050.822 Research Seminar — **The syntax and semantics of questions**
Spring 1999 — February 16, 1999
Paul Hagstrom — Week 3: *Introduction to the semantics of questions*

The Semantics of Questions (Hamblin 1958)

The issue: *What is a question?*

Hamblin decides not to address the issue, it's too hard.
Instead, he investigates the *relation* between questions, statements, and answers.

Postulates:

I. An answer to a question is a statement.
II. Knowing what counts as an answer is equivalent to knowing the question.
III. The possible answers to a question are an exhaustive set of mutually exclusive possibilities.

I: Shorter responses are possible, but they are "statements in code"
(Most extreme case being *yes* and *no*, but elliptical cases count too).

II: Not every statement will "count" as an answer to a question *Q*.
Knowing what *counts* as an answer to *Q* is equivalent only if only *Q* licenses the particular set of statements that count as answer to *Q*.
[This forces us to accept I].
A question is like a statement with a *blank*.
For yes/no questions "*S*?" and "*not S*?" are equivalent (though *S* and *not S* are not equivalent statements).

III: The possible answers must be **exhaustive**—because when they are not exhaustive, we feel that the question is "logically improper."
Have you stopped stealing office supplies? because *yes* and *no* do not "cover all logical possibilities."
Hamblin explicitly declines to address *Do you have good vision? If not, do you wear glasses?*–type "relative questions" but they seem related.
That is, *have you stopped stealing office supplies?* divides the subset of the worlds in which *I have been stealing office supplies* is true, just like *Do you wear glasses?* divides the subset of worlds in which *I do not have good vision* is true.
So: Failure of exhaustivity (maybe relative to accepted presuppositions) yields an intuitively detectable "logical impropriety."
The possible answers must be **mutually exclusive**—at least to the extent that they are **complete** answers. If answering *In which continent is Luxembourg?* with *Either Europe, Asia, or Africa*, you have not given a complete answer. Completeness is the intuitive correlate to formal mutual exclusivity.

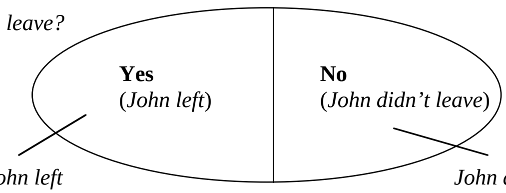
To be **exhaustive** and **mutually exclusive** is to be a **partition**.
Cf. more recent (and thorough) work by Groenendijk & Stokhof.

Consider the set of **possible worlds**.
In possible world w_1, A_1 is the true answer to the question *Q*.
In possible world w_2, A_2 is the true answer to the question *Q*.

Assuming that: In any given world w_n, exactly one answer A_m is true.

We can partition possible worlds by which answer is true in that world.

Did John leave?



John left is true in the worlds over here.
John didn't leave is true in the worlds over here.

The question *Did John leave?* then asks
Which of the two cells of the partition is the real world in?
or
Is the real world such that John left or is it such that John didn't leave?

Theorems:

A: If a question has only one possible answer, that answer is a tautology.
B: If any answer to a question is a tautology, it is the only possible answer.
C: Every question has an answer.

A: Assuming questions partition possible worlds (*exhaustively*), every possible world has to be in *some* cell. If the answer (or indeed any statement) is true in *all* possible worlds, it is a tautology.
B: True, given *mutual exclusivity* in the partition.
C: A failure to partition still gives you *one* answer (a tautology).

One question **contains** another if by answering the first, you provide the answer to the second.

In which continent is Ecuador? is contained in
What are the GPS coordinates of Ecuador's highest mountain peak?
Since if you have the answer to the second (and assuming you can translate GPS coordinates into continents), you have the information to answer the first.

