Why mereology for the linearization task in NLG?

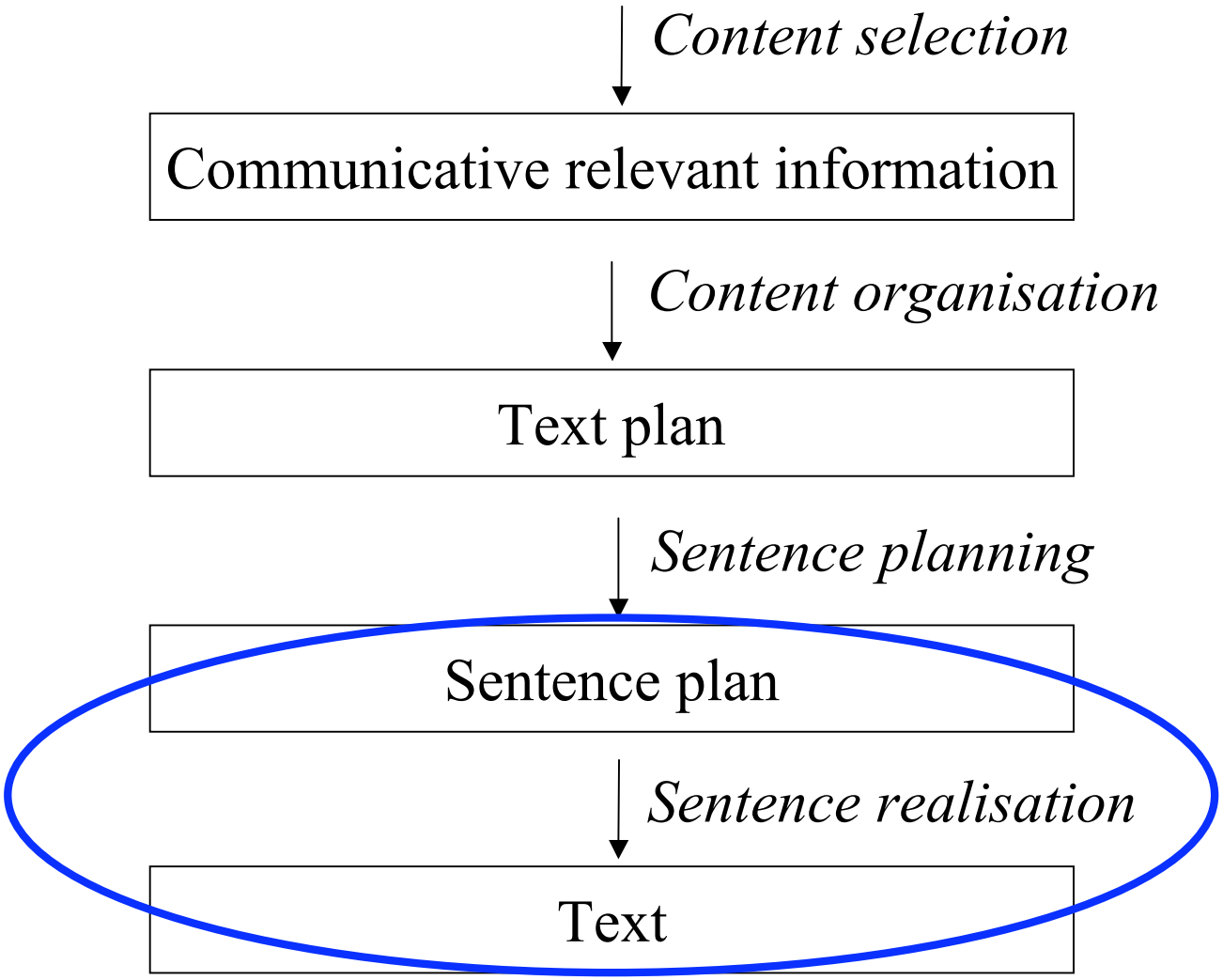
Ciprian Gerstenberger
University of Saarland

IGK 09. February 2006 Saarbrücken

Outline

1. Flexible NLG
2. Mereology
3. Linguistic Phenomena
4. Linearization
5. General Linearization Model
6. Worked Example
7. Conclusions

Natural Language Generation



Flexible Output Realisation

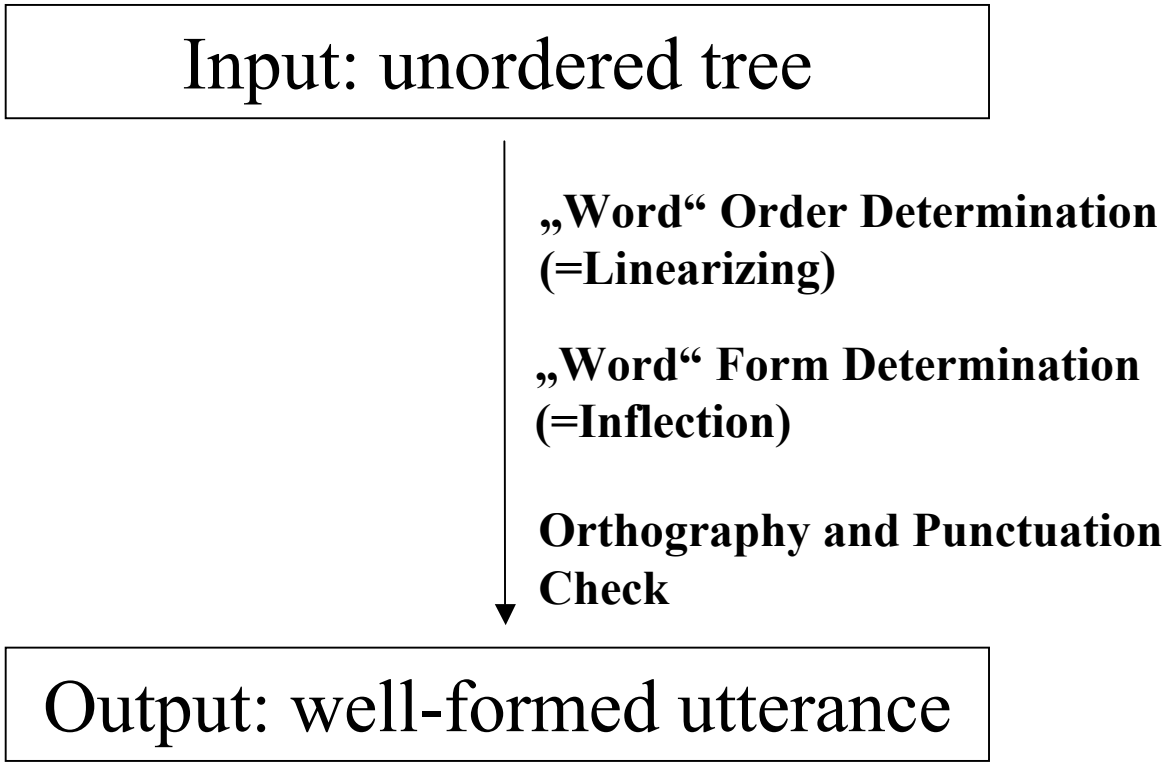
What does *flexibility* in sentence realization mean?

Ø Getting the word sequence which fits best the communicative situation in a given context!

Two parts of the task:

- **Providing the flexibility (General Linearization Model)**
- **Controlling the flexibility (To do!)**

Sentence Realisation



Mereology

Mereology (from the Greek μερος, 'part') is the theory of parthood relations:

- of the relations of part to whole
- of the relations of part to part within a whole

Phenomena - Psycholinguistics

1. **speech errors**
   - *multimodal output → mutlimodal ouptut*
   - *rote Türen → tote Rüren*
   - *snow flurries → flow snurries*
   - *self-destruct instruction → self-instruct destruction*
   - *writing a letter to my mother → writing a mother to my letter*
2. **spoonerisms**
   - *our deer old queen → our queer old dean*
   - *Kentucky Fried Chicken* → Mirco Nontschew läßt grüßen!
3. **children' secret languages**
   - *Was ist denn das? → Wabasibistdebenndabas?*

Phenomena - Comparative Linguistics

- **metathesis**
  - *brid* (Old English) vs. *bird* (modern English)
  - *brennen* (German) vs. *burn* (English)
  - *Born* (German) vs. *Brunnen* (German)/ *bron* (Dutch)
  - *Ross* (German) vs. *horse* (English)
  - *Warze* (German) vs. *wrat* (Dutch)
  - *Presse* (German)/ *press* (English) vs. *pers* (Dutch)
  - *Kreuz* (German) vs. *kors* (Norwegian)
  - *Roland* (French) vs. *Orlando* (Italian)
- **tmesis**
  - *whatsoever place* $\rightarrow$ *what place soever*
  - *absolutely* $\rightarrow$ *abso-bloody-lutely*
  - *fantastic* $\rightarrow$ *fan-f***ing-tastic*

Phenomena – Clitics in Romanian (1)

The indefinite article

- **un** *sat frumos* [a nice village]
- **un** *frumos sat*

The definite article

- *sat***ul** *frumos* [the nice village]
- *frumos***ul** *sat*

Phenomena – Clitics in Romanian (2)

The definite article

- *sat***ul** *frumos* [the nice village]
- *frumos***ul** *sat*
- *sat***ul** *meu frumos* [my nice village]
- *frumos***ul** *meu sat*
- *sat***ul** *frumos al meu*
- *frumos***ul** *sat al meu*
- *al meu sat frumos*
- *al meu frumos sat*

Phenomena – Clitics in Romanian (3)

Clitic clustering

- *mi* **l-***ai dat*
- *dă***-mi-l**
- *vedea***-le-aş** *mari şi sănătoase*
- *vedea***-ţi-le-aş** *mari şi sănătoase*
- *vedea***-ţi-i-aş** *mari şi sănătoşi*

A: *Copiii fetei mele sînt încă mici.*

B: *Vedea***-i-i-ai** *mari şi sănătoşi!*

Phenomena – Clitics in Polish (1)

The particle *śmy* [1st.Pl]

Q: *Zrobiliście to?*
[Did you do this?]

A1: *Nie zrobili***śmy** *tego.* (canonical)
[We didn't do this.]

A2: *(My) tego nie zrobili***śmy**. (canonical)

A3: *Tego (my) nie zrobili***śmy**.

A4: *Tego***śmy** *nie zrobili.*

A5: *My***śmy** *tego nie zrobili.*

A6: *Tego my***śmy** *nie zrobili.*

Phenomena – Clitics in Polish (2)

The particle *śmy* [1st.Pl]

Q: *Byliście tam pierwszego maja?*
[Were you there on the 1st of May?]

