Computational Semantics
Representation and Reasoning

Frank Richter

Goethe Universität Frankfurt a.M.
Institut für England- und Amerikastudien
Abteilung Linguistik

LACompLing2018, August 28–31
Stockholm University

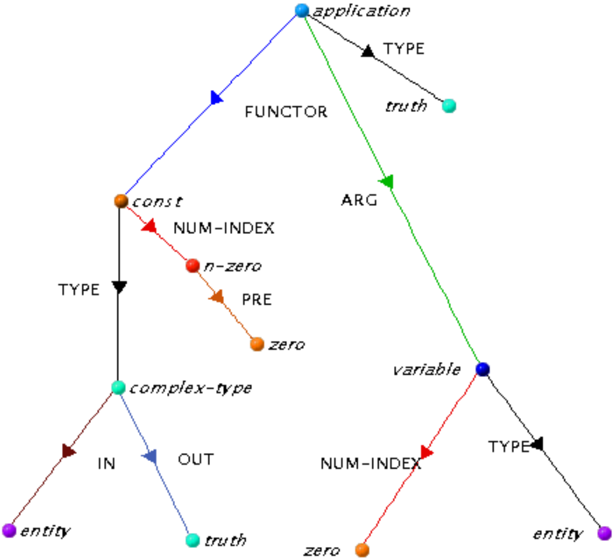
Introduction

- **L**exical **R**esource **S**emantics: Semantics in HPSG
- overview of development and state of the **C**onstraint **L**anguage for **L**exical **R**esource **S**emantics
- informal discussion of relationship between LRS and its implementation as a component of TRALE
- CLLRS in a reasoning architecture

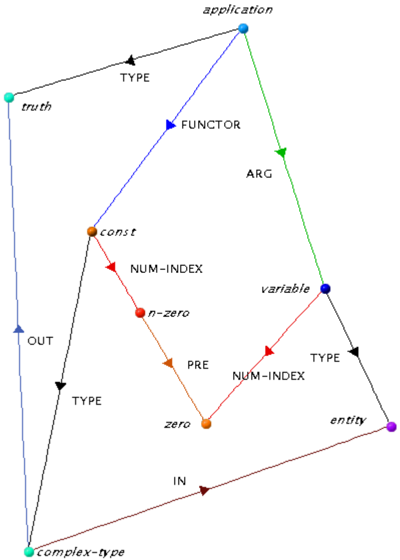
Grammar Specification in HPSG

- HPSG: Grammar = $\langle \text{Signature}, \text{Set of Principles} \rangle$
  - Signature: sort hierarchy, feature names, feature appropriateness, relation symbols and their arity
  - Principles: implicational statements (Head Feature Principle, Subcategorization Principle, ID Principle,. . .)
- Model theoretic interpretation of grammars: Linguistic expressions are structures ‘denoted’ by the grammar
- Locality assumption about principles: local ‘trees’ (or within a node)
- Consequences for semantics:
  - Semantic composition specified in the feature logic
  - Logical representations in the denotation of the grammar
  - For one sentence, several logical expressions might be possible solutions to the set of constraints imposed by the set of semantic principles

HOL Representations in HPSG (idealized)



HOL Representations in HPSG (extensional)



Lexical Resource Semantics (LRS)

1. Semantic representations from a typed logic
   - functional type theory with types *e*, *s*, and *t*
   - lambda abstraction, function application, and equality
2. Semantic composition by relations between lexical term contributions (*semantic constraints; underspecification*)
3. Central semantic composition concepts:
   - semantic term contributions (semantic resources), PARTS
   - external content: EXCONT
   - internal content: INCONT
   - subterm relationships ($\alpha \triangleleft \beta$)
4. Local semantics:
   - main content: MAIN
   - discourse referent: DR

Words: Proper Name

A proper name: *Elvis*

$$\begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{elvis} \rangle \\ \text{SYNSEM LOC CONT} & \begin{bmatrix} \text{DR} & \text{elvis}' \\ \text{MAIN} & \text{elvis}' \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \textit{lrs} & \\ \text{EXCONT} & \textit{me} \\ \text{INCONT} & \text{elvis}' \\ \text{PARTS} & \langle \text{elvis}' \rangle \end{bmatrix} \end{bmatrix}$$

