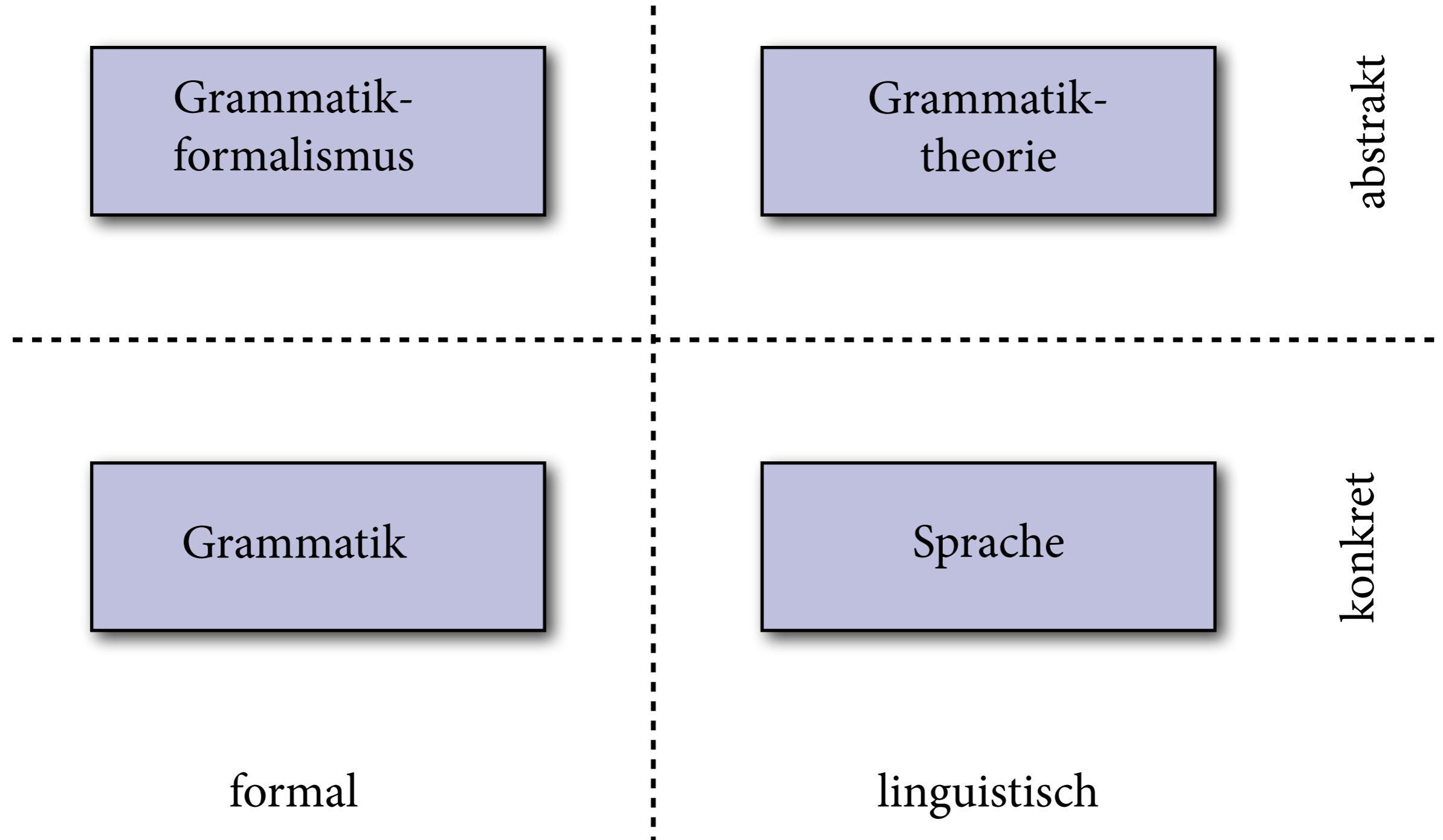


Baumadjunktionsgrammatiken

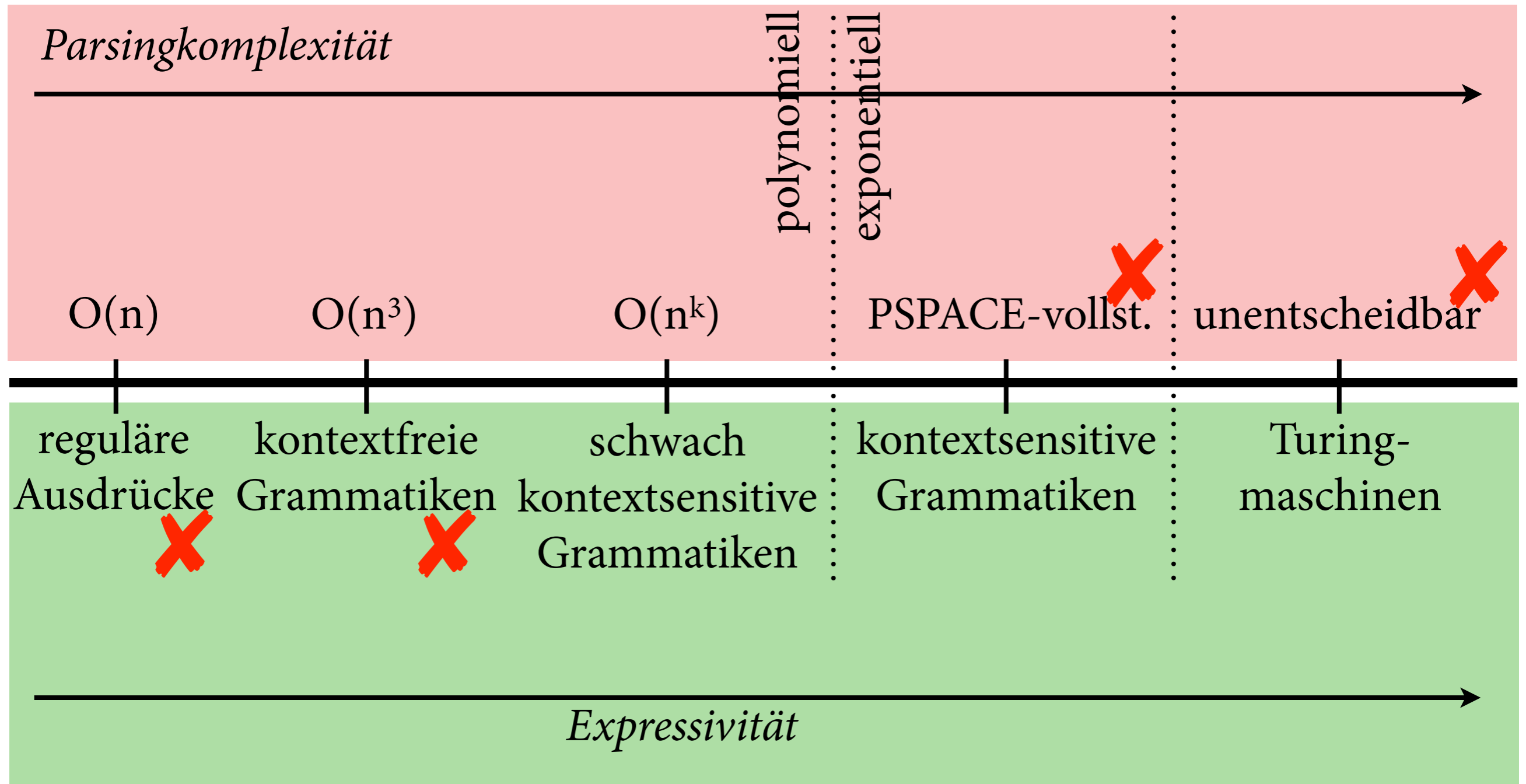
Vorlesung “Grammatikformalismen”
Alexander Koller

28. April 2015

Grammatikformalismen



Natürliche Sprachen in der Chomsky-Hierarchie



Letztes Mal

VP → IV

VP → TV NP_a

VP → DV NP_d NP_a

IV → schläft

TV → isst

DV → gibt

NP_a → den N_{masc}

NP_d → dem N_{masc}

NP_a → die N_{fem}

NP_d → der N_{fem}