A1: *Pierwszego tam nie byli***śmy**.
[We were not there on the 1st of May.]

A2: *Pierwszego nie byli***śmy** *tam.*

A3: *Tam pierwszego nie byli***śmy**.

A4: *Pierwszego***śmy** *tam nie byli.*

A5: Pierwszego że**śmy** tam nie byli. (coll.)

Phenomena – Clitics in Polish (3)

The particle *śmy* [1st.Pl]

Q: *Widzieliście białego królika?*
[Did you see the white rabbit?]

A1: *Białego królika***śmy** *nie widzieli.*
[We didn't see the white rabbit.]

A2: *Białego***śmy** *królika nie widzieli.*

Phenomena – German (1)

- **compound coordination in German**
  - **be-** *und entladen*
  - **Schweins-** *und Kalbsbraten*
  - *Das ist nicht nur eine* **Geld-**, *sondern auch eine Platzfrage.*
  - *Originaltexte und* **-melodien** *sind nicht erhalten.*
- **particle verbs in German**
  - *Maria machte die Tür* **auf**.
  - **Auf** *machte Maria die Tür.*
  - *Maria wollte die Tür* **auf***machen.*
  - *Maria versuchte die Tür* **auf***zumachen.*

What about the orthography of particle verbs?

- *Maria möchte gerne radfahren* vs. *gerne Rad fahren*
- *Maria ist radgefahren* vs. *Rad gefahren*

Phenomena – German (2)

- partial fronting:
  - *Ein Buch hat Maria gelesen.*
- extraposition:
  - *Ein Buch hat Maria gelesen, das von ihm war.*
  - *Ich möchte eine Playlist erstellen mit drei Liedern.*
- split NPs:
  - *Rote Äpfel habe ich drei gegessen.*

Phenomena – German (3)

Macro-structure of Germanic languages

- capturing the generalization of TFM to restrict linearization when necessary
- coping with special phenomena
  - partial fronting
  - scrambling in the Mittelfeld
  - extraposition
  - multiple fronting

Phenomena – Polysynthetic Languages

An example from Western Greenlandic, a polysynthetic and agglutinating (but not incorporating) language:

Aliikusersuillammassuaanerartassagaluarpaalli.

aliiku-sersu-i-llammas-sua-a-nerar-ta-ssa-galuar-paal-li

entertainment-provide-SEMITRANS-one.good.at-COP-say.that-REP-FUT-sure.but-3plSUBJ/3sgOBJ-but

'However, they will say that he is a great entertainer, but . . . '

(12:1 morpheme-to-word ratio)

Source: http://encyclopedia.thefreedictionary.com/polysynthetic+language

Linearization

Linearization $\not\equiv$ Ordering Words

1. What about “sublexical ” phenomena?
2. What about “supralexical ” phenomena (discontinuous constituents, topological fields)?

Linearization

Was is a word?

Observations

The goal of NLG is natural language output

- just a string (be it phonological or graphical)
- we do NOT convey syntactic structure when speaking/writing
- we do NOT use especially designated graphical/acoustic signs for empty nodes/topological fields, ellipsis, traces, etc.

The General Linearization Model

The GML – a mereological model featuring

- **only one type of entities: Linear Order Part (LOP)**
- **two different types of relations holding between LOPs**
  - **Part-Of Relation**
  - **Linear Order Relation**
- **two different types of rules**
  - **PO-relating rules (mereological rules)**
  - **LO-relating rules (linear rules)**
    - **horizontal**
    - **vertical**
    - **diagonal**

GLM - Definitions

- **Linear Order Part**:

  A Linear Order Part is a phonologically realisable language item which has to be linearised as a contiguous part of a grammatically well-formed utterance.

- **Part-Of Relation**:

  A Part-Of relation holding between two different LOPs λ_1 and λ_2 ($\lambda_1 \sqsubseteq \lambda_2$) states that λ_1 is part of λ_2. The Part-Of relation is reflexive, anti-symmetric, and transitive.

- **Linear Order Relation**:

  A Linear Order relation holding between two different LOPs λ_1 and λ_2 ($\lambda_1 \prec_l \lambda_2$) states that λ_1 precedes λ_2. The Linear Order relation is irreflexive, asymetric, and transitive.

GLM - Examples of LOPs (1)

- a phoneme is a LOP (the smallest? – not necessarily!)
- a syllable is a LOP
- a morpheme is a LOP
- a word is a LOP
- different group of words are LOPs:
  - contiguous parts of contiguous constituents (***der rote*** *Apfel; der* ***rote Apfel***)
  - contiguous constituents (***der rote Apfel*** *ist schoen;* ***der rote Apfel, den Maria Hans gab,*** *ist schoen*)

Note: Contiguous constituents are LOPs not by virtue of being constituents but due to the fact that they have to be realized as a contiguous part of an utterance!

GLM - Examples of LOPs (2)

- different group of words are LOPs:
  - non-empty topological fields are LOPs (*Maria gab* ***Hans einen roten Apfel;*** *(dass) Maria* ***Hans einen roten Apfel*** *gab*)
  - whole (main/subordinate) clauses are LOPs (*Peter glaubte,* ***dass Maria Hans einen roten Apfel gab***; ***Dass Maria Hans einen roten Apfel gab***, *glaubte Peter*)
  - whole sentences are LOPs
  - ...

GLM - Properties (1)

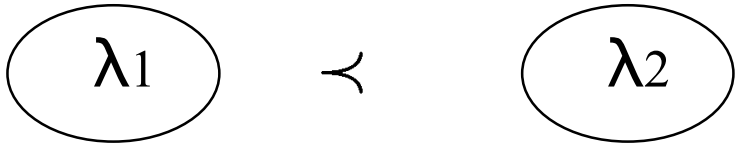
§ **Exclusivity**:

The Part-Of relation and the Linear Order relation are mutually exclusive, i.e., two different LOPs can either PO-relate or LO-relate but not both.

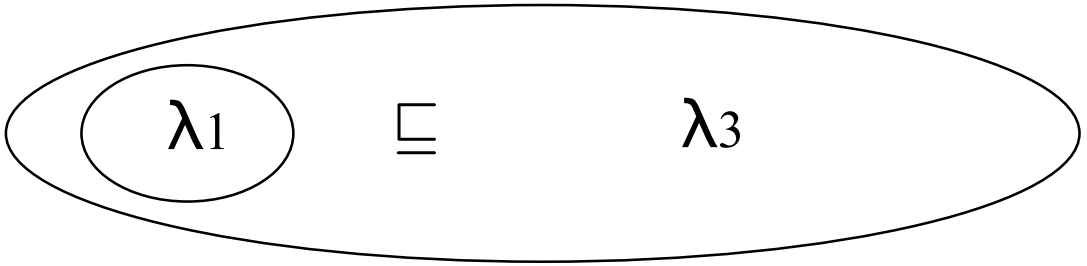
Let λ_1 and λ_2 be different LOPs:

1. if $\lambda_1 \sqsubseteq \lambda_2$ then $\lambda_1 \not\prec \lambda_2$
2. if $\lambda_1 \sqsubseteq \lambda_2$ then $\lambda_2 \not\prec \lambda_1$
3. if $\lambda_1 \prec \lambda_2$ then $\lambda_1 \not\sqsubseteq \lambda_2$
4. if $\lambda_1 \prec \lambda_2$ then $\lambda_2 \not\sqsubseteq \lambda_1$

GLM - Properties (2)



[das Buch] λ_1 [auf dem Tisch] λ_2



[[das Buch] λ_1 [auf dem Tisch] λ_2] λ_3

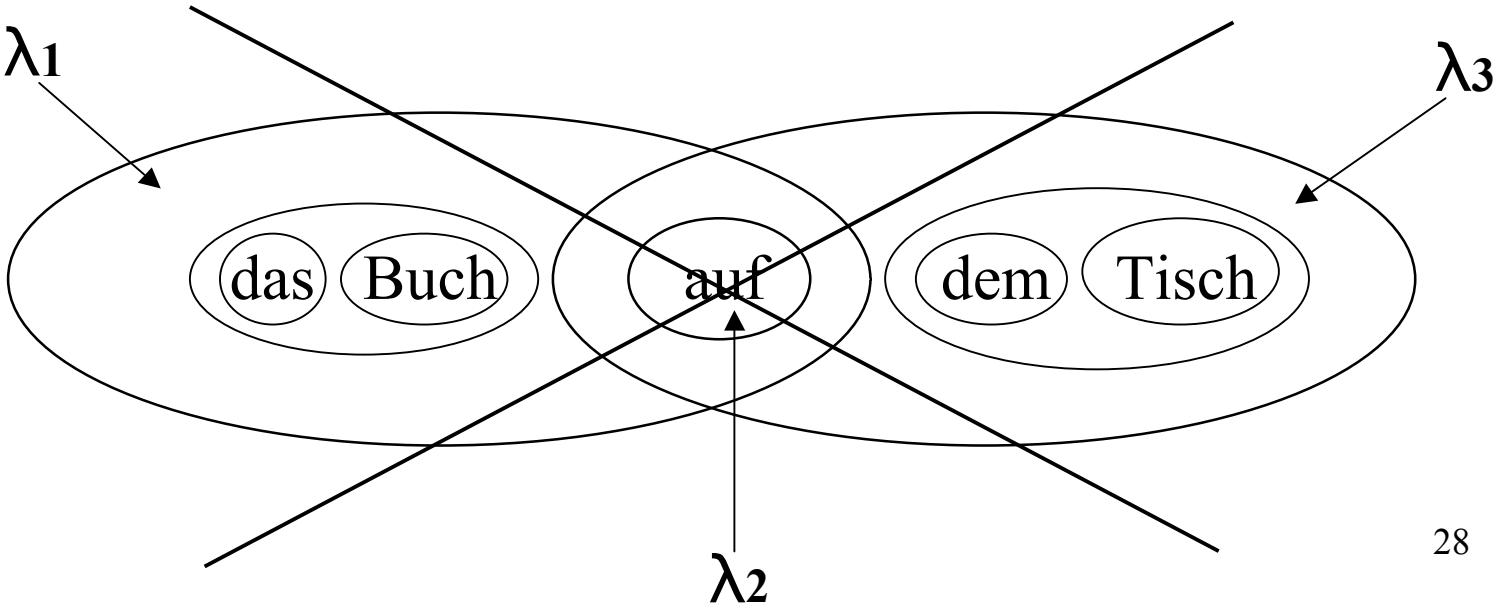
GLM - Properties (3)

§ **Non-Overlapping**:

Two different LOPs can not overlap.