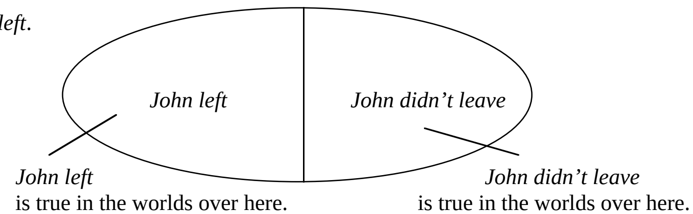
Digression into Groenendijko–Stokhofian semantics of questions

(The term *Groenendijko–Stokhofian* was in a recent paper I found by Jaroslav Peregrin and Klaus von Heusinger called “Dynamic semantics with choice functions”).

Truth and Meaning: Suppose that *knowing the meaning of S* is equivalent to knowing *the conditions under which S is true*.

We can then think of a *statement* (a *proposition*) as dividing up possible worlds.
(into those worlds in which it is true vs. those in which it is false).

John left.



This picture looks kind of familiar.
The left side cell is in fact the worlds in which *John left* is true—
we may say equivalent to the proposition *John left*.

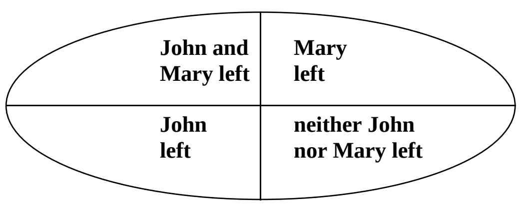
We will consider propositions to be *sets of worlds in which they’re true*.

The right side cell is the worlds in which *John didn’t leave* is true—
it’s the proposition *John didn’t leave*.

So the partition divides two propositions, *John left* and *John didn’t leave*.
Both answers to *Did John leave?*

We can partition possible worlds by which answer is true in that world.

Who left?



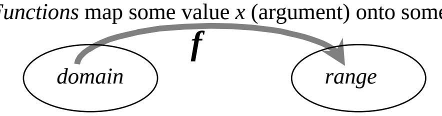
This is a step beyond Hamblin, who didn’t explicitly recognize the two on the diagonal (as far as I can tell). But if *J&M left* is a possible answer (different from *John left*) then the picture has to look like that. Lots more interesting stuff about these questions in work by Groenendijk & Stokhof—maybe we’ll look at it later.

Formalizing semantics (Heim & Kratzer 1998)

Chapter 1: *To know the meaning of a sentence is to know its truth conditions.*
Every meaningful part of a sentence contributes to its truth conditions in a systematic way.
“Saturated” vs. “Unsaturated”—Negation needs something to negate; it is “unsaturated” until it gets a “thought” to negate.
Unsaturated meanings are ***functions*** *taking an argument.*
Saturation consists of the application of a function to its arguments.

Functions and sets

Functions map some value *x* (argument) onto some other value *y*.

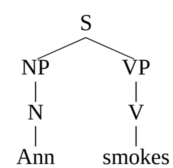


Functions cannot map any *one* element of the domain to more than one place.
(No problem if two elements of the domain land on the same value in the range).
Very simple function from domain {a,b,c} to range {d,e,f}:

$$\begin{bmatrix} a & \rightarrow & d \\ b & \rightarrow & e \\ c & \rightarrow & f \end{bmatrix}$$

Chapter 2: Some things don’t seem to be functions—seem to denote things.
So *true*, *false*, *Ann*, *Jan*. Don’t seem to be “unsaturated”

(1)



Frege: All semantic composition amounts to functional application.
Ann is not a function. But *smokes* is. It takes individuals (like *Ann*) as arguments, returns truth values (like *true* and *false*).

Nodes in the tree each have a *denotation* (assigned by the *interpretation function* ⟦ ⟧)
D ≡ the set of all individuals that exist in the world.
For the moment, denotations can be:
- Elements of D.
- Elements of {*true*, *false*}

or
- Functions from D to {*true*, *false*}

Lexicon: $[\![\text{Ann}]\!] = \textit{Ann}$ (where $\textit{Ann} \in D$).

