



Ling 566
Oct 10, 2023
Valence, Agreement

Announcements

- No Canvas answers from staff evenings/ weekends (but feel free to discuss amongst yourselves!)
- HW1 answer key is available!
- HW2 Ch 5, problem 3 is worth getting an early start on

When poll is active, respond at pollev.com/emb

Text **EMB** to **22333** once to join



W Have you joined a study group?

Yes

No

It's complicated

Total Results: 0

Powered by  **Poll Everywhere**

Start the presentation to see live content. For screen share software, share the entire screen. Get help at pollev.com/app

Overview

- Review: pizza, feature structures, well-formed trees, HFP
- A problem with the Chapter 3 grammar
- Generalize COMPS and SPR
- The Valence Principle
- Agreement
- The SHAC
- Reading Questions

Pizza review

- Unification is an operation for combining constraints from different sources.
- What are those sources in the pizza example?
- Why do we need to combine information from different sources in our grammars?

Reminder: Where We Are

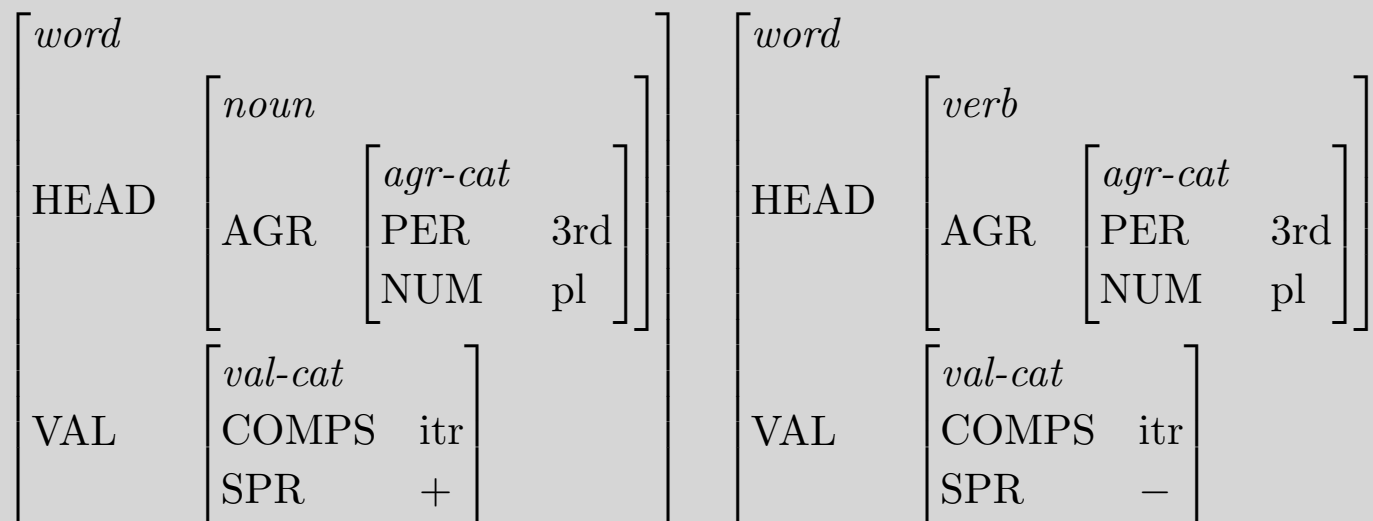
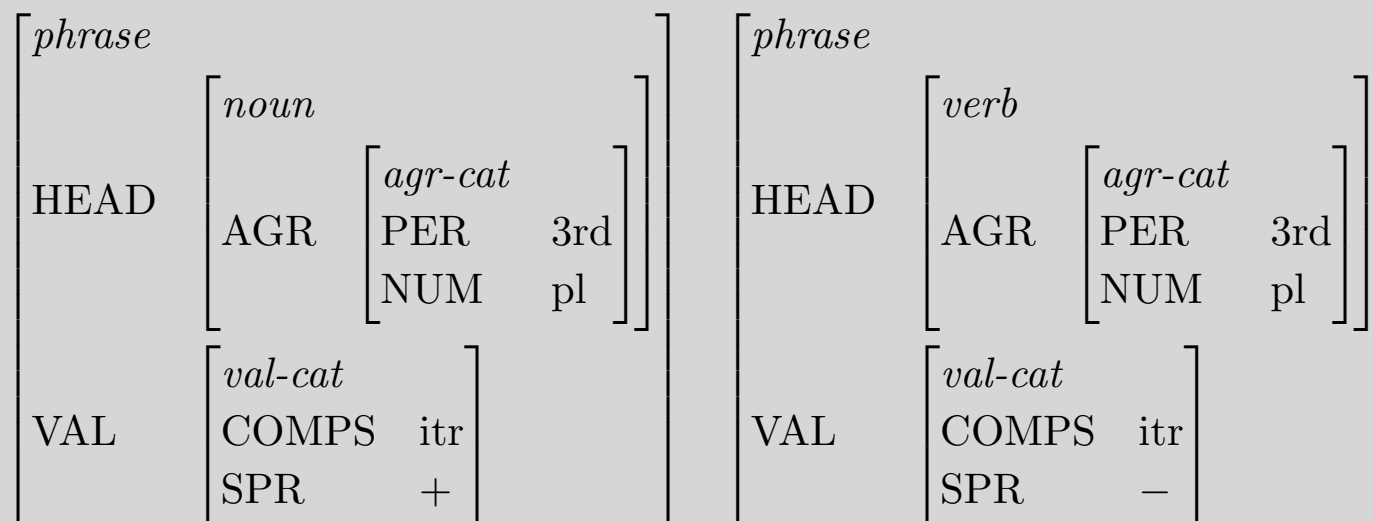
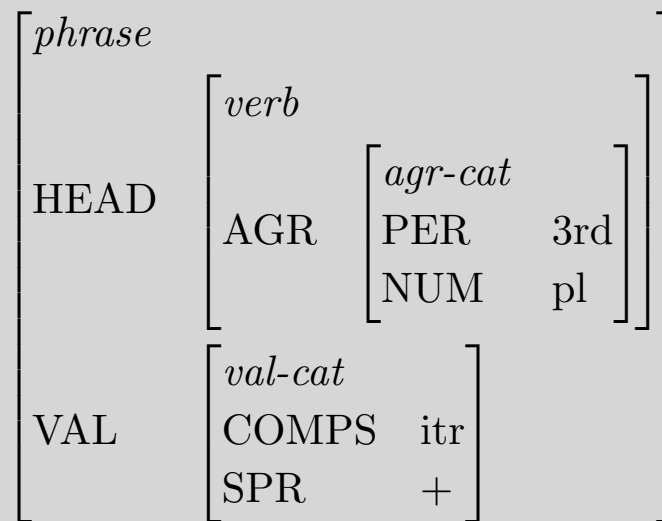
- Attempting to model English with CFG led to problems with the granularity of categories, e.g.
 - Need to distinguish various subtypes of verbs
 - Need to identify properties common to all verbs
- So we broke categories down into feature structures and began constructing a hierarchy of types of feature structures.
- This allows us to schematize rules and state cross-categorial generalizations, while still making fine distinctions.