SEM value in linear notation: [SEM elvis′]

In more detail: ^[{elvis′}]

Words: Proper Name

A proper name: *Elvis*

$$\begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{elvis} \rangle \\ \text{SYNSEM LOC CONT} & \begin{bmatrix} \text{DR} & \text{elvis}' \\ \text{MAIN} & \text{elvis}' \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \textit{lrs} & \\ \text{EXCONT} & \textit{me} \\ \text{INCONT} & \text{elvis}' \\ \text{PARTS} & \langle \text{elvis}' \rangle \end{bmatrix} \end{bmatrix}$$

SEM value in linear notation: [SEM elvis′]

In more detail: $\hat{}[\{\text{elvis}'\}]$

Words: Count Noun

A count noun (here: $\langle e, t \rangle$): *clown*

$$\begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{clown} \rangle \\ \text{SYNSEM LOC CONT} & \begin{bmatrix} \text{DR} & \boxed{X} \\ \text{MAIN} & \text{clown}' \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \textit{lrs} & \\ \text{EXCONT} & \textit{quantifier}(\boxed{X}, \alpha, \beta) \\ \text{INCONT} & \boxed{1}\ \text{clown}'(\boxed{X}) \\ \text{PARTS} & \langle \text{clown}'(\boxed{X}), \text{clown}' \rangle \end{bmatrix} \end{bmatrix} \quad \& \quad \boxed{1} \triangleleft \alpha$$

Informally, in linear notation: [SEM quantifier(x, _clown′(x)_ , _)]

In more detail: ˆ−quantifier(x, [{clown′(x)}] , _)

Words: Count Noun

A count noun (here: $\langle e, t \rangle$): *clown*

$$\begin{bmatrix} \textit{word} & \\ \text{PHON} & \langle \textit{clown} \rangle \\ \text{SYNSEM LOC CONT} & \begin{bmatrix} \text{DR} & \boxed{\text{X}} \\ \text{MAIN} & \text{clown}' \end{bmatrix} \\ \text{SEM} & \begin{bmatrix} \textit{lrs} & \\ \text{EXCONT} & \textit{quantifier}(\boxed{\text{X}}, \alpha, \beta) \\ \text{INCONT} & \boxed{1}\ \text{clown}'(\boxed{\text{X}}) \\ \text{PARTS} & \langle \text{clown}'(\boxed{\text{X}}), \text{clown}' \rangle \end{bmatrix} \end{bmatrix} \quad \& \quad \boxed{1} \triangleleft \alpha$$

Informally, in linear notation: [SEM quantifier(x, _clown′(x)_ , _)]

In more detail: ˆ−quantifier(x, [{clown′(x)}] , _)

Basic Principles 1

LRS PROJECTION PRINCIPLE: In each *phrase*,

1. the EXCONT values of the head and the mother are identical,

$$phrase \rightarrow \begin{bmatrix} \text{SEM EXCONT } \boxed{1} \\ \text{H-DTR SEM EXCONT } \boxed{1} \end{bmatrix}$$

```
phrase *>
   (sem:  @sem([^X]),
    hdtr:sem:  @sem([^X])).
```

2. the INCONT values of the head and the mother are identical,

$$phrase \rightarrow \begin{bmatrix} \text{SEM INCONT } \boxed{1} \\ \text{H-DTR SEM INCONT } \boxed{1} \end{bmatrix}$$

```
phrase *>
   (sem:  @sem([{X}]),
    hdtr:sem:  @sem([{X}])).
```

Basic Principles 2

3. the PARTS value contains all and only the elements of the PARTS values of the daughters.

$$phrase \rightarrow \left(\begin{bmatrix} \text{SEM PARTS} & \boxed{1} \\ \text{H-DTR SEM PARTS} & \boxed{2} \\ \text{NH-DTR SEM PARTS} & \boxed{3} \end{bmatrix} \wedge \ \texttt{append}(\boxed{2}, \boxed{3}, \boxed{1}) \right)$$

```
phrase *>
   (sem:  @sem([X,Y]),
    hdtr:sem:  @sem(X),
    nh_dtr:sem:  @sem(Y)).
```

