Head-Driven Phrase Structure Grammar (HPSG)

Linguistics 614

With thanks to Detmar Meurers

Spring 2015
Head-Driven Phrase Structure Grammar (HPSG)

The main mechanism of HPSG for linguistic analysis lies in its feature architecture:

- Trees, bar-level distinctions, subcategorization, agreement, etc. are handled via feature structures.

Rough comparison to LFG:

- Similar: use of feature structures, highly lexical.
- Different: only one layer of syntax, functions are not primitives, stronger reliance on structure-sharing.
Preview: lexical entry for *letter*

(This entry more or less follows Carnie (2013) / Sag & Wasow (1999), which most of the rest of the slides don’t.)
Unpacking *letter*

- **SYN**: holds syntactic information (vs. **PHON** or **SEM**)
  - SYNSEM is also commonly used (e.g., these slides)
- **HEAD**: category & inflectional properties
- **SPR & COMPS**: subcategorization properties
  - Sometimes 3 lists (**SUBJ**, **SPR**, **COMPS**), sometimes 1 list (**SUBCAT**)
  - Note that detailed properties of selected items can be specified
An HPSG grammar, from a linguistic perspective, consists of

a) a **lexicon**: licensing basic words

b) **lexical rules**: licensing derived words

c) **immediate dominance (ID) schemata**: licensing constituent structure

d) **linear precedence (LP) statements**: constraining word order

e) a set of **grammatical principles**: expressing generalizations about linguistic objects
Lexical rules

e.g., plural rule (adapted from Carnie 2013):

\[
\begin{align*}
\text{noun} & \quad \left[ \begin{array}{c}
\text{PHON} \\
\text{ARG-ST} \langle [\text{COUNT} +] \rangle
\end{array} \right] \\
& \Rightarrow \\
\text{word} & \quad \left[ \begin{array}{c}
\text{PHON} \\
\text{SYN} | \text{HEAD} | \text{AGR} | \text{NUM} \\
F_{NPL}(1)
\end{array} \right]
\end{align*}
\]

Any feature not specified is unchanged
Grammatical principles allow one to generalize over entire classes of objects

- e.g., every complete sentence in English is tensed

These can be interpreted as constraints on the acceptable feature structures

- HPSG is constraint-based: grammars are sets of constraints which hold simultaneously
- This determines the collections of admissible linguistic structures
  - Constraints do not define an order of the derivation or generation of linguistic signs
HPSG grammars

Formal perspective

An HPSG grammar formally consists of

I. the **signature** as declaration of the domain, and
II. the **theory** constraining the domain.
The **signature**

- defines the ontology (‘declaration of what exists’):
  - which kind of objects are distinguished, and
  - which properties of which objects are modeled.

- consists of
  - the **type hierarchy** (or sort hierarchy) and
  - the **appropriateness conditions**, defining which type has which appropriate attributes (or features) with which appropriate values.
Signature: some examples

Signs

\[
\begin{cases}
\text{sign} \\
\text{PHON} \\
\text{SYNSEM}
\end{cases}
\begin{cases}
\text{list(phonstring)} \\
\text{synsem}
\end{cases}
\]

word phrase

Part of speech

\[
\begin{cases}
\text{adj} \\
\text{adv} \\
\text{det} \\
\text{noun} \\
\text{prep} \\
\text{verb}
\end{cases}
\]
What do the mathematical structures used as model for HPSG theories look like?

- The objects are modeled by typed feature structures, which can be notated as directed graphs.
- These models represent objects in the world → they are total with respect to the ontology declared in the signature.
  - Naturally, knowledge about objects in the world is partial.
- Formally speaking, the feature structures used as models are:
  - *totally well-typed*: Every type has every one of the attributes and their values which are appropriate for it.
  - *sort-resolved*: Every type is maximally specific.

*type* and *sort* as well as *attribute* and *feature* are used synonymously.
An example for a typed feature structure
A **theory** is a set of description language statements, often referred to as the constraints.

