Connecting meaning and structure in questions: 
Paul Hagstrom
Some consequences of
knowing who bought what
Johns Hopkins University
University of Connecticut, Feb. 26, 1999

The macroissue:
How closely correlated is the syntactic and semantic structure in questions?
What can we learn about one from the other?

The microissue:
The availability of “pair-list” readings in multiple questions.

(1) Q. Who bought what?
A. John bought a book, Sue bought a record, and Bill bought a tape. PLA
   ?? John bought a book. ??SP

(2) a. dare-ga nani-o katta no?
    who-NOM what-ACC bought Q
    ‘Who bought what?’ (PL, SP)

   b. dare-ga nani-o katta ?
    who-NOM what-ACC bought
    ‘Who bought what?’ (*PL, SP)

Questions: • What are the properties of pair-list questions?
           • What are the structural correlates?
           • How do they translate into the semantics of pair-list questions?

The plan:
• Approaching the semantics of single questions.
• Extending to a semantics for pair-list questions.
• In pursuit of (2):
  • What is the semantic contribution of Q? A look at indefinites.
  • The syntax of Q in questions
  • The semantic effect of the interrogative complementizer
  • A proposal for pair-list readings
  • Structural implications, predictions about properties of pair-list readings.
  • A point about answerhood.

I. The semantics of questions & Flexible Functional Application

(3) To know the meaning of a sentence is to know its truth conditions.
(4) To know the meaning of a question is to know what counts as an answer.

At least a first approximation (roughly Hamblin’s 1958 view):
A question provides (say, denotes) the set of things which count as an answer.

(5) [ Who left? ] contains that Mary left
    that John left
    but not that it is raining
    or that a man is in the garden

   i.e. [ Who left? ] = {that Mary left, that John left, … }

Assumption is that information answers a question, not an utterance—
so this is a set of propositions, not a set of sentences.

(6) {p : p = that x left where x ∈ people\_{\text{human}}}

(7) \( \lambda p \exists x \in \text{people}_{\text{human}} \cdot p = \text{that x left}. \)

How do we arrive at this set starting from “Who left”? Hamblin’s idea:
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.
(Hamblin 1973:48)

(8) [ who ] = people\_{\text{human}} = \{John, Mary, … \}
   cf. [ John ] = John

Now left takes a type <e> argument (witness: John left), but [ who ] is type <et>.

Hamblin’s idea: Evaluate the predicate (left) on each of the individuals in the set.
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. (Hamblin 1973:48)

(9) left ([ who ]) = left ( {John, Mary, …} )
   = [left (John), left (Mary), …]
   = [that John left, that Mary left, …]

Adopted & clarified by Rullmann & Beck 1997 as FLEXIBLE FUNCTIONAL APPLICATION.
III. Knowing who bought what

We just derived the set of propositions like $y \text{ bought } x$ for $x$ things, $y$ people.

**Proposal:** Answering a question means picking one of the propositions.

This provides an explanation for (17b). Each proposition picks out a pair, you can pick only one. Hence, single pair.

(17) a. dare-ga nani-o katta no? 
Japanese
who-NOM what-ACC bought Q
‘Who bought what?’ (PL, SP)

b. dare-ga nani-o katta?
who-NOM what-ACC bought
‘Who bought what?’ (*PL, SP)

But what about (18)? You have to pick more than one?

(18) Who bought what?

a. ?? John bought gum. (?SP)
b. John bought gum, Mary bought coffee, and Sue bought pencils. (PL)

Assuming you have to pick more than one is not an explanatory constraint, something else must be going on in (18).

SP and PL questions seem to have different distributions. What kind of thing is a pair-list question?

**Proposal:** A pair-list question is a set of questions, each of which you answer.

So, if people$_\text{context} = \{\text{John, Mary, Bill, Sue}\}$, then $[[\text{Who bought what?}]]$ is $\{\text{What did John buy?}, \text{What did Mary buy?}, \text{What did Bill buy?}, \text{What did Sue buy?}\}$

• How do we get a set of questions from $\text{Who bought what?}$—FFA alone cannot do it.
• We need something else. (17) tells us it correlates with ‘Q’—so what’s Q do?
### III. What is the semantic contribution of Q?

