Wh Movement and QR: Deriving their Differences

Leiden University
17 March 2010

Kyle Johnson
University of Massachusetts at Amherst
kbj@linguist.umass.edu

Two properties a theory of movement should deliver are (1).

(1) If movement relates α to positions π and π', then

a. TERSENESS:
None of α may be pronounced in both π and π'.
illustration:
  Which solution to a problem does she require?
* Which solution to a problem does she require solutions to a problem?

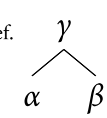
b. RECONSTRUCTABILITY:
Some of α may be semantically interpreted in the position it is not pronounced in.
illustration:
  Which story about her$_1$ should none of the women$_1$ forget?
* Which story about her$_1$ should someone who knows none of the women$_1$ forget?

Chomsky's (1995) "copy theory" of movement has elements that allow for an account of these properties.

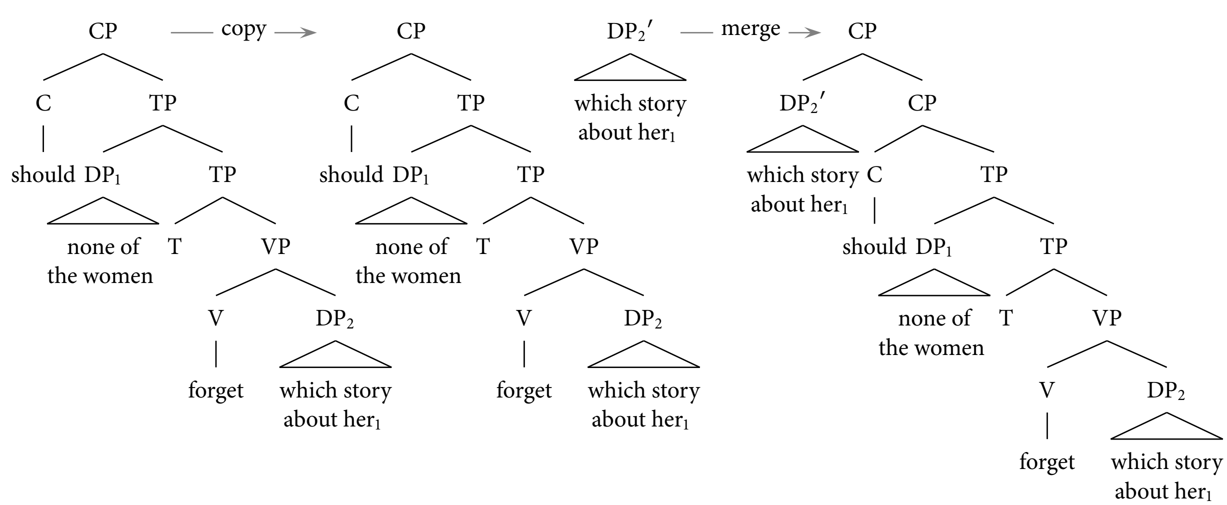
(2) MOVE $\alpha =_{\text{def.}}$

a. copy $\alpha = \alpha'$, and

b. MERGE(α', β)

(3) MERGE$(\alpha, \beta) =_{\text{def.}}$  (linear order not determined)

Illustrative derivation that derives reconstruction.

(4) 

Nunes (1999, 2004) combines the copy theory of movement with a linearization scheme to derive Terseness. He proposes that the copies produced by movement must be treated as if they are identical terms by the linearization algorithm.

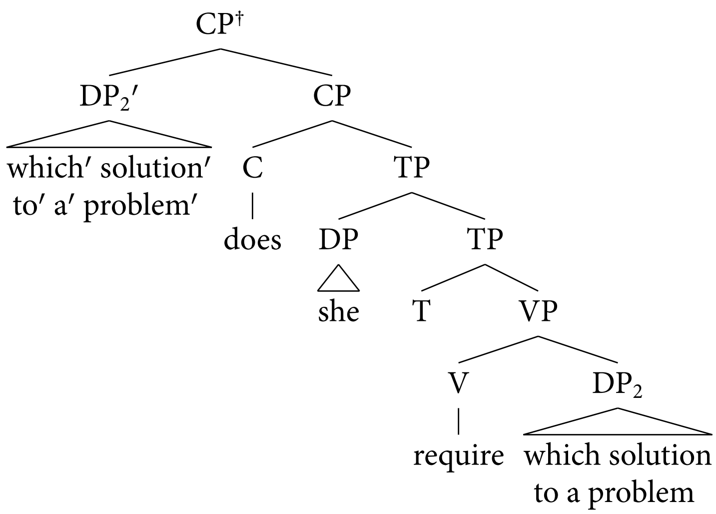
(5) Let *lin*(α) = σ represent the linearization algorithm (*lin*) applying to a phrase (α) to form a set of ordered pairs (σ), containing terminals in α ordered by precedence ("<"). σ must satisfy:

a. Every vocabulary item in α must be in an ordered pair with every other vocabulary item in α. (Totality)

b. For no terminals, τ and π, in α may $\tau < \pi$ and $\pi < \tau$. (Antisymmetry)

c. For any terminals τ, π and κ in α, if $\tau < \pi$ and $\pi < \kappa$ then $\tau < \kappa$. (Transitivity)

stipulation: If π' is part of a copy in α, then *lin* regards π' and the term it was copied from, π, as the same.



(6) $lin(\mathrm{CP}^{\dagger})$ = solution′ < require, require < solution, …

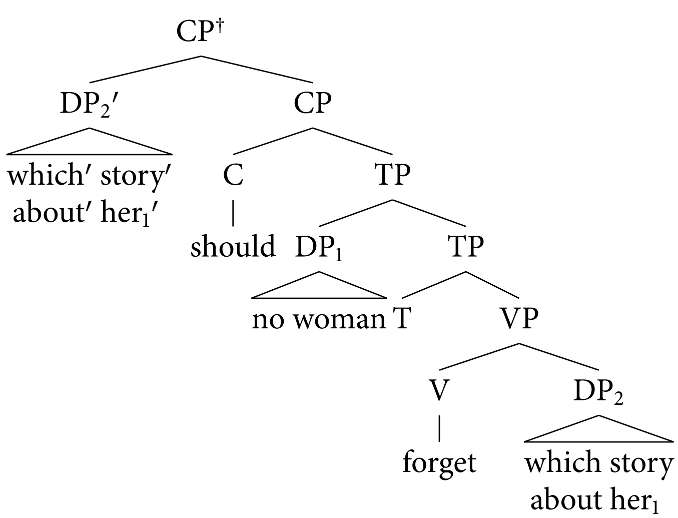
Nunes then suggests that to obey Antisymmetry, there is a procedure that gets rid of the offending ordered pairs. The result is that the moved phrase is pronounced in only one of the positions it occupies.

The difference between QR and Wh Movement that will be derived in this talk is (7).

(7) QR'd material must be semantically interpreted where it is spoken, but Wh moved material is able to be semantically interpreted in only its unspoken position.