- The theory singles out a subset of the objects declared in the signature, namely those which are grammatical.
- A linguistic object is admissible with respect to a theory iff it satisfies each of the descriptions in the theory and so does each of its substructures.
How do we express a theory? Descriptions

- A **description language** describes sets of objects.
- The **AVM** (attribute value matrix) notation is used to compactly write down these descriptions.
- Descriptions consist of three building blocks:
  - **Type** descriptions single out all objects of a given type
    - Example: *word*
  - **Attribute-value pairs** describe objects that have a particular property.
    - The attribute must be appropriate for the particular type of object, and the value can be any kind of description.
    - Example: \[
    \text{SPOUSE} \ [\text{NAME} \ mary] \]
  - **Tags** (structure sharing) to specify **token identity**
    - Example: \[1\]
Complex descriptions are obtained through
- conjunction ($\land$)
- disjunction ($\lor$)
- negation ($\neg$)

In the AVM notation, conjunction is implicit.
An example AVM - The pronoun *she*

```
word
  PHON <she>
  local
    CAT
      HEAD
        SUBCAT 〈
          ppro
            INDEX 1
              PER
                NUM
                  GEND
                    fem
        RESTR {}
    context
      BACKGR
        psoa
          RELN
            INST 1
              female
```
Aspects of the signature

**Signs**

Aspects of the signature

\[
\begin{align*}
\text{sign} & \quad \text{list(phonstring)} \\
\text{PHON} & \quad \text{synsem} \\
\text{SYNSEM} & \quad \text{word} \\
\text{DTRS} & \quad \text{phrase} \\
\text{constituent-structure} & \quad \text{local} \\
\text{category} & \quad \text{content} \\
\text{MARKING} & \quad \text{marking} \\
\end{align*}
\]
Motivating **SUBCAT**

(1) a. I *laugh*.  
    b. I *saw* him.  
    c. I *give* her the book.  
    d. I *said* that she left.

Cannot always be derived from semantics:

(2) a. Paul ate a steak.  
    b. Paul ate.

(3) a. Paul devoured a steak.  
    b. * Paul devoured
Syntactic category information

Aspects of the signature

- **head**
  - **functional**
    - **SPEC** synsem
  - **substantive**
    - **PRD** boolean
    - **MOD** mod-synsem
  - **adjective**
    - **verb** vform
    - **VFORM** boolean
    - **AUX** boolean
    - **INV** boolean
  - **noun**
    - **CASE** case
  - **preposition**
    - **PFORM** pform

Marker determiner

**Introduction**

** Signs

** Phrases

** Raising/Equi

** UDCs
Properties of particular categories
Motivating VFORM

(4) a. Peter will *win* the race.  
     b. *Peter will won* the race.  
     c. *Peter will to win* the race.

(5) a. Peter has *won* the race.  
     b. *Peter has win* the race.  
     c. Peter has *to win* the race.  
        (*→ different verb*)

(6) a. Peter seems *to win* the race.  
     b. *Peter seems win* the race.  
     c. *Peter seems won* the race.
Motivating CASE

(7) a.  *He left.
      b.  *Him left.

(8) a.  She sees *him.
      b.  *She sees *he.
Indices

\[
\begin{align*}
\text{index} & \quad \text{person} \\
\text{PERSON} & \quad \text{number} \\
\text{NUMBER} & \quad \text{gender} \\
\text{GENDER} & \quad \text{referential} \\
& \quad \text{there} \\
& \quad \text{it} \\
\end{align*}
\]

person

first, second, third

number

singular, plural

gender

masculine, feminine, neuter
Semantic representations

content

quant

psoa

[laugh’

LAUGHER ref]

give’

GIVER ref

GIVEN ref

think’

THINKER ref

THOUGHT psoa

nom-obj

INDEX index

RESTRICION set(psoa)
Auxiliary data structures

Alternative names for the attributes FIRST (FT) and REST (RT) of non-empty-list are HEAD (HD) and TAIL (TL).
Abbreviations for describing lists

empty-list is abbreviated as e-list, <>

non-empty-list is abbreviated as ne-list

\[
\begin{bmatrix}
\text{FIRST} & 1 \\
\text{REST} & 2 
\end{bmatrix}
\]

is abbreviated as \( \langle 1 \mid 2 \rangle \)