**Goal:** Determine what ‘Q’ contributes to the meaning, to help explain (17).

Q and wh-words in non-questions: (side note ka=no)

(19) Taroo-ga nani-ka-o katta.

Japanese

Taro-NOM what-Q-ACC bought

‘Taro bought something.’

Assuming these are the same Q and wh-words as we saw before, we can now “bootstrap” a hypothesis about [ Q ] by factoring out [ nani ] and [ katta ] from (19).

Notice: (19) isn’t a question.

[ nani ] is a set, but it isn’t triggering FFA.

So, [ katta ] is not getting a set of arguments, just a plain argument.

(20) 

\[ \text{Taroo} \quad \rightarrow \quad \text{katta} \]

\[ \text{nani} \quad \rightarrow \quad \text{ka} \]

[ ka ] is responsible for turning the set [ nani ] into a single argument for [ katta ].

It takes a set (type <et>) and returns an individual (type e).

What individual? Simple hypothesis: A member of the set.

In other words [ ka ] is acting like a choice function.

The meaning of (19) is (with this choice function thing in mind):

(21) \( \exists f. \text{bought'}(\text{Taroo}, f(\text{nani})) \)

‘There is a function f such that Taro bought the element from nani.’

‘There is an x you can pick from nani such that Taro bought x.’

So, maybe [ ka ] is a choice function variable, then (bound by later Existential Closure).

Well, no—not quite.

Here’s why: nanika actually behaves like a real quantifier not like a variable.

(22) a. If an article, is published in LI, John usually reads it,

b. John reads most articles that are published in LI.

usually, … [if … article, … ] … it.


MIT Press-NOM article-ACC published-if John-NOM usually it-ACC read

‘If MIT Press publishes an article, John usually reads it.’

So, that’s how variables act—and it is different from how real quantifiers act:

\[ [\text{ka}] = \lambda p, \lambda w \exists f. p(f)(w) \]

where \( p \) is a proposition-but-for-a-choice-function \( \langle \text{et,e}, \text{st} \rangle \)

\( w \) is a possible world \( \langle \text{st} \rangle \).

\( f \) is a choice function \( \langle \text{et,e}, \text{f(D)e} \rangle \).

so \( [\text{ka}] \) itself is of type \( \langle \text{et,e}, \text{st} \rangle, \langle \text{st} \rangle \).

or, more perspicuously, \( \langle \text{cp, e} \rangle \)

(30) \( [\text{ka}] = \lambda p, \lambda w \exists f. p(f)(w) \)

where \( p \) is the type of a proposition \( \langle \text{et,e}, \text{st} \rangle \)

and \( c \) is the type of a choice function \( \langle \text{et,e} \rangle \)

(24) a. * If everything (submitted) is published in LI, John (usually) reads it,


(25) * MIT Press-ga hotondo-no ranbun-o syuppansureba

MIT Press-NOM most-GEN article-ACC published-if

John-ga taitei sore-o yomu.

John-NOM usually it-ACC read

(‘If most articles, are published by MIT Press, John usually reads it.’)

(26) * MIT Press-ga dono-ranbun-mo syuppansureba

MIT Press-NOM which-article-MO published-if

John-ga taitei sore-o yomu.

John-NOM usually it-ACC read

(‘If every article, is published by MIT Press, John usually reads it.’)

Here’s the point: nanika acts like dono ranbun mo, hotondo-no ranbun and not like ranbun.

nanika acts like a real quantifier, not like a variable.

(27) * MIT Press-ga nani-o syuppansureba

MIT Press-NOM something-ACC published-if

John-ga taitei sore-o yomu.

John-NOM usually it-ACC read

(‘If something, is published by MIT Press, John usually reads it.’)

One more contrast leaning toward this conclusion:

(28) a. dare-ka-ga kita.

who-Q-NOM came.

‘Someone came.’

b. \( \exists x \in \text{people}_{\text{context}}. \text{came}'(x). \)

c. \( \exists f. \text{came}'(f(\text{people}_{\text{context}})). \)

\( \exists f. \text{came}'(f(\text{people}_{\text{context}})) \) is (basically uncontroversially) inherently quantificational.