NP_a → das N_{neut}

NP_d → dem N_{neut}

Die Katze jagt den Vogel.

S → NP_{nom₁} TV NP_{acc₂}

Der Vogel wird gejagt.

S → NP_{nom₂} wird TV_{part}

Letztes Mal

VP \rightarrow NP_n NP_a NP_d V

VP \rightarrow NP_n NP_d NP_a V

VP \rightarrow NP_a NP_n NP_d V

VP \rightarrow NP_d NP_n NP_a V

VP \rightarrow NP_a NP_d NP_n V

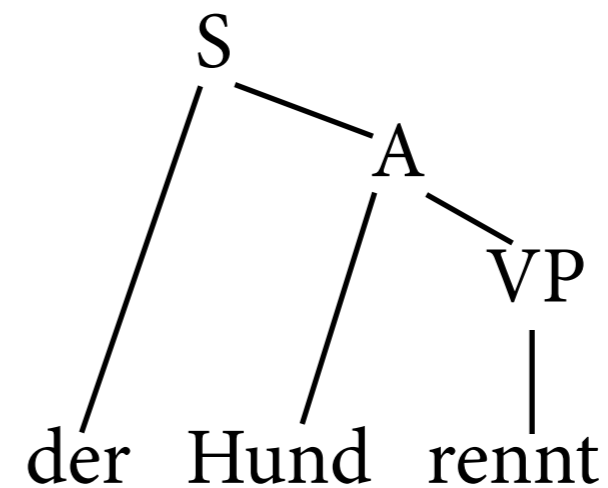
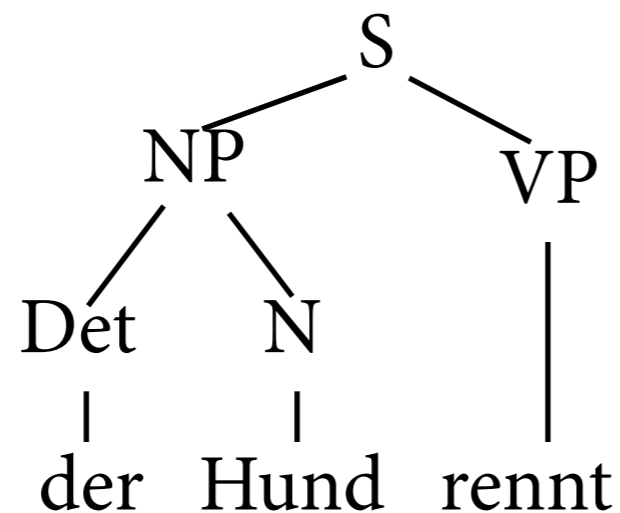
VP \rightarrow NP_d NP_a NP_n V

