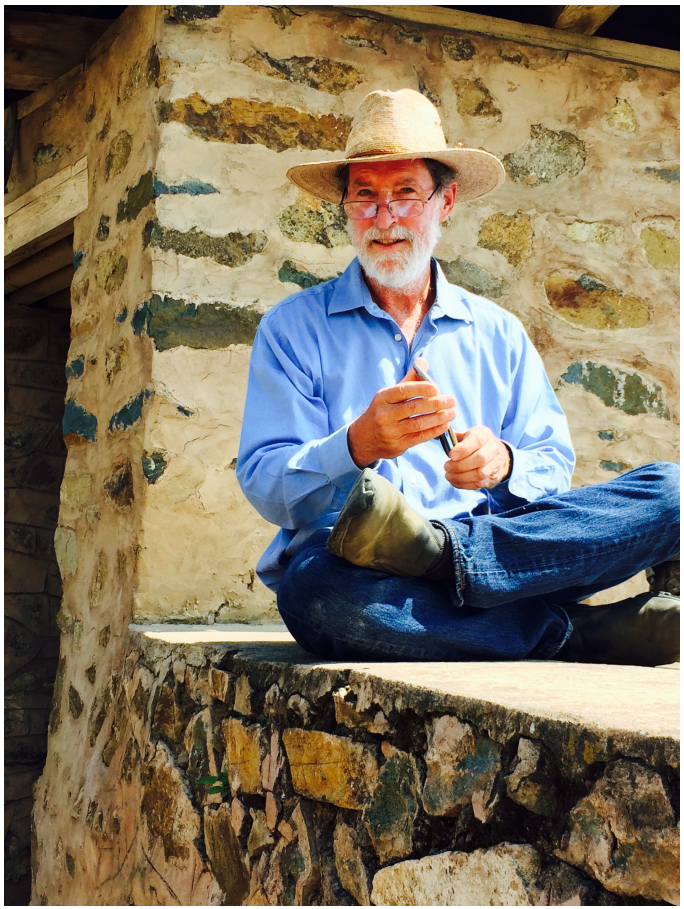
Mereological Syntax and Grammatical Locality

David Adger

Department of Linguistics
Queen Mary University of London

Dick Oehrle Memorial Lecture
ESSLI 2024, Leuven

What is Syntax?

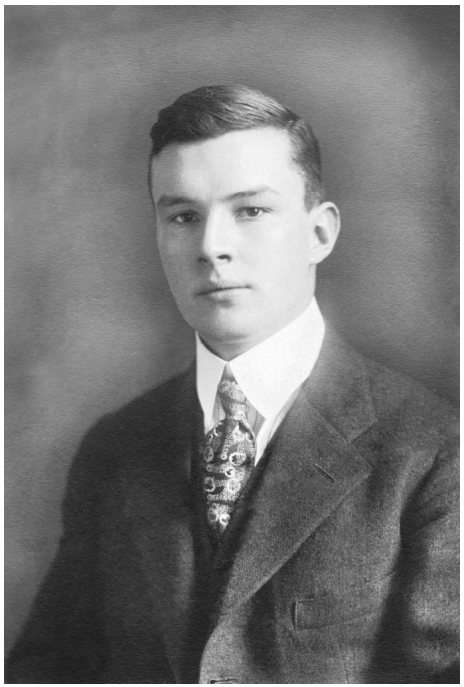


"The critical property for a complex expression is that it be partitionable into a structured set of atomic component sub-expressions whose properties we may assume are available directly. In particular, if E is a complex expression with atomic components $w_1 \ldots w_k$, we may represent the structure of these components in E by writing $C(w_1 \ldots w_k)$, where the operator C represents how the atomic components of E are put together. From this point of view, the problem of composition is to provide a general characterization of the relation between the properties of the atomic components of E and the properties of E itself." *Oehrle (2011)*

What is Syntax?

- Oehrle's perspective was to take expressions as the object that was to be analysed, where an expression is "an abstract analysis of the (subjective representation of) the physical entity in question" (Oehrle 1988).
- Expressions, in this sense, are given a compositional analysis, and they have, in Oehrle's work, the character of a (logical) language: strings of concatenated symbols.
- There is a more general alternative, noted by Curry in the late 50s.

What is Syntax?



"In combinatory logic these formal objects, called obs, were wholly unspecified; it was merely postulated that there was a binary operation of application among them, that the obs be constructed from the primitive objects, called atoms, by these operations, and that the construction of an ob be unique. This means that the obs were thought of, not as strings of atoms, but as structures like a genealogical tree. Now of course there are various ways in which such a tree can be associated with a string. "
Curry (1961)

What is syntax?

- Chomsky's early work (e.g. in The Logical Structure of Linguistic Theory, 1955) took expressions to be, much as Oehrle suggested, characterizable via concatenation algebras. Syntactic structures were sets of strings.
- More recent work (Bare Phrase Structure, Chomsky 1994) has theorized syntax not via expressions but via Syntactic Objects which are not concatenative, but are hierarchical in nature, requiring a kind of applicative system to generate them, much as Curry suggested.
- Bare Phrase Structure style Syntactic Objects are set theoretic:

  (1) $\{\text{the } \{\text{happy, cat}\}\}$

- I'm going to argue for an alternative: they are part-theoretic.

Ingredients

- Intuitively, the most obvious atomic components of syntax are perhaps words

Ingredients

- Intuitively, the most obvious atomic components of syntax are perhaps words
- which appear to be put together locally (contiguously) to make sentences

(2) I like Mo

Ingredients

- Intuitively, the most obvious atomic components of syntax are perhaps words
- which appear to be put together locally (contiguously) to make sentences

  (2) I like Mo

- but most theories annotate words (or perhaps parts of words) with special information (categories) which is what the operation(s) see:

Ingredients

- Intuitively, the most obvious atomic components of syntax are perhaps words
- which appear to be put together locally (contiguously) to make sentences

  (2) I like Mo

- but most theories annotate words (or perhaps parts of words) with special information (categories) which is what the operation(s) see:

  (3)
NP	V	NP
I	like	Mo

Ingredients

- Intuitively, the most obvious atomic components of syntax are perhaps words
- which appear to be put together locally (contiguously) to make sentences

  (2) I like Mo

- but most theories annotate words (or perhaps parts of words) with special information (categories) which is what the operation(s) see:

  (3)
NP	V	NP
I	like	Mo

- generally we don't want our rules to refer to particular words, or else we miss generalizations about classes of words.

Ingredients

- Intuitively, the most obvious atomic components of syntax are perhaps words
- which appear to be put together locally (contiguously) to make sentences

  (2) I like Mo

- but most theories annotate words (or perhaps parts of words) with special information (categories) which is what the operation(s) see:

  (3)
NP	V	NP
I	like	Mo

- generally we don't want our rules to refer to particular words, or else we miss generalizations about classes of words.
- So the ingredients of syntax are really the categories, which are combined by operations ...

Ingredients

- Intuitively, the most obvious atomic components of syntax are perhaps words
- which appear to be put together locally (contiguously) to make sentences

  (2) I like Mo

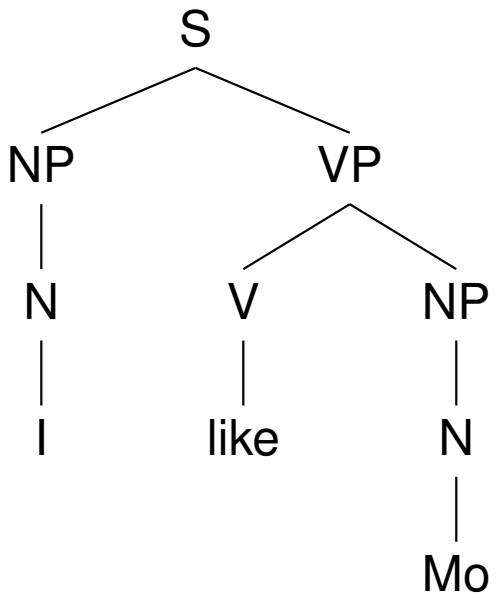
- but most theories annotate words (or perhaps parts of words) with special information (categories) which is what the operation(s) see:

  (3)
NP	V	NP
I	like	Mo

- generally we don't want our rules to refer to particular words, or else we miss generalizations about classes of words.
- So the ingredients of syntax are really the categories, which are combined by operations ... words are secondary

Three Theories

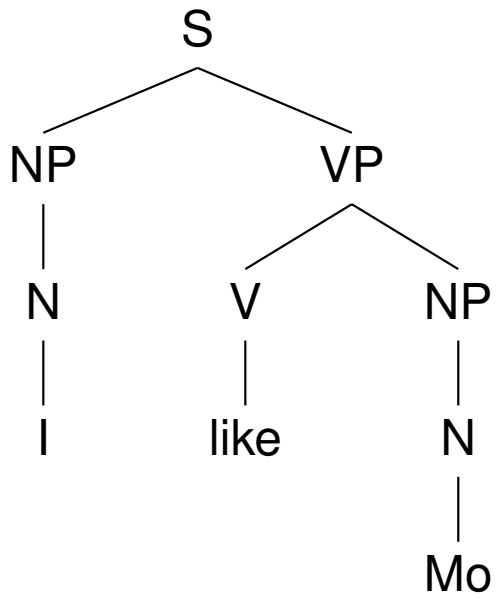
(4)



Phrase Structure Rules

Three Theories

(4)



Phrase Structure Rules

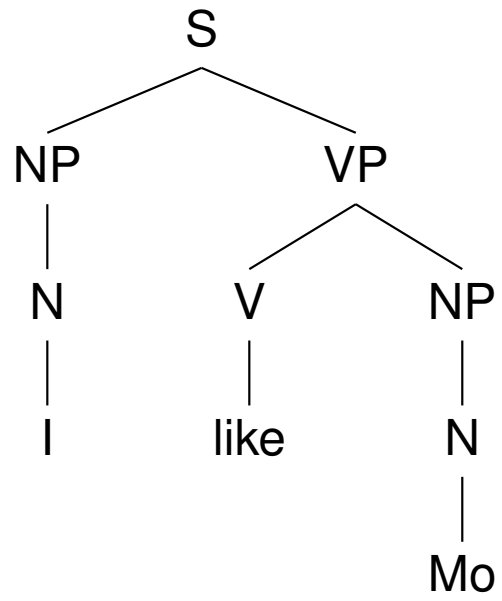
(5)

$$\dfrac{\dfrac{\text{I}}{\text{NP}} \quad \dfrac{\dfrac{\text{like}}{(\text{S}\backslash\text{NP})/\text{NP}} \quad \dfrac{\text{Mo}}{\text{NP}}}{\text{S}\backslash\text{NP}}>}{\text{S}}<$$

Application of a general operation to categories

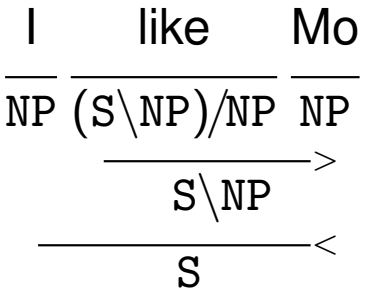
Three Theories

(4)



Phrase Structure Rules

(5)

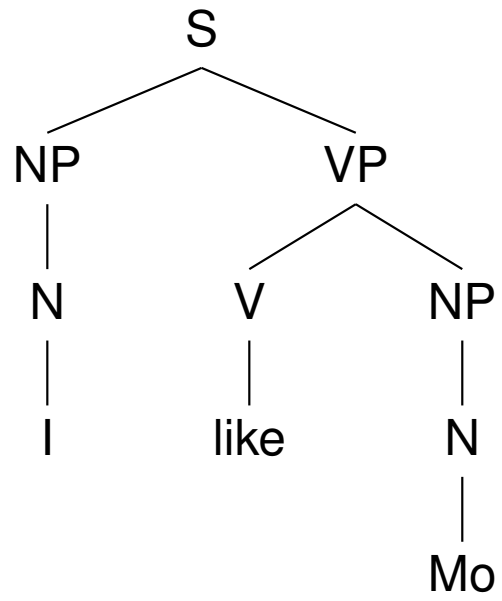


Application of a general operation to categories

Bare Phrase Structure has a single set-constructing operator

Three Theories

(4)



Phrase Structure Rules

(5)

$$\frac{\mathrm{I}}{\mathrm{NP}} \quad \frac{\mathrm{like}}{(\mathrm{S}\backslash\mathrm{NP})/\mathrm{NP}} \quad \frac{\mathrm{Mo}}{\mathrm{NP}}$$

$$\frac{\dfrac{\mathrm{I}}{\mathrm{NP}} \quad \dfrac{\mathrm{like}}{(\mathrm{S}\backslash\mathrm{NP})/\mathrm{NP}} \quad \dfrac{\mathrm{Mo}}{\mathrm{NP}}}{\dfrac{\mathrm{like}\ \mathrm{Mo}}{\mathrm{S}\backslash\mathrm{NP}}>}$$

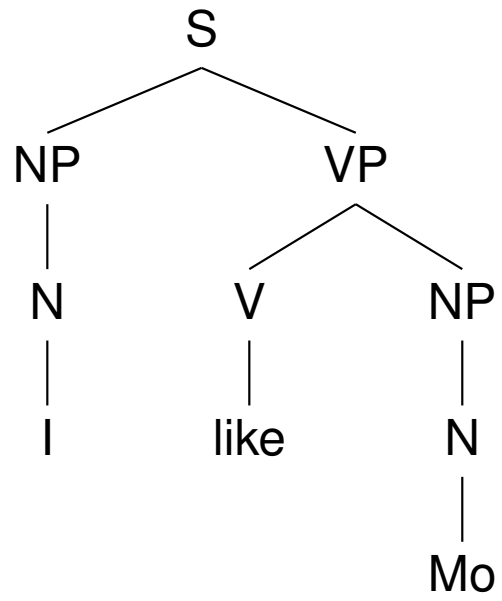
$$\frac{\mathrm{I}\ (\mathrm{like}\ \mathrm{Mo})}{\mathrm{S}}<$$

Application of a general operation to categories

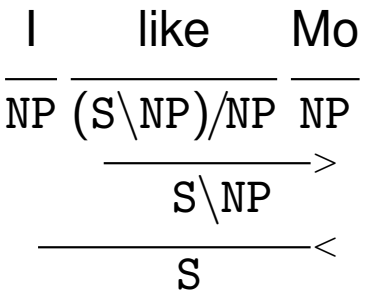
Bare Phrase Structure has a single set-constructing operator Merge

Three Theories

(4)



(5)



Application of a general operation to categories

Phrase Structure Rules

Bare Phrase Structure has a single set-constructing operator Merge

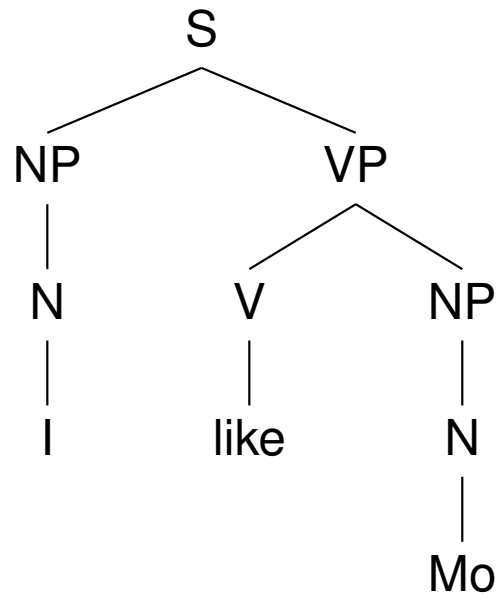
(6) a. Merge(like, Mo) = {like, Mo}

b. Merge(I, {like, Mo}) = {I, {like, Mo}}

But

Three Theories

(4)



Phrase Structure Rules

(5)

$$\dfrac{\dfrac{\text{I}}{\text{NP}} \quad \dfrac{\dfrac{\text{like}}{(\text{S}\backslash\text{NP})/\text{NP}} \quad \dfrac{\text{Mo}}{\text{NP}}}{\text{S}\backslash\text{NP}}>}{\text{S}}<$$

Application of a general operation to categories

Bare Phrase Structure has a single set-constructing operator Merge

(6) a. Merge(like, Mo) = {like, Mo}

b. Merge(I, {like, Mo}) = {I, {like, Mo}}

But needs a separate theory to integrate categories with the structures (Labelling Theory)

Non-local dependencies

(7) a. The cat that I like ⟨the cat⟩
b. (guess) which cat I like ⟨which cat⟩
c. Mo, I like ⟨Mo⟩

Non-local dependencies

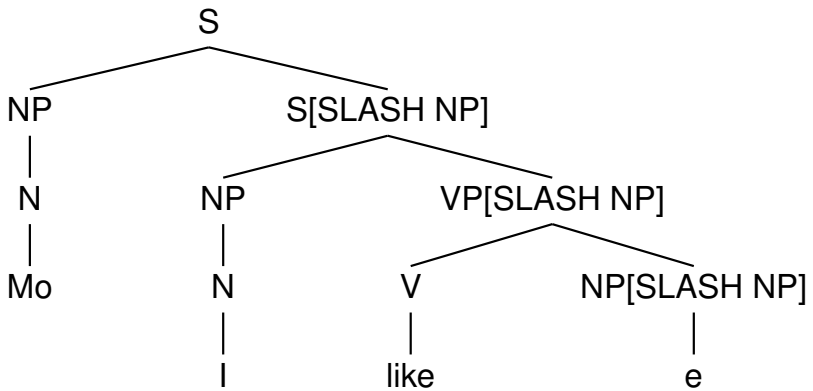
(7) a. The cat that I like ⟨the cat⟩
b. (guess) which cat I like ⟨which cat⟩
c. Mo, I like ⟨Mo⟩

And these can get quite non local:

(8) a. (I saw) the cat that everyone knows that the neighbours refused to look after ⟨the cat⟩
b. (guess) which cat Anson told Karen that he thought it would be difficult for Hedvig to like ⟨which cat⟩
c. Mo, I definitely know that people have always given food to ⟨Mo⟩ on our street.

