adding the 'no false beliefs' requirement would only complicate the formulas without adding much to the point of the present discussion, which is mainly concerned with the semantics of the concealed question itself.

This concludes my brief illustration of one way to accommodate some basic issues with concealed questions into the present framework.

5.4   Concluding Remarks

This is all I will have to say about the treatment of basic non-*'wh'* questions within this system. As we have seen, the system for building question meanings based on the Q and X operators produces acceptable results for *'yes'/'no'* questions and concealed questions without any special adjustment, at lest in simple cases. For even the simplest alternative questions, some additional machinery will be needed. I have pointed out that the basic difficulty with alternative quetstions is not really specific to any of the details of the approach I have adopted, but is a consequence of the more basic fact that the answer sets *'yes'/'no'* questions and the answer sets of alternative questions stand in a relationship that does not seem to be much like classical disjunction.

CHAPTER 6

Speculations on Mention-Some Readings

'Would you tell me, please, which way I ought to go from here?'

'That depends a good deal on where you want to get to,' said the Cat.

'I don't much care where—' said Alice.

'Then it does't matter which way you go,' said the Cat.

'—so long as I get *somewhere*,' Alice added as an explanation.

'Oh, you're sure to do that,' said the Cat, 'if you only walk long enough.'

Alice felt that this could not be denied, so she tried another question.

Carroll (1865)

The baseline theory in Chapter 2 treats the ambiguity between mention-some and strongly exhaustive questions a result of the presence or absence of the silent X operator. Thus, the system predicts that, as far as the grammar is concerned, both readings are equally available to all *'wh'*-questions.

This account is a reasonable place to start, but it is also somewhat unsatisfying. The first cause for concern is that the ambiguity involved must simply be stipulated: we allow the X operator to be present or absent, but this doesn't connect with any other observable linguistic phenomenon. The second issue is that it seems likely that this account overgenerates: there are few cases where a single embedded question clearly has access to both readings, and a number of cases where only one reading appears to be present. For example, it seems virtually impossible to find any evidence for a strongly exhaustive reading with (1-a), while (1-b) is always (or almost always) strongly exhaustive. (1-c) favors a mention-some reading, but, as noted by Groenendijk and Stokhof (1984), seems to be able to get a strongly exhaustive reading in a suitable context.

(1)   a.   Anne was surprised by who was at the party.

       b.   Anne knows who was at the party.

       c.   Rupert knows where he can buy an Italian newspaper.

In this chapter, I want to explore some possible ways of explaining away or otherwise constraining mention-some readings. Unfortunately, I will not be able to offer anything like a conclusive, unified picture, but I will try to highlight a couple of possible ways of approaching the problem. In Sections 6.1 and 6.2, I discuss two initially tempting lines of attack, and explain why, at least in their simple forms, I suspect they are inadequate. In Section 6.3, I explore the prospects of an analysis that allows responsive predicates to select for the presence or absence of the X operator, and in Section 6.4, I describe an approach that makes the X operator obligatory, and attributes the availability of mention-some readings to a scope interaction between the X operator and an existential in the question (so, on this account, the fact that a mention-some reading is available with (1-c) but not (1-b) is attributed to the presence of the possibility modal *'can'* in the former but not the latter). Finally, in Section 6.5, I try to take stock of the various approaches. I do not think any of the approaches I describe offers a complete and satisfying account of mention-some readings, but I think that several of them show promise, and might contribute to more complete pictures in the future.

6.1   Pragmatic Accounts

6.1.1   Sketch of a Pragmatic Account

A natural first reaction to the mention-some phenomenon is to attempt a Gricean explanation. The idea here is that we adopt a theory in which the *semantic* contribution of a question is always its strongly exhaustive answer set, but that various practical constraints can prevent a cooperative conversation partner from providing a strongly exhaustive answer in response to a question. In this case, the cooperative partner will attempt to furnish an answer that will be useful and informative for the asker, in light of the asker's presumed reasons for asking the question.

To see this kind of reasoning in action, consider again (2):

(2)   Where can I buy an Italian newspaper?

Even if we assume that this question's semantic contribution is a strongly exhaustive answer set, there are a number of reasons why a cooperative conversational participant might not provide an exhaustive answer when asked this question. For example, it could be that the cooperative partner does not know, or is not sufficiently confident of, any exhaustive answer. Even with a reasonable domain restriction, it will take some highly specialized expertise to be sure one's own mental list of places where Italian newspapers can be bought is complete. In this case, the conversation partner will be unable to cooperatively assert any proposition in the answer set of the question. The next best thing is to try to provide a proposition that narrows down the space of possible exhaustive answers in a way that serves the asker's needs. Now, what is the most likely reason for the asker to have uttered (2)? In an ordinary context, it is presumably that the asker wishes to buy an Italian newspaper. If this is the asker's reason for requesting the information, then answering by identifying any one place of purchase (assuming that it is nearby, open, has reasonable prices, and so forth) will serve the asker's needs almost as well as giving a complete list.

A second reason for not providing an exhaustive answer is that, even if one can confidently provide an exhaustive answer, enumerating all the places where Italian newspapers can be bought is likely to be time-consuming, and sufficiently over-informative that it will mainly cause needless confusion. We might then reason that, by providing a mention-some answer, the conversation partner is deciding not to give the asker the information they literally asked for, but instead to give them an answer that they probably prefer.

A natural approach to mention-some readings, then, would be to say that, as far as the semantics is concerned, they simply don't exist. This approach would say that there are only strongly exhaustive readings, but that the communicative acts that count as cooperative replies to a question may not all express propositions in that question's answer set, for reasons like those discussed above.

6.1.2 Problems for a Pragmatic Account

Framed in Gricean terms, this kind of response has a serious flaw: it cannot account for the availability of mention-some readings in embedded questions. We have seen embedded mention-some readings before, and as a reminder here are two examples where a question retains its mention-some reading when embedded under a responsive predicate:

(3)   a.   Joyce is certain of where she can buy an Italian newspaper.
            $\approx$ There is a place x such that Joyce is certain that she can buy an Italian newspaper at x.
    b.   I'm sure Alex remembers where he can buy an Italian newspaper.
            $\approx$ I'm sure there is x such that Alex remembers that he can buy an Italian newspaper at x.

With these examples, we seem to have truth-conditional judgments that are naturally analyzed in terms of a mention-some answer set interacting with a responsive attitude (which is what the accounts Chapters 2 and 4 do with these kinds of examples). If the real answerhood conditions of the embedded questions are exhaustive, then the truth-conditions of the embedding sentences will involve knowledge of exhaustive answers. This would make the judgments in (3) hard to explain in classical Gricean terms. Whatever mechanism accounts for mention-some readings seems to be more deeply entangled with the compositional semantics.

One refinement of this kind of pragmatic account, pursued by van Rooij (2004), is to make the contextual information that is used by the above pragmatic reasoning into a parameter of semantic evaluation. I will not fully engage with van Rooij's proposal here, but, briefly, the basic idea is that one of the parameters of evaluation is some semantic object that encodes the relative usefulness of different pieces of information, and that a proposition counts as an answer to a question, for purposes of embedding, only if it is maximally useful among a certain set of propositions that bear on the question.[1]

---

[1]This is not a completely accurate portrayal of van Rooij's account, which is quite intricate. One thing to note is that, as things are framed by van Rooij (2004), maximal usefulness is assessed by comparison with a maximal *true* answer, so some adjustments would need to be made to account for non-veridical responsive predicates, but I see every reason to think that the needed refinement

The first thing to note about this utility-based approach is that it represents a fairly radical departure from the normal approach to formal semantics. In standard semantic theories, these kinds of utility and purpose information are brought into the compositional semantics only by a small number of specialized operators (especially modals). Van Rooij proposes to make a dependency on these facts an integral part of the meaning of all questions, or of all cases of question-embedding. For van Rooij (2004), this is part of a more general program of entangling these kinds of strategic considerations with the compositional semantics. I think this extreme should be a last resort, and that we should strive for a compositional semantics that assembles conventional meanings in a way that limits contextual intrusions to cases where the data makes them unavoidable.

In addition to this admittedly rather abstract concern, the kind of approach that attributes mention-some readings to question purpose considerations has the serious drawback that it seems ill-equipped to account for the various grammatical features that seem to affect the availability of mention-some readings. An example is the contrast between (4) and (5).

(4)   Who has leprosy?

(5)   Who are some people with leprosy?

These two questions are intuitively about the same property, and they involve virtually synonymous abstracts ((6) and (7), respectively).[2]

(6) $\lambda x(person(w)(x) \wedge leper(w)(x))$

(7) $\lambda x(person(w)(x) \wedge \exists y(person(w)(x) \wedge pl(x) \wedge leper(w)(x) \wedge y = x))$

of van Rooij's account is possible.

[2]These abstracts assume an ontology of plurals. I adopt the notation that a predicate symbol without any number subscript means the set of all single or plural entities with a property, and plural and singular features are separate predicates of entities. The space of entities is presumed to be an atomic lattice – $sg(x)$ is true iff x denotes an atom of this lattice, and $pl(x)$ is true iff x denotes a non-atom that has at least one atom as a part. Most predicates, including *person* and *leper* are assumed to have extensions determined by their behavior in the atomic case, so a plural entity is in the extension of *person* iff all of its atomic parts are. The details of which plural ontology is assumed are not terribly important for our purposes. The reader is referred to, e.g., Link (1997) for a discussion of how this kind of theory can be fleshed out.

If the availability of mention-some readings is determined by an interaction between facts about the purpose of the question and some intensional object derived from the meaning of the abstract, then (4) and (5) ought to both have access to mention-some readings, but this doesn't seem to be the judgment. Instead, (4) is naturally understood as requesting a strongly exhaustive answer, while (5) is generally satisfied by, and indeed seems to prefer, a mention-some answer. That is, it appears that (5) receives a mention-some interpretation, while (4) does not, for reasons not readily attributed to any semantic difference between the abstracts involved. This is a problem for van Rooij (2004) because it is not clear why the maximal useful information about (6) would consistently be different from the maximal useful information about (7).

Although I think that the kind of pragmatic effect that I've been discussing cannot account for mention-some readings in general, I think that the Gricean story sketched above may be able to explain *some* ostensibly 'mention-some' interpretations of unembedded questions. It provides a plausible reason why a mention-some reply would be acceptable even on a strongly exhaustive semantics, so it shows that the acceptability of a mention-some answer for a unembedded question is not by itself good evidence that the semantics of that question involve a mention-some reading. When we get a perceptible contrast between two similar unembedded questions, as in (4) and (5), or when we observe a mention-some effect with an embedded question, we ought to take it seriously as possible evidence of a semantic difference, but in other cases the pragmatic story is useful in narrowing down the set of questions with mention-some readings.

6.2   Domain Effects

Another tempting approach to mention-some readings is to analyze them as an extreme case of domain restriction. This approach works for some apparent mention-some examples, but I think that it fails to handle the core cases. I will proceed to look at the issue of extreme domain restrictions as seen in a few different examples.

6.2.1 'What's awesome?'

I want to begin with example (8), in which the embedded question appears to be non-exhaustive.[3]

(8)   Do you know what's awesome?

I don't have much to say about the communicative value of the unembedded question in (8), but I want to appeal to an intuition about what's involved in *knowing what's awesome*. If my office-mate utters (8) out of the blue, and I dutifully respond with (9), I am not just denying that I know a standard strongly exhaustive answer to (10) (that is, I am not merely expressing the obvious fact that my knowledge doesn't include a complete list of things that are awesome), nor am I denying that I know any mention-some answer (that is, I am not asserting that I know of nothing that is awesome). What I commit myself to when I say (9) is that I don't know what the awesome thing that the asker had in mind is.

(9)   No, I don't know what's awesome.

(10)   What's awesome?

The apparent non-exhaustivity of the embedded occurrences of (10) in (8) and (9) is easily explained within a strongly exhaustive framework: it can be attributed to a domain restriction. The key feature of (9) is that its truth hinges on the matter of whether I know that a particular thing is awesome: the thing that the asker has in mind. I can know about dozens of other awesome things, but that will not make (9) any less true, as long as I don't know about the one awesome thing under discussion. This suggests the following account: (10) is interpreted as a strongly exhaustive question, but with a domain restriction. If Andrew asked (8), then the domain under consideration will be the domain of things that Andrew has in mind (or something similar), so in that case all instances of (10) should be understood as equivalent to (11):

---

[3]I am indebted to Heather Burnett for brining (8) to my attention.

(11)   What among the things that Andrew has in mind is Awesome?

If there is just one awesome thing that Andrew is thinking about (the Facebook integration on his new phone, say), then knowing what's awesome means knowing that that's *the* awesome thing that Andrew is thinking of – that is, knowing the strongly exhaustive answer.[4] (8) has some peculiarities that we might at first have been tempted to attribute to a mention-some reading, but it seems that it's equally well-described as a strongly exhaustive case. We might naturally ask whether *all* mention-some examples can be handled in this way. Before discussing in detail my reasons for thinking that they can't, I want to look at some other examples that can perhaps be thought of as this kind of extreme domain restriction.

6.2.2   Trivia Questions vs. Echo Questions

I now want to consider an infelicitous question-answer pair. Not long ago I encountered (12-a) on a list of trivia questions, with (12-b) as the answer listed at the bottom of the page.[5]

(12)   a.   What was considered a sin in the 16th and 17th century?
        b.   Eating chocolate.