\( \langle \ldots 1 \mid \langle \rangle \rangle \)

is abbreviated as \( \langle \ldots 1 \rangle \)

\[
\begin{bmatrix}
\text{FIRST} & 1 \\
\text{REST} & \begin{bmatrix}
\text{FIRST} & 2 \\
\text{REST} & 3 
\end{bmatrix}
\end{bmatrix}
\]

is abbreviated as \( \langle 1, 2 \mid 3 \rangle \)
Pollard and Sag (1994) make use of the following abbreviations for describing *synsem* objects:

Abbrev.  abbreviated AVM

**NP:**

\[
\begin{align*}
&\text{synsem} \\
&\text{LOCAL}\begin{cases}
&\text{CATEGORY} \\
&\text{CONTENT}\left|\text{INDEX}\right.\end{cases}
\end{align*}
\]

**S:**

\[
\begin{align*}
&S:1 \\
&\text{LOCAL}\begin{cases}
&\text{CATEGORY} \\
&\text{CONTENT}\end{cases}
\end{align*}
\]

**VP:**

\[
\begin{align*}
&\text{VP:1} \\
&\text{LOCAL}\begin{cases}
&\text{CATEGORY} \\
&\text{CONTENT}\end{cases}
\end{align*}
\]
Theory: The Lexicon

The basic lexicon is defined by the *Word Principle* as part of the theory. It is an implicational statement defining which of the ontologically possible words are grammatical:

\[ \text{word} \rightarrow \text{lexical-entry}_1 \lor \text{lexical-entry}_2 \lor \ldots \]

with each of the lexical entries being descriptions, e.g.:

\[
\begin{bmatrix}
\text{word} \\
\text{PHON} \\
\text{SYNSEM} | \text{LOC}
\end{bmatrix}
\begin{bmatrix}
<\text{laughs}> \\
\text{CAT} \\
\text{SUBCAT} \\
\text{CONTENT}
\end{bmatrix}
\begin{bmatrix}
\text{verb} \\
\text{VFORM} | \text{fin} \\
\langle \text{NP} | \text{nom} \rangle | \text{[1][3rd,sing]} \\
\text{laugh'} \\
\text{LAUGHER} | \text{[1]}
\end{bmatrix}
\]
An example lexicon

\[ \text{word} \rightarrow \]

\[
\begin{array}{c}
\text{PHON} <\text{drinks}> \\
\text{CAT} \\
\text{HEAD} \\
\text{VFORM} \quad \text{fin} \\
\text{SUBCAT} \quad \langle \text{NP}[\text{nom}]_1[3rd,sing], \text{NP}[\text{acc}]_2 \rangle \\
\text{S|L} \\
\text{drink'} \\
\text{CONT} \\
\text{DRINKER} \quad 1 \\
\text{DRUNKEN} \quad 2 \\
\end{array}
\]

\[
\begin{array}{c}
\text{PHON} <\text{drink}> \\
\text{CAT} \\
\text{HEAD} \\
\text{VFORM} \quad \text{fin} \\
\text{SUBCAT} \quad \langle \text{NP}[\text{nom}]_1[\text{plur}], \text{NP}[\text{acc}]_2 \rangle \\
\text{S|L} \\
\text{drink'} \\
\text{CONT} \\
\text{DRINKER} \quad 1 \\
\text{DRUNKEN} \quad 2 \\
\end{array}
\]
Theory: Lexicon

\[ \begin{align*}
\text{PHON} & \quad <\text{give}> \\
\text{CAT} & \quad [\text{verb} \quad \text{VFORM} \quad \text{fin}] \\
\text{SUBCAT} & \quad \langle \text{NP}[\text{nom}]_{1}[\text{plur}], \text{NP}[\text{acc}]_{2}, \text{PP}[\text{to}]_{3} \rangle \\
\text{CONT} & \quad \text{give'} \\
\text{GIVER} & \quad 1 \\
\text{GIFT} & \quad 2 \\
\text{GIVEN} & \quad 3
\end{align*} \]
Theory: Lexicon

