Linguistics unfolded

Ralph Debusmann

Programming Systems Lab

Saarland University, Saarbrücken

Grammar formalisms

- should be able to modularize the different dimensions of linguistic description, i.e. to *unfold* a linguistic description,. . .
- . . . and at the same time, be able to treat the linguistic description in an integrative way.
- existing grammar formalisms are unable to do both
- we will introduce Extensible Dependency Grammar (XDG), which *can* do both

Why modularize?

- consider *Jeder Mann liebt eine Frau.*
- *jeder Mann* is the surface subject, *eine Frau* is the surface object
- *jeder Mann* is in the Vorfeld, *eine Frau* is in the Mittelfeld
- *jeder Mann* is the deep subject, *eine Frau* is the deep object
- generalizations:
  - surface subjects=Vorfeld
  - surface objects=Mittelfeld
  - surface subjects=deep subjects
  - surface objects=deep objects

Topicalization

- consider *Eine Frau liebt jeder Mann.*
- *jeder Mann* is the surface subject, *eine Frau* is the surface object
- *jeder Mann* is in the Mittelfeld, *eine Frau* is in the Vorfeld
- *jeder Mann* is the deep subject, *eine Frau* is the deep object
- generalizations:
  - surface subjects=Vorfeld
  - surface objects=Mittelfeld
  - surface subjects=deep subjects
  - surface objects=deep objects

Topicalization

- consider *Eine Frau liebt jeder Mann.*
- *jeder Mann* is the surface subject, *eine Frau* is the surface object
- *jeder Mann* is in the Mittelfeld, *eine Frau* is in the Vorfeld
- *jeder Mann* is the deep subject, *eine Frau* is the deep object
- generalizations:
  - surface subjects=Vorfeld
  - surface objects=Mittelfeld
  - surface subjects=deep subjects
  - surface objects=deep objects

Topicalization

- consider *Eine Frau liebt jeder Mann.*
- *jeder Mann* is the surface subject, *eine Frau* is the surface object
- *jeder Mann* is in the Mittelfeld, *eine Frau* is in the Vorfeld
- *jeder Mann* is the deep subject, *eine Frau* is the deep object
- generalizations:
  - ~~surface subjects=Vorfeld~~
  - ~~surface objects=Mittelfeld~~
  - surface subjects=deep subjects
  - surface objects=deep objects

Passive

- consider *Von jedem Mann wird eine Frau geliebt.*
- *von jedem Mann* is the surface object, *eine Frau* is the surface subject
- *von jedem Mann* is in the Vorfeld, *eine Frau* is in the Mittelfeld
- *jedem Mann* is the deep subject, *eine Frau* is the deep object
- generalizations:
  - ~~surface subjects=Vorfeld~~
  - ~~surface objects=Mittelfeld~~
  - surface subjects=deep subjects
  - surface objects=deep objects

Passive

- consider *Von jedem Mann wird eine Frau geliebt.*
- *von jedem Mann* is the surface object, *eine Frau* is the surface subject
- *von jedem Mann* is in the Vorfeld, *eine Frau* is in the Mittelfeld
- *jedem Mann* is the deep subject, *eine Frau* is the deep object
- generalizations:
  - ~~surface subjects=Vorfeld~~
  - ~~surface objects=Mittelfeld~~
  - surface subjects=deep subjects
  - surface objects=deep objects

Passive

- consider *Von jedem Mann wird eine Frau geliebt.*
- *von jedem Mann* is the surface object, *eine Frau* is the surface subject
- *von jedem Mann* is in the Vorfeld, *eine Frau* is in the Mittelfeld
- *jedem Mann* is the deep subject, *eine Frau* is the deep object
- generalizations:
  - ~~surface subjects=Vorfeld~~
  - ~~surface objects=Mittelfeld~~
  - ~~surface subjects=deep subjects~~
  - ~~surface objects=deep objects~~

Modularize!

- simple generalizations will backfire when confronted with more complicated constructions
- so don’t make them
- instead, properly modularize the different dimensions of linguistic description

Why integrate?

- consider *Peter sieht die Frau mit dem Teleskop*
- PP can either attach to *sieht* or to *Frau*
- additional information needed for disambiguation
- information may come from any source, i.e. also from semantics (e.g. a database of world knowledge)

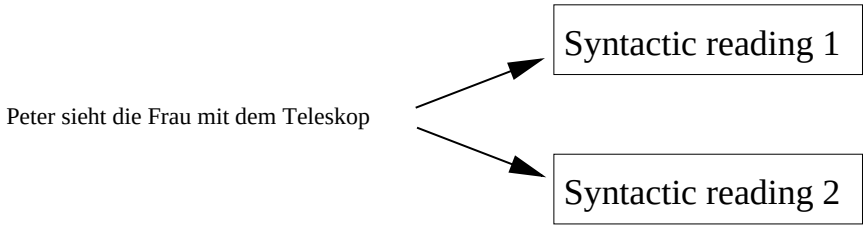
Non-integrated

- a *non-integrated* system cannot make use of information from semantics early on:

Peter sieht die Frau mit dem Teleskop

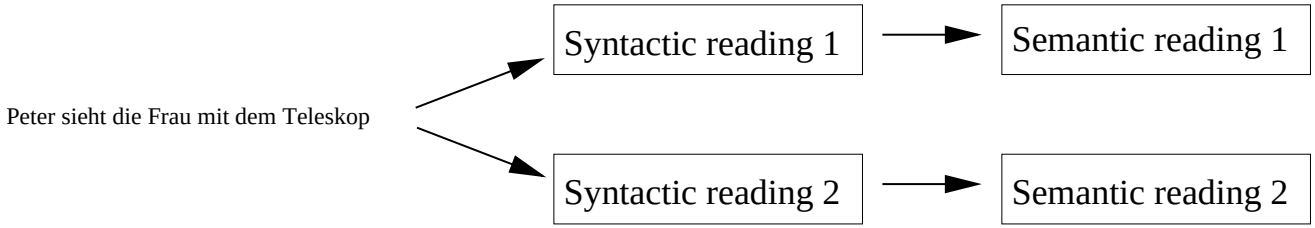
Non-integrated

- a *non-integrated* system cannot make use of information from semantics early on:



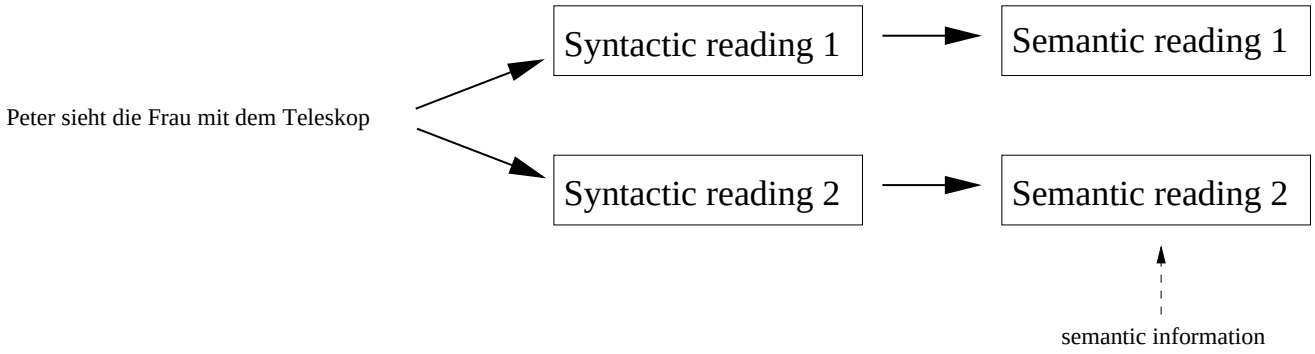
Non-integrated

- a *non-integrated* system cannot make use of information from semantics early on:



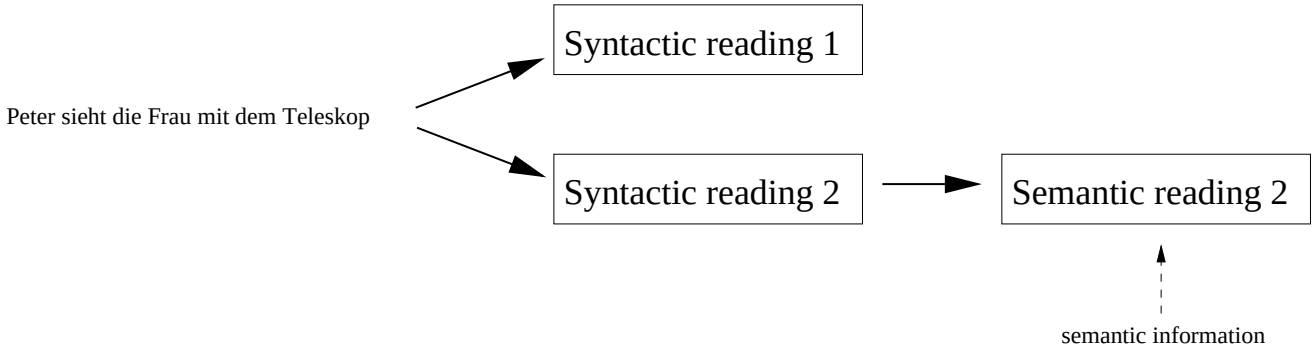
Non-integrated

- a *non-integrated* system cannot make use of information from semantics early on:



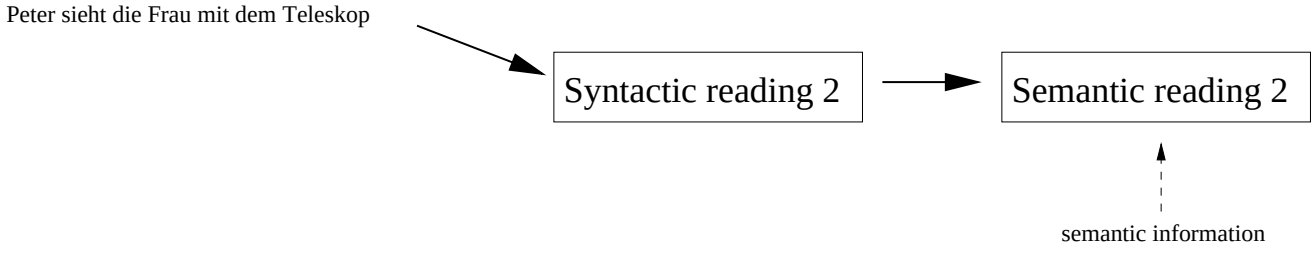
Non-integrated

- a *non-integrated* system cannot make use of information from semantics early on:



Non-integrated

- a *non-integrated* system cannot make use of information from semantics early on:



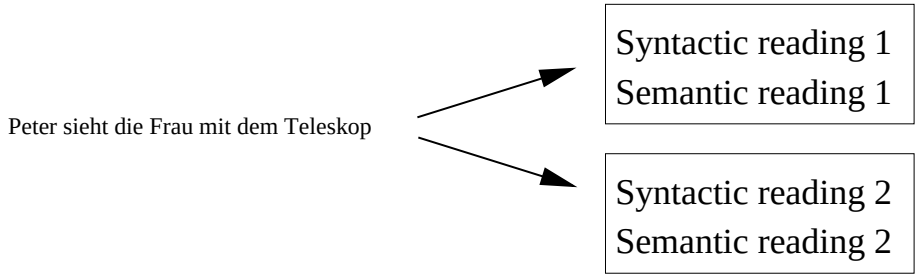
Integrate!

- in an *integrated* system, disambiguating information can come in from any dimension, and can immediately disambiguate the others:

Peter sieht die Frau mit dem Teleskop

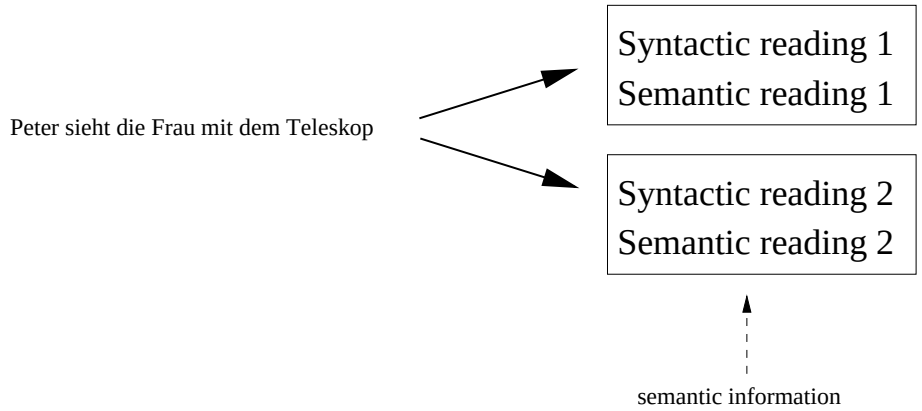
Integrate!

- in an *integrated* system, disambiguating information can come in from any dimension, and can immediately disambiguate the others:



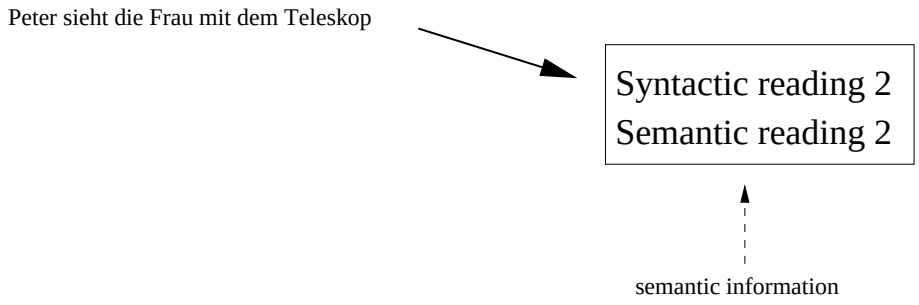
Integrate!

- in an *integrated* system, disambiguating information can come in from any dimension, and can immediately disambiguate the others:



Integrate!

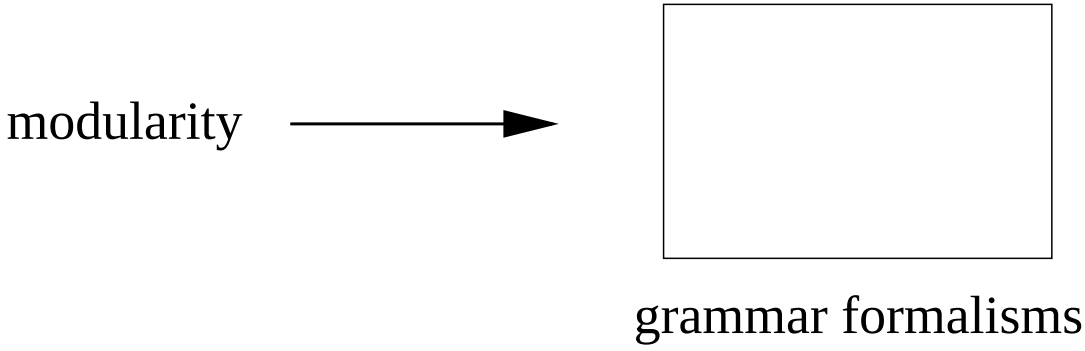
- in an *integrated* system, disambiguating information can come in from any dimension, and can immediately disambiguate the others:



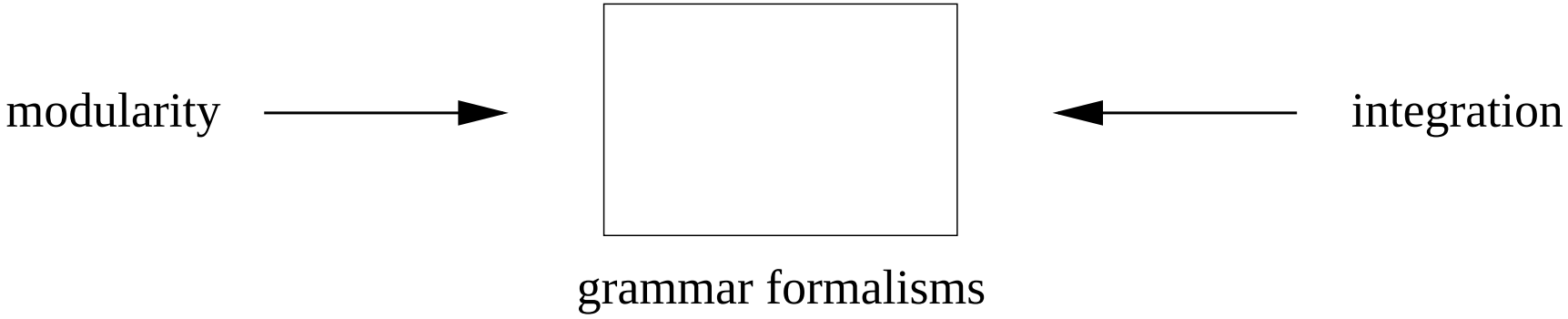
The situation

grammar formalisms

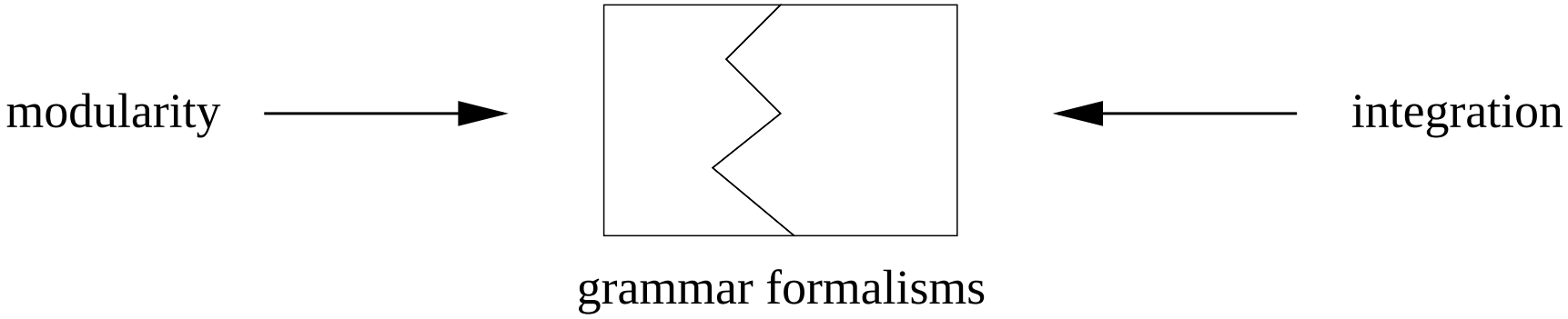
The situation



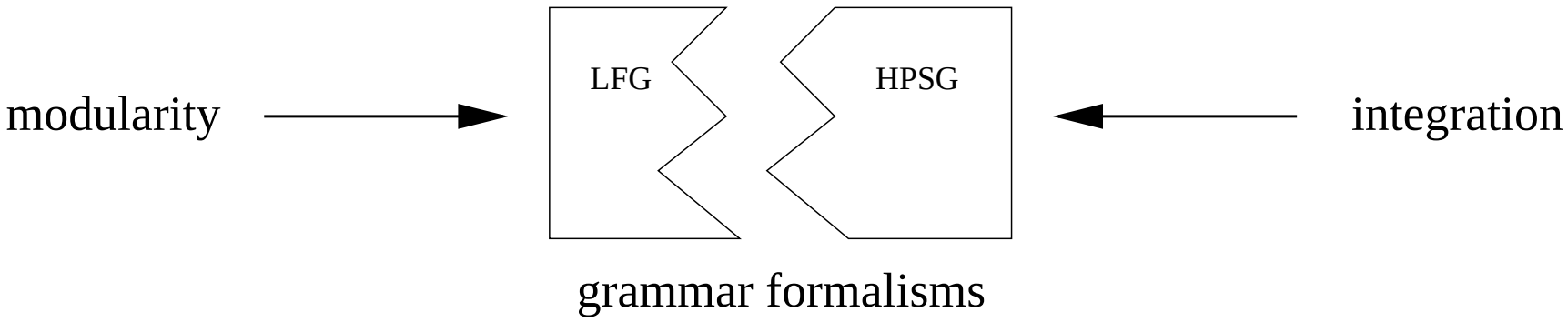
The situation



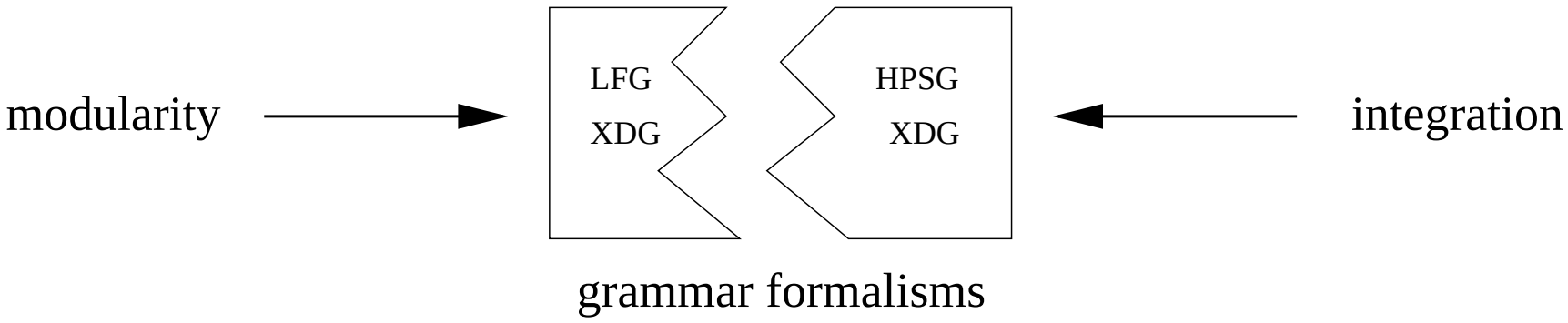
The situation



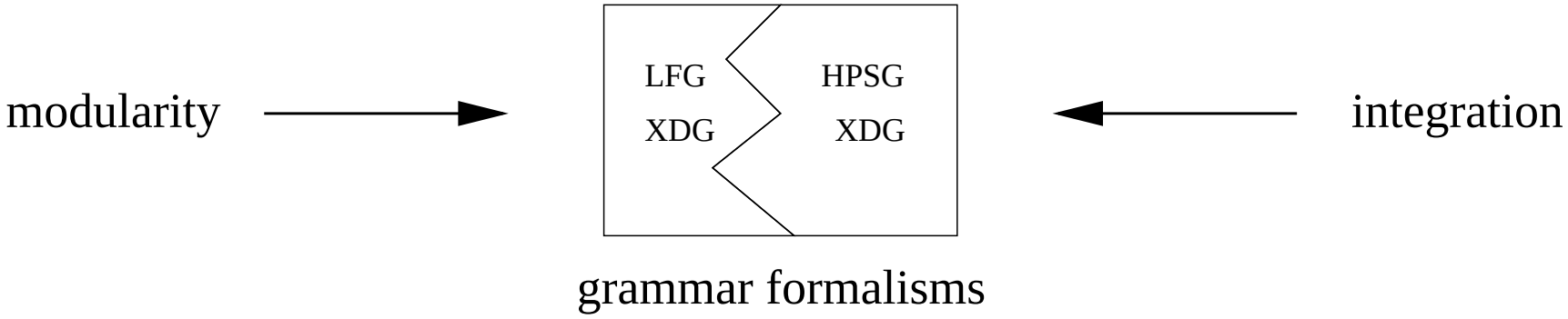
The situation



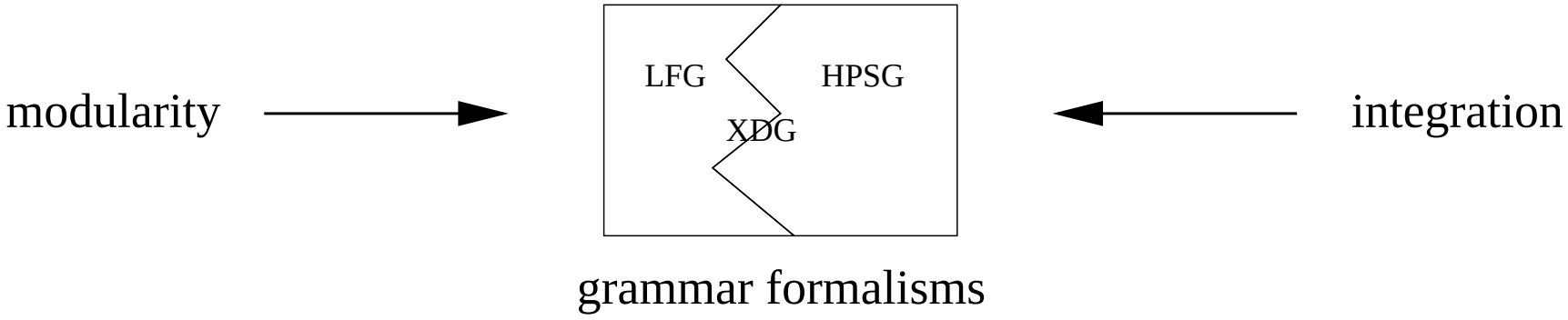
The situation



The situation



The situation

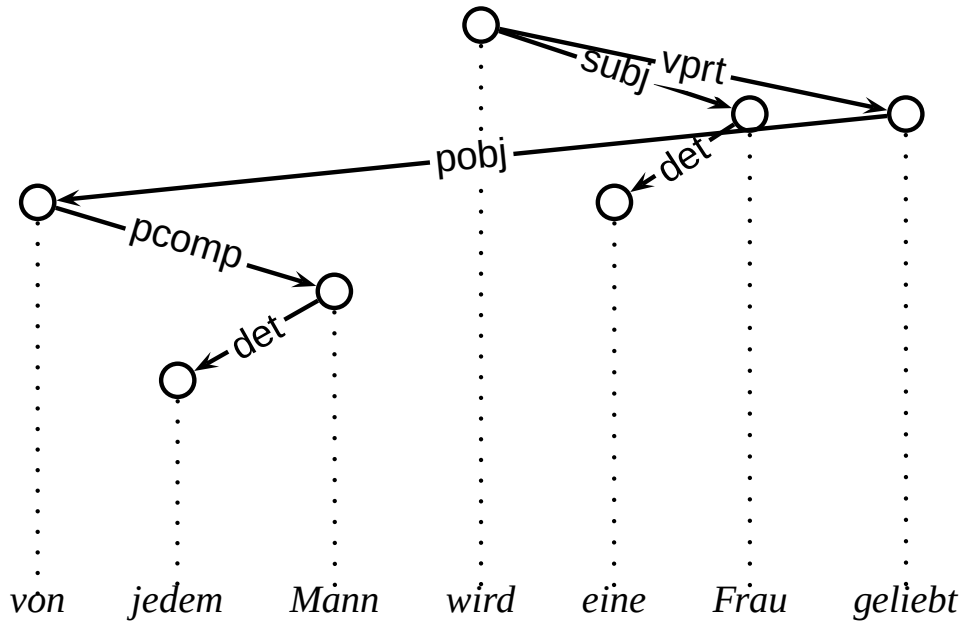


XDG dimensions

- XDG can be instantiated with *any* number of *dimensions*
- dimensions fold out the linguistic dimensions
- in the following, we show an instantiation with three dimensions: surface syntax, deep syntax, and topological fields

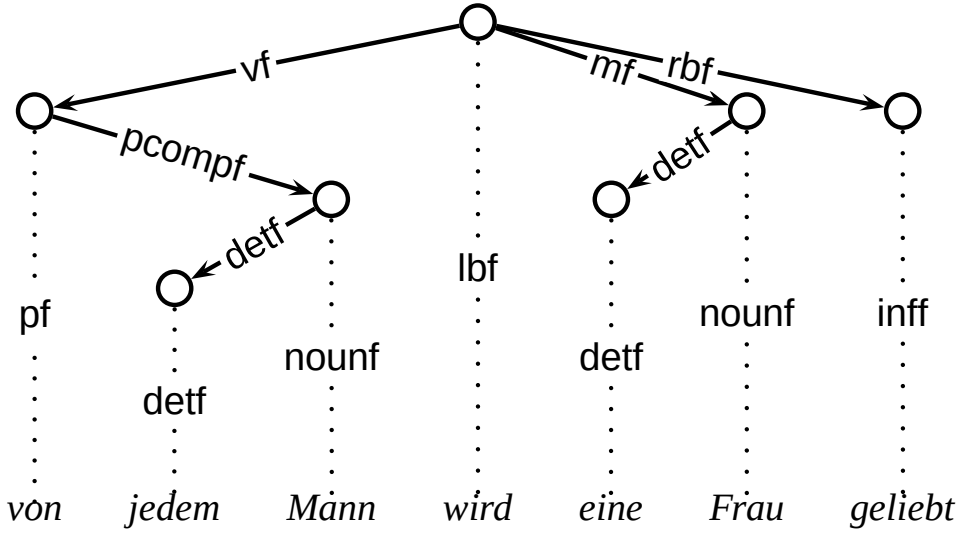
Example surface syntax analysis

- *von jedem Mann* is the surface object (of *geliebt*), *eine Frau* is the surface subject (of *wird*):



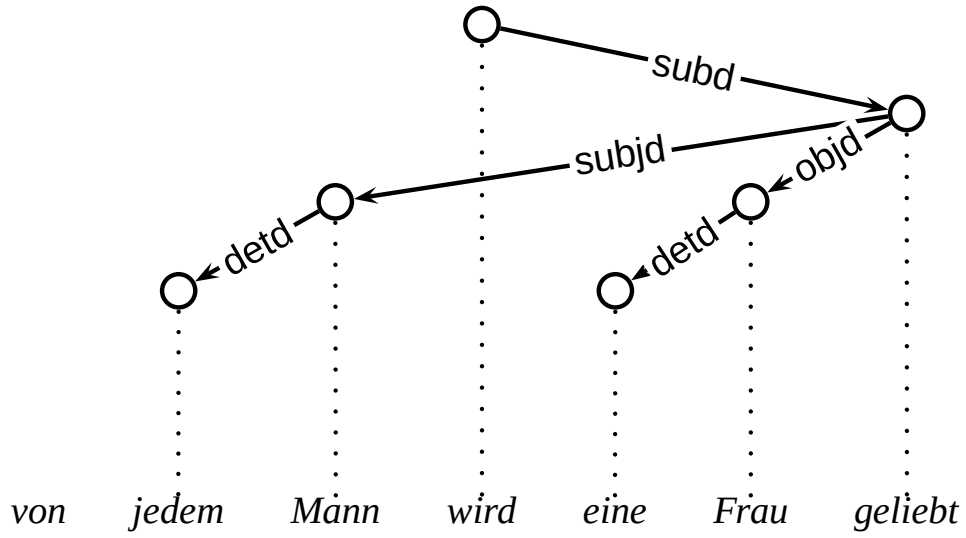
Example topological fields analysis

- *von jedem Mann* is in the Vorfeld, *eine Frau* is in the Mittelfeld of *wird*:



Example deep syntax analysis

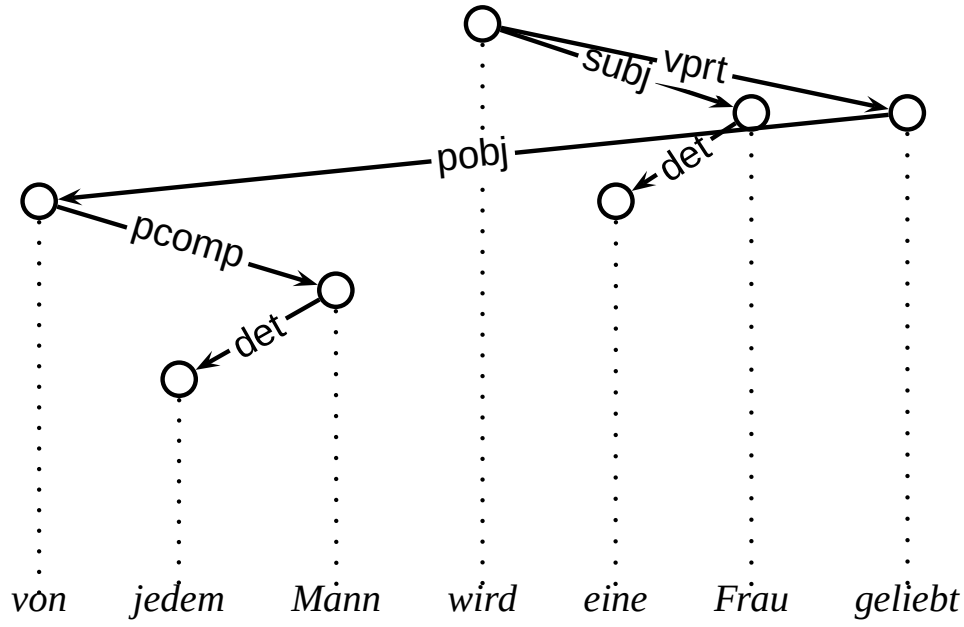
- *jedem Mann* is the deep subject, *eine Frau* is the deep object of *geliebt*:



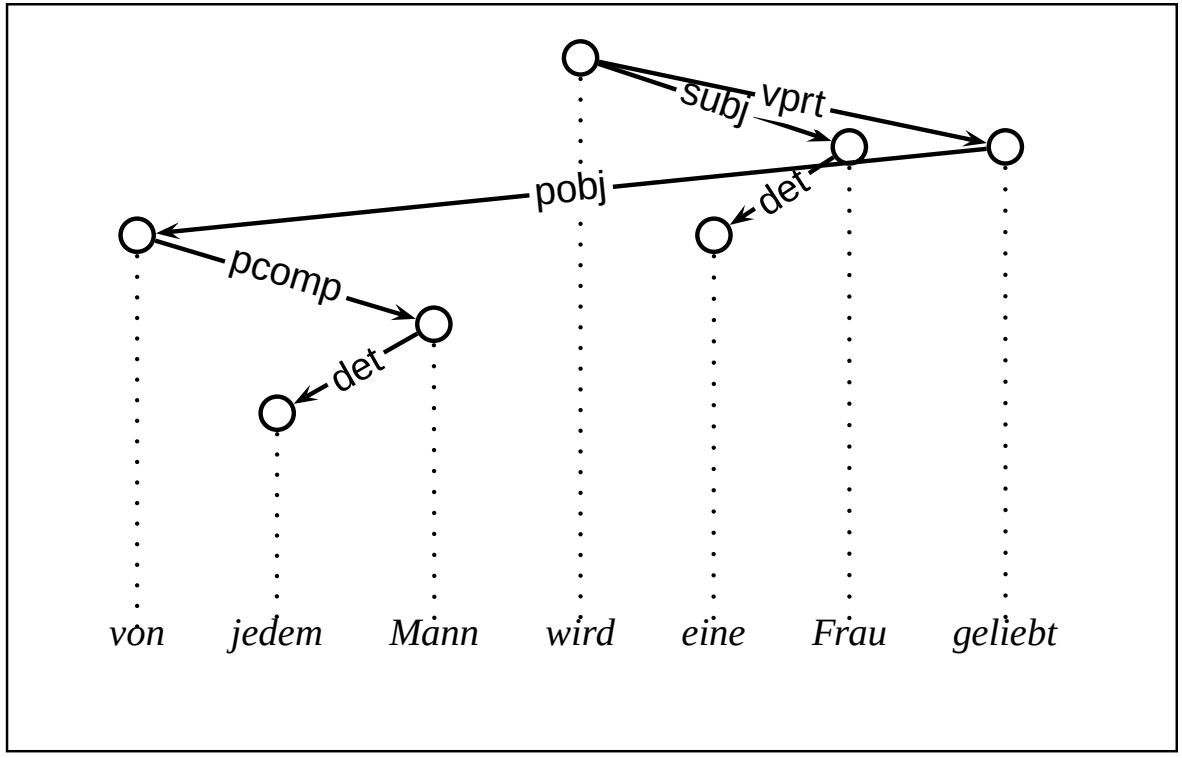
XDG principles

- an XDG grammar makes use of any number of *principles* to restrict the well-formedness conditions of analyses
- principles can be either *one-dimensional* (only restrict one dimension) or *multi-dimensional* (restrict the relation of two or more dimensions)
- principles can be either *lexicalized* or not
- the lexicon can be conveniently built up using abstractions like lexical inheritance and crossings (Candito 1996)

Tree principle (surface syntax)

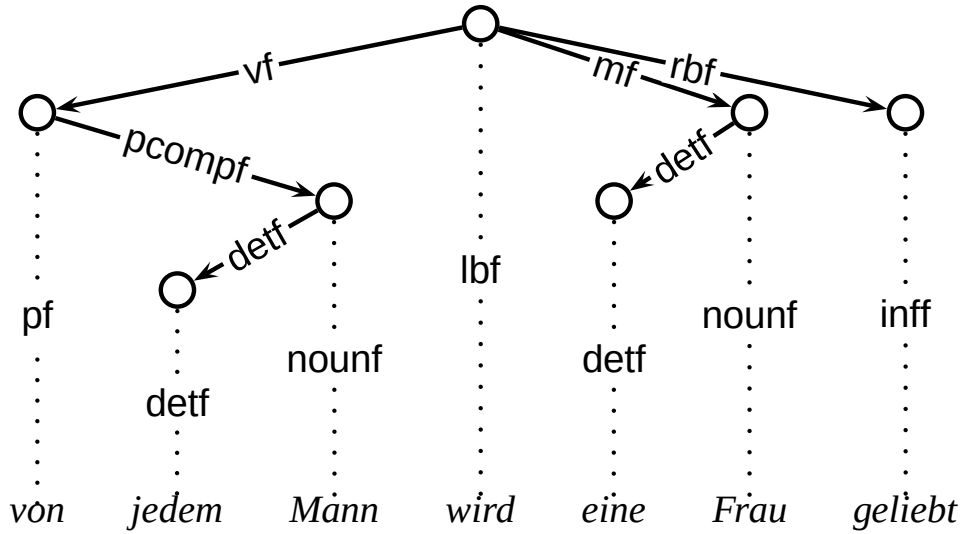


Tree principle (surface syntax)