We've already seen that Wh Movement allows something to be interpreted in only its non-spoken position. This is true of our example of Reconstructability:

(8)

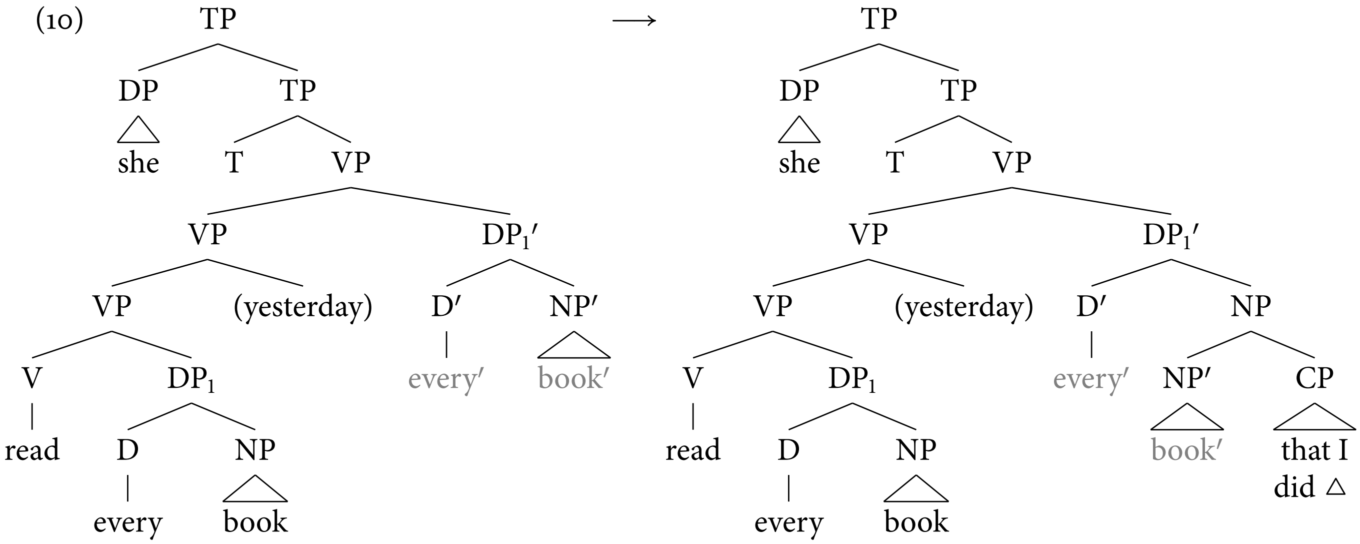


The prepositional phrase *about her* not only can be interpreted in the lower position, it cannot be interpreted in its spoken position.

To illustrate that this isn't possible for QR, I'll use the account of Antecedent Contained Deletion (ACD) in Fox (2002). On that account, ACD is licensed when the relative clause containing the ellipsis has extraposed from the antecedent for that ellipsis.

(9) She [$_{VP}$ read [$_{DP}$ every book]] (yesterday) [$_{CP}$ that I did △]

Fox (and Fox and Nissenbaum 1999) argue that these sorts of extraposition operations are the result of "late merging" a clause into a QR'd DP. (NP: shaded font indicates material that is not pronounced.)

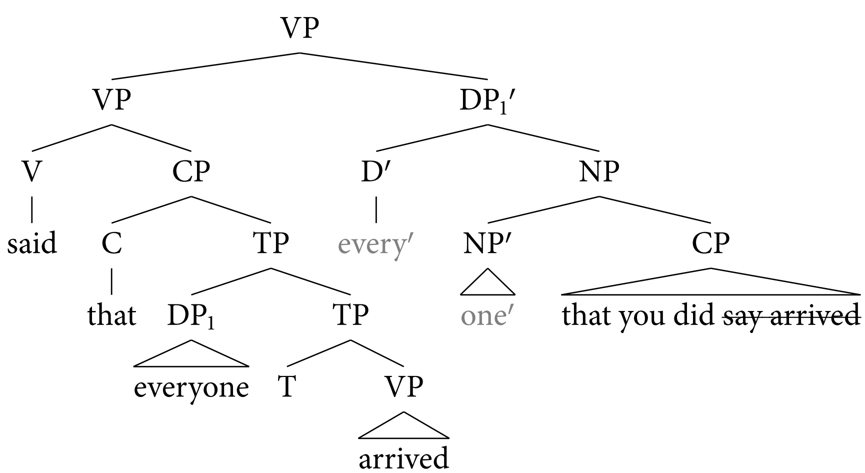
(10)


One of the most interesting arguments on behalf of this analysis is the contrast in (11), from Tiedeman (1995).

(11) a. * I said that everyone you did △ arrived.
b. I said that everyone arrived that you did △.
△ = said that x arrived

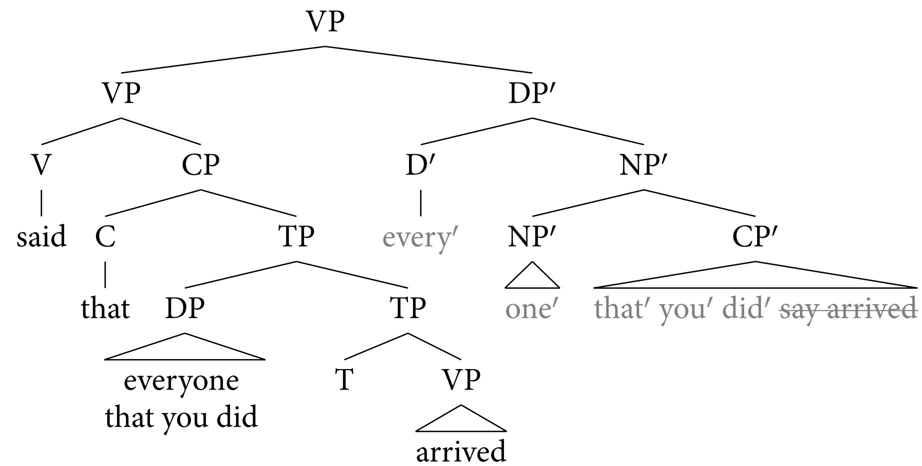
(Fox 2002, (35b), (36b): 77)

The difference is credited to extraposition being able to generate the string in (11b) but not (11a). As a consequence, the representation in (12) is only available for (11b).

(12)


But this assumes that material spoken in a QR'd DP must be interpreted there. For otherwise (11a) should be able to have the representation in (13), and still be well-formed.

(13)

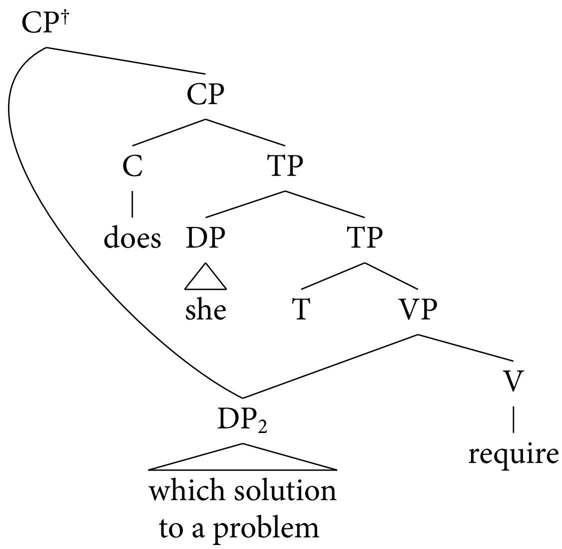


Conclusion: Material spoken in a QR'd DP must be semantically interpreted there.

I will derive this difference by adopting a "remerge" interpretation of the "copy" operation.[1] This will also allow us to remove Nunes's stipulation about how *lin* treats copies.

(14) "Move α" is the name given to MERGE(α, β), when β contains α.