\[
\begin{array}{c}
\text{PHON} \quad \langle \text{to} \rangle \\
\text{S|L} \\
\text{CAT} \\
\text{HEAD} \quad [\text{preposition}] \\
\text{SUBCAT} \quad \langle \text{NP}[\text{acc}]_1 \rangle \\
\text{CONT} \quad [\text{INDEX} \ 1] \\
\end{array}
\]

\[
\begin{array}{c}
\text{PHON} \quad \langle \text{think} \rangle \\
\text{S|L} \\
\text{CAT} \\
\text{HEAD} \quad [\text{verb}] \\
\text{VFORM} \quad \text{fin} \\
\text{SUBCAT} \quad \langle \text{NP}[\text{nom}]_1[\text{plur}], \text{S}[\text{fin}]:2 \rangle \\
\text{think'} \\
\text{THINKER} \quad 1 \\
\text{THOUGHT} \quad 2 \\
\end{array}
\]
Theory: Lexicon

```
\[
V
\]
\[
\begin{array}{c}
\text{PHON} \\
\text{SYNSEM|LOC}
\end{array}
\begin{array}{c}
<poets> \\
\begin{array}{c}
\text{CAT} \\
\text{SUBCAT} \\
\text{CONT} \\
\text{INDEX}
\end{array}
\begin{array}{c}
\text{HEAD} \\
\langle \rangle \\
\text{PER} \\
\text{NUM}
\end{array}
\begin{array}{c}
noun \\
\text{third} \\
\text{sing} \\
\text{plur}
\end{array}
\]
\]
\]
\]
\]
\]
\]
\]
\]
\]
\]
\]
\]
\]
\]
\]
\]
\]
\]
```

```
Different kinds of phrasal constructions in HPSG

For *phrases*, the **DTRS** feature has a *constituent-structure* type:

```
constituent-structure

  head-struc
  [head-dtr  sign  list(phrase)]

  head-comps-struc
  [head-marker-struc  phrase]
  [head-adjunct-struc  phrase]
  [head-filler-struc  phrase]

  head-dtr
  [HEAD-DTR  word]
  [HEAD-DTR  ADJUNCT-DTR  phrase]
  [HEAD-DTR  FILLER-DTR  phrase]