Lexikalisierung

- Lexikalisierte Grammatikformalismen:
 - ▶ jedes Stück grammatische Information ist mit einem konkreten Wort verknüpft
 - ▶ für jedes Wort gibt es endlich viele Lexikoneinträge, die grammatische Informationen vollständig darstellen
- Vorteile:
 - ▶ Grammatikentwicklung: grammatischer Beitrag jedes Wortes an einer Stelle gesammelt
 - ▶ Parsing: Parser muss für gegebenen Eingabestring nur endliche Menge von Objekten manipulieren
- Problem von kfG: i.a. *nicht* lexikalisiert

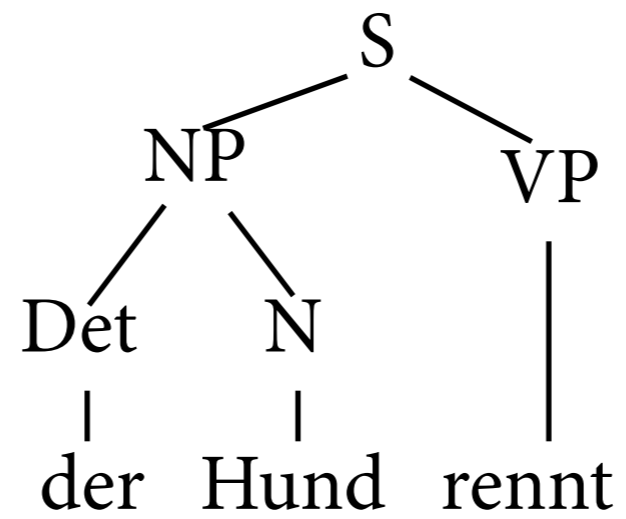
Lexikalisierung

- Greibach-NF:
 - ▶ ist lexikalisiert (genau ein Terminalsymbol pro Regel)
 - ▶ ist schwach äquivalent (gleiche Stringsprache)
 - ▶ ist *nicht* stark äquivalent (Parsebäume evtl. verschieden)

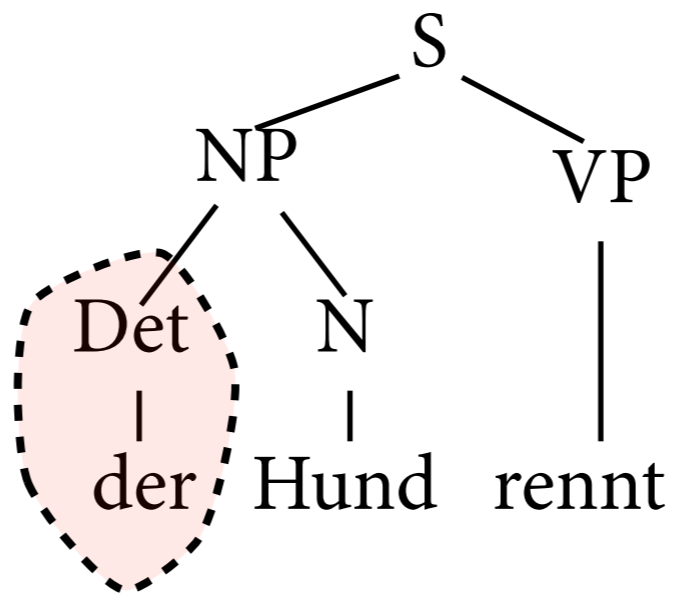


- Kann man kfGen stark lexikalisieren?