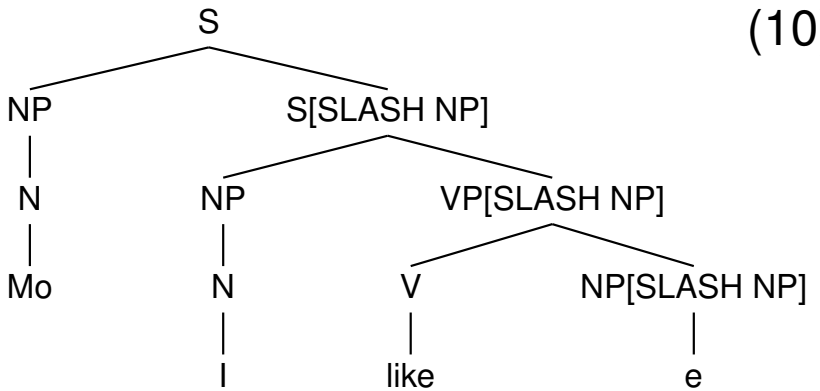
Extra Mechanisms

(9)

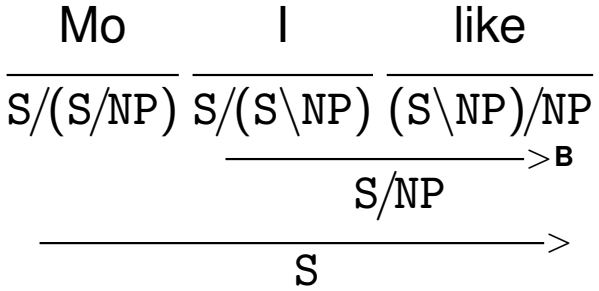


Extra Mechanisms

(9)

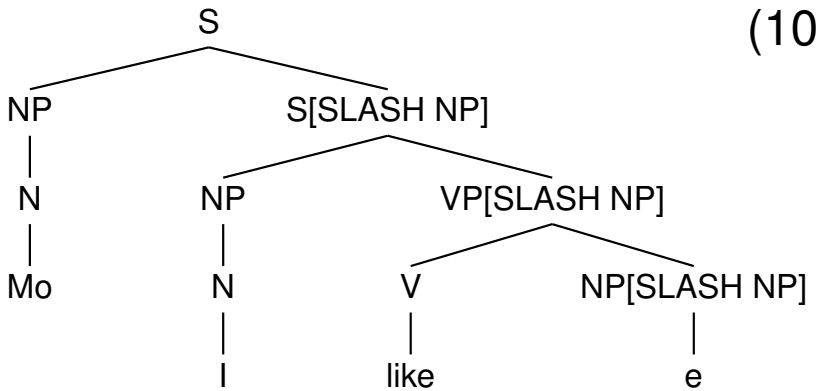


(10)

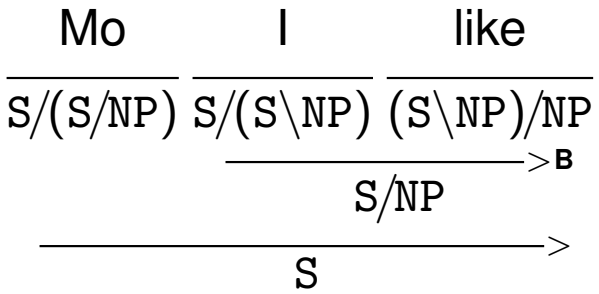


Extra Mechanisms

(9)



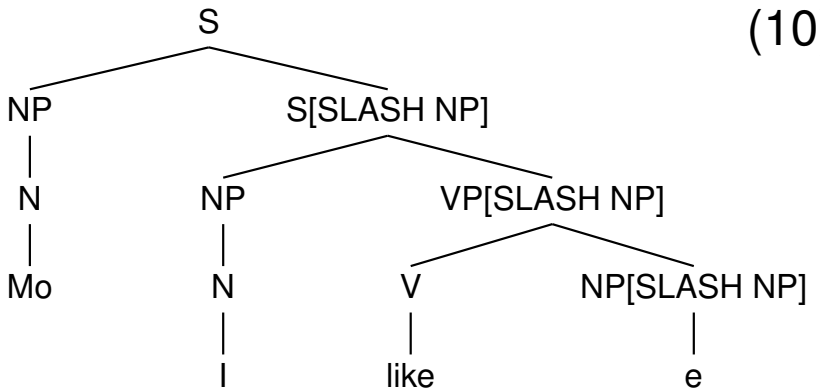
(10)



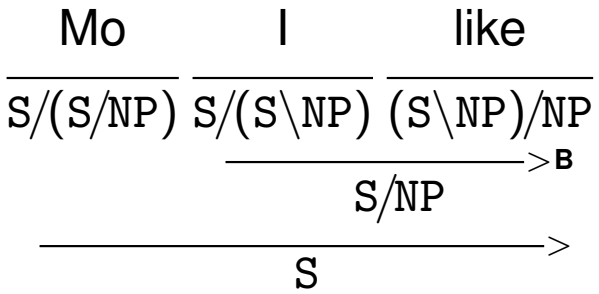
New mechanisms required to extend dependency beyond contiguity:

Extra Mechanisms

(9)



(10)

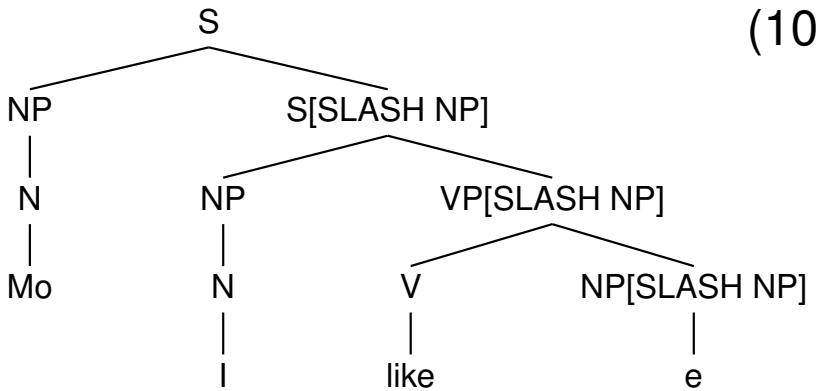


New mechanisms required to extend dependency beyond contiguity:

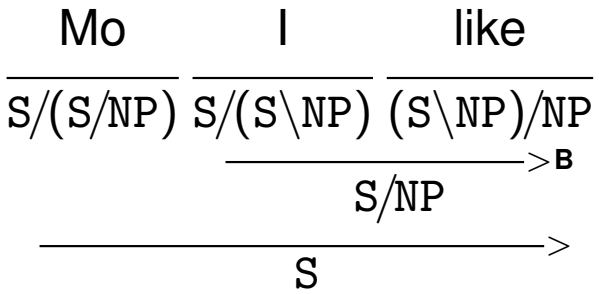
- slash passing: category valued feature plus special inheritance mechanism

Extra Mechanisms

(9)



(10)



New mechanisms required to extend dependency beyond contiguity:

- slash passing: category valued feature plus special inheritance mechanism
- composition rules: special type raising of the topic and Functional Composition

Internal vs External Merge

Merge part of a complex structure with that complex structure:

Internal vs External Merge

Merge part of a complex structure with that complex structure:

(11) a. Merge(we, {like, Mo}) = {we, {like, Mo}}

Internal vs External Merge

Merge part of a complex structure with that complex structure:

(11) a. Merge(we, {like, Mo}) = {we, {like, Mo}}

b. Merge(Mo, {we, {like, Mo}}) = {Mo, {we, {like, Mo}}}

Internal vs External Merge

Merge part of a complex structure with that complex structure:

(11) a. Merge(we, {like, Mo}) = {we, {like, Mo}}
b. Merge(Mo, {we, {like, Mo}}) = {Mo, {we, {like, Mo}}}

The same mechanism just with different inputs. Nothing extra needed which looks like a win:

Internal vs External Merge

Merge part of a complex structure with that complex structure:

(11) a. Merge(we, {like, Mo}) = {we, {like, Mo}}
b. Merge(Mo, {we, {like, Mo}}) = {Mo, {we, {like, Mo}}}

The same mechanism just with different inputs. Nothing extra needed which looks like a win:

One mechanism with two modes of application: External (Merging two separate things) and Internal (Merging part of something with that thing)

Internal vs External Merge

Merge part of a complex structure with that complex structure:

(11) a. Merge(we, {like, Mo}) = {we, {like, Mo}}
b. Merge(Mo, {we, {like, Mo}}) = {Mo, {we, {like, Mo}}}

The same mechanism just with different inputs. Nothing extra needed which looks like a win:

One mechanism with two modes of application: External (Merging two separate things) and Internal (Merging part of something with that thing) but there is a well known problem ...

Copies vs Repetitions

Distinguishing copies from repetitions

(12) a. Mo grew
b. Mo grew Mo

(13) a. Merge(Mo, grew) = {grew, Mo}
b. Merge(Mo, {grew, Mo}) = {Mo, {grew, Mo}}

So BPS needs an extra mechanism too, to distinguish (12a) and (12b). Lots of ways of doing this (indices, appealing to theta-theory, etc.) but it would be good to not have this problem.

Still on Merge

Internal Merge is sometimes blocked:

Still on Merge

Internal Merge is sometimes blocked:

(14) a. Thuirt Daibhidh [$_{CP}$ gum buail Calum an cat]
say.PAST David THAT hit.FUT Calum the cat
'David said Calum will hit the cat.'

b. An cat [$_{CP}$ a thuirt Daibhidh [$_{CP}$ a bhuaileas Calum
the cat REL say.PAST David REL hit.FUT.REL Calum]
'The cat that David said Calum will hit.'

c. *An cat [$_{CP}$ a thuirt Daibhidh [$_{CP}$ gum buail Calum
the cat REL say.PAST David THAT hit.FUT Calum]
'The cat that David said Calum will hit.'

Still on Merge

Internal Merge is sometimes blocked:

(14) a. Thuirt Daibhidh [$_{CP}$ gum buail Calum an cat]
say.PAST David THAT hit.FUT Calum the cat
'David said Calum will hit the cat.'

b. An cat [$_{CP}$ a thuirt Daibhidh [$_{CP}$ a bhuaileas Calum
the cat REL say.PAST David REL hit.FUT.REL Calum]
'The cat that David said Calum will hit.'

c. *An cat [$_{CP}$ a thuirt Daibhidh [$_{CP}$ gum buail Calum
the cat REL say.PAST David THAT hit.FUT Calum]
'The cat that David said Calum will hit.'

- Lots of similar cross-linguistic evidence that finite sentences block Merge applying internally across them, but that there are ways of circumventing this

Still on Merge

Internal Merge is sometimes blocked:

(14) a. Thuirt Daibhidh [$_{CP}$ gum buail Calum an cat]
say.PAST David THAT hit.FUT Calum the cat
'David said Calum will hit the cat.'

b. An cat [$_{CP}$ a thuirt Daibhidh [$_{CP}$ a bhuaileas Calum
the cat REL say.PAST David REL hit.FUT.REL Calum]
'The cat that David said Calum will hit.'

c. *An cat [$_{CP}$ a thuirt Daibhidh [$_{CP}$ gum buail Calum
the cat REL say.PAST David THAT hit.FUT Calum]
'The cat that David said Calum will hit.'

- Lots of similar cross-linguistic evidence that finite sentences block Merge applying internally across them, but that there are ways of circumventing this
- Nothing in Merge itself will block the ungrammatical cases.

Still on Merge

Internal Merge is sometimes blocked:

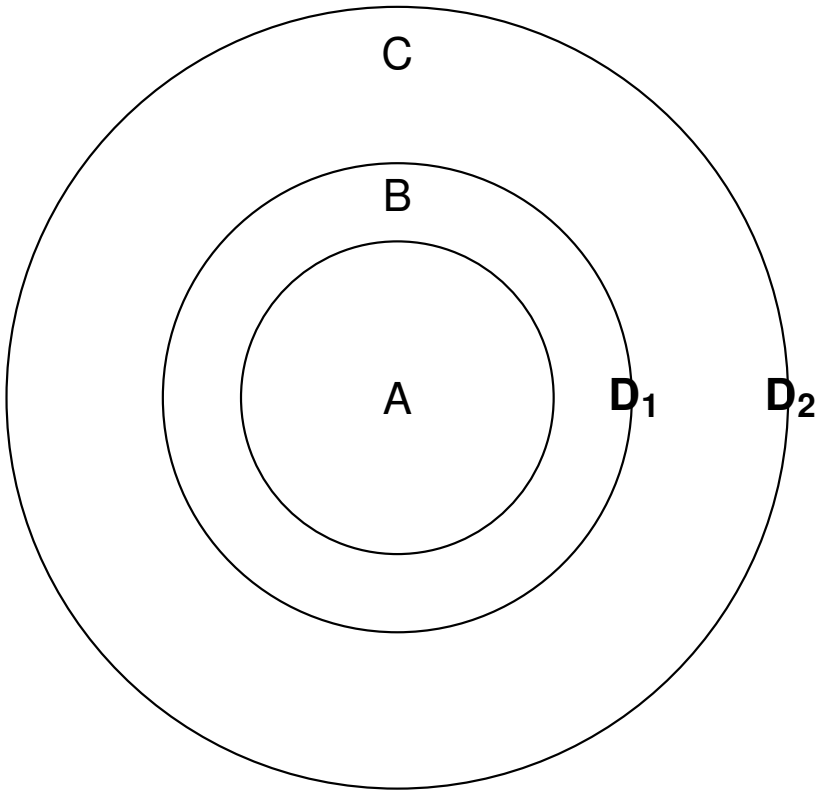
(14) a. Thuirt Daibhidh [$_{CP}$ gum buail Calum an cat]
say.PAST David THAT hit.FUT Calum the cat
'David said Calum will hit the cat.'

b. An cat [$_{CP}$ a thuirt Daibhidh [$_{CP}$ a bhuaileas Calum
the cat REL say.PAST David REL hit.FUT.REL Calum]
'The cat that David said Calum will hit.'

c. *An cat [$_{CP}$ a thuirt Daibhidh [$_{CP}$ gum buail Calum
the cat REL say.PAST David THAT hit.FUT Calum]
'The cat that David said Calum will hit.'

- Lots of similar cross-linguistic evidence that finite sentences block Merge applying internally across them, but that there are ways of circumventing this
- Nothing in Merge itself will block the ungrammatical cases.
- Merge is a theory of Phrase Structure, not Locality

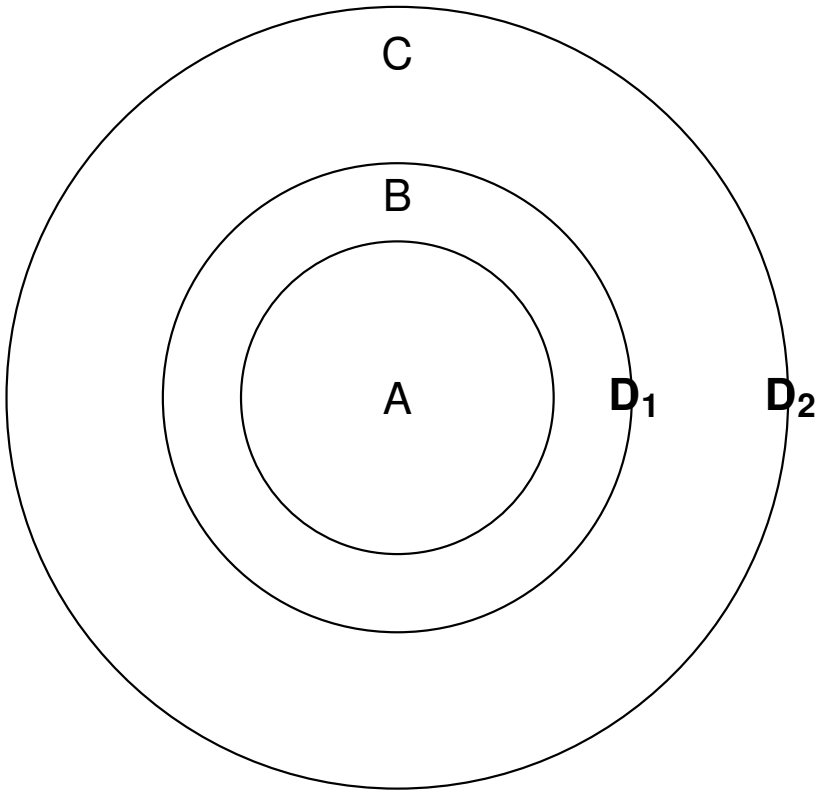
The Theory of Locality



The Phase Impenetrability Condition (PIC):

- Syntax can relate A and B inside D1

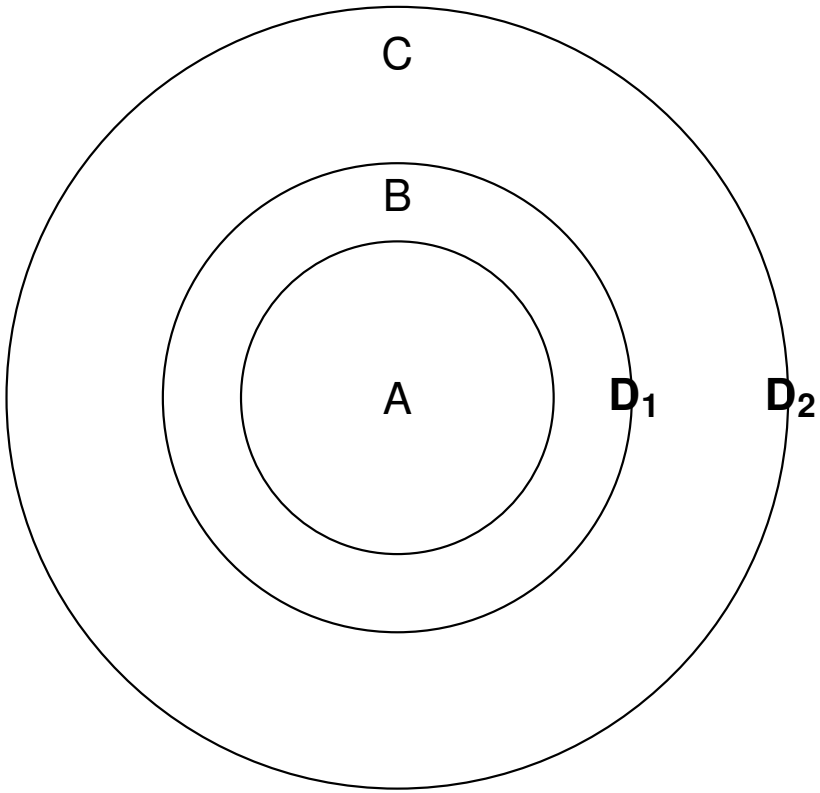
The Theory of Locality



The Phase Impenetrability Condition (PIC):

- Syntax can relate A and B inside D1
- Syntax cannot relate A and C across D1

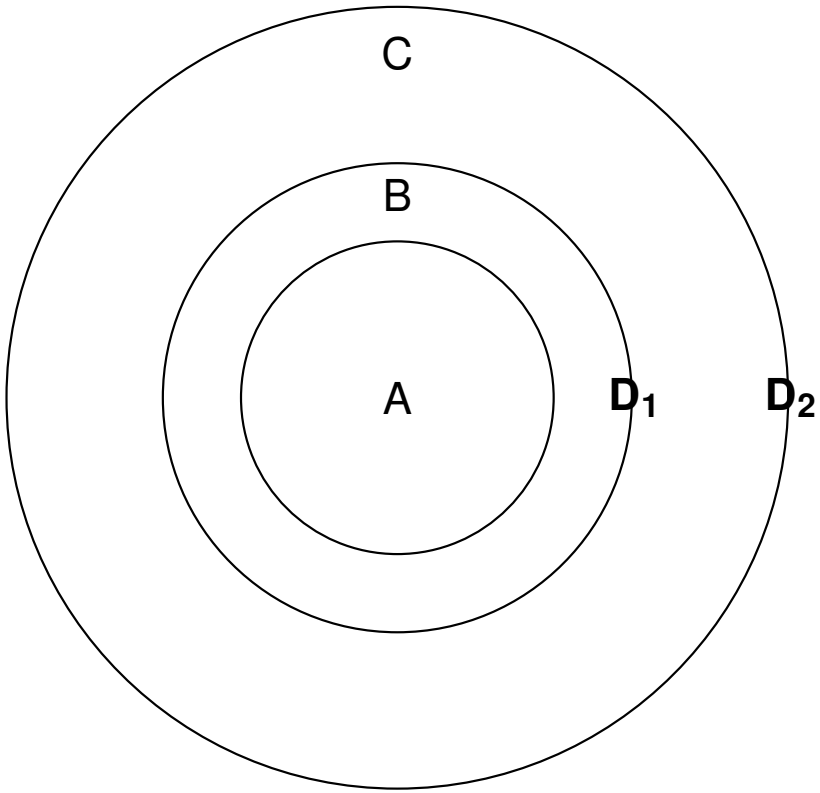
The Theory of Locality



The Phase Impenetrability Condition (PIC):

- Syntax can relate A and B inside D1
- Syntax cannot relate A and C across D1
- Syntax can relate B and C across D1

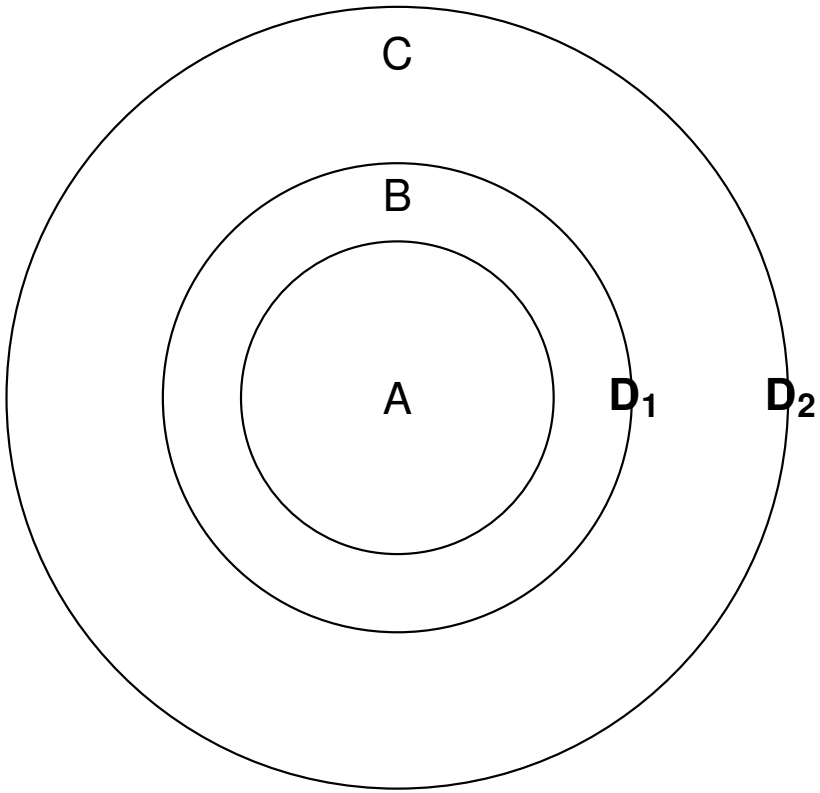
The Theory of Locality



The Phase Impenetrability Condition (PIC):

- Syntax can relate A and B inside D1
- Syntax cannot relate A and C across D1
- Syntax can relate B and C across D1
- So apparent AC relations are really AB+BC relations

The Theory of Locality



The Phase Impenetrability Condition (PIC):

- Syntax can relate A and B inside D1
- Syntax cannot relate A and C across D1
- Syntax can relate B and C across D1
- So apparent AC relations are really AB+BC relations

In Gaelic an AB relation is morphologically signalled by *a*, and the absence of one by *gu(n)*, which then precludes an AC relationship

Phases

Domains like D1 are called cyclic domains (now rechristened phases).
Derivations that cross phases via their edge are called successive cyclic.