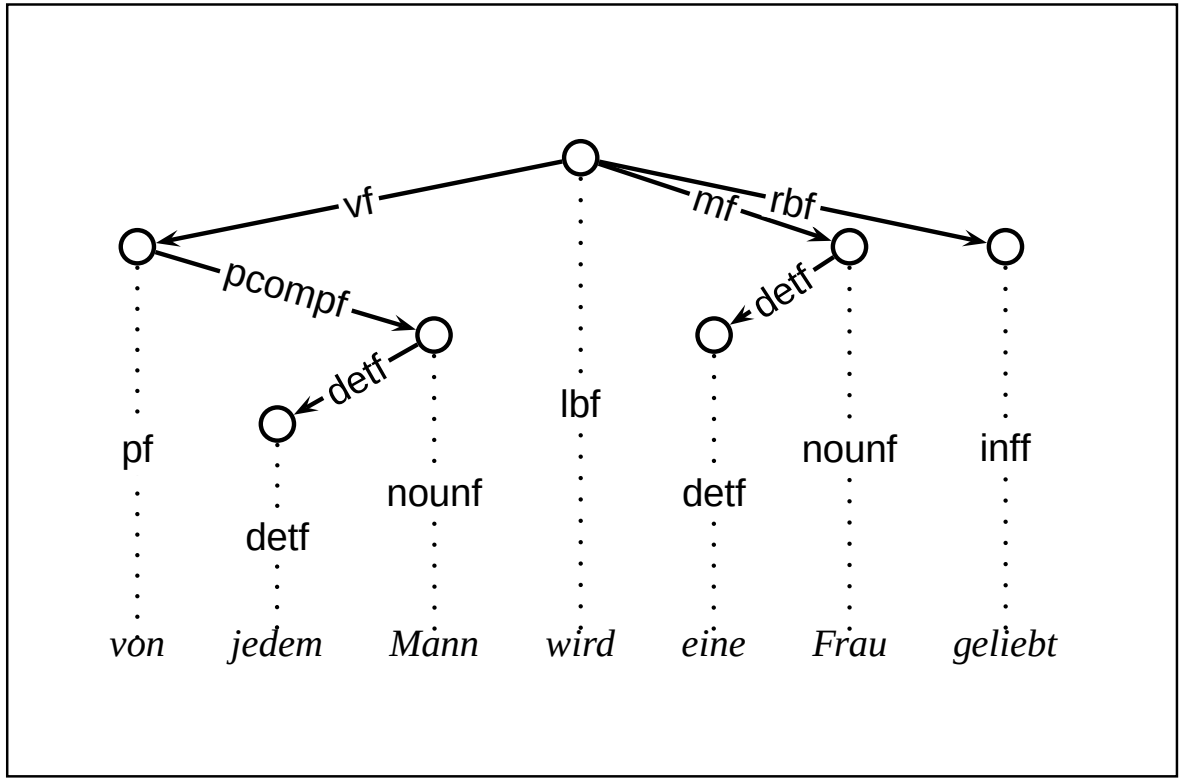


Tree

Tree principle (topological fields)

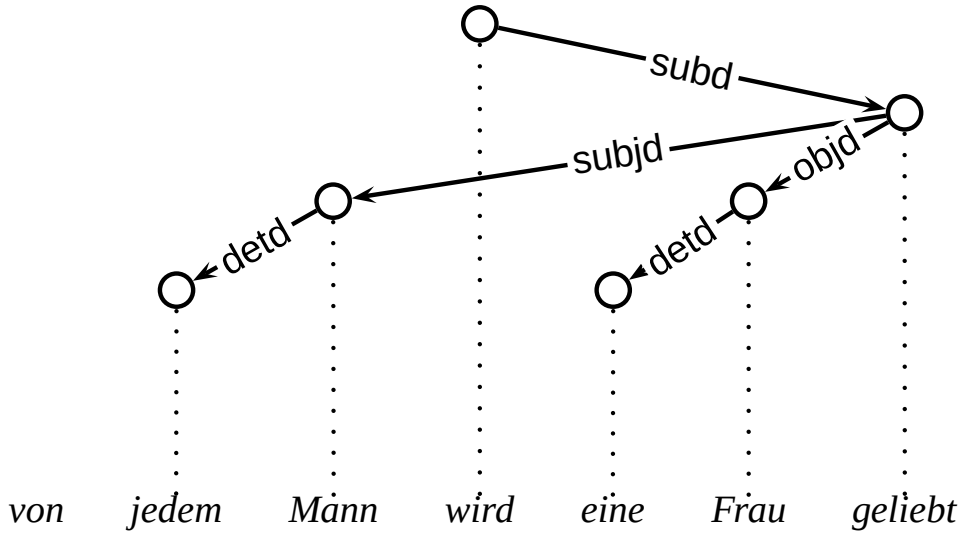


Tree principle (topological fields)

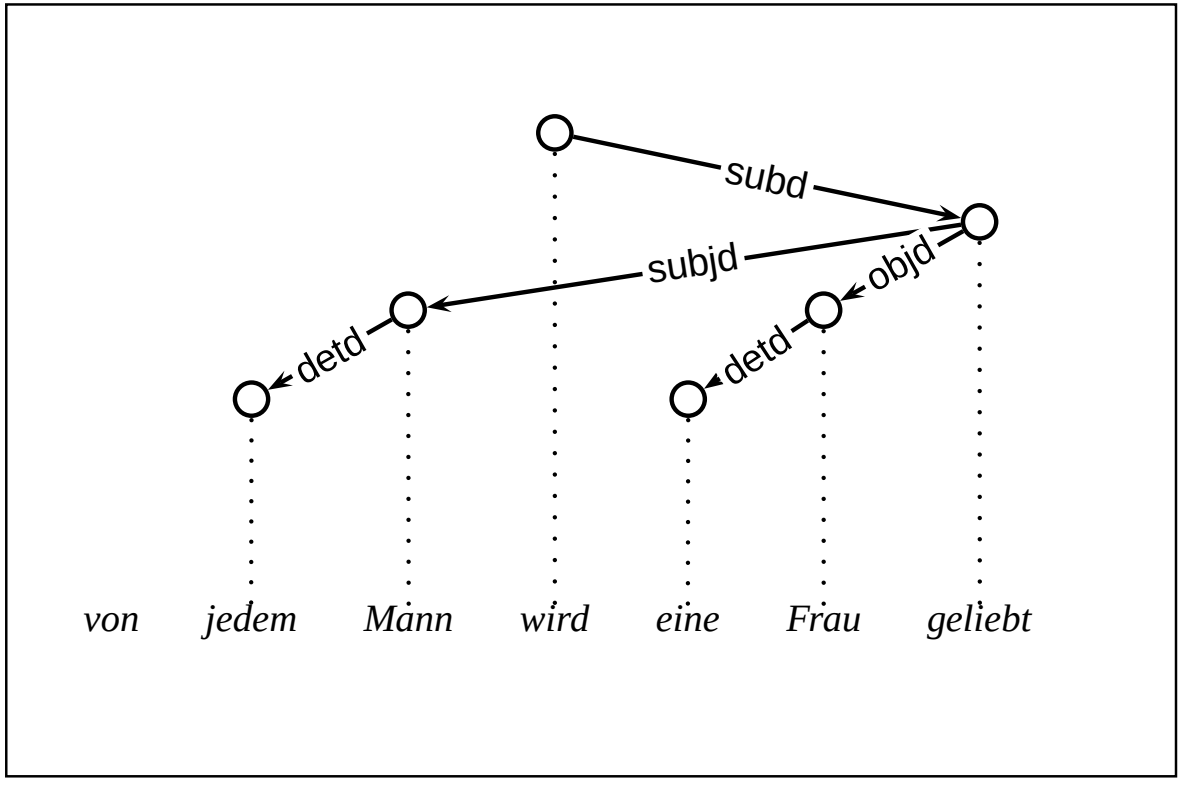


Tree

DAG principle (deep syntax)

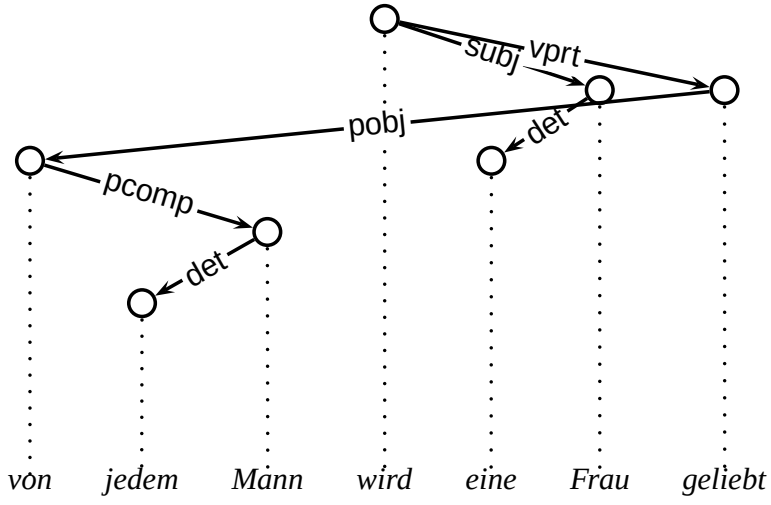


DAG principle (deep syntax)

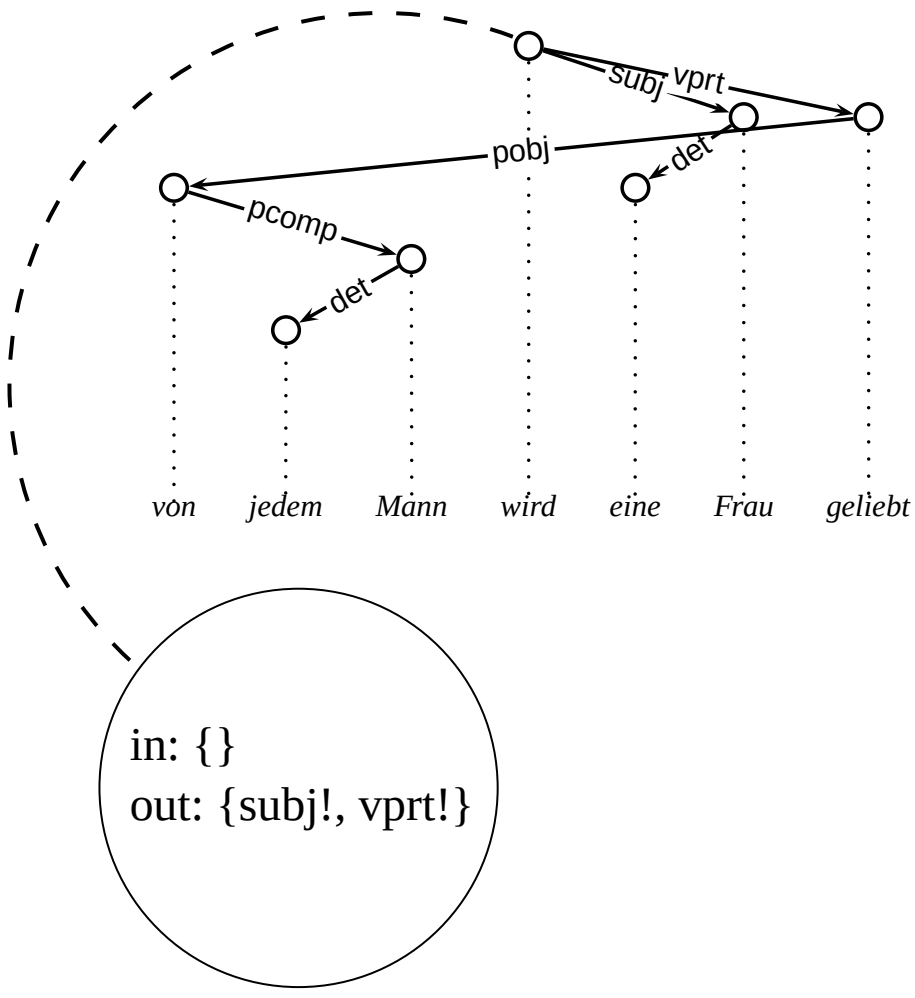


DAG

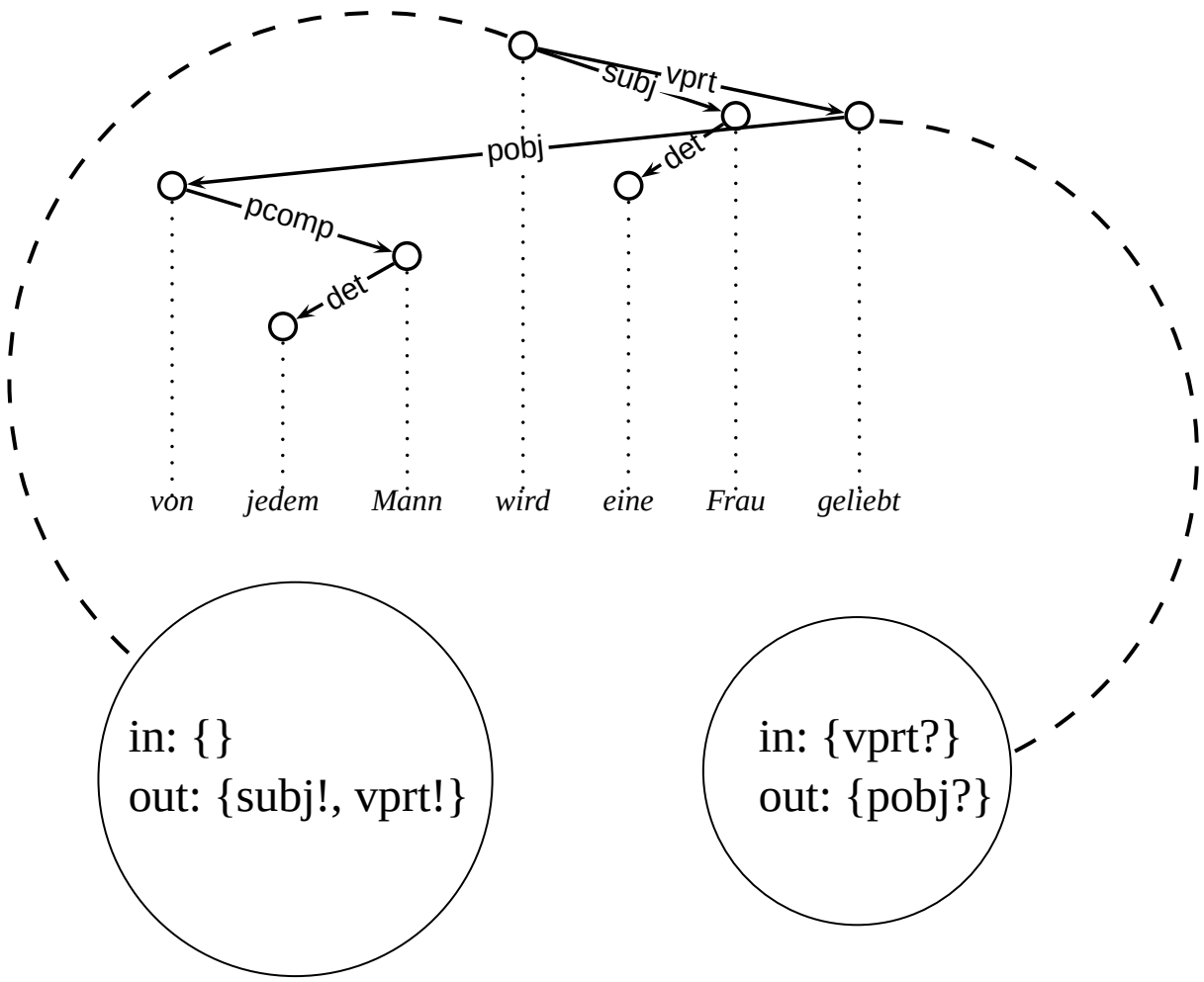
Valency principle (surface syntax)



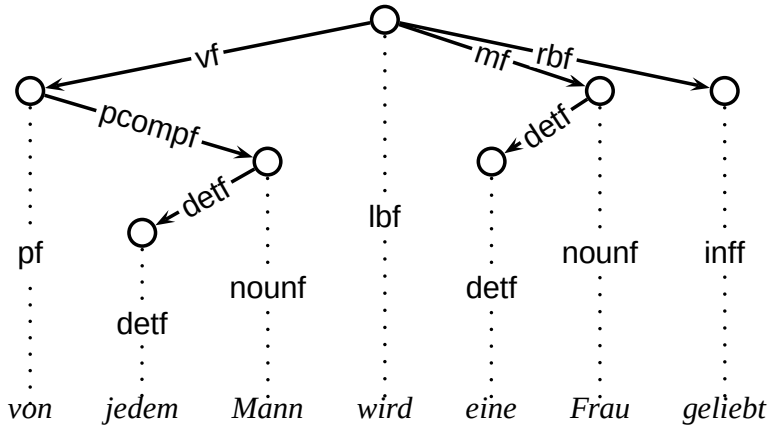
Valency principle (surface syntax)



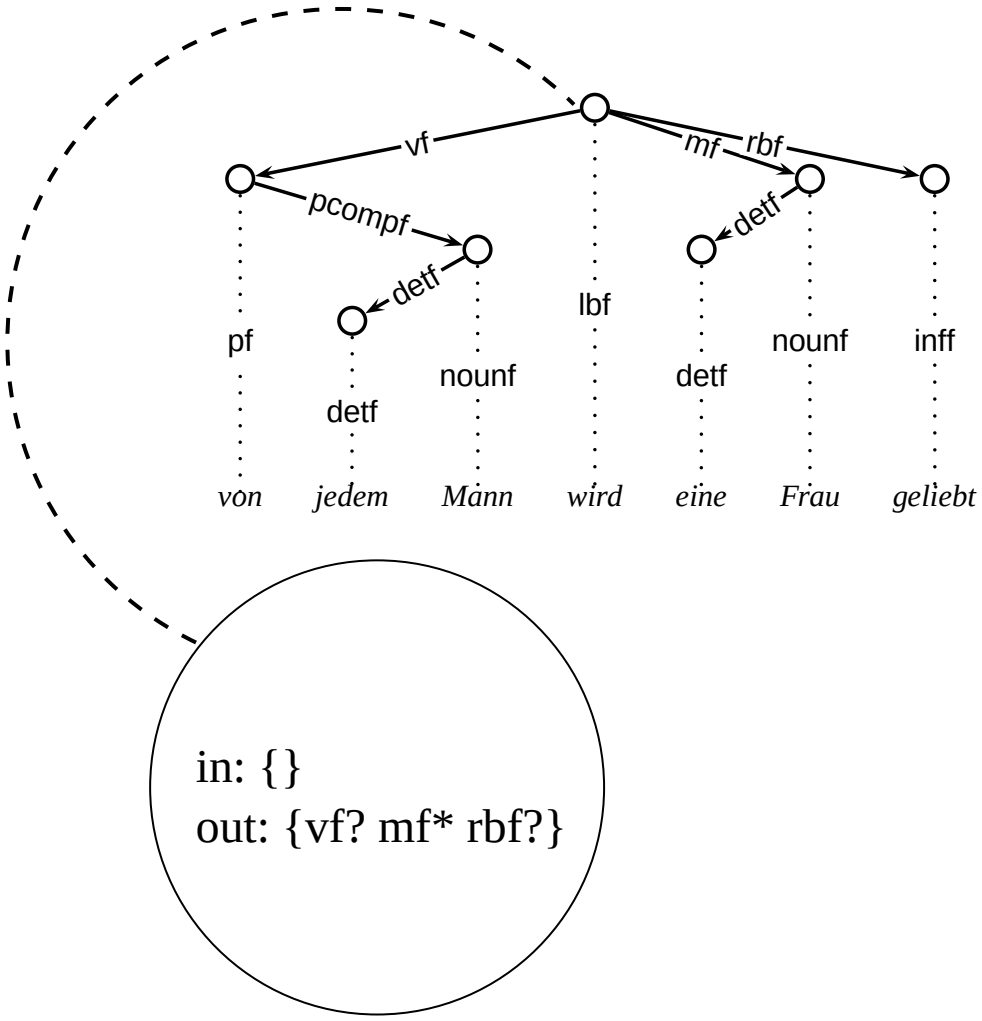
Valency principle (surface syntax)