\( \exists x \in \text{people}_{\text{context}}. \text{came}'(x) \) is (basically uncontroversially) inherently quantificational.

\( \text{ka} \) and \( \text{mo} \) seem to be the same kind of thing—so \( \text{ka} \) is also inherently quantificational.

**Proposal:** \( \text{ka} \) is an existential quantifier over choice functions.

(30) \[ [\text{ka}] = \lambda p, \lambda w \exists f. p(f)(w) \]

where \( p \) is a proposition-but-for-a-choice-function \( \langle \text{et,e}, \text{st} \rangle \)

\( w \) is a possible world \( \langle \text{st} \rangle \).

\( f \) is a choice function \( \langle \text{et,e}, \text{f(D)e} \rangle \).

so \( [\text{ka}] \) itself is of type \( \langle \text{et,e}, \text{st} \rangle, \langle \text{st} \rangle \).

or, more perspicuously, \( \langle \text{cp, e} \rangle \)

(30) \[ [\text{ka}] = \lambda p, \lambda w \exists f. p(f)(w) \]

where \( p \) is the type of a proposition \( \langle \text{et,e}, \text{st} \rangle \)

and \( c \) is the type of a choice function \( \langle \text{et,e} \rangle \)
How this definition for \textit{ka} (‘Q’) in \textit{nanika} gets the right meaning…

\begin{equation}
\text{Taroo} \quad \text{katta} \quad \text{nanika} \quad \text{<e,t> nani ka} \quad \text{<<et,e>,p>,p>}
\end{equation}

Type mismatch—FFA can’t help. But this is just the “quantifier in object position” problem. Solution? QR.

\begin{equation}
\text{Taroo} \quad \text{ka} \quad \text{<e,t> nani t ka} \quad \text{<<e,t>,e>,p>,p>}
\end{equation}

This evaluates just like we’d like it to:

\begin{itemize}
\item[(a)] \( t_u = f \)  
\item[(b)] \( \text{nani} = \text{things}_{\text{context}} \)  
\item[(c)] \( \text{nani} \ t_u = f(\text{things}_{\text{context}}) \)  
\item[(d)] \( \text{katta} = \lambda x \lambda y \lambda w. \ \text{bought'}(y, x) \) in \( w \)  
\item[(e)] \( \text{katta nani} \ t_u = \lambda y \lambda w. \ \text{bought'}(y, f(\text{things}_{\text{context}})) \) in \( w \)  
\item[(f)] \( \text{Taroo katta} \text{ nani} \ t_u = \lambda f. \ \text{bought'}(\text{Taroo}, f(\text{things}_{\text{context}})) \) in \( w \)  
\item[(g)] \( \text{ka} = \lambda p. \ \lambda w \exists f. \ p_0(f) \) (w)  
\item[(h)] \( (31) \) \( \lambda = \lambda w \exists f. \ \text{bought'}(\text{Taroo}, f(\text{things}_{\text{context}})) \) in \( w \)  
\end{itemize}

\textbf{Status report:} We have proposed a semantics for Q that works for indefinites. We wanted this so we could figure out how the presence of Q figures into the availability of the pair-list reading in Japanese.

Now, we’ll turn to questions: First, a comment about the \textit{syntax} of Q in questions. Then, a go at the semantics of questions in light of the syntax. And finally, back to pair-list questions with a proposal.

IV. \textbf{The syntax of Q in questions} (see Hagstrom 1998, chapters 1–4)

Turns out: \textit{ka} in questions is syntactically a lot like \textit{ka} in declaratives with indefinites.

\begin{itemize}
\item[(a)] \text{Taroo-ga nani-o katta no?} \hspace{1cm} ‘What did Taro buy?’
\item[(b)] \text{Taroo nani-t_Q katta no?} \hspace{1cm} ‘QR’ but:
\begin{itemize}
\item overt
\item moves higher
\end{itemize}
\end{itemize}

Historical evidence from Premodern Japanese shows Q in this internal position.

\begin{itemize}
\item[(34)] \text{tare-ka mata hanatathibana-ni omoi-idem} \hspace{1cm} \text{Premodern Japanese}
\item[(35)] \text{mokak d} \hspace{1cm} \text{Sinhala}
\end{itemize}

Crosslinguistic evidence (Sinhala) shows analogous Q particle in internal position.