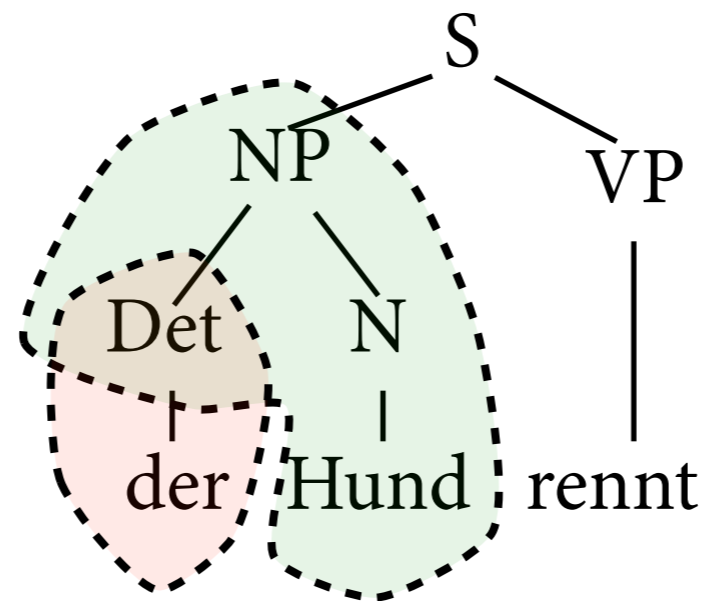
Ein Versuch



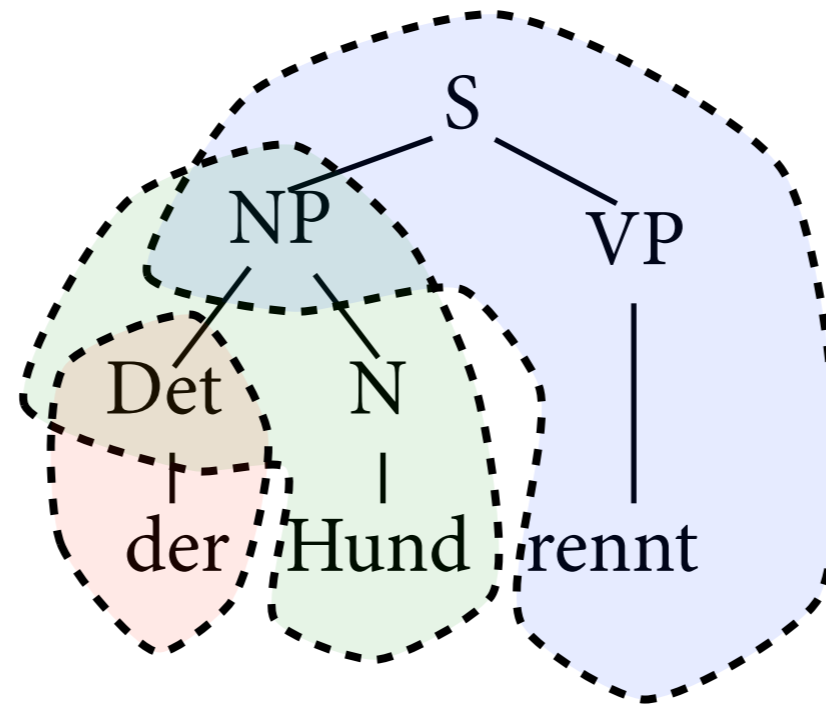
Ein Versuch



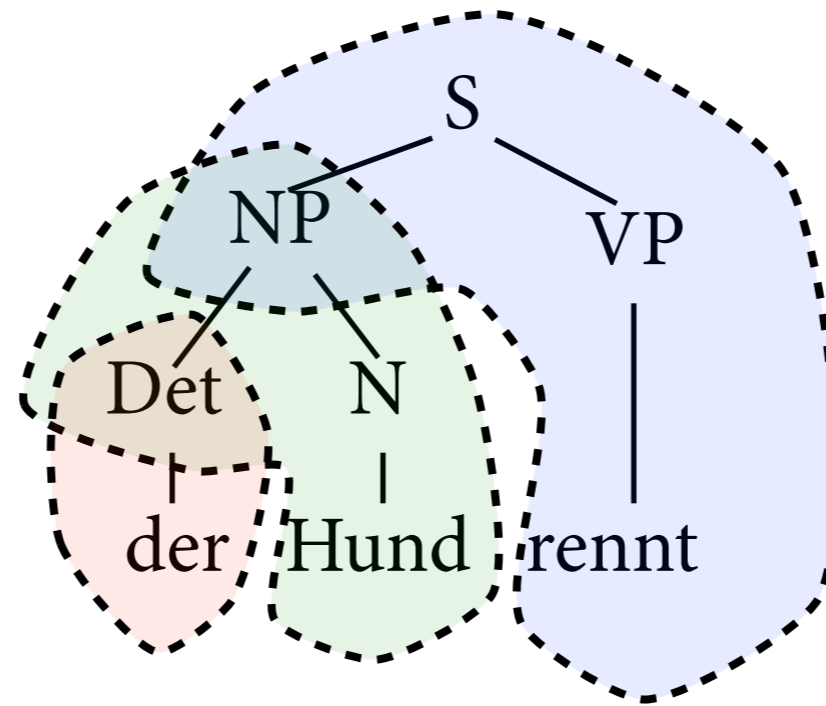
Ein Versuch



Ein Versuch

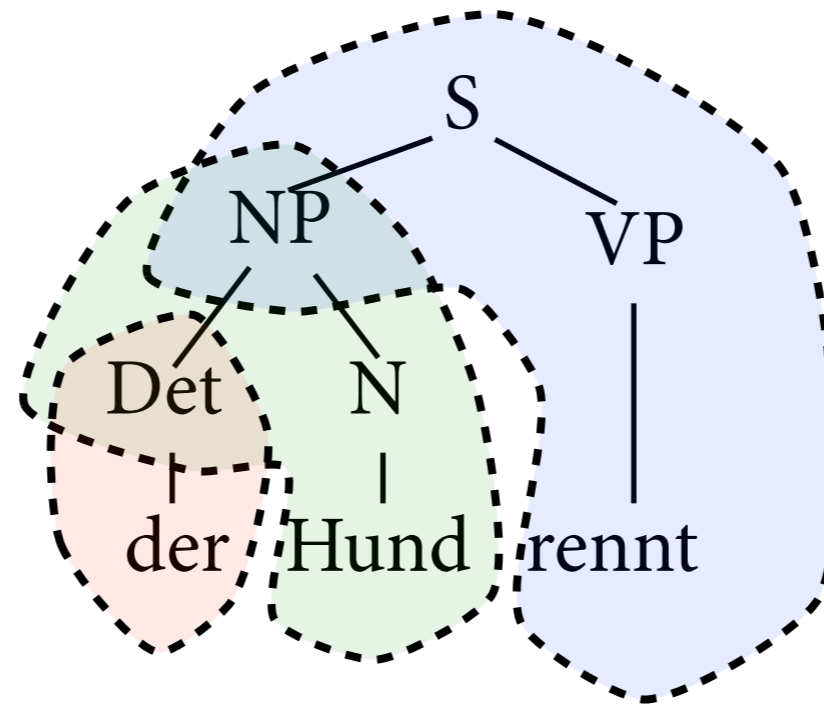


Ein Versuch

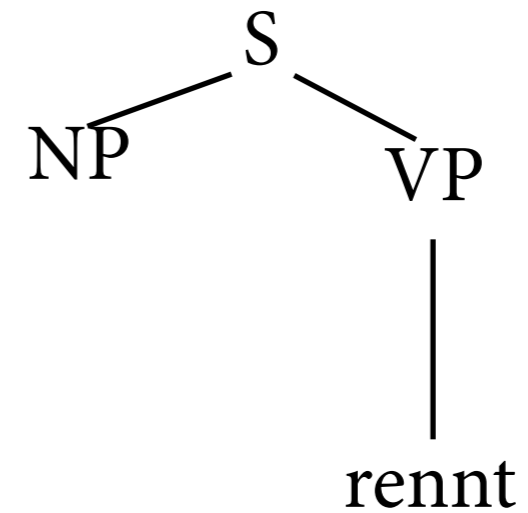


Baum kann man wie folgt zusammenbauen:

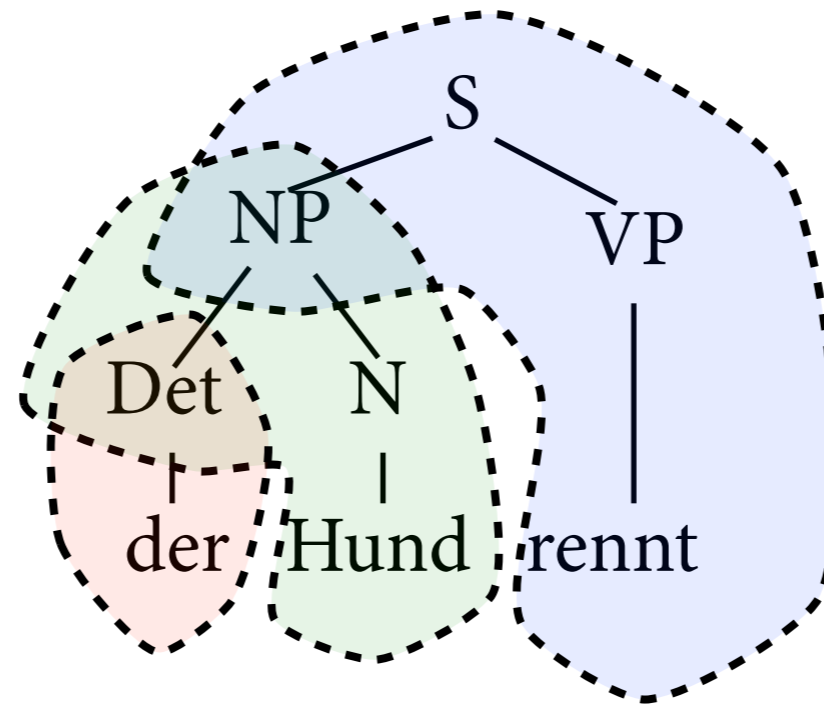
Ein Versuch



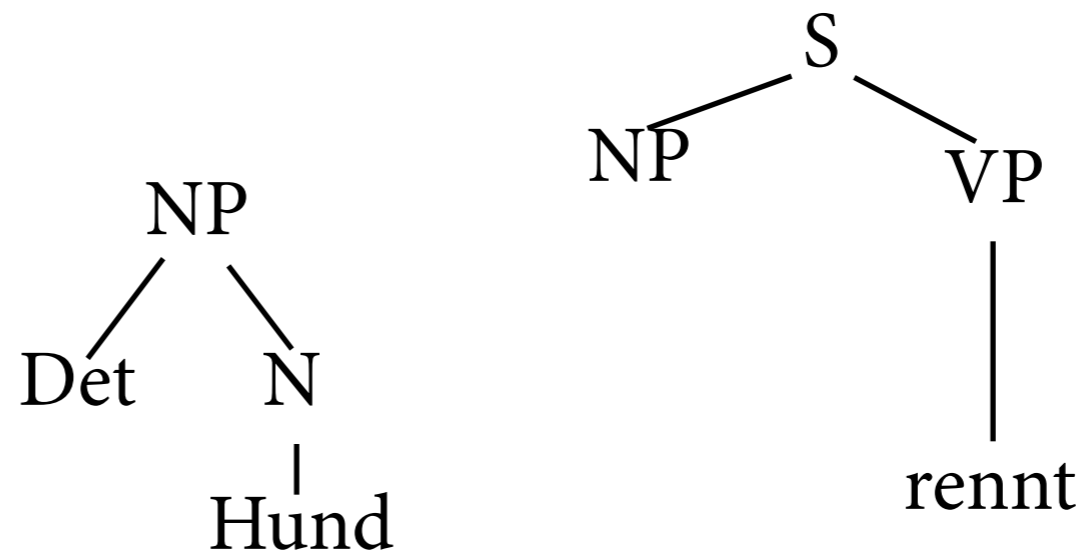
Baum kann man wie folgt zusammenbauen:



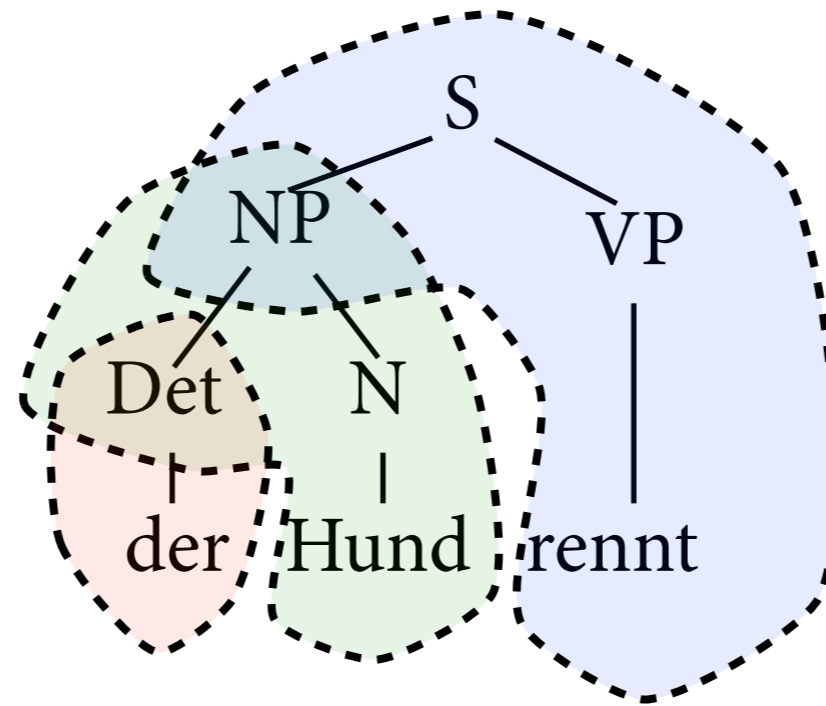
Ein Versuch



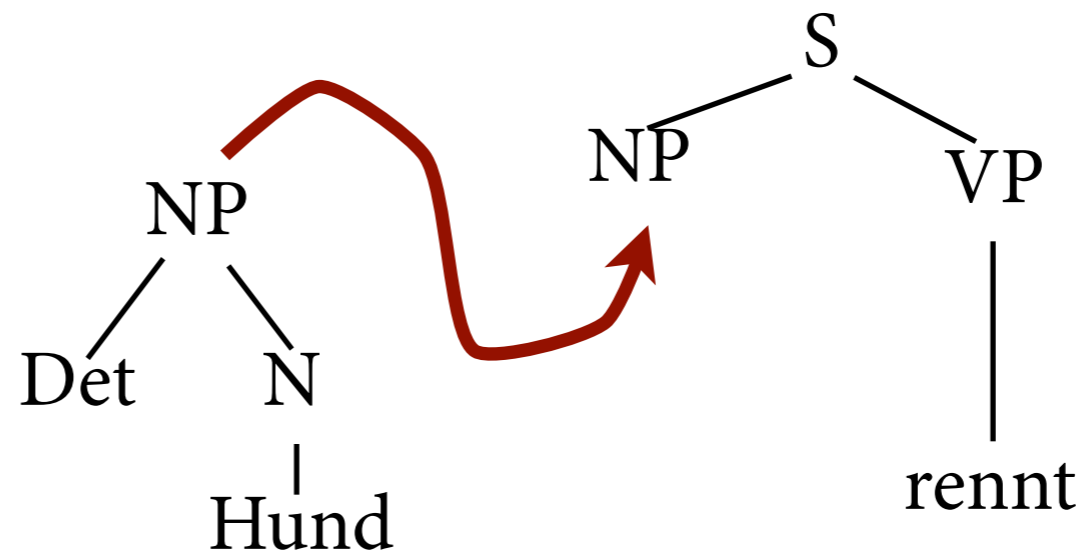
Baum kann man wie folgt zusammenbauen:



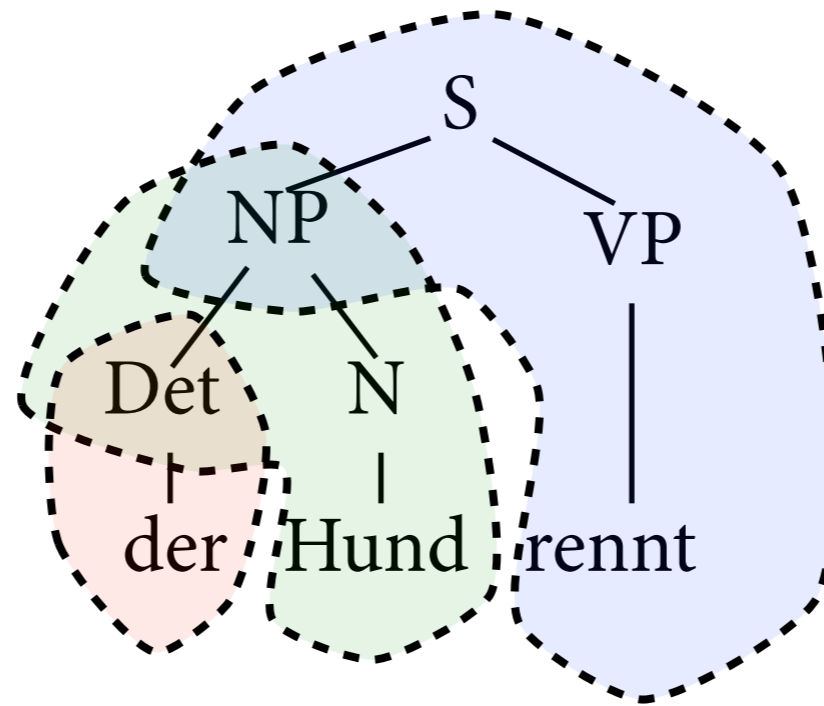
Ein Versuch



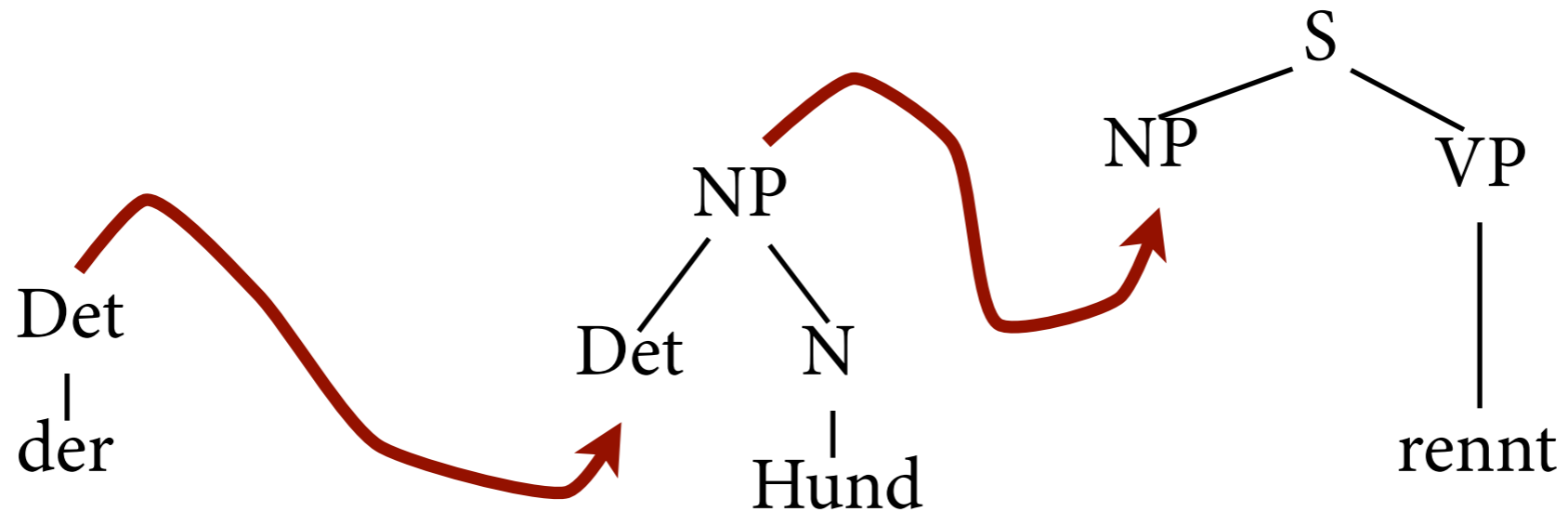
Baum kann man wie folgt zusammenbauen:



Ein Versuch



Baum kann man wie folgt zusammenbauen:

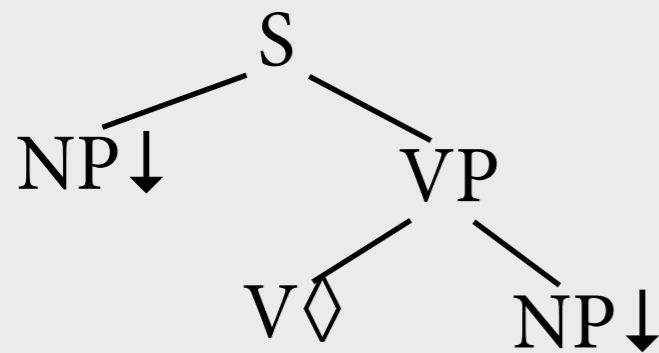


Baumsubstitutionsgrammatiken

- *Baumsubstitutionsgrammatik*
(tree substitution grammar, TSG):
endliche Menge von *Elementarbäumen*.
- Knoten von Elementarbäumen:
 - ▶ *innere Knoten*, mit NT-Symbolen markiert
 - ▶ *lexikalische Anker*, mit Terminalsymbolen markierte Blätter
(manchmal mit Raute markiert: $A\Diamond$)
 - ▶ *Substitutionsknoten*, mit NT-Symbolen markierte Blätter
(normalerweise mit Pfeil markiert: $A\Downarrow$)