  comp-dtrs
  [COMP-DTRS  word]
  [COMP-DTRS  phrase]
  [COMP-DTRS  phrase]
```
The structure of a simple phrase

\[
\begin{align*}
\text{phrase} & : <\text{Kim, walks}> \\
\text{PHON} & : \langle\text{Kim, walks}\rangle \\
\text{SYNSEM} & : \left[\begin{array}{c}
\text{LOCAL|CAT} \\
\text{HEAD} \\
\text{SUBCAT} \right. \\
\left. \text{: verb} \\
\text{vform} \text{ fin} \right] \\
\text{head-comp-struc} & : \left[\begin{array}{c}
\text{word} \\
\text{PHON} \\
\text{SYNSEM|LOC|CAT|HEAD} \right. \\
\left. \text{: verb} \\
\text{vform} \text{ fin} \right] \\
\text{DTRS} & : \left[\begin{array}{c}
\text{HEAD-DTR} \\
\text{SYNSEM|LOC|CAT|HEAD} \right. \\
\left. \text{: verb} \\
\text{vform} \text{ fin} \right] \\
\text{COMP-DTRS} & : \left[\begin{array}{c}
\text{PHON} \\
\text{SYNSEM|LOC|CAT|HEAD} \right. \\
\left. \text{: noun} \\
\text{case} \text{ nom} \right]
\end{align*}
\]
Sketch of an example for head-complement structures
Sketch of an example for head-complement structures with a ditransitive verb
**Head-Feature Principle:**

The **HEAD** value of any headed phrase is structure-shared with the **HEAD** value of the head daughter.

\[
\begin{array}{l}
\text{phrase} \\
\text{DTRS} \; \text{headed-structure} \\
\text{SYNSEM} | \text{LOC} | \text{CAT} | \text{HEAD} \\
\text{DTRS} | \text{HEAD-DTR} | \text{SYNSEM} | \text{LOC} | \text{CAT} | \text{HEAD}
\end{array}
\]
Subcategorization Principle:

In a headed phrase (i.e. a phrasal sign whose DTRS value is of sort head-struc), the SUBCAT value of the head daughter is the concatenation of the phrase’s SUBCAT list with the list (in order of increasing obliqueness) of SYNSEM values of the complement daughters.
Subcat Principle: (cont.)

\[
[DTRS \text{ headed-structure}] \rightarrow \\
\left[\begin{array}{c}
\text{SYNSEM} | \text{LOC} | \text{CAT} | \text{SUBCAT} \\
\text{DTRS} \\
\text{HEAD-DTR} | \text{SYNSEM} | \text{LOC} | \text{CAT} | \text{SUBCAT} \\
\text{COMP-DTRS} \ synsem2sign(2)
\end{array}\right]
\]

with \( \oplus \) standing for list concatenation, i.e., \textit{append}, defined as follows

\[
e-list \oplus 1 := 1.
\]

\[
\begin{bmatrix}
\text{FIRST} \\
\text{REST}
\end{bmatrix} \oplus 3 := \begin{bmatrix}
\text{FIRST} \\
\text{REST} \oplus 3
\end{bmatrix}.
\]

and \( \text{synsem2sign} \) encoding conversion of \textit{synsem} to \textit{sign} lists defined as follows

\[
\text{synsem2sign}(e-list) := e-list.
\]

\[
\text{synsem2sign}\left(\begin{bmatrix}
\text{FIRST} \\
\text{REST}
\end{bmatrix}\right) := \begin{bmatrix}
\text{FIRST} \\
\text{REST} \ synsem2sign(2)
\end{bmatrix}.
\]
Immediate Dominance Schemata

Immediate Dominance Principle (for English):

\[
\begin{align*}
\text{[phrase} & \text{DTRS headed-struct]} \\
\text{SYNSEM|LOC|CAT} & \text{HEAD } \left( \begin{array}{c} \text{verb INV \_} \\
\text{\_ INV \_} \text{verb} \end{array} \right) \text{] (Head-Subject)} \\
\text{DTRS} & \text{HEAD-DTR phrase} \\
\text{COMP-DTRS} & \text{sign} \\
\text{SYNSEM|LOC|CAT} & \text{HEAD } \left( \begin{array}{c} \text{verb INV \_} \\
\text{\_ INV \_} \text{verb} \end{array} \right) \text{] (Head-Complement)} \\
\text{DTRS} & \text{HEAD-DTR word} \\
\text{\_ INV \_} & \text{verb}\end{align*}
\]
Immediate Dominance Principle (for English):

\[
\begin{align*}
\text{SYNSEM} & | \text{LOC} | \text{CAT} \\
\lor & \\
\text{DTRS} & \\