A Tree is Well-Formed if ...

- It and each subtree are licensed by a grammar rule or lexical entry
- All general principles (like the HFP) are satisfied.
- NB: Trees are part of our model of the language, so all their features have values (even though we will often be lazy and leave out the values irrelevant to our current point).

The Head Feature Principle

- Intuitive idea: Key properties of phrases are shared with their heads
- The HFP: In any headed phrase, the HEAD value of the mother and the head daughter must be identical.
- Sometimes described in terms of properties “percolating up” or “filtering down”, but this is just metaphorical talk



they

swim

But it's still not quite right...

- There's still too much redundancy in the rules.
- The rules and features encode the same information in different ways.

Head-Complement Rule 1:

$$\left[\begin{array}{l} \textit{phrase} \\ \text{VAL} \end{array} \left[\begin{array}{ll} \text{COMPS} & \textit{itr} \\ \text{SPR} & - \end{array} \right] \right] \rightarrow \mathbf{H} \left[\begin{array}{l} \textit{word} \\ \text{VAL} \end{array} \left[\begin{array}{ll} \text{COMPS} & \textit{itr} \\ \text{SPR} & - \end{array} \right] \right]$$

Head Complement Rule 2:

$$\left[\begin{array}{l} \textit{phrase} \\ \text{VAL} \end{array} \left[\begin{array}{ll} \text{COMPS} & \textit{itr} \\ \text{SPR} & - \end{array} \right] \right] \rightarrow \mathbf{H} \left[\begin{array}{l} \textit{word} \\ \text{VAL} \end{array} \left[\begin{array}{ll} \text{COMPS} & \textit{str} \\ \text{SPR} & - \end{array} \right] \right] \text{ NP}$$

Head Complement Rule 3:

$$\left[\begin{array}{l} \textit{phrase} \\ \text{VAL} \end{array} \left[\begin{array}{ll} \text{COMPS} & \textit{itr} \\ \text{SPR} & - \end{array} \right] \right] \rightarrow \mathbf{H} \left[\begin{array}{l} \textit{word} \\ \text{VAL} \end{array} \left[\begin{array}{ll} \text{COMPS} & \textit{dtr} \\ \text{SPR} & - \end{array} \right] \right] \text{ NP NP}$$

Solution:

More Elaborate Valence Feature Values

- The rules just say that heads combine with whatever their lexical entries say they can (or must) combine with.
- The information about what a word can or must combine with is encoded in list-valued valence features.
 - The elements of the lists are themselves feature structures
 - The elements are “cancelled” off the lists once heads combine with their complements and specifiers.

Complements

Head-Complement Rule:

$$\left[\begin{array}{l} \textit{phrase} \\ \text{VAL} \left[\text{COMPS} \langle \rangle \right] \end{array} \right] \rightarrow \mathbf{H} \left[\begin{array}{l} \textit{word} \\ \text{VAL} \left[\text{COMPS} \langle \boxed{1}, \dots, \boxed{n} \rangle \right] \end{array} \right] \boxed{1}, \dots, \boxed{n}$$

- This allows for arbitrary numbers of complements, but only applies when there is at least one.
 - Heads in English probably never have more than 3 or 4 complements
 - This doesn't apply where Head-Complement Rule 1 would. (Why?)
- This covers lots of cases not covered by the old Head-Complement Rules 1-3. (Examples?)

Question: How would the grammar change if English had **postpositions**, instead of **prepositions**?

Head-Complement Rule

$$\left[\begin{array}{l} \textit{phrase} \\ \text{VAL} \left[\text{COMPS} \langle \rangle \right] \end{array} \right] \rightarrow \mathbf{H} \left[\begin{array}{l} \textit{word} \\ \text{HEAD} \textit{verb} \mid \textit{adj} \mid \textit{noun} \\ \text{VAL} \left[\text{COMPS} \langle \boxed{1}, \dots, \boxed{n} \rangle \right] \end{array} \right] \boxed{1}, \dots, \boxed{n}$$

PP Rule

$$\left[\begin{array}{l} \textit{phrase} \\ \text{VAL} \left[\text{COMPS} \langle \rangle \right] \end{array} \right] \rightarrow \boxed{1}, \dots, \boxed{n} \mathbf{H} \left[\begin{array}{l} \textit{word} \\ \text{HEAD} \textit{prep} \\ \text{VAL} \left[\text{COMPS} \langle \boxed{1}, \dots, \boxed{n} \rangle \right] \end{array} \right]$$

Specifiers

Head-Specifier Rule (Version I)

$$\left[\begin{array}{l} \textit{phrase} \\ \text{VAL} \left[\begin{array}{l} \text{COMPS} \langle \rangle \\ \text{SPR} \langle \rangle \end{array} \right] \end{array} \right] \rightarrow \boxed{2} \mathbf{H} \left[\begin{array}{l} \text{VAL} \left[\begin{array}{l} \text{COMPS} \langle \rangle \\ \text{SPR} \langle \boxed{2} \rangle \end{array} \right] \end{array} \right]$$

- Combines the rules expanding S and NP.
- In principle also generalizes to other categories.
- Question: Why is SPR list-valued?