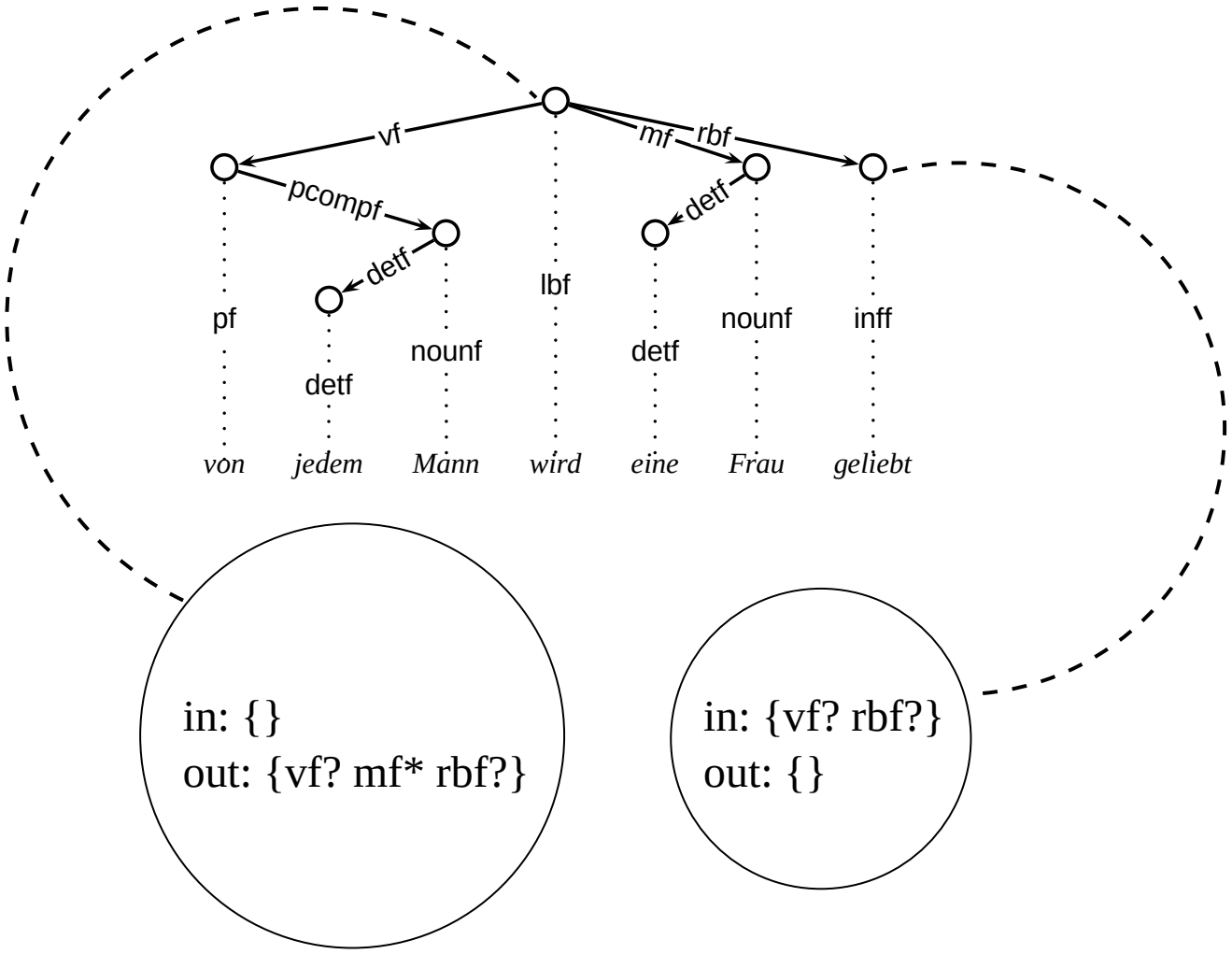
Valency principle (topological fields)



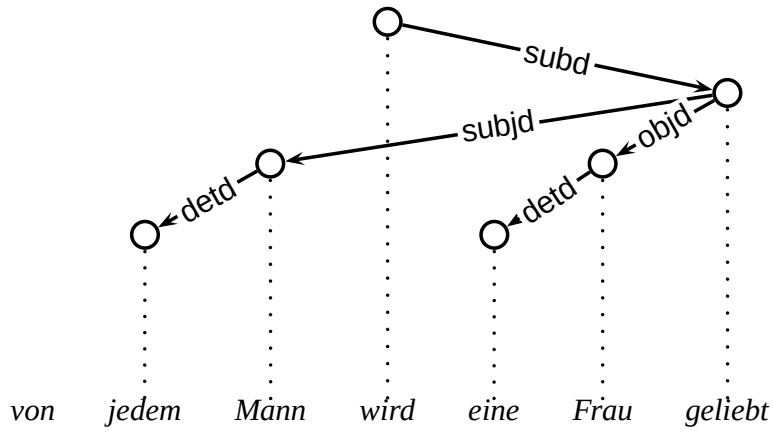
Valency principle (topological fields)



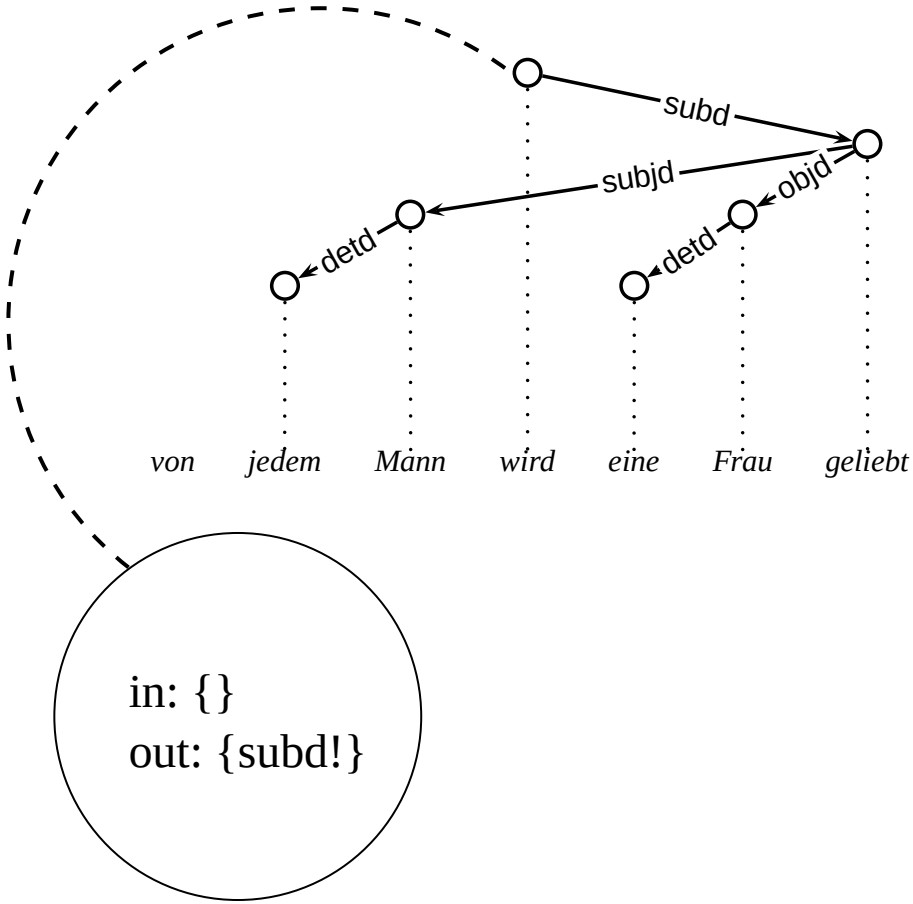
Valency principle (topological fields)



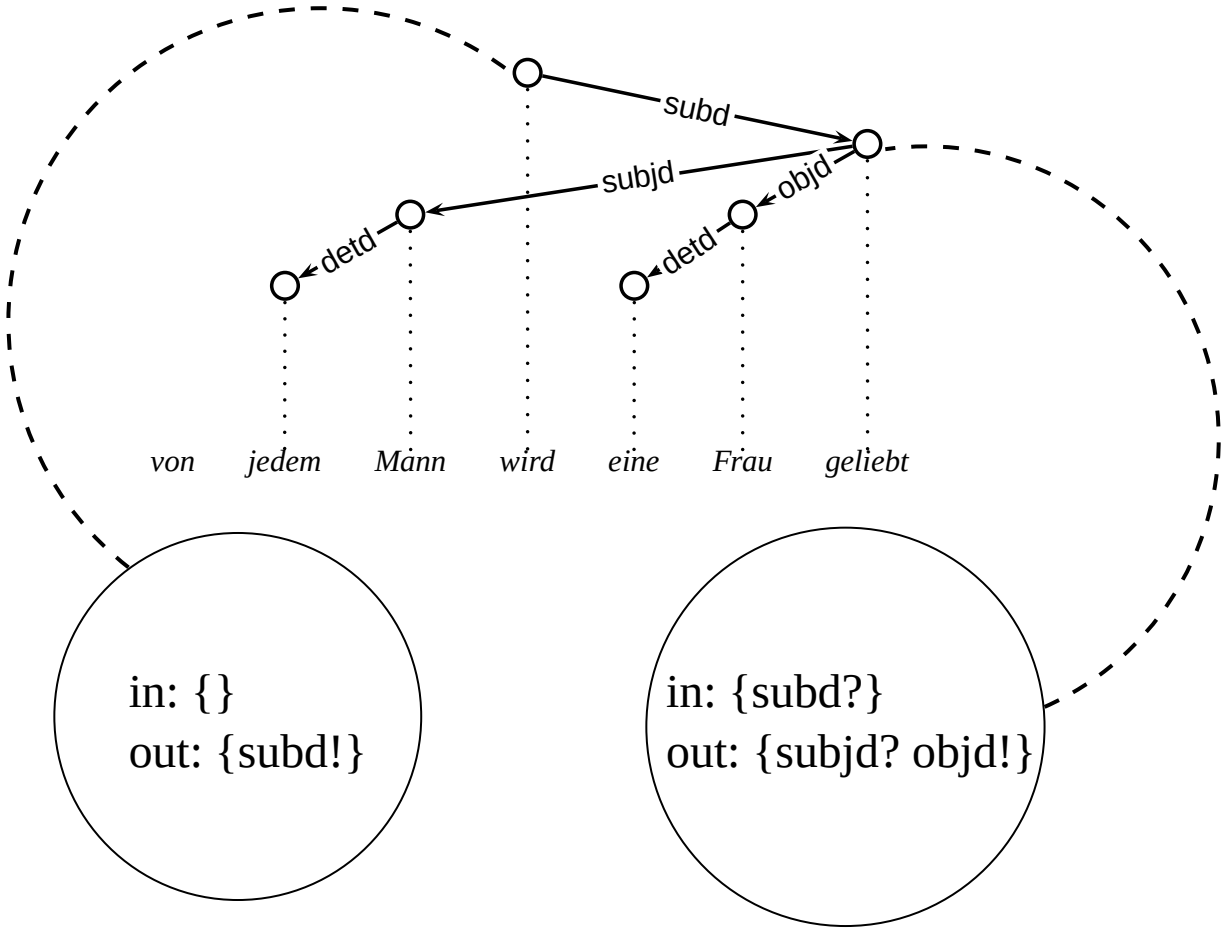
Valency principle (deep syntax)



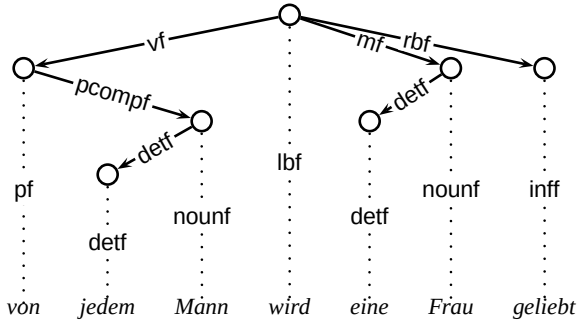
Valency principle (deep syntax)

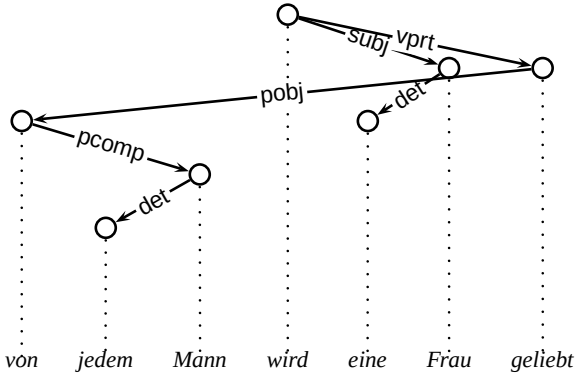


Valency principle (deep syntax)