\text{HEAD} & \left[ \begin{array}{c}
\text{INV} \\
+ 
\end{array} \right] \\
\text{SUBCAT} & \left\langle \right\rangle \\
\text{head-comp-struc} & \\
\text{HEAD-DTR} & \text{word} \\
\hline
\text{HEAD-Marker} & \\
\text{MARKER-DTR} & \text{SYNSEM} | \text{LOC} | \text{CAT} | \text{HEAD} & \textbf{marker} \\
\text{DTRS} & \\
\text{HEAD-Adjunct} & \\
\text{ADJ-DTR} & \text{SYNSEM} | \text{LOC} | \text{CAT} | \text{HEAD} | \text{MOD} & 1 \\
\text{HEAD-DTR} & \text{SYNSEM} & 1
\end{align*}
\]
Head-Subject Phrases (Schema 1)

\[
\begin{array}{c}
\text{SYNSEM} | \text{LOC} | \text{CAT} \\
\text{HEAD} \\
\text{SUBCAT} \langle 1 \rangle \\
\end{array}
\]

\[
\begin{array}{c}
\text{SYNSEM} | \text{LOC} | \text{CAT} \\
\text{HEAD} \\
\text{SUBCAT} \langle 2 \rangle \\
\end{array}
\]
Head-Complement Phrases (Schema 2)
Head-Subject-Comp Phrases (Schema 3)
Sketch of an example with an auxiliary
Immediate Dominance Schemata

Sketch of an example with an inverted auxiliary
Some verbs select a complementized sentential complement headed by a verb with a specific verb form:

(9) I demand that he leave/*leaves immediately.
Markers

- head
  - functional
    - SPEC
      - synsem
  - substantive
    - adjective
    - verb
    - noun
    - preposition
  - marker determiner

- marking
  - unmarked
  - marked
    - comp
      - ...
        - that
        - for
Marking Principle:

In a headed structure, the MARKING value coincides with that of the marker daughter if there is one, and with that of the head daughter otherwise.

\[
\text{phrase} \\
\text{DTRS} \quad \text{headed-structure} \\
\] \rightarrow \\
\[
\begin{align*}
&\text{SYNSEM} | \text{LOC} | \text{CAT} | \text{MARKING} [1] \\
&\text{DTRS} \quad \text{head-mark-struc} \\
&\text{MARKER-DTR} | \text{SYNSEM} | \text{LOC} | \text{CAT} | \text{MARKING} [1] \\
\end{align*}
\]
\[
\vee \\
\begin{align*}
&\text{SYNSEM} | \text{LOC} | \text{CAT} | \text{MARKING} [1] \\
&\text{DTRS} \quad \neg \text{head-mark-struc} \\
&\text{HEAD-DTR} | \text{SYNSEM} | \text{LOC} | \text{CAT} | \text{MARKING} [1] \\
\end{align*}
\]
Lexical entry of the marker *that*
Immediate Dominance Schemata

Sketch of an example for a head-marker structure
SPEC Principle:

If a nonhead daughter in a headed structure bears a SPEC value, it is token-identical to the SYNSEM value of the head daughter.

\[
\begin{align*}
\text{phrase} & \\
\text{DTRS} & \left[(\text{MARKER-DTR} \lor \text{COMP-DTRS}|\text{FIRST}) \mid \text{SYNSEM}|\text{LOC}|\text{CAT}|\text{HEAD } \text{functional}\right] \\
\rightarrow & \\
\text{DTRS} & \left[(\text{MARKER-DTR} \lor \text{COMP-DTRS}|\text{FIRST}) \mid \text{SYNSEM}|\text{LOC}|\text{CAT}|\text{HEAD}|\text{SPEC} \left\lfloor 1 \right\rfloor\right] \\
& \quad \text{HEAD-DTR}|\text{SYNSEM} \left\lfloor 1 \right\rfloor
\end{align*}
\]
Common nouns subcategorize for their determiners:

\[
\begin{align*}
\text{PHON} & \quad <book> \\
\text{SYNSEM|LOC} & \\
\text{CAT} & \\
\text{SUBCAT} & \\
\text{INDEX} & \quad 1 \\
\text{CONT} & \\
\text{RESTR} & \\
\text{HEAD} & \\
\text{LOC|CAT} & \\
\text{RELN} & \quad \text{book} \\
\text{INST} & \quad 1 \\
\text{HEAD} & \\
\text{det} & \\
\end{align*}
\]
Determiners select their nominal sister:

\[
\begin{array}{c}
\text{PHON} \ <\text{every}> \\
\text{SYNSEM}\mid\text{LOC} \\
\text{QSTORE} \{5\}
\end{array}