Movement of Q sensitive to “relativized minimality”-type effects.

\item[(36)] \text{?? dare} \hspace{1cm} \text{Japanese}
\end{itemize}

\textbf{The point:} Even in questions, the tail of Q’s chain is by the \textit{wh}-word. Being a choice function, it will take the \textit{wh}-word as an argument.
V. Connecting meaning and structure in questions, \[ C_{\text{interrogative}} \]

Connecting the semantics we got from the indefinites and the structure we got from the preceding syntactic proposal, what do we get as the semantics for questions?

**Same** between indefinite declarative and \(wh\)-question:

Syntax: Some movement of \(ka\) (covert QR, overt movement into CP).

Semantically: \(Wh\)-element in situ, denoting a set of individuals, argument of \(t_a\).

**Different** between an indefinite declarative and a \(wh\)-question:

Semantically: We end up with a set of propositions, not a proposition.

Syntax: \(C_{\text{interrogative}}\)

So, let’s tie the semantic function to the syntactic presence of \(C_{\text{interrogative}}\).

Syntactic assumption about the landing site of \(ka\) in questions.

\(ka\) moves (clitic-like?) to adjoin to \(C_{\text{interrogative}}\)—forming a constituent.

\[ t_a \rightarrow C_{\text{interrogative}} \rightarrow ka \]

Compositional implication: \(C_{\text{interrogative}}\) should take \(ka\) as its first argument, then the IP as its second argument, and should return a set of propositions.

**Target:** What we want to end up with is something like this:

\[ ka \rightarrow \text{IP} \rightarrow \text{those } p \text{ we can form by } p_i(f) \text{ for some } f. \]

\[ C_{\text{interrogative}} \rightarrow \lambda Q \lambda p_i . \lambda p . \exists f . p = p_i(f) \]

But the central meaning component of \(ka\), \(\exists f\) appears here—Let’s try to factor that out...

Somewhat complex & sneaky—(this is a technical correction to the version in my thesis).

\[ C_{\text{interrogative}} \rightarrow \lambda Q \lambda p_i . \lambda p . \left( \lambda g \lambda w [ p = p_i(g) ] \right) \neq \emptyset \]

(somewhat inelegant, uses vacuous \(\lambda w\) to get T from tautology, F from contradiction)

**Take-home point:**

\(ka\) is responsible for existential quantification over choice functions.

\(C_{\text{interrogative}}\) is responsible for abstraction over propositions.

VI. Back to pair-list questions—using both \(ka\) and FFA

*Here’s where we are:* Our original problem was the PL availability in (43).

It correlates with \(Q\), we have a semantics for \(Q\).

We’ve seen how \(Q\) interacts with \(C_{\text{interrogative}}\) to form sets.

We’ve also seen how FFA forms sets to get (43b).

(and it doesn’t form sets of sets—needed for a PL question)

(43) a. \(\text{dare-ga nani-o katta no?}\)  
\(\text{Japanese: Taro-who bought what-ACC?}\)

‘Who bought what?’ (PL, SP)

b. \(\text{dare-ga nani-o katta?}\)  
\(\text{Japanese: Taro-who bought what-ACC?}\)

‘Who bought what?’ (*PL, SP)

We have two ways to derive sets of propositions in single questions:

- Flexible Functional Application
- \(ka + C_{\text{interrogative}}\)

**Idea:** We can get a set of sets of propositions (a set of questions) by using both FFA and \(ka + C_{\text{interrogative}}\).
Let’s try it. Replace Taroo with dare in the example we just did.

(44) Taroo-ga nani-o katta no?
    ‘What did Taroo buy?’

   (that Taroo bought α, that Taroo bought β, …) if thingscontext = {α, β, …}

(45) dare-ga nani-o katta no?
    ‘Who bought what?’

The spirit of Flexible Functional Application says:
“Give something a set of arguments, and you’ll get a set of what you would have gotten with a simple argument.”

