On Concealed Questions and Specificational Subjects
Maribel Romero
University of Pennsylvania

1. INTRODUCTION

Concealed Question NPs:
The underlined NPs in (1) are called “concealed questions” because sentences that embed have the same truth-conditional meaning as the corresponding version with a full-fleged embedded interrogative clause, as illustrated in (2).

(1) a. John knows Bill’s telephone number. 
   b. They revealed / announced the winner of the contest. 
   c. The temperature of the lake depends on the season. 
   (Heim 1979)

(2) a. John knows what’s Bill’s telephone number. 
   b. They revealed / announced who won the contest. 
   c. How warm the lake is depends on the season.

Specificational copular sentences are a subtype of copular sentences different from regular predicational copular sentences. Specificational sentences show Connectivity Effects (ConnE), as in (5-9a), and can have two semantically disjoint NPs after the copula, as in (10a), whereas predicational sentences do not: (5-10b). (Higgins 1973, Sharvit 1999)

(3) The number of planets is nine. 
   SPECIFICATIONAL

(4) The number of planets is large. 
   PREDICATIONAL

(5) Binding Theory Principle A:
   a. Specificational: What John1 is is a nuisance to himself1. 
   b. Predicational: What John1 is is a nuisance to him1. 
   * As Specif, ok as Pred

(6) Binding Theory Principle C:
   a. Specificational: * What he1 has claimed in print is that Sue hates John1. 
   b. Predicational: What he1 has claimed in print is important to John1.

(7) Variable binding:
   a. Specificational: The woman no man1 likes is his1 mother. 
   b. Predicational: * The woman no man1 danced with last night invited him1 over.

(8) Opacity:
   a. Specificational: The person John wants to meet is the Dean of the University. 
   ↳ The Dean of the University can be de dicto under want.
b. Predicational: The person John wants to meet is sitting next to the Dean of the University.
\[\ast : \text{The Dean of the University cannot be de dicto under want}.\]

(9) NPI licensing.
a. Specificational: What Pat didn’t buy was any novels.
b. Predicational: What Pat didn’t buy was important to anyone. (Only ok as free choice)

(10) Coordination of semantically disjoint NPs:
a. Specificational: What John saw was a man and a woman.
b. Predicational: # What John saw was a man and a woman.

Goals of this paper:
FIRST, to point out some semantic similarities between concealed questions (CQs) and specificational subjects (i.e., subjects of specificational copular sentences) (SSs):

1. Certain ambiguity that occurs with nested concealed questions (Heim 1979) also obtains with nested specificational subjects: reading A and reading B.

2. Although particle also cannot occur in certain configurations within regular definite NPs with Relatives Clauses, it can appear in the same configurations in specificational subjects (Higgins 1973). We will see that concealed questions pattern like specificational subjects.

SECOND, to sketch a unified analysis of CQs with know and SSs with be that captures those characteristics, making CQs/SSs different from regular referential NPs:

1. Know is an intensional verb wrt its object position and be is an intensional verb wrt its subject position, much like look for is intensional wrt its object position.

2. The, when applied to CQs/SSs, is compatible with any of the complete answer operators: ANS2 (strongly exhaustive answer), ANS1 (weakly exhaustive answer) or ANS3 (mention-one answer).

Implications: for specificational be, the present account offers a compromise between two of the current analyses of specificational sentences in the literature.

  \[\text{NP}_\tau = \text{YP}_\tau\]
  The pre-copular NP, of any type \(\tau\), is interpreted like a regular NP.
  \(\text{Be}\) is taken as the crosscategorial expression of identity (‘=’).

- QUESTION + DELETION ACCOUNT (Ross 72, den Dikken et al 00, Ross 00, Schlenker 01).
  \[\text{Question} = [\text{IP} \quad \text{XP} \quad \text{]}\]
  The pre-copular constituent is a question and the post-copular constituent is its (partially elided) answer.
2. AMBIGUITIES IN CONCEALED QUESTION NPS AND SPECIFICATIONAL SUBJECTS.

2.1. Ambiguities for concealed questions with *know*.

- Semantic background on *know*:
  
  - *Know* + declarative CP:
    
    (11) John knows Mary is tall.
    
    (12) \( \lambda w. \forall w' \in \text{Dox}_j(w) \ [ \text{tall}(m,w') ] \) (… and John has reasons to believe that.)
    
    (13) \([[[\text{know}_{\text{decl}}]]] = \lambda p_{<\text{st}>} \lambda x \lambda w. \forall w' \in \text{Dox}_x(w) \ [ \ p(w')=1 \ ]\)
    
  - *Know* (exhaustive use) + interrogative CP:
    
    (14) John knows who came.
    
    (15) \([[[\text{who came}]]= \lambda w. \{ p: p(w) \ & \ \exists x [ p=\lambda w''.\text{came}(x,w'') ] \} \) (Karttunen 1977)
    
    (16) \([[[\text{know}_{\text{qu}}]]] = \lambda q_{<s,\text{st},<\text{st}>} \lambda x \lambda w. \forall w' \in \text{Dox}_x(w) \ [ q(w') = q(w) \] \) (Heim 1994)
    
    (17) \([[[\text{John knows who came}]]= \lambda w. \forall w' \in \text{Dox}_j(w) \ [ \ { p: p(w') \ & \ \exists x [ p=\lambda w''.\text{came}(x,w'') ] } = \ { p: p(w) \ & \ \exists x [ p=\lambda w''.\text{came}(x,w'') ] } \] \)
    
  - *Know* (exhaustive use) + concealed question NP: individual concept approach.
    