(15) a. the cat ....[$_{CP}$ ⟨the cat⟩ [... ⟨the cat⟩

b. *the cat ....[$_{CP}$ [... ⟨the cat⟩

Phases

Domains like D1 are called cyclic domains (now rechristened phases).
Derivations that cross phases via their edge are called successive cyclic.

(15) a. the cat ....[$_{CP}$ ⟨the cat⟩ [... ⟨the cat⟩

b. *the cat ....[$_{CP}$ [... ⟨the cat⟩

- what are the phases?

Phases

Domains like D1 are called cyclic domains (now rechristened phases).
Derivations that cross phases via their edge are called successive cyclic.

(15) a. the cat ....[$_{CP}$ ⟨the cat⟩ [... ⟨the cat⟩
b. *the cat ....[$_{CP}$ [... ⟨the cat⟩

- what are the phases?
- what are the subparts of phases that are exceptions to locality (classically known as escape hatches or edges)?

Phases

Domains like D1 are called cyclic domains (now rechristened phases).
Derivations that cross phases via their edge are called successive cyclic.

(15) a. the cat ....[$_{CP}$ ⟨the cat⟩ [... ⟨the cat⟩
b. *the cat ....[$_{CP}$ [... ⟨the cat⟩

- what are the phases?
- what are the subparts of phases that are exceptions to locality (classically known as escape hatches or edges)?

Classical answers:

Phases

Domains like D1 are called cyclic domains (now rechristened phases).
Derivations that cross phases via their edge are called successive cyclic.

(15) a. the cat ....[$_{CP}$ ⟨the cat⟩ [... ⟨the cat⟩
b. *the cat ....[$_{CP}$ [... ⟨the cat⟩

- what are the phases?
- what are the subparts of phases that are exceptions to locality (classically known as escape hatches or edges)?

Classical answers:

(16) a. CP is a phase, so is vP (but to this have been added DP, and many other contenders (PP, NP, ...), maybe all phrases?)

Phases

Domains like D1 are called cyclic domains (now rechristened phases).
Derivations that cross phases via their edge are called successive cyclic.

(15) a. the cat ....[$_{CP}$ ⟨the cat⟩ [... ⟨the cat⟩
b. *the cat ....[$_{CP}$ [... ⟨the cat⟩

- what are the phases?
- what are the subparts of phases that are exceptions to locality (classically known as escape hatches or edges)?

Classical answers:

(16) a. CP is a phase, so is vP (but to this have been added DP, and many other contenders (PP, NP, ...), maybe all phrases?) why?
b. The head plus anything Merged to (the projection of) the head constitute the edge

Phases

Domains like D1 are called cyclic domains (now rechristened phases).
Derivations that cross phases via their edge are called successive cyclic.

(15) a. the cat ....[$_{CP}$ ⟨the cat⟩ [... ⟨the cat⟩
b. *the cat ....[$_{CP}$ [... ⟨the cat⟩

- what are the phases?
- what are the subparts of phases that are exceptions to locality (classically known as escape hatches or edges)?

Classical answers:

(16) a. CP is a phase, so is vP (but to this have been added DP, and many other contenders (PP, NP, ...), maybe all phrases?) why?
b. The head plus anything Merged to (the projection of) the head constitute the edge why?

A bigger question

- Why does human language have this cyclic property at all (assuming the, by now, extensive evidence for it is correct)?

A bigger question

- Why does human language have this cyclic property at all (assuming the, by now, extensive evidence for it is correct)?
- Even if we accept that there are cyclic domains, why do they have *exceptions*. Why not just do long distance dependencies with semantic binding?

A bigger question

- Why does human language have this cyclic property at all (assuming the, by now, extensive evidence for it is correct)?
- Even if we accept that there are cyclic domains, why do they have *exceptions*. Why not just do long distance dependencies with semantic binding?

(17) The boat that Tenar learned that Ged sailed it is called Lookfar.

A bigger question

- Why does human language have this cyclic property at all (assuming the, by now, extensive evidence for it is correct)?
- Even if we accept that there are cyclic domains, why do they have *exceptions*. Why not just do long distance dependencies with semantic binding?

  (17) The boat that Tenar learned that Ged sailed it is called Lookfar.

- Maybe there are no syntactic locality domains (the Reductionist View, as in construction grammar), but if there are, we need answers to these *why*-questions.

A bigger question

- Why does human language have this cyclic property at all (assuming the, by now, extensive evidence for it is correct)?
- Even if we accept that there are cyclic domains, why do they have *exceptions*. Why not just do long distance dependencies with semantic binding?

  (17) The boat that Tenar learned that Ged sailed it is called Lookfar.

- Maybe there are no syntactic locality domains (the Reductionist View, as in construction grammar), but if there are, we need answers to these *why*-questions.
- Phase Theory essentially stipulates the answers (PIC and a list of categories)

Islands

Given the idea that certain categories are cyclic, we had, back the 1970s, a beginning of an explanation for certain Islands (this was the idea behind Subjacency):

Islands

Given the idea that certain categories are cyclic, we had, back the 1970s, a beginning of an explanation for certain Islands (this was the idea behind Subjacency):

(18) a. What did Tenar learn $[_{CP}$ ⟨what⟩ that $[_{TP}$ Ged sailed ⟨what⟩?
b. *What did Tenar wonder $[_{CP}$ where $[_{TP}$ Ged sailed ⟨what⟩?

Islands

Given the idea that certain categories are cyclic, we had, back the 1970s, a beginning of an explanation for certain Islands (this was the idea behind Subjacency):

(18) a. What did Tenar learn $[_{CP}$ ⟨what⟩ that $[_{TP}$ Ged sailed ⟨what⟩?
b. *What did Tenar wonder $[_{CP}$ where $[_{TP}$ Ged sailed ⟨what⟩?

But Merge allows multiple specifiers, so even with phases, we always have a way of circumventing locality domains. Phases are not a theory of Islands (unlike Subjacency).

Islands

Given the idea that certain categories are cyclic, we had, back the 1970s, a beginning of an explanation for certain Islands (this was the idea behind Subjacency):

(18) a. What did Tenar learn $[_{CP}$ ⟨what⟩ that $[_{TP}$ Ged sailed ⟨what⟩?
b. *What did Tenar wonder $[_{CP}$ where $[_{TP}$ Ged sailed ⟨what⟩?

But Merge allows multiple specifiers, so even with phases, we always have a way of circumventing locality domains. Phases are not a theory of Islands (unlike Subjacency).

(19) *What did Ged wonder $[_{CP}$ ⟨what⟩ $[_{CP}$ where $[_{TP}$ Ged sailed ⟨what⟩?

For the linguists in the audience!

Current Minimalist Theory doesn't really have a theory of Islands, partly because the empirical cross-linguistic domain has become more complex over the years

For the linguists in the audience!

Current Minimalist Theory doesn't really have a theory of Islands, partly because the empirical cross-linguistic domain has become more complex over the years

(20)
a. Attract Closest plus a bar on multiple specifiers for certain CPs Wh-Islands
b. Edge Condition (don't look inside phase edges) plus ...
c. Inactivity Condition (don't mess with phrases that have had all their features checked) Subject Islands
d. AntiLocality (movements can't be too short) plus Conditional Phasehood (NP a phase for elements it doesn't theta mark) Complex Noun Phrase Islands

The absence of a good theory here leaves open the space for a rich set of principles that are proposed, somewhat willy-nilly, to capture these various phenomena (I've just listed a few - there are more). I'm not sure other approaches are in a much better position

Three issues

Diagnosis: Merge is an interesting idea, unifying the theory of local and non-local dependency in a way that other theories do not, but ...

Three issues

Diagnosis: Merge is an interesting idea, unifying the theory of local and non-local dependency in a way that other theories do not, but ...

(21) a. We need an extra theory to label the output of Merge, and that has turned out to be less simple than we'd like

Three issues

Diagnosis: Merge is an interesting idea, unifying the theory of local and non-local dependency in a way that other theories do not, but ...

(21) a. We need an extra theory to label the output of Merge, and that has turned out to be less simple than we'd like

b. We need an extra theory to distinguish copies from repetitions (*Mo grew* vs *Mo grew Mo*), and that has turned out to be quite difficult

Three issues

Diagnosis: Merge is an interesting idea, unifying the theory of local and non-local dependency in a way that other theories do not, but ...

(21)
a. We need an extra theory to label the output of Merge, and that has turned out to be less simple than we'd like
b. We need an extra theory to distinguish copies from repetitions (*Mo grew* vs *Mo grew Mo*), and that has turned out to be quite difficult
c. We need an extra theory of Phasehood to capture successive cyclicity effects

Three issues

Diagnosis: Merge is an interesting idea, unifying the theory of local and non-local dependency in a way that other theories do not, but ...

(21)
a. We need an extra theory to label the output of Merge, and that has turned out to be less simple than we'd like
b. We need an extra theory to distinguish copies from repetitions (*Mo grew* vs *Mo grew Mo*), and that has turned out to be quite difficult
c. We need an extra theory of Phasehood to capture successive cyclicity effects
d. We need a bunch of extra principles to capture islands because Merge allows circumvention of phases

Three issues

Diagnosis: Merge is an interesting idea, unifying the theory of local and non-local dependency in a way that other theories do not, but ...

(21)
a. We need an extra theory to label the output of Merge, and that has turned out to be less simple than we'd like
b. We need an extra theory to distinguish copies from repetitions (*Mo grew* vs *Mo grew Mo*), and that has turned out to be quite difficult
c. We need an extra theory of Phasehood to capture successive cyclicity effects
d. We need a bunch of extra principles to capture islands because Merge allows circumvention of phases

Remedy: not more ingredients for our theory but start from scratch with the basics of how structure is built.

Replacing Merge with Subjoin

Thinking back to Curry's notion that the relevant formal objects have a hierarchical structure, I'll pursue the idea that that structure is organized via parthood not set membership, so syntactic objects are mereological, not set-theoretic. The operation is then one that creates part-relations.

(22) Subjoin(α,β) $\Rightarrow$ β modified so that $\alpha < \beta$

Replacing Merge with Subjoin

Thinking back to Curry's notion that the relevant formal objects have a hierarchical structure, I'll pursue the idea that that structure is organized via parthood not set membership, so syntactic objects are mereological, not set-theoretic. The operation is then one that creates part-relations.

(22) Subjoin(α,β) $\Rightarrow$ β modified so that $\alpha < \beta$

(23) $<$ is irreflexive, transitive (and asymmetric)

Replacing Merge with Subjoin

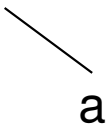
Thinking back to Curry's notion that the relevant formal objects have a hierarchical structure, I'll pursue the idea that that structure is organized via parthood not set membership, so syntactic objects are mereological, not set-theoretic. The operation is then one that creates part-relations.

(22) Subjoin(α,β) $\Rightarrow$ β modified so that $\alpha < \beta$

(23) $<$ is irreflexive, transitive (and asymmetric)

We can think of this as 'inserting' one object into another:

(24) Subjoin(*a*,*b*) $\Rightarrow$ b



Replacing Merge with Subjoin

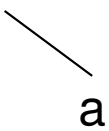
Thinking back to Curry's notion that the relevant formal objects have a hierarchical structure, I'll pursue the idea that that structure is organized via parthood not set membership, so syntactic objects are mereological, not set-theoretic. The operation is then one that creates part-relations.

(22) Subjoin(α,β) $\Rightarrow$ β modified so that $\alpha < \beta$

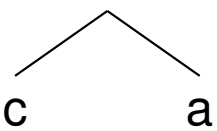
(23) $<$ is irreflexive, transitive (and asymmetric)

We can think of this as 'inserting' one object into another:

(24) Subjoin(*a*,*b*) $\Rightarrow$



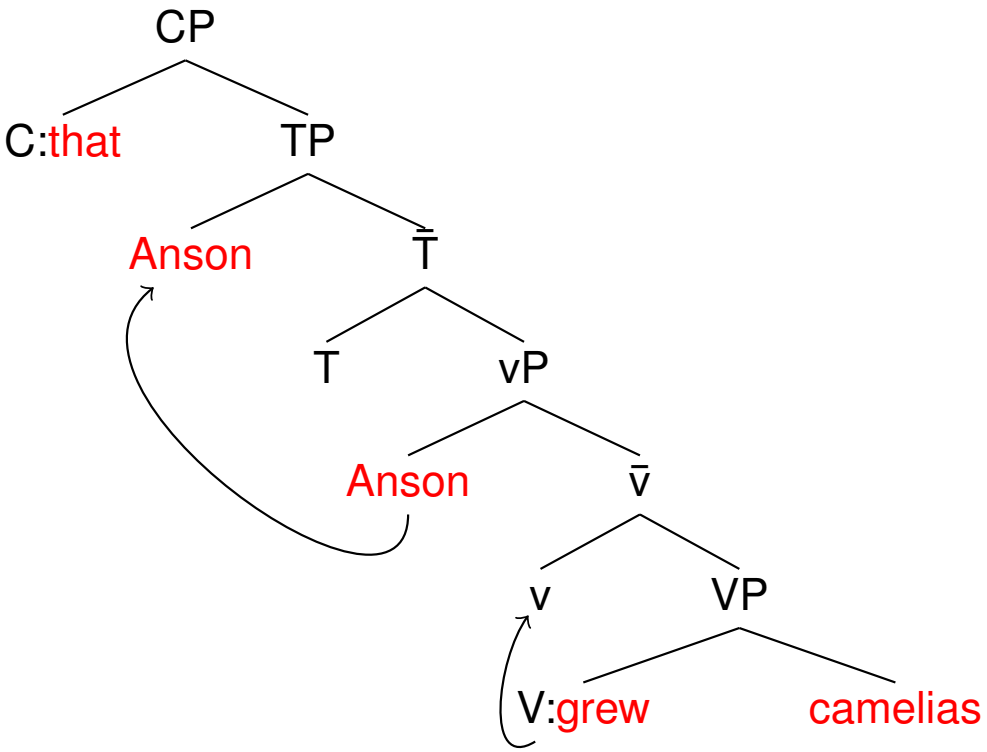
Subjoin(*c*, *b*) $\Rightarrow$



We are adding part relations to an object (after subjunction, *b* has *a* as a part)

Standard Analysis of a Sentence

(25) ... that Anson grew camelias



- v introduces an event variable e; its specifier is interpreted as the Agent of e (Kratzer 1996)
- Different positions of subjects and verbs in different languages arise via movement (Emonds 1978, Koopman and Sportiche 1991)
- T adds temporal semantics and morphology to the verb (Stowell 1993)
- V-v-T-(C) are the Extended Projection of the verb (Grimshaw 1990)

'Officially' this is a big complex set-theoretic structure

Dimensionality

How can we rethink this using Subjoin?

(26) Dimensionality (Part 1)
The first application of Subjoin to an object x is in dimension 1 (a 1-part: Extended Projection Complement, but Transitive)

Dimensionality

How can we rethink this using Subjoin?

(26) Dimensionality (Part 1)
The first application of Subjoin to an object x is in dimension 1 (a 1-part: Extended Projection Complement, but Transitive)

Take a verb like *zoom* and assume it's the complement of v:

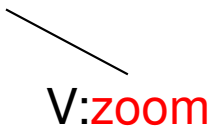
Dimensionality

How can we rethink this using Subjoin?

(26) Dimensionality (Part 1)
The first application of Subjoin to an object x is in dimension 1 (a 1-part: Extended Projection Complement, but Transitive)

Take a verb like *zoom* and assume it's the complement of v:

(27) Subjoin(V,v) $\Rightarrow$ v



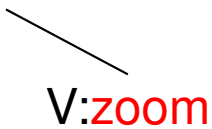
Dimensionality

How can we rethink this using Subjoin?

(26) Dimensionality (Part 1)
The first application of Subjoin to an object x is in dimension 1 (a 1-part: Extended Projection Complement, but Transitive)

Take a verb like *zoom* and assume it's the complement of v:

(27) Subjoin(V,v) $\Rightarrow$ v



Note: the Labels issue goes away since no new labelless object is constructed (cf. {v, zoom})

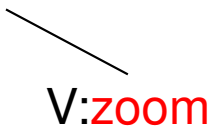
Dimensionality

How can we rethink this using Subjoin?

(26) Dimensionality (Part 1)
The first application of Subjoin to an object x is in dimension 1 (a 1-part: Extended Projection Complement, but Transitive)

Take a verb like *zoom* and assume it's the complement of v:

(27) Subjoin(V,v) $\Rightarrow$ v



Note: the Labels issue goes away since no new labelless object is constructed (cf. {v, zoom})
How is such a structure linearized and phonologized? Come back to this.

Dimensionality

(28) Dimensionality (Part 2)
If an object x already has a part in dimension 1, then an application of Subjoin to x will be in dimension 2 (a 2-part: sort-of Specifier, but Transitive).

Dimensionality

(28) Dimensionality (Part 2)
If an object x already has a part in dimension 1, then an application of Subjoin to x will be in dimension 2 (a 2-part: sort-of Specifier, but Transitive).

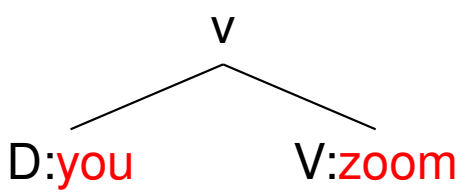
Add the (agent) subject, say a pronoun *you*, as the 'spec' of v, in *you zoom*:

Dimensionality

(28) Dimensionality (Part 2)
If an object x already has a part in dimension 1, then an application of Subjoin to x will be in dimension 2 (a 2-part: sort-of Specifier, but Transitive).