From the Semantics Principle (1)

SEMANTICS PRINCIPLE (clause for Det + N′):
If the non-head is a quantificational determiner then its INCONT value is of the form *quantifier*(x, ρ, ν), the INCONT value of the head is a component of ρ, and the INCONT value of the non-head daughter is identical with the EXCONT value of the head daughter

$$\left[\text{NH-DTR SS LOC}\ \begin{bmatrix}\text{CAT HEAD} & \textit{det}\\ \text{CONT MAIN} & \textit{quantifier}\end{bmatrix}\right] \rightarrow$$

$$\left(\begin{bmatrix}\text{H-DTR SEM} & \begin{bmatrix}\text{EXCONT} & \boxed{1}\\ \text{INCONT} & \boxed{2}\end{bmatrix}\\ \text{NH-DTR SEM} & \begin{bmatrix}\text{INCONT} & \boxed{1}\begin{bmatrix}\textit{quantifier}\\ \text{RESTR}\ \boxed{3}\end{bmatrix}\end{bmatrix}\end{bmatrix} \wedge\ \boxed{2} \triangleleft \boxed{3}\right)$$

From the Semantics Principle (1, continued)

$$
\left[\text{NH-DTR SS LOC} \begin{bmatrix} \text{CAT HEAD} & \textit{det} \\ \text{CONT MAIN} & \textit{quantifier} \end{bmatrix}\right] \rightarrow
$$

$$
\left(\begin{bmatrix} \text{H-DTR SEM} & \begin{bmatrix} \text{EXCONT} & \boxed{1} \\ \text{INCONT} & \boxed{2} \end{bmatrix} \\ \text{NH-DTR SEM} & \left[\text{INCONT} \ \boxed{1} \begin{bmatrix} \textit{quantifier} \\ \text{RESTR} \ \boxed{3} \end{bmatrix}\right] \end{bmatrix} \wedge \ \boxed{2} \triangleleft \boxed{3} \right)
$$

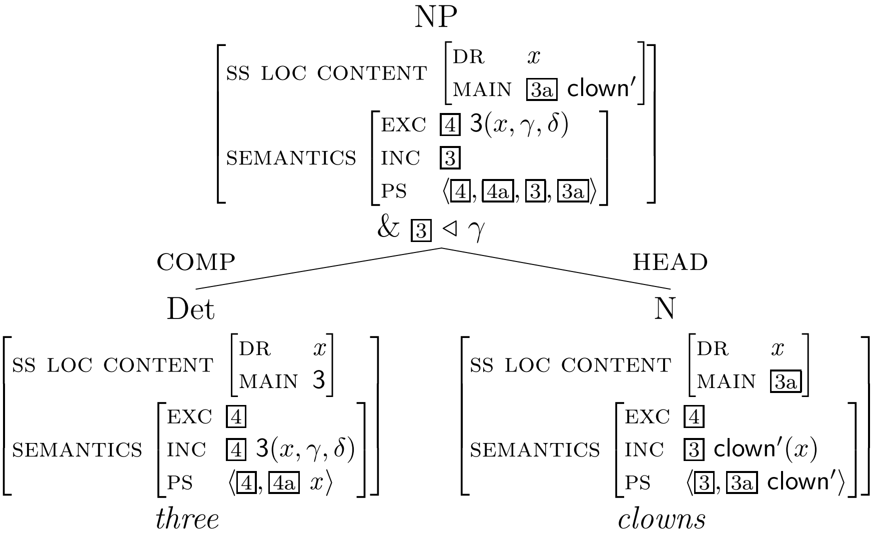
```
(phrase,
 nh_dtr:synsem:loc:(cat:head:det,
                    cont:main:@sem(quantifier)) *>
   (nh_dtr:sem:  (@sem([{quantifier(x,[Two],_)}]),
                  @sem([{One}]) ),
    hdtr:sem:  @sem([^One:[{Two}]]) ).
```