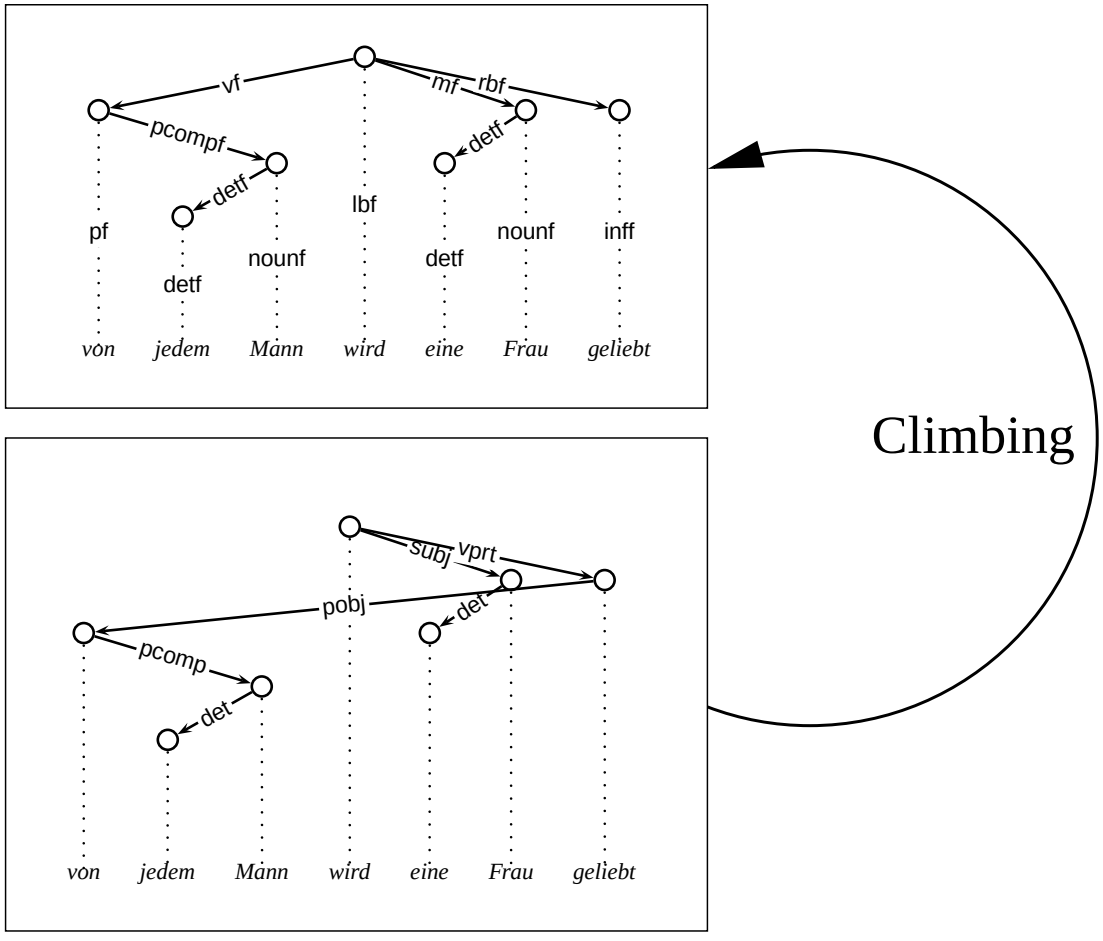


Climbing principle

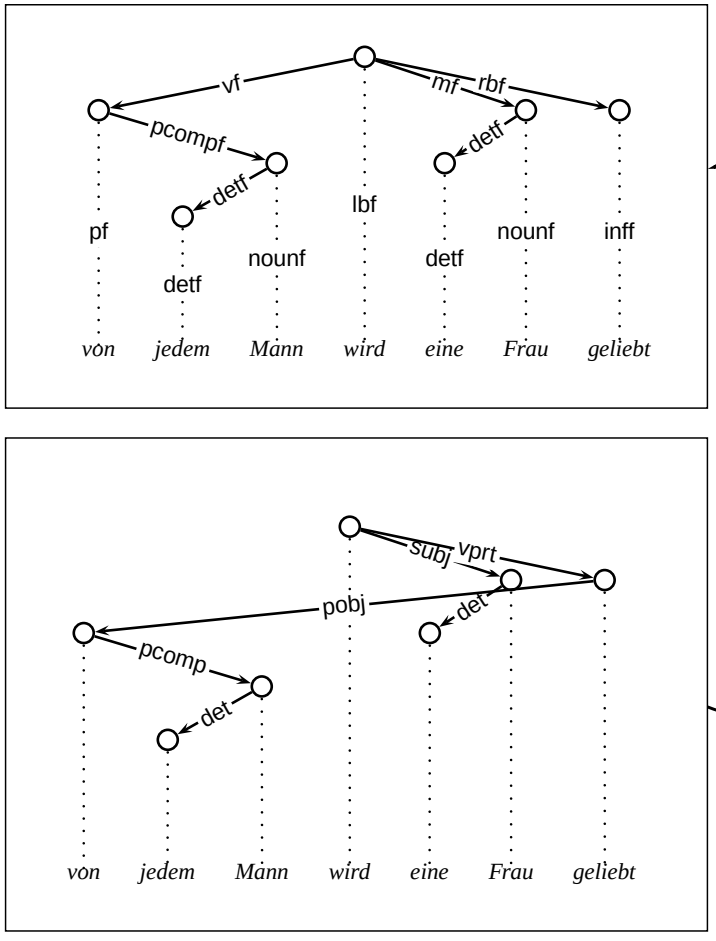




Climbing principle



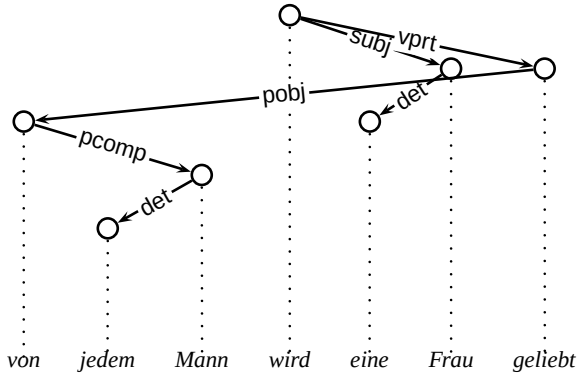
Climbing principle

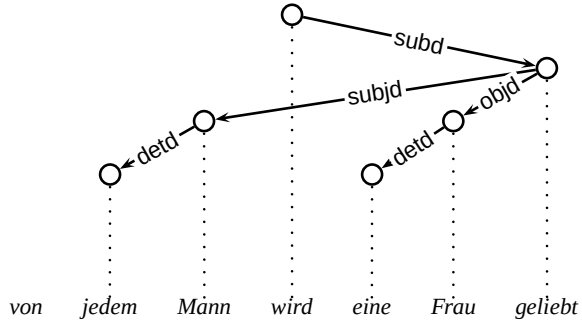


Climbing

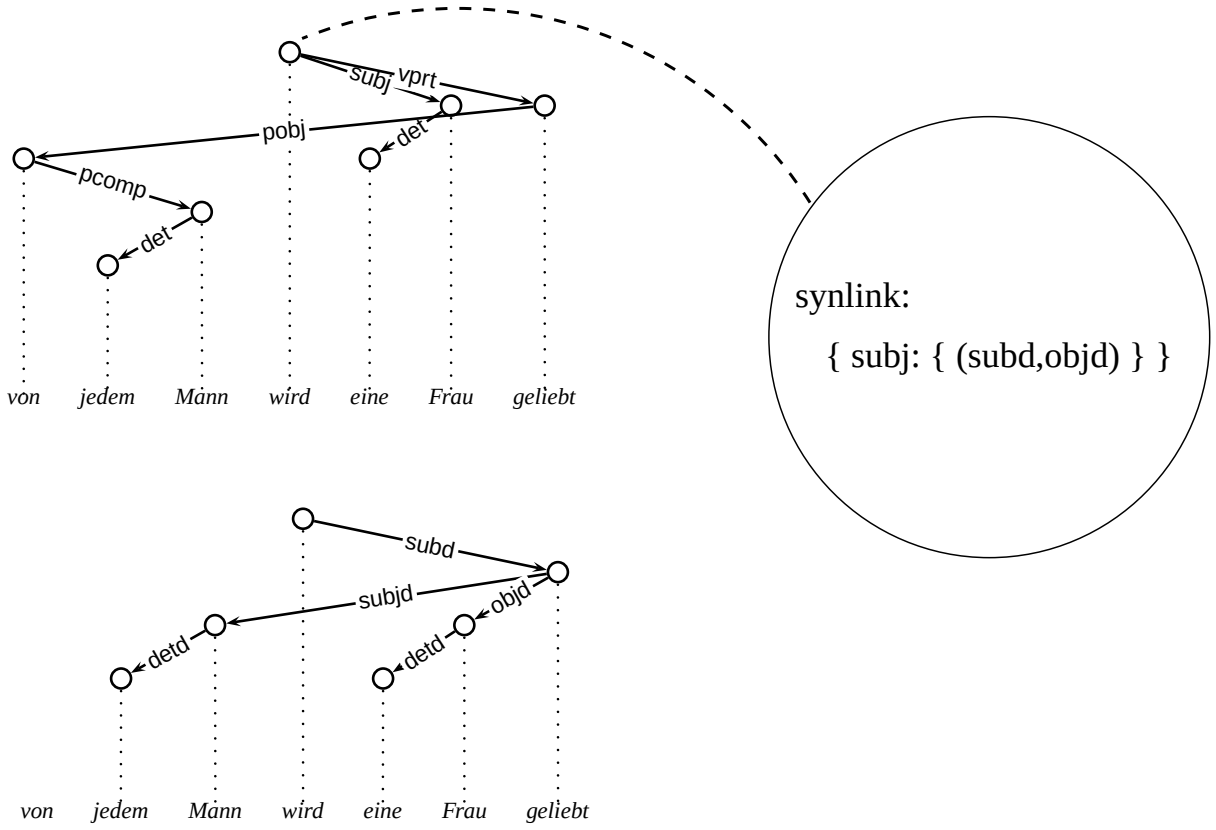
The topology is a flattening of the surface syntax.

Syntactic linking

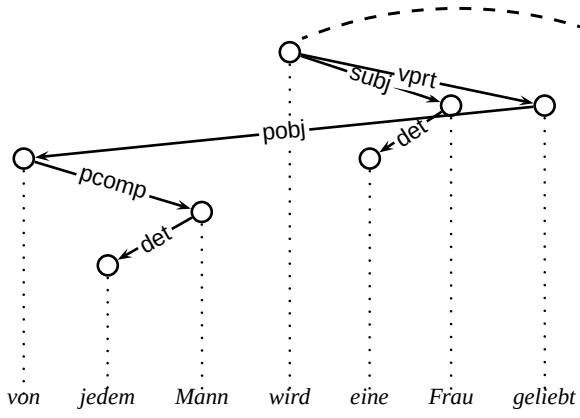




Syntactic linking

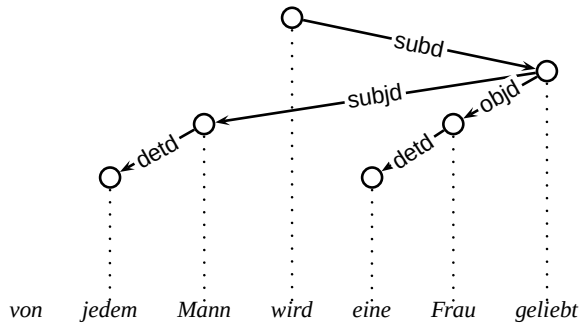


Syntactic linking



synlink:

{ subj: { (subd,objd) } }



The subject of the passive auxiliary becomes the deep object of a subordinated verb.

Representing semantics

- *Von jedem Mann wird eine Frau geliebt.*
- semantics (weak reading):
  $\forall x.\mathit{man}(x) \rightarrow \exists y.\mathit{woman}(y) \wedge \mathit{love}(x, y)$
- two dimensions: predicate-argument structure and scope
- in XDG, we can use the same ideas also to represent semantics

Predicate-argument dimension

$$\forall x \;\; \text{man}(x) \quad \Rightarrow \quad \exists y \;\; \text{woman}(y) \wedge \text{love}(x,y)$$

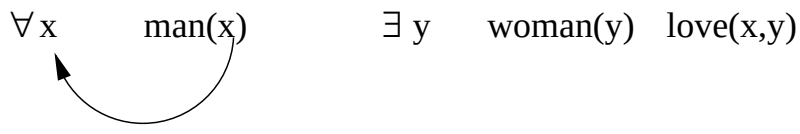
Predicate-argument dimension

$$\forall x \quad man(x) \quad \Rightarrow \quad \exists y \quad woman(y) \wedge love(x,y)$$

$$\forall x \qquad man(x) \qquad \exists y \qquad woman(y) \quad love(x,y)$$

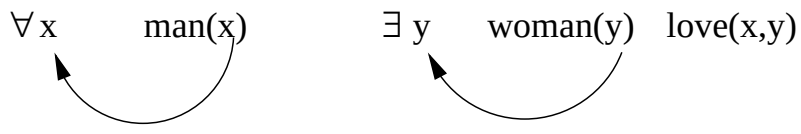
Predicate-argument dimension

$$\forall x \quad man(x) \quad \Rightarrow \quad \exists y \quad woman(y) \wedge love(x,y)$$



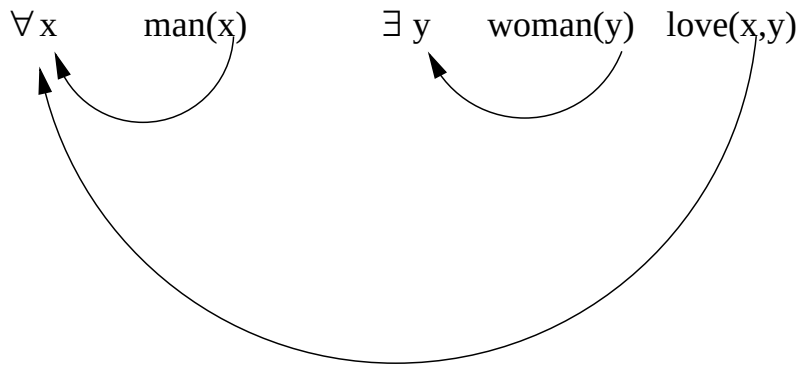
Predicate-argument dimension

$$\forall x \;\; \text{man}(x) \quad \Rightarrow \quad \exists y \;\; \text{woman}(y) \wedge \text{love}(x,y)$$



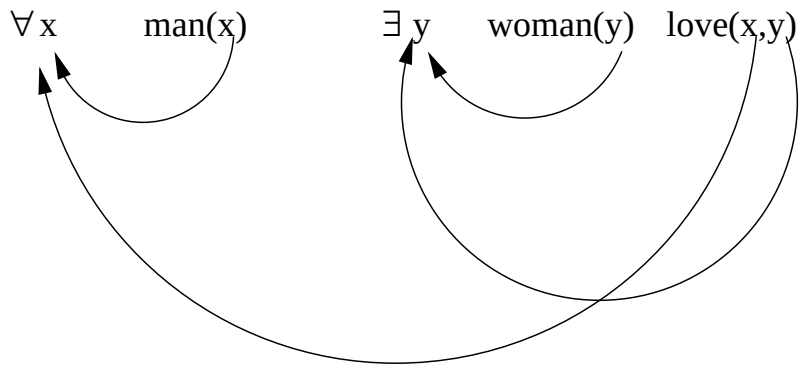
Predicate-argument dimension

$$\forall x \;\; man(x) \;\; \Rightarrow \;\; \exists y \;\; woman(y) \wedge love(x,y)$$



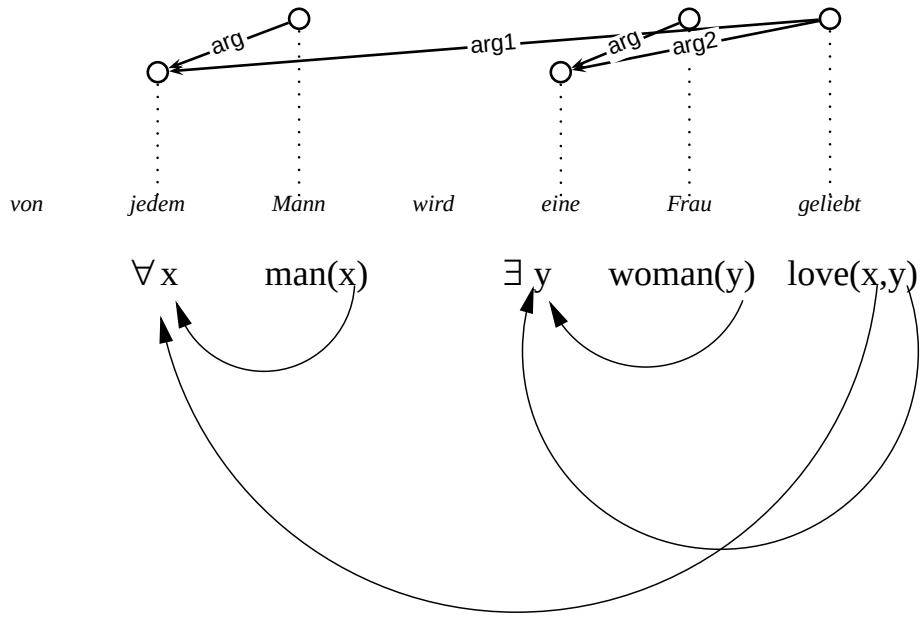
Predicate-argument dimension

$$\forall x \;\; \text{man}(x) \quad \Rightarrow \quad \exists y \;\; \text{woman}(y) \wedge \text{love}(x,y)$$



Predicate-argument dimension

$$\forall x \;\; man(x) \;\; \Rightarrow \;\; \exists y \;\; woman(y) \wedge love(x,y)$$



Scope dimension

$$\forall x \quad man(x) \quad \Rightarrow \quad \exists y \quad woman(y) \wedge love(x,y)$$

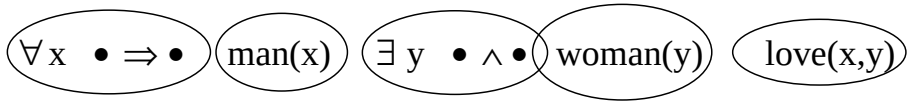
Scope dimension

$$\forall x \;\; man(x) \;\; \Rightarrow \;\; \exists y \;\; woman(y) \wedge love(x,y)$$

$$\forall x \;\; \bullet \Rightarrow \bullet \;\; man(x) \;\; \exists y \;\; \bullet \wedge \bullet \;\; woman(y) \;\; love(x,y)$$

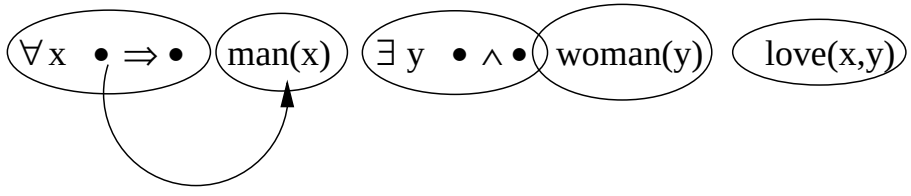
Scope dimension

$$\forall x \quad man(x) \quad \Rightarrow \quad \exists y \quad woman(y) \wedge love(x,y)$$



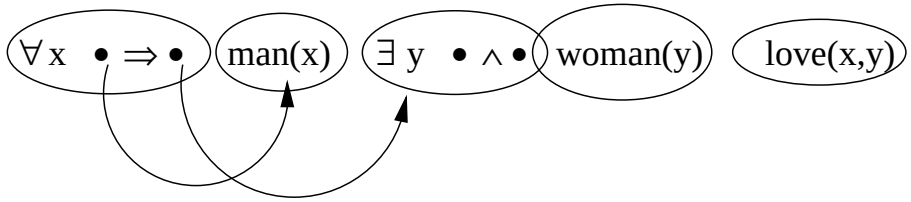
Scope dimension

$$\forall x \quad man(x) \quad \Rightarrow \quad \exists y \quad woman(y) \wedge love(x,y)$$



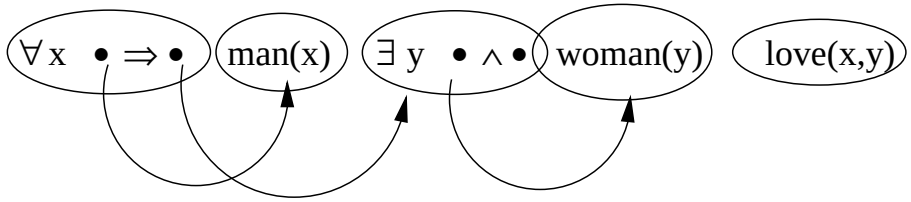
Scope dimension

$$\forall x \quad man(x) \quad \Rightarrow \quad \exists y \quad woman(y) \wedge love(x,y)$$



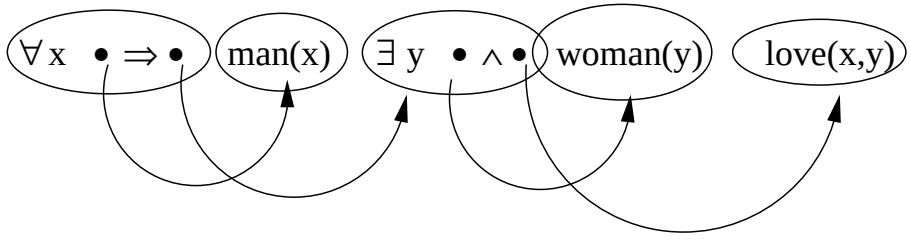
Scope dimension

$$\forall x \quad man(x) \quad \Rightarrow \quad \exists y \quad woman(y) \wedge love(x,y)$$



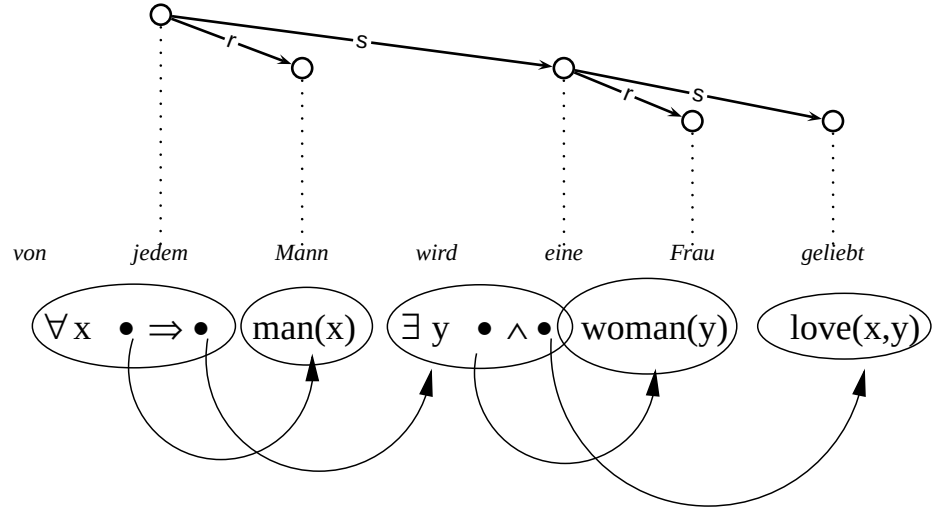
Scope dimension

$$\forall x \quad man(x) \quad \Rightarrow \quad \exists y \quad woman(y) \wedge love(x,y)$$

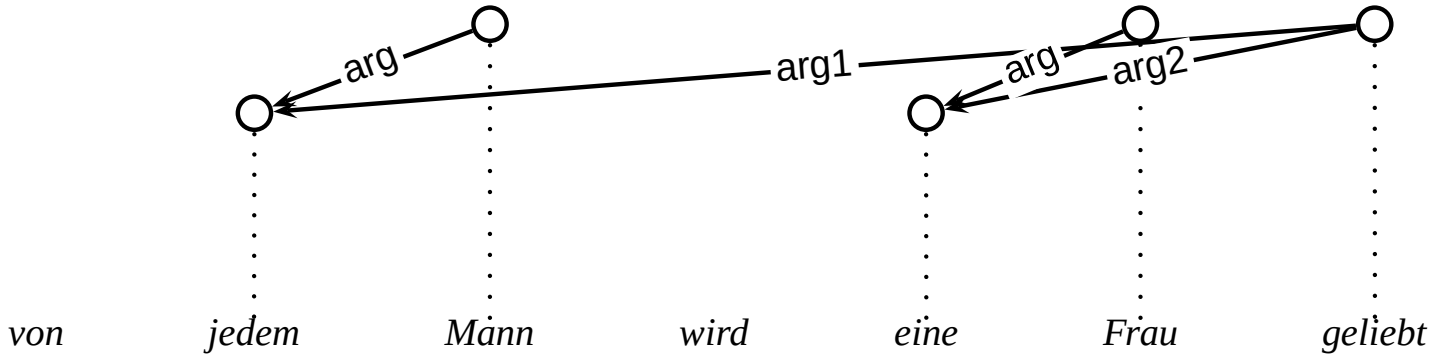


Scope dimension

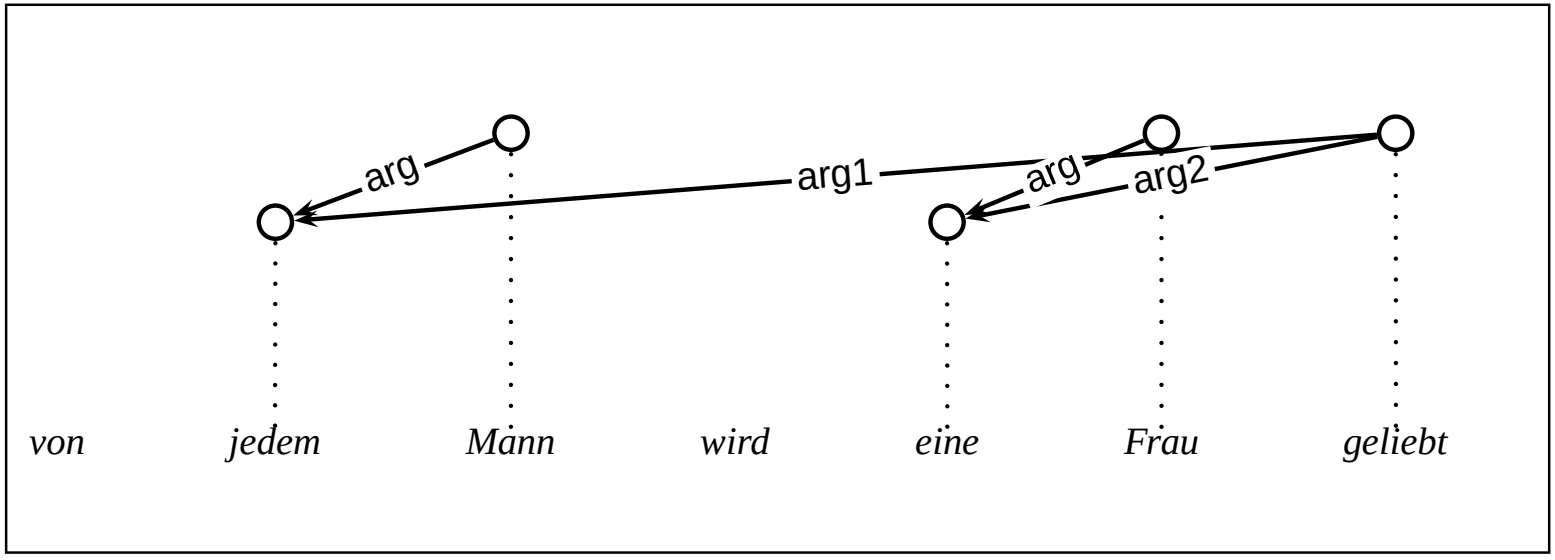
$$\forall x \quad man(x) \quad \Rightarrow \quad \exists y \quad woman(y) \wedge love(x,y)$$