Add the (agent) subject, say a pronoun *you*, as the 'spec' of v, in *you zoom*:

(29)

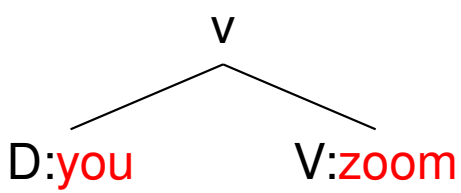


Dimensionality

(28) Dimensionality (Part 2)
If an object x already has a part in dimension 1, then an application of Subjoin to x will be in dimension 2 (a 2-part: sort-of Specifier, but Transitive).

Add the (agent) subject, say a pronoun *you*, as the 'spec' of v, in *you zoom*:

(29)



(30) Dimensionality (Part 3)
The maximal dimension is 2 (for now)

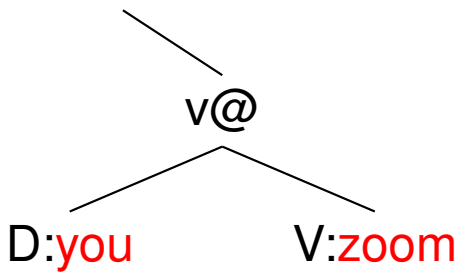
Subjunction

Now add tense information in, assuming v is the complement of T, in *You zoomed*.

Subjunction

Now add tense information in, assuming v is the complement of T, in *You zoomed*.

(31) Subjoin(v, T) $\Rightarrow$



The 'span' (string in dimension 1) zoom⌢v@⌢/D/ is interpreted by the phonology and linearized at the point in the structure marked with @ (following Brody 2000).

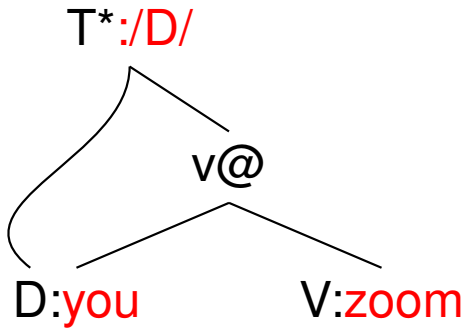
Subjunction

Assuming the standard analysis where the subject is a specifier of T, we derive this by simply subjoining it to T (adding another part relation). In English, T's specifier is privileged for linearization, marked with a * diacritic

Subjunction

Assuming the standard analysis where the subject is a specifier of T, we derive this by simply subjoining it to T (adding another part relation). In English, T's specifier is privileged for linearization, marked with a * diacritic

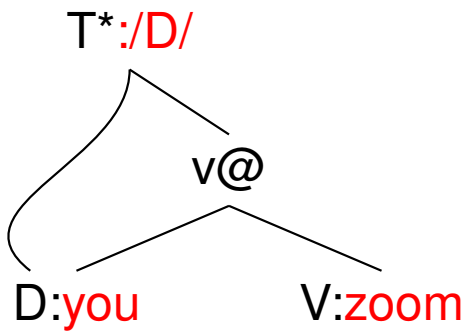
(32) Subjoin(v, T) $\Rightarrow$



Subjunction

Assuming the standard analysis where the subject is a specifier of T, we derive this by simply subjoining it to T (adding another part relation). In English, T's specifier is privileged for linearization, marked with a * diacritic

(32) Subjoin(v, T) $\Rightarrow$

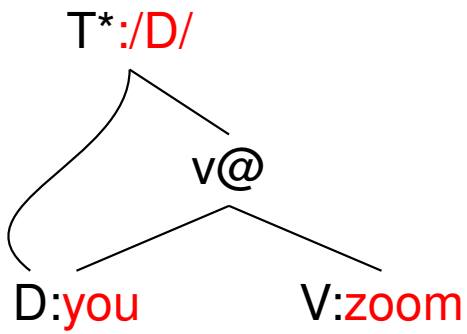


This 'Internal Subjunction' is the analogous operation to 'Internal Merge' so we keep the single mechanism idea

Subjunction

Assuming the standard analysis where the subject is a specifier of T, we derive this by simply subjoining it to T (adding another part relation). In English, T's specifier is privileged for linearization, marked with a * diacritic

(32) Subjoin(v, T) $\Rightarrow$



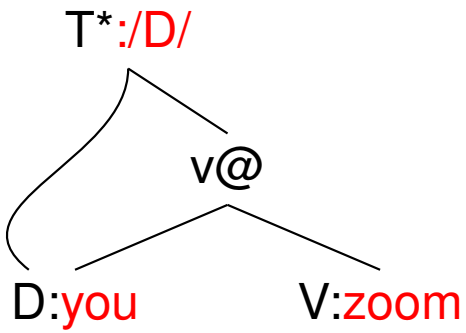
This 'Internal Subjunction' is the analogous operation to 'Internal Merge' so we keep the single mechanism idea

Again, thinking of Curry's point that 'there are various ways in which [a structure] can be associated with a string', how do we linearize this?

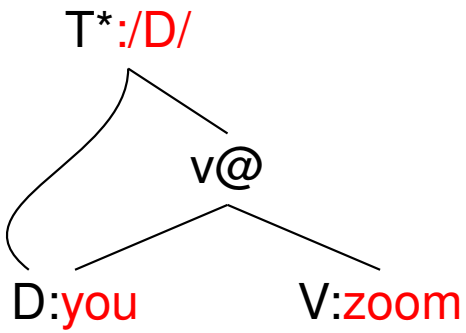
Linearization

For English:

- * says put T's 2-part (spec) to the left of (the linearization of) T.



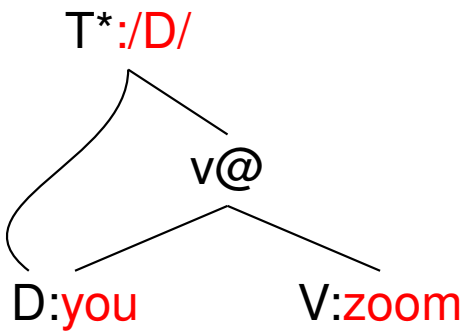
Linearization



For English:

- * says put T's 2-part (spec) to the left of (the linearization of) T.
- T without @ has a zero exponent which is to the left of its complement.

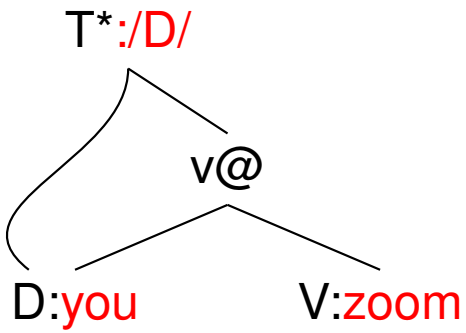
Linearization



For English:

- * says put T's 2-part (spec) to the left of (the linearization of) T.
- T without @ has a zero exponent which is to the left of its complement.
- v@ says put all of the zoom⌒/D/ information at v.

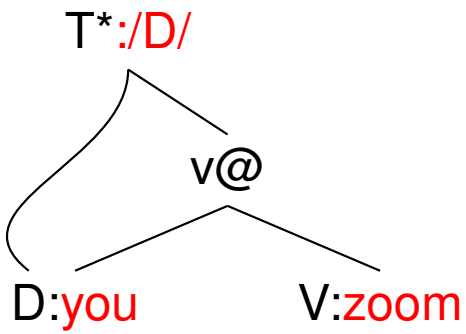
Linearization



For English:

- * says put T's 2-part (spec) to the left of (the linearization of) T.
- T without @ has a zero exponent which is to the left of its complement.
- v@ says put all of the zoom⌒/D/ information at v.
- v's 2-part is already linearized so no change

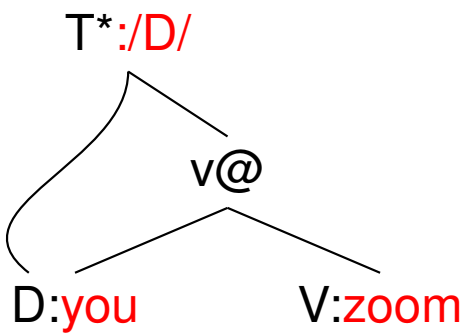
Linearization



For English:

- * says put T's 2-part (spec) to the left of (the linearization of) T.
- T without @ has a zero exponent which is to the left of its complement.
- v@ says put all of the zoom⌒/D/ information at v.
- v's 2-part is already linearized so no change
- V without @ has a zero exponent

Linearization



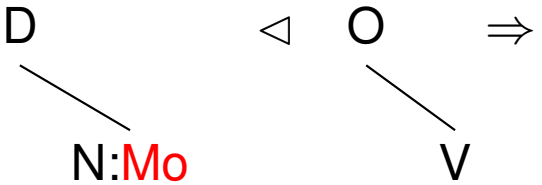
For English:

- * says put T's 2-part (spec) to the left of (the linearization of) T.
- T without @ has a zero exponent which is to the left of its complement.
- v@ says put all of the zoom⌢/D/ information at v.
- v's 2-part is already linearized so no change
- V without @ has a zero exponent
- The result for the phonology is the (abstract sequence) *you 0⌢zoom+/D/⌢0*.

Subjunction

An unaccusative like *grow* will work just the same. So think of our previous example of *Mo grew*. Assume that the argument of an unaccusative is the specifier of a functional head, as Borer, Ramchand, and others have argued. I'll just call that head O, leaving it to others to figure out the details:

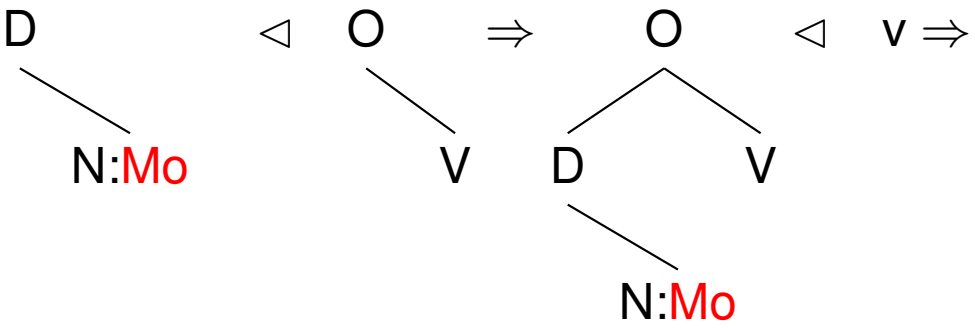
(33)



Subjunction

An unaccusative like *grow* will work just the same. So think of our previous example of *Mo grew*. Assume that the argument of an unaccusative is the specifier of a functional head, as Borer, Ramchand, and others have argued. I'll just call that head O, leaving it to others to figure out the details:

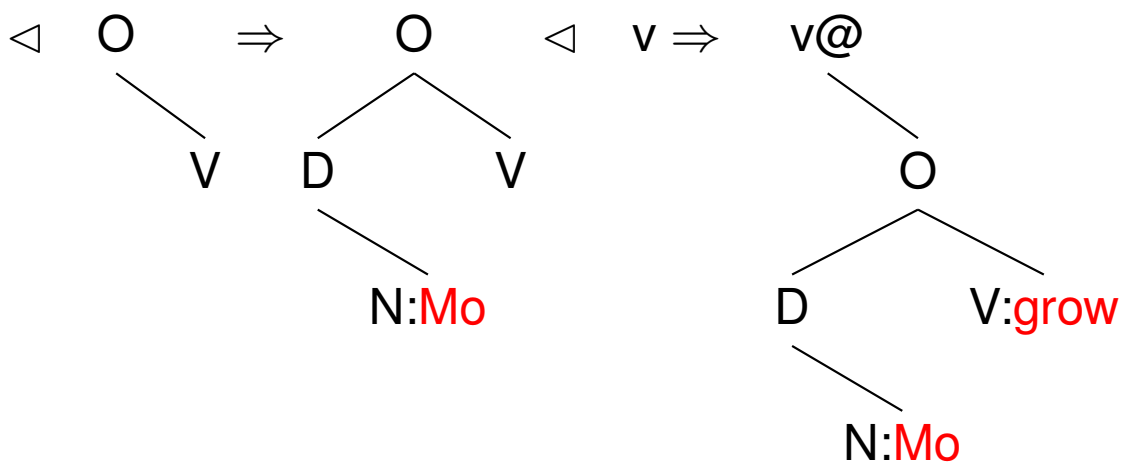
(33)



Subjunction

An unaccusative like *grow* will work just the same. So think of our previous example of *Mo grew*. Assume that the argument of an unaccusative is the specifier of a functional head, as Borer, Ramchand, and others have argued. I'll just call that head O, leaving it to others to figure out the details:

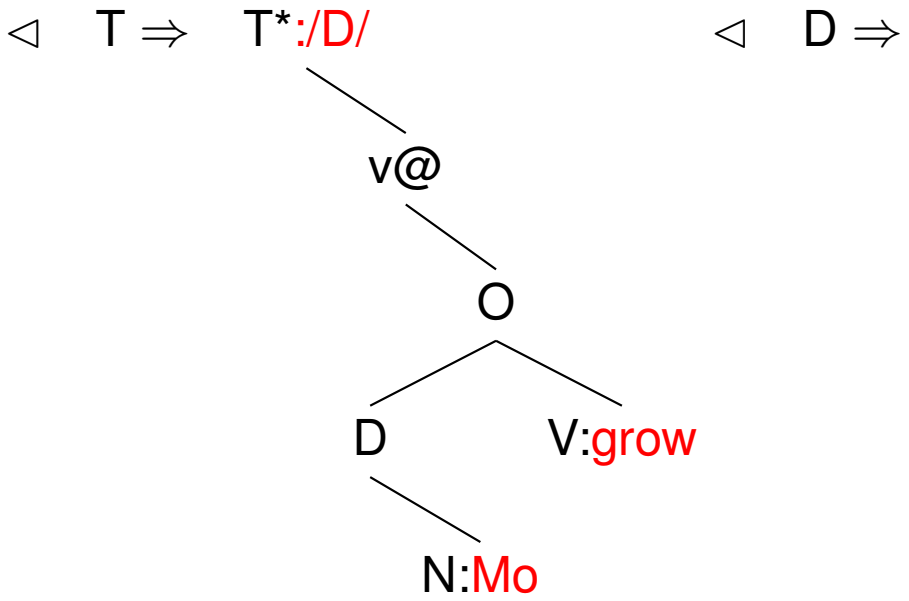
(33)



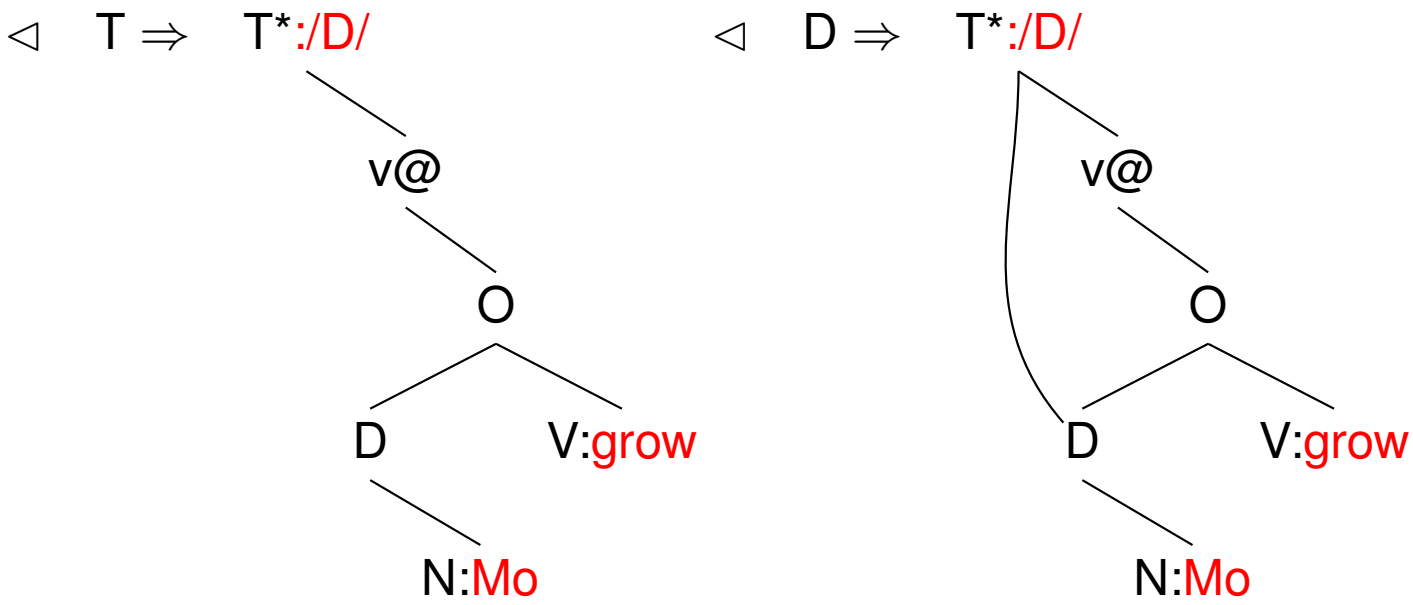
Subjunction

$\triangleleft \quad T \Rightarrow$

Subjunction



Subjunction



The ‘span’ of categories V⌢O⌢v@⌢T has a single associated exponent grew (Svenonius passim)

Copies vs Repetitions Again

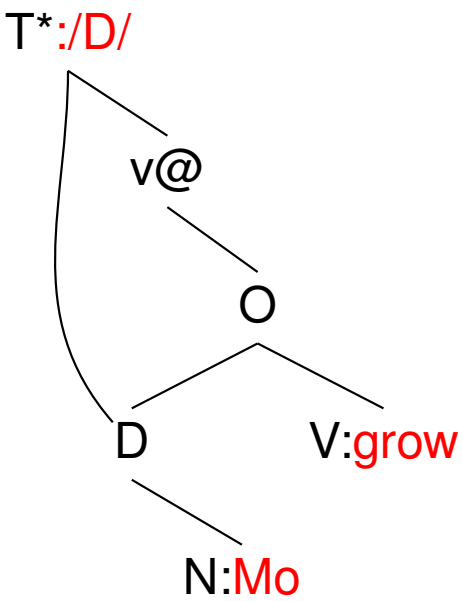
Internal Subjunction of the unaccusative subject vs External Subjunction of an unergative subject result in quite different structures: so no copy vs repetition problem in this system

Copies vs Repetitions Again

Internal Subjunction of the unaccusative subject vs External Subjunction of an unergative subject result in quite different structures: so no copy vs repetition problem in this system

(34) a. Mo grew

b.