    (18) John knows the capital of Italy.
    
    (19) \([[[\text{the capital of Italy}]]= \lambda P \lambda w. \exists y_{<s,e>} \ [ \ \forall x [ \text{capital-of-Italy}(x) \leftrightarrow x=y ] \ & \ P(y)(w) ] \)  
    
      a. \( \lambda P \lambda w. \exists y_{<s,e>} \ [ \ \forall x [ \text{capital-of-Italy}(x) \leftrightarrow x=y ] \ & \ P(y)(w) ] \)  
    
      b. \( \lambda x_{<s,e>} [ \ \text{capital-of-Italy}(x) ] \)  
    
      or  
    
      \( \lambda w. \exists x_{<s,e>} [ \ \text{capital-of-Italy}(x,w) ] \)
    
    (20) \([[[\text{know}_{\text{quNP}}]]] = \lambda y_{<s,e>} \lambda x \lambda w. \forall w' \in \text{Dox}_x(w) \ [ y(w') = y(w) \ ] \)
    
    (21) \([[[\text{John knows the capital of Italy}]]= \lambda w. \exists y_{<s,e>} \ [ \ \forall x [ \text{capital-of-Italy}(x) \leftrightarrow x=y ] \ & \ \forall w' \in \text{Dox}_j(w) \ [ y(w') = y(w) ] \ ] \)  
    
      a. \( \lambda w. \exists y_{<s,e>} \ [ \ \forall x [ \text{capital-of-Italy}(x) \leftrightarrow x=y ] \ & \ \forall w' \in \text{Dox}_j(w) \ [ y(w') = y(w) ] \ ] \)  
    
      b. \( \lambda w. \forall w' \in \text{Dox}_j(w) \ [ \ t x_{<s,e>} [ \ \text{capital-of-Italy}(x) ](w') = t x_{<s,e>} [ \ \text{capital-of-Italy}(x) ](w) ] \)
    
    (22) Roughly: \( x_{<s,e>} \) knows \( y_{<s,e>} \) at \( w \) iff \( x \) is at \( w \) able to identify the value \( y(w) \) that \( y \) yields when applied to \( w \). (Heim 1979, p. 56)
Example (23) is ambiguous between a Reading A and a Reading B described below. (24) provides yet another reading: Reading C. (Heim 1979)

(23) John knows the price that Fred knows. (Heim 1979)

(24) **Reading A:** \([ N' \text{Concealed.Question}] + \text{Relative Clause} \)
There are several relevant questions about prices: “How much is the milk?”
“How much is the oil?”
“How much is the ham?”

Fred knows the answer to one of these questions, e.g., to the first one.  
John knows the answer to this question too.

(25) **Reading B:** \([ [ N' \text{Concealed.Question}] + \text{Relative Clause} ] \text{Concealed.Question} \)
There are several relevant questions about prices: “How much is the milk?”
“How much is the oil?”
“How much is the ham?”

Fred knows the answer to one of these questions, e.g., to “How much is the milk?”.
Then, there is the “meta-question” asking which of these questions is the one whose answer Fred knows.
John knows the answer to the meta-question. I.e., John knows that the question about prices whose answer Fred knows is “How much is the milk?”.

(26) John knows the price that Fred knows: the price announced yesterday morning.

(27) **Reading C:** \([ [ N' \text{Conc.Question}] + \text{Modifier } \text{Conc.Question} + \text{Relative Clause } ] \text{Conc.Question} \)
There are several relevant questions about prices: “How much is the milk?”
“How much is the oil?”
“How much is the ham?”

The answer to each of these questions was announced yesterday at different times of the day --morning, afternoon, evening, night, etc.--, and it is at stake to find out which question was announced when. That is, the following are relevant meta-questions:

“Which of those questions was announced in the morning?”
“Which of those questions was announced in the afternoon?”
“Which of those questions was announced in the evening?”

Fred knows the answer to the first of the meta-questions (e.g., he correctly believes that the question whose answer was announced in the morning is “How much is the milk?”). Then, we can construct the meta-meta-question asking what of the meta-questions above Fred knows the answer for. The meta-meta-question is:

“What of the meta-questions above does Fred know the answer for?”
John knows the answer to the meta-meta-question. I.e., John knows that the meta-question that Fred knows the answer for is “What of those questions was announced in the morning?”.
General syntax and semantics for NPs that combine with extensional verbs (i.e., extension taking verbs):

(28) the price that Fred knows

\[ \lambda x. \text{price}(x,w') \ \forall w'' \in \text{Dox}_f(w') \ [ x(w'') = x(w') ] \]

\[ \lambda x. \text{price}(x,w') \ \forall w'' \in \text{Dox}_f(w') \ [ x(w'') = x(w') ] \]

\[ \lambda \lambda \text{price}(x,w') \ \forall w'' \in \text{Dox}_f(w') \ [ x(w'') = x(w') ] \]

\[ \text{that}_1 \ \forall w'' \in \text{Dox}_f(w') \ [ (g(1))(w'') = (g(1))(w') ] \]