🌐 When poll is active, respond at pollev.com/emb

📱 Text **EMB** to **22333** once to join



W SPR as a list

Love the symmetry
with COMPS

Lists that max out
at 1 item are weird

I preferred +/-

None of the above

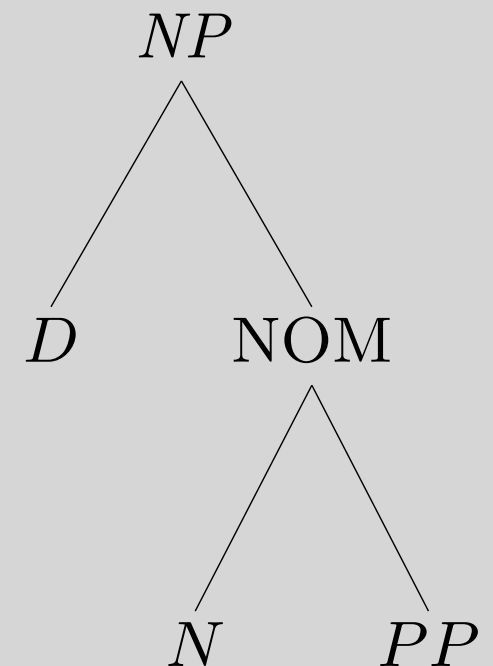
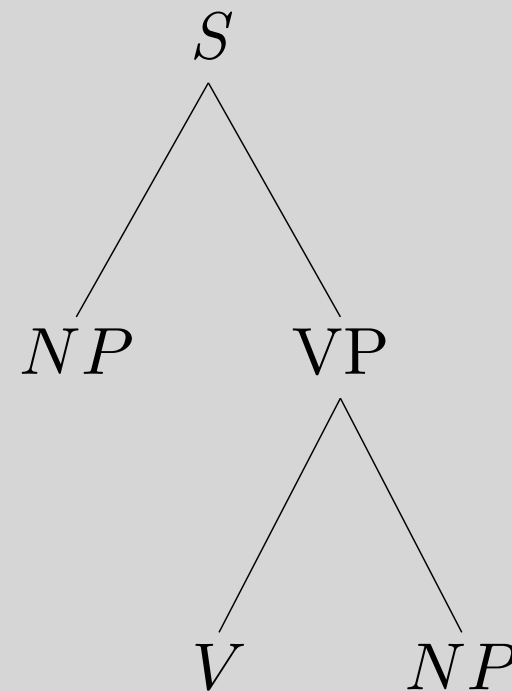
Total Results: 0

Powered by  **Poll Everywhere**

Start the presentation to see live content. For screen share software, share the entire screen. Get help at pollev.com/app

Question:

Why are these right-branching? That is, what formal property of our grammar forces the COMPS to be lower in the tree than the SPR?



Another Question...

What determines the VAL value of phrasal nodes?

ANSWER: The Valence Principle

Unless the rule says otherwise, the mother's values for the VAL features (SPR and COMPS) are identical to those of the head daughter.

More on the Valence Principle

- Intuitively, the VAL features list the contextual requirements that haven't yet been found.
- This way of thinking about it (like talk of “cancellation”) is bottom-up and procedural.
- But formally, the Valence Principle (like the rest of our grammar) is just a well-formedness constraint on trees, without inherent directionality.

So far, we have:

- Replaced atomic-valued VAL features with list-valued ones.
- Generalized Head-Complement and Head-Specifier rules, to say that heads combine with whatever their lexical entries say they should combine with.
- Introduced the Valence Principle to carry up what's not “canceled”.