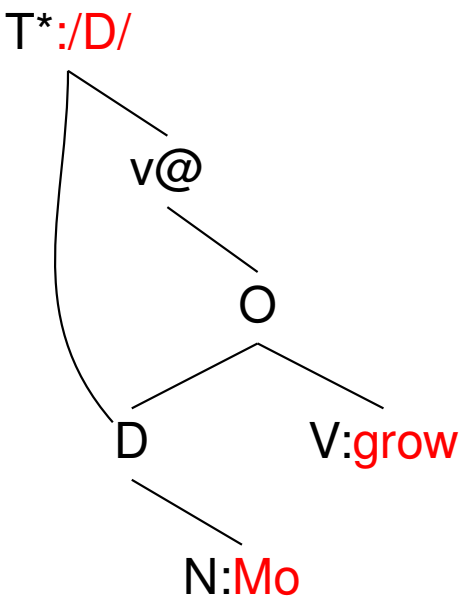


Copies vs Repetitions Again

Internal Subjunction of the unaccusative subject vs External Subjunction of an unergative subject result in quite different structures: so no copy vs repetition problem in this system

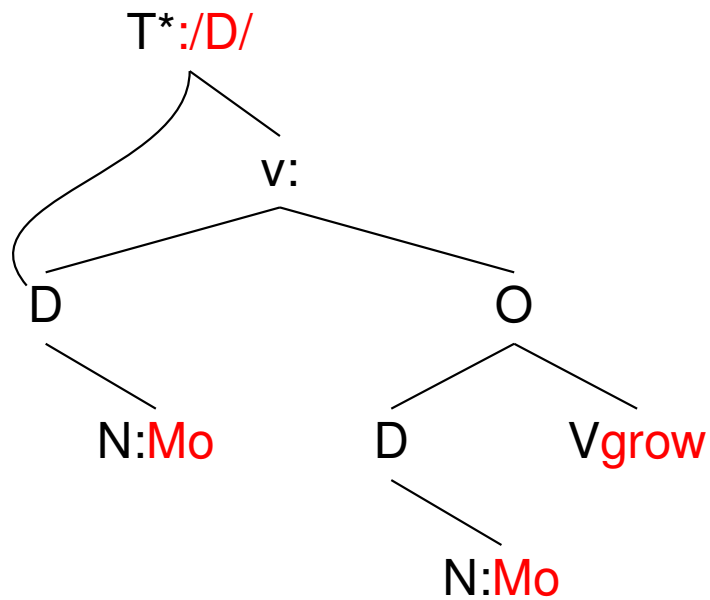
(34) a. Mo grew

b.



(35) a. Mo grew Mo

b.

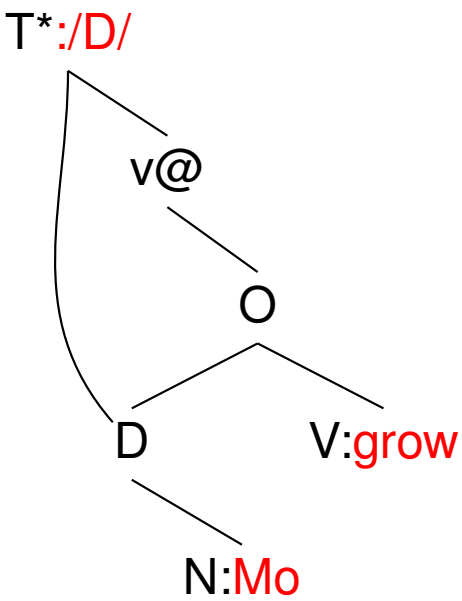


Copies vs Repetitions Again

Internal Subjunction of the unaccusative subject vs External Subjunction of an unergative subject result in quite different structures: so no copy vs repetition problem in this system

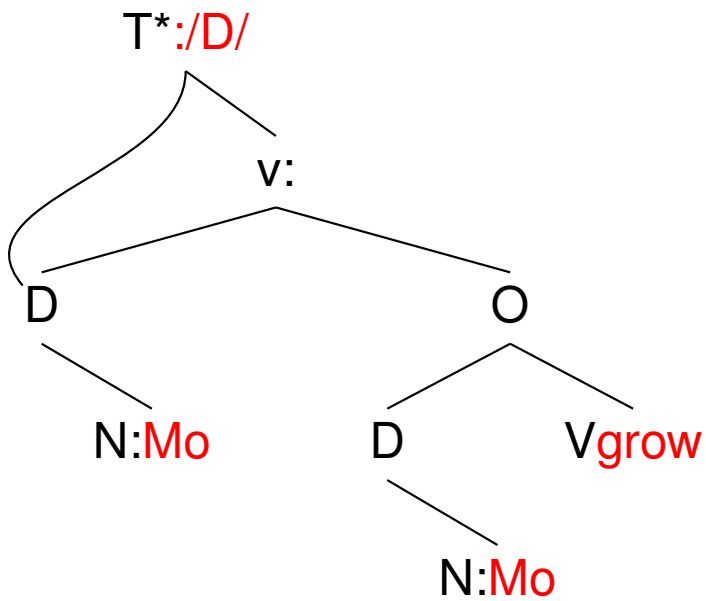
(34) a. Mo grew

b.



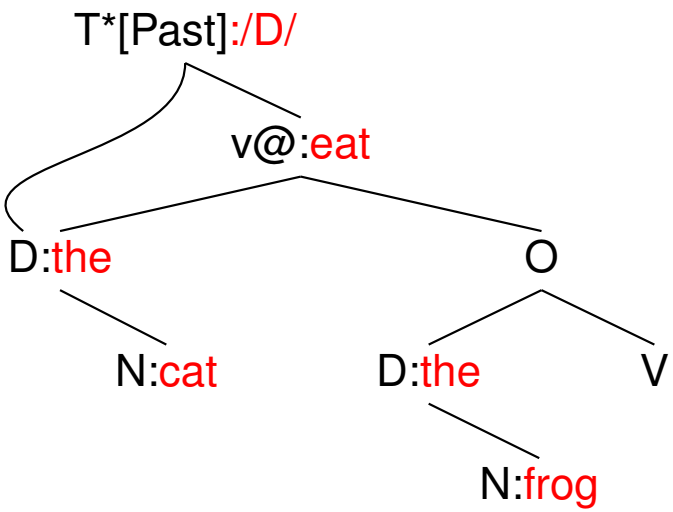
(35) a. Mo grew Mo

b.



(36) { Mo { T { Mo {v {grow, Mo} }}}}

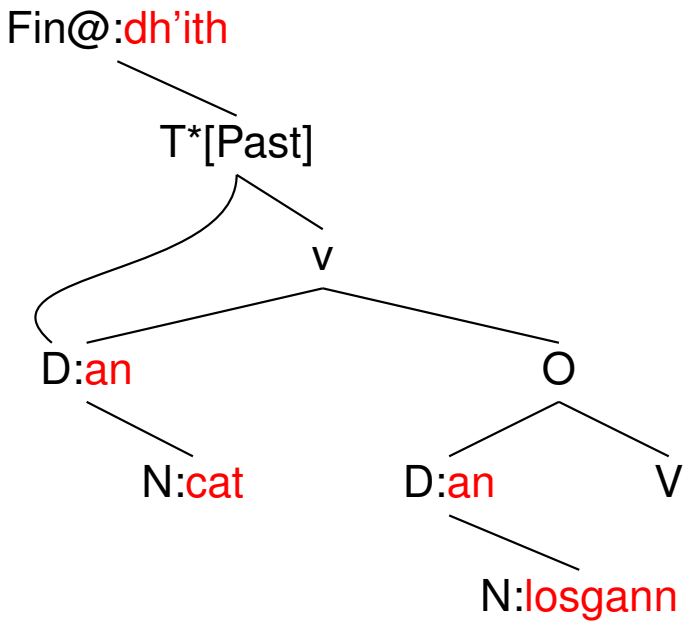
Cross Linguistic Variation



- In English, the verb is pronounced ‘at’ v, and the subject is subjoined to and pronounced ‘as the specifier of’ T
- In English, D is linearized to the left of the linearization of its complement, so the⌢cat precedes eat⌢/D/ which itself precedes the⌢frog

Cross-Linguistic Variation

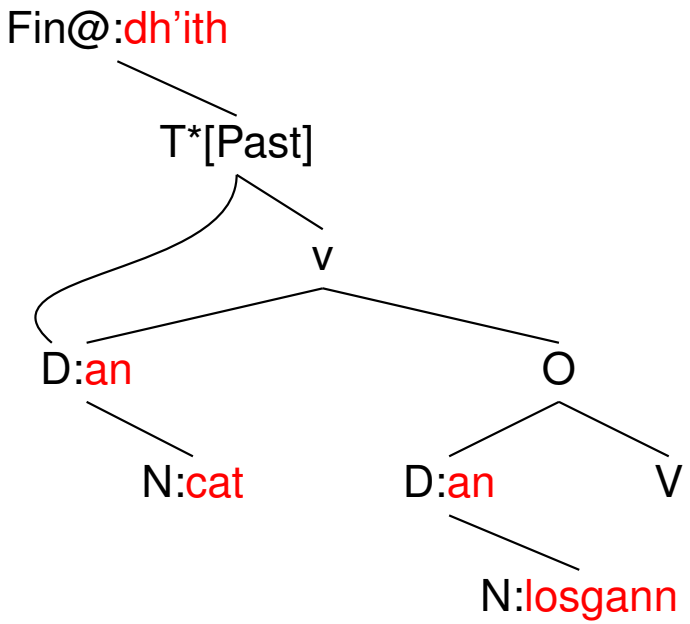
Scottish Gaelic VSO vs Malayalam SOV



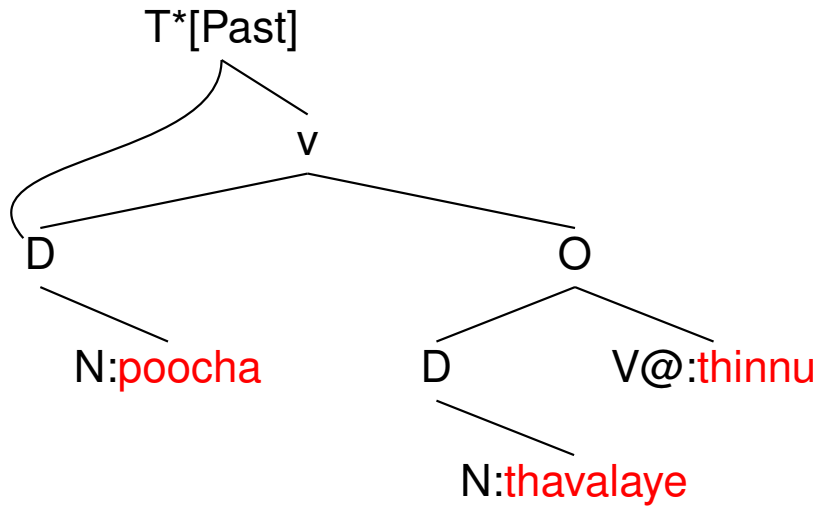
(37) dh'ith an cat an losgann
eat.PAST the cat the frog
'the cat ate the frog'

Cross-Linguistic Variation

Scottish Gaelic VSO vs Malayalam SOV



(37) dh'ith an cat an losgann
eat.PAST the cat the frog
'the cat ate the frog'



(38) poocha thavalaye thinnu
cat frog eat.PAST
'the cat ate the frog'

Locality and Transitivity

(39) Subjoin(α,β) $\Rightarrow$ β modified so that $\alpha < \beta$

Locality and Transitivity

(39) Subjoin(α,β) $\Rightarrow$ β modified so that $\alpha < \beta$

- Internal Subjunction is when α is part of β

Locality and Transitivity

(39) Subjoin(α,β) $\Rightarrow$ β modified so that $\alpha < \beta$

- Internal Subjunction is when α is part of β
- We have seen there are 2 dimensions

Locality and Transitivity

(39) Subjoin(α,β) $\Rightarrow$ β modified so that $\alpha < \beta$

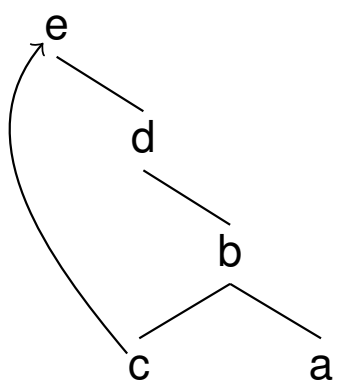
- Internal Subjunction is when α is part of β
- We have seen there are 2 dimensions
- my proposal for locality of internal subjunction is that at most one dimension can be crossed

Locality and Transitivity

(39) Subjoin(α,β) $\Rightarrow$ β modified so that $\alpha < \beta$

- Internal Subjunction is when α is part of β
- We have seen there are 2 dimensions
- my proposal for locality of internal subjunction is that at most one dimension can be crossed

(40) (a)

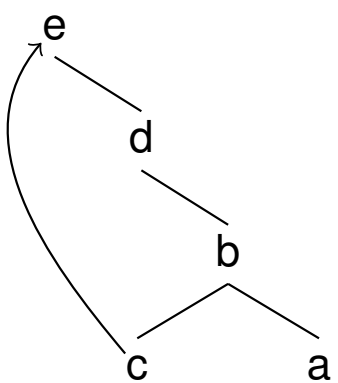


Locality and Transitivity

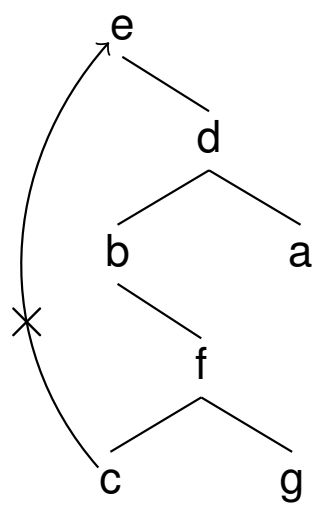
(39) Subjoin(α,β) $\Rightarrow$ β modified so that $\alpha < \beta$

- Internal Subjunction is when α is part of β
- We have seen there are 2 dimensions
- my proposal for locality of internal subjunction is that at most one dimension can be crossed

(40) (a)



(b)

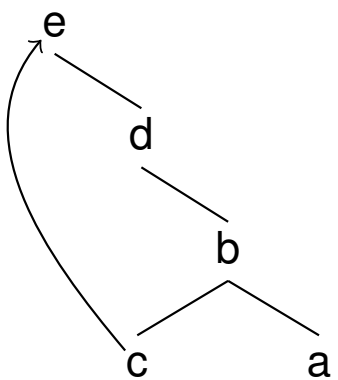


Locality and Transitivity

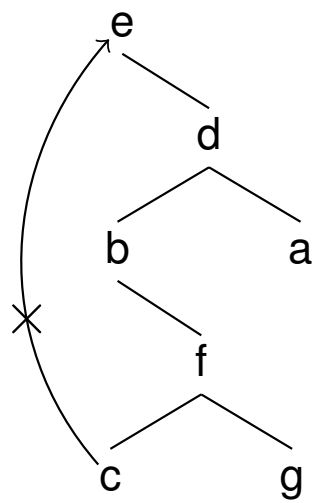
(39) Subjoin(α,β) $\Rightarrow$ β modified so that $\alpha < \beta$

- Internal Subjunction is when α is part of β
- We have seen there are 2 dimensions
- my proposal for locality of internal subjunction is that at most one dimension can be crossed

(40) (a)



(b)



Angular Locality

(41) if γ is a part, then it can subjoin to β only if there is an α s.t. γ is an n-part of α and α is a 1-part of β

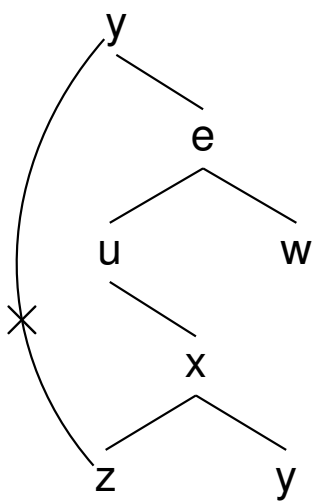
Successive subjunction

‘Long-distance’ subjunction (from inside a 2-part/‘specfier’ of some object to a higher object (i.e. subjunction across extended projections) is ruled out by Angular Locality

Successive subjunction

'Long-distance' subjunction (from inside a 2-part/'specfier' of some object to a higher object (i.e. subjunction across extended projections) is ruled out by Angular Locality

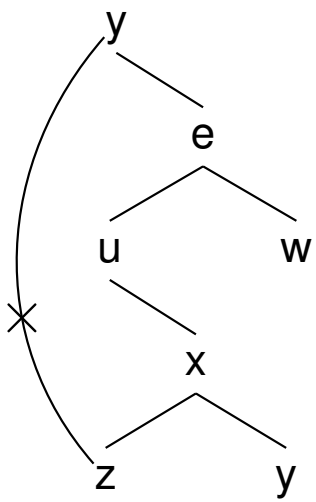
(42) (a)



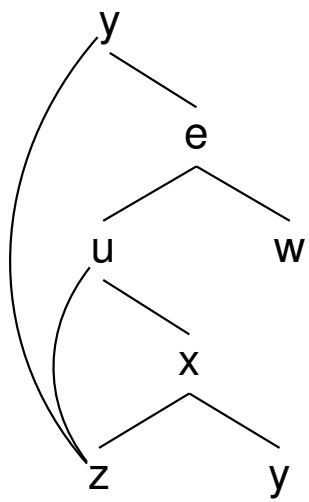
Successive subjunction

'Long-distance' subjunction (from inside a 2-part/'specfier' of some object to a higher object (i.e. subjunction across extended projections) is ruled out by Angular Locality

(42) (a)



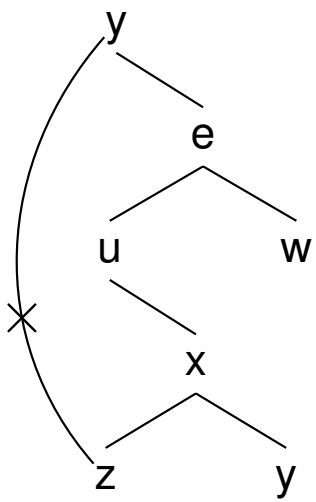
(b)



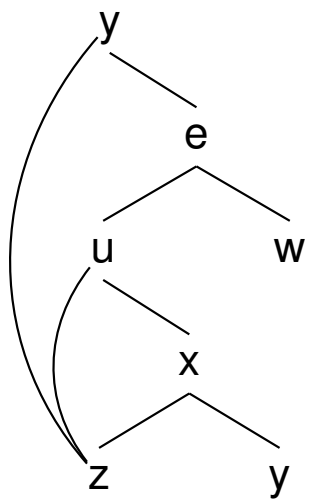
Successive subjunction

'Long-distance' subjunction (from inside a 2-part/'specfier' of some object to a higher object (i.e. subjunction across extended projections) is ruled out by Angular Locality

(42) (a)



(b)



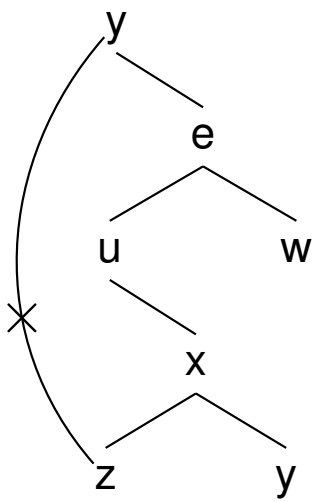
Additional consequences:

(43) a. It is just a theorem of the system that 'Exceptions' to Locality are the 'edges' of a domain

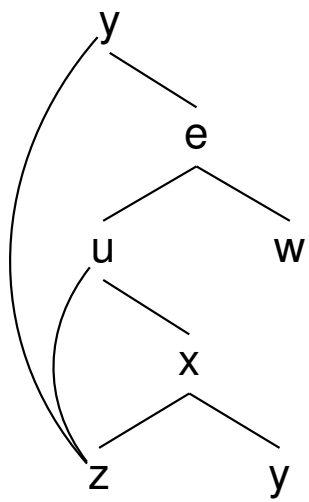
Successive subjunction

'Long-distance' subjunction (from inside a 2-part/'specfier' of some object to a higher object (i.e. subjunction across extended projections) is ruled out by Angular Locality

(42) (a)



(b)



Additional consequences:

(43) a. It is just a theorem of the system that 'Exceptions' to Locality are the 'edges' of a domain

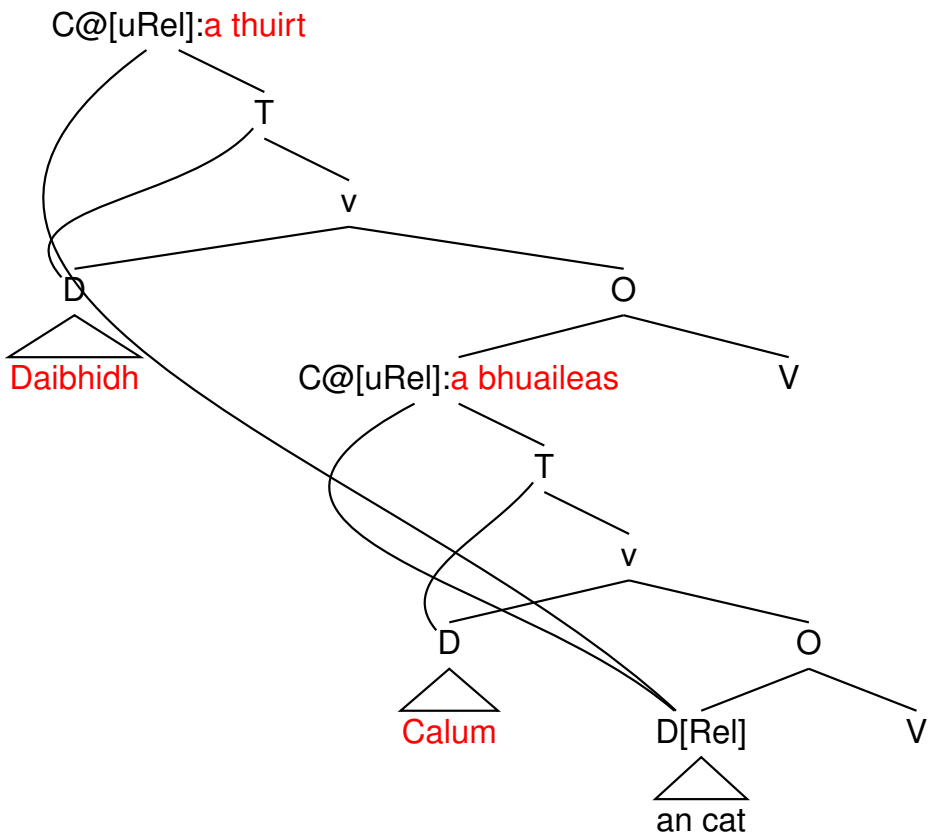
b. Dimensionality entails that there is only one 'escape hatch' (objects can have only one 2-part, and lower objects are inaccessible because Dimensionality restricts transitivity)

Reflexes of Successive Cyclicity

(44) An cat a thuirt Daibhidh a/*gu(m) bhuaileas Calum
the cat REL say.PAST David REL/*THAT hit.FUT.REL Calum
'The cat that David said Calum will hit.'