Local Semantic Projection

Local semantic values are inherited along syntactic head paths:

$$\left[\textit{headed\_phrase}\right] \rightarrow \begin{bmatrix} \text{SS LOC CONT} \begin{bmatrix} \text{DR} & \boxed{1} \\ \text{MAIN} & \boxed{2} \end{bmatrix} \\ \text{H-DTR SS LOC CONT} \begin{bmatrix} \text{DR} & \boxed{1} \\ \text{MAIN} & \boxed{2} \end{bmatrix} \end{bmatrix}$$

A Noun Phrase in LRS Notation



Basic Principles 3

The INCONT PRINCIPLE:
In each *lrs*, the INCONT value is an element of the PARTS list and a component of the EXCONT value.

$$lrs \rightarrow \left(\begin{bmatrix} \text{EXCONT} & \boxed{1} \\ \text{INCONT} & \boxed{2} \\ \text{PARTS} & \boxed{3} \end{bmatrix} \wedge \ \texttt{member}(\boxed{2}, \boxed{3}) \wedge \ \boxed{2} \triangleleft \boxed{1} \right)$$

Basic Principles 4

The EXCONT PRINCIPLE:

Clause (a):
In every phrase, the EXCONT value of the non-head daughter is an element of the non-head daughter's PARTS list.

$$phrase \rightarrow \left(\left[\text{NH-DTR SEM} \begin{bmatrix} \text{EXCONT} & \boxed{1} \\ \text{PARTS} & \boxed{2} \end{bmatrix} \right] \wedge \ \mathtt{member}(\boxed{1}, \boxed{2}) \right)$$

Clause (b):
In every utterance, every subexpression of the EXCONT value of the utterance is an element of its PARTS list, and every element of the utterance's PARTS list is a subexpression of the EXCONT value.

$$u\text{-}sign \rightarrow$$

$$\forall \boxed{1}\ \forall \boxed{2}\ \forall \boxed{3}\ \forall \boxed{4} \left(\begin{array}{l} \left(\left[\text{SEM} \begin{bmatrix} \text{EXCONT} & \boxed{1} \\ \text{PARTS} & \boxed{2} \end{bmatrix} \right] \wedge \ \boxed{3} \triangleleft \boxed{1} \ \wedge \ \mathtt{member}(\boxed{4}, \boxed{2}) \right) \rightarrow \\ (\mathtt{member}(\boxed{3}, \boxed{2}) \ \wedge \ \boxed{4} \triangleleft \boxed{1}) \end{array} \right)$$

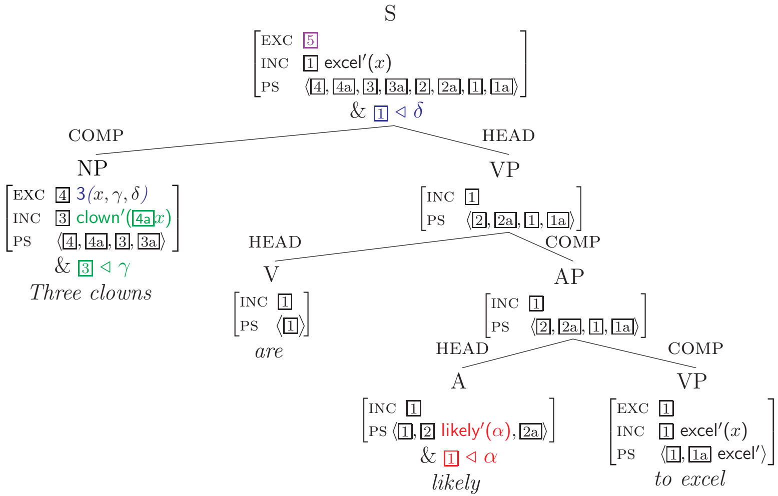
From the Semantics Principle (2)

SEMANTICS PRINCIPLE (clause for NP + VP):

2. if the non-head is a quantified NP with an EXCONT value of the form *quantifier*(x, ρ, ν), then the INCONT value of the head is a component of ν,

$$\forall \boxed{1} \left(\begin{bmatrix} \text{NH-DTR} & \begin{bmatrix} \text{SS LOC CAT} & \begin{bmatrix} \text{HEAD} & \textit{noun} \\ \text{SUBCAT} & \langle\rangle \end{bmatrix} \\ \text{SEM EXCONT} & \begin{bmatrix} \textit{quantifier} \\ \text{SCOPE} & \boxed{1} \end{bmatrix} \end{bmatrix} \end{bmatrix} \rightarrow \exists \boxed{2} \left(\begin{bmatrix} \text{H-DTR SEM INCONT} & \boxed{2} \end{bmatrix} \wedge \boxed{2} \triangleleft \boxed{1} \right) \right)$$