$[\![\text{smokes}]\!] = f{:}D \to \{\textit{true}, \textit{false}\}$

For all $x \in D$, $f(x) = \textit{true}$ iff x smokes.

Rules for assigning a denotation to non-terminal nodes:

• Non-branching nodes: $[\![\alpha]\!] = [\![\beta]\!]$ where α is like: X
|
β

• Functional Application: $[\![\alpha]\!] = [\![\beta]\!]([\![\gamma]\!])$ where α is like: X
/ \
β γ

So:

(2)

```
                     S ——— true iff Ann smokes.
                   /   \
                 NP     VP
         /       |       |      \
Ann  ——<——————   N       V   ——————>  f:D→{true, false}
         \       |       |      /     ∀x∈D, f(x)=true iff x smokes.
                Ann    smokes
```

Characteristic functions

$[\![\text{smokes}]\!](x)$ is *true* iff x smokes.
We can make a set of people from D such that they smoke.
Then $[\![\text{smokes}]\!]$ will be true of all of them (and false of everyone else).

We can walk back & forth between sets and functions.
(*Sets are often easier to think about*.)

• Suppose we have a set A. We can define a *characteristic function*:
$f(x) = \textit{true}$ iff $x \in A$. (*false* otherwise)

• Suppose we have a function to $\{\textit{true}, \textit{false}\}$. We can treat it as the characteristic function of the set of all elements in the domain of the function that map to *true*.

Semantic types

e is a semantic type (individual).
t is a semantic type (truth value).
if σ and τ are semantic types, then $\langle\sigma,\tau\rangle$ is a semantic type.
nothing else is a semantic type.

Something of type <e,t> is a function from things of type e (D_e) to things of type t (D_t).
($D_t = \{\textit{true}, \textit{false}\}$. D_e is the set of all individuals.)

Lambda notation:

It is handy (and common) to write functions in "lambda notation."

$\lambda x \in A \,.\, \alpha$

This is a function which takes an argument, x, from the domain A, and returns the value α.

So: $[\![\text{smoke}]\!] = \lambda x \in D_e \,.\, (\textit{true}$ iff) x smokes *or sometimes just* $\lambda x.x$ smokes.

The result of a function can itself be a function. So, transitive verbs:

$[\![\text{love}]\!] = \lambda x \in D_e \,.\, [\, \lambda y \in D_e \,.\, y \text{ loves } x \,]$

so $[\![\text{love}]\!]([\![\text{Ann}]\!])$ returns the function $\lambda y \in D_e . y$ loves *Ann*.

Section 6.3.1 (*Generalized Quantifiers*)

Something, *everything*, *nothing* — not individuals (of which something can be true).
not *sets* of individuals.

Something vanished.

What *something* does here is tell us something about *vanished*.
It is true if there is *some* $x \in D_e$ for which x *vanished* is true.
So *something* actually takes *vanished* as its argument.

$[\![\text{something}]\!] = \lambda f \in D_{\langle e,t\rangle}$. there is some $x \in D_e$ such that $f(x) = \textit{true}$.

or $\lambda f \in D_{\langle e,t\rangle} \,.\, \exists x \in D_e \,.\, f(x)$.

Still conforms to the "functional application" view. In a binary branching structure, one node is the function (*something*) the other is the argument (*vanished*).

So *something*, *everything* are of type <<e,t>,t> (takes an <e,t> function, returns T/F).

Section 7.1 (*The problem of quantifiers in object position*)

John offended every linguist.

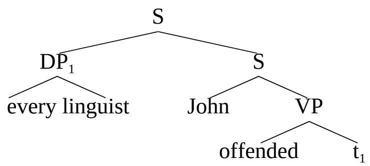
Every linguist like *everyone* is type <<e,t>,t>.
Offended needs a type e argument though (it is type <e,<e,t>>; cf. *John offended Mary*).
No way to combine <<e,t>,t> and <e,<e,t>> as function/argument. A **type mismatch**.