Reflexes of Successive Cyclicity

(44) An cat a thuirt Daibhidh a/*gu(m) bhuaileas Calum
the cat REL say.PAST David REL/*THAT hit.FUT.REL Calum
'The cat that David said Calum will hit.'



Those three issues Revisited

Diagnosis: Merge unifyies the theory of local and non-local dependency in a way that other theories do not, but ...

Those three issues Revisited

Diagnosis: Merge unifyies the theory of local and non-local dependency in a way that other theories do not, but ...

(45) a. We need an extra theory to label the output of Merge, and that has turned out to be less simple than we'd like

Those three issues Revisited

Diagnosis: Merge unifyies the theory of local and non-local dependency in a way that other theories do not, but ...

(45) a. We need an extra theory to label the output of Merge, and that has turned out to be less simple than we'd like
Subjoin maintains the label of the target

Those three issues Revisited

Diagnosis: Merge unifyies the theory of local and non-local dependency in a way that other theories do not, but ...

(45) a. We need an extra theory to label the output of Merge, and that has turned out to be less simple than we'd like
Subjoin maintains the label of the target

b. We need an extra theory to distinguish copies from repetitions (*Mo grew* vs *Mo grew Mo*), and that has turned out to be quite difficult

Those three issues Revisited

Diagnosis: Merge unifyies the theory of local and non-local dependency in a way that other theories do not, but ...

(45) a. We need an extra theory to label the output of Merge, and that has turned out to be less simple than we'd like
Subjoin maintains the label of the target

b. We need an extra theory to distinguish copies from repetitions (*Mo grew* vs *Mo grew Mo*), and that has turned out to be quite difficult
Subjoin generates different syntactic representations

Those three issues Revisited

Diagnosis: Merge unifyies the theory of local and non-local dependency in a way that other theories do not, but ...

(45)
a. We need an extra theory to label the output of Merge, and that has turned out to be less simple than we'd like
Subjoin maintains the label of the target
b. We need an extra theory to distinguish copies from repetitions (*Mo grew* vs *Mo grew Mo*), and that has turned out to be quite difficult
Subjoin generates different syntactic representations
c. We need an extra theory of Phasehood to capture successive cyclicity effects

Those three issues Revisited

Diagnosis: Merge unifyies the theory of local and non-local dependency in a way that other theories do not, but ...

(45)
a. We need an extra theory to label the output of Merge, and that has turned out to be less simple than we'd like
Subjoin maintains the label of the target
b. We need an extra theory to distinguish copies from repetitions (*Mo grew* vs *Mo grew Mo*), and that has turned out to be quite difficult
Subjoin generates different syntactic representations
c. We need an extra theory of Phasehood to capture successive cyclicity effects
Locality domains are just 2-parts (specifiers)

Those three issues Revisited

Diagnosis: Merge unifyies the theory of local and non-local dependency in a way that other theories do not, but ...

(45)
a. We need an extra theory to label the output of Merge, and that has turned out to be less simple than we'd like
Subjoin maintains the label of the target
b. We need an extra theory to distinguish copies from repetitions (*Mo grew* vs *Mo grew Mo*), and that has turned out to be quite difficult
Subjoin generates different syntactic representations
c. We need an extra theory of Phasehood to capture successive cyclicity effects
Locality domains are just 2-parts (specifiers)
Locality is incorporated into the theory of phrase structure with the basic property of transitivity of 2-parthood giving escape hatches

Those three issues Revisited

Diagnosis: Merge unifyies the theory of local and non-local dependency in a way that other theories do not, but ...

(45)
a. We need an extra theory to label the output of Merge, and that has turned out to be less simple than we'd like
Subjoin maintains the label of the target
b. We need an extra theory to distinguish copies from repetitions (*Mo grew* vs *Mo grew Mo*), and that has turned out to be quite difficult
Subjoin generates different syntactic representations
c. We need an extra theory of Phasehood to capture successive cyclicity effects
Locality domains are just 2-parts (specifiers)
Locality is incorporated into the theory of phrase structure with the basic property of transitivity of 2-parthood giving escape hatches
d. We need a bunch of extra principles to capture islands because Merge allows circumvention of phases

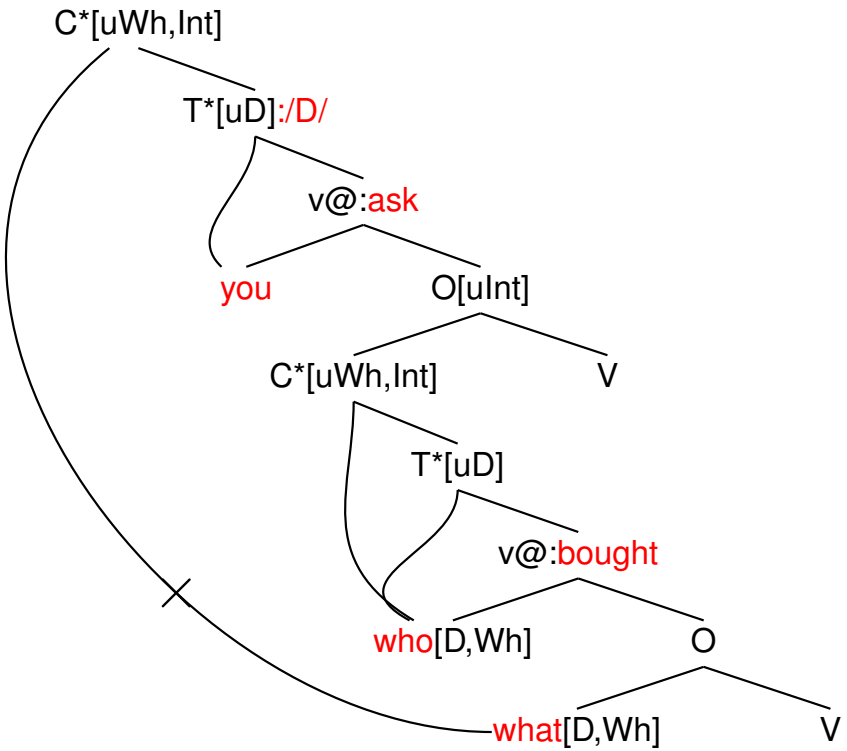
Those three issues Revisited

Diagnosis: Merge unifyies the theory of local and non-local dependency in a way that other theories do not, but ...

(45)
a. We need an extra theory to label the output of Merge, and that has turned out to be less simple than we'd like
Subjoin maintains the label of the target
b. We need an extra theory to distinguish copies from repetitions (*Mo grew* vs *Mo grew Mo*), and that has turned out to be quite difficult
Subjoin generates different syntactic representations
c. We need an extra theory of Phasehood to capture successive cyclicity effects
Locality domains are just 2-parts (specifiers)
Locality is incorporated into the theory of phrase structure with the basic property of transitivity of 2-parthood giving escape hatches
d. We need a bunch of extra principles to capture islands because Merge allows circumvention of phases
Dimensionality puts an upper limit on specifiers allowing us to begin to think about islands in terms of cyclicity, hence phrase structure, once more

Deriving Wh-Islands

(46) *(guess) what you asked who bought?



Wh-island Violations

Complex (D-linked, Pesetsky 1987) wh-DPs/relatives violate wh-islands (Maling 1978, Rizzi 1982, Cinque 1990 etc.):

(47) a. Which cake did Ans forget [that we had put ⟨which cake⟩ in the fridge]

b. ?/% Which cake did Ans forget [who had put ⟨which cake⟩ in the fridge]

c. ?/% This is the cake that Ans forgot [who had put ⟨which cake⟩ in the fridge]

Why is escape possible from wh-islands? (also Norwegian, Danish, Hebrew etc.)

(48) eyze sefer Yoni taha le-mi ha-mora natna
which book Yoni wondered to-whom the-teacher gave
'Which book did John wonder to whom the teacher gave?' Hebrew

An ‘extra’ escape hatch approach

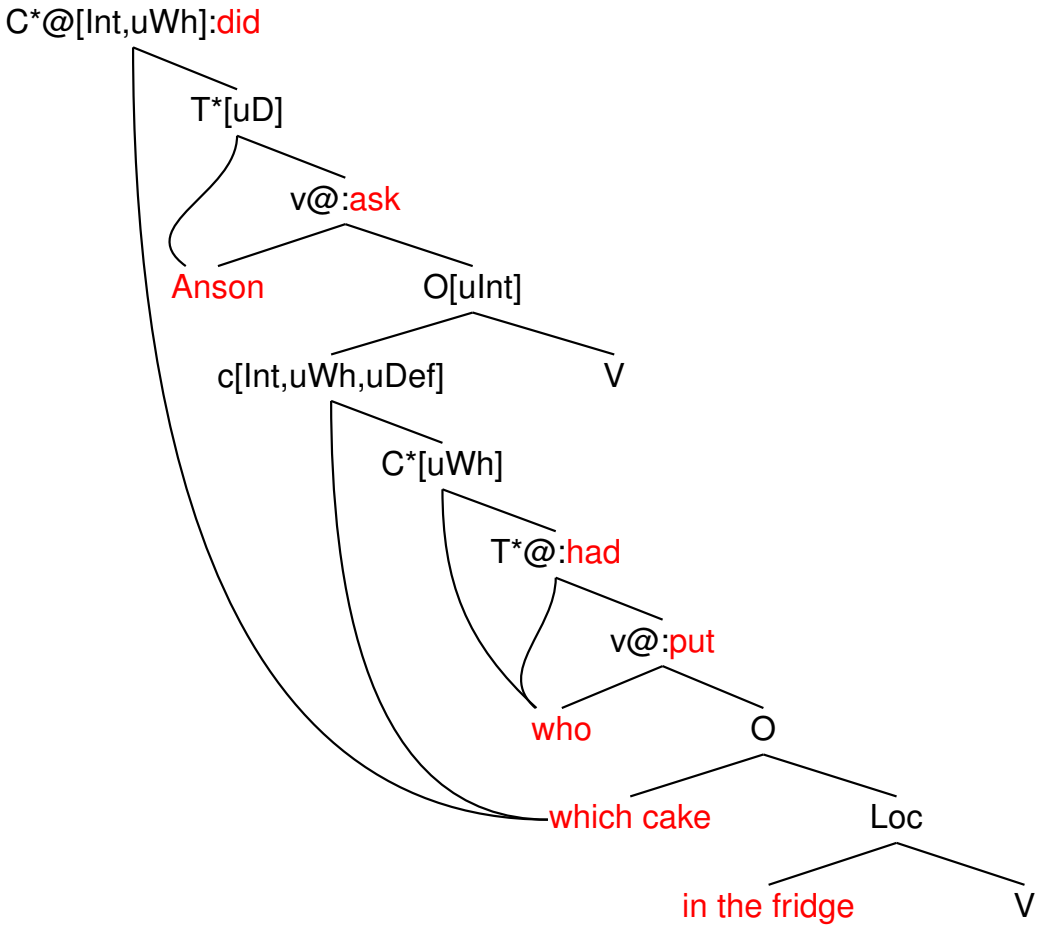
An extra category c[uDef] above the apparent edge of embedded question (cf. Nyvad et al. (2017))

(49) Which cake did Anson ask who had put in the fridge?

An ‘extra’ escape hatch approach

An extra category c[uDef] above the apparent edge of embedded question (cf. Nyvad et al. (2017))

(49) Which cake did Anson ask who had put in the fridge?



Wh-Island Re-Emergence (WIRE)

The system disallows (because of Dimensionality) two elements to both extract out of a locality domain. Predicts, correctly, the following effect:

Wh-Island Re-Emergence (WIRE)

The system disallows (because of Dimensionality) two elements to both extract out of a locality domain. Predicts, correctly, the following effect:

(50) a. Which cake [$_{CP}$ were we told [$_{CP}$ ⟨which cake⟩ that we should enquire [$_{cP}$ ⟨which cake⟩ [$_{CP}$ **where** we should put ⟨which cake⟩ ⟨where⟩]]]

b. *Which cake [$_{CP}$ did we enquire [$_{CP}$ **where** she was told [$_{cP}$ ⟨which cake⟩ [$_{CP}$ ⟨where⟩ we should put ⟨which cake⟩ ⟨where⟩]]]

Wh-Island Re-Emergence (WIRE)

The system disallows (because of Dimensionality) two elements to both extract out of a locality domain. Predicts, correctly, the following effect:

(50) a. Which cake [$_{CP}$ were we told [$_{CP}$ ⟨which cake⟩ that we should enquire [$_{cP}$ ⟨which cake⟩ [$_{CP}$ **where** we should put ⟨which cake⟩ ⟨where⟩]]]

b. *Which cake [$_{CP}$ did we enquire [$_{CP}$ **where** she was told [$_{cP}$ ⟨which cake⟩ [$_{CP}$ ⟨where⟩ we should put ⟨which cake⟩ ⟨where⟩]]]

The bolded **where** can't have subjoined from the lower clause as it already has a 2-part (⟨which cake⟩)

Wh-Island Re-Emergence (WIRE)

The system disallows (because of Dimensionality) two elements to both extract out of a locality domain. Predicts, correctly, the following effect:

(50) a. Which cake [$_{CP}$ were we told [$_{CP}$ ⟨which cake⟩ that we should enquire [$_{cP}$ ⟨which cake⟩ [$_{CP}$ **where** we should put ⟨which cake⟩ ⟨where⟩]]]

b. *Which cake [$_{CP}$ did we enquire [$_{CP}$ **where** she was told [$_{cP}$ ⟨which cake⟩ [$_{CP}$ ⟨where⟩ we should put ⟨which cake⟩ ⟨where⟩]]]

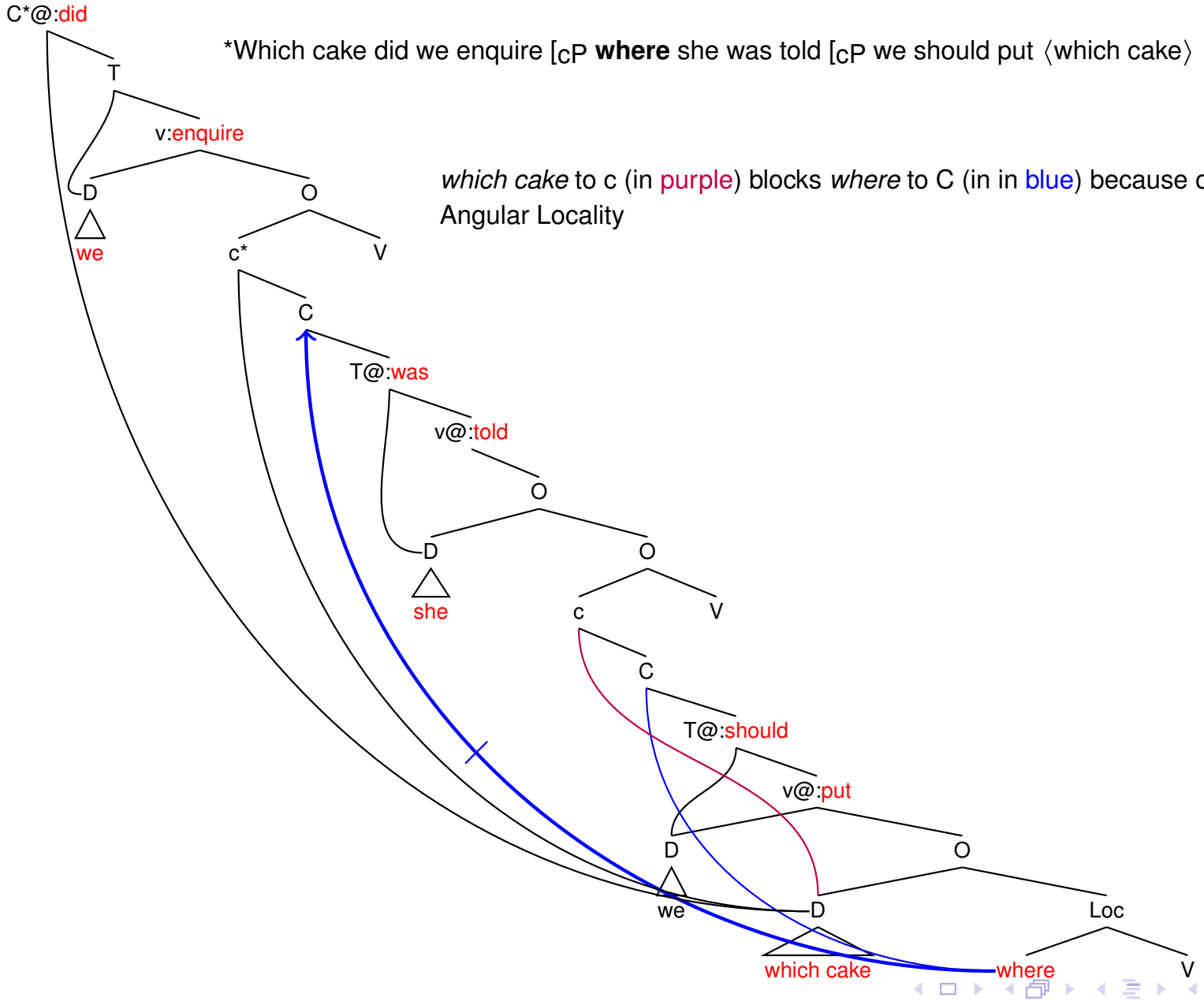
The bolded **where** can't have subjoined from the lower clause as it already has a 2-part (⟨which cake⟩)

(51) Wh-Island Re-Emergence (WIRE):
If it is possible to form a long distance dependency across a wh-island, the expression that induces the island must remain in its original clause. If the island inducing expression extracts out of its clause, the long(er)-distance extractee cannot cross it.