\]

\[
\begin{array}{c}
\text{CAT} \\
\text{HEAD} \\
\text{SPEC} \\
\text{LOC} \\
\text{CAT} \\
\text{HEAD NOUN} \\
\text{SUBCAT} \langle \text{DetP} \rangle \\
\text{CONT} \{2\} \\
\text{RESTIND} \{2\} \\
\text{quant} \\
\text{for all}
\end{array}
\]
Sketch of the head-comp phrase *every book*
Lexical entry for the possessive pronoun *my*
Lexical entry for possessive 's
Sketch of an example for a determiner phrase

PHON <Mary's>

CAT

HEAD

SPEC

N'

INDEX

RESTR

QSTORE

RESTIND

DET

the

INDEX

RESTR

RELN

poss

RESTR

POSSESSOR

POSSESSED

SYNSEM

LOC

npro

INDEX

CONT

INDEX

bear

name

context

backgr

RELN

naming

BEARER

INDEX

1

1

2

3

3

3

Mary

Mary
Lexical entry of an attributive adjective
Semantics Principle

In a headed phrase, the \textbf{CONTENT} value is token-identical to that of the adjunct daughter if the \textbf{DTRS} value is of sort \textit{head-adj-struc}, and with that of the head daughter otherwise.

\[
\begin{align*}
\text{[phrase} & \text{[DTRS } \text{headed-structure]} \rightarrow \\
\text{[SYNSEM|LOC|CONTENT } & \text{[head-adj-struc]} \\
\text{DTRS } & \text{[ADJUNCT-DTR|SYNSEM|LOC|CONTENT } \text{1]} \\
\text{SYNSEM|LOC|CONTENT } & \text{1} \\
\text{DTRS } & \text{[¬head-adj-struc]} \\
\text{SYNSEM|LOC|CONTENT } & \text{1} \\
\text{DTRS } & \text{[HEAD-DTR|SYNSEM|LOC|CONTENT } \text{1]}
\end{align*}
\]
Sketch of an example for a head-adjunct structure
Raising/Equi

Review of ID Schemata

\[
\text{[phrase headed-struct]} \rightarrow \begin{cases}
\text{SYNSEM|LOC|CAT} \\
\text{DTRS}
\end{cases}
\]

\[
\text{SYNSEM|LOC|CAT} \quad \text{HEAD} \left( \left[ \begin{array}{c}
\text{verb} \\
\text{INV} \\
- \text{verb}
\end{array} \right] \lor \neg \text{verb} \right) \quad \text{SUBCAT} \langle \rangle \\
\text{DTRS} \\
\text{head-comps-struc} \\
\text{HEAD-DTR phrase} \\
\text{COMP-DTRS} \langle \text{sign} \rangle
\]

\[
\text{SYNSEM|LOC|CAT} \quad \text{HEAD} \left( \left[ \begin{array}{c}
\text{verb} \\
\text{INV} \\
- \text{verb}
\end{array} \right] \lor \neg \text{verb} \right) \quad \text{SUBCAT} \langle \text{synsem} \rangle \\
\text{DTRS} \\
\text{head-comps-struc} \\
\text{HEAD-DTR word}
\]

\[
\text{SYNSEM|LOC|CAT} \quad \text{HEAD} \left( \left[ \begin{array}{c}
\text{verb} \\
\text{INV} \\
+ \text{verb}
\end{array} \right] \right) \quad \text{SUBCAT} \langle \rangle \\
\text{DTRS} \\
\text{head-comps-struc} \\
\text{HEAD-DTR word}
\]
Towards an analysis: Unsaturated Complements

Where does it say in these ID schemata that every subcategorized item must be realized?

- In English, many verbs and adjectives subcategorize for an unsaturated complement.
- In other words, a complement can be specified as \([\text{SUBCAT } \langle \text{NP} \rangle]\), rather than \([\text{SUBCAT } \langle \rangle]\)
  - The Head-Subject Schema allows for this.
  - And this will give *consider* access to the lower subject, as well as its own subject.
Subject oriented raising verbs

\[\text{word} \quad \text{PHON} \quad \langle \text{seem} \rangle \quad \text{synsem} \quad \text{local} \quad \text{cat} \quad \text{cat} \quad \text{verb} \quad \text{VFORM} \quad bse \quad \text{subcat} \quad \langle 1 \rangle \quad \text{inf} \quad \text{subcat} \quad \langle 1 \rangle : 2 \quad \text{cont} \quad \text{soa-arg} \quad 2\]
Subject oriented equi verbs

