



Implementacja gramatyk HPSG w systemie TRALE

Beata Trawiński

`trawinski@sfs.uni-tuebingen.de`

Instytut Podstaw Informatyki Polskiej Akademii Nauk,
Warszawa, 29.11.2002



- **MiLCA** = (**M**edien**i**ntensive **L**ehrmodule in der **C**omputerlinguistik-**A**usbildung)

<http://milca.sfs.uni-tuebingen.de>

- Moduł A4 (*Formalizmy Gramatyczne i Parsing*)
Podmoduły:

- Rozwój teorii lingwistycznej w oparciu o HPSG
- Podstawy formalne HPSG
- Implementacja gramatyk HPSG w systemie TRALE

<http://milca.sfs.nphil.uni-tuebingen.de/A4/HomePage/top.html>

- **ILIAS**

<http://www.ilias.uni-koeln.de>



Cel implementacji komputerowej gramatyk

Implementacja komputerowa gramatyk przydatna jest w sensie:

- weryfikacji formalizacji teorii lingwistycznej,
- stymulacji rozwoju systemu komputerowego w kierunku osiągnięcia coraz ściślejszego powiązania pomiędzy teorią a implementacją oraz w sensie doskonalenia procesu przetwarzania.



System przystosowany do implementacji gramatyk powinien:

- umożliwiać taką implementację gramatyk, aby była ona jak najbardziej zbliżona do teorii lingwistycznej

oraz

- być jak najbardziej efektywny z komputerowego punktu widzenia.



Formalizm HPSG 87:

- wprowadzony nieformalnie w Carl J. Pollard i Ivan A. Sag (1987). *Information-Based Syntax and Semantics. Volume 1: Fundamentals*. Stanford: CSLI Publications.
- oparty na *unifikacji (unification-based)* czy też *informacji (information-based)*
- zdefiniowany np. w Paul J. King (1989). *A Logical Formalism for Head-Driven Phrase Structure Grammar*. PhD thesis, University of Manchester.

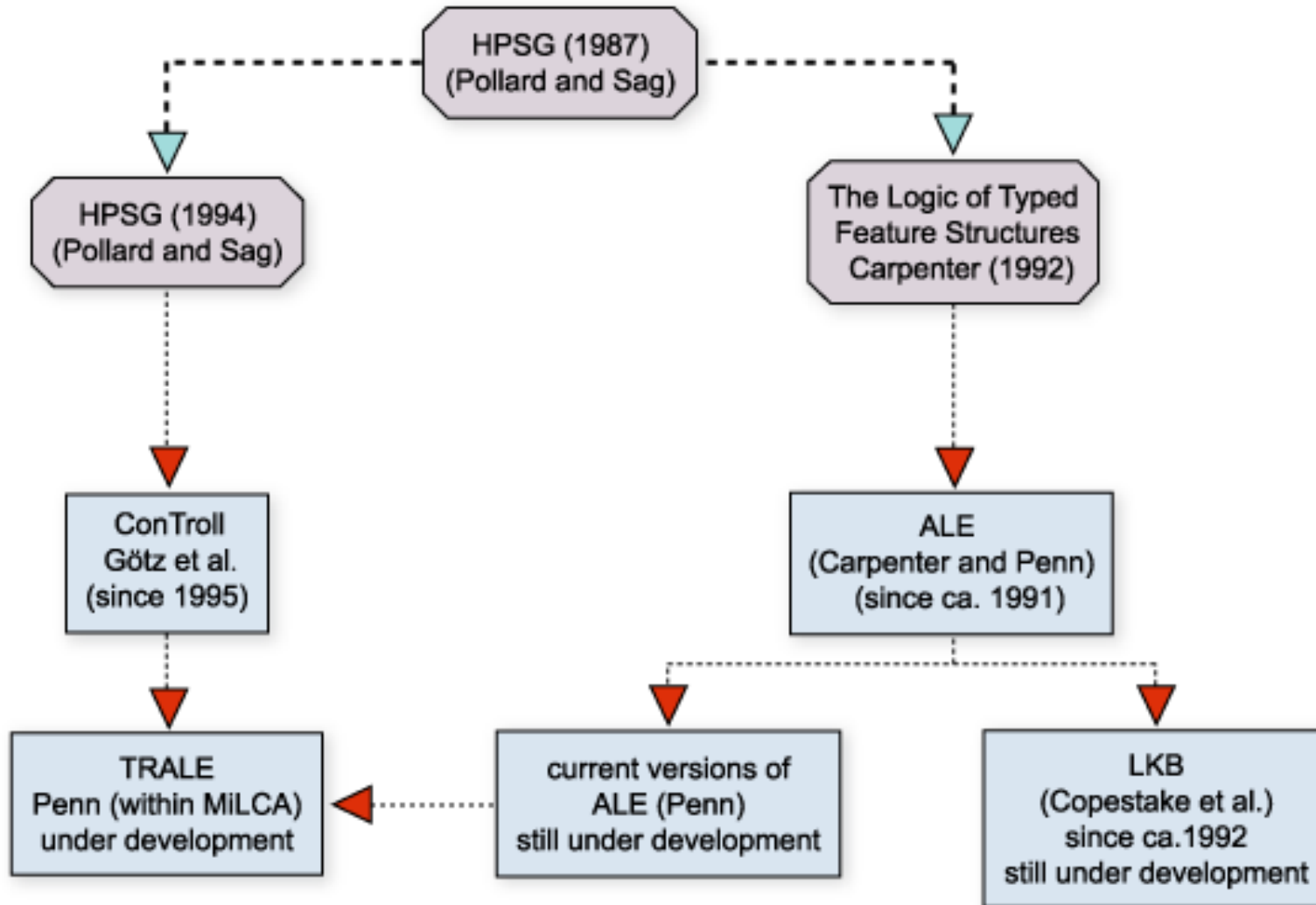


Formalizm HPSG 94:

- wprowadzony nieformalnie w Carl J. Pollard i Ivan A. Sag (1994). *Head-Driven Phrase Structure Grammar*. Chicago: The University of Chicago Press.
- oparty na *obiektach* (*object-based*) czy też *ograniczeniach* (*constraint-based*)
- zdefiniowany np. w Frank Richter (2000). *A Mathematical Formalism for Linguistic Theories with an Application in Head-Driven Phrase Structure Grammar*. PhD thesis, Eberhard-Karls-Universität Tübingen, Version of April 28th, 2000.



System TRALE





Gramatyka HPSG w ujęciu lingwistycznym

Z punktu widzenia lingwistycznego gramatyka HPSG może obejmować następujące komponenty:

- **słownik**, który dostarcza podstawowe formy wyrazowe,
- **reguły leksykalne**, które dostarczają derywowane formy wyrazowe,
- ***immediate dominance (ID) schemata***, które są odpowiedzialne za strukturę składniową zdań,
- ***linear precedence (LP) statements***, które określają szyk wyrazów w zdaniu oraz
- **zbiór ograniczeń**, które wyrażają generalizacje na temat cech gramatycznych obiektów językowych.