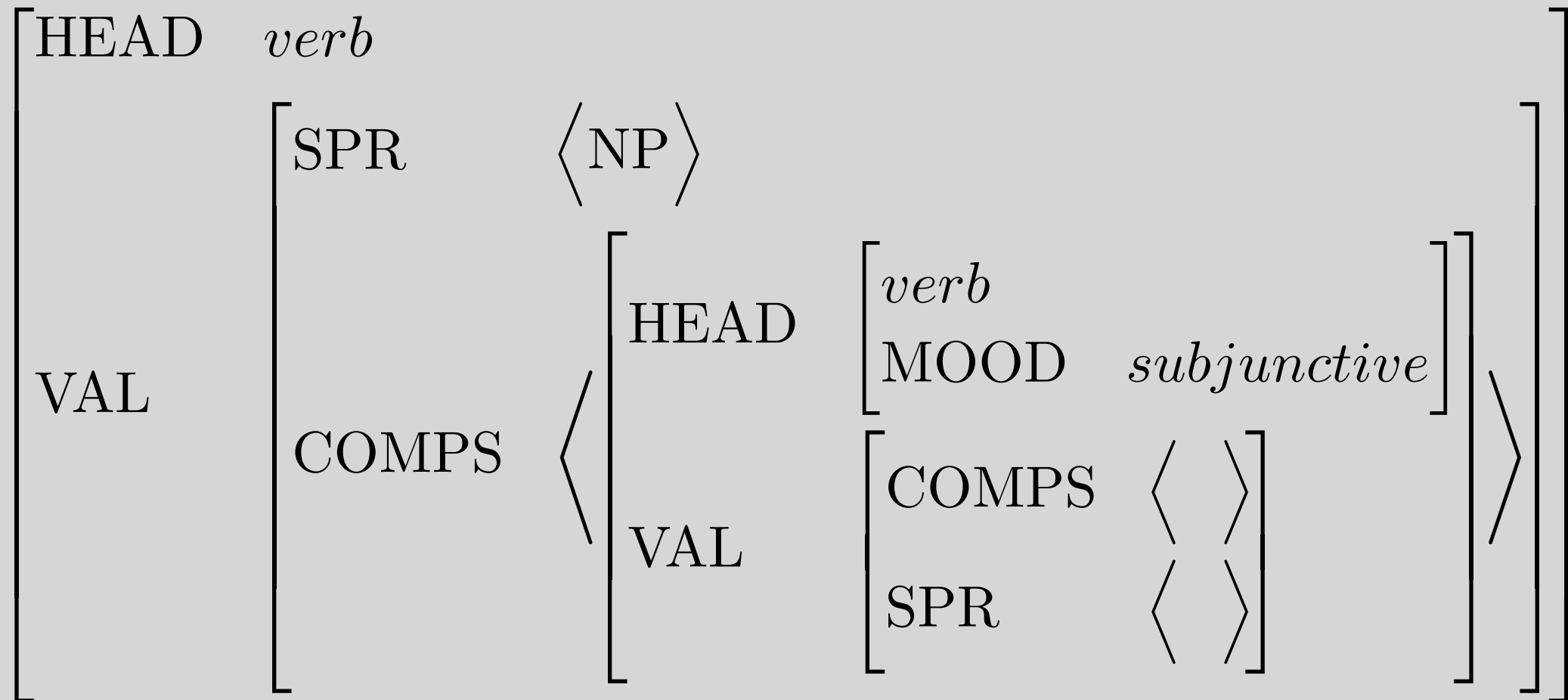
The Parallelism between S and NP

- Motivation:
 - pairs like *Chris lectured about syntax* and *Chris's lecture about syntax*.
 - both S and NP exhibit agreement
 - The bird sings/*sing* vs. *The birds sing/*sings*
 - this/*these bird* vs. *these/*this birds*
- So we treat NP as the saturated category of type *noun* and S as the saturated category of type *verb*.

Question: Is there any other reason to treat V as the head of S?

- In mainstream American English, sentences must have verbs. (How about other varieties of English or other languages?)
- Verbs taking S complements can influence the form of the verb in the complement:
*I insist/*recall (that) you be here on time.*
- Making V the head of S helps us state such restrictions formally

A possible formalization of the restriction on *insist*



Note that this requires that the verb be the head of the complement. We don't have access to the features of the other constituents of the complement.

An Overlooked Topic: Complements *vs.* Modifiers

- Intuitive idea: Complements introduce essential participants in the situation denoted; modifiers refine the description.
- Generally accepted distinction, but disputes over individual cases.
- Linguists rely on heuristics to decide how to analyze questionable cases (usually PPs).

Heuristics for Complements vs. Modifiers

- Obligatory PPs are usually complements.
- Temporal & locative PPs are usually modifiers.
- An entailment test: If **X Ved (NP) PP** does not entail **X did something PP**, then the PP is a complement.

Examples

- *Pat relied on Chris* **does not entail** *Pat did something on Chris*
- *Pat put nuts in a cup* **does not entail** *Pat did something in a cup*
- *Pat slept until noon* **does entail** *Pat did something until noon*
- *Pat ate lunch at Bytes* **does entail** *Pat did something at Bytes*