LRS: A Sentence



1. $\boxed{5} = 3(x, \text{clown}'(x), \text{likely}'(\text{excel}'(x)))$
2. $\boxed{5} = \text{likely}'(3(x, \text{clown}'(x), \text{excel}'(x)))$

CLLRS timeline

- authors: Gerald Penn, Frank Richter, Manfred Sailer
- a joint project (Tübingen/Toronto) in 2002/2003 on electronic resources for HPSG resulted in a first prototype implementation
- Penn & Richter (2004): *Lexical Resource Semantics: From Theory to Implementation* (HPSG Proceedings)
  Penn & Richter (2005): *The Other Syntax: Approaching Natural Language Semantics through Logical Form Composition* (in volume on constraint solving and language processing)
- GUI components by Martin Lazarov, ca. 2007–2011
- LSA summer school 2011, Penn & Richter in Boulder, Colorado
- status: work in progress in Toronto and Frankfurt

Semantic Typing

- Let T be a countable set of Roman symbols, called *basic types*.
- Let TV be a countable set of Greek symbols, called *type variables*.
- Let Types_T be the smallest set such that:
  - $T \subseteq \mathrm{Types}_T$,
  - $TV \subseteq \mathrm{Types}_T$, and
  - if $s, t \in \mathrm{Types}_T$, then $s \to t \in \mathrm{Types}_T$.
- Let Ground_T be the smallest set such that:
  - $T \subseteq \mathrm{Ground}_T$, and
  - if $s, t \in \mathrm{Ground}_T$, then $s \to t \in \mathrm{Ground}_T$.
- Every type in Types_T can be thought of as denoting a set of types from Ground_T in which each type variable ranges over the types of Ground_T.

CLLRS: Summary of Syntax

Constraint Language for Lexical Resource Semantics:

Description	Abstract Syntax	Concrete Example
literal/arity	lit/n	`see(_,_)`
pivot	$\{\phi\}$	`if(P,{Q})`
root	$\hat{}\phi$	`^forall(x,if(P,{Q}))`
object variable	x	`^lambda(x,P)`
meta-variable	X	`S:see(x,y)`
subterm(s)	$\phi \triangleleft X$	`P:[see(x,y)]`
immediate subterm	$\phi \swarrow_n \text{lit}/a$	`see(Y,Z)`
not contributed	$-\text{lit}$	`-neg([∃(x,[human(-w,x)],[x])])`
application	ϕ ap $\vec{\eta}$	`see ap (w,x,y)`

CLLRS: A Grammar Fragment

Purpose and scope of the grammar fragment:

- testing environment for development
- captures central LRS principles
- intensionality, event variables, generalized quantifiers
- embedded complement clauses
- iota operator for definite noun phrases
- different kinds of adjectives (intersective, subsective, privative)
- perspective: provide logical representations for sophisticated reasoning architecture

Semantic Type Declarations

```
semtype [t,f]: t.
semtype neg: (t->t).
semtype [and,or,impl,repl,equi]: (t->t->t).
semtype lambda: (A->B->(A->B)).

semtype w: var(s).
semtype [a,e,x,y,z]: var(e).

semtype [peter,mary]: e.
semtype [student,book,girl,person]: (s->e->t).

semtype walk: (s->e->e->t).
semtype [read,like]: (s->e->e->e->t).
semtype say: (s->e->e->(s->t)->t).

findom quantifier:[every,indefinite,some,exists].
```

Examples of Lexical Entries

Source code and graphical representation of CLLRS term descriptions of:

- proper name: *Peter*: e
- count noun: *student*: (s->e->t)
- quantifier determiner: *every*: (e->t->t->t)
- verb: *walks*: (s->e->e->t)

Generalized Quantifier, Simple Sentence

- *every student*
  $\rightarrow$ Det+Noun semantic composition
- *every student walks*
  $\rightarrow$ generalized quantifier+VP semantic composition

The composition rules mirror the corresponding clauses of the LRS *Semantics Principle*.