Let λ_1, λ_2 and λ_3 be different LOPs:

if $\lambda_1 \sqsubseteq \lambda_2$ and $\lambda_2 \sqsubseteq \lambda_3$ then either $\lambda_1 \sqsubseteq \lambda_3$ or $\lambda_3 \sqsubseteq \lambda_1$



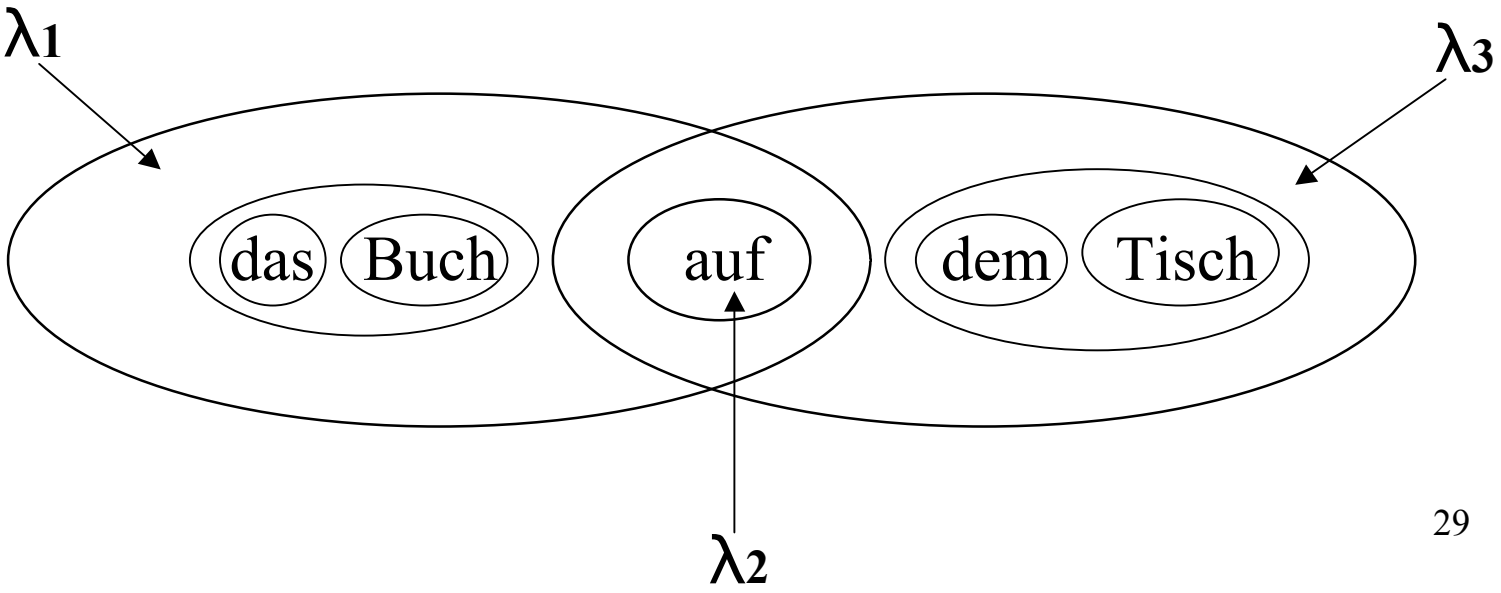
GLM - Properties (3)

§ **Non-Overlapping**:

Two different LOPs can not overlap.

Let λ_1, λ_2 and λ_3 be different LOPs:

if $\lambda \sqsubseteq \lambda_1$ and $\sqsubseteq_2 \lambda_3$ then either $\sqsubseteq \lambda_1$ $\lambda_3 \sqsubseteq_r \lambda_3$ λ_1



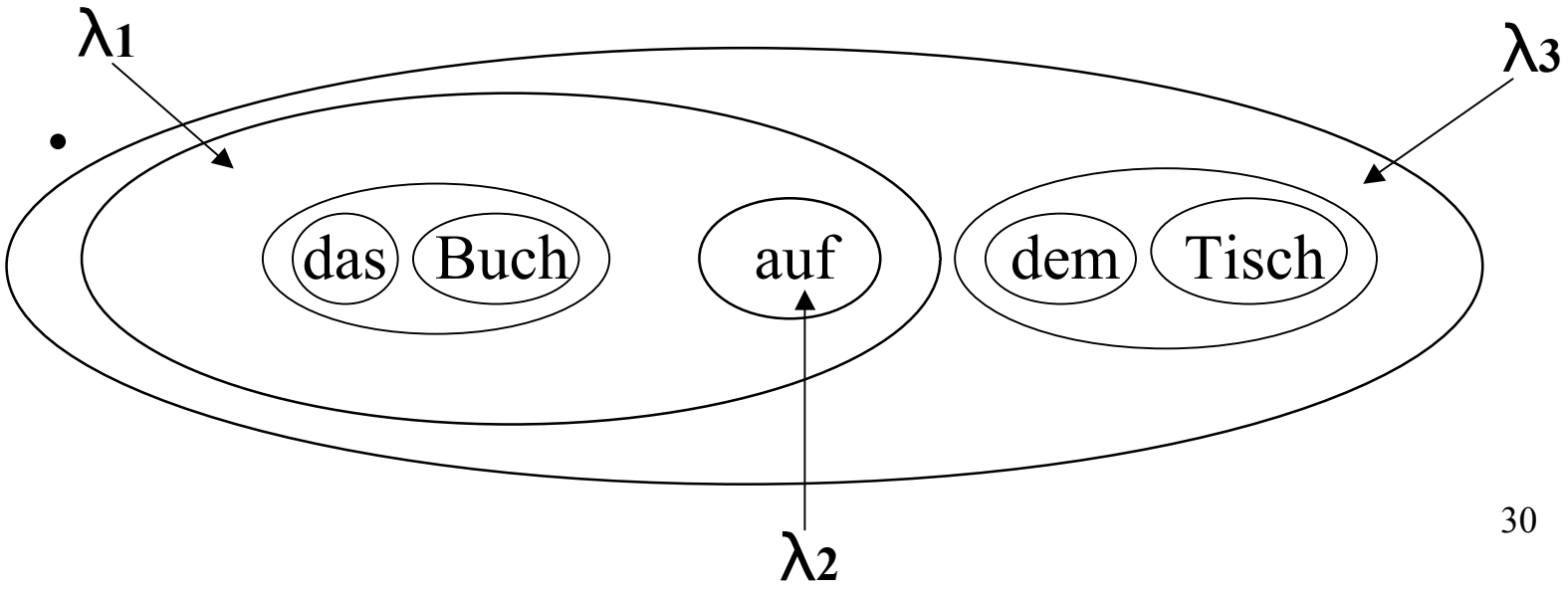
GLM - Properties (3)

§ **Non-Overlapping**:

Two different LOPs can not overlap.

Let λ_1, λ_2 and λ_3 be different LOPs:

if $\lambda_1 \sqsubseteq \lambda_2$ and $\lambda_2 \sqsubseteq \lambda_3$ then either $\lambda_1 \sqsubseteq \lambda_3$ or $\lambda_3 \sqsubseteq \lambda_1$



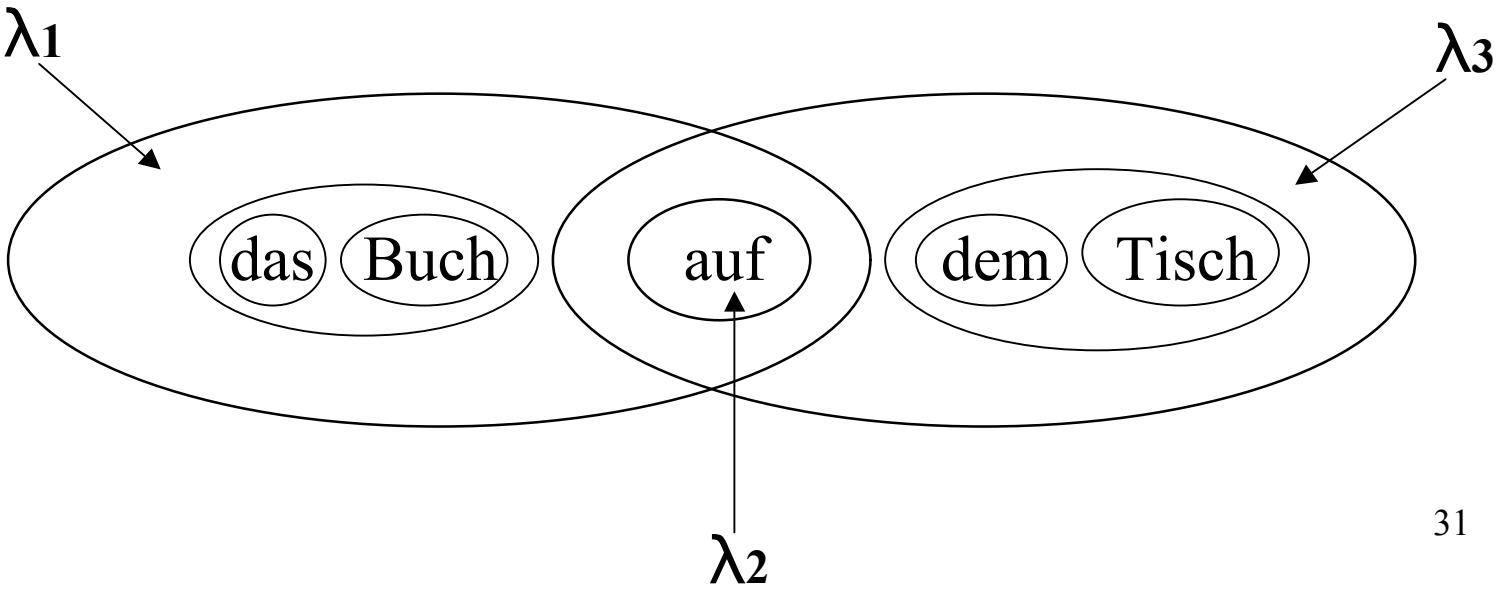
GLM - Properties (3)

§ **Non-Overlapping**:

Two different LOPs can not overlap.