DAG principle (predicate-argument)

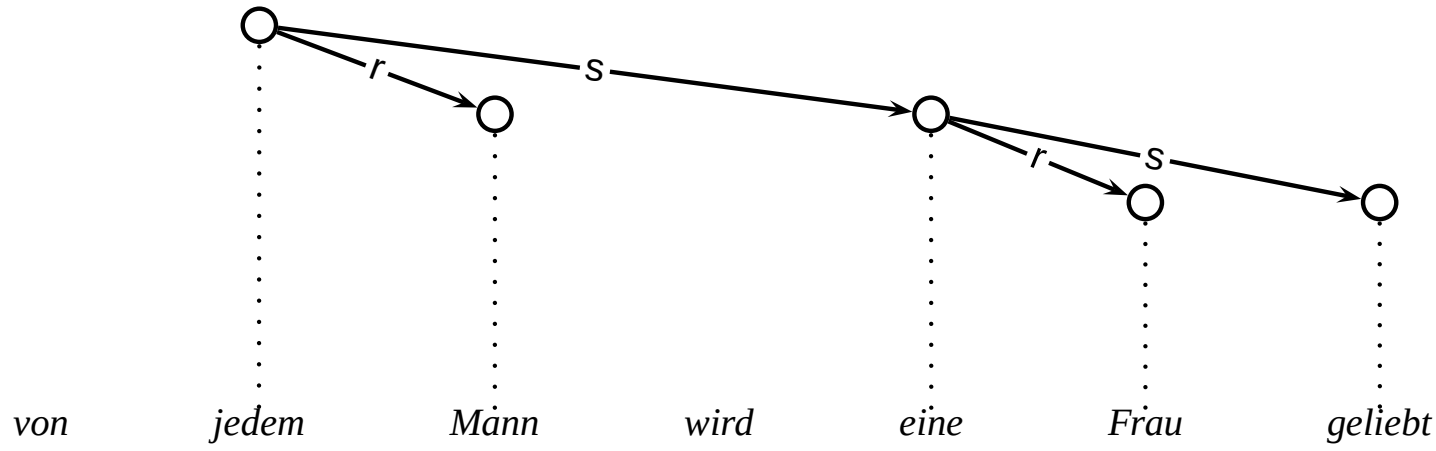


DAG principle (predicate-argument)

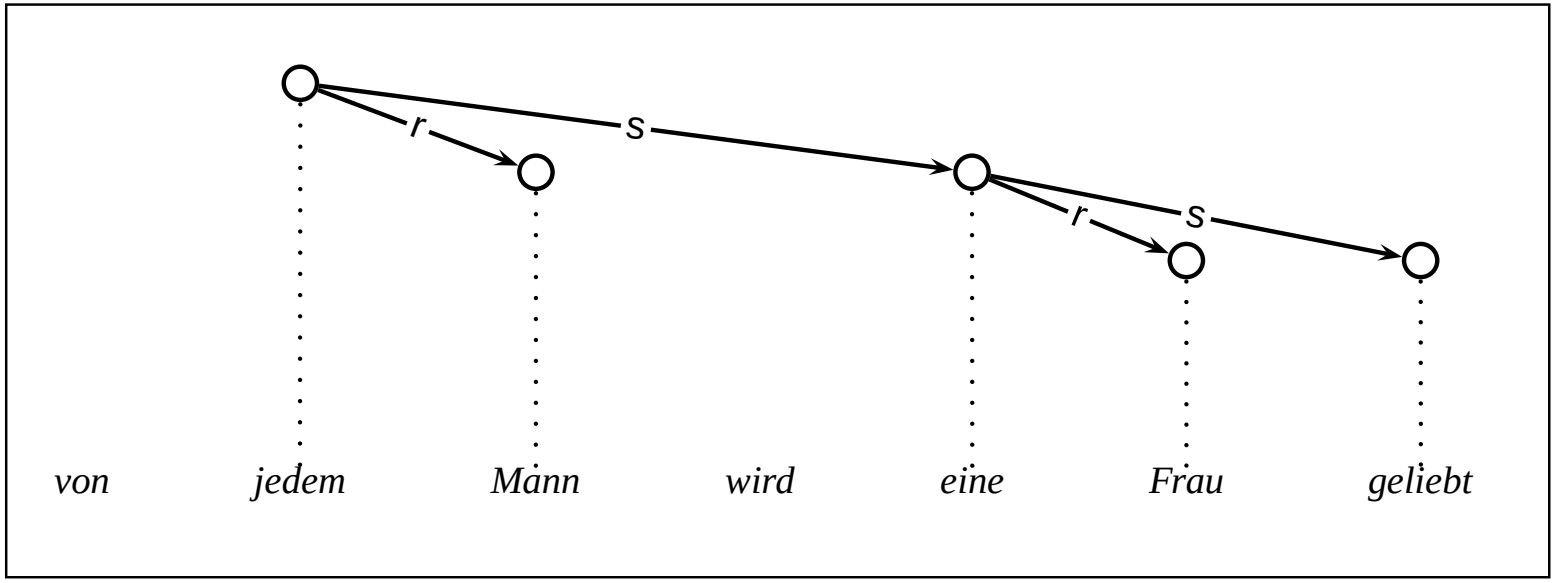


DAG

Tree principle (scope)

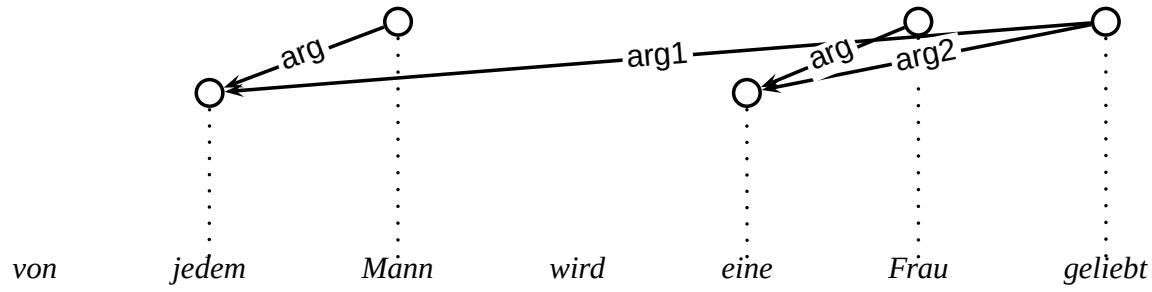


Tree principle (scope)

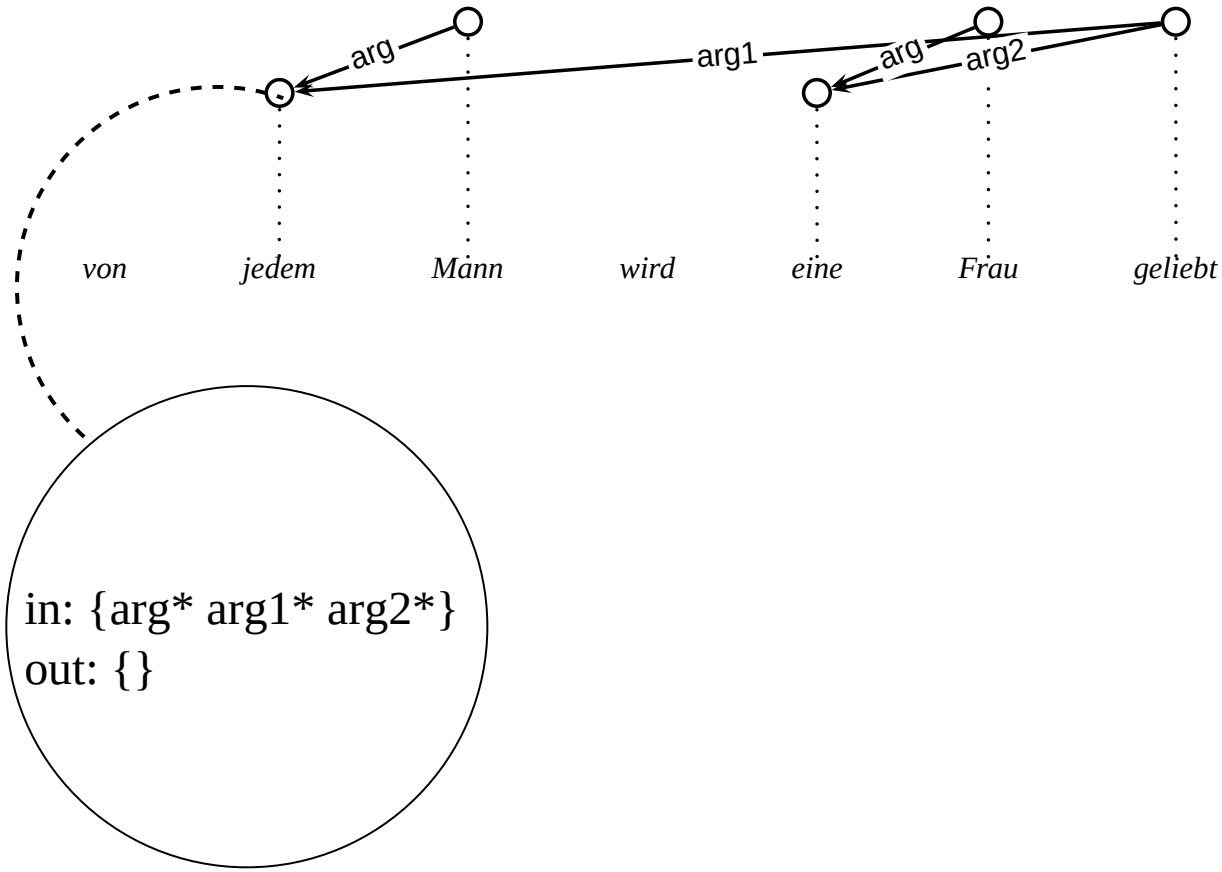


Tree

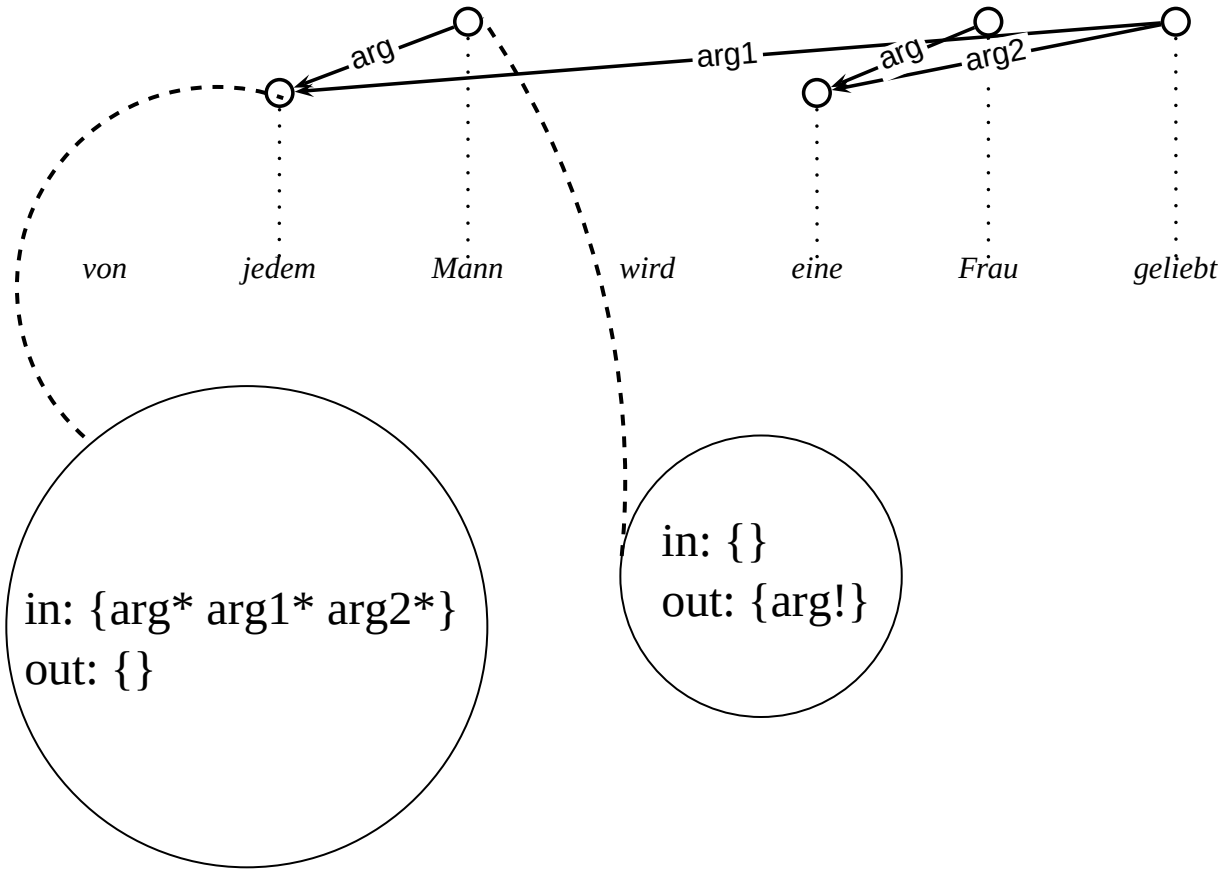
Valency principle (predicate-argument)



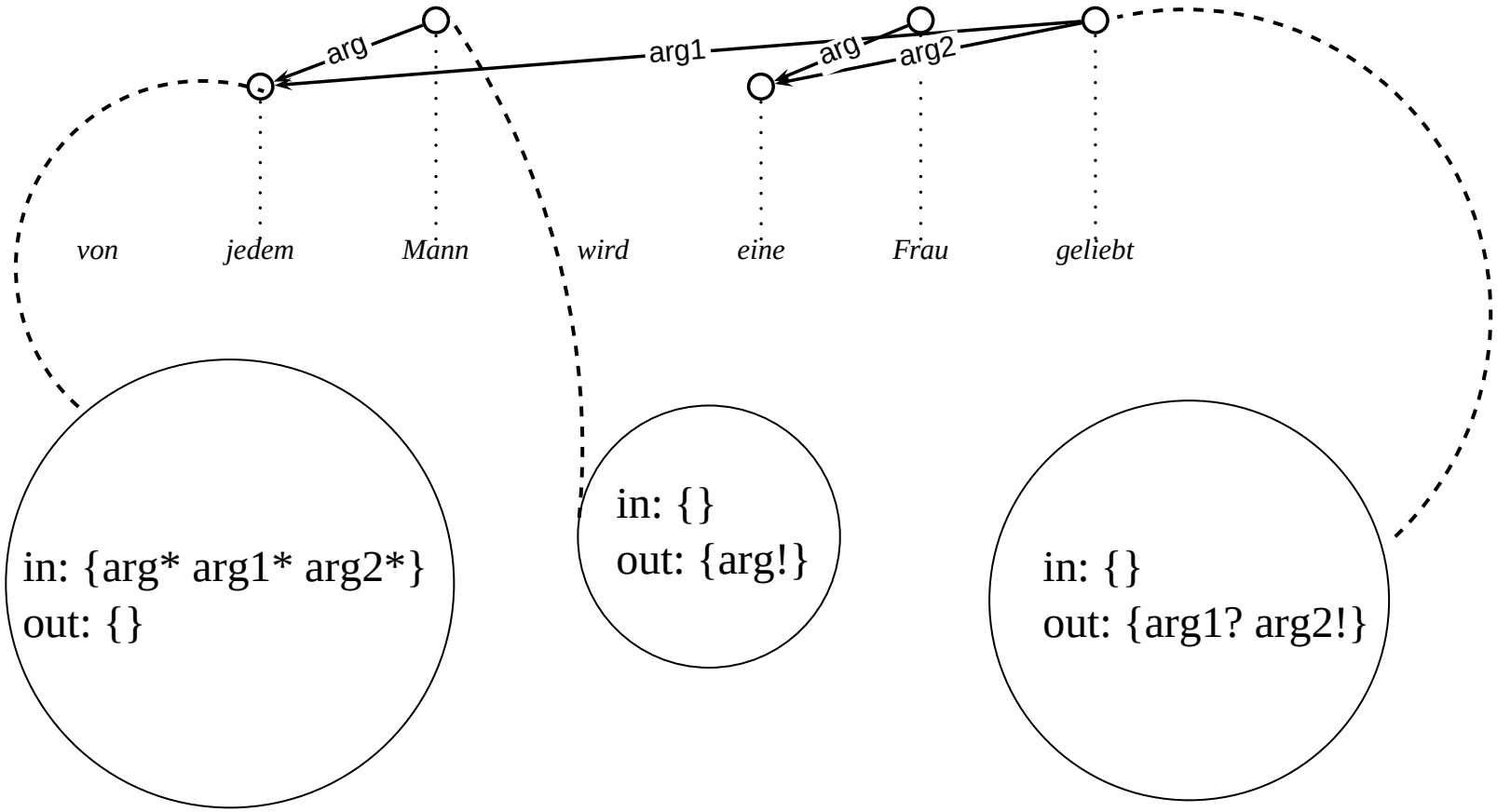
Valency principle (predicate-argument)