Explaining the contrast

*Which cake did we enquire [$_{CP}$ **where** she was told [$_{CP}$ we should put ⟨which cake⟩ ⟨**where**⟩

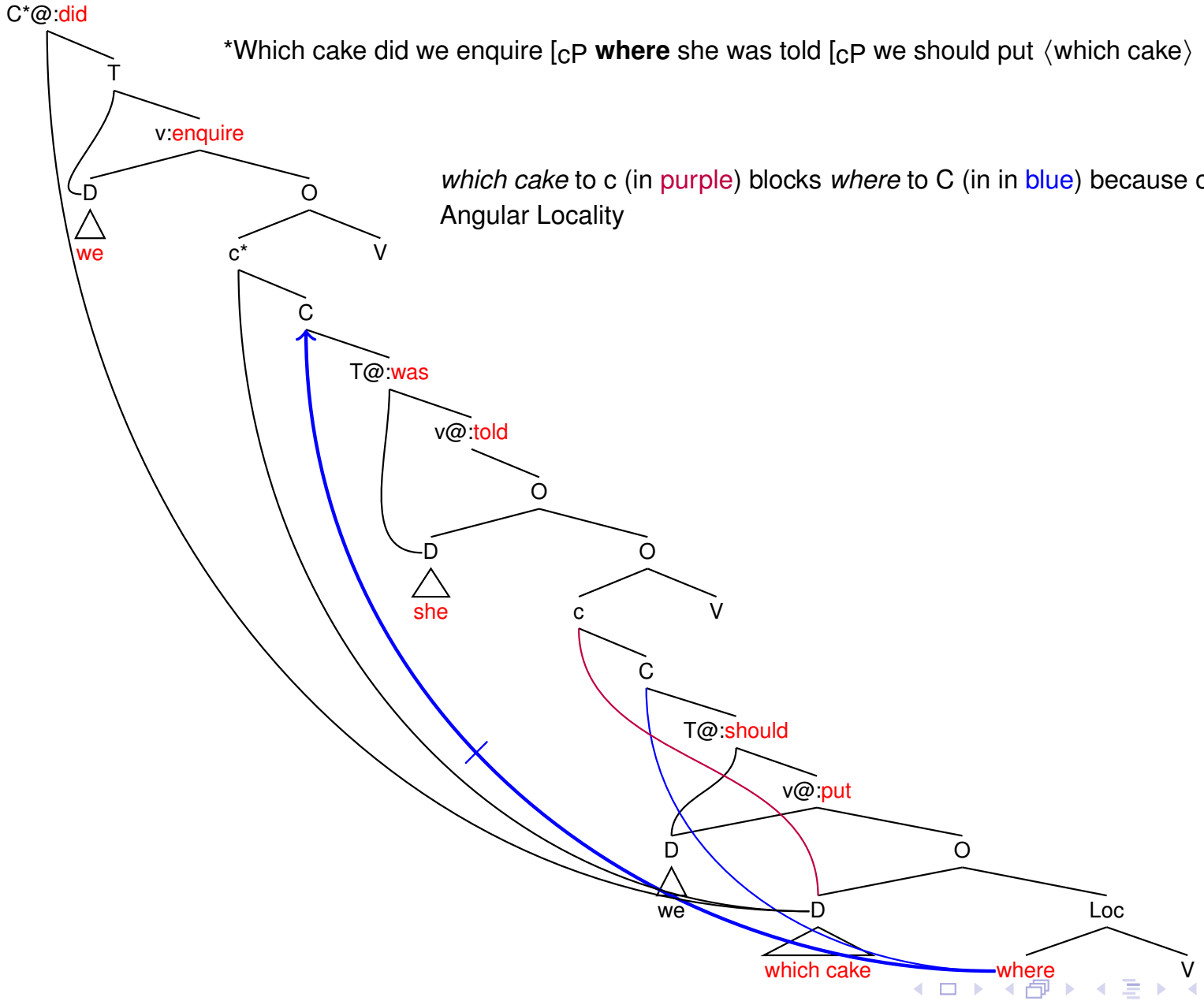
which cake to c (in purple) blocks *where* to C (in in blue) because of Angular Locality



Explaining the contrast

*Which cake did we enquire [$_{CP}$ **where** she was told [$_{CP}$ we should put ⟨which cake⟩ ⟨**where**⟩

which cake to c (in purple) blocks *where* to C (in in blue) because of Angular Locality



Wh-Island Re-Emergence (WIRE)

Same effects in Norwegian, Danish (not shown), Hebrew

(52) *Dette er kaka som vi spurte hvor hun ble bedt om å
this is cake.DEF SOM we asked where she was asked about to
plassere
put.
'This is the cake that we asked where she was asked about to put.'

(53) *eyze sefer$_2$ shaxaxta [$_{CP}$ le-mi$_1$ Rina xashva [$_{CP}$ she-Dan
which book forgot.2SG DAT-who Rina thought that-Dan
shalax t_1 t_2]]
sent
'Which book did you forget to whom Rina thinks that Dan sent?'

A Further Prediction

While it is possible to extract two items from the same clause, one to the edge, one further, it should be impossible to extract three.

A Further Prediction

While it is possible to extract two items from the same clause, one to the edge, one further, it should be impossible to extract three.
Engdahl (1986) gives the magnificent:

(54) Var det Södersjukhuset som hälsovårdsmyndigheterna lät
was it South-Hospital-DEF that Public-Health-Department-DEF was
undersöka [$_{CP}$ **vilka färskvaror** det var oklart [$_{cP}$ ⟨vilka färskvaror⟩ [$_{CP}$
investigating which produce it was unclear
vilken grossist som ⟨*vilken grossist*⟩ levererat ⟨**vilka färskvaror**⟩ till
which caterer that deliver.PRF to
⟨Södersjukhuset⟩

'Was it the South Hospital that the Department of Public Health was investigating which produce it was unclear which caterer had delivered that produce to that place.'

A Further Prediction

While it is possible to extract two items from the same clause, one to the edge, one further, it should be impossible to extract three.
Engdahl (1986) gives the magnificent:

(54) Var det Södersjukhuset som hälsovårdsmyndigheterna lät
was it South-Hospital-DEF that Public-Health-Department-DEF was
undersöka [$_{CP}$ **vilka färskvaror** det var oklart [$_{cP}$ ⟨vilka färskvaror⟩ [$_{CP}$
investigating which produce it was unclear
vilken grossist som ⟨*vilken grossist*⟩ levererat ⟨**vilka färskvaror**⟩ till
which caterer that deliver.PRF to
⟨Södersjukhuset⟩

'Was it the South Hospital that the Department of Public Health was investigating which produce it was unclear which caterer had delivered that produce to that place.'

Engdalh remarks that (54) would 'sound extremely odd' but suggests that the theory could not mark it ungrammatical because that would involve 'counting', so she puts the unacceptability down to the parser. The current theory rules it out as a straightforward WIRE effect.

What I've done so far

- We saw that standard theory has a number of problems (labelling, copies vs repetitions, unmotivated theory of cyclicity, disconnect between cyclicity and island locality)

What I've done so far

- We saw that standard theory has a number of problems (labelling, copies vs repetitions, unmotivated theory of cyclicity, disconnect between cyclicity and island locality)
- I proposed new system built on the notion of 'part': Mereological Syntax

What I've done so far

- We saw that standard theory has a number of problems (labelling, copies vs repetitions, unmotivated theory of cyclicity, disconnect between cyclicity and island locality)
- I proposed new system built on the notion of 'part': Mereological Syntax
- This avoids certain theoretical problems (labelling and copy repetition issues) and has a new theory of Locality (Angular Locality) which *makes cyclicity part of structure*; this motivates why syntax is local where it is

What I've done so far

- We saw that standard theory has a number of problems (labelling, copies vs repetitions, unmotivated theory of cyclicity, disconnect between cyclicity and island locality)
- I proposed new system built on the notion of 'part': Mereological Syntax
- This avoids certain theoretical problems (labelling and copy repetition issues) and has a new theory of Locality (Angular Locality) which *makes cyclicity part of structure*; this motivates why syntax is local where it is
- The system derives as a theorem the requirement that dependencies through local domains must go through their edges,

What I've done so far

- We saw that standard theory has a number of problems (labelling, copies vs repetitions, unmotivated theory of cyclicity, disconnect between cyclicity and island locality)
- I proposed new system built on the notion of 'part': Mereological Syntax
- This avoids certain theoretical problems (labelling and copy repetition issues) and has a new theory of Locality (Angular Locality) which *makes cyclicity part of structure*; this motivates why syntax is local where it is
- The system derives as a theorem the requirement that dependencies through local domains must go through their edges,
- and Dimensionality places a limit on the number of 2-parts a category can have (unlike in the Merge system). These two factors motivate why syntax is not local where it isn't.

What I've done so far

- Empirically the new theory does what the old one did, but further cuts the empirical facts in just the right way in the domain of wh-islands, predicting an effect (the WIRE effect) that does not follow from the standard system.

What I've done so far

- Empirically the new theory does what the old one did, but further cuts the empirical facts in just the right way in the domain of wh-islands, predicting an effect (the WIRE effect) that does not follow from the standard system.
- I argue in Adger (to appear) that bringing cyclicity and some island locality together again is also the right move for Freezing Effects, Subject Islands, Complex Noun Phrase Islands, Relative Clause Islands, and at least some Clausal Adjunct Islands.

What I've done so far

- Empirically the new theory does what the old one did, but further cuts the empirical facts in just the right way in the domain of wh-islands, predicting an effect (the WIRE effect) that does not follow from the standard system.
- I argue in Adger (to appear) that bringing cyclicity and some island locality together again is also the right move for Freezing Effects, Subject Islands, Complex Noun Phrase Islands, Relative Clause Islands, and at least some Clausal Adjunct Islands.
- The theory unifies these via Angular Locality in quite a different way to Subjacency (or to other 'geometrical' approaches like Connectedness)

Adger, David. to appear. *Mereological syntax: phrase structure, cyclicity, and islands*. Cambridge MA: MIT Press.

Cinque, Guglielmo. 1990. *Types of $\overline{A}$-dependencies*. Cambridge, MA: MIT Press.

Engdahl, Elisabet. 1986. *Constituent Questions*. Dordrecht: D. Reidel Publishing Company.

Grewendorf, Günther. 2001. Multiple wh-fronting. *Linguistic Inquiry* 32:87–122.

Maling, Joan. 1978. An asymmetry with respect to *wh* islands. *Linguistic Inquiry* 9:75–89.

Nyvad, Anne Mette, Christensen, Ken Ramshøj, and Vikner, Sten. 2017. CP-recursion in Danish: A cP/CP-analysis. *The Linguistic Review* 34:449–477.

Pesetsky, David. 1987. Wh-in-situ: Movement and unselective binding. In Eric J. Reuland and Alice G. B. ter Meulen, eds., *The Representation of (In)definiteness*, 98–129, Cambridge, Massachusetts: MIT Press.

Rizzi, Luigi. 1982. *Issues in Italian Syntax*. Dordrecht: Foris.

Rudin, Catherine. 1988. On multiple questions and multiple wh-fronting. *Natural Language and Linguistic Theory* 6:445–501.

Multiple Wh-Extraction

Bulgarian and Romanian (though not many other multiple wh-movement languages) allow multiple extraction out of a locality domain:

Multiple Wh-Extraction

Bulgarian and Romanian (though not many other multiple wh-movement languages) allow multiple extraction out of a locality domain:

(55) Koj kŭde misliš cě ⟨koj⟩ e otišŭ ⟨kŭde⟩
who where think-2.S that ⟨who⟩ has gone ⟨where⟩
'Who do you think went where?'

Multiple Wh-Extraction

Bulgarian and Romanian (though not many other multiple wh-movement languages) allow multiple extraction out of a locality domain:

(55) Koj kŭde misliš cě ⟨koj⟩ e otišŭ ⟨kŭde⟩
who where think-2.S that ⟨who⟩ has gone ⟨where⟩
'Who do you think went where?'

And violate wh-islands

Multiple Wh-Extraction

Bulgarian and Romanian (though not many other multiple wh-movement languages) allow multiple extraction out of a locality domain:

(55) Koj kŭde misliš cě ⟨koj⟩ e otišŭ ⟨kŭde⟩
who where think-2.S that ⟨who⟩ has gone ⟨where⟩
'Who do you think went where?'

And violate wh-islands

(56) Vidjah edna kniga, kojato se čudja koj znae koj prodava
saw.1S a book which wonder.1S who know.3S who sells
⟨kojato⟩
⟨which⟩
'I saw a book which I wonder who knows who sells.' Bulgarian

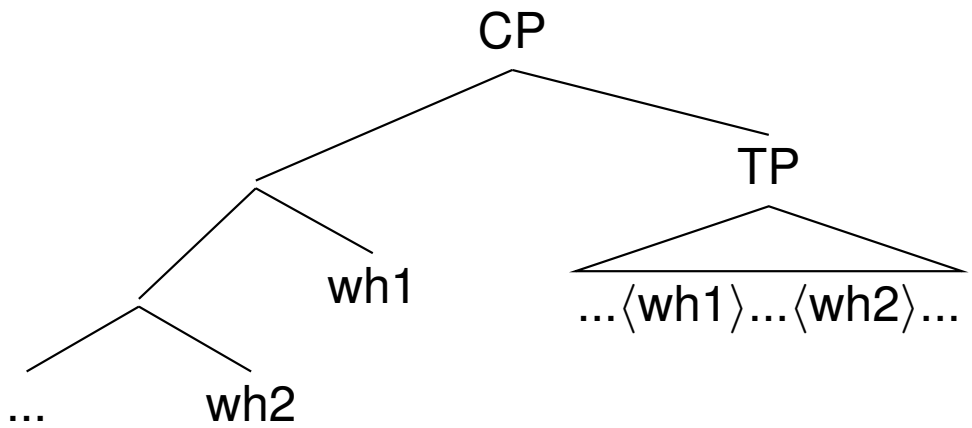
Classical Cluster Analysis

Rudin (1988) and others since have proposed that the multiple wh-expressions in languages of the Bulgarian/Romanian type move into a ‘cluster’:

Classical Cluster Analysis

Rudin (1988) and others since have proposed that the multiple wh-expressions in languages of the Bulgarian/Romanian type move into a 'cluster':

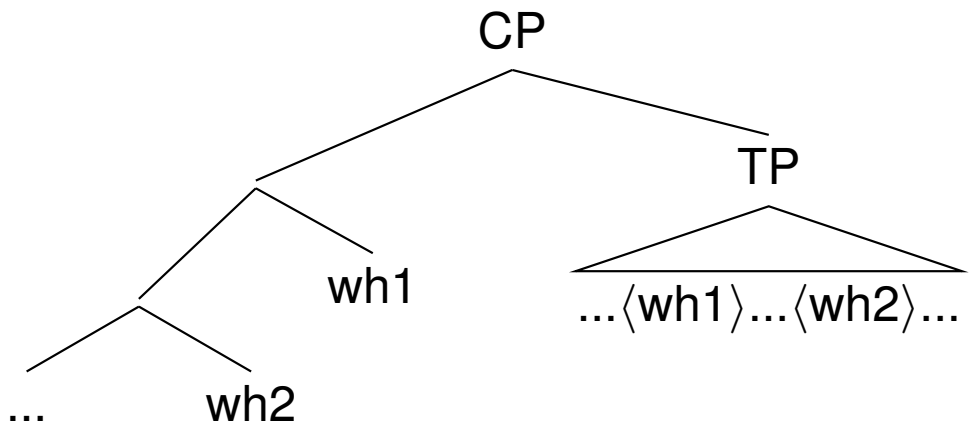
(57)



Classical Cluster Analysis

Rudin (1988) and others since have proposed that the multiple wh-expressions in languages of the Bulgarian/Romanian type move into a 'cluster':

(57)



In current terms this requires a parallel (or sidewards) Merge analysis, where wh1 Merges with wh2 even though wh2 is Merged with some clausal projection already.

Parallel Merge Impossible

Angular Locality

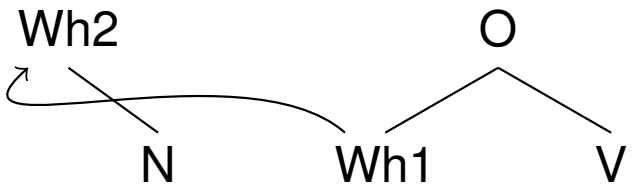
(58) if γ is a part, then it can subjoin to β only if there is an α s.t. γ is an n-part of α and α is a 1-part of β

Parallel Merge Impossible

Angular Locality

(58) if γ is a part, then it can subjoin to β only if there is an α s.t. γ is an n-part of α and α is a 1-part of β

(59)

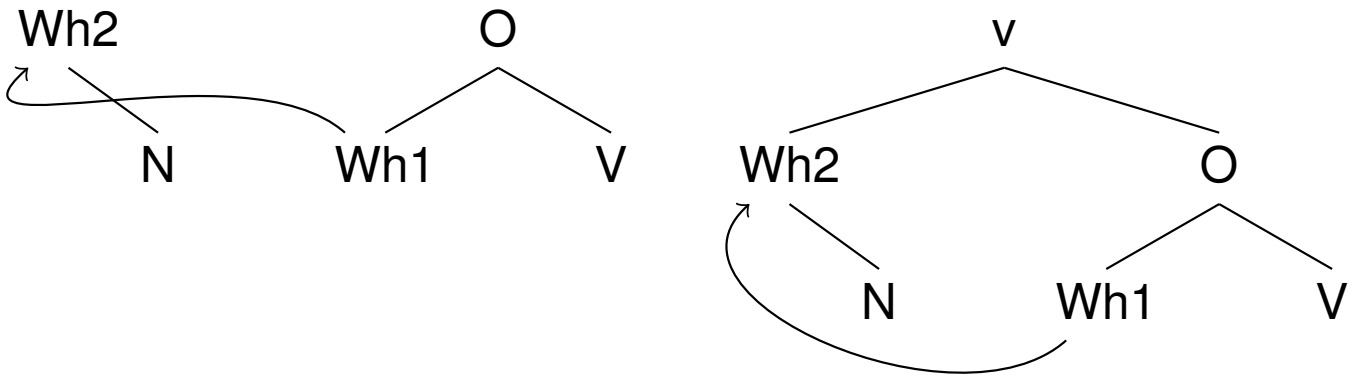


Parallel Merge Impossible

Angular Locality

(58) if γ is a part, then it can subjoin to β only if there is an α s.t. γ is an n-part of α and α is a 1-part of β

(59)

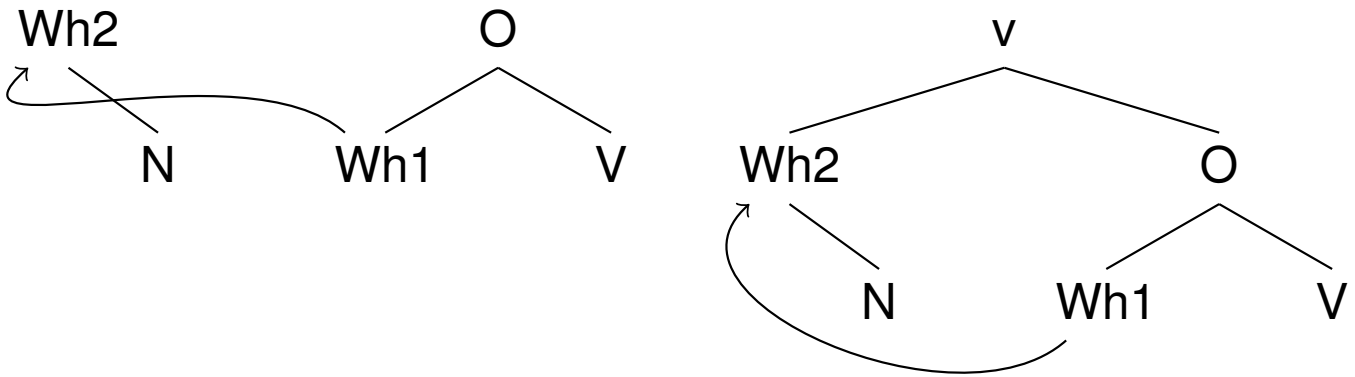


Parallel Merge Impossible

Angular Locality

(58) if γ is a part, then it can subjoin to β only if there is an α s.t. γ is an n-part of α and α is a 1-part of β