Let λ_1, λ_2 and λ_3 be different LOPs:

if $\lambda_1 \sqsubseteq \lambda_2$ and $\lambda_2 \sqsubseteq \lambda_3$ then either $\lambda_1 \sqsubseteq \lambda_3$ or $\lambda_3 \sqsubseteq \lambda_1$



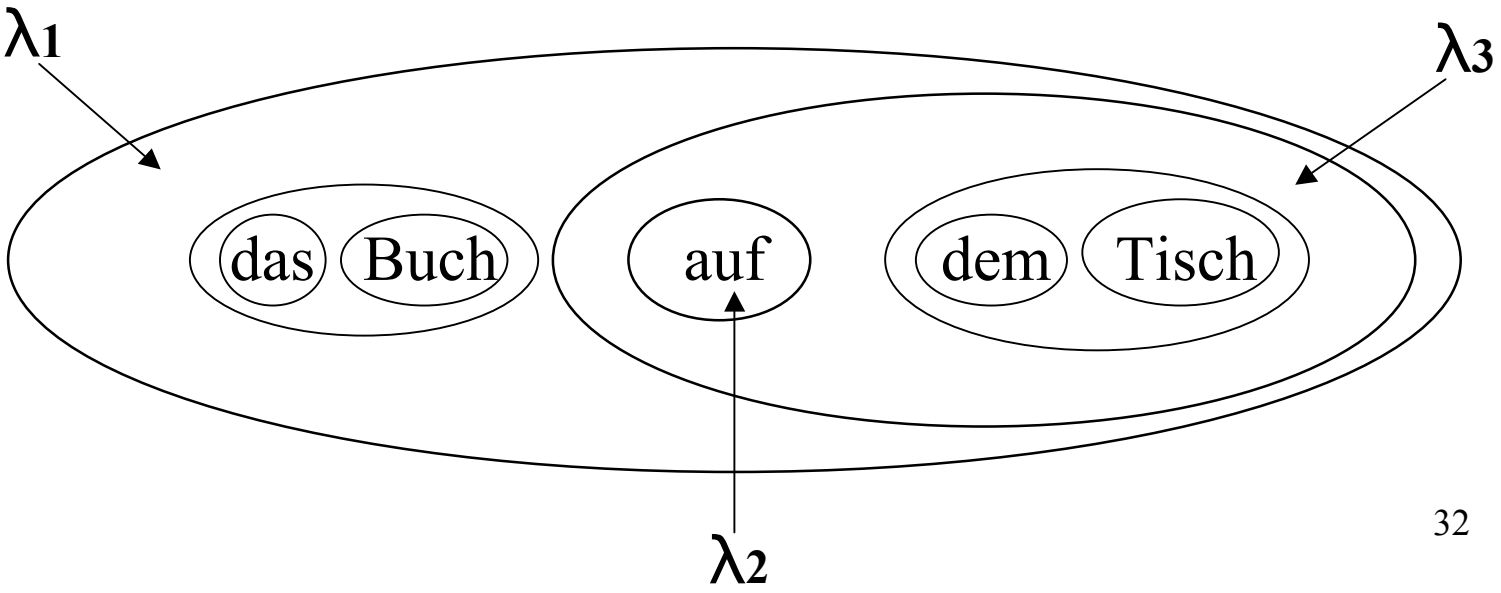
GLM - Properties (3)

§ **Non-Overlapping**:

Two different LOPs can not overlap.

Let λ_1, λ_2 and λ_3 be different LOPs:

if $\lambda_2 \sqsubseteq \lambda_1$ and $\lambda_2 \sqsubseteq \lambda_3$ then either $\lambda_1 \sqsubseteq \lambda_3$ or $\lambda_3 \sqsubseteq \lambda_1$



GLM - Corrolaries

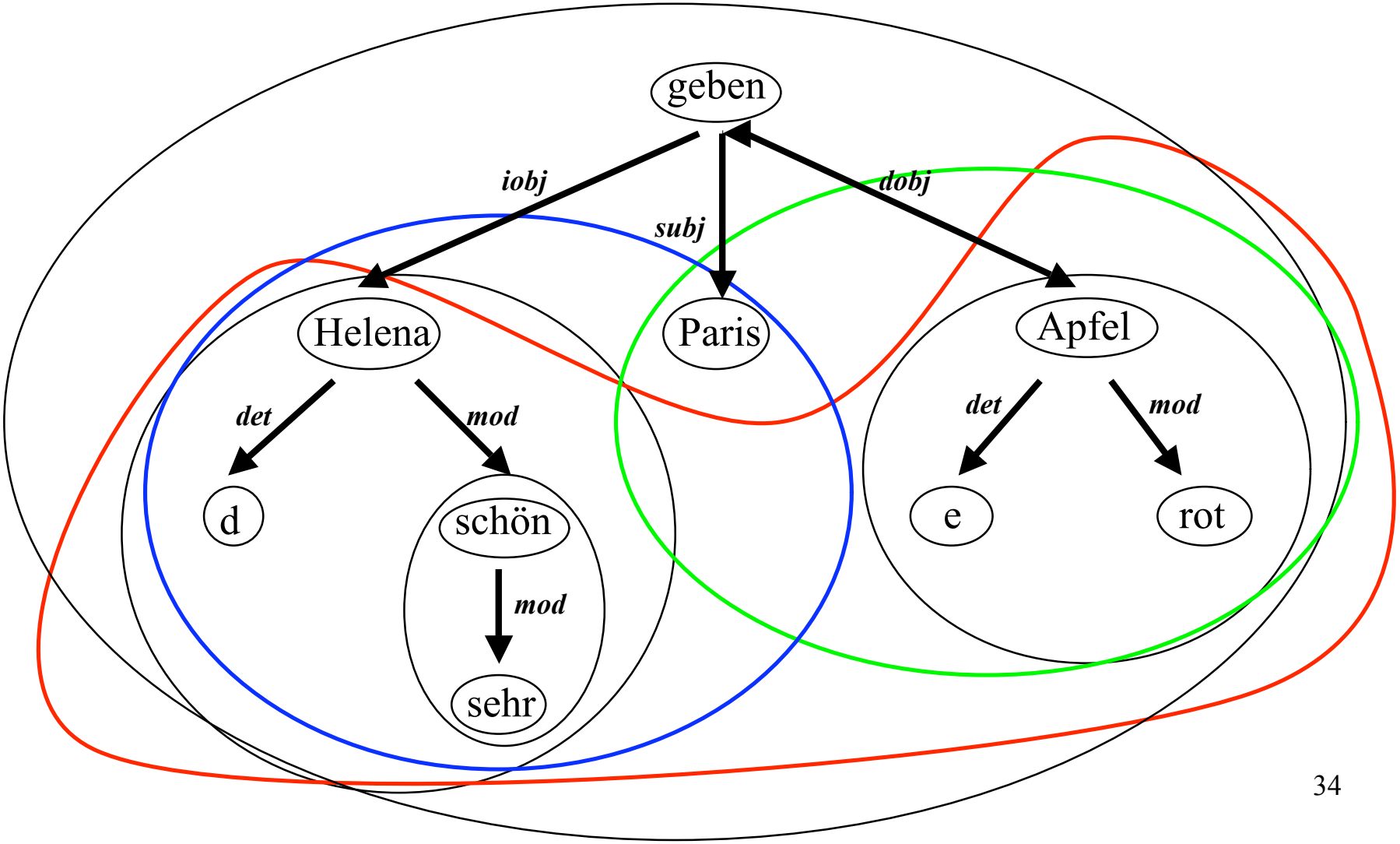
§ Let λ_1, λ_2, λ_3 and λ_4 be different LOPs, an$\prec$ λ_3 λ_4:

1. if $\lambda_1 \sqsubseteq \lambda_3$ then $\prec_1$ λ_4
2. if $\lambda_2 \sqsubseteq \lambda_4$ then $\prec_3$ λ_2
3. if $\lambda_1 \sqsubseteq \lambda_3$ and $\sqsubseteq_2$ λ_4 th$\prec$ λ_1 λ_2

§ Let λ_1, λ_2, λ_3 and λ_4 be different LOPs, an$\prec$ λ_1 λ_3:

1. if $\lambda_1 \sqsubseteq \lambda_3$ and $\not\sqsubseteq_2$ λ_3 th$\prec$ λ_3 λ_2
2. if $\lambda_2 \sqsubseteq \lambda_4$ and $\not\sqsubseteq$ λ_4 th$\prec$ λ_1 λ_4
3. if $\lambda_1 \sqsubseteq \lambda_3$, $\sqsubseteq_2$ λ_4, $\not\sqsubseteq \lambda_3$ λ_4, and $\not\sqsubseteq$ λ_4 λ_3 then λ_3 λ_4

GLM - Forming LOPs (1)



GLM - Forming LOPs (2)

§ LOP-forming rules

Rule Name: *AD[J|V]Modification_lop*

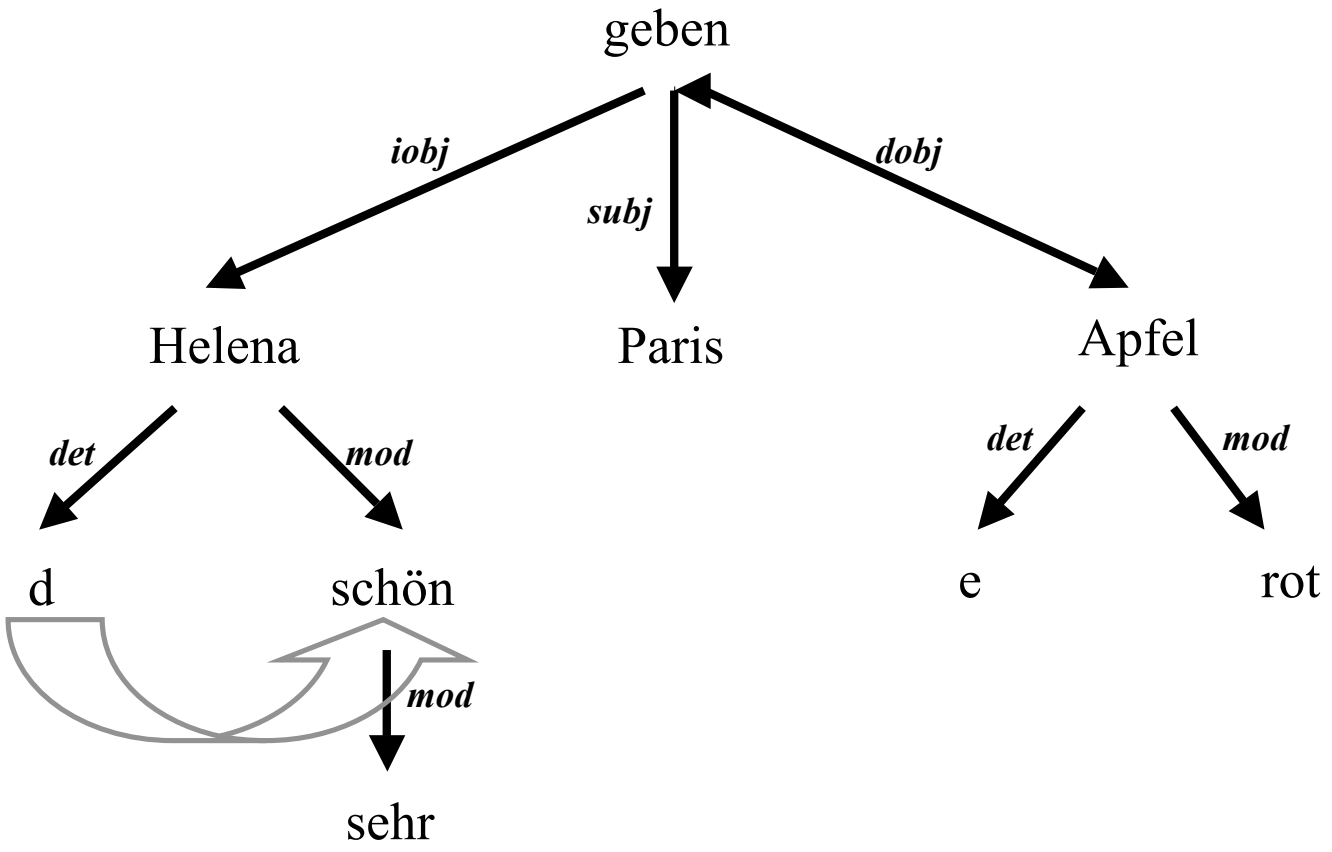
Condition Slot

```
X mod-> Y;
X.pos = ADJA | ADV;
```