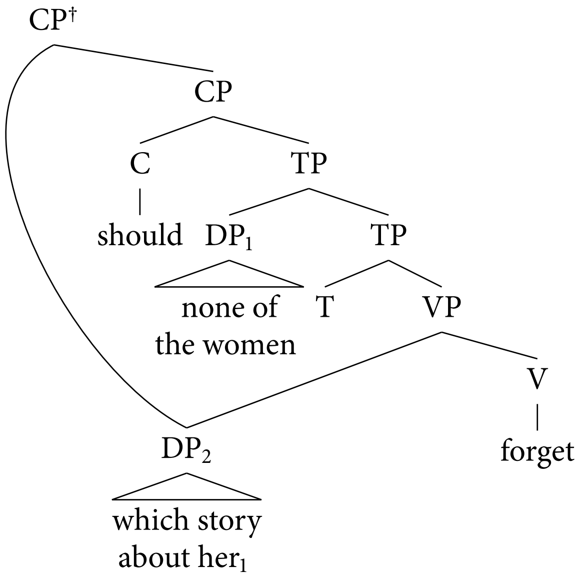
(15)



If *lin* evaluates the moved phrase in all of the positions it occupies in (15), *solution* will follow *require*, because that is what *lin* does with the contents of objects and the verbs they are the objects of, and *solution* will also precede *require*, because that is what *lin* does with the terminals in phrases that occupy the Specifier of a CP and everything else in that CP. That would produce a violation of Antisymmetry, however, and so *lin* is allowed to evaluate the moved phrase in only one of its two positions. Terseness is derived. (We need no special procedure to avoid a violation of Antisymmetry, as on Nunes's proposal.)

[1] As, for instance, in Engdahl (1980), Gärtner (1997, 2001), Starke (2001) and many others.

The remerge interpretation of movement preserves Chomsky's method of deriving Reconstructability.

(16)


A banal definition of c-command will correctly put *her* within the scope of *none of the women*.

A problematic aspect of these representations, however, is that they say that the same phrase must be interpreted differently depending on the positions it occupies: as a variable in its lowest position and as an operator in its highest position. The previous solutions to these problems invent a semantics that does just that. See Engdahl (1980, 1986) and Fox (2003). In both cases, the solution gives the phrase in its lowest position the denotation of a definite description — the kind of definite description that can vary with values given to some other term, like that found in (17).

(17) Every woman who owns a donkey feeds the donkey.

I suggest that our semantics not be enriched in this way, and instead that the syntactic representations resulting from movement simply put a definite description in the trace position. The leading idea is that constituent questions involve a DP that introduces a variable and a question morpheme (Q) that binds off that variable.[2] These two components are transparently involved in many languages.

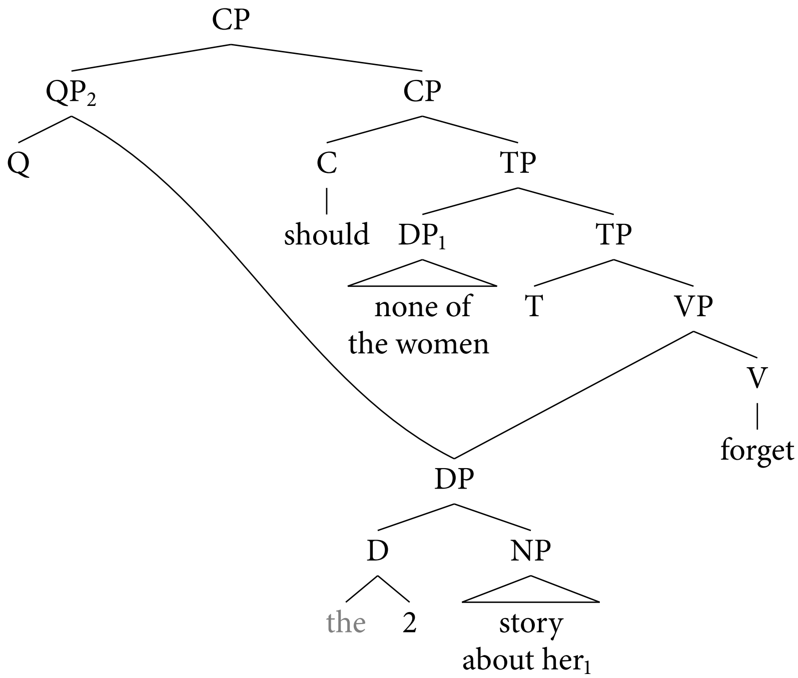
(18) (Kimi-wa) dono-gakusei-ga nattoo-o tabe-tagatte-iru-to omoimasu-ka?
(you-top) which-student-nom natto-acc eat-desirous-be-C think-Q
(Which student do you think wants to eat natto?)

In English, there is no independent morpheme associated with Q, but I will assume it is there nonetheless. The term that introduces the variable I will assume is the definite description that Engdahl and Fox assume makes up the trace of movement.

As in Chierchia (1992) and Elbourne (2005), I will adopt the view that the index used to represent binding of a trace is inside the determiner. (The semantics I will adopt are fashioned after an unpublished handout by Irene Heim for her course with Polly Jacobson at the LSA Institute at MIT several years ago.) An English constituent question has a representation like that in (19).

[2] See Hagstrom (1998, 2000) and Kishimoto (2005).

(19)

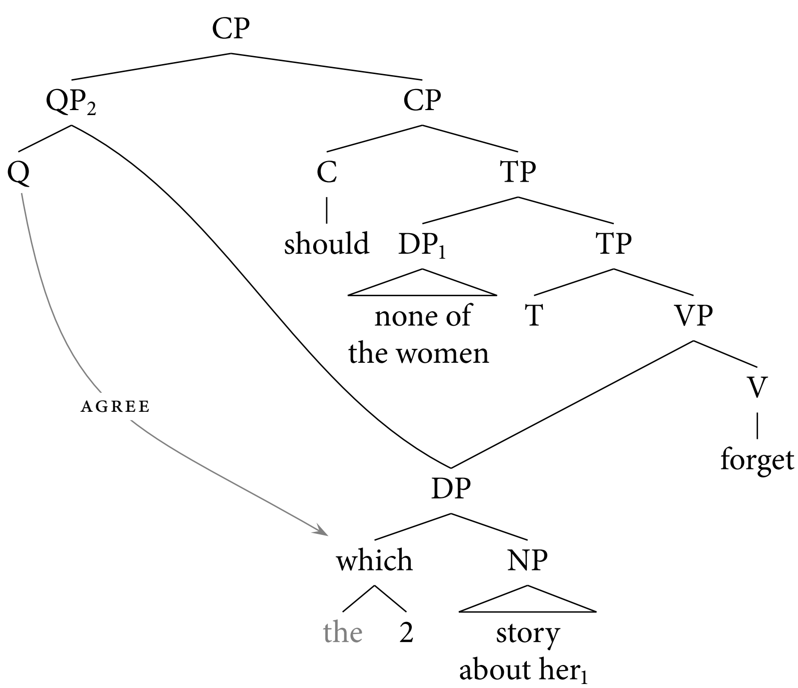


(20) $[\![\text{the}]\!] = \lambda n.\lambda P.n$, only if $P(n) = 1$

The denotation of *the* makes the denotation of the phrase it heads the same as the value of the variable it is in construction with, and introduces the presupposition that the value of this variable will be something that satisfies the NP it combines with.