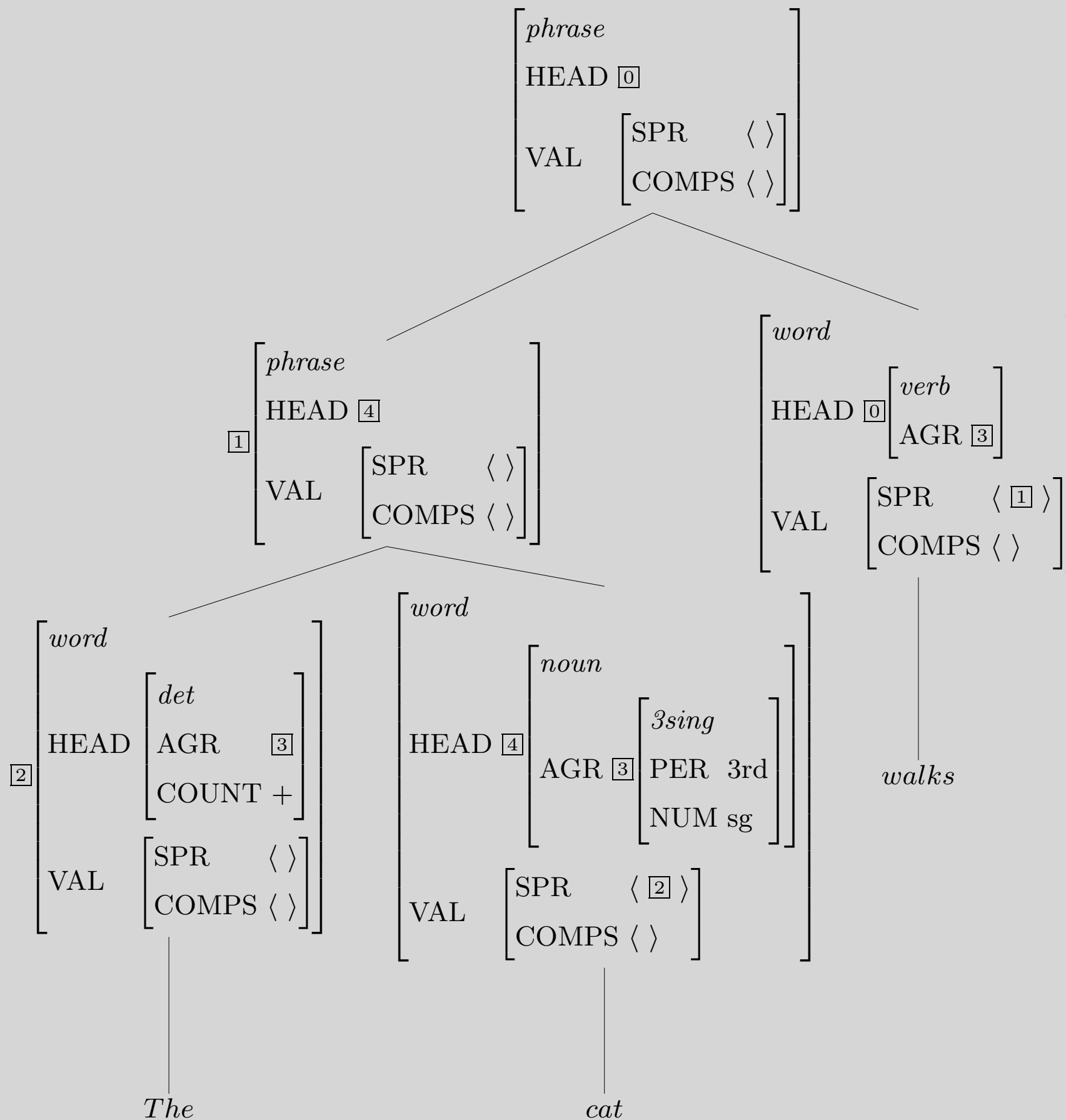
Agreement

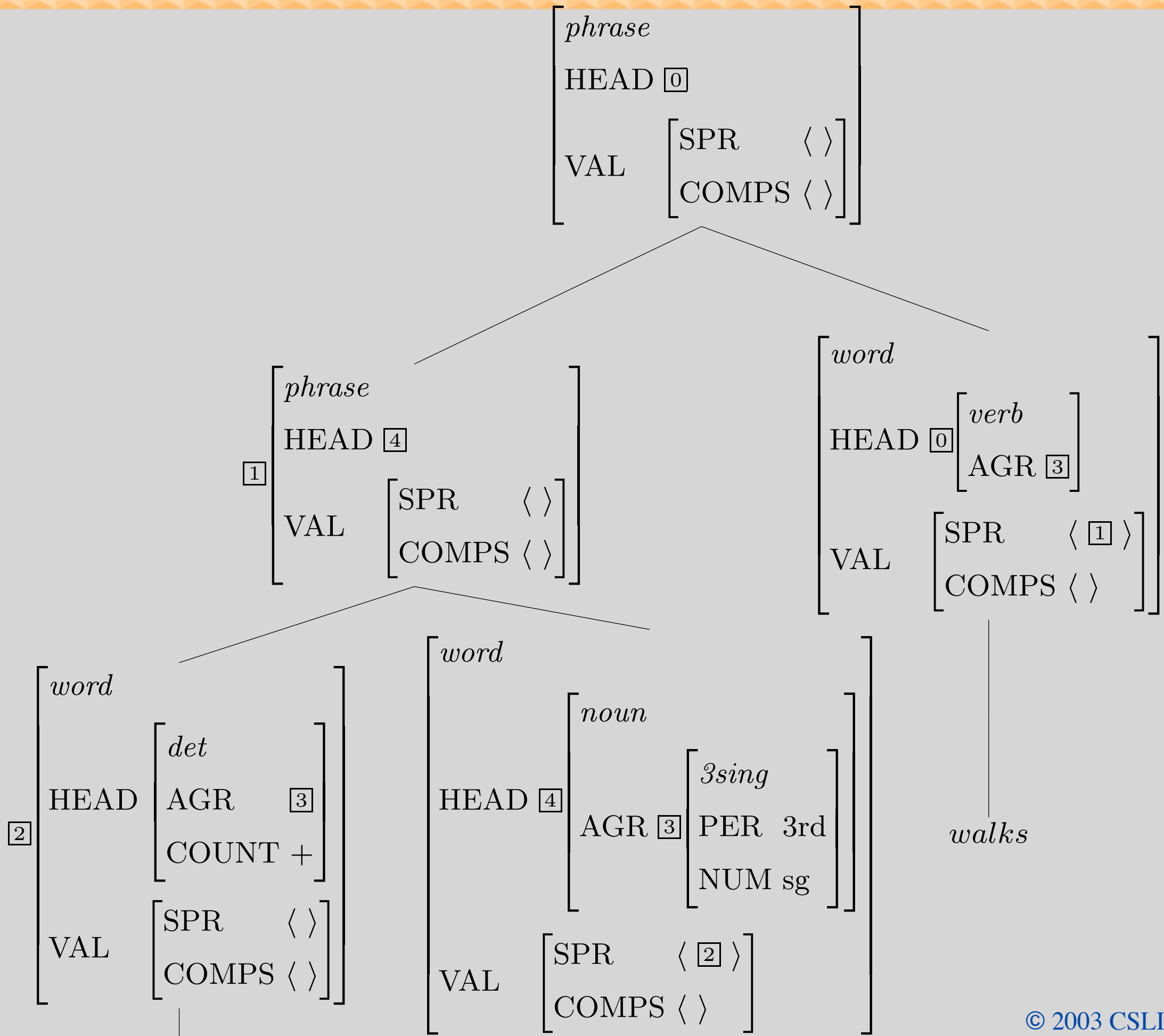
- Two kinds so far (namely?)
- Both initially handled via stipulation in the Head-Specifier Rule
- But if we want to use this rule for categories that don't have the AGR feature (such as PPs and APs, in English), we can't build it into the rule.

The Specifier-Head Agreement Constraint (SHAC)

Verbs and nouns must be specified as:

$$\left[\begin{array}{l} \text{HEAD} \left[\text{AGR} \quad \boxed{1} \right] \\ \text{VAL} \left[\text{SPR} \left\langle \left[\text{AGR} \quad \boxed{1} \right] \right\rangle \right] \end{array} \right]$$





The Count/Mass Distinction

- Partially semantically motivated
 - mass terms tend to refer to undifferentiated substances (*air, butter, courtesy, information*)
 - count nouns tend to refer to individuatable entities (*bird, cookie, insult, fact*)
- But there are exceptions:
 - *succotash* (mass) denotes a mix of corn & lima beans, so it's not undifferentiated.
 - *furniture, footwear, cutlery*, etc. refer to individuatable artifacts with mass terms
 - *cabbage* can be either count or mass, but many speakers get *lettuce* only as mass.
 - borderline case: *data*

Our Formalization of the Count/Mass Distinction

- Determiners are:
 - [COUNT –] (*much* and, in some dialects, *less*),
 - [COUNT +] (*a, six, many*, etc.), or
 - lexically underspecified (*the, all, some, no*, etc.)
- Nouns select appropriate determiners
 - “count nouns” say SPR <[COUNT +]>
 - “mass nouns” say SPR <[COUNT –]>
- Nouns themselves aren’t marked for the feature COUNT
- So the SHAC plays no role in count/mass marking.