So we expect a set of questions like what did x buy? for the x’es in I dare I.

   { (that A bought α, that A bought β, …), (that B bought α, that B bought β, …), … } if thingscontext = {α, β, …} and peoplecontext = {A, B, …}

This of course is what we’d hoped for—that’s the pair-list question who bought what?

A technical obstacle: To do this, we need to specify how we interpret (46):

(46) λf. { Taroo bought f(thingscontext), Hanako bought f(thingscontext), … } 

What we want is for this to work just like FFA—that is, we want it to yield:

(47) λf.Taroo bought f(thingscontext), λf.Hanako bought f(thingscontext), … }

So we need to state a rule to handle this.

(48) FLEXIBLE λ-ABSTRACTION

   [ λi. Φ ]’ = λA. ∃φΦ. A = [ λx. [ Φ ]’[x/α] ] for some x not occurring in Φ.

   where a) Φ is a set (type <μ>), b) the result is composable.

This will get us from (46) to (47).
Exhaustiveness properties of lists (see Dayal 1996)

(51) Which man is playing against which woman?

A: Men = \{John, Bill, Pat\}  [sounds bad in this context]
Women = \{Mary, Sue\}

B: Men = \{John, Bill\}  [sounds ok in this context]
Women = \{Mary, Sue, Pat\}

“Universal force” over the set of men (where \textit{which man} is the uppermost \textit{wh}-word)
Falls out from: • The set of men \{ which man \} generates the set of questions.
• You answer a pair-list question by answering each component question.

VIII. A somewhat subtle consequence concerning answerhood

Let me come back to something which we didn’t pay much attention to before:

$\#^\wedge$ When faced with a set of propositions,
(a single question or a single-pair question)
you pick one as the answer.

We wouldn’t have the \textit{single pair} property without that. So which one do you choose?

(52) $\text{answer}(Q)(w)$ is the unique \( p \in Q \) such that
\[ p(w) \land \forall q \in Q, q(w) \rightarrow p \supset q \]

The \textit{one} proposition in the set of propositions which is:
• true
and • which entails all the other true propositions.

(53) \[ \text{\[ Who left \]} = \{John left, Mary left, …, John+Mary left, … \} = Q \]

If John and Mary both left in \( w \), all three will be true—
but \text{answer}(Q)(w) will be \textit{John+Mary left}.

This doesn’t mistakenly license lists in cases where we want only single-pair answers
because there isn’t an entailment relationship between the members of the list, so the
uniqueness of \textit{answer} isn’t satisfied if more than one pair is true.

(54) \[ \text{\[ Who bought what \]} = \{John bought gum, Mary bought coffee, John+Mary bought gum+coffee, …\} \]

\textit{John+Mary bought gum+coffee} does not imply that \textit{John bought gum}
and \textit{Mary bought coffee}.

And there is no proposition in there like \textit{John bought gum} and \textit{Mary bought coffee},
(which is what would be needed to get something like a list out)

Essentially: “Pick the maximally informative answer \textit{from the set} \( Q \)”
(not quite the same as this notion is used in Beck & Rullmann 1996)

IX. Summarizing…

• We worked out a semantic representation for single-pair & pair-list questions.
• We found a consistent semantic representation to assign to \textit{wh}-words and to \textit{Q}’
• We tied the \textit{syntactic} properties of \textit{Q} to some observed semantic correlates.
• Big picture: We hope we’ve learned something about how questions are formed.
• And: we hope we can use this close match between syntax & semantics to inform both.

References (cited above)—there’s lots of other related stuff that wasn’t mentioned.

Beck, Sigrid, and Hotze Rullmann (1996). Degree questions, maximal informativeness, and exhaustivity.
10th Amsterdam Colloquium.

the “Chomsky Celebration” archive at MIT Press).


Gair, James, and Lelwala Sumangala (1991). What to focus in Sinhala. In the proceedings of the Eastern
States Conference on Linguistics (ESCOL) ’91.

Distributed by MIT Working Papers in Linguistics.


Ogawa, Kunihiko (1977). Where diachronic and synchronic rules meet: A case study from Japanese