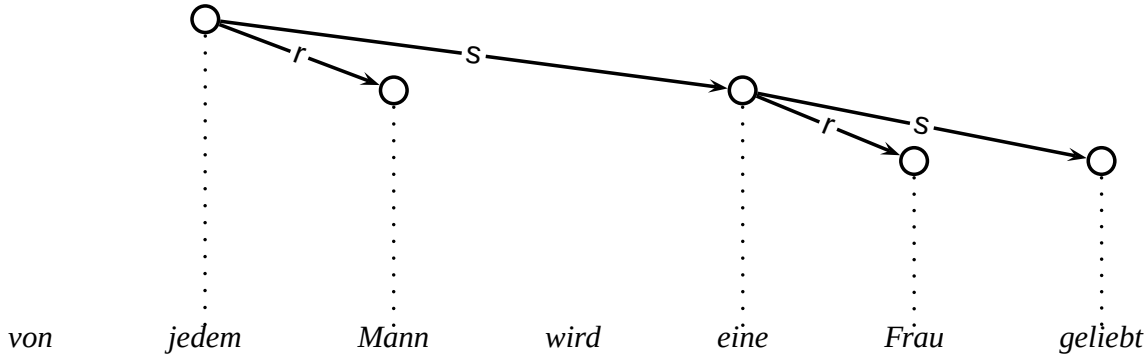
Valency principle (predicate-argument)



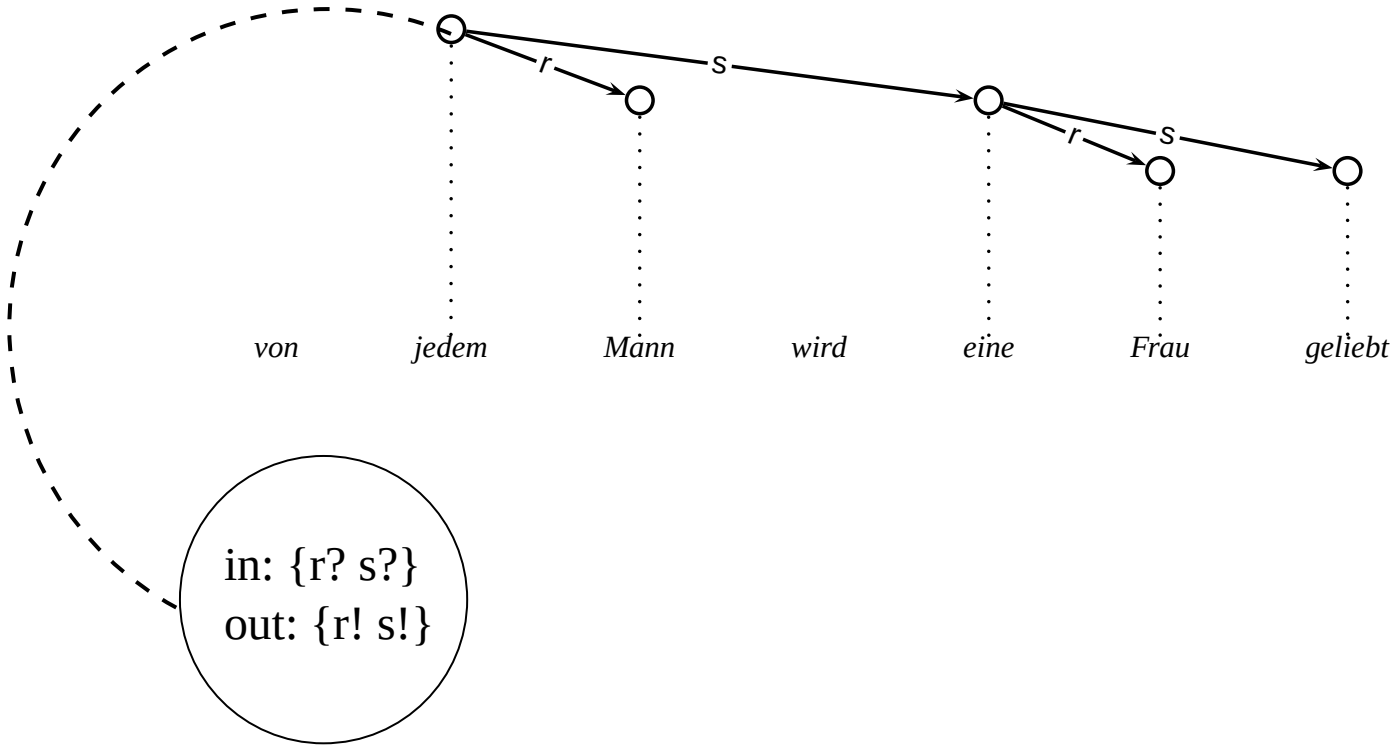
Valency principle (predicate-argument)



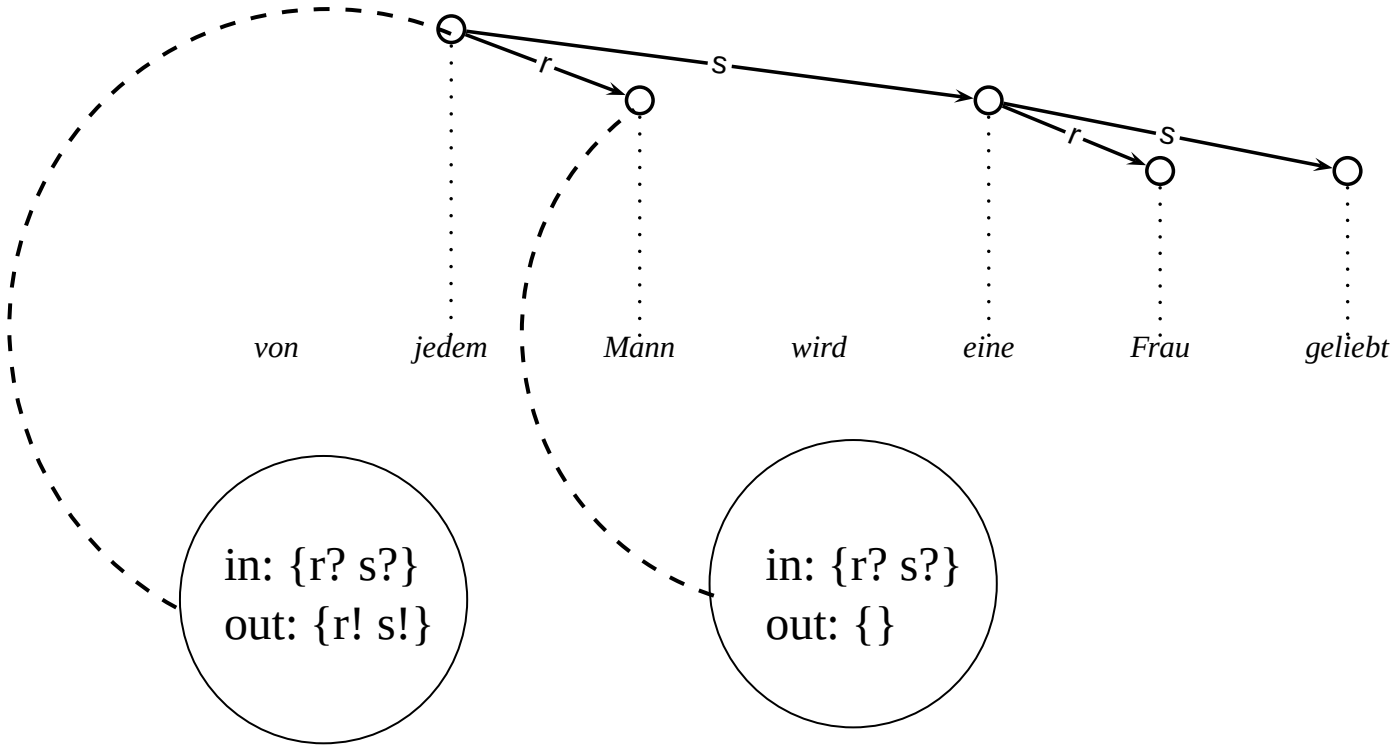
Valency principle (scope)



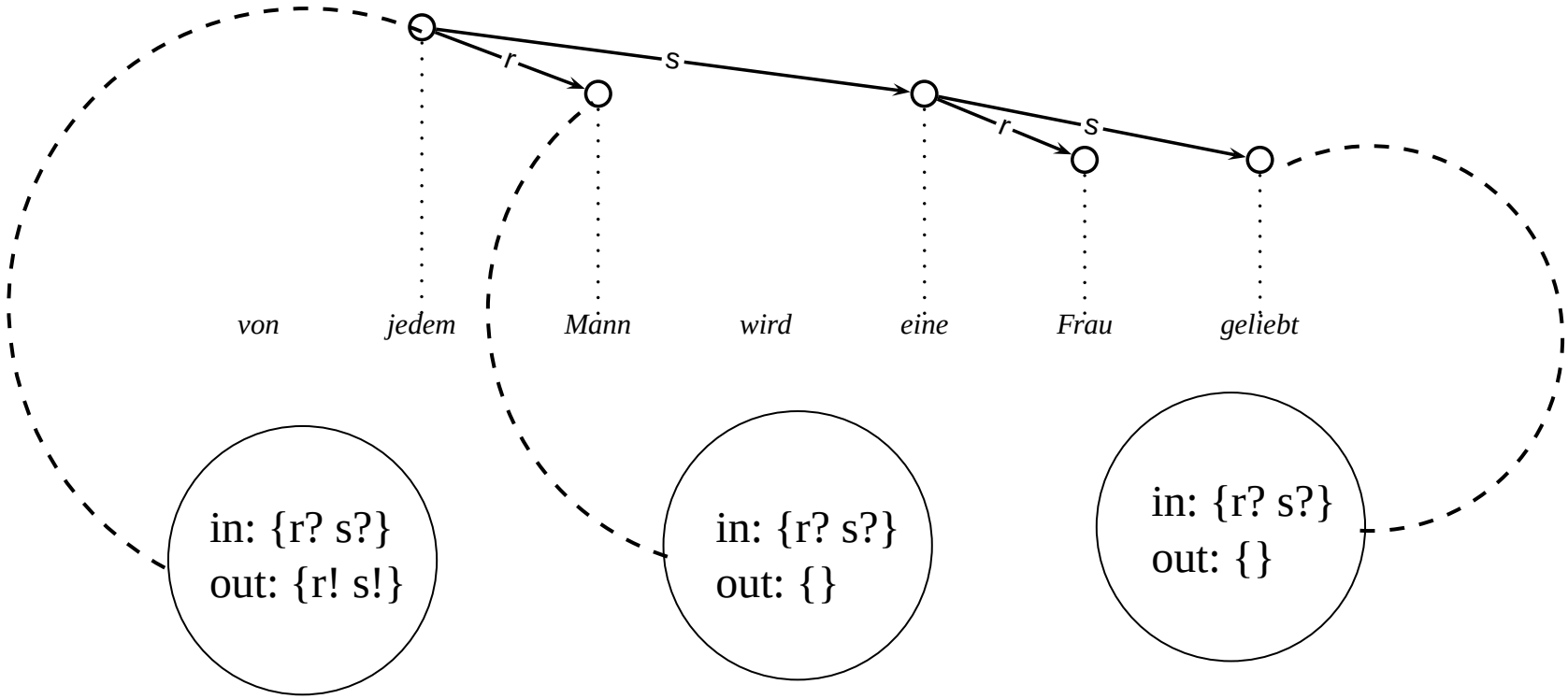
Valency principle (scope)



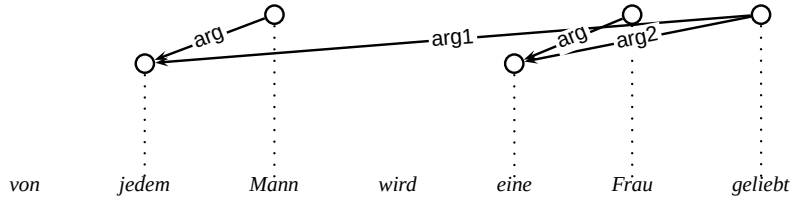
Valency principle (scope)

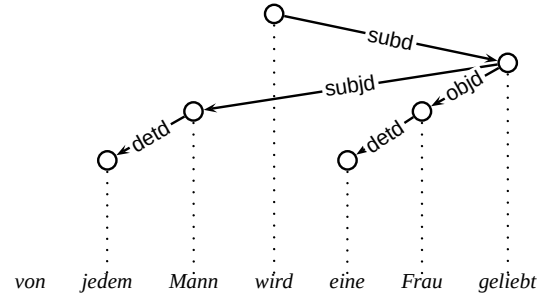


Valency principle (scope)

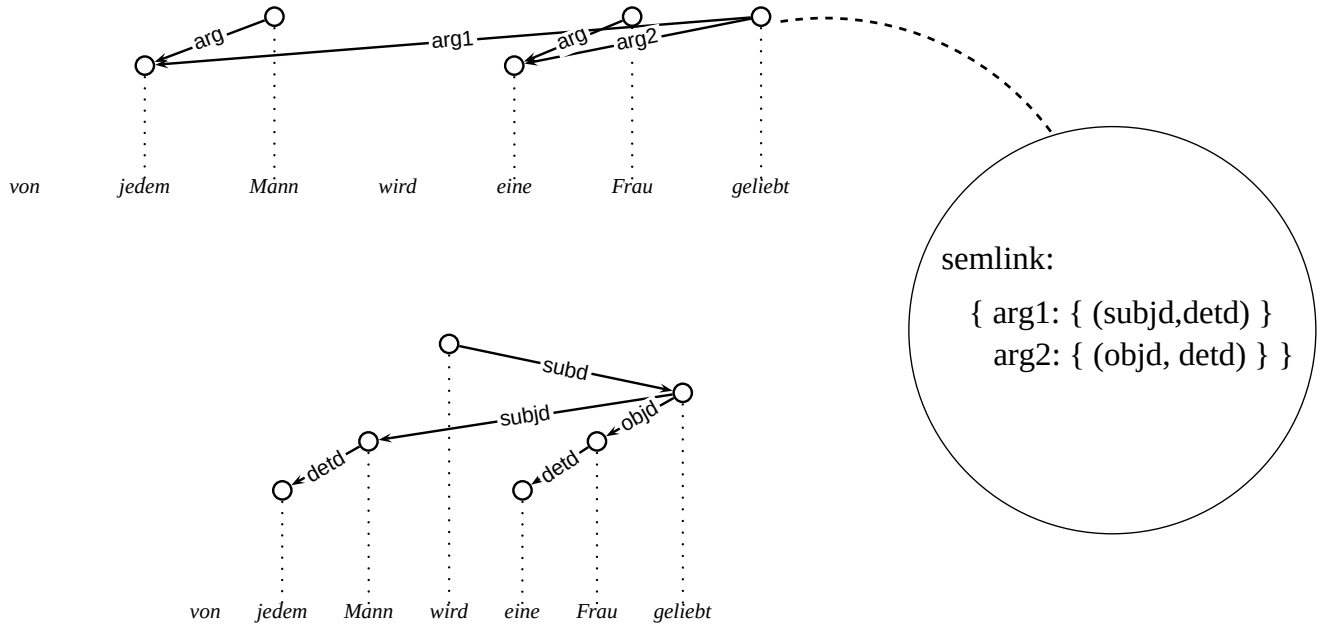


Semantic linking

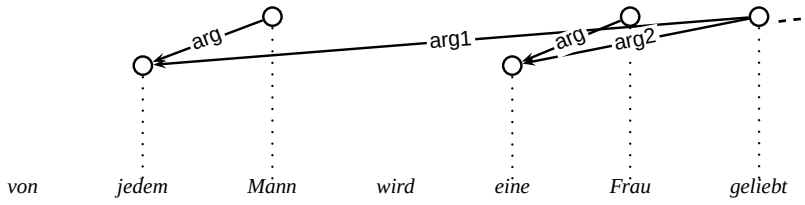


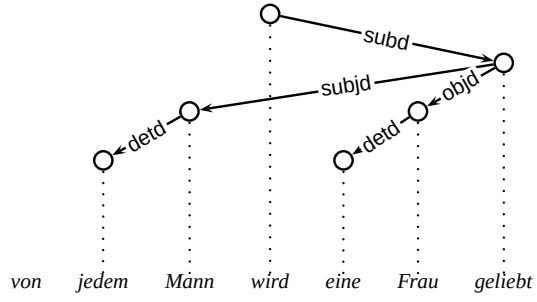


Semantic linking



Semantic linking





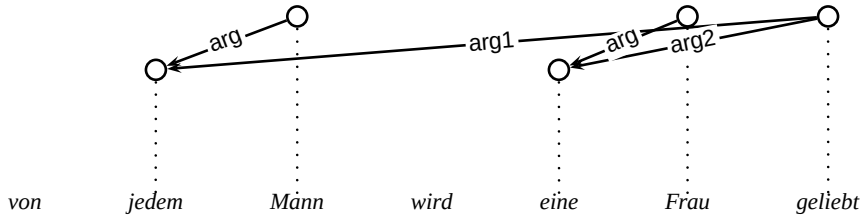
semlink:

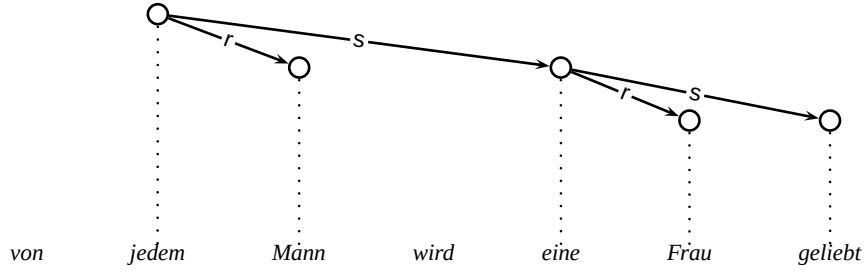
{ arg1: { (subjd,detd) }
arg2: { (objd, detd) } }

The first argument is realized by the determiner under the deep subject.

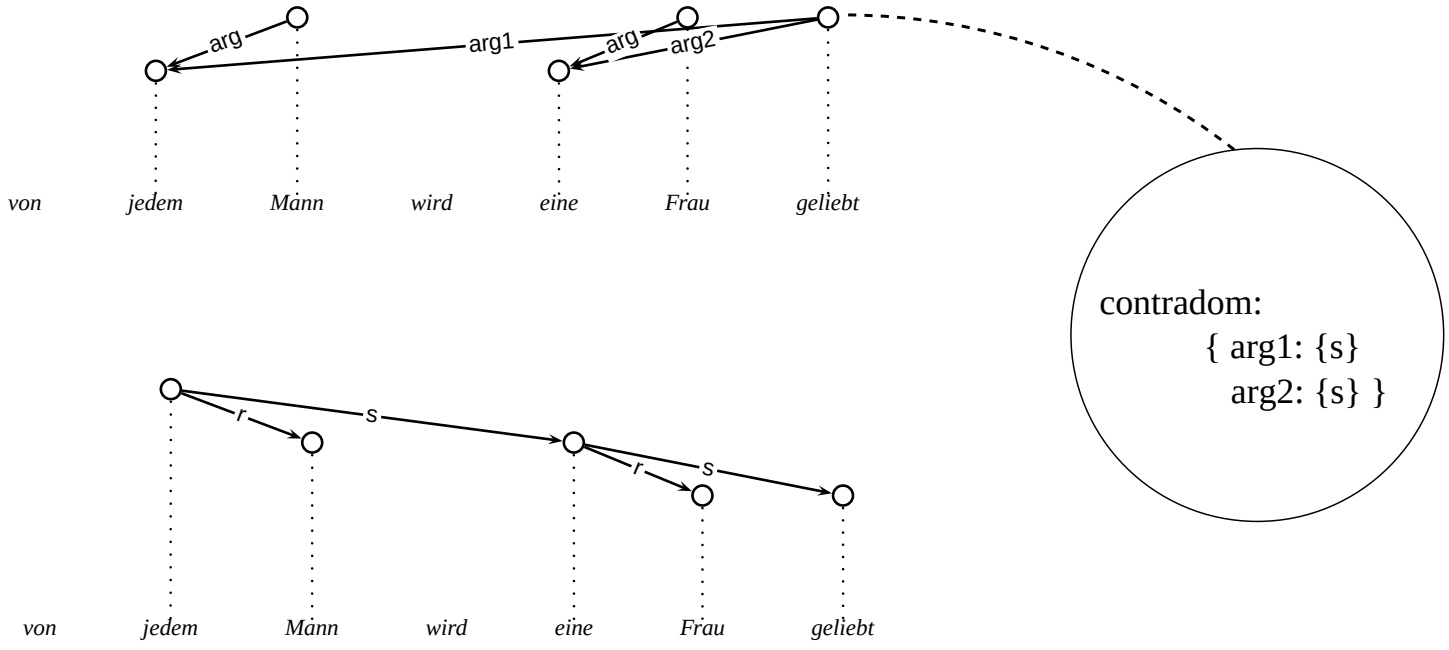
The second argument is realized by the determiner under the deep object.