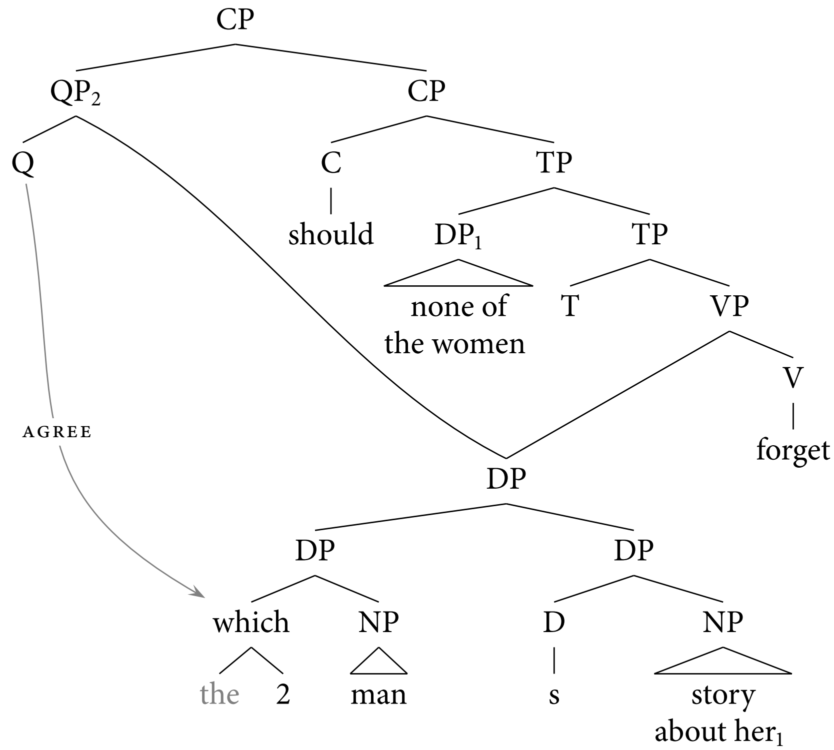
I'll follow Adger and Ramchand (2005) and Cable (2007) and assume that the lexical item *which* is the way D is spelled out under agreement with Q.

(21)



Cable (2007) argues that Q can be separated from the *which* phrase by a short distance, and that this is what gives rise to Pied-Piping.

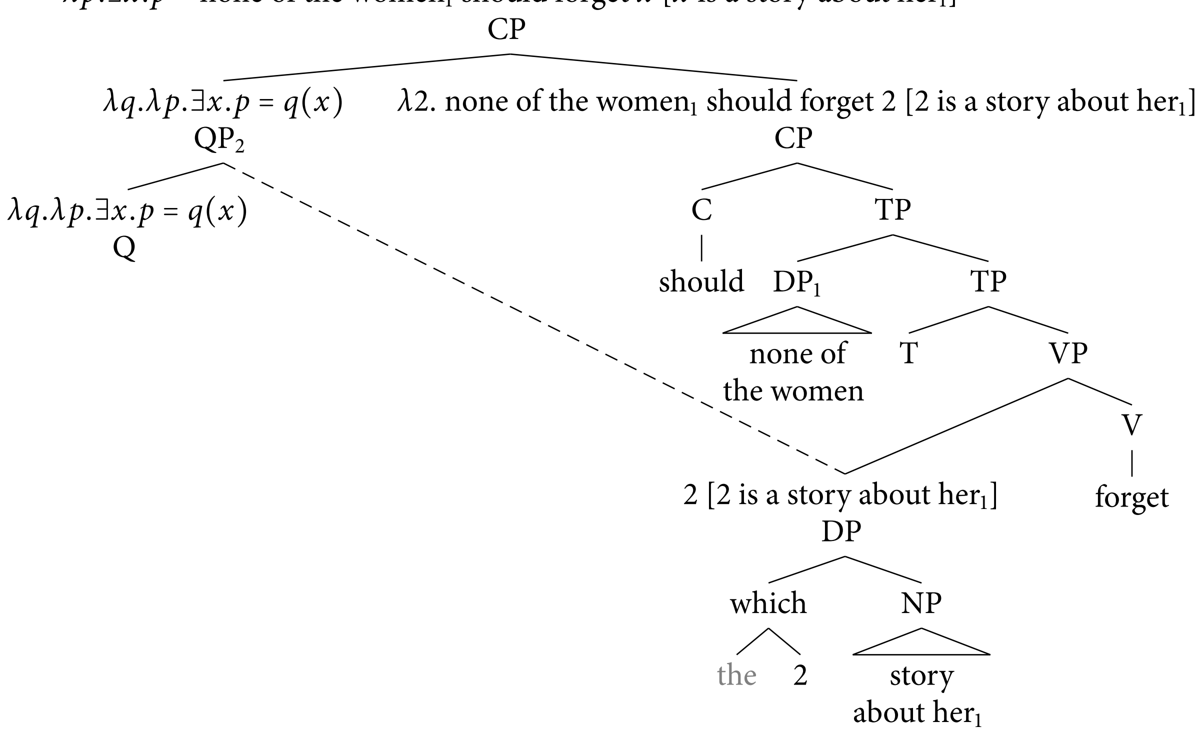
(22) Which man's story about her should none of the women forget?



I'll assume that Q binds off the variable introduced by *which* and produces the question meaning.

(23) $[\![\mathrm{Q}]\!] = \lambda q.\lambda p.\exists x.p = q(x)$

(24) $\lambda p.\exists x.p$ = none of the women$_1$ should forget x [x is a story about her$_1$]



I've enclosed the presupposition introduced by *the* in square brackets. Note that the denotation of Q prevents it from combining semantically with the DP (=*which story about her*) that it has merged with. I indicate this with a dashed line. As long as this DP makes its semantic contribution somewhere, it will obey the Principle of Full Interpretation.

(25) Principle of Full Interpretation
Every term in a phrase marker must semantically combine with at least one of its sisters.

Because Q does not semantically combine with its sister, the QP it heads will have the same denotation as Q. This combines successfully with the CP that is its sister, and forms the question thereby.

We have one last detail before we'll have a working semantics for questions. If x is an individual in (23), then the meaning assigned in (24) isn't quite right. It says that the question seeks the identity of a single individual story which none of the women should forget. But we want the stories that none of the women should forget to vary as a function of the values given to *none of the women*. If the set of women we are quantifying over is Eleanor, Julie and Lisa, then we want (26) to be a possible answer:

(26) Eleanor shouldn't forget Vern's story about her, and Julie shouldn't forget Jim's story about her, and Lisa shouldn't forget Rint's story about her.

The answer in (26) is a set of ordered pairs, then:

(27) (Eleanor,Vern's), (Julie,Jim's), (Lisa, Rint's)

A set of ordered pairs is a relation, and sometimes relations themselves have names. In this case, an alternative way of giving the answer in (26) is (28).