Gramatyka HPSG w ujęciu formalnym

W sensie formalnym gramatyka HPSG składa się z:

- **sygnatury** (określającej domenę), która obejmuje:
 - zbiór symboli typów (w tym typów maksymalnych),
 - zbiór symboli atrybutów,
 - zbiór symboli relacyjnych,
 - zbiór arności relacji,
 - funkcję określającą, jakie atrybuty mogą występować z jakim typami,
 - hierarchię typów;
- **teorii** (ograniczającej tę domenę), która jest zbiorem ograniczeń czy też zasad, będących formułami języka służącego do zapisu teorii HPSG.

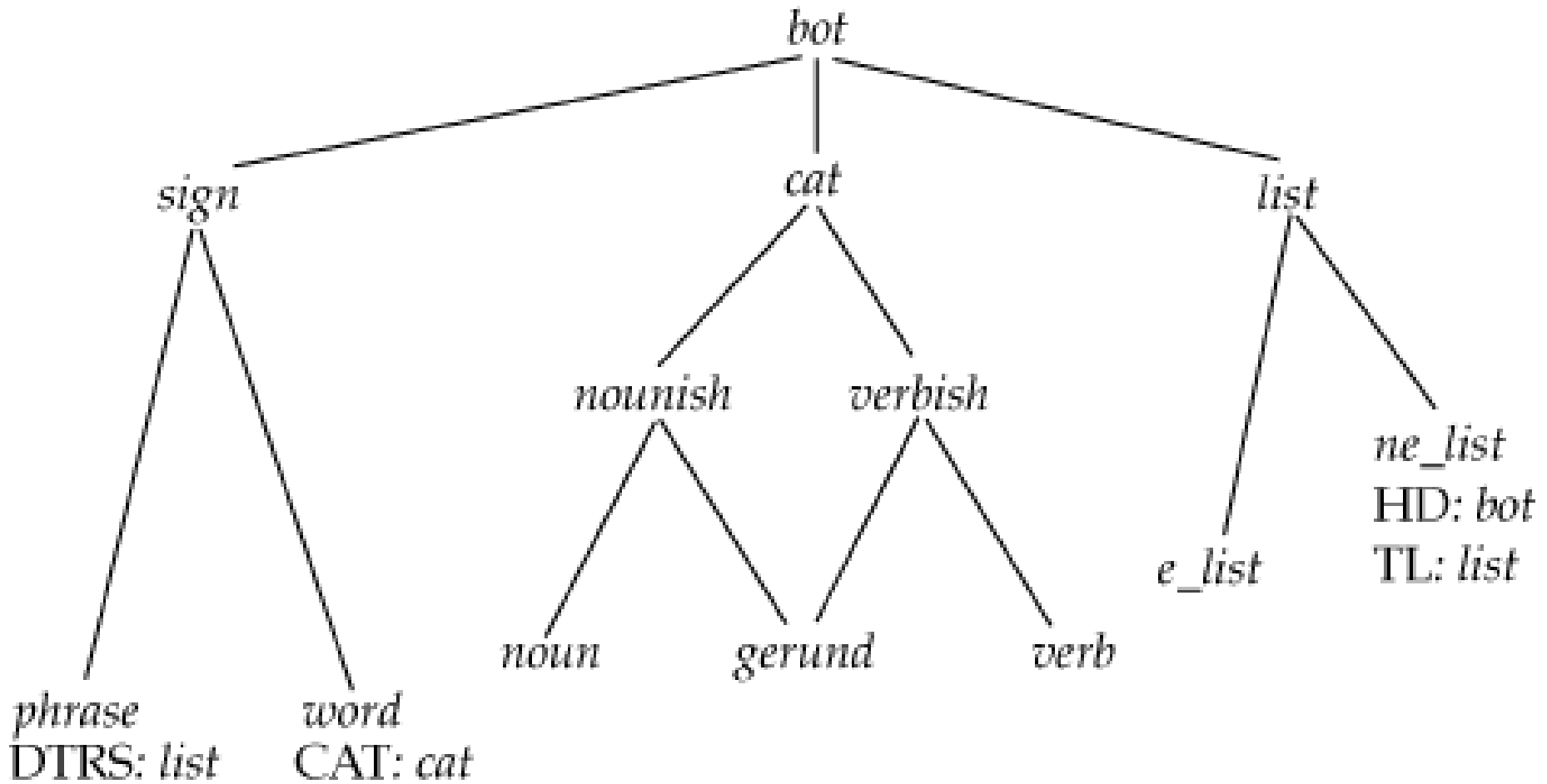


Cechy sygnatur HPSG

- Sygnatura HPSG organizuje typy w **hierarchię z wielodziedziczeniem** (*multiple inheritance hierarchy*).
- Relację pomiędzy typami w hierarchii charakteryzuje **porządek częściowy** (*partial order*).
- Hierarchia typów spełnia warunek **utypowienia zupełnego** (*exhaustive typing* czy też *partition condition*); interpretacja hierarchii typów sygnatur HPSG charakteryzowana jest jako tzw. *closed word*.



Przykładowa sygnatura w formie drzewa





Theoria gramatyk HPSG

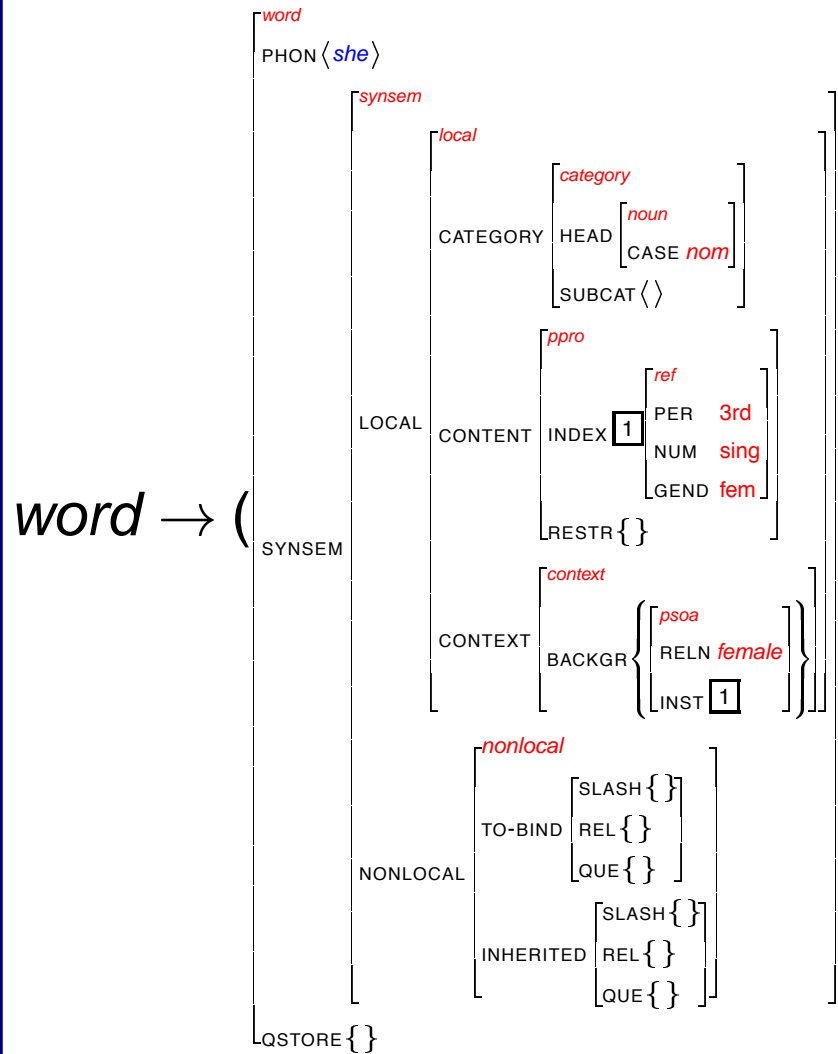
- Theoria gramatyk HPSG jest zbiorem ograniczeń, takim że każde ograniczenie opisuje każdy obiekt w modelu.
- Ograniczenia mogą mieć formę:
 - prostej specyfikacji typu,
 - złożonej deskrypcji z przydzieleniem typu,
 - złożonej deskrypcji z przydzieleniem typu i równością ciągów atrybutów,
 - koniunkcji,
 - dysjunkcji,
 - negacji,
 - implikacji,
 - ekwiwalencji,
 - wyrażeń zawierających kwantyfikację uniwersalną i egzystencjalną.