Fred knows\[ t_1 \]

**ATTEMPT 1:**

(29) John knows the price that Fred knows.

\[ \lambda w. \forall w' \in \text{Dox}_j(w) \ [ \lambda x <s,e> \ \text{price}(x,w') \ \forall w'' \in \text{Dox}_f(w') \ [ x(w'') = x(w') ] \] (w') = \[ \lambda x <s,e> \ \text{price}(x,w') \ \forall w'' \in \text{Dox}_f(w') \ [ x(w'') = x(w') ] \] (w') \]

\[ \lambda w. \forall w' \in \text{Dox}_j(w) \ [ \lambda x <s,e> \ \text{price}(x,w') \ \forall w'' \in \text{Dox}_f(w') \ [ x(w'') = x(w') ] \] (w') = \[ \lambda x <s,e> \ \text{price}(x,w') \ \forall w'' \in \text{Dox}_f(w') \ [ x(w'') = x(w') ] \] (w') \]

**Reading A**

(30) \[ \lambda w. \forall w' \in \text{Dox}_j(w) \ [ \lambda x <s,e> \ \text{price}(x,w') \ \forall w'' \in \text{Dox}_f(w') \ [ x(w'') = x(w') ] \] (w') = \[ \lambda x <s,e> \ \text{price}(x,w') \ \forall w'' \in \text{Dox}_f(w') \ [ x(w'') = x(w') ] \] (w') \]

\[ \lambda w. \forall w' \in \text{Dox}_j(w) \ [ \lambda x <s,e> \ \text{price}(x,w') \ \forall w'' \in \text{Dox}_f(w') \ [ x(w'') = x(w') ] \] (w') = \[ \lambda x <s,e> \ \text{price}(x,w') \ \forall w'' \in \text{Dox}_f(w') \ [ x(w'') = x(w') ] \] (w') \]

**Reading B???

(31) \[ \lambda w. \forall w' \in \text{Dox}_j(w) \ [ \lambda x <s,e> \ \text{price}(x,w') \ \forall w'' \in \text{Dox}_f(w') \ [ x(w'') = x(w') ] \] (w') = \[ \lambda x <s,e> \ \text{price}(x,w') \ \forall w'' \in \text{Dox}_f(w') \ [ x(w'') = x(w') ] \] (w') \]

**Reading B???

(32) Scenario 1:

John wrongly thinks that Fred knows how much the milk costs and that Fred knows no other price question. But, in fact, Fred knows how much the olive oil costs and he knows no other price question. John knows how much the milk is and he doesn’t know how much the oil is.

Formula (31) yields TRUE in this scenario.

But the sentence is false in this scenario under reading B. For it to be true, John would have to (correctly) believe that Fred knows how much the olive oil costs.

(33) Scenario 2:

John correctly thinks that Fred knows how much the milk costs and that Fred knows no other price. But John himself does not know how much the milk costs.

Formula (31) is yields FALSE in this scenario.

But the sentence is true in this scenario under reading B.

The truth-conditions of (31) do not correspond to reading B.
(34) **Reading A:**
There are several relevant questions about prices: “How much is the milk?”
“How much is the oil?”
“How much is the ham?”
Fred knows the answer to one of these questions, e.g., to the first one.
John knows the answer to this question too.

(35) **Reading A:**

a. $\lambda w. \exists y <s,e> [ \forall x [ price(x,w) \& \forall w'' \in Dox_{fred}(w) [ x(w'') = x(w)] \leftrightarrow x=y ] \& \forall w'' \in Dox_{john}(w) [ y(w'') = y(w) ] ]$

b. $\lambda w. \forall w' \in Dox_j(w)$
   $[ t_{<s,e>} [ price(x,w) \& \forall w'' \in Dox_{i}(w) [ x(w'') = x(w)] (w') =
   t_{<s,e>} [ price(x,w) \& \forall w'' \in Dox_{i}(w) [ x(w'') = x(w)] (w) ] ]$

$\Rightarrow$ We use only the denotations we already have:

a. $[[price_1]](x_{<s,e>})(w) = 1$ iff $price_1(x,w)$

b. $[[know_1]](x_{<s,e>})(z)(w) = 1$ iff $\forall w' \in Dox_{z}(w) [ x(w') = x(w) ]$

(36) **Reading B:**
There are several relevant questions about prices: “How much is the milk?”
“How much is the oil?”
“How much is the ham?”
Fred knows the answer to one of these questions, e.g., to “How much is the milk?”.
Then, there is the “meta-question” asking which of these questions is the one whose answer Fred knows.
John knows the answer to the meta-question. I.e., John knows that the question about prices whose answer Fred knows is “How much is the milk?”.