(28) her sweetheart's

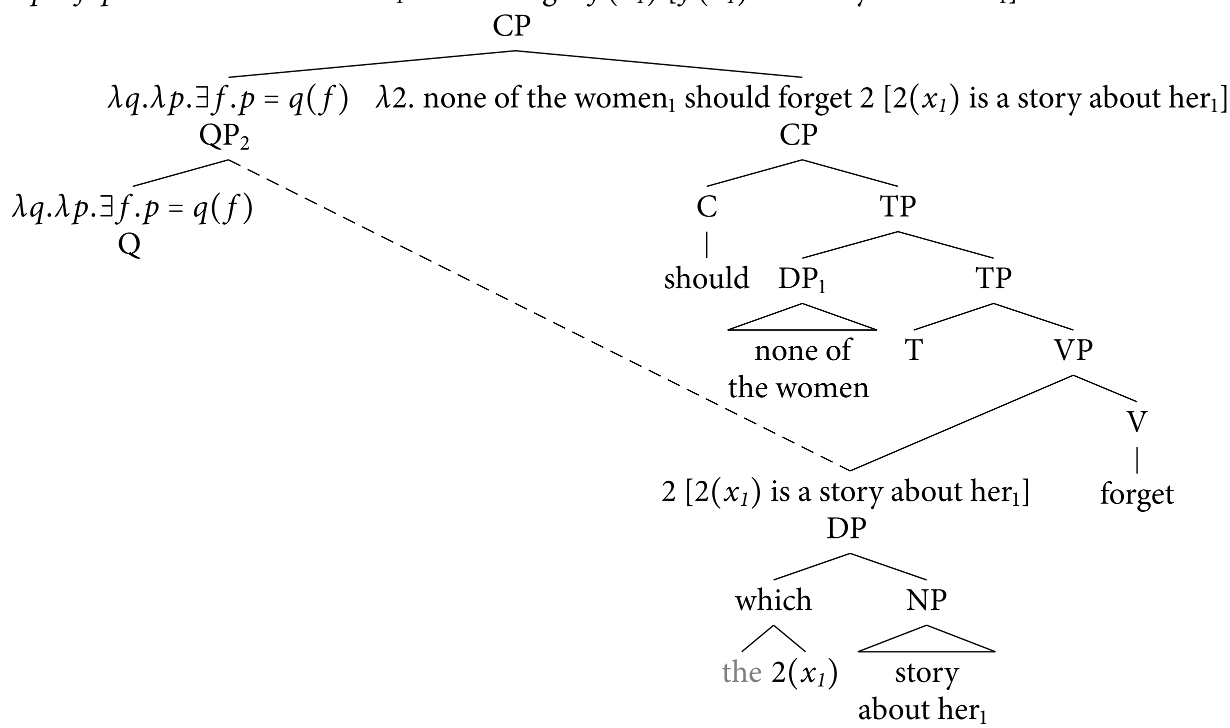
This ability of *wh*-phrases to have values that depend on the values given to some quantifier arises only when that *wh*-phrase is within the scope of that quantifier.

(29) a. Which book should none of the women forget?
her first
b. Which book should someone who dislikes none of the women forget?
* her first

Engdahl (1980) proposed that the variable we've put inside *which* can vary over functions that can pick out a different individual depending on the values given to something that scopes over *which*. If we represent that function with f, then the denotation we want to give to *which story about her should none of the women forget* is (31).

(30) $[\![\mathrm{Q}]\!] = \lambda q.\lambda p.\exists f.p = q(f)$

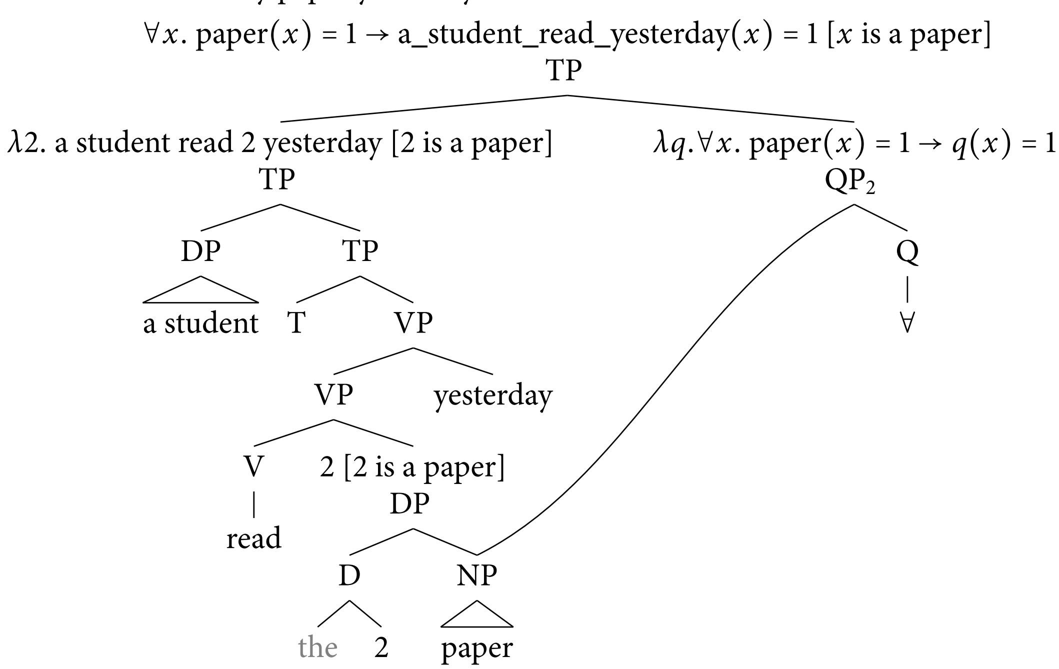
(31) $\lambda p.\exists f.p$ = none of the women$_1$ should forget $f(x_1)$ [$f(x_1)$ is a story about her$_1$]



Let's now consider how QR can be modeled. I'll use a remerge view in order to derive Terseness, and in order to achieve the difference between the variable and the binder, I'll again generate a definite description in the lowest position that is bound by an operator in the higher position. Unlike the case with questions, however, in this case the determiner in the lower position and the quantifier in the higher position must both combine semantically with the NP. We'll have a picture like that in (33).

(32) $[\![\forall]\!] = \lambda p.\lambda q.\forall x.p(x) = 1 \rightarrow q(x) = 1$

(33) A student read every paper yesterday.

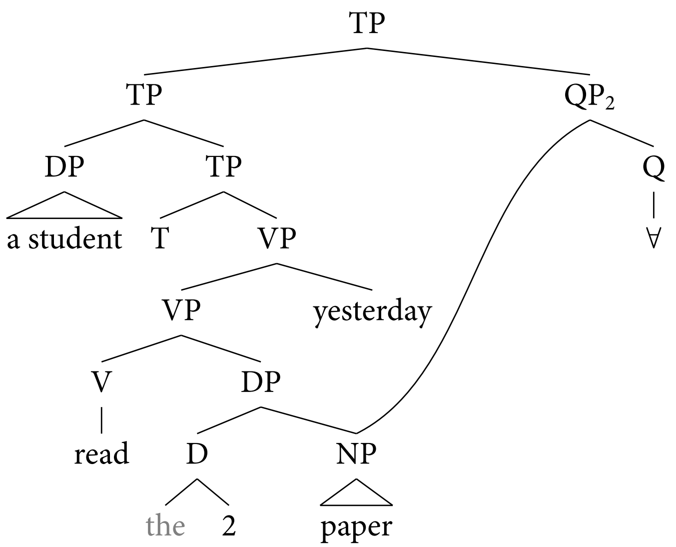


As with the case of questions, we want the form of D to be determined by the quantifier that is in the higher position. When the quantifier is ∀ we'll want D to be spelled-out as *every*, when it's "¬ any" we'll want D to be spelled-out as *no*, and so on. We can't do this with Agree, as Q does not c-command D. I suggest instead that D+Q are fused, in the sense of Distributed Morphology, and matched against the appropriate vocabulary item. Here, concretely, is how that works.