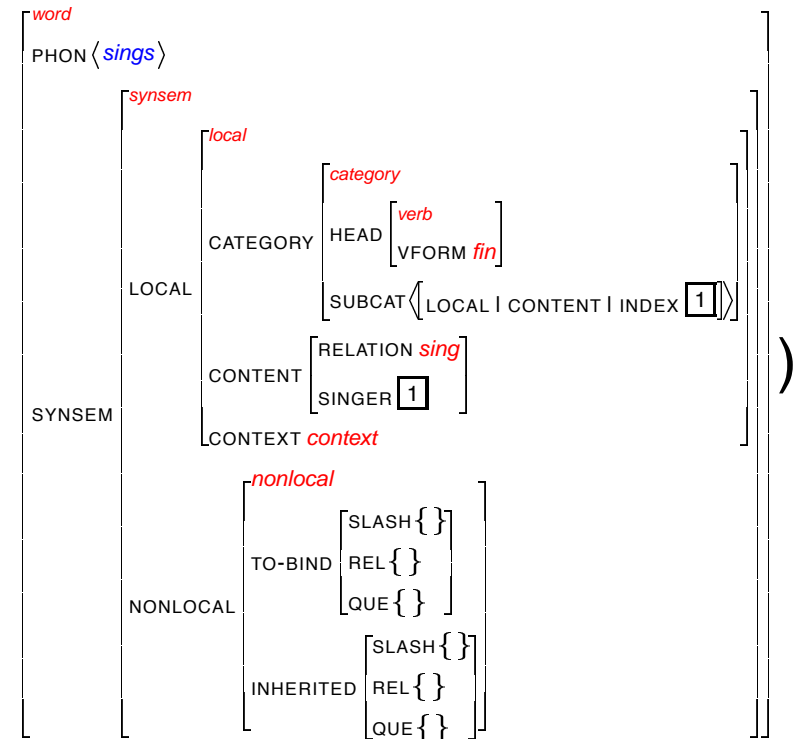
$$\textit{word} \rightarrow (\mathbf{LE}_1 \vee \dots \vee \mathbf{LE}_n)$$



Leksykon



∨





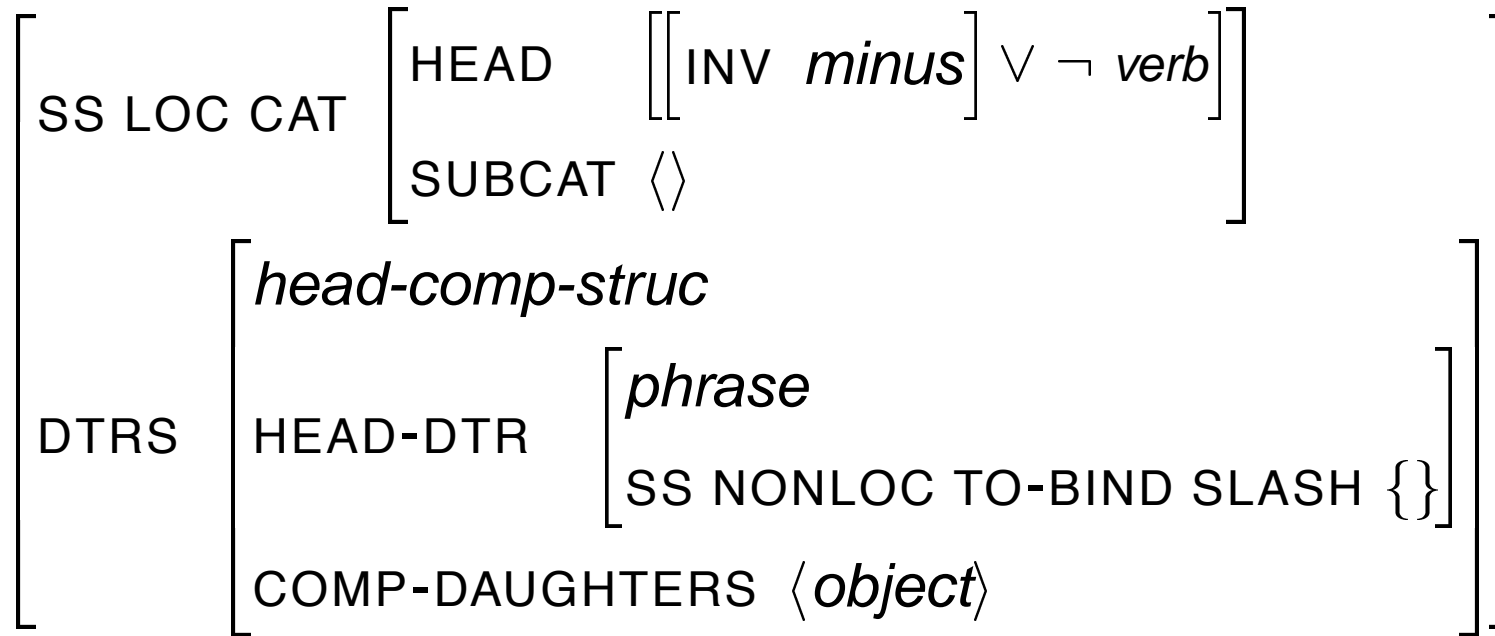
Every headed phrase must satisfy exactly one of the ID schemata.