Action Slot

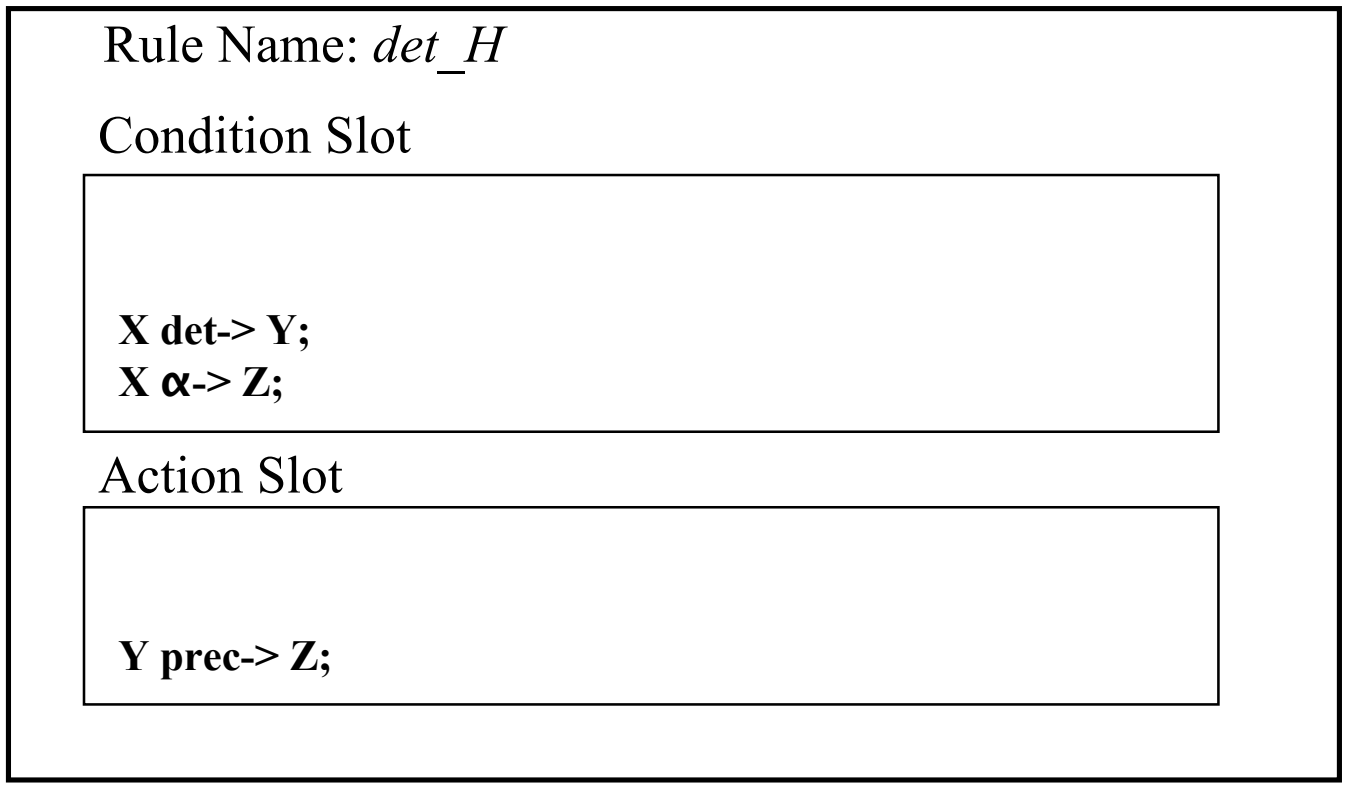
```
[X; Y] :::- lop007;
```

GLM - Linearising LOPs (1)

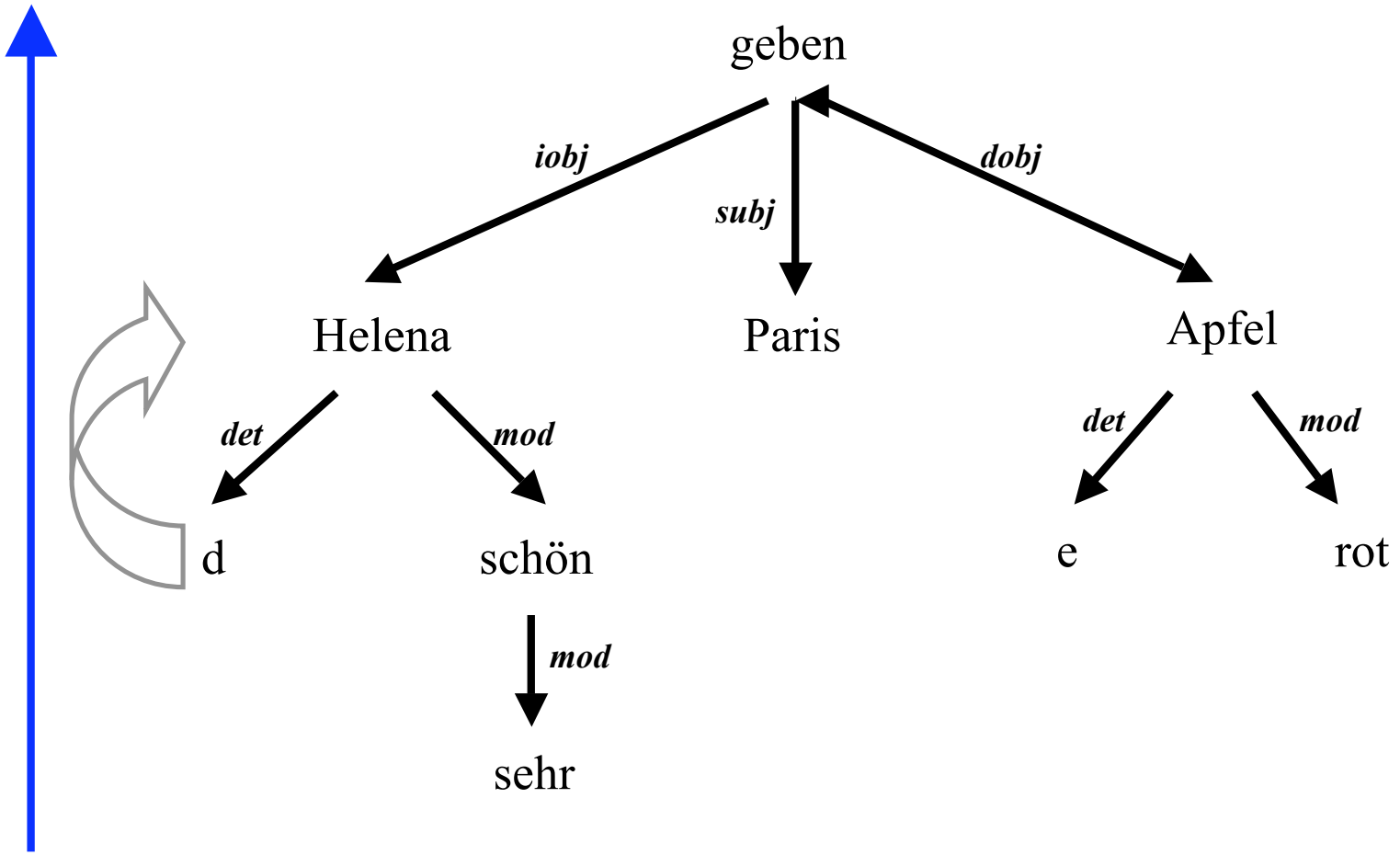


GLM - Linearising LOPs (2)

§ horizontal Linear Order rules



GLM - Linearising LOPs (3)



GLM - Linearising LOPs (4)