(34) a. The lexical items *every*, *no*, *some* and certain others are mapped onto fused D+Q complexes.
b. Two terminal positions in a phrase marker are fusible just in case the linearization algorithm does not put anything between them.
c. The linearization algorithm assigns to Q and D the leftmost position in the phrases they head.

This will require that *lin* run before QP has merged in, because only then (usually) will Q and D be adjacent and therefore able to fuse.

(35)



If *lin* applies to this, it will spit out the ordered pairs in (36).[3]

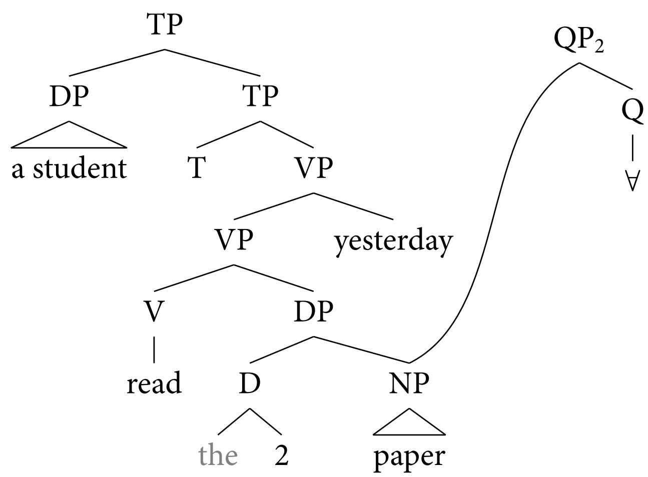
(36)

a < student	student < T	read < D	D < paper	paper < yesterday	yesterday < ∀
a < T	student < read	read < paper	D < yesterday	paper < ∀	
a < read	student < D	read < yesterday	D < ∀		
a < D	student < paper	read < ∀			
a < paper	student < yesterday				
a < yesterday	student < ∀				
a < ∀					

This doesn't put D and Q adjacent to each other (because: D < yesterday and yesterday < Q). And it violates (34) by including paper < Q. Instead, we'll have to run *lin* before QP has merged in:

[3] This assumes that *lin* puts the material in the QP merged with TP to the right of the material in VP. If *lin* instead puts the material in QP to the left of the material in the TP it is merged with, the same problem described below will arise, although in a different way. I am assuming that *lin* has only those two options: to put QP before everything in the phrase it has merged with, or to put it after everything in the phrase it has merged with. There may be a mistake buried in that assumption.

(37)



If we run *lin* on TP and QP, we'll get the ordered pairs in (38).

(38)

$$lin(\text{TP}) = \left\{ \begin{array}{lllll} \text{a} < \text{student} & \text{student} < \text{T} & \text{read} < \text{D} & \text{D} < \text{paper} & \text{paper} < \text{yesterday} \\ \text{a} < \text{T} & \text{student} < \text{read} & \text{read} < \text{paper} & \text{D} < \text{yesterday} & \\ \text{a} < \text{read} & \text{student} < \text{D} & \text{read} < \text{yesterday} & & \\ \text{a} < \text{D} & \text{student} < \text{paper} & & & \\ \text{a} < \text{paper} & \text{student} < \text{yesterday} & & & \\ \text{a} < \text{yesterday} & & & & \end{array} \right\}$$

$$lin(\text{QP}) = \forall < \text{paper}$$

This linearization puts nothing between D and ∀, and so they can fuse. Once they've fused and been mapped onto *every*, it is *every* that will occupy the positions assigned to D and ∀ in (38). After QP and TP have merged, no new ordering statements need to be added to meet the requirement of totality. That's because totality only requires that every vocabulary item in a phrase marker be assigned a position relative to every other vocabulary item, and that will be achieved by (38) (after D and Q fuse) for all the vocabulary items that will be matched to the terminals in (37).

Note, then, that in many cases this will derive the fact that QR gets its material spelled out in the lower of the two positions. We are therefore in a position to derive the difference in how Wh Movement and QR get spelled out.

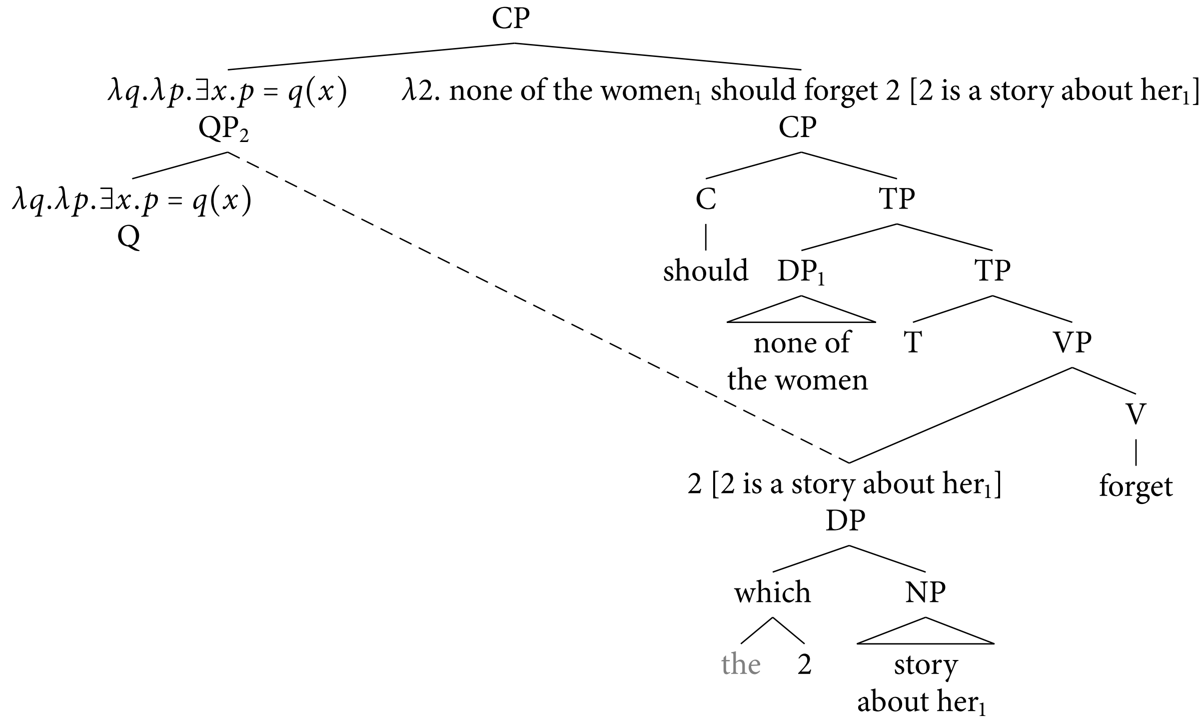
(39) If *lin* is run as late in the derivation as possible, then:
   a. A QR'd DP will be pronounced in the lower position, and
   b. A Wh moved DP can be pronounced in the higher position.

But the difference I promised to deliver is (7)

(7) QR'd material must be semantically interpreted where it is spoken, but Wh moved material is able to be semantically interpreted in only its unspoken position.

We've already seen how the case of Wh Movement works. An interrogative DP can be semantically interpreted entirely in its lower position, and yet be part of a phrase that *lin* puts in a different position. If *lin* aligns the material in *which story about her* so that it precedes everything else in the sentence — a consequence of the fact that the material in Specifier of CP is linearized at the leftmost edge of that CP — then (24) will map onto the right string.