$[DTRS \textit{headed-struct}] \rightarrow$

$(SCHEMA1 \vee SCHEMA2 \vee SCHEMA3 \vee SCHEMA4 \vee SCHEMA5 \vee SCHEMA6)$

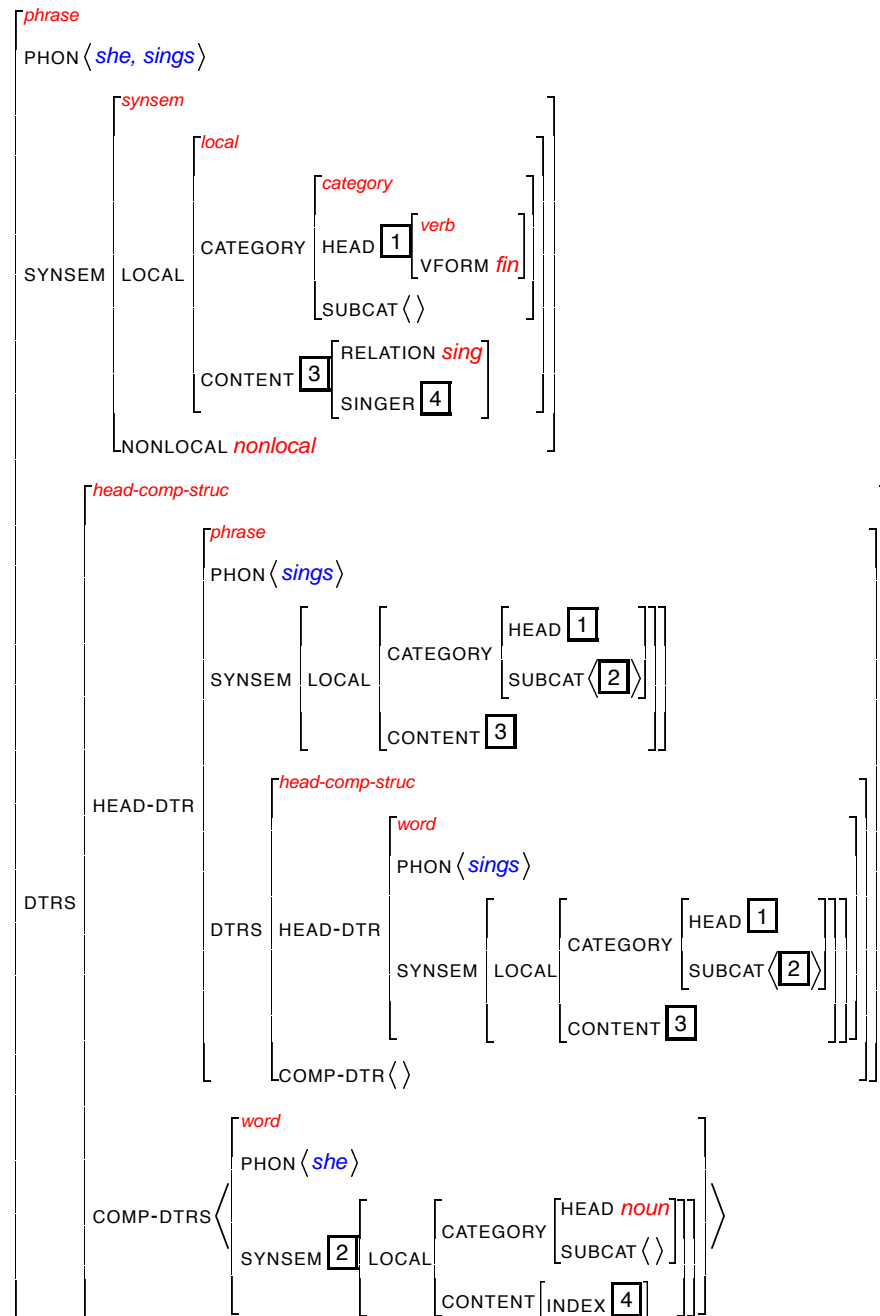


SCHEMAT 1 (HEAD-SUBJECT SCHEMA)