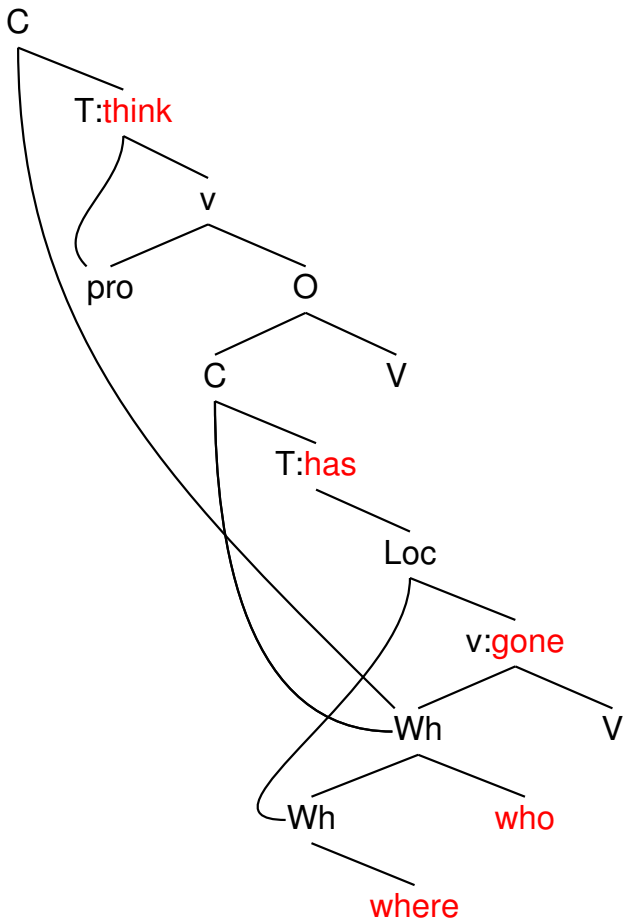
(59)



'Parallel' or 'Sidewards' Subjoin is impossible

A ‘declustering’ analysis

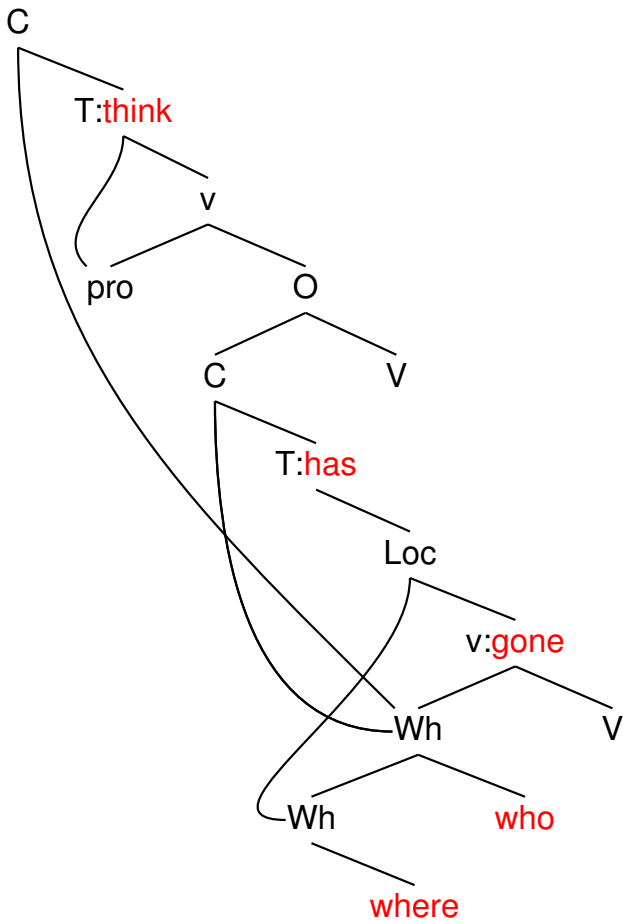
cf. Grewendorf (2001)



- (60) Koj kŭde misliš cě ⟨koj⟩ e
  who where think-2.S that ⟨who⟩ has
  otišŭ ⟨kŭde⟩
  gone ⟨where⟩
  ‘Who do you think went where?’

A ‘declustering’ analysis

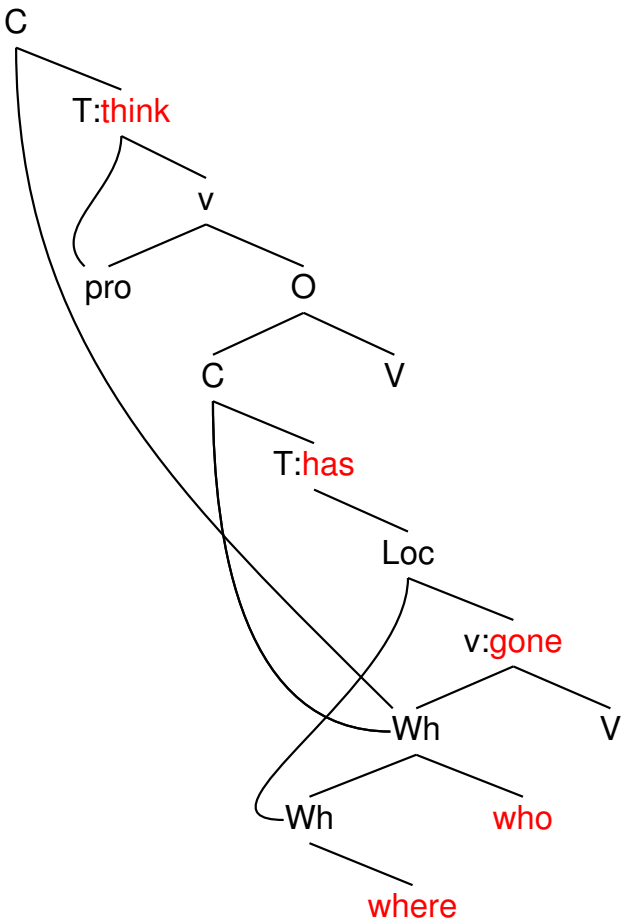
cf. Grewendorf (2001)



- (60) Koj kŭde misliš cě ⟨koj⟩ e
  who where think-2.S that ⟨who⟩ has
  otišŭ ⟨kŭde⟩
  gone ⟨where⟩
  ‘Who do you think went where?’
- Subjoin P to Wh (*where*) and Subjoin N to Wh (*who*)

A ‘declustering’ analysis

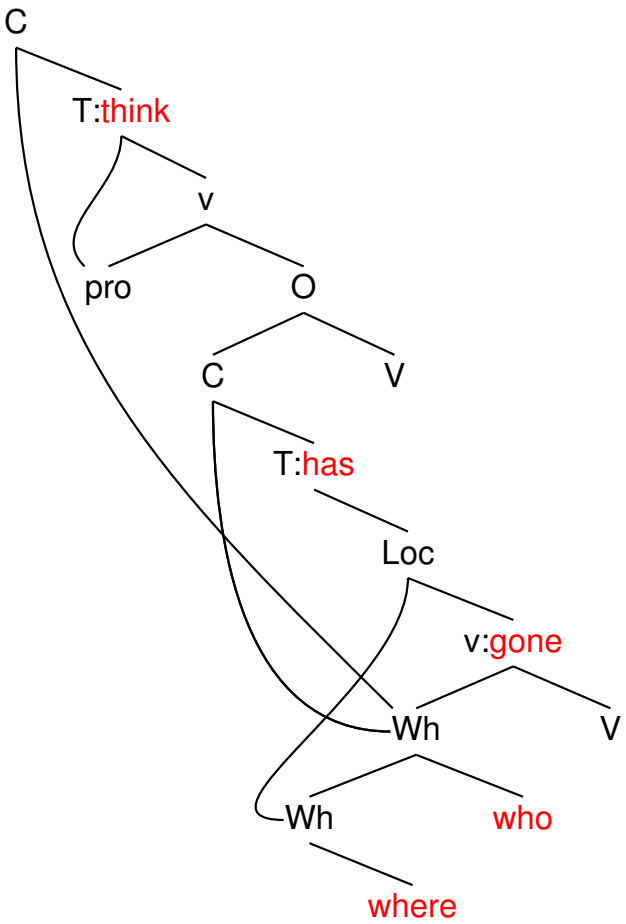
cf. Grewendorf (2001)



- (60) Koj kŭde misliš cě ⟨koj⟩ e
  who where think-2.S that ⟨who⟩ has
  otišŭ ⟨kŭde⟩
  gone ⟨where⟩
  ‘Who do you think went where?’
- Subjoin P to Wh (*where*) and Subjoin N to Wh (*who*)
- Subjoin *where* to *who* as a 2-part

A ‘declustering’ analysis

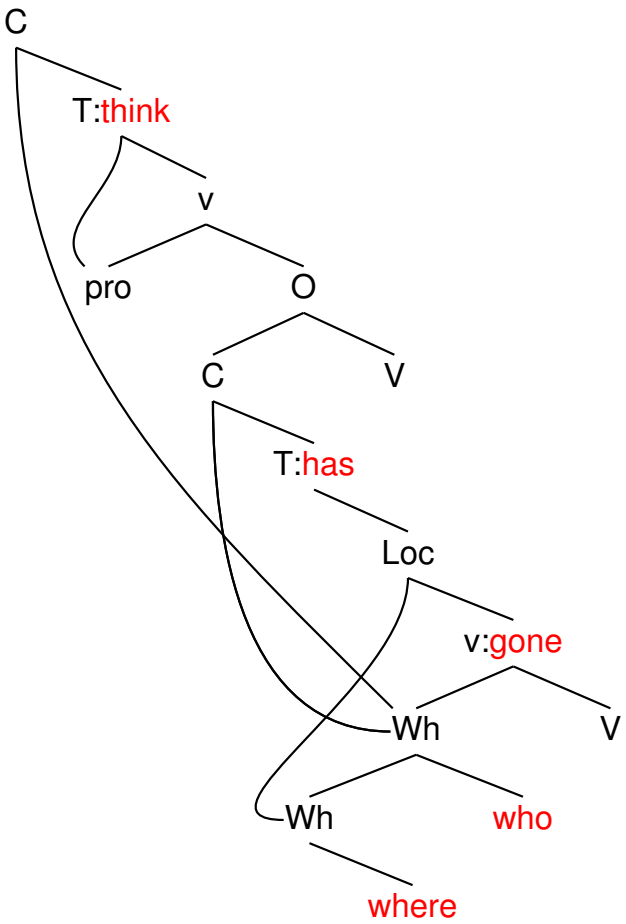
cf. Grewendorf (2001)



- (60) Koj kŭde misliš cě ⟨koj⟩ e
  who where think-2.S that ⟨who⟩ has
  otišŭ ⟨kŭde⟩
  gone ⟨where⟩
  ‘Who do you think went where?’
- Subjoin P to Wh (*where*) and Subjoin N to Wh (*who*)
- Subjoin *where* to *who* as a 2-part
- Subjoin who to v so it is interpreted as subject

A ‘declustering’ analysis

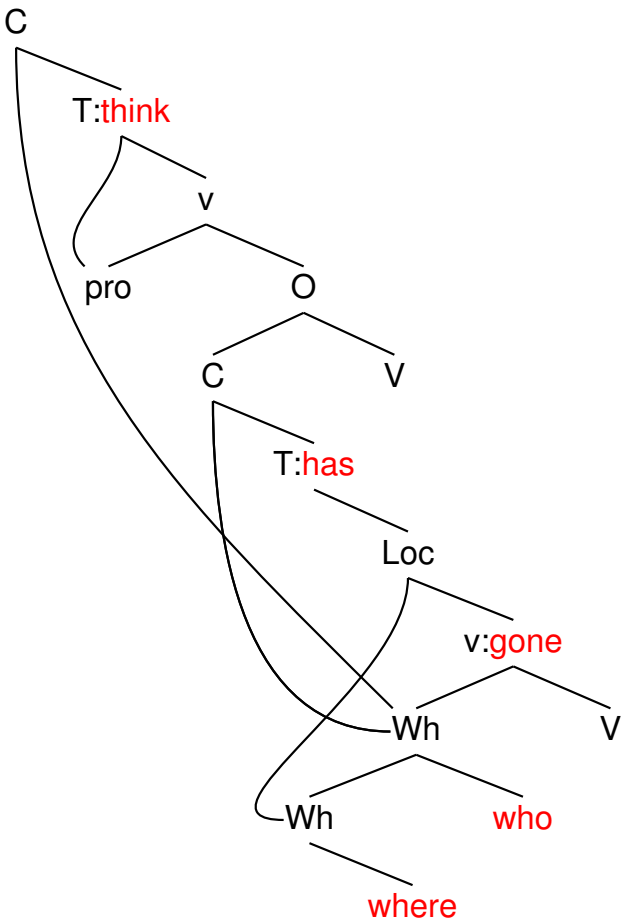
cf. Grewendorf (2001)



- (60) Koj kŭde misliš cě ⟨koj⟩ e
  who where think-2.S that ⟨who⟩ has
  otišŭ ⟨kŭde⟩
  gone ⟨where⟩
  ‘Who do you think went where?’
- Subjoin P to Wh (*where*) and Subjoin N to Wh (*who*)
- Subjoin *where* to *who* as a 2-part
- Subjoin who to v so it is interpreted as subject
- Internally Subjoin where to Loc so it is interpreted as a location of the event

A ‘declustering’ analysis

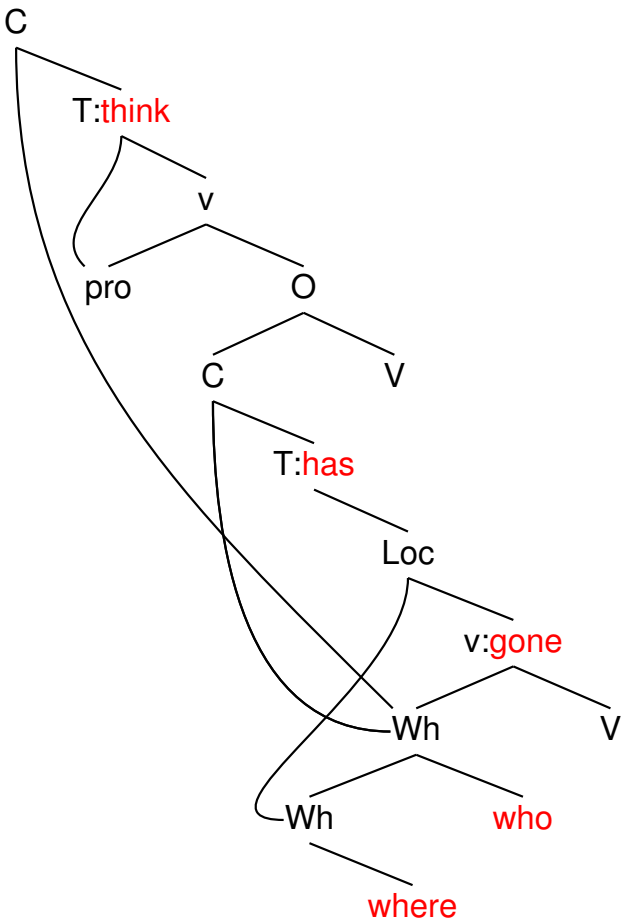
cf. Grewendorf (2001)



- (60) Koj kŭde misliš cě ⟨koj⟩ e
  who where think-2.S that ⟨who⟩ has
  otišŭ ⟨kŭde⟩
  gone ⟨where⟩
  ‘Who do you think went where?’
- Subjoin P to Wh (*where*) and Subjoin N to Wh (*who*)
- Subjoin *where* to *who* as a 2-part
- Subjoin who to v so it is interpreted as subject
- Internally Subjoin where to Loc so it is interpreted as a location of the event
- keep going then ...

A ‘declustering’ analysis

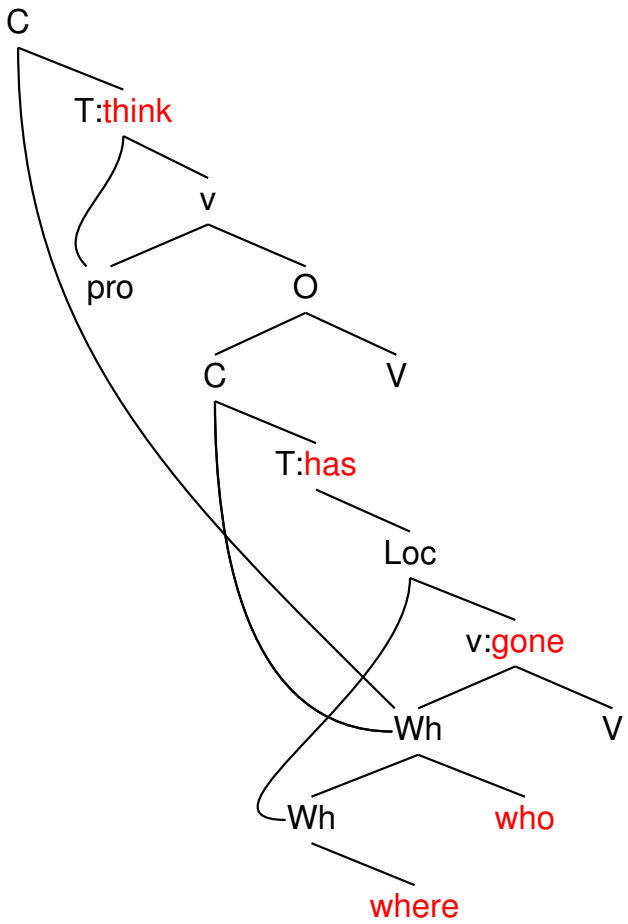
cf. Grewendorf (2001)



- (60) Koj kŭde misliš cě ⟨koj⟩ e
  who where think-2.S that ⟨who⟩ has
  otišŭ ⟨kŭde⟩
  gone ⟨where⟩
  ‘Who do you think went where?’
- Subjoin P to Wh (*where*) and Subjoin N to Wh (*who*)
- Subjoin *where* to *who* as a 2-part
- Subjoin who to v so it is interpreted as subject
- Internally Subjoin where to Loc so it is interpreted as a location of the event
- keep going then ...
- Subjoin the cluster to intermediate C

A ‘declustering’ analysis

cf. Grewendorf (2001)



- (60) Koj kŭde misliš cě ⟨koj⟩ e
  who where think-2.S that ⟨who⟩ has
  otišŭ ⟨kŭde⟩
  gone ⟨where⟩
  ‘Who do you think went where?’
- Subjoin P to Wh (*where*) and Subjoin N to Wh (*who*)
- Subjoin *where* to *who* as a 2-part
- Subjoin who to v so it is interpreted as subject
- Internally Subjoin where to Loc so it is interpreted as a location of the event
- keep going then ...
- Subjoin the cluster to intermediate C
- now because Angular Locality is satisfied, Subjoin the cluster to matrix C

A (correct) prediction

WIRE effect should vanish: subjunction of the cluster to a higher C will make any wh-expression which is a 2-part of that cluster also a 2-part of whatever the higher C is a 2-part of, and hence, by Angular Locality, able to subjoin outside of that higher C.

(61) Čoveka, kojto ne znaeš [$_{CP}$ kakvo kazvat [$_{CP}$ če e kupil]]
man who NEG know.2S what say.3P that has bought
'The man who you don't know what they say that he bought ...'