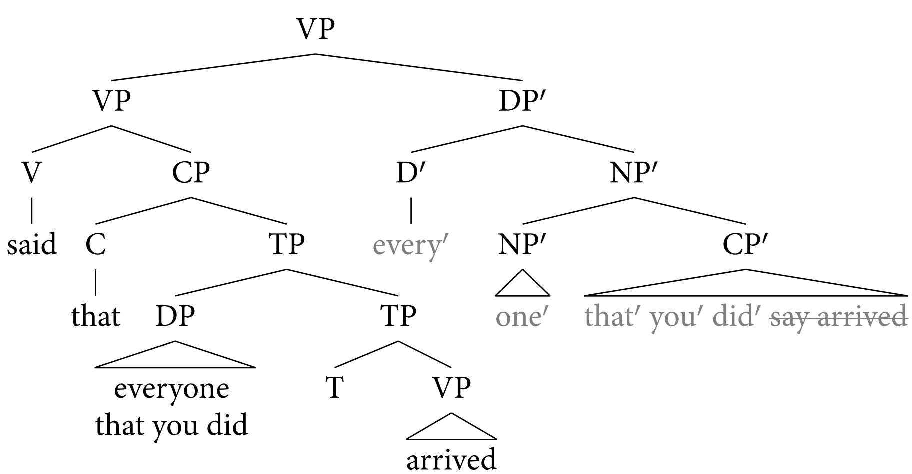
(24) $\lambda p.\exists x.p$ = none of the women$_1$ should forget x [x is a story about her$_1$]



= Which story about her should none of the women forget?

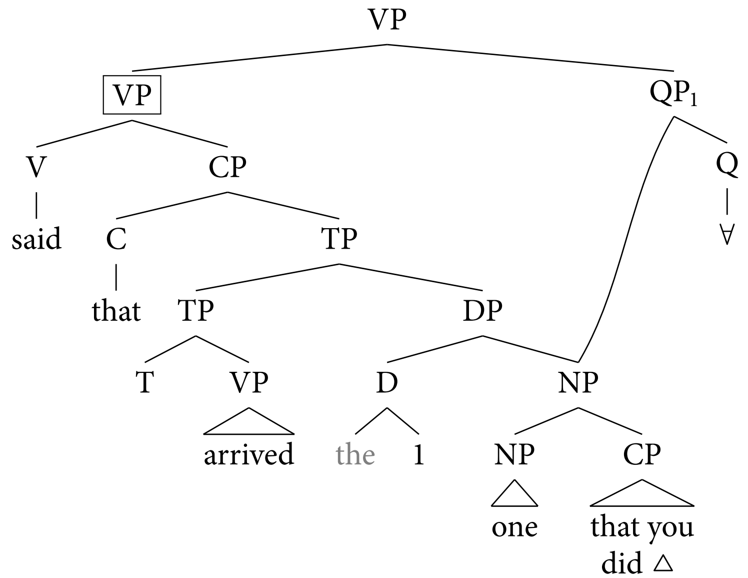
Let's now consider why something parallel is not possible for QR. The case we looked at which demonstrates this is (13), repeated below.

(13) * I said that everyone you did △ arrived.



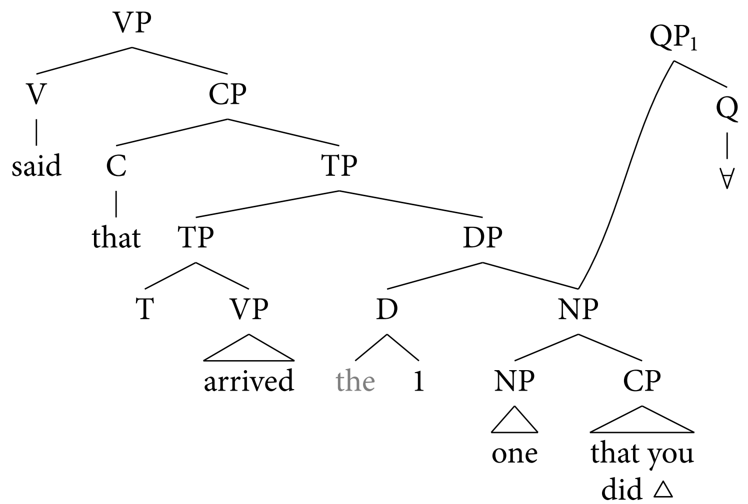
On the proposal here, the string in (13) can only get the representation in (40), which does not satisfy the structural requirements for resolving the ellipsis (since △ is inside its antecedent: $\boxed{\text{VP}}$).

(40)



As we've seen, *lin* must run before the QP is merged to $\boxed{\text{VP}}$ in order to fuse D and Q into *every*. So, the string associated with (40) will be built upon the output *lin* produces from applying to (41). That is (42).

(41)



(42) = said that everyone that you did arrived.

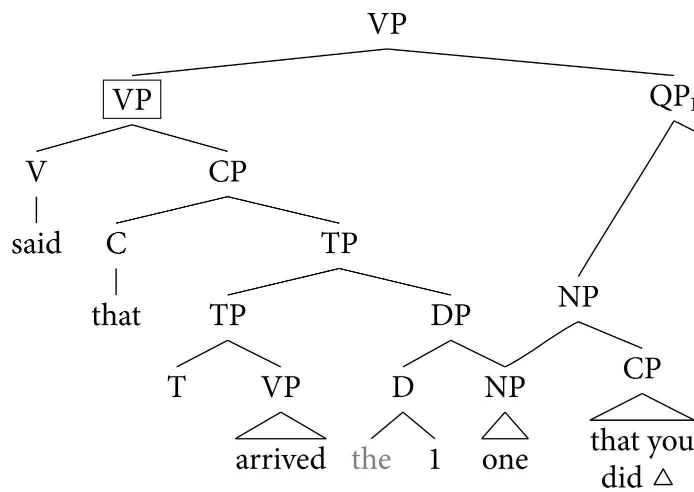
$$lin(\text{VP}) = \left\{ \begin{array}{llllllll} \text{said} < \text{that} & \text{that} < \text{T} & \text{T} < \text{arrived} & \text{arrived} < \text{D} & \text{D} < \text{one} & \text{one} < \text{that} & \text{that} < \text{you} & \text{you} < \text{did} \\ \text{said} < \text{T} & \text{that} < \text{arrived} & \text{T} < \text{D} & \text{arrived} < \text{one} & \text{D} < \text{that} & \text{one} < \text{you} & \text{that} < \text{did} & \\ \text{said} < \text{arrived} & \text{that} < \text{D} & \text{T} < \text{one} & \text{arrived} < \text{that} & \text{D} < \text{you} & \text{one} < \text{did} & & \\ \text{said} < \text{D} & \text{that} < \text{one} & \text{T} < \text{that} & \text{arrived} < \text{you} & \text{D} < \text{did} & & & \\ \text{said} < \text{one} & \text{that} < \text{that} & \text{T} < \text{you} & \text{arrived} < \text{did} & & & & \\ \text{said} < \text{that} & \text{that} < \text{you} & \text{T} < \text{did} & & & & & \\ \text{said} < \text{you} & \text{that} < \text{did} & & & & & & \\ \text{said} < \text{did} & & & & & & & \end{array} \right\}$$

$$lin(\text{QP}) = \left\{ \begin{array}{llll} \forall < \text{one} & \text{one} < \text{that} & \text{that} < \text{you} & \text{you} < \text{did} \\ \forall < \text{that} & \text{one} < \text{you} & \text{that} < \text{did} & \\ \forall < \text{you} & \text{one} < \text{did} & & \\ \forall < \text{did} & & & \end{array} \right\}$$

When *lin* runs again, subsequent to merging QP and VP, no new ordered pairs will be introduced, and so we will get the simple union of *lin*(VP) and *lin*(QP). That corresponds to the string indicated.