(37) **Reading B:**

a. $\lambda w. \exists y_{<s,e>} [ \forall x [ price(x,w) \& \forall w'' \in Dox_{fred}(w) [ x(w'') = x(w)] \leftrightarrow x=y ] \& \forall w'' \in Dox_{john}(w) [ y(w'') = y(w) ] ]$

b. $\lambda w. \forall w' \in Dox_j(w)$
   $[ t_{<s,s,e,>} [ price(x,w) \& \forall w'' \in Dox_{i}(w) [ x(w'') = x(w)] (w') =
   t_{<s,s,e,>} [ price(x,w) \& \forall w'' \in Dox_{i}(w) [ x(w'') = x(w)] (w) ] ]$

$\Rightarrow$ We would need:

a. $[[price_2]](x_{<s,s,e,>})(w) = 1$ iff $price_2(x,w)$

b. $[[know_2]](x_{<s,s,e,>})(z)(w) = 1$ iff $\forall w'' \in Dox_{z}(w) [ x(w'(w'') = x(w'(w)) ]$

c. $[[know_3]](x_{<s,s,e,>})(z)(w) = 1$ iff $\forall w'' \in Dox_{z}(w) [ x'(w'') = x'(w) ]$
The problem with ATTEMPT 2:

Let us concede that we have all the lexical entries listed above.

We can combine these lexical entries in several ways. Two possibilities are these:

i. John and Fred know the same exact question, i.e., we plug in the same number of w variables for Fred’s \( x \) and for John’s \( y \). Reading A

ii. John knows a higher meta-question than Fred, i.e., we plug in more w variables for Fred’s \( x \) than for John’s \( y \). Reading B

Interestingly, possibility (iii) is missing:

iii. Fred knows a higher meta-question than John, i.e., we plug in more w variables for John’s \( y \) than for Fred’s \( x \). Reading B’

We know that possibility (iii) is missing because, if it available, (38) would be true in scenario (40), having the inverse reading of reading B. But the scenario (40) does not make (38) true.

(38) John knows the price that Fred knows.

(39) **Reading B’**:

a. \( \lambda w. \exists_{\less<\less>_{se}} [ \forall \lambda [ \text{price}(x(w),w) \& \forall w'' \in \text{Dox}_{\text{fred}}(w) [x(w'') = x(w)] \leftrightarrow \lambda=y ] \]

b. \( \lambda w. \forall w' \in \text{Dox}_{\text{j}}(w) [ t_{\less<\less>_{se}} \text{price}(x(w),w) \& \forall w'' \in \text{Dox}_{\text{j}}(w) [x(w'') = x(w)] (w)(w') = t_{\less<\less>_{se}} \text{price}(x(w),w) \& \forall w'' \in \text{Dox}_{\text{j}}(w) [x(w'') = x(w)] (w)(w) ] \]

(40) Scenario:

John knows that the milk costs $1.97. Fred knows that the price that John knows is the price of milk (and Fred doesn’t know any other meta-question about prices), though Fred does not know the price of milk itself.

**Given the lexical entries that we need to generate readings A and B, there is no way to rule out compositionally the unavailable reading B’**.
ATTEMPT 3: *know* as an intensional verb.

(41) John knows the price that Fred knows.
   a. Reading A: John and Fred know the same price, e.g. the price of milk.
   b. Reading B: John knows what is the price known by Fred.

(42) John is looking for the unicorn with the longest horn.
   **Parallel to Reading B:** John wants in w that he finds in w’ the individual that is the
   unicorn with the longest horn in w’, whichever that may be.

   *†* *look for* takes as its argument INTENSION of the DP.  (Cf. Zimmermann 1993)

(43) John is looking for the unicorn Fred wanted to catch: the one with the longest horn.
   **Parallel to Reading A:** John and Fred are looking for the same (type of) unicorn, e.g, the
   one with the longest horn.

   *†* *look for* takes as its argument EXTENSION of the DP, which is an intensional object.

(44) We’ll use only the following lexical entries:
   a. \[ [[\text{price}]](x_{se})(w) = 1 \iff \text{price}(x,w) \]
   b. \[ [[\text{know}]](x_{se})(z)(w) = 1 \iff \forall w’ \in \text{Dox}_z(w) [ x(w’) = x(w) ] \]
   c. \[ [[\text{know}]](x_{se})(z)(w) = 1 \iff \forall w” \in \text{Dox}_z(w) [ x(w”) = x(w) ] \]

(45) the price that Fred knows

\[
\lambda w. \text{DP } \lambda x_{se} [ \text{price}(x,w) & \forall w’ \in \text{Dox}_f(w) [ x(w’) = x(w) ] ]
\]

\[
\lambda P_{<se>,<st>} \lambda w. \text{the NP } \lambda x_{ss,e} \lambda w. \text{price}(x,w) & \forall w’ \in \text{Dox}_f(w) [ x(w’) = x(w) ]
\]

\[
\lambda x_{se} [ P(x)(w)=1 ]
\]

\[
\lambda x_{se} \lambda w. \text{price}(x,w)
\]

\[
\lambda x_{ss,e} \lambda w. \forall w’ \in \text{Dox}_f(w) [ x(w’) = x(w) ]
\]

\[
\lambda t_{<se>} [ P(x)(w)=1 ]
\]

\[
\lambda x_{<se>} \lambda z \lambda w. \forall w’ \in \text{Dox}_f(w) [ x(w’) = x(w) ]
\]

\[
\text{Fred VP}
\]

\[
\lambda w_{0} \forall w’ \in \text{Dox}_j(w_{0})
\]

\[
\lambda x_{se} [ \text{price}(x,w) & \forall w’ \in \text{Dox}_f(w) [ x(w’) = x(w) ] ]
\]

\[
\lambda w_{0} \forall w’ \in \text{Dox}_j(w_{0})
\]

Reading B: *know* takes the INTENSION of the DP as its argument.