- vertical Linear Order rules

Rule Name: *det_V*

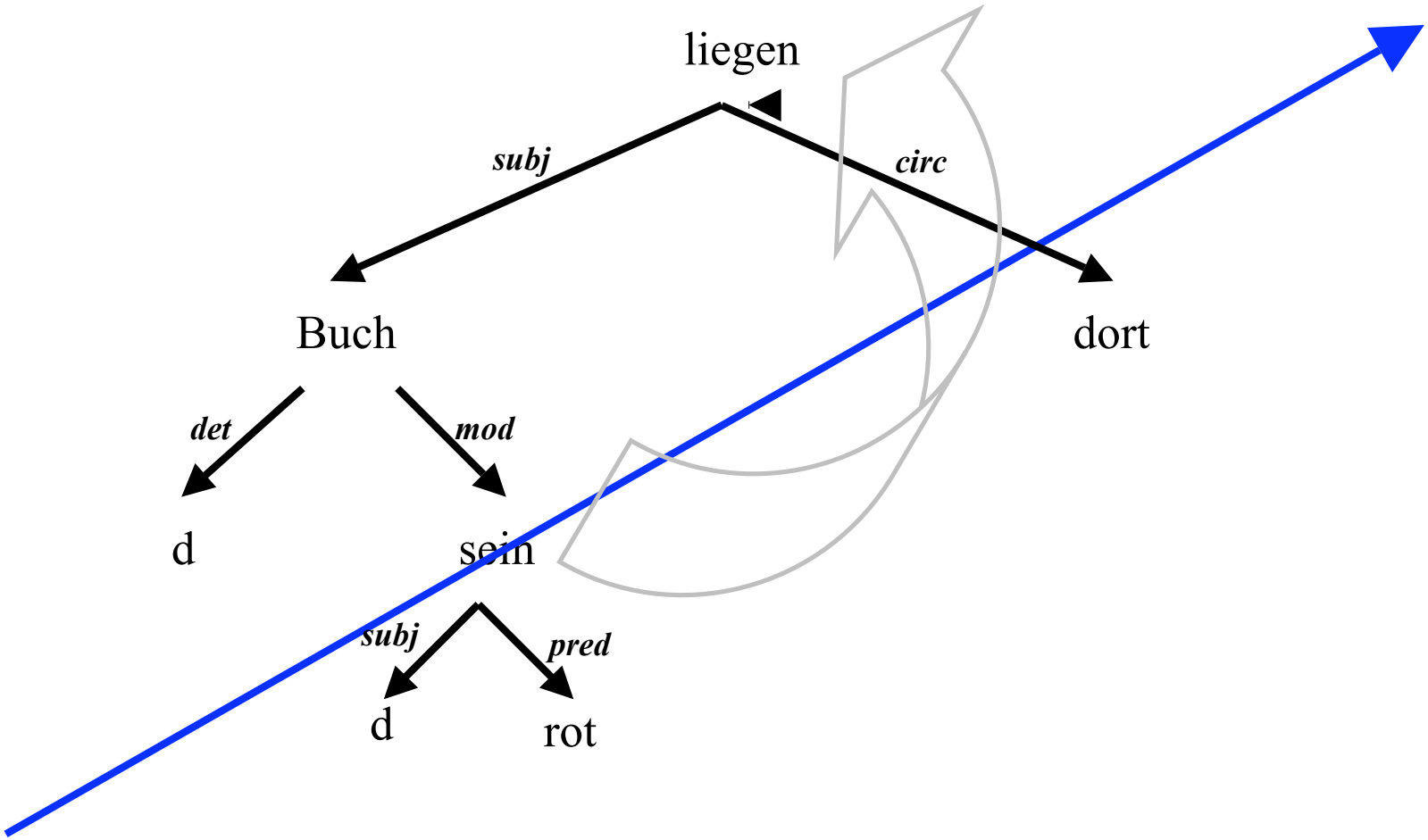
Condition Slot

X det-> Y;

Action Slot

Y prec-> X;

GLM: Linearising LOPs (5)



GLM - Linearising LOPs (6)

§ diagonal Linear Order rules

Rule Name: *relClauseVerb_D*

Condition Slot

X α -> Y; Y mod-> Z; Z β-> V; V.pos = PRELS;

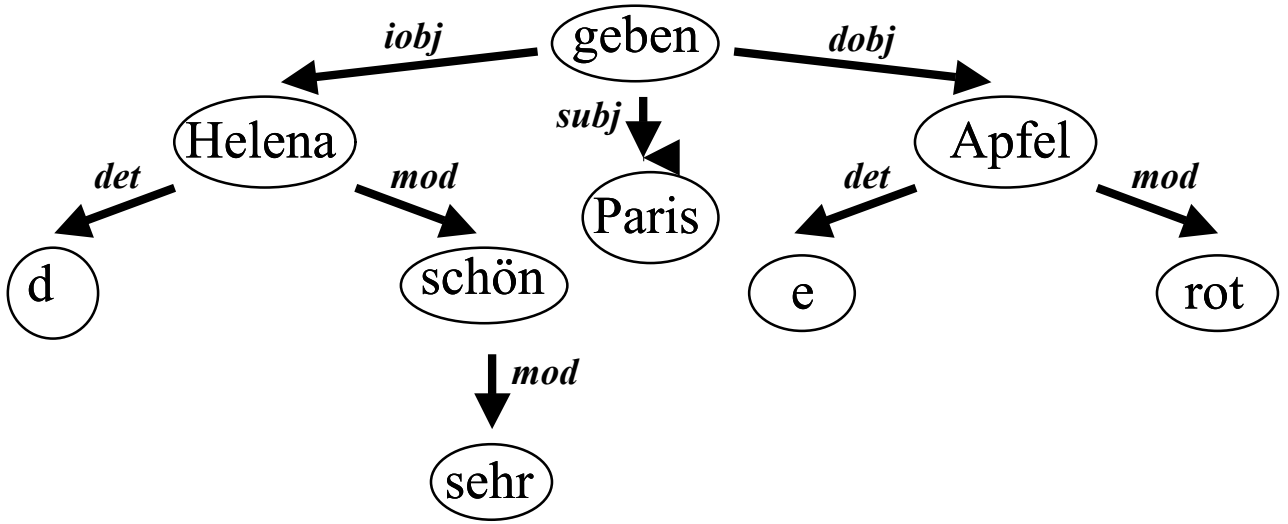
OUTPUT: **{X_lop prec-> Q_lop;**

Q_lop.gender ≠ V.gender; ...}

Action Slot

X prec-> V;

GLM - Worked Example (1)

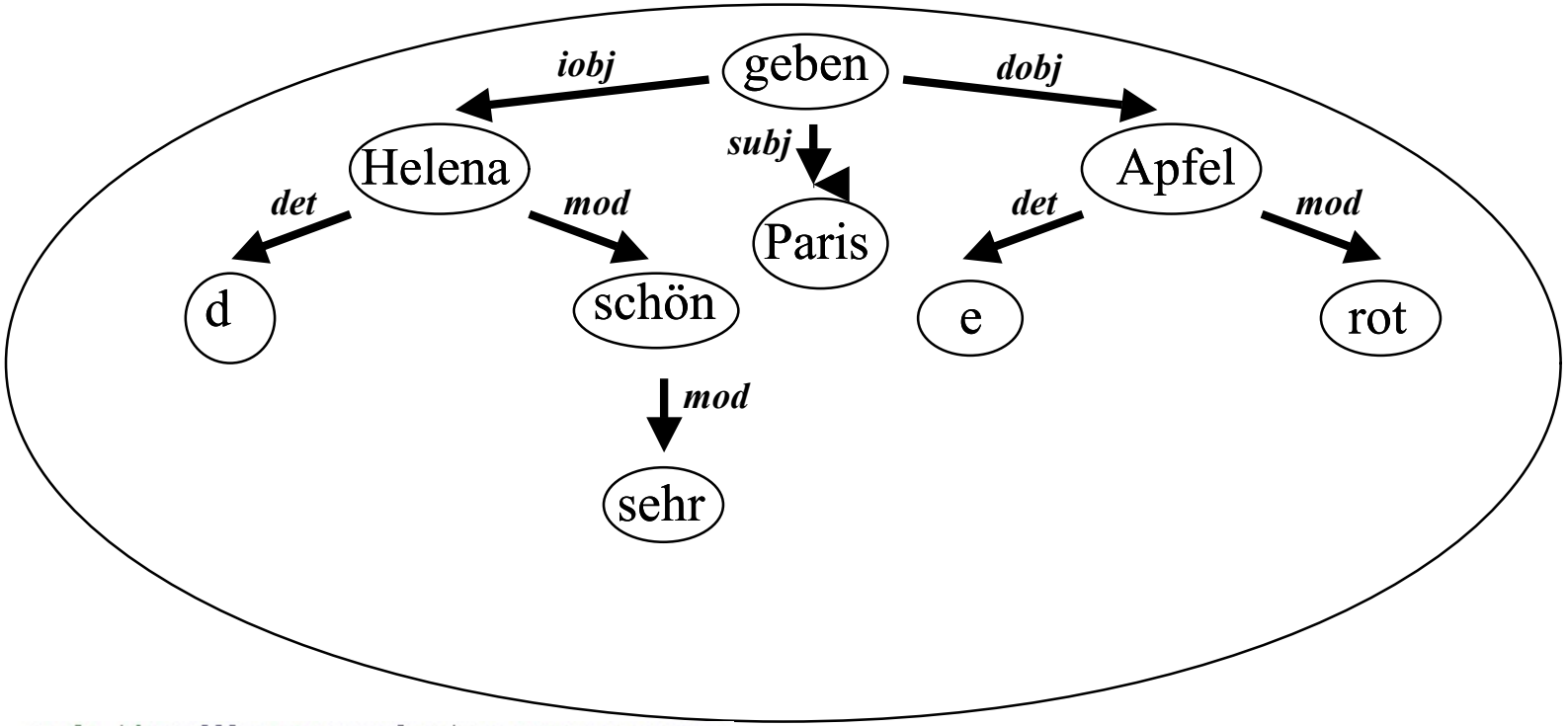



```
<rule id="rL001" type="LOPforming" name="nodeLOP">
  <conditions>
    <cond>X</cond>
  </conditions>
  <actions>
    <act>[X]::lopIDx</act>
  </actions>
</rule>
```

Legend:

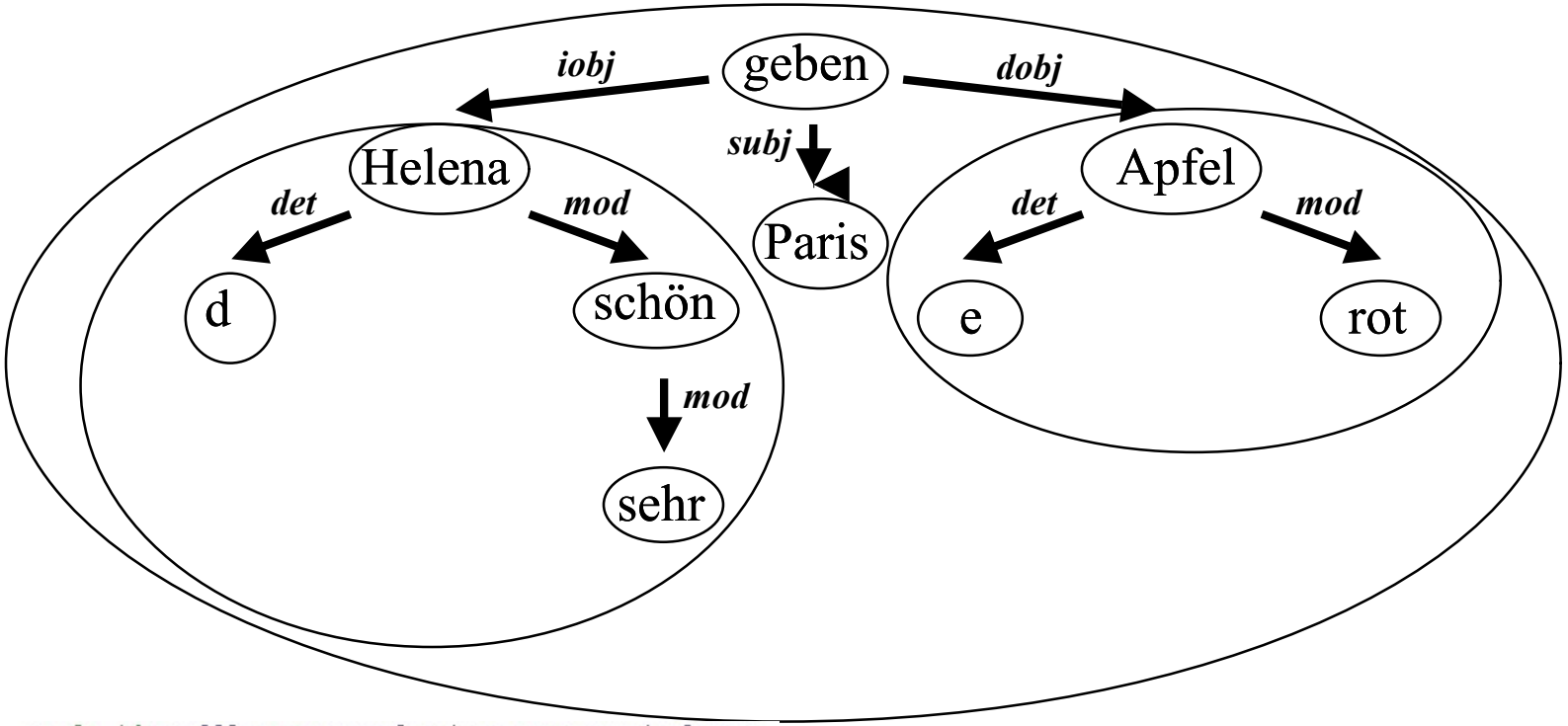
X	**node variable;**
-rName->	**immediate dominance;**
-*->	**immediate dominance;**
-*->>	**dominance relation;**
[…]	**Linear Order Part**
~>	**linear precedence;**

GLM - Worked Example (2)



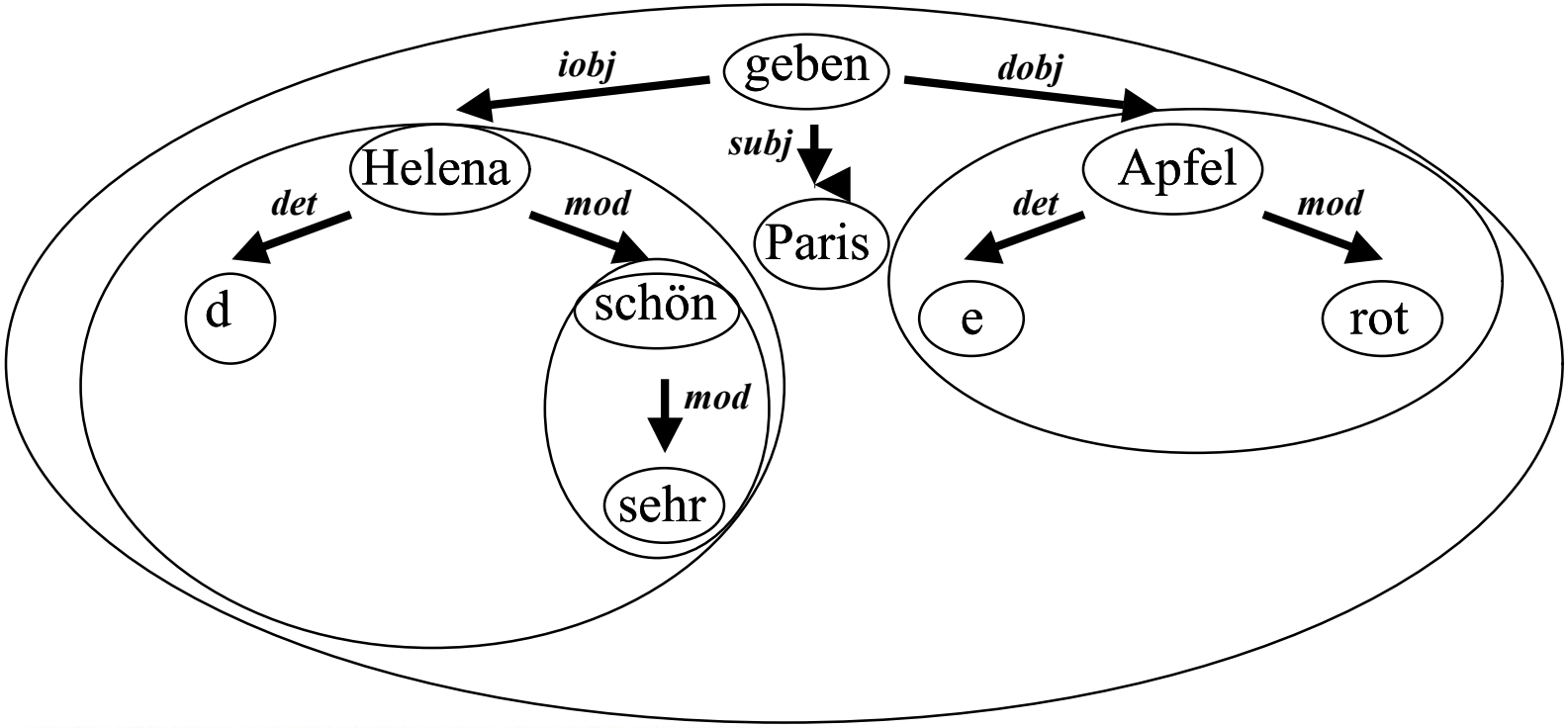

```
<rule id="rL002" type="LOPforming" name="treeLOP">
  <conditions>
    <cond>X-*->Y</cond>
    <cond>X=root</cond>
  </conditions>
  <actions>
    <act>[X;Y]::lopIDx</act>
  </actions>
</rule>
```

GLM - Worked Example (3)



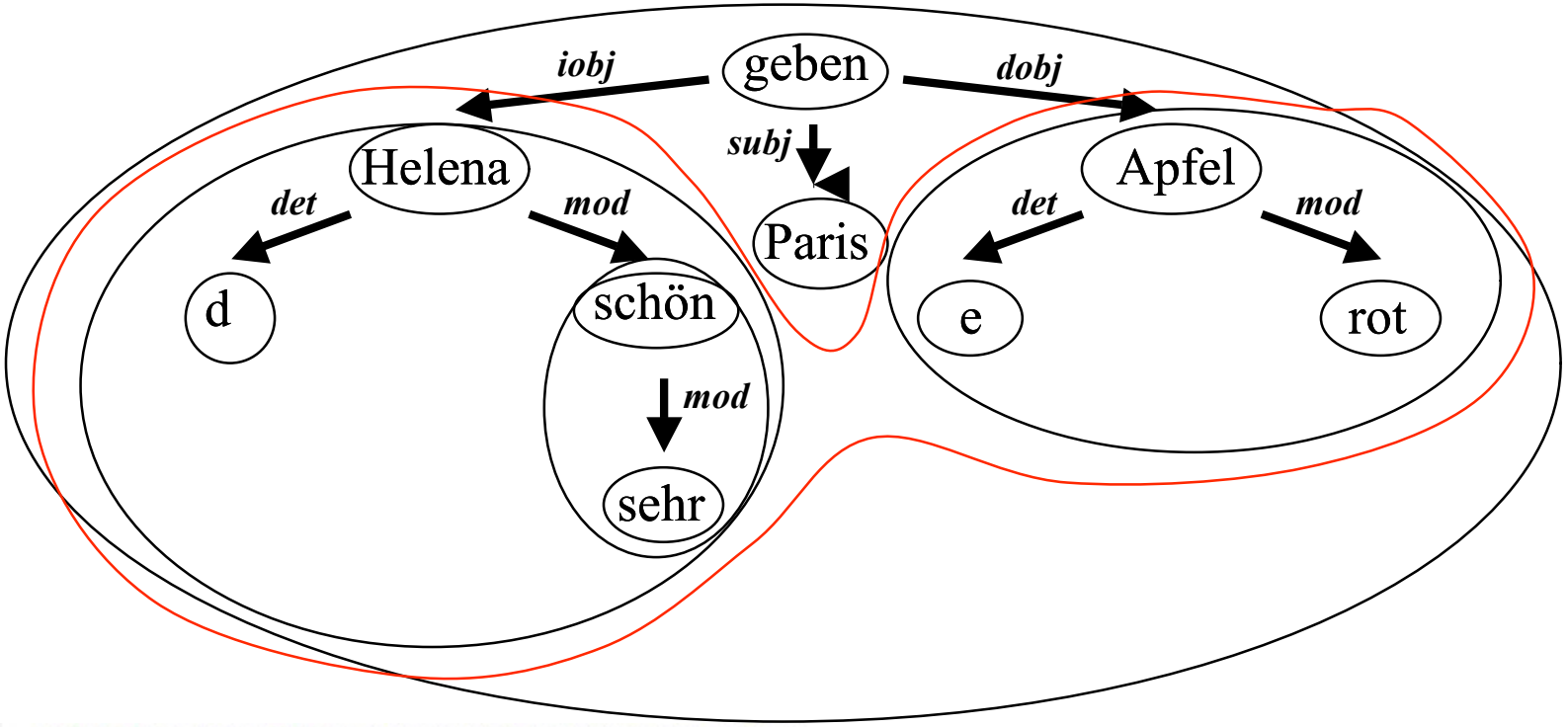

```
<rule id="rL003" type="LOPforming" name="nominalLOP">
  <conditions>
    <cond>X-*->Y</cond>
    <cond>X.pos={NN|PN}</cond>
    <cond>Y.extractable=no</cond>
  </conditions>
  <actions>
    <act>[X;Y]::lopIDx</act>
  </actions>
</rule>
```

GLM - Worked Example (4)