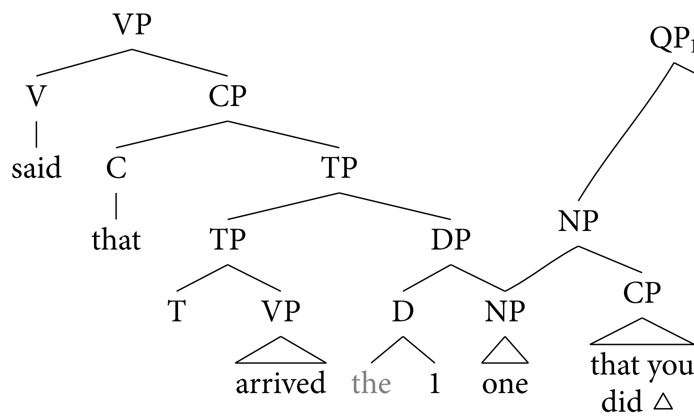
To resolve the ACD, we must have a representation that involves "late merge" of the relative clause. Under the present proposal, this will look like (43).

(43)



When *lin* runs on the representation that arises before QP merges with $\boxed{\text{VP}}$ (=(44)), it'll produce (45).

(44)



(45) = said that everyone arrived & everyone that you did

$$lin(\text{VP}) = \left\{ \begin{array}{lllll} \text{said} < \text{that} & \text{that} < \text{T} & \text{T} < \text{arrived} & \text{arrived} < \text{D} & \text{D} < \text{one} \\ \text{said} < \text{T} & \text{that} < \text{arrived} & \text{T} < \text{D} & \text{arrived} < \text{one} & \\ \text{said} < \text{arrived} & \text{that} < \text{D} & \text{T} < \text{one} & & \\ \text{said} < \text{D} & \text{that} < \text{one} & & & \\ \text{said} < \text{one} & & & & \end{array} \right\}$$

$$lin(\text{QP}) = \left\{ \begin{array}{llll} \forall < \text{one} & \text{one} < \text{that} & \text{that} < \text{you} & \text{you} < \text{did} \\ \forall < \text{that} & \text{one} < \text{you} & \text{that} < \text{did} & \\ \forall < \text{you} & \text{one} < \text{did} & & \\ \forall < \text{did} & & & \end{array} \right\}$$

Notice that because the relative clause is not yet inside VP, it is not included in the string associated with VP. As a consequence, only *lin*(QP) has information about where the relative clause will be positioned: it will follow everything else in QP. It is only after QP has merged with VP — to form (43) — that *lin* can order the material in the relative clause with the material in the VP. If *lin* applies to the VP formed in (43), it will add the ordered pairs in (46) to those collected in (45).

(46)

$$\text{The new outputs from } \textit{lin}(\ (43)\ ) = \left\{ \begin{array}{llll} \text{said} < \text{that} & \text{that} < \text{that} & \text{T} < \text{that} & \text{arrived} < \text{that} \\ \text{said} < \text{you} & \text{that} < \text{you} & \text{T} < \text{you} & \text{arrived} < \text{you} \\ \text{said} < \text{did} & \text{that} < \text{did} & \text{T} < \text{did} & \text{arrived} < \text{did} \\ \text{said} < \forall & \text{that} < \forall & \text{T} < \forall & \text{arrived} < \forall \end{array} \right\}$$

When this is combined with (45), we will fit the string in (47).

(47) …said that everyone arrived that you did △

For the relative clause to be positioned outside the VP that is serving as the antecedent for the ellipsis it contains, it will necessarily be positioned linearly in the "extraposed" position. This result is perfectly general. We derive that QR cannot put spoken material in a position where it is not semantically interpreted.

In fact, we derive something else:

(48) In English, if material is spoken in the higher position of a QR'd DP, it will show up to the right of the material in the lower position.

References

Adger, David, and Gillian Ramchand. 2005. Merge and move: *Wh*-dependencies revisited. *Linguistic Inquiry* 36:161–193.

Cable, Seth. 2007. The grammar of Q: Q-particles and the nature of wh-fronting, as revealed by the wh-questions of Tlingit. Doctoral Dissertation, Massachusetts Institute of Technology, Cambridge, Massachusetts.

Chierchia, Gennaro. 1992. Questions with quantifiers. *Natural Language Semantics* 1:181–234.

Chomsky, Noam. 1995. *The minimalist program*. Cambridge, Massachusetts: MIT Press.

Elbourne, Paul D. 2005. *Situations and individuals*. MIT Press.

Engdahl, Elisabet. 1980. The syntax and semantics of questions in Swedish. Doctoral Dissertation, University of Massachusetts, Amherst, Massachusetts.

Engdahl, Elisabet. 1986. *Constituent questions*. Dordrecht, The Netherlands: D. Reidel Publishing Company.

Fox, Danny. 2002. Antecedent-contained deletion and the copy theory of movement. *Linguistic Inquiry* 33:63–96.

Fox, Danny. 2003. On logical form. In *Minimalist syntax*, ed. Randall Hendrick, 82–123. Oxford: Blackwell Publishers.

Fox, Danny, and Jon Nissenbaum. 1999. Extraposition and scope: a case for overt QR. In *Proceedings of the West Coast Conference on Formal Linguistics 18*, ed. Sonya Bird, Andrew Carnie, Jason D. Haugen, and Peter Norquest, 132–144. Somerville, Massachusetts: Cascadilla Press.

Gärtner, Hans-Martin. 1997. Generalized transformations and beyond. Doctoral Dissertation, University of Franfurt/Main.

Gärtner, Hans-Martin. 2001. Phase-linking meets Minimalism. In *Proceedings of WCCFL 18*.

Hagstrom, Paul. 1998. Decomposing questions. Doctoral Dissertation, Massachusetts Institute of Technology, Cambridge, Massachusetts.

Hagstrom, Paul. 2000. The movement of question particles. In *Proceedings of the North East Linguistic Society*, ed. Masako Hirotani, Andries Coetzee, Nancy Hall, and Ji-yung Kim, 275–286. Rutgers University: Graduate Linguistic Student Association.

Kishimoto, Hideki. 2005. Wh-in-situ and movement in Sinhala questions. *Natural Language and Linguistic Theory* 23:1–51.

Nunes, Jairo. 1999. Linearization of chains and phonetic realization of chain links. In *Working minimalism*, ed. Samuel Epstein and Norbert Hornstein, 217–249. Cambridge, Massachusetts: MIT Press.

Nunes, Jairo. 2004. *Linearization of chains and sideward movement*. Linguistic Inquiry Monographs. Cambridge, Massachusetts: MIT Press.

Starke, Michal. 2001. Move dissolves into merge: A theory of locality. Doctoral Dissertation, University of Geneva.

Tiedeman, Robyne. 1995. Some remarks on antecedent contained deletion. In *Minimalism and Linguistic Theory*, ed. Shosuke Haraguchi and Michio Funaki, 67–103. Tokyo: Hituzi Syobo.