(46) Reading B: *know* takes the INTENSION of the DP as its argument.

(47) Reading A: *know* takes the EXTENSION of the DP as its argument.
2.2. Ambiguities for specificational subjects with be.

(48) The price that Fred thought was $1.29 was (actually) $1.79.
   a. “The question whose answer Fred thought was ‘$1.29’ has as its real answer ‘$1.79’.”

(49) **Reading A:**
There are several relevant questions about prices:
   “How much is the milk?”
   “How much is the oil?”
   “How much is the ham?”
For one of these questions --e.g., the first one—, Fred thought the answer was ‘$1.29’.
But the actual answer to this question is ‘$1.79’.

(50) What (/ The price that) Fred thought was $1.29 was the price of milk.
   a. “The question whose answer Fred thought was ‘$1.29’ is “How much is the milk?’.”

(51) **Reading B:**
There are several relevant questions about prices:
   “How much is the milk?”
   “How much is the oil?”
   “How much is the ham?”
For one of these questions, Fred thought the answer was ‘$1.29’.
Then, there is the “meta-question” asking which of these questions is the one whose
answer Fred thought was $1.29.
The answer to the meta-question is “How much is the milk?”.
That is, Fred thought that the price of milk is $1.29.

(52) Scenario for (53)-(54): A group of 3-year old boys from the Ukraine was given in
adoption to several families in Barcelona. The director of the adoption program
encouraged the biological relatives of each boy to keep in touch with him by writing
letters, telling them thought that they should not identify themselves using their name,
family relationship or address.

After a couple of years, the boys have developed some hypotheses on who every secret
writer may or may not be. For example, no boy thinks that the one who writes to him the
least can possibly be his mother. In fact, they are all right about that, since, for every boy,
the one who writes to him the least is his uncle.

(52) The anonymous writer that no boy1 thinks can possibly be his1 mother is (in fact) his1 uncle.
   **Reading A**

(53) The anonymous writer that no boy1 thinks can possibly be his1 mother is the one who
writes to him1 the least.
   **Reading B**

(50’) **Reading B:** *be* takes the INTENSION of the DP as its argument.
\[
\lambda w_0. \ [ \lambda w. \, \lambda x <se> \ [\text{price}(x,w') \land \forall w' \in \text{Dox}_{\text{fred}}(w) [x(w') = \$1.29]] \] (w_0) = \\
\lambda x <se> \ [\text{price-of-milk}(x,w_0)]
\]

(48’) **Reading A:** *be* takes the EXTENSION of the DP as its argument.
\[
\lambda w_0. \ [ \lambda x <se> \ [\text{price}(x,w_0) \land \forall w' \in \text{Dox}_{\text{fred}}(w_0) [x(w') = \$1.29]] \] (w_0) = \$1.79
\]
3. Also in Specificalional Subjects and in Concelealed Question NPs.

- Higgins (1973: 10): “[56] has the specificalional reading, [57] the predicational reading”.

(56) What he is *also* pointing at is a kangaroo. Specificalional
(57) What he is pointing at is *also* a kangaroo. Predicational

- Exhaustive use: it seems that embedded also is acceptable (though not perfect for everybody) in predicational sentences if the referent of the containing regular definite NP exhausts the additional relevant individuals:

(58) Joanna liked three songs out of this compilation. She liked these two a lot: 99 Luftballons is and Fragezeichen. These are from the singer’s early period. The song she also liked is from a later period. Predicational Subject

(59) Joanna liked three songs out of this compilation. She liked these two a lot: 99 Luftballons is and Fragezeichen. These are from the singer’s early period. The song she also liked is Carpe Diem. Specificalional Subject

- Non-exhaustive use. Here the asymmetry hinted at by Higgins surfaces clearly: a definite NP containing also (associated with the trace) can have a non-exhaustive reading when it is a SS but not when it is the subject of a predicational sentence. We note, furthermore, that CQs pattern like SSs in allowing this non-exhaustive reading. The same pattern obtains in Spanish. Same for German, except that CQs cannot be tested.

(60) Subject of a predicational sentence:
A: I heard that Carlos was wearing a very nice hat yesterday that everybody admired.
B: What he was also wearing didn't suit him at all. And, on top of that, it was really expensive.
A: I hadn't heard anything about that...
\[\Rightarrow\text{Implies everything other than the hat that Carlos was wearing looked bad on him.}\]

(61) Specificalional Subject:
A: I heard that Carlos was wearing a very nice hat yesterday that everybody admired.
B: What he was also wearing was a pair of tight orange pants that didn't suit him at all. And, on top of that, they were really expensive.
A: I hadn't heard anything about that...