There seems to be something not quite conversationally coherent about asking this trivia question. In the absence of any significant context, there are certainly many other things that were considered sins in the centuries in question, so insisting on (12-b) as the only acceptable or correct answer seems to place to great a burden on the answerer. At the same time, (12-a), being a trivia question, clearly expects a

---

[4]This use of domain restrictions is my way of implementing the idea that some kind of specificity adheres to *'what'* in this example. It may be that some more nuanced notion of specificity is needed here, but the only thing that matters for present purposes is that specificity issues can be sorted out within the abstract. The main point is that the abstract for (10) picks out the set containing only the thing that is picked out as the specific awesome entity in the context, and that, given such a meaning for the abstract, the strongly exhaustive reading built by applying X and Q will give the desired answerhood conditions. Even if the reader is unconvinced by the particulars of my approach to specificity, they should note that the sense of a specificity effect seen in (8) highlights the difference between (8) and a standard mention-some reading, which typically allows great freedom to pick one answer or another, instead of singling out one specific answer.

[5]The source for this example is a small piece of folded cardboard placed on the tables at the vegan restaurant Native Foods in Los Angeles, California.

unique answer that can be given briefly. For this reason, we understand (12-a) as a question about which activity or activities in some suitably restricted domain was or were considered sinful in the centuries in question, but, in the context of a trivia card, we have no way of knowing what this domain might be – the question becomes a game not of testing our trivia knowledge, but of asking us to guess which sin the author of the question was thinking of, leading to our unease with the question-answer pair.

To make (12-b) a coherent answer to (12-a), we need to go to some lengths to contextualize the question, perhaps turning it into an echo question (or something similar) in the process:

(13)   Red [to Alex]: And so, in Europe, eating chocolate was ...
        [Janna walks in]
        Red: ... regarded as sinful in the 16th and 17th centuries.
        Janna: Excuse me, what was considered a sin in the 16th and 17th centuries?
        Alex: Eating chocolate.

In (13), Alex's response is perfectly appropriate. This response is also a sufficient answer for purposes of embedded questions. If I am describing the events related in (13), I can accurately (if a bit clumsily) describe Alex's utterance with (14-a), and the change in Janna's information state after the exchange with (14-b):

(14)   a.   Alex told Janna what was considered a sin in the 16th and 17th centuries.
        b.   So now Janna knows what was considered a sin in the 16th and 17th centuries.

The truth of (14-a) and (14-b) in this context indicates that the proposition Alex communicated enjoys the status of an answer for purposes of semantic answerhood conditions, and is not just a 'good enough' pragmatic substitute for an answer. Since it appears that Alex only communicated that eating chocolate was considered a sin in the 16th and 17th centuries, this might seem to indicate that here the question receives a mention-some reading.

Like the *'What's awesome?'* case, this example has a natural explanation in terms of exhaustive answers and domain restrictions. Note that, although Alex need only identify one thing that was considered sinful to answer Janna's question, Alex has no choice in which thing to identify – the alleged sinfulness of chocolate consumption must be mentioned. This property of there being a unique true answer is unlike what we associate with mention-some readings, but, again, just like what we get for strongly exhaustive readings. The natural solution, as with the other example, is to say that the Janna's question is has a narrow, strongly context-influenced domain restriction, so that its more explicit paraphrase is something like (15):

(15)   What activity that was just mentioned was considered a sin in the 16th and 17th centuries?

What Alex's answer in fact conveys, in context, is not just that eating chocolate was considered a sin, but that eating chocolate was the very thing that Red was just talking about being considered a sin. That is, the proposition Alex communicated was the unique true answer to an exhaustive reading of (15). The apparent acceptability of a less-than-exhaustive answer is just the result of immediate conversational context providing an extremely narrow domain restriction.

To see that we have exhaustivity with an extreme restriction, consider the variant (16):[6]

(16)   Red [to Alex]: And so, in Europe, eating chocolate and drinking coffee were both ...
         [Janna walks in]
         Red: ... regarded as sinful in the 16th and 17th centuries.
         Janna: Excuse me, what was considered a sin in the 16th and 17th centuries?
         Alex: Eating chocolate.

In this case, something is conspicuously incomplete about Alex's answer, and saying that Alex told Janna what was considered sinful, or that Janna now knows what was

---

[6]I don't have any good reason to believe that either eating chocolate or drinking coffee was actually considered sinful at the time in question, but I will assume both were considered sins for purposes of developing these questions.

sinful, is not appropriate. Here, the natural domain restriction is still to things that were regarded as sinful and were under discussion, but now an exhaustive answer over this restricted domain requires mentioning two activities, and, in (16), Alex has failed to provide this.

Things get even worse if Alex provides an answer that identifies a single sin that was not mentioned at all, as in (17):

(17)   Red [to Alex]: And so, in Europe, drinking coffee was ...
        [Janna walks in]
        Red: ... regarded as sinful in the 16th and 17th centuries.
        Janna: Excuse me, what was considered a sin in the 16th and 17th centuries?
        Alex: Eating chocolate.

In (17), Alex has not provided a good answer at all (even if eating chocolate was in fact considered a sin), and describing this situation by saying that Alex told Janna what was considered a sin, or that Janna came to know what was considered a sin, is inappropriate.

The patterns we see indicate that, in all these examples, Alex must answer Janna's question by exhaustively identifying the sin or sins that Red was just talking about, and that Alex has no freedom to choose between competing answers. This is what we would expect from a domain-restricted strongly exhaustive reading, not a mention-some reading.

6.2.3   Domain Restrictions as a General Account of Mention-Some Readings

The kind of extreme domain restriction seen is, as argued, perfectly well analyzed in terms of strong exhaustivity, but it produces a superficially non-exhaustive effect, accounting for a variety of cases where a single instance seems to be a sufficient answer to a 'wh'-question. This effect can also be seen in the relative naturalness of the following exchange (at least in a situation where Professor Worth is widely known as a recent critic of the plan):

(18)   Anne: Who criticized the mayor's plans for renovating the high school?

211

Red: Professor Worth did.

Anne: Thanks! Now I know who criticized the mayor's plans for renovating the high school.

It is, of course, pretty implausible that only one person criticized a prominent local politician's position on this sort of issue. Red is naming only one person instead of providing what we would normally think of as an exhaustive answer, so, to the extent that this is a natural exchange, we apparently have something that looks similar to a mention-some example, without the somewhat contrived and specific contexts and turns of phrase seen earlier.

In spite of this, a domain-type account can plausibly handle (18): as in the other cases, the context we imagine to make (18) okay is one in which it is understood that some particular criticizing event is the topic of conversation. For example, if somebody's having criticized the Mayor's plans was all over the newspapers that day, and a topic of general conversation, but Anne managed not to exactly catch *who* the critic was (or she has forgotten), then the exchange is natural, and in that case, if Professor Worth is the one whose criticism was widely reported, then Professor Worth is the only person that Red can cooperatively name. If at some previous point the principal of the school also denounced the mayor's plans, but that's not the denouncing event that everybody's been talking about, then identifying the principal as a denouncer of the mayor's position does not resolve the question, and Anne's claim to now know the answer is, in context, inappropriate.

Given the generality of this kind of analysis, It is tempting to try to use it to account for all mention-some examples. As with the Gricean approach, the domain approach faces some serious difficulties. Most notably, it fails to account for the way that canonical mention-some readings (like (19)), unlike strongly exhaustive readings, allow freedom to choose one incomplete answer over another.

(19)   Where can I buy an Italian newspaper?

On a strongly exhaustive reading, there may be any number of wrong answers to the question, but, once the context and intent of the question are sufficiently nailed down, the domain restriction ought to be unique or nearly unique, at which point we

should expect that there will be only one acceptable true answer. A mention-some question like (19) could, when asked in a particular context, be equally well answered with the names of any of several newsstands: unlike a strongly exhaustive reading, it can allow many different true propositions as completely adequate answers. With a domain restriction (or other narrowing or specificity-inducing effect) in a strongly exhaustive reading, there is still a unique right answer, in a way that there is not with a mention-some question like (19). Consider, for example, the domain restriction in (20). It is true that (20) might not require a list of all the people who attended the talk in question – it might, as we've seen, require only an identification of the people in a particular department that showed up (so that it could be paraphrased as (21)), but, once the context is fixed, (20) is still exhaustive with respect to its domain restriction. In a context that makes the domain restriction clear, (20) has a unique right answer: the exhaustive identification of all people in the restricted domain who showed up for the talk. With (20), unlike (19), it is not enough to list any subset of the people who showed up for the talk: to fully answer the question, one must provide an exhaustive answer to the restricted question (21).

(19)   Where can I buy an Italian newspaper?

(20)   Who showed up for that talk?

(21)   Who in our department showed up for that talk?

When each context seems to allow a question only one satisfying true answer, even if that answer is in one sense or another incomplete, then we may hope to attribute this incompleteness to an interaction between strong exhaustivity and a domain effect. If, as with (19), the semantics really regards more than one answer as acceptable, then we have a more substantial challenge to exhaustivity. This feature of uniqueness of answers will be important in distinguishing domain-restricted exhaustive readings from genuine mention-some readings.

We saw the unique true answer effect of strong exhaustivity in several of the examples above, but this is not what we see with classic mention-some examples. In the context of (13), (14-b) can only be true if Janna knows that eating chocolate was

considered a sin, but there is no comparable unique proposition for (19), or for the related embedding sentence (22).

(22)   Rupert now knows where he can buy an Italian newspaper.

It seems that in the context of Rupert having set out to learn where he can buy an Italian newspaper, (22) can be made true by Rupert's knowing any of a number of independent propositions – any proposition that identifies some reasonably accessible place that will sell her an Italian newspaper promptly and at a reasonable price. When the domain contains multiple nearby sellers, we do not have exhaustivity even with respect to these sellers. For this reason, a domain restriction account, at least in this simple formulation, is not well suited to account for these sorts of data.

In spite of its failure to explain away mention-some readings entirely, the domain account may be able to account for many examples. In this way, it can thin out the space of apparent mention-some readings, leaving a more manageable space of examples.

6.3   Selecting for (Non-)Exhaustivity

So far, I have argued that we ought to keep mention-some readings in the semantics, and that an effort to explain them away completely in terms of domain restrictions will not be adequate. In light of this, making the mention-some/strongly exhaustive ambiguity an independent semantic ambiguity, as I did in Chapter 2, may be unavoidable. Even if we do accept this sort of stipulated ambiguity, however, we still ought to try to do something about the overgeneration problem. One conspicuous case of overgeneration relates to *'surprise'* (and perhaps other emotive factives). As argued in Section 3.2, there is reason to think that *'surprise'* has access to only a mention-some reading, and not a strongly exhaustive one. If this generalization is right, we ought to say something about the unavailability of an exhaustive reading with *'surprise'*.[7] One simple solution is to say that the presence or absence of the X operator is visible to the syntax, and that *'surprise'* selects for an argument that

---

[7]One common reply, which deserves brief mention, is to say that *'surprise'* embeds exclamatives rather than questions. This line of analysis is refuted to my satisfaction in Lahiri (2002), to which the reader is referred of the many problems with this approach.

lacks the X operator.[8] This solution is entirely stipulative, but the grammatical degree of freedom that it makes use of (selection), is at least a familiar and widely accepted one.

The suggestion that *'surprise'* selects for the absence of X handles *'wh'*-questions quite well. The empirical generalization seems to be that *'wh'*-clause that is the argument of *'surprise'* always receives a mention-some reading, and never a strongly exhaustive reading, and the theory says that a *'wh'*-question receives a strongly exhaustive reading if it has an X operator, and a mention-some reading otherwise, so selecting for the absence of an X operator is selecting for exactly the readings we want for embedded *'wh'*-questions. This sacrifices some uniformity, but the non-uniformity we get is limited: a predicate can select for one reading of an embedded question, or the other, or be indifferent between them.

If, as in Chapter 5, we attempt to handle other kinds of questions with the Q and X operators, this selectional account delivers some interesting (but mixed) results. Adopting the account of *'yes'*/*'no'* questions in Section 5.1, we derive an encouraging prediction: since the account in 5.1 says that *'yes'*/*'no'* questions *always* contain an X operator, we predict that if *'surprise'* selects for the absence of the X operator, then it should be unable to embed *'yes'*/*'no'* questions. Happily, this is what we find, as can be seen in the ungrammaticality of the examples in (26):

---

[8]More precisely, we should say that *'surprise'* selects for an argument that is not of the following form:

(23)

If there is an X operator buried someplace deep inside the argument of *'surprise'*, this should not be a problem. For example, in (24), the embedded question (25) can still receive a strongly exhaustive reading (resulting, on my account, from the presence of an X operator), and (24) is perfectly acceptable on this reading:

(24)   It surprised Anne that Rupert knew which students were in the band.

(25)   Which students were in the band?

That is, the feature that communicates the presence of the X operator can't cross multiple clause boundaries or otherwise percolate through large amounts of syntactic structure, so when *'surprise'* selects for the absence of this feature, it will only rule out Xs that are relatively high up in the structure of its argument.

(26)   a. *Whether William was a spy wouldn't surprise Anne.

b. *Whether William was a spy didn't surprise Anne.

c. *It wouldn't surprise Anne whether William was a spy.

d. *It didn't surprise Anne whether William was a spy.

e. *Anne wouldn't be surprised (by/at) whether William was a spy.

f. *Anne wasn't surprised (by/at) whether William was a spy.

Although this result for *'yes'/'no'* questions is tempting, we should not put too much stock in it. There are two reasons for this. First, to be satisfying, such a story would also have to account for the unacceptability of alternative questions with *'surprise'* (illustrated in (27)). It might turn out that the right generalization of my account to alternative questions makes alternative questions exhaustive, but, at present, I have not provided a well-justified account of alternative questions. For this reason, I am not well-equipped to explain the unacceptability of alternative questions with *'surprise'*.