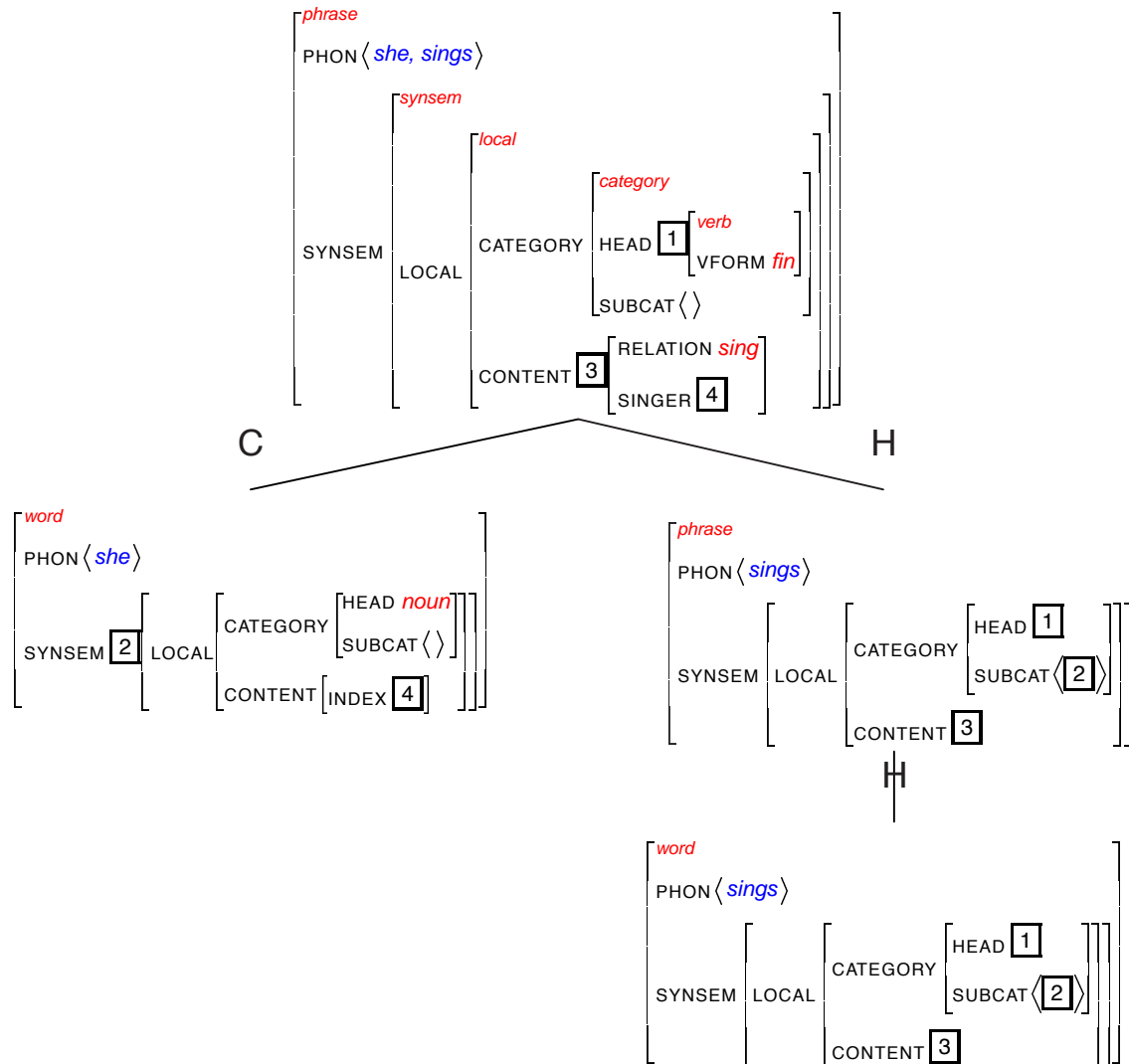


Opis frazy *She sings* w notacji AVM





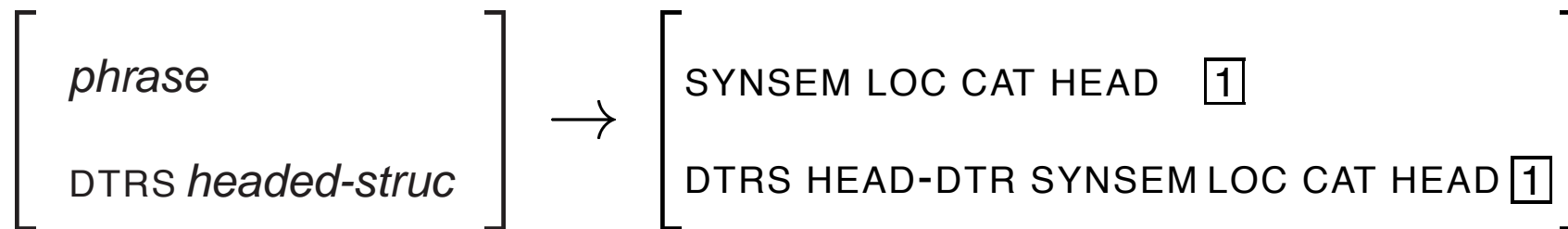
Opis frazy *She sings* w formie drzewa





HEAD FEATURE PRINCIPLE

In a headed phrase, the values of SYNSEM | LOCAL | CATEGORY | HEAD and DAUGHTERS | HEAD-DAUGHTER | SYNSEM | LOCAL | CATEGORY | HEAD are token-identical.





SUBCATEGORIZATION PRINCIPLE

In a headed phrase, the list value of DAUGHTERS | HEAD-DAUGHTER | SYNSEM | LOCAL | CATEGORY | SUBCAT is the concatenation of the list value of SYNSEM | LOCAL | CATEGORY | SUBCAT with the list consisting of the SYNSEM values (in order) of the elements of the list value of

DAUGHTERS | COMPLEMENT-DAUGHTERS. $\left[\begin{array}{l} \textit{phrase} \\ \textit{DTRS headed-struct} \end{array} \right] \rightarrow$

$\left[\begin{array}{l} \text{SYNSEM LOC CAT SUBCAT } \boxed{1} \\ \text{DTRS } \left[\begin{array}{l} \text{HEAD-DTR SYNSEM LOC CAT SUBCAT } \boxed{2} \\ \text{COMP-DTRS } \boxed{3} \end{array} \right] \end{array} \right]$

$\wedge \text{sign-to-synsem}(\boxed{3}, \boxed{4})$

$\wedge \text{append}(\boxed{1}, \boxed{4}, \boxed{2})$



Relacja sign-to-synsem

The Relation `sign-to-synsem` holds between a list x of *sign* objects and a list y of *synsem* objects iff

- x is an empty list and y is an empty list or
- the `SYNSEM` value of the first element of x is the first element of y and the relation `sign-to-synsem` holds between the lists x and y each shortened with respect to the first element.

Formalization:

$\forall x \forall y$

$(\text{sign-to-synsem}(x,y) \leftrightarrow$

$((x \text{ [elist]} \wedge y \text{ [elist]}) \vee$

$\exists \boxed{1} \exists \boxed{2} \exists \boxed{3}$

$((x \text{ [FIRST SYNSEM } \boxed{3}] \wedge y \text{ [FIRST } \boxed{3}]} \wedge \text{sign-to-synsem}(\boxed{1}, \boxed{2})) \vee$
 $(x \text{ [REST } \boxed{1}] \wedge y \text{ [REST } \boxed{2}]) \wedge \text{sign-to-synsem}(\boxed{1}, \boxed{2}))$))



Relacja append

The relation $\text{append}(x,y,z)$ holds iff the list z is the concatenation of the list x and the list y .

Formalization:

$\forall x \forall y \forall z (\text{append}(x,y,z) \leftrightarrow$

$((^x[\text{elist}] \wedge y = z) \vee$

$\exists [1] \exists [2] \exists [3] \left(x \begin{bmatrix} \text{FIRST } [1] \\ \text{REST } [2] \end{bmatrix} \wedge z \begin{bmatrix} \text{FIRST } [1] \\ \text{REST } [3] \end{bmatrix} \wedge \text{append}([2],y,[3]) \right)))$



Komponenty gramatyk TRALE

Gramatyka TRALE obejmuje następujące komponenty:

- sygnaturę,
- teorię, specyfikowaną z użyciem:
 - opisów utypowionych struktur atrybutów (w tym makrodefinicji),
 - ograniczeń implikacyjnych,
 - procedur,
 - reguł frazowych,
 - jednostek leksykalnych,
 - reguł leksykalnych.



Format sygnatur TRALE

Sygnatury TRALE charakteryzują się tym, że są one

- specyfikowane w osobnym pliku (zwykle o nazwie *signature*),
- umieszczane pomiędzy słowem kluczowym **type_hierachy** a kropką (.),
- inicjowane poprzez symbol typu **bot** czy też **top**,
- zbudowane z wierszy, składających się z:
 - wcięcia,
 - symbolu typu,
 - pewnej liczby (także zerowej) symboli atrybutów oraz oddzielonych od nich dwukropkiem (:) symboli typów, które są dla tych atrybutów odpowiednie.



Przykładowa sygnatura TRALE

type_hierarchy

bot

sign

phrase dtrs:list

word cat:cat

cat

nounish

noun

gerund

verbish

verb

&gerund

list

e_list

ne_list hd:bot tl:list



Przykładowa teoria TRALE I

```
% Specify signature file
signature(signature).

% Lexicon
she ---> (word,phon:[(a_ she)], cat:noun).

signs ----> (word,phon:[(a_ sings)], cat:verb).

% Phrase Structure Rule
subj_head_rule rule
(phrase,phon:[SubjPhon,HeadPhon],dtrs:[Subj,Head] ==>
cat> (Subj,word,phon:[SubjPhon],cat:noun),
cat> (Head,word,phon:[HeadPhon],cat:verb)).
```



Screenshot 1

```
emacs: *prolog*
File Edit Mule Apps Options Buffers Tools Complete In/Out Signals Prolog Help
Open Dired Save Print Cut Copy Paste Undo Spell Replace Mail Info Compile Debug
% restoring /home/milca/a4/systems/trale/Trale.SunOS-5.8.sav...
% /home/milca/a4/systems/trale/Trale.SunOS-5.8.sav restored in 310 ms
ec 1873624 bytes

TRALE Milca environment (version 2.1.8)
Copyright (C) 2002 Project MILCA A4
PIs: Gerald Penn (Toronto), Detmar Meurers (OSU), Frank Richter (Tübingen)
All rights reserved

SICStus 3.9.0 (sparc-solaris-5.7): Tue Feb 12 17:06:38 MET 2002
Licensed to sfs.nphil.uni-tuebingen.de
| ?- yes
% source_info
| ?-

IS08---*-XEmacs: *prolog* (Inferior Prolog: run)----All-----
Loading prolog...done
```