(62) Concealed Question NP:
A: I heard that Carlos was wearing a red hat yesterday that everybody admired.
B: Do you know the garment he was also wearing? A pair of tight orange pants that didn't suit him at all. And, on top of that, they were really expensive.
A: I hadn't heard anything about that...
(63) Subject of a predicational sentence: SPANISH.
A: He oído que Carlos ayer llevaba un sombrero de ala ancha
Have-1s heard that Carlos yesterday was-wearing a hat of wing wide
que causó la admiración de todos los presentes.
that caused the admiration of all the those-present
B: (??) Lo que también llevaba no le sentaba nada bien. Y, además,
The that also was-wearing-3s not to-him suit at-all well. And, on-top-of-that,
era carísimo.
was-3s very-expensive
A: De eso no había oído nada...
About that not have-1s heard nothing…
† Implies everything other than the hat that Carlos was wearing looked bad on him.

(64) Specificational Subject: SPANISH.
B: Lo que también llevaba eran unos pantalones naranjas estrechos
The that also was-wearing-3s were some pants orange tight
que no le sentaban nada bien. Y, además, eran carísimos.
that not to-him suit at-all well. And, on-top-of-that, were-3p very-expensive

(65) Specificational Subject: SPANISH.
B: Sabes lo que también llevaba? Unos pantalones naranjas estrechos
Know-2s the that also was-wearing-3s? Some pants orange tight
que no le sentaban nada bien. Y, además, eran carísimos.
that not to-him suit at-all well. And, on-top-of-that, were-3p very-expensive

(66) Subject of a predicational sentence: GERMAN.
A: Ich habe gehört, dass Carlos gestern einen Hut getragen hat,
I have heard that Carlos yestersay a hat worn has
den alle Anwesenden bewundert haben.
that all those-present admired have
B: */ Was er auch noch getragen hat stand ihm überhaupt nicht gut.
What he also still worn has suited to-him absolutely not well
Außerdem war es sehr teuer.
Besides, was it very expensive

(67) Specificational Subject: GERMAN.
B: Was er auch noch getragen hat war eine grüne Hose die ihm überhaupt
What he also still worn has was some green pants that to-him absolutely
nicht gut stand.
not well suited.

(68) Concealed Question NP: GERMAN.
B: Weisst du was er auch noch getragen hat? Eine grüne Hose, die ihm
Know you what he also still worn has? Some green pants that to-him
überhaupt nicht gut stand.
absolutely not well suited.
Note that interrogative clauses with a non-exhaustive reading can also host this use of *also*. This reading has been called “mention-some reading”.

(69)  A: I don’t want to go to Sarcone’s just to buy Spanish ham. It’s too far.
     B: I know where you can also get Spanish ham: At DiBruno’s.

Idea:
*The* (or definiteness in general), when applied to CQs/SSs, is compatible with the nominal counterpart of any of the *complete* answer operators ANS2 (strongly exhaustive answer), ANS1 (weakly exhaustive answer) or ANS3 (mention-some answer).

Degrees of exhaustivity with embedded interrogatives. (Heim 1994, Beck-Rullmann 1999)

(70)  John knows who came.
     ⇒ Strongly exhaustive: For every x that came, John knows x came, and he knows that nobody other than those came.

(71)  John was surprised at who came (… but not at who didn’t come).
     ⇒ Weakly exhaustive: For every x that came, John was surprised that x came.

(72)  John knows where you can buy Spanish ham.
     ⇒ Mention-some: For some x such that you can buy Spanish ham at x, John knows that you can buy Spanish ham at x.

(73)  \[ [[\text{who came}]] = \lambda w. \{ p: p(w)=1 \land \exists x \ [p=\lambda w''.\text{came}(x,w'')]] \]
     = e.g. \[ \lambda w. \{ \lambda w''.\text{came}(\text{pat},w''), \lambda w''.\text{came}(\text{sue},w'') \} \]

(74)  \[ [[\text{know ANS1}}]] = \lambda Q_{s<,st,t>}, \lambda x, \lambda w. \forall w' \in \text{Dox}_x(w) \ [ Q(w') \supseteq Q(w) ] \]
     WEAKLY EXH.

(76)  \[ [[\text{know ANS2}}]] = \lambda Q_{s<,st,t>}, \lambda x, \lambda w. \forall w' \in \text{Dox}_x(w) \ [ Q(w') = Q(w) ] \]
     STRONGLY EXH.

(77)  \[ [[\text{know ANS3}}]] = \lambda Q_{s<,st,t>}, \lambda x, \lambda w. \exists q_{s<}\ [ q \in Q(w) \land \forall w' \in \text{Dox}_x(w) \ [ q \in Q(w') ] ] \]
     MENTION-SOME

1 Heim’s actual ANS1 and ANS2 operators and Beck-Rullmann’s ANS3 operator are the following:

(75')  \[ \text{ANS1}(Q_{s<,st,t>},w) = \cap [[Q]](w) \]
     WEAKLY EXH.

(76')  \[ \text{ANS2}(Q_{s<,st,t>},w) = \lambda w'. [\text{ANS1}(Q,w') = \text{ANS1}(Q,w)] \]
     STRONGLY EXH.