(27)   a. *Whether William or Andrew was the spy wouldn't surprise Anne.

b. *Whether William or Andrew was the spy didn't surprise Anne.

c. *It wouldn't surprise Anne whether William or Andrew was the spy.

d. *It didn't surprise Anne whether William or Andrew was the spy.

e. *Anne wouldn't be surprised (by/at) whether William or Andrew was the spy.

f. *Anne wasn't surprised (by/at) whether William or Andrew was the spy.

A second, more serious, issue relates to concealed questions: the account in Section 5.3 builds all concealed questions with an X operator, so, if *'surprise'* selects for the absence of X, it we would expect it to be unable to embed concealed questions. This doesn't seem to be what we find: (28-a) appears to have a (perhaps slightly awkward) reading on which the arguments given in (28) and (29) are judged valid, indicating that a definite description in the subject position of *'surprise'* can receive a concealed question reading.

(28)   a.   The capital of Oregon surprised Alex.

b.   The capital of Oregon is Salem (and has been for the entire time period

under discussion).

     c.   Therefore, it surprised Alex that the capital of Oregon was Salem.

(29)   a.   Cordelia was surprised at the winner of the election.

     b.   The winner of the election was Michelle.

     c.   Therefore, Cordelia was surprised that Michelle was the winner of the election.

This is, of course, only a problem if we accept the data. There is perhaps something a little odd about (28) and (29), but my initial impression is that it is still good enough, and that, at least until some competing explanation is developed and justified, we ought to accept that *'surprise'* allows concealed question readings. In light of this, we either need a more subtle account of the behavior of *'surprise'* than the story on which it selects for the absence of X, or we need a different theory of concealed questions. I do not know how we should approach this issue, and will not pursue it further here.

'surprise', along with at least some other emotive factives, has two important peculiarities among responsive predicates: it only allows (or at least strongly favors) mention-some readings with embedded *'wh'*-questions, and it does not embed *'yes'*/*'no'* or alternative questions. It is extremely tempting to try to connect these two peculiarities, and the remarks above seem to offer an indication of how this might be done, but they do not provide anything that could be called a satisfying theory, and I do not know how to refine them into one.

It also not clear how (or whether) the mention-some behavior associated with *'surprise'* is related to the way that in other contexts mention-some readings are readily available for some questions, but unavailable or marginal for others (as discussed below in Section 6.4). In any case, I know no way of linking the two, although I will consider a distinct treatment of the second tendency below.

6.4 Mention-Some Readings as a Scope Effect

6.4.1 A Possible Generalization

Once domain effects and Gricean reasons for giving an incomplete answer are considered, the availability of genuine mention-some readings starts to look fairly limited, although, as we have seen, a number of important examples remain. Putting aside *'surprise'* and similar predicates, the clearest class of examples involves questions embedded under responsive embedders where mention-some and exhaustive readings are easy to distinguish intuitively, such as *'know'*. In some such embedding sentences, the truth-conditions do not require knowledge of any exhaustive answer (even a narrowly domain-restricted one). The following examples are representative:

(22)   Rupert now knows where he can buy an Italian newspaper.

(30)   Anne knows where there's a 24-hour pharmacy.

(31)   Red knows where William might have hidden the murder weapon.

(32)   Alex knows who can get our wireless working.

In each of these examples, we can imagine a context in which there is more than one way to make the sentence true – more than one proposition the knowing of which is sufficient to ensure knowledge of the question. To confidently utter (22), for example, I don't need to be sure that Rupert knows the best place, the most salient place, the place that somebody else mentioned before, or anything of the sort. If there are a dozen reasonably accessible places that could sell Rupert an Italian newspaper, then Rupert's knowing of any one of them is sufficient to make (22) true. Unlike the domain (or specificity) cases we saw before, the one Rupert knows about need not be a specific one that's privileged over all the others by some contextual factor. On a somewhat fuzzier note, the truth-conditional intuitions for the examples above feel different, and more open-ended, than the domain examples we saw before. For this reason, I think that the above examples will probably have to be admitted as genuine mention-some readings.

Another important class of mention-some examples that we've seen involves pairs of questions that ought to have equivalent or near-equivalent meanings (at least judging by their abstracts), but one of which favors mention-some answers in dialog more than the other. This has already been seen in the contrast between (4) (favors exhaustive answers) and (5) (favors mention-some answers), and something similar is seen in the contrast between (33) and (34).

(4)   Who has leprosy?

(5)   Who are some people with leprosy?

(33)   What netbooks have decent battery life?

(34)   What are some netbooks with decent battery life?

Here, we see (34) is a question completely resolved by an incomplete list of products meeting the description, while an incomplete list is, intuitively, not a wholly resolving answer to (33). With both this pair and (4) and (5), domain effects or pragmatic notions of what constitutes a good enough answer do not seem well-suited to explain the observed contrast. We should expect (33) and (34) to involve (nearly) equivalent abstracts, and likewise for (4) and (5). This suggests that the mention-some readings in (34) and (5) are a genuine mention-some readings of the sort we saw with embedded questions. This also suggests that the contrasts seen should be attributed, at least in part, to grammatical differences between the sentences involved.

My suggestion is that the major similarity among the mention-some examples we have seen is the presence of items that can, loosely, be thought of as existentials, including indefinites and possibility modals.[9] The presence or absence of an existential

---

[9]The idea that mention-some readings are associated with modals appears have been recognized for some time, although I am not aware of any place that it is addressed in the literature. It was brought to my attention separately by Yael Sharvit and Ivano Caponigro. This generalization extends the idea about modals to account for (30) and (34), which don't involve modals, but do provide other opportunities for existentials to engage in scope interactions.

One major class of apparent exceptions to this generalizations, discussed by Lahiri (2002) and Hintikka (1976), among others, is infinitive embedded questions, such as (35):

(35)   Alex knows how to get to the station.

is probably not the only factor in the availability and unavailability of mention-some readings, but the apparent association is interesting enough to be worth exploring. Below, I sketch one possible way of accounting for mention-some readings as a scope effect.

6.4.2   Sketch of the Scope Account

My proposed scope account begins with the following assumption: all *'wh'*-questions are built with all the pieces needed to derive strong exhaustivity. That is, the X operator is obligatory in *'wh'*-question formation, so the configuration used to derive mention-some readings in Chapter 2 is disallowed. Instead, every *'wh'*-question is formed by the successive application of X and then Q to an abstract, as in (36):

(36)

$$Q \quad X \quad \text{abstract}$$

$$\ldots$$

This would ordinarily make all questions exhaustive. On this account, the source of mention-some readings is that certain existentials are allowed to take scope between Q and X, as schematized in (37):
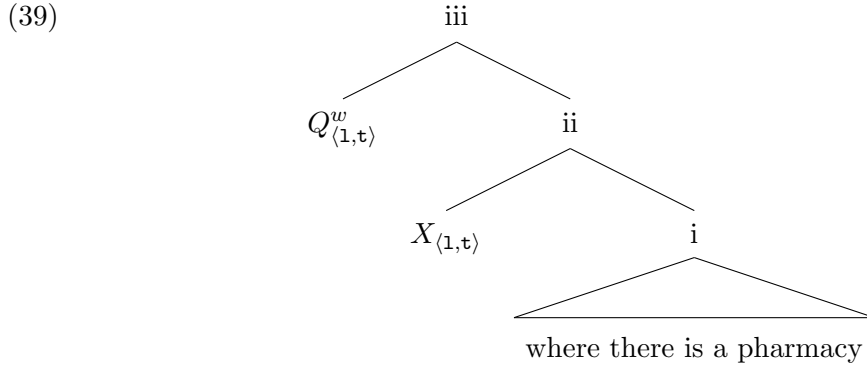
(37)

$$Q \quad \exists^v \quad X \quad \text{abstract}$$

$$\ldots e_v \ldots$$
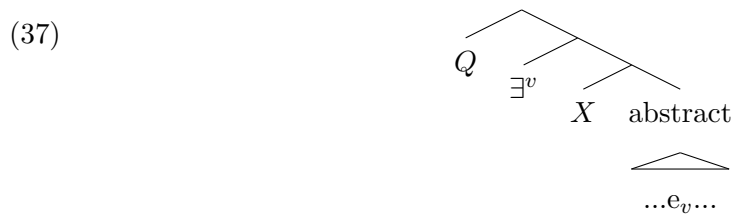
Scoping an existential between Q and X in this way gives us something that could plausibly be called a mention-some reading.

These embedded questions can receive mention-some readings even without an overt modal or existential. I do not know how to analyze these sentences, but I note that their meaning often seems to have a modal component ((35) means that Alex knows one possible way of getting to the station, or knows how one can get to the station). If the notion that sentences like (35) involve a covert possibility modal can be made precise, then they can be treated as another case of this generalization, but more work is needed to sort out the details.

To begin, consider (38).

(38)   Where is there a pharmacy?

Let's first remember how we derive a strongly exhaustive reading for (38). If we call the type of spatial locations 'l', the application of X and Q to an abstract to give us (38) is schematized by (39), where the meanings of the various nodes are given by (40):

(39)

$$iii$$

$$Q^w_{\langle l,t\rangle} \qquad ii$$

$$X_{\langle l,t\rangle} \qquad i$$

where there is a pharmacy

(40)   i : $\lambda z_l \exists x(pharmacy(w)(x) \wedge locationof(w)(x)(z))$

$X_{\langle l,t\rangle}$: $\lambda \gamma_{\langle l,t\rangle} \lambda \delta_{\langle l,t\rangle}(\gamma = \delta)$

ii : $\lambda \delta_{\langle l,t\rangle}(\lambda z_l \exists x(pharmacy(w)(x) \wedge locationof(w)(x)(z)) = \delta)$

$Q_{\langle l,t\rangle}$: $\lambda \alpha_{\langle s,\langle\langle l,t\rangle,t\rangle\rangle} \lambda p_{\langle s,t\rangle} \exists \beta_{\langle l,t\rangle}(p = \lambda w'_s(\alpha(w')(\beta)))$

iii : $\lambda p_{\langle s,t\rangle} \exists \beta_{\langle l,t\rangle}(p = \lambda w'_s(\lambda z_l \exists x \begin{pmatrix} pharmacy(w')(x) \wedge \\ locationof(w')(x)(z) \end{pmatrix} = \beta))$

That is, the abstract ((i) of (39)) will be the set of all locations that are the location of at least one pharmacy. If we apply X and then Q to this, as shown in (40) we get a strongly exhaustive reading for the question – one that is answered fully and truthfully only by a complete identification of all the locations (up to the limits of contextual restrictions on, for example, the granularity with which locations are individuated and the spatial region under consideration) at which there are pharmacies.

221

This strongly exhaustive reading is probably available, but, as already noted, the most prominent reading of (38) is a mention-some reading. To derive the mention-some reading, we need to give *a pharmacy* wide scope with respect to X, as schematized by (41):

(41)
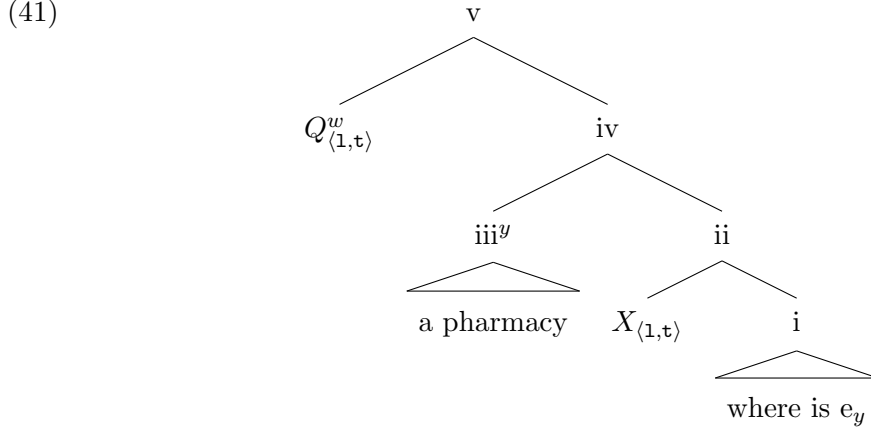
Now let's see what meaning we derive for (41):

(42) $i : \lambda z_1(locationof(w)(y)(z))$

$X_{\langle 1,t \rangle} : \lambda\gamma_{\langle 1,t \rangle}\lambda\delta_{\langle 1,t \rangle}(\gamma = \delta)$

     $ii : \lambda\delta_{\langle 1,t \rangle}(\lambda z_1(locationof(w)(y)(z)) = \delta)$

     $iii : \lambda S_{\langle e,t \rangle}\exists x(pharmacy(w)(x) \wedge S(x))$

     $iv : \lambda\delta_{\langle 1,t \rangle}\exists x(pharmacy(w)(x) \wedge \lambda z_1(locationof(w)(y)(z)) = \delta)$

$Q_{\langle 1,t \rangle} : \lambda\alpha_{\langle s,\langle\langle 1,t \rangle,t \rangle\rangle}\lambda p_{\langle s,t \rangle}\exists\beta_{\langle 1,t \rangle}(p = \lambda w'_s(\alpha(w')(\beta)))$

     $v : \lambda p_{\langle s,t \rangle}\exists\beta_{\langle 1,t \rangle}(p = \lambda w'_s \exists x \left(\begin{array}{c} pharmacy(w')(x)\wedge \\ \lambda z_1(locationof(w')(y)(z)) = \beta \end{array} \right)$

Let's go through the intuitive meaning of the main nodes of (42) step by step. First, for (i), we have the property of being the location of whatever y denotes, where y is the trace variable associated with the generalized quantifier (iii). Applying X to (i), we get (ii), which is just the packaging of the meaning of (i) inside a singleton set. Next we have the interesting stage, where (iii) is quantified into (ii) for the variable y, producing (iv). The meaning of (iv) is the property of being the set of all

the locations of *some* pharmacy. Quantifying in and then applying X would have given us instead the property of being the set of all locations at which there were pharmacies located, leading to strong exhaustivity. That is, while the low scope for '*a pharmacy*' would have produced a set containing only one set of locations (the set of all locations of pharmacies), we now have a set containing as many distinct sets of locations (one set for each pharmacy, assuming no two distinct pharmacies occupy the same space). The fact that the extension of (iv) can contain more than one set of locations is what will give us a mention-some reading. Finally, Q binds the world variable w to give us (v).