Przykładowa teoria TRALE II

```
% Specify signature file
signature(signature).

:- tree_extensions.
:- multifile if/2.
:- multifile ~~>/2.

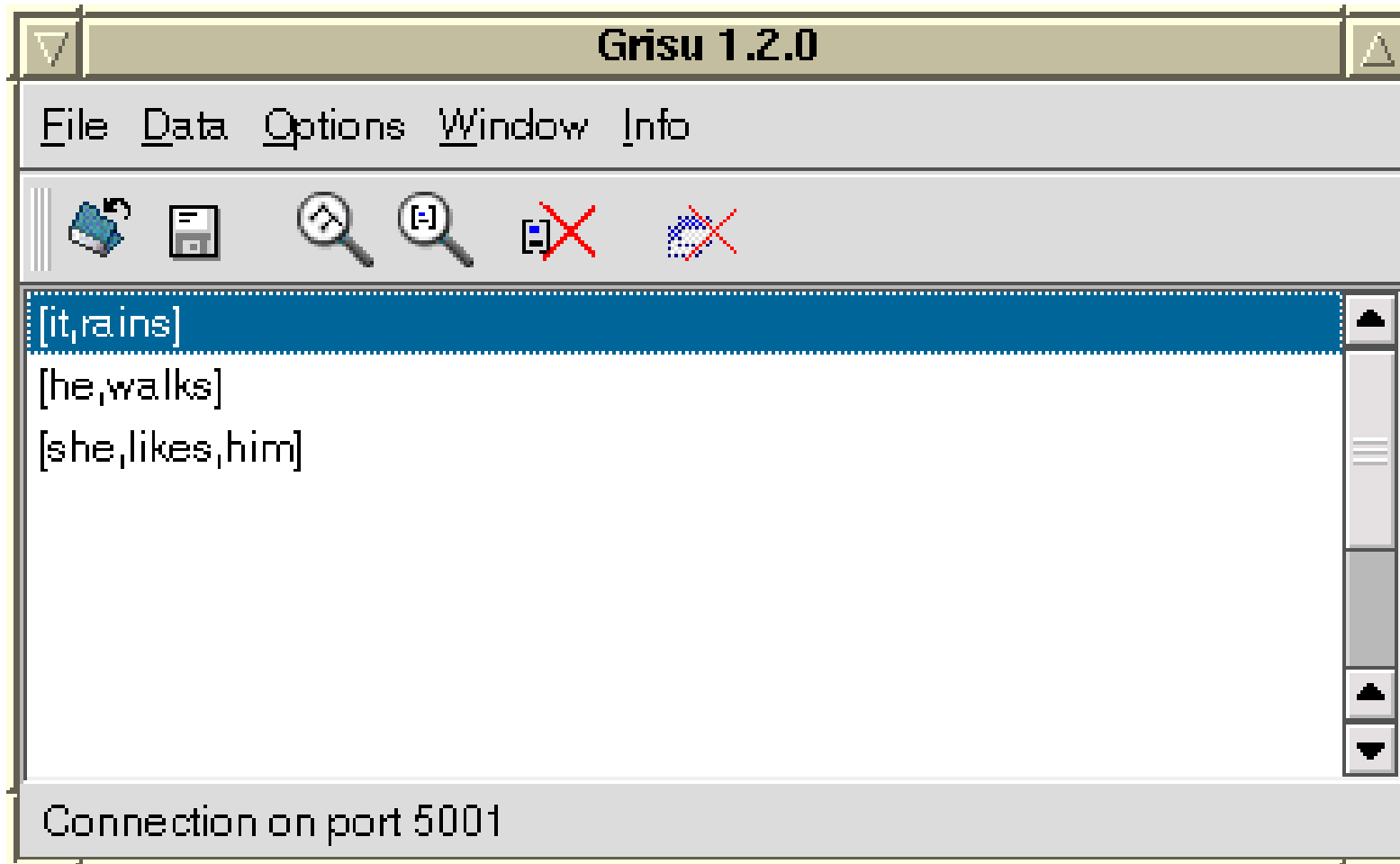
% Lexicon
she ~~> (word, cat:noun).

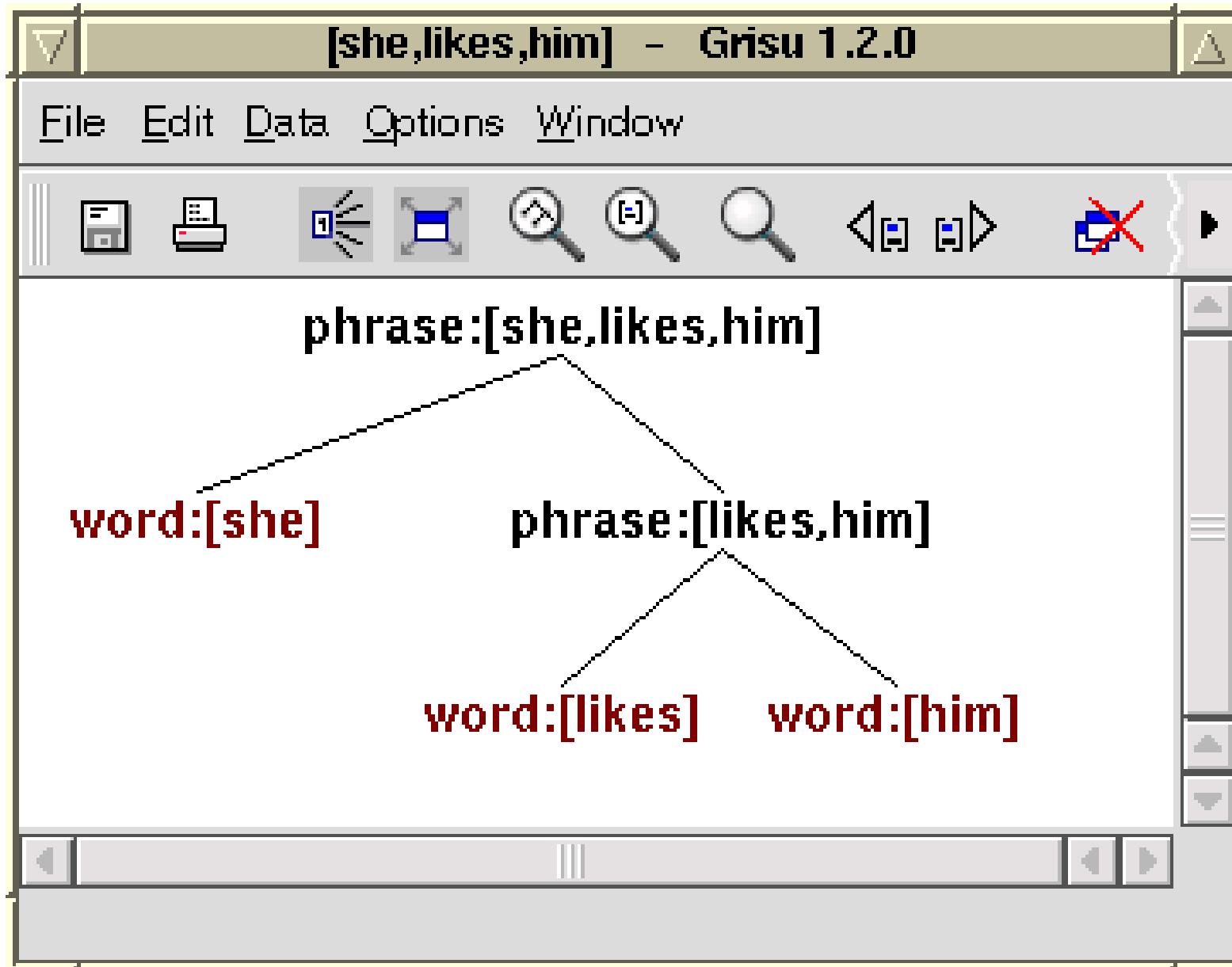
signs ~~> (word, cat:verb).

% Phrase Structure Rule
subj_head_rule ##
    phrase
    cat> (word,cat:noun),
    cat> (word,cat:verb).
```