```
word
PHON <try>

synsem
local

SYNSEM
LOCAL

cat
HEAD
VFORM
bse

SUBCAT ⟨NP[1], VP[inf], SUBCAT ⟨NP[1]:2⟩⟩

try
TRYER [1]

SOA-ARG [2]
```
Object oriented raising verbs
Object oriented equi verbs

<table>
<thead>
<tr>
<th>word</th>
<th>PHON</th>
<th>&lt;&lt;persuade&gt;&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>synsem</td>
<td></td>
<td></td>
</tr>
<tr>
<td>local</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cat</td>
<td>verb</td>
<td></td>
</tr>
<tr>
<td>HEAD</td>
<td>VFORM bse</td>
<td></td>
</tr>
<tr>
<td>SUBCAT</td>
<td>NP 1, NP 2, VP [inf, subcat ⟨NP 2⟩; 3]</td>
<td></td>
</tr>
<tr>
<td>persuade</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PERSUADER</td>
<td>1ref</td>
<td></td>
</tr>
<tr>
<td>PERSUADEE</td>
<td>2ref</td>
<td></td>
</tr>
<tr>
<td>SOA-ARG</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>
Analysis of: *It seems to rain*
Analysis of: *It wants to rain

```
word
P <wants>
  CAT
  HEAD 9 verb VFORM fin
  SUBC 3NP 11, 8VP [inf, SUBC 5NP 11]:7
  CON 10 WANTER 11ref
  SOA-ARG 7

phrase
P <wants to rain>
  CAT
  HEAD 9 SUBC 3
  CON 10

H

phrase
P <to rain>
  CAT
  HEAD 6 verb VFORM inf
  SUBC 5NP it
  CON 7 rain

C
```

*It wants to rain*
To account for UDCs, we will use the feature \textsc{slash}.

- Named after the categorial grammar categories ($S/NP$ as an $S$ missing an NP).

This is a \textsc{non-local} feature which:

- originates with a trace,
- gets passed up the tree,
- and is finally bound by a filler
An example for a strong UDC
The bottom of a UDC: Traces

- phonologically null, but structure-shares LOCAL and SLASH
- we'll talk about TO-BIND later
Because the **LOCAL** value of a trace is structure-shared with the **SLASH** value, constraints on the trace will be constraints on the filler.

- For example, *hates* specifies that its object be accusative, and this **CASE** information is local
- So, the trace has `[SYNSEM|LOCAL|CAT|HEAD|CASE  acc]` as part of its entry, and thus the filler will also have to be accusative

(10) *He$_i$/Him$_i$, John likes $_i$
The middle of a UDC
The Nonlocal Feature Principle (NFP)

For each nonlocal feature, the \textsc{INHERITED} value on the mother is the union of the \textsc{INHERITED} values on the daughter minus the \textsc{to-bind} value on the head daughter.

- In other words, the \textsc{slash} information (which is part of \textsc{INHERITED}) percolates “up” the tree.
- This allows the all the local information of a trace to “move up” to the filler.
The top of a UDC: Filler-head structures
Filler-head schema

\[
\begin{align*}
\text{phrase} & \quad \text{DTRS} \quad \text{head-filler-struc} \\
\text{HEAD-DTR} | \text{SYNSEM} & \quad \text{LOC} | \text{CAT} \\
\text{NONLOC} & \quad \text{HEAD} \quad \text{vform} \quad \text{fin} \\
\text{INHERITED} | \text{SLASH} & \quad \text{SUBCAT} \langle \rangle \\
\text{TO-BIND} | \text{SLASH} & \quad \text{element}(1) \\
\text{FILLER-DTR} | \text{SYNSEM} | \text{LOCAL} & \quad 1
\end{align*}
\]
The top of a UDC: Filler-head structures

Explanation of the schema

- Filler and trace are identified as the exact same thing (as far as their local structure is concerned)
- The trace is “bound” by the **TO-BIND** feature; this prevents the **SLASH** value from going any higher in the tree
- Only saturated finite verbs (i.e., sentences) license such structures
The top of a UDC: Filler-head structures
Example for a structure licensed by the filler-head schema

```
[LOCAL [1]]

[F]

[NLOC | INHERITED | SLASH {}]

[H]

[NLOC]

[INHERITED | SLASH \{ \ldots, [1], \ldots \}]

[TO-BIND | SLASH \{ [1] \}]
```
The analysis of the strong UDC example