If (v) expresses the answerhood conditions of one reading of the question '*where is there a pharmacy?*', what propositions are potential answers to this question? To be in the set, the proposition must identify some set as the set of locations of a pharmacy, without specifying which pharmacy that is (informally, it must say of a set β, 'There is some pharmacy x such that any location is in β iff it is a location of x'). Assuming that a given context will fix a granularity with which we divide space into locations, and that physical objects like pharmacies typically occupy a single continuous spatial region, such a set will typically include one location (or a perhaps collection of locations of varying sizes all overlapping at the region of space occupied by the pharmacy). Thus, every answer will basically identify some location as the location of a pharmacy. It will not identify the specific pharmacy, nor will it identify the locations of all pharmacies. These are answerhood conditions that we could reasonably describe as 'mention-some'. The question is satisfied by an answer that tells us a location that is the location of a pharmacy, and knowing the location of a single pharmacy will be enough to make the sentence embedding the question under *know* true.

Now let's consider a more concrete example of the kinds of propositions we have in the answer set expressed by node (v). For any set of locations β, the proposition 'there is a pharmacy x such that β is the set of all locations of x' will be in the answer set. In general, contextual restrictions will determine the level of granularity with respect to which 'the set of all locations of x' is understood. Suppose the contextually provided level of granularity is street addresses. With this background assumption, let $\beta = \{314 \text{ Elm St.}\}$. Then the proposition 'there is a pharmacy x such that $\{314 \text{ Elm St.}\}$ is the set of all locations of x' will be in the answer set.

Assuming *'location'* is understood as *'street address'*, this will be understood as 'there is a pharmacy x such that {314 Elm St.} is the set of all street addresses of x'. If there is in fact a pharmacy located at 314 Elm St., and that pharmacy does not span multiple addresses, then this will be a true answer. The answer set will likewise include analogous propositions for every other address.

Something similar can be done in the modal case, if possibility modals are understood as existential quantifiers over worlds, and, like nominal existentials, are allowed to scope between X and Q. Consider, for example (43):

(43)   Who can fix the printer?

When *'can'* scopes over X, the order of combinations for (43) is represented by (44), which is interpreted as in (45)

(44)
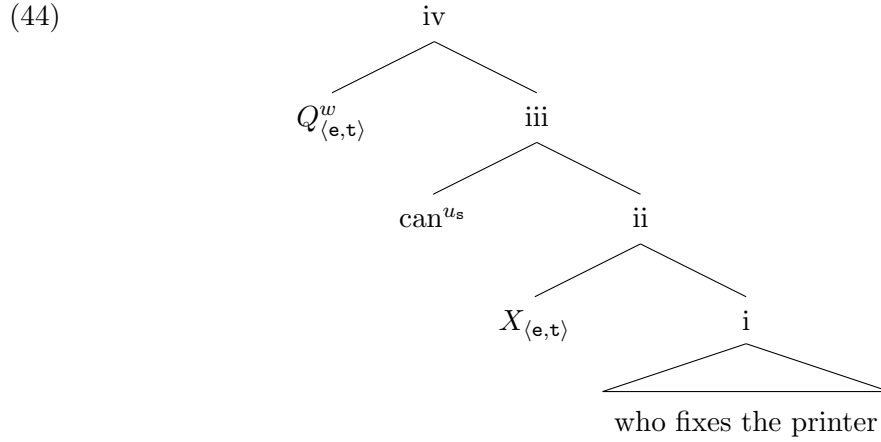
(45)   i : $\lambda x(person(u_{\mathsf{s}})(x) \wedge fixes(u_{\mathsf{s}})(printer)(x))$

$X_{\langle \mathsf{e},\mathsf{t}\rangle}$: $\lambda\gamma_{\langle \mathsf{e},\mathsf{t}\rangle}\lambda\delta_{\langle \mathsf{e},\mathsf{t}\rangle}(\gamma = \delta)$

ii : $\lambda\delta_{\langle \mathsf{e},\mathsf{t}\rangle}(\lambda x(person(u_{\mathsf{s}})(x) \wedge fixes(u_{\mathsf{s}})(printer)(x)) = \delta)$

can : $\lambda p_{\langle \mathsf{s},\mathsf{e}\rangle}\exists u'_{\mathsf{s}}(can(w,u') \wedge p(u'))$

iii : $\lambda\delta_{\langle \mathsf{e},\mathsf{t}\rangle}\exists u'_{\mathsf{s}}(can(w,u') \wedge \lambda x \left(\begin{array}{c} person(u')(x)\wedge \\ fixes(u')(printer)(x) \end{array} \right) = \delta)$

$Q_{\langle \mathsf{e},\mathsf{t}\rangle}$: $\lambda\alpha_{\langle \mathsf{s},\langle\langle \mathsf{e},\mathsf{t}\rangle,\mathsf{t}\rangle\rangle}\lambda p_{\langle \mathsf{s},\mathsf{t}\rangle}\exists\beta_{\langle \mathsf{e},\mathsf{t}\rangle}(p = \lambda w'_{\mathsf{s}}(\alpha(w')(\beta)))$

$$\text{iv}: \lambda p \exists \beta_{\langle \mathbf{e,t} \rangle} (p = \lambda w'_{\mathbf{s}} \exists u'_{\mathbf{s}} \left(\lambda x \left(\begin{array}{c} can(w', u') \wedge \\ person(u')(x) \wedge \\ fixes(u')(printer)(x) \end{array} \right) = \beta \right))$$

The interesting step above is in the composition of node (iii) – a set has the property given by (iii) iff there is some suitably accessible world u' such that it is the set of all people who fix the printer in u'. Like the case with a nominal existential, but unlike the case without an existential scoped over X, a choice of value for the free world variable (here w) does not guarantee that there is a unique argument that makes (iii) true. There may be many, and this diversity of alternatives will again give rise to a mention-some effect. In (iv) we see the resulting answerhood conditions: a proposition is an answer to (43) (on the reading where *'can'* scopes over X) iff it identifies some set of people β as having the property that there is some suitable world in which β is the set of all people who fix the printer. Assuming that a particular occasion is talked about, so that the printer is only being fixed once, many complete and satisfying answers will identify one person as a person who can fix the printer, so again we have something that could be described as a mention-some reading, resulting in this case from a possibility modal.

There is one important issue with the analysis of (43) that needs to be mentioned: there are, we must presume, possible worlds in which the printer goes unfixed. Thus, it looks (46) should be an adequate answer for (43), which runs contrary to our judgments.

(46)   It could be that nobody fixes the printer.

This issue can probably be resolved by elaborating the semantics of *'can'*. On the reading seen here, $can(w, u')$ will be true iff the world u' is accessible from the world w (in the sense that there is some possible course of action in w that will give rise to the state of affairs associated with u'), and u' is a world in which the goals under discussion are achieved. In (43), the goal under discussion is presumably the fixing of the printer, so u' should be a world in which somebody *does* fix the printer. Thus, the existential quantification over worlds will be a quantification over those worlds in which somebody fixes the printer (under the goal-satisfaction restriction contributed by *'can'*). This will exclude worlds in which nobody fixes the printer, so there will
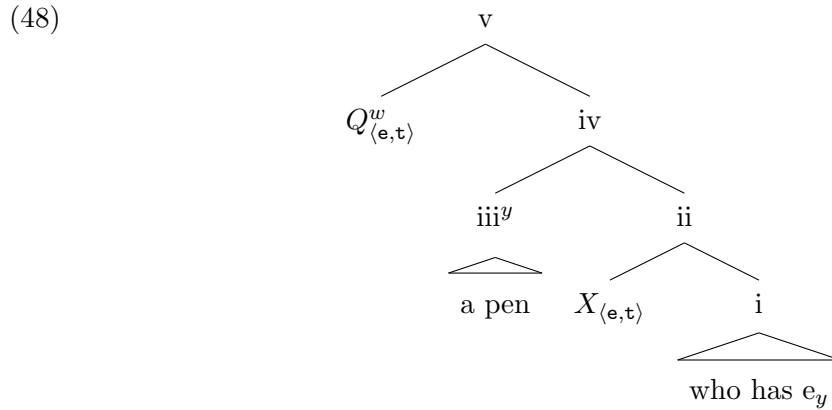
225

be no world u' such that $can(w, u')$ and nobody fixes the printer in u', making (46) unacceptable as an answer for (43).

6.4.3 The Scope Account and Other Quantifiers

The scope account sketched above postulates a new scope position, but so far we've only seen examples of existentials scoping into this position. To succeed, this account would need to offer some account of how other quantifiers interact with this position, or to explain why placing other quantifiers in this position is forbidden. I do not know what the prospects of such a story are, but I would like to note briefly that some other quantifiers produce answerhood conditions that are sufficiently bizarre that the reading would be arguably ruled out as useless. First, I want a convenient question into which different quantifiers can be easily substituted:

(47)   Who has a pen?

If we analyze (47) by scoping 'a pen' between Q ad X, we get (48), interpreted as in (49):

(48)

(49)   i : $\lambda z_{\mathbf{e}}(person(w)(z) \wedge has(w)(y)(z))$

$X_{\langle 1, t \rangle} : \lambda \gamma_{\langle \mathbf{e}, t \rangle} \lambda \delta_{\langle \mathbf{e}, t \rangle} (\gamma = \delta)$

ii : $\lambda \delta_{\langle \mathbf{e}, t \rangle} (\lambda z_{\mathbf{e}}(person(w)(z) \wedge has(w)(y)(z)) = \delta)$

iii : $\lambda S_{\langle \mathbf{e}, t \rangle} \exists x (pen(w)(x) \wedge sg(x) \wedge S(x))$

iv : $\lambda \delta_{\langle \mathbf{e}, t \rangle} \exists x (pen(w)(x) \wedge sg(x) \wedge \lambda z_{\mathbf{e}}(person(w)(z) \wedge has(w)(y)(z)) = \delta)$

$$Q_{\langle e,t\rangle} : \lambda\alpha_{\langle s,\langle\langle e,t\rangle,t\rangle\rangle}\lambda p_{\langle s,t\rangle}\exists\beta_{\langle e,t\rangle}(p = \lambda w'_{s}(\alpha(w')(\beta)))$$
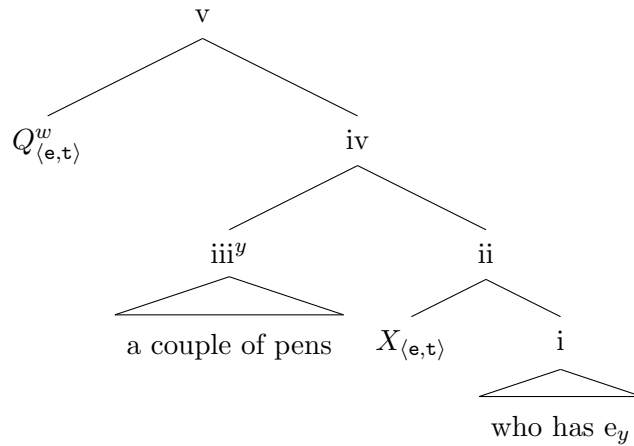
$$\text{v} : \lambda p\exists\beta(p = \lambda w'_{s}\exists x \left(\begin{array}{c} pen(w)(x) \wedge sg(x)\wedge \\ \lambda z_{e}(person(w)(z) \wedge has(w)(y)(z)) = \beta \end{array} \right))$$

That is, we get the set of propositions that identify some set of entities β as having the property that there is some pen x such that β is the set of people to whom that pen belongs. This is our usual mention-some reading.

Now, let's substitute *'a couple of'* for *'a'*:

(50)   Who has a couple of pens?