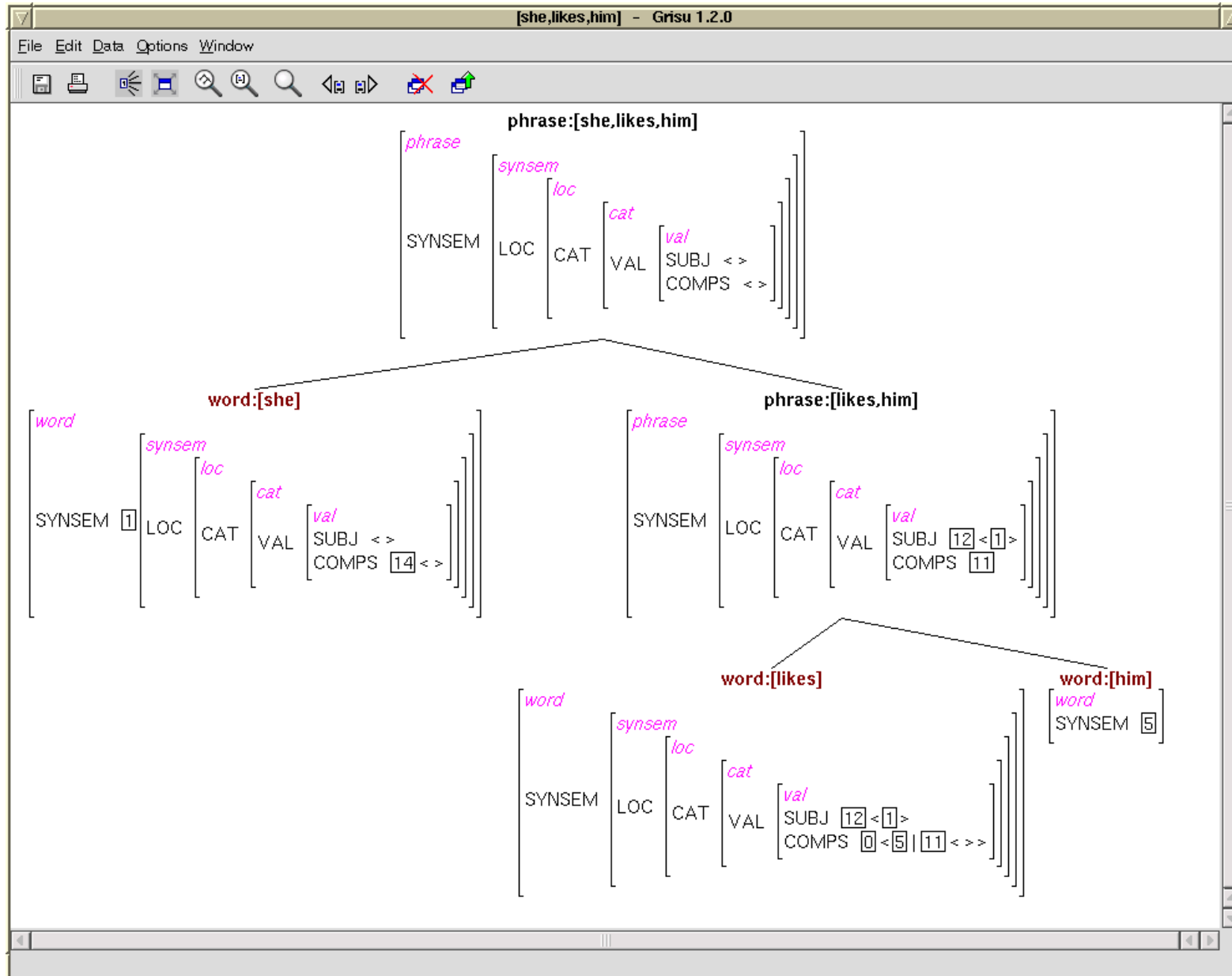
Grisu I





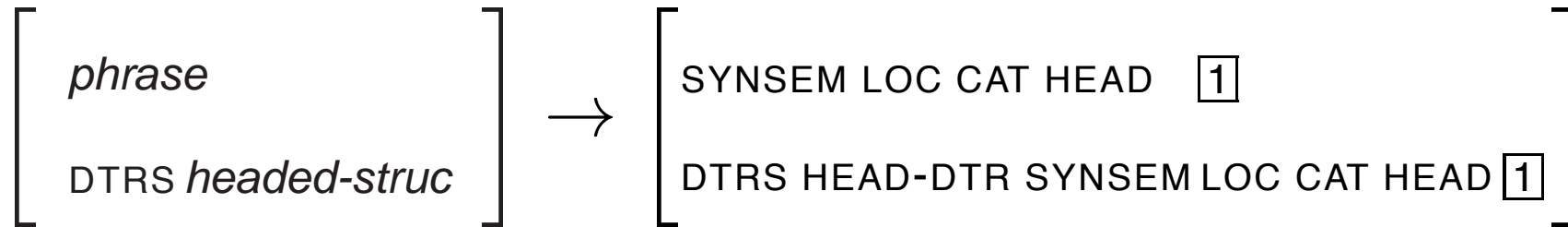


Grisu III





Ograniczenia implikacyjne



```
(phrase, dtrs:headed_struct) *>
(synsem:loc:cat:head:H,
 dtrs:head_dtr:synsem:loc:cat:head:H).
```