(77')  \[ \text{ANS3}(Q_{s<,st,t>},w) = \lambda P_{s<,st,t>}. \exists p_{s<}\ [ P(w)(p) \land p \in [[Q]](w) ] \]
     MENTION-SOME
(78) \[ [[\text{John knows ANS1 who came}]] \]
\[
= \lambda w. \forall w' \in \text{Dox}_\text{john}(w) [\{p: p(w')=1 \& \exists x [p=\lambda w''.\text{came}(x,w'')]) \supseteq \{p: p(w)=1 \& \exists x [p=\lambda w''.\text{came}(x,w'')]\} ]
\]
\[= \text{e.g. } \lambda w. \forall w' \in \text{Dox}_\text{john}(w) [\{p: p(w')=1 \& \exists x [p=\lambda w''.\text{came}(x,w'')]) \supseteq \{\lambda w''.\text{came}(\text{pat},w''), \lambda w''.\text{came}(\text{sue},w'')\} ]
\]
\[= \text{e.g. } \lambda w. \forall w' \in \text{Dox}_\text{john}(w) [\text{came}(\text{pat},w') \& \text{came}(\text{sue},w') ]
\]

(79) \[ [[\text{John knows ANS2 who came}]] \]
\[
= \lambda w. \forall w' \in \text{Dox}_\text{john}(w) [\{p: p(w')=1 \& \exists x [p=\lambda w''.\text{came}(x,w'')]) = \{p: p(w)=1 \& \exists x [p=\lambda w''.\text{came}(x,w'')]\} ]
\]
\[= \text{e.g. } \lambda w. \forall w' \in \text{Dox}_\text{john}(w) [\{p: p(w')=1 \& \exists x [p=\lambda w''.\text{came}(x,w'')]) = \{\lambda w''.\text{came}(\text{pat},w''), \lambda w''.\text{came}(\text{sue},w'')\} ]
\]
\[= \text{e.g. } \lambda w. \forall w' \in \text{Dox}_\text{john}(w) [\text{came}(\text{pat},w') \& \text{came}(\text{sue},w') \& \neg \exists x [ x \neq \text{pat} \& x \neq \text{sue} \& \text{came}(x,w') ]
\]

(80) \[ [[\text{John knows ANS3 who came}]] \]
\[
= \lambda w. \exists q_{<}\exists [ q \in \{p: p(w)=1 \& \exists x [p=\lambda w''.\text{came}(x,w'')]) \& \forall w' \in \text{Dox}_\text{john}(w) [q(w')] ]
\]
\[= \text{e.g. } \lambda w. \exists q_{<}\exists [ q \in \{\lambda w''.\text{came}(\text{pat},w''), \lambda w''.\text{came}(\text{sue},w'')\} \& \forall w' \in \text{Dox}_\text{john}(w) [q(w')] ]
\]

Extending degrees of exhaustivity to CQs and SSs: ONE POSSIBLE IMPLEMENTATION

(81) \[ [[\text{the persons that came}]] = \lambda w. \sigma x: \text{came}(x,w)\]
\[= \text{e.g. } \lambda w. \text{pat+sue}\]

- Know + CQ:

a. Juan sabe las personas que han venido.
b. John knows (epistemically) the persons that came.

(83) \[ [[\text{KnowCQ ANS1}]] = \lambda x_{<}\lambda x_{<}\lambda w. \forall w' \in \text{Dox}_x(w) [ y(w') \geq y(w) ] \]
\[\text{WEAKLY EXH.}\]

(84) \[ [[\text{KnowCQ ANS2}]] = \lambda x_{<}\lambda x_{<}\lambda w. \forall w' \in \text{Dox}_x(w) [ y(w') = y(w) ] \]
\[\text{STRONGLY EXH.}\]

(85) \[ [[\text{KnowCQ ANS3}]] = \lambda x_{<}\lambda x_{<}\lambda w. \forall z \in \mathbb{E} [ z \leq y(w) \& \forall w' \in \text{Dox}_x(w) [ z \leq y(w') ] ] \]
\[\text{MENTION-SOME}\]
John knows the persons that came

\[ \lambda w. \forall w' \in \text{Dox}_{\text{John}}(w) [ \sigma x: \text{came}(x, w') \geq \sigma x: \text{came}(x, w) ] \]

= e.g. \[ \lambda w. \forall w' \in \text{Dox}_{\text{John}}(w) [ \sigma x: \text{came}(x, w') \geq \text{pat+sue} ] \]

John knows the persons that came

\[ \lambda w. \forall w' \in \text{Dox}_{\text{John}}(w) [ \sigma x: \text{came}(x, w') = \sigma x: \text{came}(x, w) ] \]

= e.g. \[ \lambda w. \forall w' \in \text{Dox}_{\text{John}}(w) [ \sigma x: \text{came}(x, w') = \text{pat+sue} ] \]

John knows the persons that came

\[ \lambda w. \exists z_e [ z \leq \sigma x: \text{came}(x, w) \land \forall w' \in \text{Dox}_{x}(w) [ z \leq \sigma x: \text{came}(x, w') ] ] \]

= e.g. \[ \lambda w. \exists z_e [ z \leq \text{pat+sue} \land \forall w' \in \text{Dox}_{x}(w) [ z \leq \sigma x: \text{came}(x, w') ] ] \]