(51)

(52)   i : $\lambda z_{e}(person(w)(z) \wedge has(w)(y)(z))$

$X_{\langle 1,t\rangle} : \lambda\gamma_{\langle e,t\rangle}\lambda\delta_{\langle e,t\rangle}(\gamma = \delta)$

ii : $\lambda\delta_{\langle e,t\rangle}(\lambda z_{e}(person(w)(z) \wedge has(w)(y)(z)) = \delta)$

iii : $\lambda S_{\langle e,t\rangle}\exists x(pen(w)(x) \wedge 2(x) \wedge S(x))$

iv : $\lambda\delta_{\langle e,t\rangle}\exists x(pen(w)(x) \wedge 2(x) \wedge \lambda z_{e}(person(w)(z) \wedge has(w)(x)(z)) = \delta)$

$Q_{\langle e,t\rangle} : \lambda\alpha_{\langle s,\langle\langle e,t\rangle,t\rangle\rangle}\lambda p_{\langle s,t\rangle}\exists\beta_{\langle e,t\rangle}(p = \lambda w'_{s}(\alpha(w')(\beta)))$

$$\text{v} : \lambda p\exists\beta(p = \lambda w'_{s}\exists x \left(\begin{array}{c} pen(w)(x) \wedge 2(x)\wedge \\ \lambda z_{e}(person(w)(z) \wedge has(w)(x)(z)) = \beta \end{array} \right))$$

That is, we get the set of propositions that identify some set of people as the set of all owners of a plural collection of two pens, so, for example, the answer set will

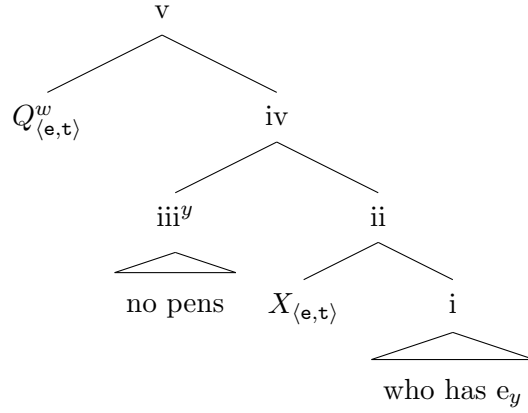include the proposition described in (53-a), which is, for normal purposes, virtually equivalent to (53-b).

(53) a. There are two pens x_1 and x_2 such Janna owns x_1 and x_2 and nobody else owns both x_1 and x_2

b. Janna has (at least) two pens.

That is, we get something like a mention-some reading, which is a plausible reading for (50), so again we get plausible results.

Now, let's turn to the quantifier *'no'*:

(54) Who has no pens?

(55)

(56) i : $\lambda z_{\mathbf{e}}(person(w)(z) \wedge has(w)(y)(z))$

$X_{\langle 1,\mathbf{t}\rangle} : \lambda\gamma_{\langle\mathbf{e},\mathbf{t}\rangle}\lambda\delta_{\langle\mathbf{e},\mathbf{t}\rangle}(\gamma = \delta)$

ii : $\lambda\delta_{\langle\mathbf{e},\mathbf{t}\rangle}(\lambda z_{\mathbf{e}}(person(w)(z) \wedge has(w)(y)(z)) = \delta)$

iii : $\lambda S_{\langle\mathbf{e},\mathbf{t}\rangle}\neg\exists x(pen(w)(x) \wedge S(x))$

iv : $\lambda\delta_{\langle\mathbf{e},\mathbf{t}\rangle}\neg\exists x(pen(w)(x) \wedge \lambda z_{\mathbf{e}}(person(w)(z) \wedge has(w)(y)(z)) = \delta)$

$Q_{\langle\mathbf{e},\mathbf{t}\rangle} : \lambda\alpha_{\langle\mathbf{s},\langle\langle\mathbf{e},\mathbf{t}\rangle,\mathbf{t}\rangle\rangle}\lambda p_{\langle\mathbf{s},\mathbf{t}\rangle}\exists\beta_{\langle\mathbf{e},\mathbf{t}\rangle}(p = \lambda w'_{\mathbf{s}}(\alpha(w')(\beta)))$

v : $\lambda p\exists\beta(p = \lambda w'_{\mathbf{s}}\neg\exists x\left(\begin{array}{c}pen(w)(x)\wedge\\ \lambda z_{\mathbf{e}}(person(w)(z) \wedge has(w)(y)(z)) = \beta\end{array}\right))$

228

That is, p is in the answer set iff there is some set β such p is the proposition that there is no pen x such that β is the set of people who have x. Most propositions of this form will be so uninformative as to be trivial: it will suffice to identify any set of people β such that there is no pen (or plurality of pens) that every member of β owns. Questions resulting from scoping *'no'* into the target position will generally be far too easy to answer, so we might hope that some (synchronic or diachronic) force that agitates against near-triviality might rule out this ruling. This is, of course, far from a complete story, in any case, I am not aware of any evidence that the reading sketch above is available for (54).

As one more example, consider a universal:

(57)   Who has every pen?

(58)
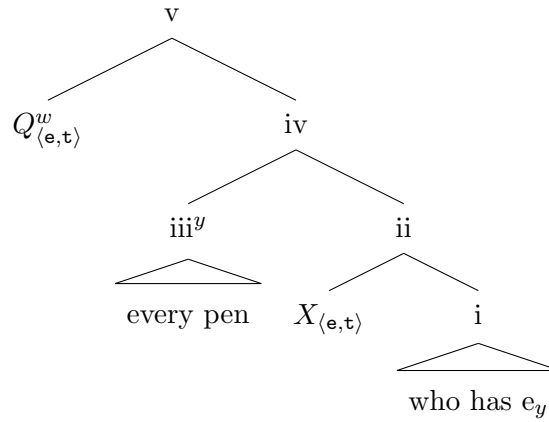
(59)   i : $\lambda z_{\mathsf{e}}(person(w)(z) \wedge has(w)(y)(z))$

$X_{\langle 1,\mathsf{t}\rangle} : \lambda\gamma_{\langle\mathsf{e},\mathsf{t}\rangle}\lambda\delta_{\langle\mathsf{e},\mathsf{t}\rangle}(\gamma = \delta)$

ii : $\lambda\delta_{\langle\mathsf{e},\mathsf{t}\rangle}(\lambda z_{\mathsf{e}}(person(w)(z) \wedge has(w)(y)(z)) = \delta)$

iii : $\lambda S_{\langle\mathsf{e},\mathsf{t}\rangle}\forall x((pen(w)(x) \wedge sg(x)) \to S(x))$

iv : $\lambda\delta_{\langle\mathsf{e},\mathsf{t}\rangle}\forall x \left(\begin{array}{c} (pen(w)(x) \wedge sg(x)) \to \\ \lambda z_{\mathsf{e}}(person(w)(z) \wedge has(w)(y)(z)) = \delta \end{array} \right)$

$Q_{\langle\mathsf{e},\mathsf{t}\rangle} : \lambda\alpha_{\langle\mathsf{s},\langle\langle\mathsf{e},\mathsf{t}\rangle,\mathsf{t}\rangle\rangle}\lambda p_{\langle\mathsf{s},\mathsf{t}\rangle}\exists\beta_{\langle\mathsf{e},\mathsf{t}\rangle}(p = \lambda w'_{\mathsf{s}}(\alpha(w')(\beta)))$

v : $\lambda p\exists\beta(p = \lambda w'\forall x \left(\begin{array}{c} (pen(w)(x) \wedge sg(x)) \to \\ \lambda z(person(w)(z) \wedge has(w)(y)(z)) = \beta \end{array} \right))$

Where (54) was to easy to answer, (57) will frequently be unanswerable. Note that a proposition p is in this answer set if and only if there is a set β such that, for every pen x, β is the set of people who possess x. That is, this sentence will only have a true answer if there is some set β such that every pen is owned by every member of β, and by nobody else. That is, this answer set will typically not contain *any* true propositions. The difficulty of answering the question on this reading might exclude it. In any case, this reading does not seem to be available for (57), so we will have to exclude it somehow. Of course, (57) has two other readings: the normal strongly exhaustive reading on which it asks which set is the set of people who have all pens (and on which *'there isn't anybody who has every pen'* is a possible answer), and the pair-list reading, on which an answer identifies, for every pen, who has that pen. Since the pair-list reading also seems to be associated with a wide-scope quantifier, it would be desirable to unite it with the account of the mention-some reading discussed here, but, although this may be possible on some theory, it is not what the present scope account of mention-some readings delivers.

6.5 Concluding Remarks

In this chapter, I've looked at some possible ways of accounting for mention-some readings that might improve on the account developed in Chapter 2, on which every *'wh'*-question has access to both strongly exhaustive and mention-some readings. I have argued that simple attempts to explain away mention-some readings with domain effects and pragmatic considerations do not seem to be adequate for all cases of embedded mention-some questions, although they do limit the range of cases in which there is clear evidence that mention-some readings are available. I have looked at the possibility of allowing some predicates (like *'surprise'*) to select for mention-some readings, and found that it is adequate as far as *'wh'*-questions are concerned, but problematic as an account of the full range of facts for question-embedding under these predicates. I have also explored the possibility of accounting for mention-some readings as a result of existentials taking wide scope with respect to the X operator, again with mixed results: the judgments for the correlation between existentials and mention-some readings are somewhat slippery, and it is not clear why the proposed

scope position should be available only to existentials.[10]

Another problem with these last two accounts is that each one accounts for a different pattern, and the two accounts appear to make incompatible assumptions. The selection account handles differences in the availability of mention-some and exhaustive readings for different embedders (by allowing embedders to select for the presence or absence of an X operator), but doesn't say anything about why some questions more naturally receive mention-some readings. The scope account, on the other hand, offers an explanation of why mention-some readings are available for some questions, but apparently unavailable for others, but it says nothing about why questions that do not normally have access to mention-some readings receive them freely when combined with embedders like *'surprise'*. Unfortunately, the assumptions of these two accounts seem not to be cleanly compatible. The selection account assumes that mention-some readings result from the absence of an optional X operator, while the scope account makes the X operator obligatory, and makes wide-scope existentials the *only* source of mention-some readings, which in turn makes it difficult to explain the availability of mention-some readings without a scopable existential in the case of *'surprise'* embedding.

Straightforward attempts to combine the two accounts are not terribly elegant. On the most natural approach, we would have to make the X operator obligatory in some cases and prohibited in others, and perhaps optional in still others. We would say that *'surprise'* selects for the absence of the X operator (so that mention-some readings are the only option with *'surprise'*) while *'know'* requires an X operator (so that mention-some readings cannot be derived as they are for *'surprise'*, but only arise when we scope an existential over the X operator). This would, awkwardly, mean that the mention-some readings found under *'know'* and under *'surprise'* were not really the same, but were the result of the grammar's having two completely different ways of deriving mention-some answer-sets. For unembedded questions, we would most likely declare the X operator to be optional, although making it obligatory would be another credible option. (If X were obligatory in unembedded questions, however, we would effectively be declaring that the embedded questions seen with *'surprise'* were not really the same thing as most canonical 'questions' found elsewhere in the grammar, but were in fact a special class of *'wh'*-clause asso-

---

[10]It is also not clear why this scope would only be available with *'wh'*-questions.

ciated only with *'surprise'* and similar embedders.) Besides the sense of inelegance at having two separate systems for deriving mention-some readings, this hybrid account also combines all the weaknesses already observed for the selection account and the scope account. In light of this, more work is needed, although I hope that the two accounts I have proposed at least suggest promising lines of attack.

CHAPTER 7

Review and Open Issues

> But questions, I've learned since, can be like ocean currents.
> Wade in a little too far and they can carry you away.
>
> <div align="right">Hohn (2011)</div>

7.1 Review of Main Themes

The main project of this dissertation has been to sketch a relatively uniform account of question-embedding by combining elements from Groenendijk and Stokhof (1984), Hamblin (1973), Lahiri (2002), and Egré and Spector (2007), and then to see how far this relatively simple and well behaved account can be pushed, and how it breaks.

The baseline theory, described in Chapter 2, assumes one major ambiguity for 'wh'-questions: they can receive a strongly exhaustive reading or a mention-some reading. In both cases, the question's semantic contribution is an answer-set, but the nature of the answer-propositions included is different in the two cases. Both types of answer-sets are derived by the same compositional machinery, with the sole difference that the grammar includes the X operator in the derivation of strongly exhaustive answer-sets, but not the derivation of mention-some answer-sets. Both types of answer-sets are also subject to the same embedding rule: to 'know' (or 'forget', or 'be surprised at', or 'be certain of' ...) a question is to 'know' (or 'forget' ...) at least one proposition in its answer-set.

In Chapter 3, having set out this baseline account, I explored a number of problems and possible revisions and extensions. Two categories of problems that I dismissed were so-called weakly exhaustive readings, and variations in the granularity of 'where' questions. I argued that both of these could be handled with independently motivated considerations about domain effects and intensionality. I more tentatively endorsed a similar response to arguments (seen in Lahiri (2002), and elsewhere) that

individual embedders needed to be arbitrarily marked for whether they privileged true answers, arguing, following Egré and Spector (2007), that the problematic effects (with, for example *'tell'*) were implicatures that, although poorly understood, were not peculiar to question-embedding. Here, unlike the cases of weak exhaustivity and granularity effects, there are some important issues that I do not begin to address, but the scope of the problem seems modest enough that I decided I was justified in setting it aside.

I also briefly discussed two more serious issues: pair-list readings with universals, and homogeneity effects. I endorsed the standard view that the former involved some special wide scope for the universal with respect to the question, and briefly discussed two implementations of this idea that might be attached to the baseline account. For homogeneity effects, I discussed the possibility of a pragmatic explanation, but also discussed a revision that wrote a general homogeneity presupposition into the embedding rule.

In Chapter 4, I turned to a less widely known, but more serious, problem. This is the problem was the general inadequacy of accounts that reduce the question-embedding use of a responsive predicate to its propositional use. I first sketched a very broad reducibility criterion, and then argued that certain examples (involving mention-some readings of questions) showed that *'know'* and possibly *'forget'* failed to meet this criterion. I then reviewed various responses, and argued that a response that radically revised our approach to question-embedding and embedder meaning was probably required. I advanced one account along these lines, the Twin Relations account, as an example of a way to handle the core non-reducibility data while predicting general (non-reductive) constraints on the relationship between the question-embedding and propositional uses of all responsive predicates. The Twin Relations account is not the only viable line of attack, but the way that it allows non-reducibility while at the same time ruling out arbitrary pairings of unrelated question-oriented and propositional behaviors provides an example of the kind of balance we should try to achieve in confronting these problems.