Section 7.3 (*Repairing the type mismatch by movement*)

Posit: A (semantically motivated) syntactic movement "Quantifier Raising" (QR).
It is "invisible" movement—covert movement—post-Spell-out movement.

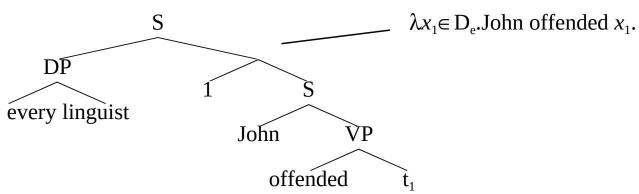
Syntactic movement leaves a trace.
Suppose the trace it leaves is interpreted as type e (like pronouns, actually).

Syntactically, this looks something like:

(3)


So as of the lower S we would have something which is true iff *John offended '1'*.
The t_1 is a variable which must be bound by the moved *every linguist*.
(There still needs to be some relation here—*John offended '1'* means little.)

every linguist is type <<e,t>,t> remember—it needs a sister of type <e,t>.
Heim & Kratzer propose that the movement index is a λ-operator to bind the variable:

(4)


So *every linguist* takes "λ*x*.John offended *x*" and returns *true* if the predicate is *true* of every *x* (such that *x* is a linguist).

Formalizing the semantics of questions (Hagstrom 1998, §§5.1–5.2)

Hagstrom 1998, ch. 5, sec. 1–2

Essentially, Hamblin 1958 + Heim & Kratzer 1998.
The only addition is: Possible worlds (a new type, *s*, and what follows from that).
So, propositions are of type <s,t> (functions from worlds to truth values).

Reminder: Questions divide up possible worlds. Like propositions, actually.
So, *John left* divides up possible worlds into two
(those where it is *true*, those where it is *false*).

We can write the set of possible worlds in which *John left* is true like this:
λ*w* . John left in *w*.
It's a function from possible worlds to *true*/*false*.
It's the characteristic function of the set of worlds in which *John left* is *true*.

We are taking *questions* to be *sets of their possible answers*.

So ⟦ *Who left?* ⟧ is supposed to be the set of propositions like *x left*.
We can write that like:

(5) $\{p : p = [\lambda w.x \text{ left in } w] \text{ where } x \in \text{people}\}$

(6) $\lambda p.\ p = [\lambda w.x \text{ left in } w] \text{ where } x \in \text{people}.$

(7) $\lambda p\ .\ \exists x \in \text{people}\ .\ p = \lambda w\ .\ x \text{ left in } w.$

So this is true if *John left* is the *p* because there is an *x*∈ people (namely, *John*) such that λ*w*.*x* left in *w* is the same as λ*w*.*John* left in *w*.

It wouldn't be true of *Stapler left*, assuming *stapler* is not in the set of people, since there is nothing we could choose from the set of people such that λ*w*.*x* left in *w* would be the same as λ*w*.*Stapler* left in *w*.

This is the basis of what I did in my thesis.
I assume a set of possible answers like *John left* and *Mary left*…
—but I made no allowances for *John and Mary left*.
—In fact, this is why I did *not* adopt Hamblin's "Postulate III"
(which said answers must be mutually exclusive and exhaustive).

Question: What about answers like *John and Mary left*?

This is something I did *not* address adequately. A couple of ways to think about it:

- Let *John and Mary* be an individual in the set D_e. ("a group").
- Allow *John and Mary left* to be "short for" answering *both* *John left* and *Mary left*—that is, you're allowed to pick more than one answer.
- Or, follow Postulate III and require each answer to be *complete*.
  (So, *John left, Mary left*, *John and Mary left*, are each independent answers. Moreover, if *John left* is true and these answers are exhaustive and mutually exclusive, we know *John and Mary left* is false—hence, that *Mary left* is false.) (Groenendijk & Stokhof's way).