Overview

- Review: pizza, feature structures, well-formed trees, HFP
- A problem with the Chapter 3 grammar
- Generalize COMPS and SPR
- The Valence Principle
- Agreement
- The SHAC
- Reading Questions

RQs: Arguments vs. adjuncts

- In a previous syntax class I took, I learned about specifiers, adjuncts and complements, and how to differentiate between them using models from the X-bar theory. As a result, I'm curious about how these terms are defined in this particular class. Could you provide definitions for these terms in this context, or perhaps share some examples?

RQs: Arguments vs. adjuncts

- How can we understand the syntactic structure of “put the flower in a vase”?
What is the difference between making [NP [the flowers]] and [PP [in a vase]] sister and making NP under PP?
- Should complements and modifiers be distinguished in the grammar (i.e., should we make them into features)?

RQs: Specifiers

- In (25c) there is a lexical entry for "the" that has both an empty SPR and COMPS field. Doesn't the fact that it is a determiner and therefore must proceed a noun mean it must have 'NP' in its COMPS field?

RQs: Modifiers

- The Valence Principle and Head-Modifier Rule in Section 4.5: If I understood it correctly, their purpose is to prevent modifiers from satisfying SPR or AGR requirements, but I'm not sure how exactly the Head-Modifier Rule in (30) contributes to that effect. Also, what is the reason for including [VAL [COMPS < >]] in the rule notation? Is it because head-modifier phrases are always higher than head-complement phrases?

RQs: Type hierarchy, constraints

- For exercise 4 on page 111, can I write the AGR value for are as [AGR non-1singl non-3sing]?
- For something like [SPR <[HEAD det]>] Why is it formatted to include the word "Head"?

RQs: Crosslinguistic

- We combined the two head specifier rules into one, which always puts the head after the specifier. This might work for English, but what about languages that aren't strictly head-initial or head-final? Or languages that put adjectives before the nouns they modify depending on the semantics of the sentence (e.g. Spanish: "pobre niño" 'poor (fig.) child' vs. "niño pobre" 'poor (lit.) child')?

RQs: Crosslinguistic

- The book gave the example of the general word order in English and Japanese as reflecting the location of HEAD. Is this true for other languages? If it's a universal feature in human languages, what causes it to happen?

RQs: *word vs. phrase*

- The text states that "we are in effect shifting to a perspective where phrasality has a much smaller role to play in syntax". If we now can use N and NP interchangeably for example, what is the merit in initially considering phrasality only to abandon it? Is there any merit left to considering phrasality, seeing as we can now have a more simplified system without it, or are we just looking at the remnants of an old abandoned system that has no use in modern theory?

RQs: Big picture

- Why do we want to reduce the number of rules in our grammar in the first place?
- Section 4.4 says that compared to Chapter 3, the grammar in Chapter 4 has no non-branching nodes. Why is it better not to have non-branching nodes?
- Why do we favor simpler trees over a simpler lexicon? Which is easier to do computationally?

RQs: Formalism

- Regarding the use of tags in features, can we arbitrarily choose numbers starting from 1 or letters starting from A? Or do they need to be arranged in order?
- Is there any way (other than an empty list) to specify that some feature has no value? Or in those cases, would you just exclude it entirely?

RQs: Formalism

- Something I noticed while reading this chapter was the bottom-up nature of this grammar. Head daughters pass their features onto their mothers in the new head rules, and we start to see examples where child nodes with certain requirements are able to combine into a parent node where those requirements are satisfied. If this is the case, why are rules still written like, "PARENT \rightarrow DAUGHTERS," instead of, "PARENT \leftarrow DAUGHTERS"? Is this an artifact of CFG, or is it for convenience (since other syntactic frameworks use this format)? Will it be important to the theory later in the book? And/or is it the convention for bottom-up frameworks to begin with?