Chapter 5 was devoted to some theoretical housekeeping and to an exploration of the issues raised by non-*'wh'* questions. I argued that question-semantics that I had already adopted for *'wh'*-questions could be applied without modification to concealed questions and *'yes'*/*'no'* questions, and discussed the problems that arose

when trying to generalize the account to alternative questions.

Finally, in Chapter 6, I revisited some problems with mention-some readings. Although the account in Chapter 2 makes mention-some readings equally available in all embedded *'wh'*-questions, the data seem more complex. The main difficulty is that there are two generalizations that cut in different directions, and that the empirical robustness of the second generalization is unclear.

The first generalization is that certain embedders, like *'surprise'*, seem to always select mention-some readings, to the point of being unable to embed a *'wh'*-question with a strongly exhaustive interpretation. These predicates can apparently get a mention-some reading with *any* embedded *'wh'*-question. I briefly discussed the prospects of an account that allowed *'surprise'* to select for mention-some questions, and the possibility of connecting this behavior with other selectional peculiarities of *'surprise'*, but I was not able to offer any solid conclusion, in part because of my difficulties in integrating an account of alternative questions into my theory of question-meaning.

The second generalization is that certain questions, in both embedded and un-embedded uses, get mention-some readings fairly easily, while others (except when embedded under predicates like *'surprise'*) either cannot get them at all, or get them only with difficulty. Questions of the former kind often involve some kind of modality or existential quantification, and a few examples can even be constructed in which the inclusion of an overt existential seems to make a mention-some reading (more) available in a case where we would not expect it to have any substantial impact on the meaning, but the judgments are often less clear than we might like. If existentials are the source of mention-some readings, there is at least a reasonably promising way of adjusting the account in Chapter 2 to account for this: scoping an existential over the X operator was shown to produce a mention-some reading, so, if we were to make the X operator obligatory, this would become the only source of mention some readings. The major unresolved issues for this account include the inability of other quantifiers to scope into this position, and the mention-some facts for *'surprise'*.

This dissertation has been an exploration of the strengths and weaknesses of one mostly uniform approach to questions and question embedding. It has argued that this approach fairs better than is generally assumed (in particular, that it does not need to be extended to include weak exhaustivity, and probably does not need

special provisions to deal with predicates like *'tell'*), but has also shown that this approach, along with many others, faces serious problems, especially in its adoption of a reductive account of embedding. It has also highlighted a number of more problematic areas.

7.2 Open Issues and Future Directions

There are, of course, many unresolved issues in the semantics of questions. I highlight some of the more important ones below.

7.2.1 Strong-ish Exhaustivity

Consider the following example (due to Kratzer (2006)):

(1)   The check-marks indicate which students passed.

The judgment reported by Kratzer (2006) is that, for (1) to be true, the check-marks must be associated with the passing students, not the non-passing students. In principle, on the assumption that it is clear that the larger list to which the check-marks are applied is the list of all students, checking the non-passing students and checking the passing students should provide the same information (assuming the function of the check-marks is made clear), and in particular one should be as good as the other at providing a strongly exhaustive answer. As Kratzer notes for a similar example, weak exhaustivity is no help: the check-marks must mark only the students who passed, and cannot inaccurately mark any non-passing students. The status of (1) within mainstream accounts of question semantics and question embedding is thus something of a mystery, and it is one that requires further exploration.

I do not know how to address this problem. It bears some resemblance to certain examples related to apparently weakly exhaustive readings found with plural subjects of *'know'* Lahiri (2002), and also brings to mind the 'no false belief' requirement for knowledge discussed in Chapter 4, and the distinction between what is said and what is communicated in the semantics for *'tell'* proposed in Heim (1994). I suspect that the right answer is to try to bring these phenomena together, perhaps in combination with a notion of proposition that is more fine-grained than mere world-sets, but an

exploration of these issues would be beyond the scope of the present discussion. In any case, I think that this example presents one of the most interesting current problems in question semantics.

7.2.2 Selectional Issues

There are many issues in the selectional behavior of different embedders that invite further investigation. Why can *'be certain'* embed questions, while *'believe'* cannot? Why is it that most factive propositional attitudes can embed questions, while only a few non-factive ones can? Why do so many non-factive responsive predicates, but so few factive ones, require a prepositional particle when embedding a question (e.g. *'be certain of'*, *'agree on'*)? Why is it that (in English) concealed questions are available as objects for responsive predicates, and (probably) the *'depend'*/*'determine'* class of rogative predicates, but not for the rogative predicate *'wonder'*? Why are the emotive factives (like *'surprise'*) unable to embed *'yes'*/*'no'* and alternative questions?

Almost all of these questions have been discussed in the literature. Few, if any, have received satisfying and conclusive resolutions. I have, for the most part, ignored these issues here, but they are relevant, and they certainly deserve more attention than I have been able to give them.

7.2.3 Anomalous Predicates

Although the baseline theory in Chapter 2 mostly works for most responsive predicates, and the Twin Relations theory in Chapter 4 addresses some of the more troubling anomalies, many other issues remain.

As discussed in Sections 4.2.1 and 4.6.1, Beck and Rullmann (1999) suggest a semantics for *'agree'* that does not admit either a reductive treatment or a treatment in terms of the Twin Relations theory, but it is not clear that they have the data right (related treatments of *'agree'* appear in Sharvit (2002) and Lahiri (2002), but I have not investigated these in as much detail). Even if Beck and Rullmann's account of semantics of *'agree'* is not exactly accurate, it seems likely that *'agree'* is non-reducible, and it deserves further attention. I think that what is called for is a more systematic attempt to get a handle on the truth of agreement attributions in different

situations, and I hope to explore this issue with a variety of methods in the future.

The truthiness implicature with communication verbs like *'tell'* represents another anomaly. I have argued that this arises not just with question embedding, but also with inanimate subjects, but this does not obviously get us closer to an explanation of why this implicature arises, since it is not at all clear what the commonality might be between sentences with inanimate subjects and sentences with embedded questions for objects. These kinds of predicates also feature in most of the troubling examples like the one discussed in Section 7.2.1, and this may be related to the truthiness implicature. The intuition behind the proposal in Heim (1994) for a non-reductive treatment of question-embedding *'tell'* may also be relevant.

'surprise' (probably along with at least some other emotive factives) presents a third troubling case. As noted elsewhere, *'wh'*-questions embedded under surprise seem to always receive mention-some readings, independent of whether there is any evidence that mention-some readings are available for those questions in other contexts, and in spite of the fact that strongly exhaustive readings are available with all the other predicates considered (with the possible exception of *'agree'*). I have discussed these issues briefly, but I do not know of any really satisfying account of these facts.

7.2.4 Alternative Questions

As I noted in Chapter 5, alternative questions are an anomaly for the kind of account that I have been presenting. Ideally, we would want to build a system that gets alternative questions out of general-purpose question-building tools and a general-purpose theory of disjunction, and this seems difficult for the kind of theory that I have been using. Recent work on the ability of disjunction to raise alternatives and on the logic of alternative questions (e.g. Groenendijk (2009) and Mascarenhas (2009)) holds some promise of shedding new light on the nature of disjunction in alternative questions.

7.2.5 Non-Reducibility

The biggest new problem discussed in this dissertation was the non-reducibility of certain responsive predicates, most prominently including *'know'*. I proposed one

solution to this that handled the two problematic examples that were my primary focus, but further investigation of these issues is needed, besides the general value of further exploration, I want to highlight a few particular issues that deserve attention.

First, it is likely that *'agree'* exhibits some kind of nonreducibility, for reasons discussed in Section 4.2.1. If this turns out to be the case (and if domain accounts, pragmatic accounts, decomposition accounts, and similar responses are deemed inadequate or implausible), then some revision of the Twin Relations account will most likely be needed to accommodate it. The details must await an exploration of the key facts for *'agree'*.

Second, the interaction of these non-reducibility effects with homogeneity presuppositions deserves further explanation, and I suspect that a theory unifying the two phenomena may be possible. I do not know how to proceed with this, so my own discussion of these issues has been brief, but further work is called for.

Third, the interactions of the Twin Relations theory (and its competitors) with various other ambiguities, such as the pair-list reading, deserve further attention. The rough treatment pair-list readings that I presented in Section 3.5 makes use of the 'quantifier raising' approach to question embedding, while the Twin Relations account is a refinement of the lexical rules approach. For this reason, some other way of attack pair-list readings will be needed to get them to work properly with the Twin Relations account.

7.2.6 Quantificational Variability

In the course of this dissertation, I have ignored one of the most famous phenomena in the study of question embedding: the phenomenon of quantificational variability (QV). QV is the phenomenon of the interaction of certain quantificational adverbs with embedded questions, which causes (2) to mean something close to (3):

(2)   Anne and Rupert mostly agree on who is a traitor.

(3)   For most salient people x, Anne and Rupert agree on whether x is a traitor.

For a good overview of this phenomenon and of historically important approaches to it, the reader is referred to Lahiri (2002). I have ignored QV in order to focus on other issues (especially non-reducibility), but of course it is an important phenomenon that deserves attention. My under-informed impression is that the general strategy seen in Beck and Sharvit (2002) is a promising direction, and in the future I hope to combine an account in that spirit with a version of the approach to question embedding set out in Chapters 2 and 4.

7.3   Final Remarks

The structure of this document makes it difficult to offer a concise summary. The best I can offer is 'questions are complicated', or, in a bit more detail, 'questions are complicated, but not in the ways that we thought they were'. In particular, the supposed complication of weak exhaustivity, which is prominent in most recent accounts of question-embedding, seems to me to be dispensable, but deeper complications like non-reducibility (among many others) force us to reevaluate the way we think about question semantics and the basic meaning of responsive predicates. I hope that my exploration of these complications has begun to shed some light on the nature of these problems and on possible directions of attack, but we do not yet have a clear view of the whole picture, and so it seems almost certain that considerable refinements, at least, will be needed.

> It was stated at the outset, that this system would not be here, and at once, perfected. You cannot but plainly see that I have kept my word.
>
> Melville (1851)

APPENDIX A

Sketch of the Assumed Logic and Type System

This appendix describes briefly the kind of logic I assume as a convenient language for semantic representations, with a particular emphasis on my use of tuple types. Except for my use of tuple types, and adjustments related to this, the assumed logic is like the typical intensional logics and multi-sorted type theories used as representation languages in the formal semantics literature, and described in detail in, for example, Janssen (1984) and Dowty et al. (1981). My description here is neither very rigorous nor very pedagogically transparent, but it should provide enough detail for readers familiar with the formal descriptions the these kinds of systems to get a sense of what I have in mind with my inclusion of tuple types.

A.1 Types

A.1.1 Basic Types

The basic types are `t`, `e`, `s`, and `l`. These are chosen for concreteness, but nothing will hinge on this precise choice, and all the constructions below will go through for other choices of basic types.

A.1.2 Assumptions about Sequences/Tuples

I assume the existence of n-ary sequences or n-tuples (I'll use the terms interchangeably), for any $n \geq 0$.

There is a unique zero-ary sequence, written ε, and n-ary sequences will generally be written $(x_1, ..., x_n)$ or $(x_1; ...; x_n)$ (the latter notation being reserved to make tuples of types more easily distinguish). Where $n = 1$, we have $(x) = x$. That is, there is no distinction between a 1-ary sequence and its sole component. For sets $S_1, ..., S_n$, the Cartesian product of $S_1, ..., S_n$, written $S_1 \times ... \times S_n$, is the set of all n-ary sequences

$(x_1, ..., x_n)$ such that $x_1 \in S_1$, ... $x_n \in S_n$.

A.1.3   Complex Types

The vocabulary of complex types is given below. A special subset of the types is the singleton types. The set of types is defined recursively in (1):

(1)   a.   Every basic type is a singleton type (i.e. `t`, `e`, `s`, and `l` are all singleton types).

b.   For all singleton types σ and τ, $[\sigma \to \tau]$ is a singleton type. (That is, for any two singleton types, the function type mapping one to the other is also a singleton type.)

c.   For every $n \geq 0$, and all singleton types $\tau_1, ..., \tau_n$, the tuple $(\tau_1; ...; \tau_n)$ is a type. If $n \neq 1$, then $(\tau_1; ...; \tau_n)$ is not a singleton type. If $n = 1$, then we have the case where $(\tau_1) = \tau_1$, which is a singleton type. (That is, any tuple of singleton types is a type.)

d.   Nothing else is a type.

The notation $[\sigma \to \tau]$ is chosen because the notation $\langle \sigma, \tau \rangle$ has been repurposed. Essentially, $[\sigma \to \tau]$ means what $\langle \sigma, \tau \rangle$ usually means – it is the type of functions between types σ and τ in the straightforward sense. In a moment, $\langle \sigma, \tau \rangle$ will receive a nonstandard definition in terms of $[\sigma \to \tau]$, which will allow it to handle tuples with the necessary flexibility.

A.1.4   Operations on Types

The above types are all the types that there are in the language. However, we have two additional binary operations for manipulating types: concatenation (written $\frown$), and taking generalized function types (written $\langle , \rangle$).These are defined below.

(2)   For all types τ and σ...

a. $\tau \frown () = () \frown \tau = \tau$.

b.   If there are $m \geq 1$, $n \geq 1$ such that $\tau = (\tau_1; ...; \tau_m)$, and $\sigma = (\sigma_1; ...; \sigma_n)$, the $\tau \frown \sigma = (\tau_1; ...; \tau_m; \sigma_1; ...; \sigma_n)$.