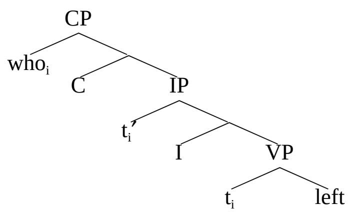
Connecting semantics to syntax

The semantics of *Who left?* is purported to be

(8) $\lambda p.\exists x\in \text{people}.p=\lambda w.x$ left in w.

The syntax of *Who left?* is something like:

(9)



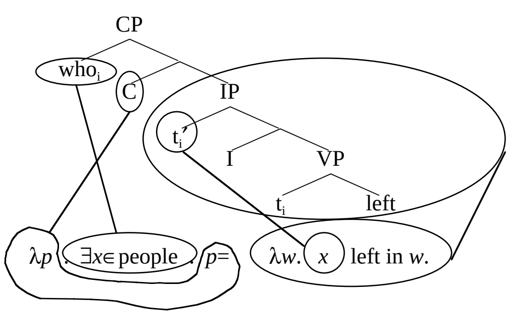
The idea is that if we assign the right denotations to the lexical items, we should get (8).

Commonly, movement chains are taken to represent operator-variable relations
So *who*…*t* is interpreted like Op(x)…x
(e.g. quantifiers after QR: *every x*…*x*, *at most seven x*…*x*, …)

Standing back and squinting,
maybe *who* is "$\exists x\in$ people" an operator over x, binding x in its trace position.
maybe the contents of IP is the "proposition", the part after "p="
maybe $C_{\text{interrogative}}$ is responsible for the abstraction over propositions ("λp.p=")

(10)

(11)



If this works, we *can* (given the right definitions) read the semantics right off the syntax.
(And, movement *could* be important for semantics—maybe $\exists x\in$ people has to get *into* C).

A couple of classic approaches (Hamblin 1973, Karttunen 1977)

Hamblin (1973) Questions in Montague English. *Foundations of Language* 10:41–53.

(*Hamblin's own goal*: Extend Montague's *English as a Formal Language* to questions.)

Hamblin 1973 looked at the problem a different way—(actually, historically prior)
Sure, the *set* is that characterized by $\lambda p\exists x\in \text{people}.p=\lambda w.x$ left in w,
but we can *get* this set by different means.

The set is like {*John left*, *Mary left*, *Bill left*, *Sue left*, …} i.e. contains *x left* for *x* people.

Hamblin makes the following suggestions:

Although we are inclined to class 'who' and 'what' with proper names we cannot by any stretch regard them as denoting individuals. But there is a simple alternative: **they *can* be regarded as denoting *sets* of individuals**, namely the set of humans and the set of non-humans respectively. (48)

This does not mean, of course, that the formula 'who walks' asserts that the set of human individuals walks: we must modify other stipulations in sympathy. We shall need to regard **'who walks'** as itself denoting a set, namely, **the set whose members are the propositions denoted by 'Mary walks', 'John walks', … and so on for all individuals.** (48)

The idea is that if you just make a *set* of people the argument, since *walks* is only a function from (single, atomic) people to truth conditions, you apply *walks* to each person in the set. [This is the vending machine from my Swarthmore talk, what Rullmann & Beck later called "flexible functional application"]

This *gives* you the set {*John left*, *Mary left*, *Bill left*, *Sue left*, …} but without any explicit operator-variable structure in the syntax. In fact, he points out:

Although standard English word-order places the interrogative word or phrase (or the main one, if there are more than one) first, with inversion of the verb, **there is no real need for an order different from that appropriate to indicatives**. So let us assume that no special rules about word-order are needed. (48).

(This is the approach Rullmann & Beck 1997, and then I, picked up.)