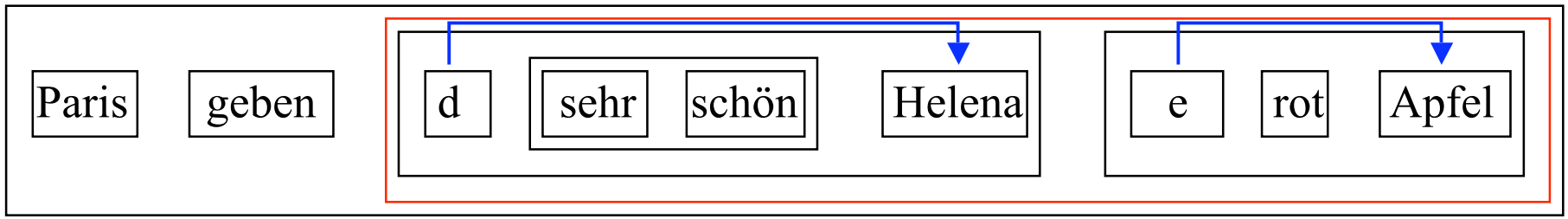

```
<rule id="rL004" type="LOPforming" name="modifierLOP">
  <conditions>
    <cond>X-mod->Y</cond>
    <cond>X.pos={ADJA|ADJD|ADV}</cond>
  </conditions>
  <actions>
    <act>[X,Y]::lopIDx</act>
  </actions>
</rule>
```

GLM - Worked Example (5)



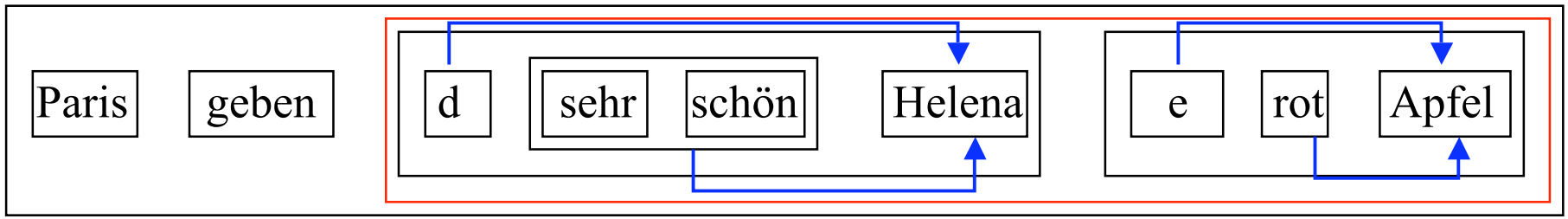

```
<rule id="rL005" type="LOPforming" name="mittelfeldLOP">
  <conditions>
    <cond>X-*->Y</cond>
    <cond>X=root</cond>
    <cond>Y.fronted=no</cond>
  </conditions>
  <actions>
    <act>[Y]::lopIDx</act>
  </actions>
</rule>
```

GLM - Worked Example (6)



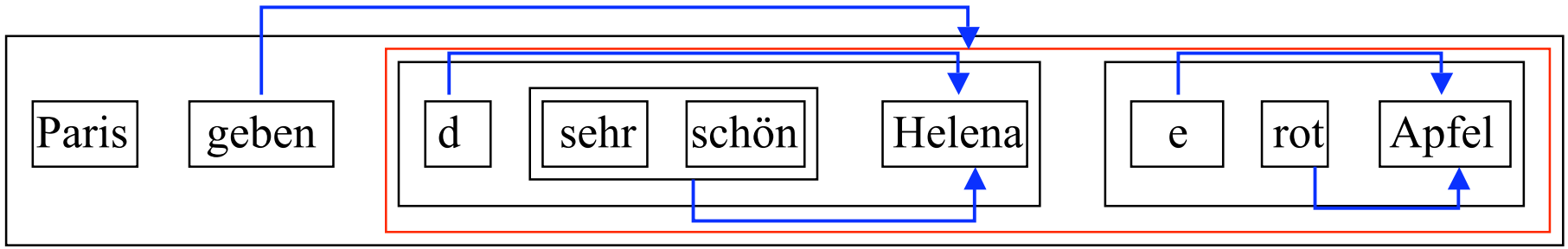

```
<rule id="rV001" type="vertical" name="determiner">
  <conditions>
    <cond>X-det->Y</cond>
  </conditions>
  <actions>
    <act>Y~>X</act>
  </actions>
</rule>
```

GLM - Worked Example (7)



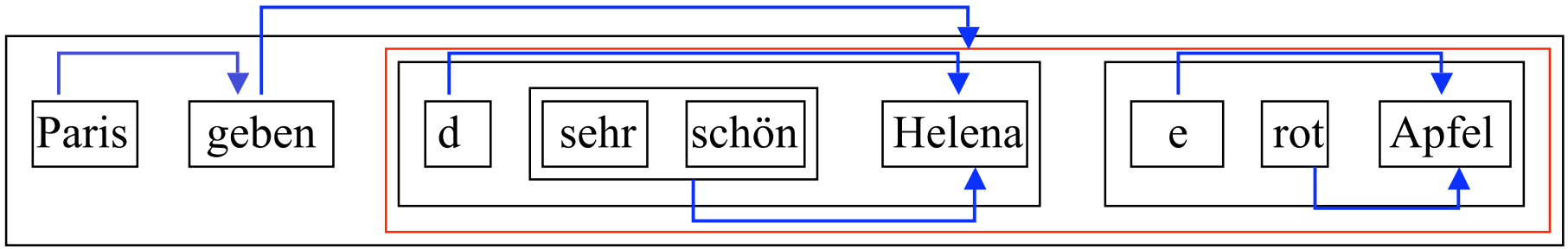

```
<rule id="rV002" type="vertical" name="nPremodifier">
  <conditions>
    <cond>X-mod->Y</cond>
    <cond>X.pos={NN|PN}</cond>
    <cond>Y.pos!=APPR</cond>
  </conditions>
  <actions>
    <act>Y~>X</act>
  </actions>
</rule>
```

GLM - Worked Example (8)



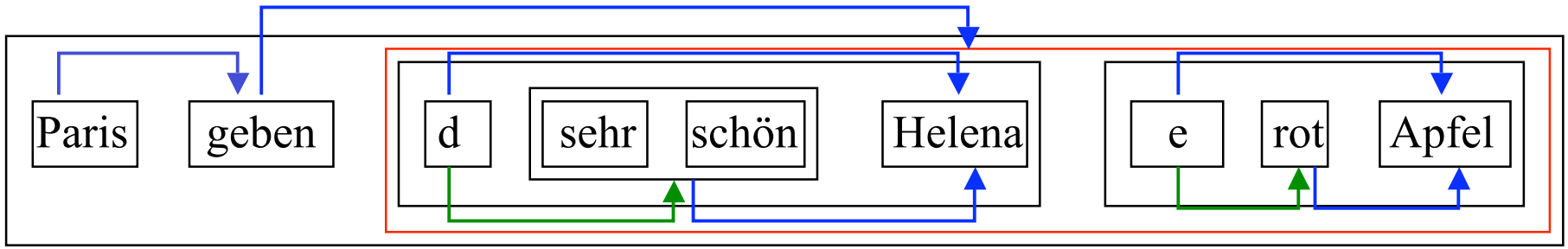

```
<rule id="rV003" type="vertical" name="mfRest">
  <conditions>
    <cond>X-*->Y</cond>
    <cond>X=root</cond>
    <cond>Y.fronted=no</cond>
  </conditions>
  <actions>
    <act>X~>Y</act>
  </actions>
</rule>
```

GLM - Worked Example (9)



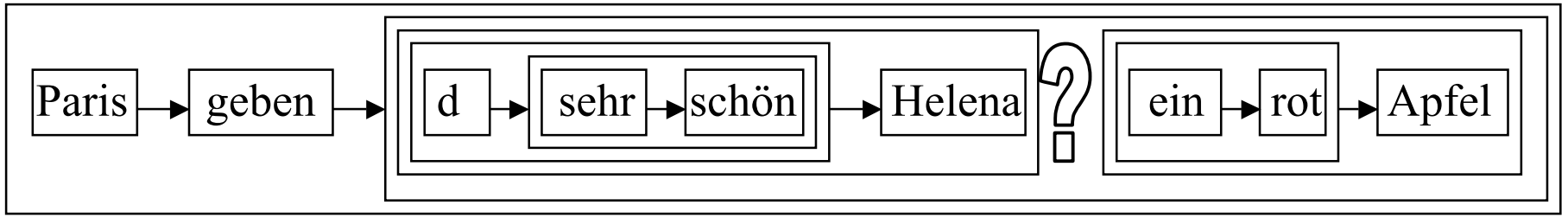

```
<rule id="rV004" type="vertical" name="fronting">
  <conditions>
    <cond>X-*->Y</cond>
    <cond>X-*->Z</cond>
    <cond>X=root</cond>
    <cond>X.cType=declarative</cond>
    <cond>Y.frontable=yes</cond>
    <cond>Y.fronted=yes</cond>
    <cond>Z.fronted=no</cond>
  </conditions>
  <actions>
    <act>Y~>X</act>
  </actions>
</rule>
```

GLM - Worked Example (10)




```
<rule id="rH001" type="horizontal" name="nLOP">
  <conditions>
    <cond>X-det->Y</cond>
    <cond>X-*->Z</cond>
    <cond>X.pos={NN|PN}</cond>
  </conditions>
  <actions>
    <act>Y~>Z</act>
  </actions>
</rule>
```

GLM - Worked Example (11)



Paris	gibt	der	sehr	schönen	Helena	einen	roten	Apfel
Paris	gibt	einen	roten	Apfel	der	sehr	schönen	Helena

Paris gibt der sehr schönen Helena einen roten Apfel.

Paris gibt einen roten Apfel der sehr schönen Helena.

GLM - Results

Paris gibt der sehr schönen Helena einen roten Apfel.

Paris gibt einen roten Apfel der sehr schönen Helena.

Der sehr schönen Helena gibt Paris einen roten Apfel.

Der sehr schönen Helena gibt einen roten Apfel Paris.

Einen roten Apfel gibt Paris der sehr schönen Helena.

Einen roten Apfel gibt der sehr schönen Helena Paris .

Questions

- What about Information Structural load of each individual utterance?
- What about prosody?
  - Ø Ranking the results wrt. context
  - ØDefining the interface between sentence realization and the previous steps of the NLG process

Tools

- **testing tools to implement the GLM for German/Polish/etc.**
  - **XDK** (XDG – Ralph Debusmann, Saarbrücken)
  - **MATE** (MTT - Bernd Bohnet, Stuttgart)
  - **DepLin** (MTT – Kim Gerdes, Bordeaux)
  - **OxiGen** (AMR - Nizar Habash, Maryland)
  - **PGW** (PG - Camiel van Breugel, Leiden)
  - **LKB** (HPSG – J. Carroll; A. Copestake, Standford)

Conclusions

§ GLM as utterance surface-oriented model:

- § describing utterances as mereological structures
- § treating different linguistic entities in a uniform way
- § accounting for context in an easy, flexible way
- § neutral with respect to syntactic theories
- § language-independent