(3)   If τ is any type and σ is a singleton type, $\langle \tau, \sigma \rangle$ is well-defined and is given
by the following recursive definition:

    a. $\langle (), \sigma \rangle = \sigma$

    b.   If $\tau \neq ()$, then there are some singleton type τ' and some type τ'' such
that $\tau = \tau' \frown \tau''$. In this case $\langle \tau, \sigma \rangle = [\tau' \to \langle \tau'', \sigma \rangle]$. (The action of $\langle, \rangle$
on tuple types is defined recursively, so that in general $\langle (\tau_1; ...; \tau_n), \sigma \rangle =$
$[\tau_1 \to ...[\tau_n \to \sigma]...].$)

This gives us many different ways of naming the same type. For example, all the
expressions in (4) name the same type:

(4)   a. $\langle e, \langle e, \langle e, t \rangle \rangle \rangle$

    b. $\langle (e; e), \langle e, t \rangle \rangle$

    c. $\langle e, \langle (e; e), t \rangle \rangle$

    d. $\langle (e; e; e), t \rangle$

    e. $\langle (e; e) \frown e, t \rangle$

    f. $\langle (e; e), [e \to t] \rangle$

    g. $\langle e \frown e, [e \to t] \rangle$

    h. $[e \to [e \to [e \to t]]]$

A.2   Domains, Vocabulary, Models, Contexts, and Assignments

A.2.1   Domains for Types

A D is a function from the basic types other than t (that is, the domain of D is
$\{e, s, l\}$) that maps each of these basic types to a nonempty set.

For every type τ, the D_τ (the domain of τ) is as follows:

(5)   a. $D_{()} = \{\varepsilon\}.$

    b.   For every non-t basic type τ, $D_\tau = D(\tau)$.

    c. $D_t = \{\top, \bot\}.$

    d.   For all singleton types τ and σ, $D_{[\tau \to \sigma]} = D_\sigma^{D_\tau}$ (that is, $D_{[\tau \to \sigma]}$ is the
set of functions from D_τ into D_σ).

    e.   For every $n \geq 2$, and all singleton types $\tau_1, ..., \tau_n$, $D_{(\tau_1; ...; \tau_n)} = D_{\tau_1} \times$

$\ldots \times D_{\tau_n}$ (that is, the set of all n-tuples whose first element is from D_{τ_1}, whose second element is D_{τ_2}, and so on).

Domain D is 'well behaved' iff, for all types τ and τ', $D_\tau \cap D'_\tau \neq \emptyset$ iff $\tau = \tau'$.

A.2.2 Signatures and Non-Logical Vocabulary

A 'signature' κ is an ordered triple $(C_\kappa, J_\kappa, t_\kappa)$ where C_κ and J_κ are disjoint, possibly empty sets of symbols, none of which are among the symbols of the logic employed below, (the set of non-logical constants and the set of indexical symbols, respectively), and t_κ is a function from $C_\kappa \cup J_\kappa$ into the set of types. That is, t_κ stores information about which type each constant and each indexical belongs to.

A.2.3 Models

For every signature κ, a κ-model M is an ordered pair (D_M, ι_M) such that D_M is a well-behaved domain, and ι_M has as its domain the elements of C_κ, and has the property that, for every $c \in C_\kappa$, $\iota_M(c) \in D_{M,t_\kappa(c)}$. That is, a model provides a domain, and and provides a type-appropriate semantic value for every non-logical constant of the language.

A.2.4 Contexts

For every signature κ and every κ-model M, a κ,M-context is a function γ with domain J_κ such that, for every $j \in J_\kappa, \gamma(j) \in D_{M,t_\kappa(j)}$. That is, a context provides type-appropriate semantic values for all the indexicals.

A.2.5 Assignments

For any model M, an M-assignment α is a function whose domain is the set of all ordered pairs of a type and a non-negative integer, with the property that, for every non-negative integer n and every type τ, $\alpha(\tau, n) \in D_{M,\tau}$. (Variables will be indexed by non-negative integers, so an assignment is a function that will serve to assign a type-appropriate semantic value to every variable.)

A.3 The Representation Language

For every signature κ, L_κ will be our representation language for κ. L_κ gives us a set of expressions $L_{\kappa,\tau}$, for every type τ. These $L_{\kappa,\tau}$s are as defined below.

(6)   Definition of the representation language:

    a.   For every $c \in C_\kappa$, $c \in L_{\kappa,t_\kappa(c)}$. (Constants are expressions of the language.)

    b.   For every $j \in J_\kappa$, $j \in L_{\kappa,t_\kappa(j)}$. (Indexicals are expression of the language.)

    c.   For every type τ, and every integer n, $v_{\tau,n} \in L_{\kappa,\tau}$. (Variables are expressions of the language.)

    d. 0, $1 \in L_{\kappa,\mathbf{t}}$. (The language has names for the truth values.)

    e.   For every type τ, and all expressions φ, $\psi \in L_{\kappa,\tau}$, $(\varphi = \psi) \in L_{\kappa,\mathbf{t}}$. (The language is equipped with a general equality relation.)

    f.   For all φ, $\psi \in L_{\kappa,\mathbf{t}}$, $(\varphi \vee \psi) \in L_{\kappa,\mathbf{t}}$. (The language has a notation for disjunction.)

    g.   For all singleton types τ and all types σ, all integers n, and all $\varphi \in L_{\kappa,\tau}$, $\lambda v_{n,\sigma}(\varphi) \in L_{\kappa,\langle\sigma,\tau\rangle}$. (The language has λ-abstraction.)

    h.   For all types τ, all types σ, all $\varphi \in L_{\kappa,\langle\sigma,\tau\rangle}$ and all $\psi \in L_{\kappa,\sigma}$, $\varphi(\psi) \in L_{\kappa,\tau}$. (The language expresses function application in the usual way.)

    i.   For all types τ and σ, all $\varphi \in L_{\kappa,\tau}$, and all $\psi \in L_{\kappa,\sigma}$, $\varphi \otimes \psi \in L_{\kappa,\tau\frown\sigma}$. (The language has an expression for sequence concatenation.)

    j.   For all singleton types $\tau_1, ..., \tau_n$, all m such that $m \geq 1$ and $m \leq n$, and all $\varphi \in L_{\kappa,(\tau_1;...;\tau_n)}$, $\varphi_{[m]} \in L_{\kappa,\tau_m}$. (The language has an expression for the mth element of a sequence.)

    k.   For all types τ, nothing is in $L_{\kappa,\tau}$ that is not required to be by these rules.

A.4 Notational Embellishments

Some additional notations are defined in terms of the above logic:

(7)   a.   For every φ of type $\mathbf{t}$, $\neg\varphi$ is a shorthand for $(\varphi = 0)$.

       b.   For all φ and ψ of type $\mathbf{t}$, $(\varphi \wedge \psi)$ is a shorthand for $\neg(\neg\varphi \vee \neg\psi)$.

c.	For all φ and ψ of type $\mathtt{t}$, $(\varphi \to \psi)$ is a shorthand for $(\neg\varphi \vee \psi)$

d.	For every variable v and every φ of type $\mathtt{t}$, $\forall v(\varphi)$ is a shorthand for $\lambda v(\varphi) = \lambda v(1)$.

e.	For every variable v and every φ of type $\mathtt{t}$, $\exists v(\varphi)$ is a shorthand for $\neg\forall v(\neg\varphi)$.

f.	Various traditional variable names like x, p, P, y', and so on are presumed to be nicknames of various $v_{\tau,n}$s.

A.5   Interpretation of the Representation Language

I now want to sketch the interpretation of the representation language with respect to a model, context, and assignment. Before I define the interpretation, I want to define a couple more utility operations in our technical English metalanguage.

A.5.1   Metalanguage Operations

(8)	Assignment update:

For every signature κ, every κ-model M, every M-assignment α, every type τ, every non-negative integer n, and every $a \in D_{M,\tau}$ and every $\alpha\frac{a}{n,\tau}$ is the unique assignment such that $\alpha\frac{a}{n,\tau}(n,\tau) = a$, and for all integers m and all $\sigma \in T_\beta$ s.t. $m \neq n$ or $\sigma \neq \tau$, $\alpha\frac{a}{n,\tau}(m,\sigma) = \alpha(m,\sigma)$.

This is the usual operation of minimally adjusting an assignment to give a different value to a particular variable.

(9)	Sequence concatenation:

For all sequences a and b, the concatenation of a and b, written ab, is defined as follows:

If $a = \varepsilon$, then $ab = b$.

If $b = \varepsilon$, then $ab = a$.

Otherwise, there are $m \geq 1$ and $n \geq 1$ s.t. $a = (a_1, ..., a_m)$ and $b = (b_1, ..., b_m)$. In this case, $ab = (a_1, ..., a_m, b_1, ..., b_m)$.

(10)	Sequence-wise function application:

For every domain D, every type σ, every singleton type τ, every $f \in D_{\langle \sigma, \tau \rangle}$, and every $a \in D_\sigma$, $f \backslash a$ is defined as follows:

If $\sigma = ()$ (i.e. $a = \varepsilon$), then $f \backslash a = f$.

Otherwise, there are a singleton type σ' and a type σ'' such that $\sigma = \sigma' \frown \sigma''$, and likewise there are $a' \in D_{\sigma'}$ and $a'' \in D_{\sigma''}$ such that $a = a'a''$. In this case, $f \backslash a = f(a') \backslash a''$.

Note that in the case where σ is a singleton type, this means $f \backslash a = f(a) \backslash \varepsilon = f(a)$.

The $f \backslash a$ defines the evaluation of a Curried function with respect to a sequence. This will be needed to support the polymorphic interaction of functions with sequences required by the syntax of the representation language.

A.5.2 The Interpretation Function

For signature κ, every κ-model M, every κ,M-context γ, and every M-assignment α, $[\![\,]\!]^{M,\gamma,\alpha}$ is the interpretation function for L_κ. This function has the property that, for every type τ, and every $\varphi \in L_{\kappa,\tau}$, $[\![\varphi]\!]^{M,\gamma,\alpha} \in D_{M,\tau}$.

(11)   Definition of $[\![\,]\!]^{M,\gamma,\alpha}$

   a.   For every $c \in C_\kappa$, $[\![c]\!]^{M,\gamma,\alpha} = \iota_M(c)$. (That is, the semantic value of a constant is given by the model.)

   b.   For every $j \in J_\kappa$, $[\![j]\!]^{M,\gamma,\alpha} = \gamma(j)$. (That is, the semantic value of an indexical is given by the context.)

   c.   For every type τ, and every non-negative integer n, $[\![v_{\tau,n}]\!]^{M,\gamma,\alpha} = \alpha(\tau, n)$. (That is, the semantic value of a variable is given by the assignment.)

   d. $[\![0]\!]^{M,\gamma,\alpha} = \bot$ and $[\![1]\!]^{M,\gamma,\alpha} = \top$. (That is, the truth-value names always mean the appropriate truth values.)

   e.   For every τ, and all expressions $\varphi, \psi \in L_{\kappa,\tau}$, $[\![(\varphi = \psi)]\!]^{M,\gamma,\alpha} = \top$ iff $[\![\varphi]\!]^{M,\gamma,\alpha} = [\![\psi]\!]^{M,\gamma,\alpha}$, and otherwise $[\![(\varphi = \psi)]\!]^{M,\gamma,\alpha} = \bot$. (That is, $\varphi = \psi$ is true iff φ and ψ have the same semantic value.)

   f.   For all $\varphi, \psi \in L_{\kappa,\mathbf{t}}$, $[\![\varphi \vee \psi]\!]^{M,\gamma,\alpha} = \top$ iff $[\![\varphi]\!]^{M,\gamma,\alpha} = \top$ or $[\![\psi]\!]^{M,\gamma,\alpha} = \top$ (or both), and otherwise $[\![\varphi \vee \psi]\!]^{M,\gamma,\alpha} = \bot$. (That is, $\vee$ is inclusive disjunction.)

   g.   For all singleton types τ and all types σ, all non-negative integers n, and

all $\varphi \in L_{\kappa,\tau}$, $[\![\lambda v_{n,\sigma}(\varphi)]\!]^{M,\gamma,\alpha}$ is the unique $f \in D_{M,\langle\sigma,\tau\rangle}$ such that, for all $a \in D_{M,\sigma}$ $f\backslash a = [\![\varphi]\!]^{M,\gamma,\alpha\frac{a}{n,\sigma}}$. (That is, λ-abstraction is λ-abstraction with respect to the general reduction of a function by a sequence.)[1]

h. For all singleton types τ, all types σ, all $\varphi \in L_{\kappa,\langle\sigma,\tau\rangle}$ and all $\psi \in L_{\kappa,\sigma}$, $[\![\varphi(\psi)]\!]^{M,\gamma,\alpha} = [\![\varphi]\!]^{M,\gamma,\alpha}\backslash[\![\psi]\!]^{M,\gamma,\alpha}$. (That is, $\varphi(\psi)$ is evaluated in terms of reduction of a function by a sequence, not in terms of function application.)

i. For all types τ and σ, all $\varphi \in L_{\kappa,\tau}$, and all $\psi \in L_{\kappa,\sigma}$, $[\![\varphi \otimes \psi]\!]^{M,\gamma,\alpha} = [\![\varphi]\!]^{M,\gamma,\alpha}[\![\psi]\!]^{M,\gamma,\alpha}$. (That is, the semantic value of $\varphi \otimes \psi$ is the sequence concatenation of the semantic values of φ and ψ.)

j. For all singleton types $\tau_1, ..., \tau_n$, all m such that $m \geq 1$ and $m \leq n$, and all $\varphi \in L_{\kappa,(\tau_1;...;\tau_n)}$, there are $a_1 \in D_{M,\tau_1}, ..., a_n \in D_{M,\tau_n}$ such that $[\![\varphi]\!]^{M,\gamma,\alpha} = (a_1, ..., a_n)$. Given this, $[\![\varphi_{[m]}]\!]^{M,\gamma,\alpha} = a_m$. (That is, the semantic value of $\varphi_{[m]}$ is the mth sequence element of the semantic value of φ.)