Contra-dominance

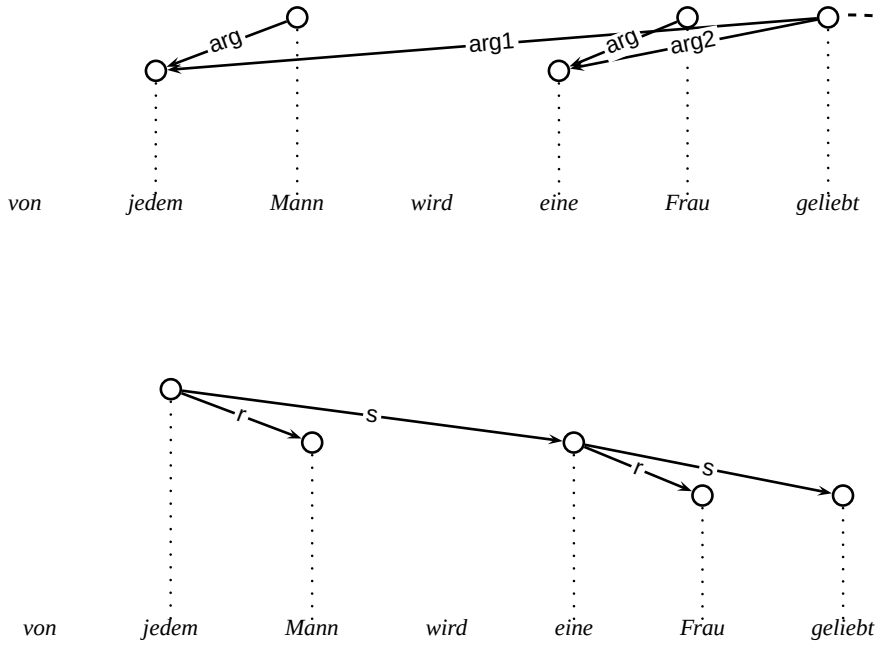




Contra-dominance



Contra-dominance



contradom:
{ arg1: {s}
arg2: {s} }

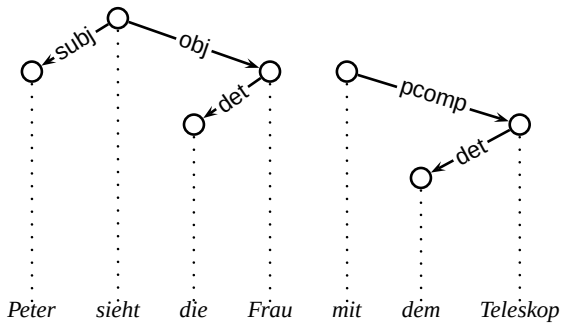
The first argument takes scope over the verb.

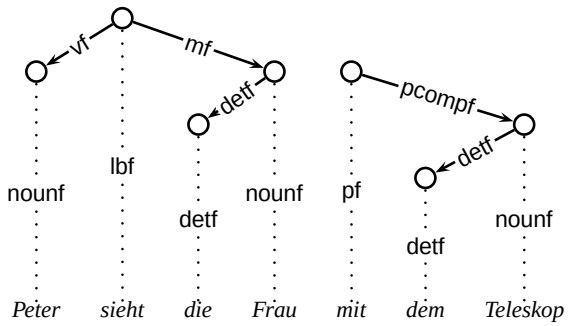
The second argument takes scope over the verb.

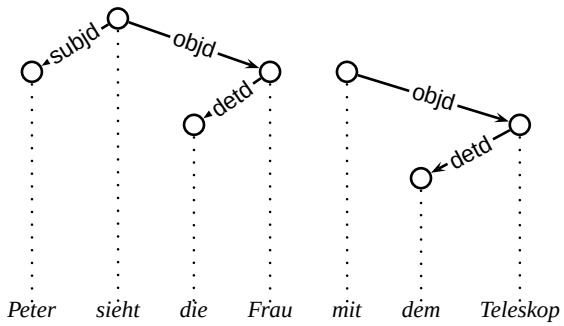
Modularity and integration

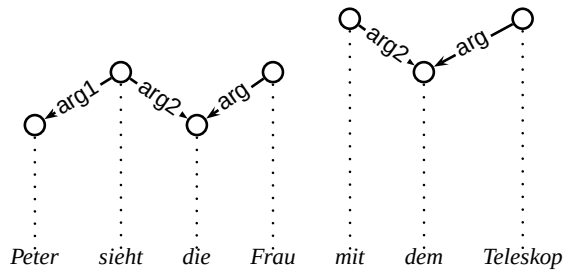
- XDG is modular *and* integrated
- modularity by unfolding the dimensions of linguistic description, one-dimensional principles
- integration by multi-dimensional principles
- integration: inferences can flow from any dimension to any other

Inferences from syntax to semantics

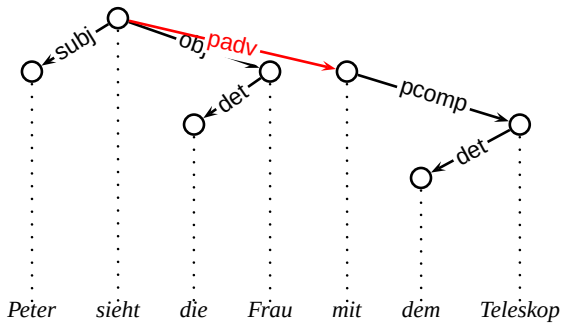


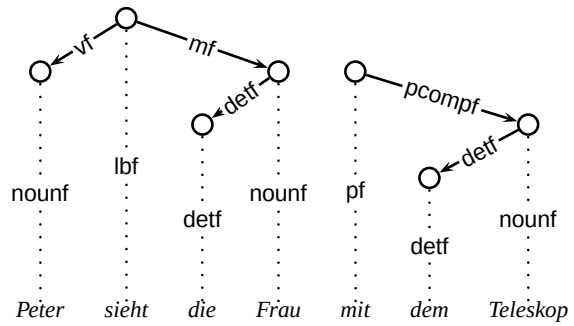


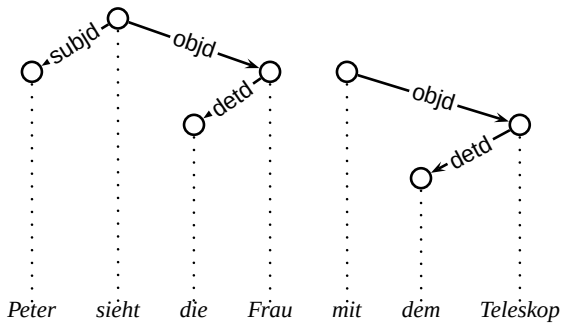


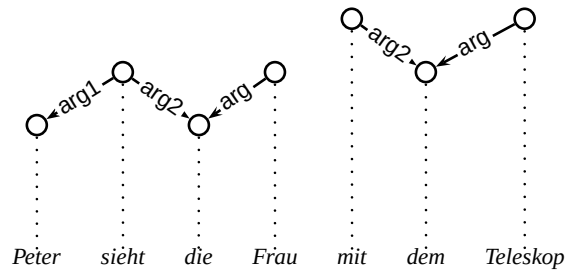


Inferences from syntax to semantics

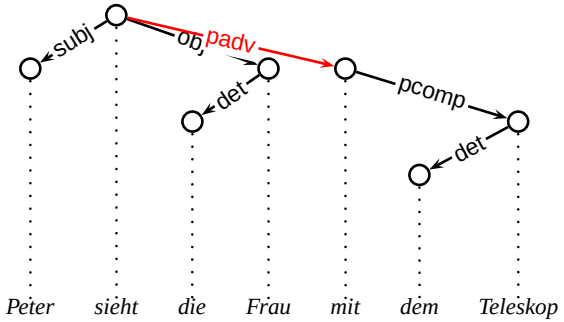


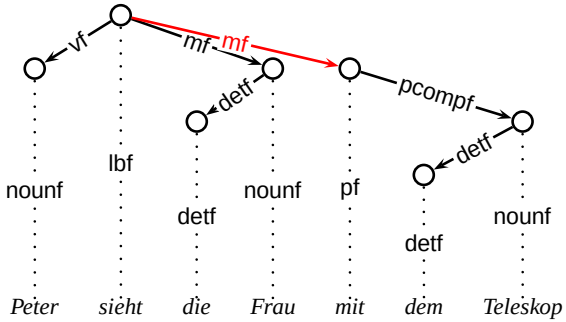


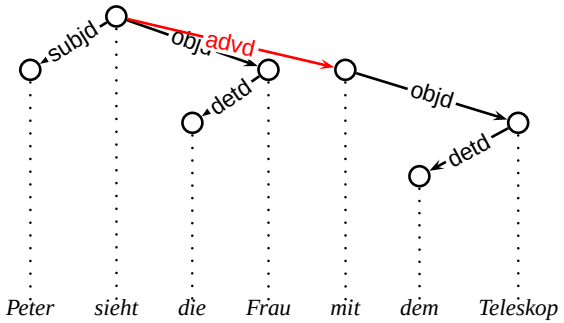


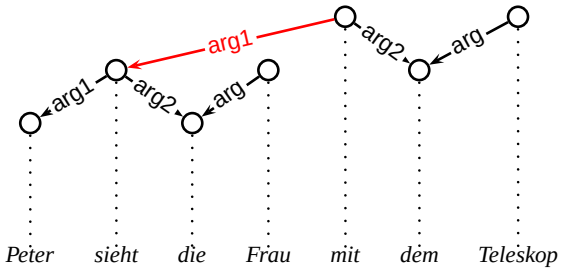


Inferences from syntax to semantics

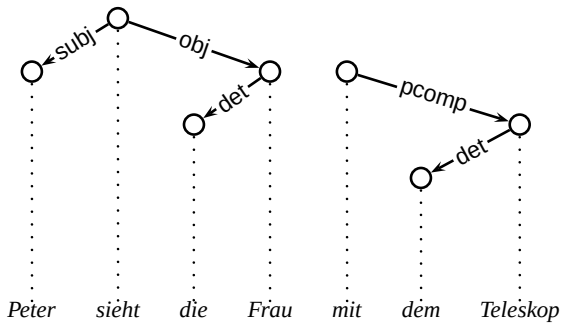


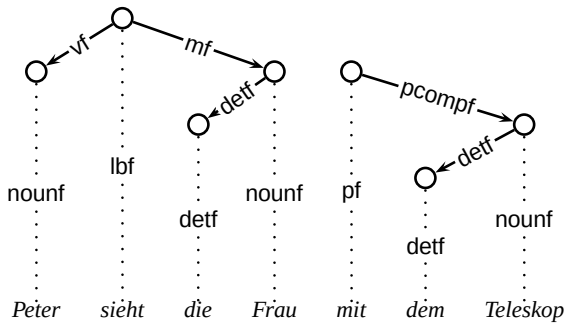


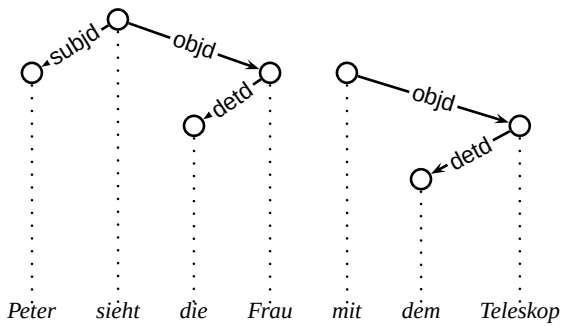


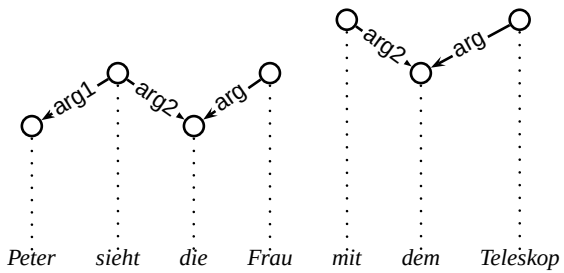


Inferences from semantics to syntax

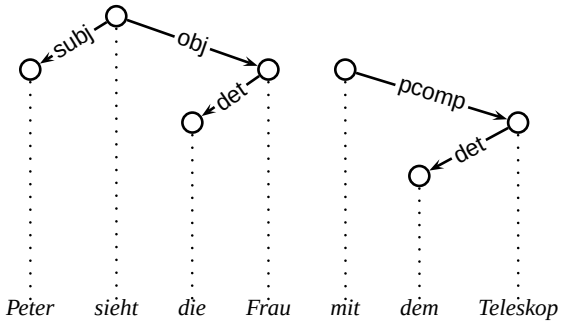


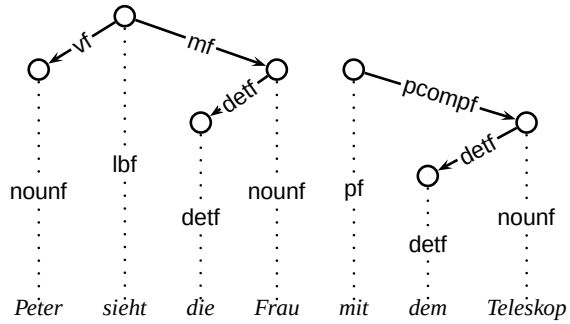


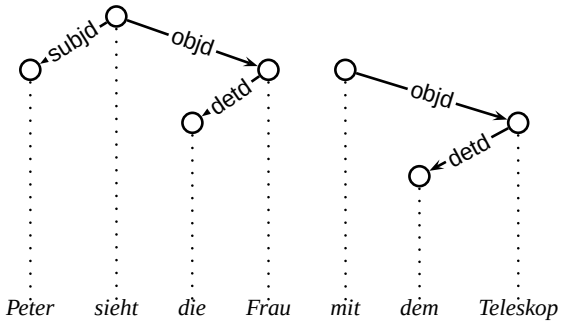


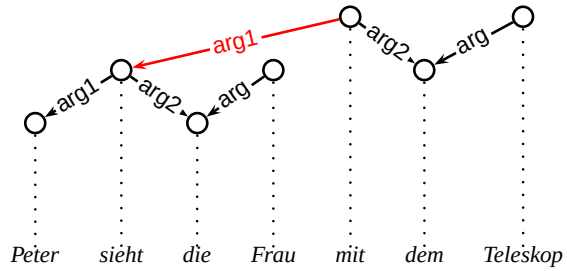


Inferences from semantics to syntax

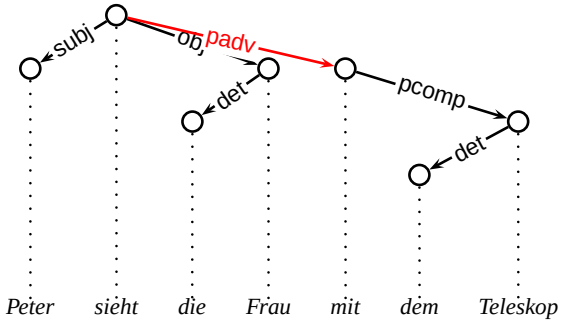


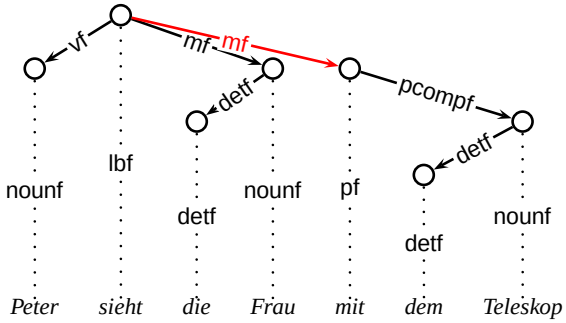


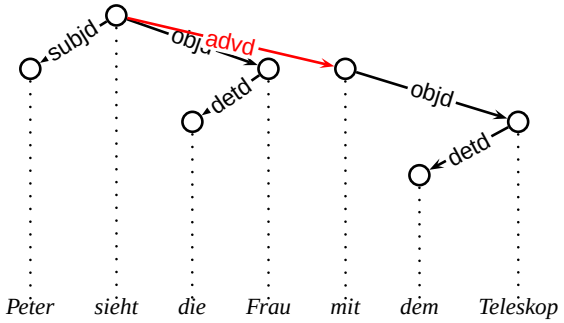


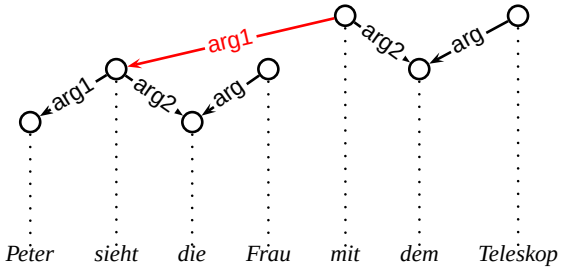


Inferences from semantics to syntax









Conclusions

- it is important to distinguish the different dimensions of linguistic description,. . .
- . . . and at the same time to integrate them.
- existing grammar formalisms cannot do both
- XDG can

Related work

- handwritten grammars for Arabic, Dutch, German, English (Duchier/Debusmann ACL 2001, Debusmann Diplom 2001, Duchier/Debusmann 2002, Bader/Foeldesi/Pfeiffer/Steigner Softwareprojekt 2004, Odeh Fopra 2004)
- syntax-semantics interface (Korthals/Debusmann COLING 2002, Debusmann PASSI 2003, Debusmann/Duchier/Koller/Kuhlmann/Smolka/Thater COLING 2004, Debusmann/Duchier/Kruijff COLING 2004 DG Workshop)
- interface to Information structure (Kruijff/Duchier EACL 2003)
- grammar induction (NEGRA (German): Korthals Diplom 2003, Möhl Fopra 2004, PDT (Czech): Bojar 2004)
- TAG to XDG encoding (generation: Koller/Striegnitz ACL 2002, parsing: Debusmann/Duchier/Kuhlmann/Thater TAG+7 2004)

Future work

- obtain large-scale grammars
- find out how to parse them efficiently with XDG
- many many more issues to solve (e.g. coordination, incremental parsing. . .)
- but before all that: start writing the dissertation ;-)

Thank you!