Note that *the student* follows the pattern of *every student* in semantic composition.

Adjectives

Starting point for the representation of adjectives:

$\lambda P_{\langle s \langle et \rangle \rangle} \lambda w_s \lambda x_e . tall_{\langle s \langle \langle s \langle et \rangle \rangle \langle et \rangle \rangle \rangle}(w, P, x)$

Motivation:

Uniform syntactic form for intersective, subsective, privative and other types of adjectives. Meaning postulates guarantee the intended inferential behavior.

- *blond student* (intersective)
- *successful student* (subsective)
- *fake student* (privative)
- *alleged student*

Adjectives: Meaning postulates

from Hahn & Richter (2015); in (1)-(4), α is the adjective:

1. intersective adjectives: *blond, Scandinavian, Irish, British, female, male*
   $\exists P^1_{\langle s\langle et\rangle\rangle} \forall w_s \forall P^2_{\langle s\langle et\rangle\rangle} \forall x_e (\alpha(w, P^2, x) \leftrightarrow (P^1(w, x) \wedge P^2(w, x)))$
2. subsective, non-intersective adjectives: *genuine, skillful, successful, interesting, large, small, fat, tall, blue*
   $\forall P_{\langle s\langle et\rangle\rangle} \forall x_e \forall w_s (\alpha(w, P, x) \rightarrow P(w, x))$
3. privative adjectives: *fake, former*
   $\forall P_{\langle s\langle et\rangle\rangle} \forall x_e \forall w_s (\alpha(w, P, x) \rightarrow \neg P(w, x))$
4. *alleged*
   $\forall P_{\langle s\langle et\rangle\rangle} \forall x_e \forall w^1_s (\mathit{alleged}(w^1, P, x) \leftrightarrow$
   $\mathit{allegedly}(w^1, (\lambda w^2 P(w^2, x))))$

Adjectives: Implementation

adjectives like *smart*, and its combinatorial semantics:

- internal content: adjectives and it's arguments
- world variable not contributed
- DR available by MOD

combinatorics of head-adjective structures:

- takes INCONT of head as argument
- INCONT is inherited from adjective daughter
- EXCONT of adjective daughter remains underspecified

The Definite Article and Definite NPs

ι operator of type (e->t)->e:

- idea: definite noun phrases provide discourse referent in DR
- definite article selects DR value of head via SPEC
- semantics: `@semcontrib(^{Y:iota(lambda(X:x,[x]))})` (specify only INCONT for cases like *all the*?)
- its own DR value contains the ι term
- Semantics Principle Det + N′: definite article takes INCONT of head as subterm of the lambda abstract
- DR value is inherited from determiner daughter in phrases with determiner daughter
- Semantics Principle: clause for quantifiers + VP ‘ignores’ definite NPs and proper names

Complex Sentences

- *Peter says [that Mary reads the book]*
- *Peter says [that every student walks]*
  $\rightarrow$ V+S semantic composition

Note the quantifier island status of the complement clause in the current implementation.

It is due to the *Sentential Proposition Restriction*.

Observation: EXTERNAL content plays a central role for statements on scope restrictions, but it interacts with other specifications.

Noun Phrases: *apparently* + Adjective

- *the student*: semantic composition in DR
- *apparently smart student*
- *apparently fake student*
- *every/the apparently smart student*

Analysis: *apparently* syntactically combines with the adjective, with the adjective the syntactic head of the construction.
Semantically, *apparently* is a function that takes the adjectival head as argument and returns an expression of the same type.

Next Steps

- closure operator for utterances
- full set of negative concord constraints
- support for polyadic quantifiers
- syntactic primitives for constraints on readings
- enumeration of (filtered) fully specified readings
- integration in higher-order reasoning architecture

Conclusions

- LRS supports the integration of a semantics with a higher-order language in HPSG
- the usual underspecification techniques are available...
- and identity of meaning contributions
- CLLRS constructs representations with CHR
- CLLRS supports underspecification of arguments and functors
- semantics must support reasoning