Rules (11-g) and (11-h) are the main changes from the kinds of logics usually assumed for semantic representation. These have been generalized to allow us to λ-abstract over a variable for tuples (instead of abstracting over many variables individually), and to apply a function to a tuple (with the same results as applying it to singleton arguments one at a time). All the other rules, with the exception of the rules for the sequence-manipulation operations (11-i) and (11-j), are given their familiar interpretations.

A.6 Example

Consider the formula (12):

(12) $\lambda v_{6,(e;e)} loves(v_{6,(e;e),[2]} \otimes v_{6,(e;e),[1]})$

Some additional parentheses have been furnished to clarify the structure above. *loves* is assumed to a constant of type $[e \to [e \to t]]$.

---

[1]The existence of a unique such f is not proven here, but is, I hope, plausible enough that this lapse will be forgiven.

The derivation of (12) by the grammar of the logic is sketched in (13).

(13)
$$\lambda v_{6,(\mathsf{e};\mathsf{e})} loves(v_{6,(\mathsf{e};\mathsf{e}),[2]} \otimes v_{6,(\mathsf{e};\mathsf{e}),[1]})$$
$$\langle (\mathsf{e};\mathsf{e}), \mathsf{t} \rangle = [\mathsf{e} \to [\mathsf{e} \to \mathsf{t}]]$$
(6-g)

|

$$loves(v_{6,(\mathsf{e};\mathsf{e}),[2]} \otimes v_{6,(\mathsf{e};\mathsf{e}),[1]})$$
$$\mathsf{t}$$
(6-h)

$loves$ $v_{6,(\mathsf{e};\mathsf{e}),[2]} \otimes v_{6,(\mathsf{e};\mathsf{e}),[1]}$

$[\mathsf{e} \to [\mathsf{e} \to \mathsf{t}]] = \langle (\mathsf{e};\mathsf{e}), \mathsf{t} \rangle$ $(\mathsf{e};\mathsf{e})$

(6-a)   (6-i)

$v_{6,(\mathsf{e};\mathsf{e}),[2]}$ $v_{6,(\mathsf{e};\mathsf{e}),[1]}$

e e

(6-j)   (6-j)

| |

$v_{6,(\mathsf{e};\mathsf{e})}$ $v_{6,(\mathsf{e};\mathsf{e})}$

$(\mathsf{e};\mathsf{e})$ $(\mathsf{e};\mathsf{e})$

(6-c)   (6-c)

Now, let's consider the interpretation of each node, for any model $M = (D_M, \iota_M)$, context γ, and assignment α:

$[\![v_{6,(\mathsf{e};\mathsf{e})}]\!]^{M,\gamma,\alpha} = \alpha((\mathsf{e};\mathsf{e}), 6)$.

$[\![v_{6,(\mathsf{e};\mathsf{e}),[2]}]\!]^{M,\gamma,\alpha} = $ the second component of $\alpha((\mathsf{e};\mathsf{e}), 6)$.

$[\![v_{6,(\mathsf{e};\mathsf{e}),[1]}]\!]^{M,\gamma,\alpha} = $ the first component of $\alpha((\mathsf{e};\mathsf{e}), 6)$.

$[\![v_{6,(\mathsf{e};\mathsf{e}),[2]} \otimes v_{6,(\mathsf{e};\mathsf{e}),[1]}]\!]^{M,\gamma,\alpha} = $ the sequence resulting from concatenating the singleton sequence $[\![v_{6,(\mathsf{e};\mathsf{e}),[2]}]\!]^{M,\gamma,\alpha}$ and the singleton sequence $[\![v_{6,(\mathsf{e};\mathsf{e}),[1]}]\!]^{M,\gamma,\alpha}$, which is to say the sequence resulting from reversing the first and second components of $\alpha((\mathsf{e};\mathsf{e}), 6)$.

$[\![loves]\!]^{M,\gamma,\alpha} = \iota_m(loves)$.

249

$[\![loves(v_{6,(e;e),[2]} \otimes v_{6,(e;e),[1]})]\!]^{M,\gamma,\alpha} = \iota_m(loves) \backslash [\![v_{6,(e;e),[2]} \otimes v_{6,(e;e),[1]}]\!]^{M,\gamma,\alpha} = \iota_M(loves)([\![v_{6,(e;e),[2]}]\!]^{M,\gamma,\alpha})([\![v_{6,(e;e),[1]}]\!]^{M,\gamma,\alpha}).$

$[\![\lambda v_{6,(e;e)} loves(v_{6,(e;e),[2]} \otimes v_{6,(e;e),[1]})]\!]^{M,\gamma,\alpha} =$ the unique $f \in D_{M,[e\to[e\to t]]}$ such that, for all $a \in D_{M,(e;e)}$, $fa\backslash = [\![loves(v_{6,(e;e),[2]} \otimes v_{6,(e;e),[1]})]\!]^{M,\gamma,\alpha \frac{a}{6,(e;e)}}$. That is, for $a = (a', a'') \in D_{M,(e;e)}$, $f\backslash a = \iota_m(loves)(a'')(a')$. The unique f with this property is the f such that, for all $b', b'' \in D_{M,e}$, $f(b')(b'') = \iota_m(loves)(b'')(b')$.

Bibliography

Douglas Adams. *The Hitchhiker's Guide to the Galaxy*. Pan Books, 1979.

Maria Aloni. Questions under cover. In *Words, Proofs, and Diagrams*. CSLI, Stanford, California, 2002.

Sigrid Beck and Hotze Rullmann. A flexible approach to exhaustivity in questions. *Natural Language Semantics*, 7:249–298, 1999.

Sigrid Beck and Yael Sharvit. Pluralities of questions. *Journal of Semantics*, 19: 105–157, 2002.

Ivano Caponigro. The semantic contribution of wh-words and type shifts: evidence from free relatives crosslinguistically. In Robert B. Young, editor, *Proceedings from Semantics and Linguistic Theory XIV*, pages 38–55, 2004.

Ivano Caponigro and Daphna Heller. The non-concealed nature of free relatives: Implications for connectivity crosslinguistically. 2007.

Lewis Carroll. *Alice's Adventures in Wonderland*. 1865. URL `http://www.gutenberg.org/ebooks/11`.

Gennaro Chierchia. Questions with quantifiers. *Natural Language Semantics*, pages 181–234, 1993.

Sam Cumming. Concealed questions. Manuscript, Rutgers, 2006.

Annie Dillard. *An American Childhood*. Harper & Row, 1987.

David R. Dowty, Robert E. Wall, and Stanley Peters. *Introduction to Montague Semantics*. Kluwer, 1981.

Paul Egré and Benjamin Spector. Embedded questions revisited: *an* answer, not necessarily *the* answer. Presentation at MIT Ling-Lunch Seminar and Journées Sémantique et Modélisation, 2007.

Elisabet Engdahl. *Constituent questions: The syntax and semantics of questions with special reference to Swedish*. D. Reidel Publishing Company, 1986.

Jonathan Ginzburg. Resolving questions, part I. *Linguistics and Philosophy*, 18: 459–527, 1995a.

Jonathan Ginzburg. Resolving questions, part II. *Linguistics and Philosophy*, 18: 567–609, 1995b.

William Goldman. The princess bride. Film, 1987. Goldman is credited writer.

William W. Greener. *The Science of Gunnery, as applied to the use and construction of Fire arms.* E. Churton, 1846.

Jeroen Groenendijk. Inquisitive semantics: Two possibilities for disjunction. In *Seventh International Tbilisi Symposium on Language, Logic, and Computation.* Springer-Verlag, 2009.

Jeroen Groenendijk and Martin Stokhof. Semantic analysis of wh-complements. *Linguistics and Philosophy*, 1982.

Jeroen Groenendijk and Martin Stokhof. *Studies on the Semantics of Questions and the Pragmatics of Answers.* PhD thesis, University of Amsterdam, 1984.

Elena Guerzoni and Yael Sharvit. A question of strength: on NPIs in interrogative clauses. *Linguistics and Philosophy*, 30:361–391, 2007.

C. L. Hamblin. Questions in Montague English. *Foundations of Language*, 10:41–53, 1973.

Irene Heim. Interrogative semantics and Karttunen's semantics for *know*. In Rhonna Buchalla and Anita Mitwoch, editors, *IATL 1*, pages 128–144, 1994.

Irene Heim and Angelika Kratzer. *Semantics in Generative Grammar.* Blackwell, Oxford, 1998.

James Higginbotham and Robert May. Questions, quantifiers, and crossing. *The Linguistic Review*, 1:41–80, 1981.

Jaakko Hintikka. *The Semantics of Questions and the Questions of Semantics.* North Holland Publishing Company, Amsterdam, 1976.

Donovan Hohn. *Moby-Duck: The True Story of 28,800 Bath Toys Lost at Sea and of the Beachcombers, Oceanographers, Environmentalists, and Fools, Including the Author, Who Went in Search of Them.* Viking, 2011.

Joachim Jacobs. Lexical decomposition in Montague-grammar. *Theoretical Linguistics*, 7:121–136, 1980.

Pauline Jacobson. On the quantificational force of English free relatives. In *Quantification in Natural Languages*, pages 451–486. Kluwer Academic Publishers, 1995.

Theo Janssen. *Foundations and Applications of Montague Grammar.* PhD thesis, University of Amsterdam, 1984.

Elsi Kaiser. A question of case. In *Proceedings of the Workshop on Generative Approaches to Finnic Languages.* 2004.

Lauri Karttunen. Syntax and semantics of questions. *Linguistics and Philosophy*, 1: 3–44, 1977.

Edward L. Keenan and Leonard M. Faltz. *Boolean Semantics for Natural Language.* D. Reidel Publishing Company, 1985.

Edward L. Keenan and Jonathan Stavi. A semantic characterization of natural language determiners. *Linguistics and Philosophy*, 9:253–326, 1986.

Angelika Kratzer. Exhaustive questions, 2006. Linguistics Colloquium, Massachusetts Institute of Technology.

Manfred Krifka. Quantifying into question acts. *Natural Language Semantics*, 9: 1–40, 2001.

Utpal Lahiri. *Questions and Answers in Embedded Contexts.* Oxford University Press, 2002.

Godehard Link. *Algebraic Semantics in Language and Philosophy.* CSLI Publications, 1997.

Salvador Mascarenhas. Inquisitive semantics and logic. Master's thesis, Institute for Logic, Language and Computation, University of Amsterdam, 2009.

Herman Melville. *Moby Dick, or, the whale.* 1851. URL `http://www.gutenberg.org/ebooks/2701`.

A. A. Milne. *The House at Pooh Corner.* 1928.

Richard Montague. The proper treatment of quantification in ordinary English. In Richmond H. Thomason, editor, *Formal Philosophy: Selected Papers of Richard Montague*, pages 247–270. Yale University Press, New Haven, 1974.

Lance Nathan. *On the interpretation of concealed questions.* PhD thesis, Massachusetts Institute of Technology, 2006.

Oxford English Dictionary. URL `http://www.oed.com/`. Online resource, articles retrieved May 2011.

Barbara H. Partee. Noun phrase interpetation and type-shifting principles. In *Studies in Discourse Representation Theory and the Theory of Generalized Quantifiers*, pages 115–143. Foris Publications Holland, 1986.

Chris Potts. When even *No*'s neg is splitsville, 2000. URL `http://ling.ucsc.edu/Jorge/potts.html`. Squib for the Jorge Hankamer WebFest.

Maribel Romero. Concealed questions and specificational subjects. *Linguistics and Philosophy*, 28:687–737, 2005.

Maribel Romero. Concealed questions with quantifiers. In *Logic, Language, and Meaning: 17th Amsterdam colloquium*, pages 21–31, 2010.

William Shakespeare. Much ado about nothing, c. 1600.

Yael Sharvit. Emedded questions and 'de dicto' readings. *Natural Language Semantics*, 10:97–123, 2002.

Benjamin Spector. Exhaustive interpretations: what to say and what not to say. presented at the LSA workshop on Context and Content, 2005.

Benjamin Spector. Modalized questions and exhaustivity. In *Proceedings of Semantics and Linguistic Theory 17.* CLC publications, Cornell University, 2007.

Benjamin Spector. Topics in the semantics of interrogative clauses, 2010. Mini-course, Vienna.

Anna Szabolcsi. Quantifiers in pair-list readings. In Anna Szabolcsi, editor, *Ways of Scope Taking*, pages 311–348. Kluwer Academic Publishers, 1997.

Wataru Uegaki. Content nouns and the semantics of question-embedding predicate. Manuscript, MIT, 2011.

Robert van Rooij. The utility of mention-some questions. *Research on Language and Computation*, 2:401–416, 2004.

Dan Vebber. Daria: Art burn. Television episode, 2001. Season 5, episode 7. Vebber is credited writer.

Brian Weatherson. 'sure' and questions, 2009. URL `http://tar.weatherson.org/2009/10/03/sure-and-questions/`. Blog post.