Sekwencje testowe

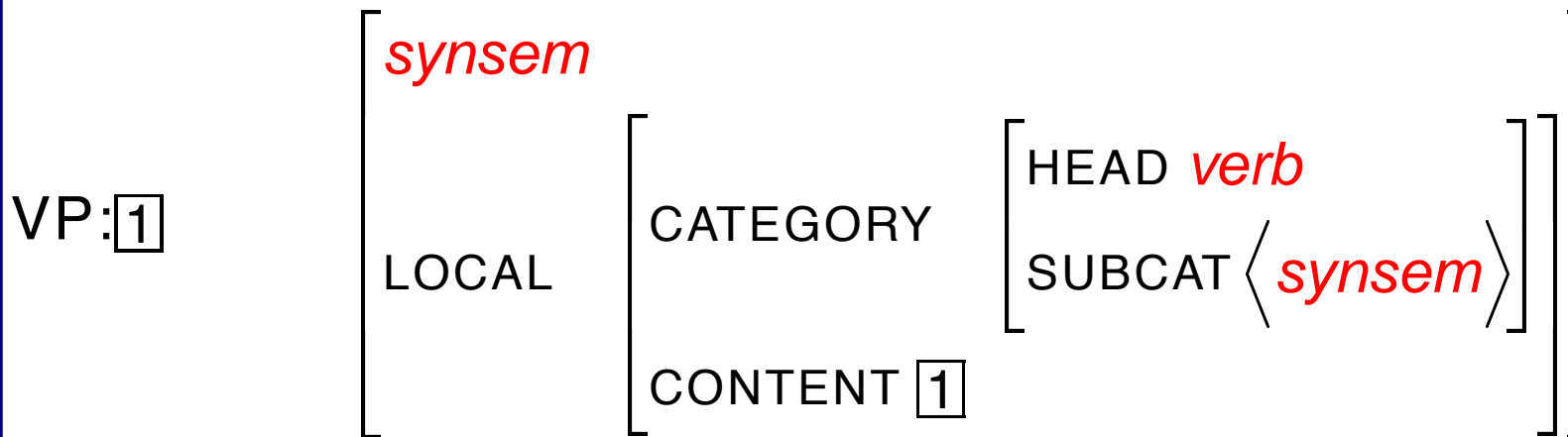
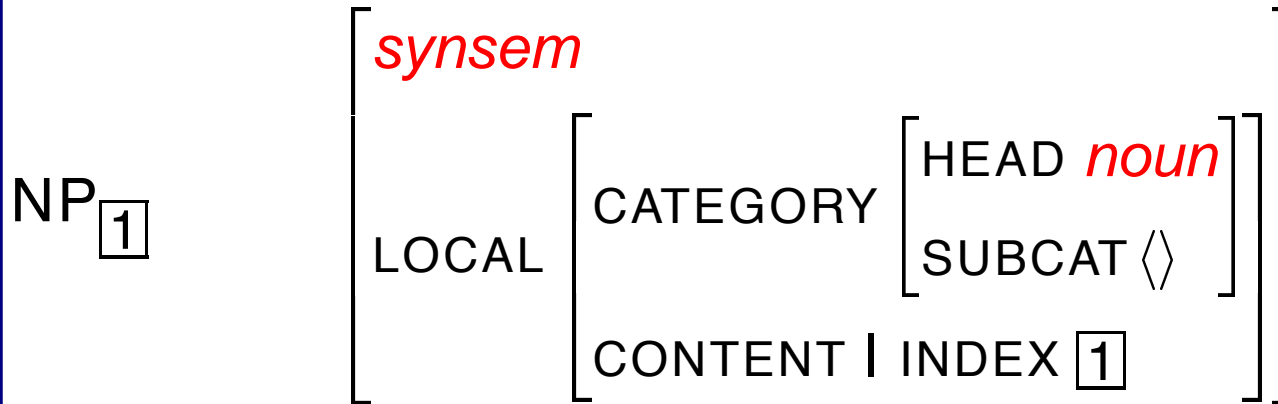
```
% Example test file test.pl
t(1,"ein dickes pferd",_,4,'all parses').
t(2,"ein dickes pferd",@noun_s,2,'nominal parses only').
t(3,"hat gelacht",@clause_s,0,_).
t(4,"das pferd lacht",_,2,_).

t(Nr,Item,(@noun_s,@case_s(nom),@sat_s,@no_slash_s),ExpSols,Comment) :-
    npt(Nr,Item,ExpSols,Comment).

npt(5,"drei buecher",1,_).
```



Skróty stosowane w HPSG





Makrodefinicje

Definicje:

```
np(Index-index) := (local:(category:(head:noun,  
                                     subcat:[]),  
                    content:index:Index)).
```

```
vp(Cont-cont) := (local:(category:(head:verb,  
                                   subcat:[synsem]),  
                    content:Cont)).
```

Zastosowanie:

```
she ~~> (word, synsem: @np((per:third,num:sing))).  
sings ~~> (word, synsem: (@vp((sing-rel, singer:Index)),  
                    local:category:subcat:head:@np(Index))).
```



Definicje:

```
append([],ne_list,L) if true.
```

```
append([H|T1],L,[H|T2]) if append(T1,L,T2).
```

```
synsem2sign(e_list,e_list) if true.
```

```
synsem2sign([H|T],[synsem:H|NewT]) if synsem2sign(T,NewT).
```

Zastosowanie:

```
phrase *> (synsem:category:subcat:PhrSubcat,  
           dtrs: (head_dtr:synsem:category:subcat:HeadSubcat,  
                 comp_dtrs:CompDtrs)
```

```
goal
```

```
    synsem2sign(CompSynsems,CompDtrs),
```

```
    append(CompSynsems,PhrSubcat,HeadSubcat).
```



Reguły leksykalne

```
% Present Participle Lexical Rule

prp_lex_rule ##
(word,
  synsem:loc:(cat:head:(vform:base,
                        aux:Aux,
                        pred:Pred),
                cont:Cont),
  arg_st:Arg)
**>
(synsem:loc:(cat:head:(vform:prp,
                        aux:Aux,
                        pred:Pred),
                cont:(cont_rel,
                       soa_arg:Cont)),
  arg_st:Arg)
morphs
be becomes being,
(X,[e]) becomes (X,ing),
X becomes (X,ing).
```



- Bob Carpenter and Gerald Penn (1999). ALE 3.2 User's Guide. Technical Memo Nr. CMU-LTI-99-MEMO. Carnegie Mellon Language Technologies Institute.
- Mohammad Haji-Abdolhosseini and Gerald Penn (2002). TRALE 1.0. User's Manual.
- Paul J. King (1989). *A Logical Formalism for Head-Driven Phrase Structure Grammar*. PhD thesis, University of Manchester.
- Gerald Penn, Detmar Meurers, Kordula De Kuthy, Mohammad Haji-Abdolhosseini, Vanessa Metcalf, and Holger Wunsch (2002). Trale Milca Environment v. 2.1.4. User's Manual (Draft).
- Carl J. Pollard and Ivan A. Sag (1987). *Information-Based Syntax and Semantics. Volume 1: Fundamentals*. Stanford: CSLI Publications.
- Carl J. Pollard and Ivan A. Sag (1994). *Head-Driven Phrase Structure Grammar*. Chicago: The University of Chicago Press.
- Frank Richter (2000). *A Mathematical Formalism for Linguistic Theories with an Application in Head-Driven Phrase Structure Grammar*. PhD thesis, Eberhard-Karls-Universität Tübingen, Version of April 28th, 2000.

oraz

materiały do zajęć *Grammar Formalisms and Parsing* prowadzonych na Eberhard-Karls-Universität w Tybindze przez Franka Richtera i Beatę Trawiński