One point worth mentioning: how do we know it is a question? Hamblin says:

Pragmatically speaking a question sets up a choice-situation between a set of propositions, namely, those propositions that count as answers to it. (48)

i.e. stipulate: When faced with an utterance that is a set of propositions, choose one.

Another point worth mentioning: There is an additional level of complexity we may have to worry about someday. Namely: possible answers may be different in different possible worlds. (Ooo… my brain hurts!) Heim (1994) gave the example *What subway line runs to the airport?* where in some worlds, there are 4 subway lines, in some there are 5, …

Hamblin's reply (and maybe what ours should be): Let's artificially treat certain facts are fixed and only consider possible worlds which do not vary along those dimensions. At least until we are more secure with our semantics.

We would like to think that the phrase 'what dog' could be treated as an interrogative proper name denoting the set of *dogs*, and that 'what dog walks with Mary' has as answers just the set '*x* walks with Mary' where '*x*' is the name of a *dog*. But the composition of the set of dogs does not necessarily remain constant from universe to universe: **in some universes Rover may be a horse**, and Mary herself a dog. I have taken the attitude that when someone answers 'what dog walks with Mary' with 'Rover' he states not merely that Rover walks with Mary but also implicitly that Rover is a dog, and hence that he states the conjunction. (51).

Karttunen, Lauri (1977). Syntax and semantics of questions. *Ling & Phil* 1:3–44.

More like that tree–semantics mapping drawn above.
Built on certain intuitions from Hamblin 1973, but explores *much* further.

Point One: Embedded questions are questions too.
Point Two: Rather than having a question denote all its *possible* answers, we should have them denote all its *true* answers.

(12) a. John told Mary that Bill and Susan passed the test.
b. John told Mary who passed the test.

Karttunen says *questions denote the set of propositions expressed by their true answers*. So if Mary and John left but Bill didn't, *John left*, *Mary left*, and *John and Mary left* but not *Bill left* all among the propositions in the question denotation. This is to get (12).

It isn't completely clear whether this is a contentful distinction; depends on what you take *propositions* to be. If a proposition is something of type <st>, then it doesn't make any sense to talk of "true propositions"—truth depends on the evaluation world. Any non-contradictory proposition is true in *some* world.

Point Three: Single & multiple-*wh*-questions have the same syntactic distribution. Therefore, we don't want to assign them different semantic types. [I don't adopt this entirely myself—e.g., pair-list vs. single pair readings]

Karttunen's system (roughly)
- forms "Proto-questions" with C, setting up the λp.p= part.
  The proposition has pronouns in place of *wh*-words.
- *Wh*-words are introduced in a series;
  the first one appears sentence initially, the pronoun is deleted.
  later ones simply replace their pronouns.
  [this part is just about pronunciation—
  they all have interpretive effects "at the top"]
- *Wh*-words are interpreted as indefinites (i.e. *what* is like *something*).

Advance notice on upcoming stuff…
Pesetsky (1987), Nishigauchi (1990), Watanabe (1992), Legendre et al (1995)

Pesetsky (1987): Arguing for a syntactic level of LF by distinguishing *two kinds* of *wh*-in-situ, those which move "at LF" and those which don't. If movement is interpreted, we need a way for non-movement to be interpreted; introduces "unselective binding," adopts "LF pied piping," assumes Subjacency constrains LF movement.

Nishigauchi (1990): Sympathetic to Pesetsky 1987, outlines "LF pied piping" approach in more detail. Kind of long, primarily syntax. Partly an argument that Subjacency constrains LF movement.

Watanabe (1992): Argues that Japanese is just like English, moving one *wh*-word overtly—except that what gets moved is phonologically null. Re-arguing that Subjacency does *not* constrain LF movement.

Legendre et al (1995): Optimality approach, mainly to English, Chinese, and Bulgarian. My interest was in looking at how this analysis connects to the things we've been talking about (distinguishing differences in content from differences in terminology), including questions of the role of semantic interpretation.