• Be + SS:

The persons that (also) came were Pat and Sue.

\[ \lambda x, \lambda y <\text{ss}\lambda w'. [ y(w') \geq x ] \text{ WEAKLY EXH.} \]

\[ \lambda x, \lambda y <\text{ss}\lambda w'. [ y(w') = x ] \text{ STRONGLY EXH.} \]

\[ \lambda x, \lambda y <\text{ss}\lambda w'. \exists z_e [ z = x \land z \leq y(w') ] \text{ MENTION-SOME} \]

The persons that came were Pat and Sue

\[ \lambda w'. [ \sigma x: \text{came}(x, w') \geq \text{pat+sue} ] \]

The persons that were Pat and Sue

\[ \lambda w'. [ \sigma x: \text{came}(x, w') = \text{pat+sue} ] \]

The persons that (also) came were Pat and Sue

\[ \lambda w'. \exists z_e [ z = \text{pat+sue} \land z \leq \sigma x: \text{came}(x, w') ] \]

Some (or indefiniteness in general), when applied to a CQ/SS, stands for partial answer (or, more generally, for unsatisfying answer).

I’ll tell you a problem he has.

One problem he has is his taste on ties.

Watching a contest on a local TV channel, where they require a mention-some answer.

A: They just asked the participant where one can buy Spanish ham in Philadelphia.

B: I know the answer! At DiBruno’s.

B’: # I know an answer. At DiBruno’s.

A: How did Sally do on question 3 on the quiz? Did she leave it blank too?

B: Well, she gave an answer for this one, but it is far too incomplete.
4. CONCLUSIONS AND IMPLICATIONS.

Conclusions:

1. *Know* is an intensional verb wrt its object position and specificational *be* is an intensional verb wrt its subject position, much like *look for* is intensional wrt its object position. Hence, NP objects of *know* and NP subjects of specificational *be* are simply interpreted as NPs that contribute an intensional object, which obtains either from the intension of the NP or from the (individual concept) extension of the NP.

2. The meaning of *know* and *be* has the same flexibility observed in verbs taking embedded interrogatives. This allows for a strongly exhaustive, a weakly exhaustive and a mention-some interpretation when they combine with definite intensional NP arguments.

A compromise between two leading accounts of specificational sentences:

  \[ \text{NP}_\text{τ} = \text{YP}_\text{τ} \]
  The pre-copular NP, of any type \text{τ}, is interpreted like a regular NP.
  \text{Be} is taken as the crosscategorial expression of identity (‘=’).

- **QUESTION + DELETION ACCOUNT** (Ross 72, den Dikken et al 00, Ross 00, Schlenker 01).
  \[ \text{Question} = \text{[IP —–XP——]} \]
  The pre-copular constituent is a question and the post-copular constituent is its (partially elided) answer.

- **COMPROMISE:**
  CQs and SSs are NPs, syntactically and semantically.
  But the meaning of *know* and *be*, when applied to them, yields parallel semantics to the semantics of embedded interrogatives.

Grouping together *know+CQs and *be+SSs as a special kind of construction with NPs also mirrors the distribution of matching effects in Catalan and Spanish free relatives: regular free relatives must obey matching, whereas CQs and SSs do not need to. The account of this phenomenon is left for future research.

- Hirschbühler-Rivero (1983a) note that (regular) free relatives must obey certain morpho-syntactic matching between the relative pronoun and the function of the entire free relative. The data below are from Catalan, but the same facts hold for Spanish.

(60) Invito qui / la que has invitat. (Hirschbühler-Rivero 1983a)
I-invite who / the that you-have invited.
“I invite who you have invited.”

(61) * Invito amb qui / la que te n’aniràs. (Hirschbühler-Rivero 1983a)
I-invite with whom / the that you-will-leave.
* “I invite (the person) with whom you’ll leave.”
(62) Vaig ballar amb qui / la que vaig venir.  
I-PAST dance with whom / the that I-PAST come.  
“I danced with whom I came.”

• Hirschbühler-Rivero (1983b) further note that CQ free relatives are not subject to this matching requirement:

(67) Sé amb qui / la que t’aniràs.  
I-know with whom / the that you’ll-leave.  
“I know (epistemically) the person that you’ll leave with.” I.e.,  
“I know who you will leave with”

(68) Observa en lo que ens vam posar. (Hirschbühler-Rivero 1983b)  
Observe in the that us we-PAST put  
“Observe the (situation) we got ourselves into”. I.e.,  
“Observe what we got ourselves into.”

• Interestingly, SSs pattern with CQs: they do not obey this matching requirement either.

(69) Amb qui / la que vaig ballar va ser amb la Joana.  
With whom / the that I-PAST dance was with the Joana.  
“With whom I danced was with Joana.”

(70) En qui / el que més has de pensar és en tu mateix.  
In who / the that most you-have to think is in yourself.  
“In who you have to think most is in yourself.”

Some references
Schlenker, P. 2001. A Note on the Connectivity Problem, USC ms., extensively revised version of ‘Clausal Equations (A Late Note on the Connectivity Problem)’, MITWPL 39.