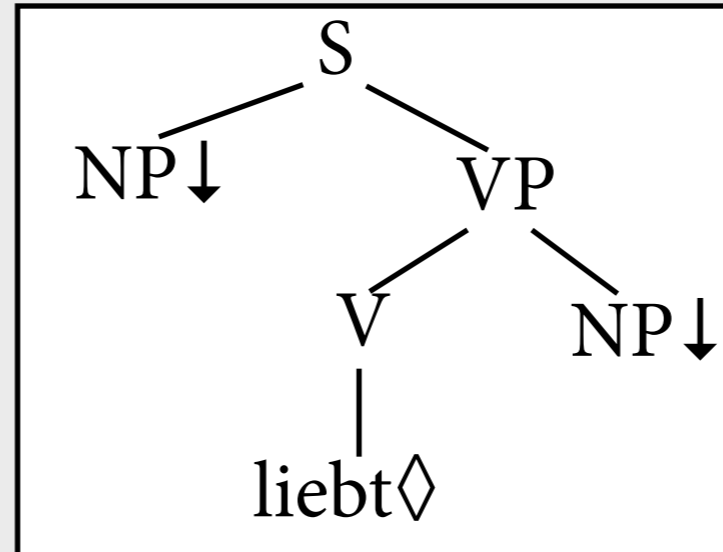
Lexikalisierte Elementarbäume

unlexikalisiert

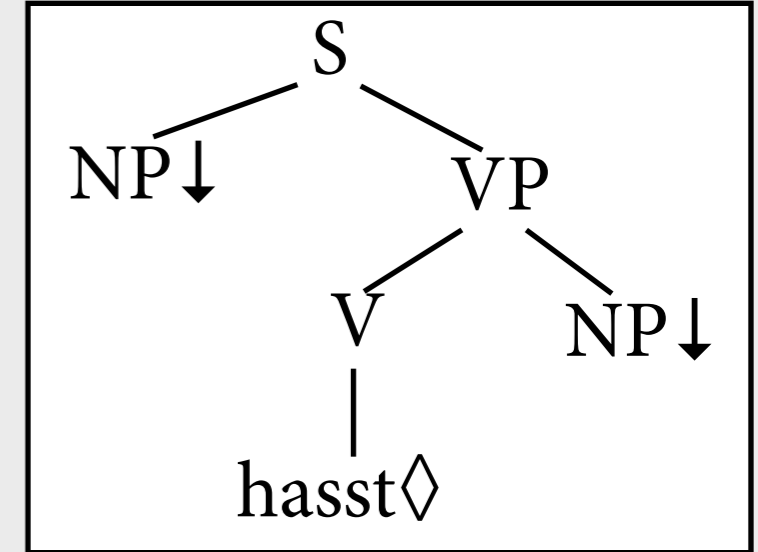


α_1

lexikalisiert



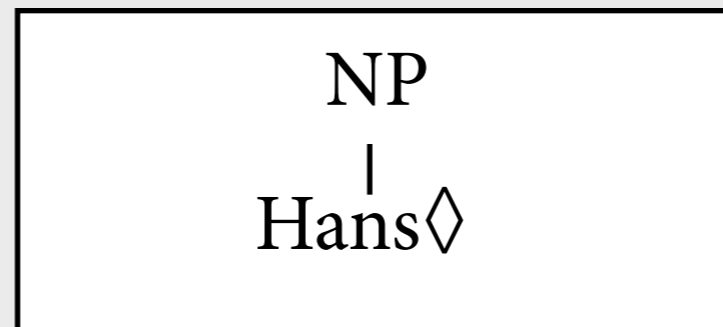
$\alpha_1(\text{liebt})$



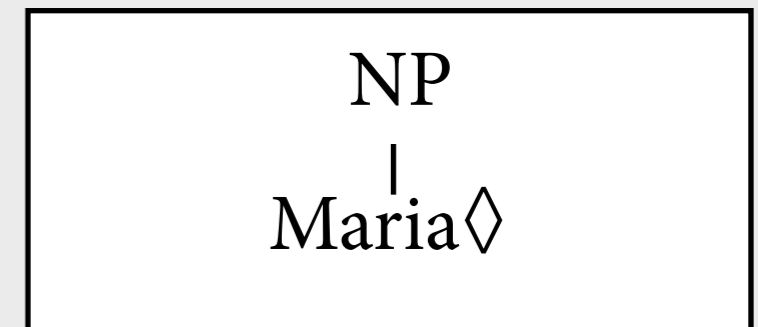
$\alpha_1(\text{hasst})$

NP◇

α_2



$\alpha_2(\text{Hans})$

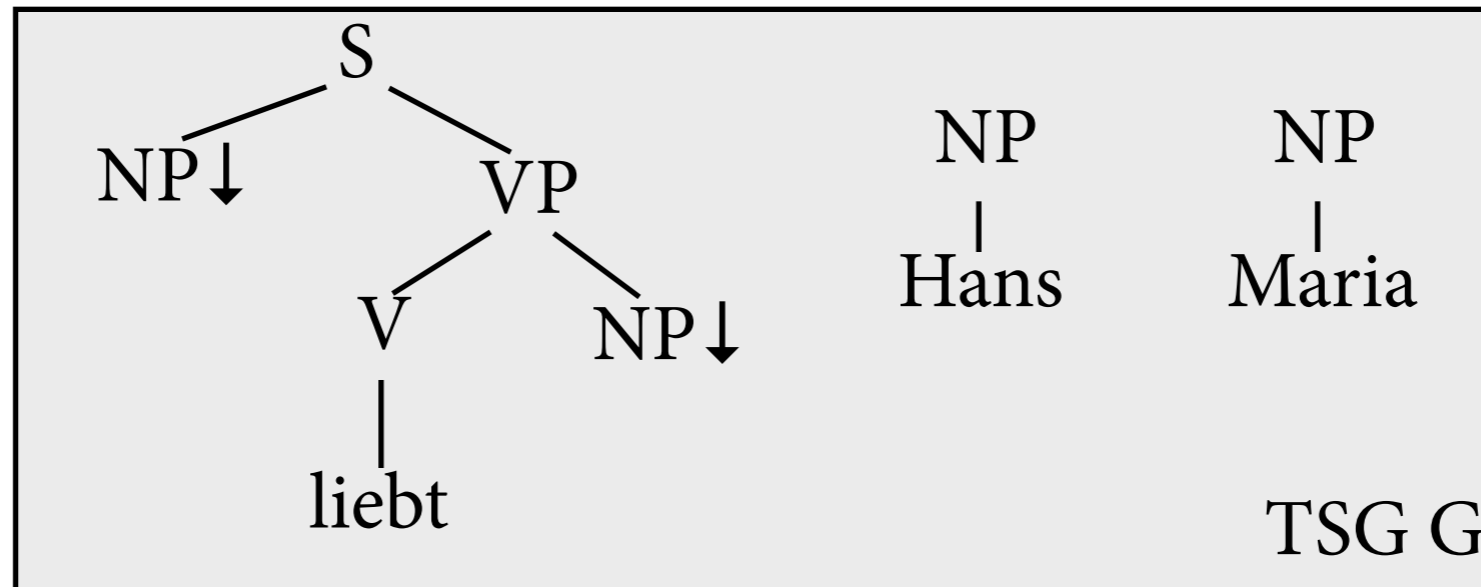


$\alpha_2(\text{Maria})$

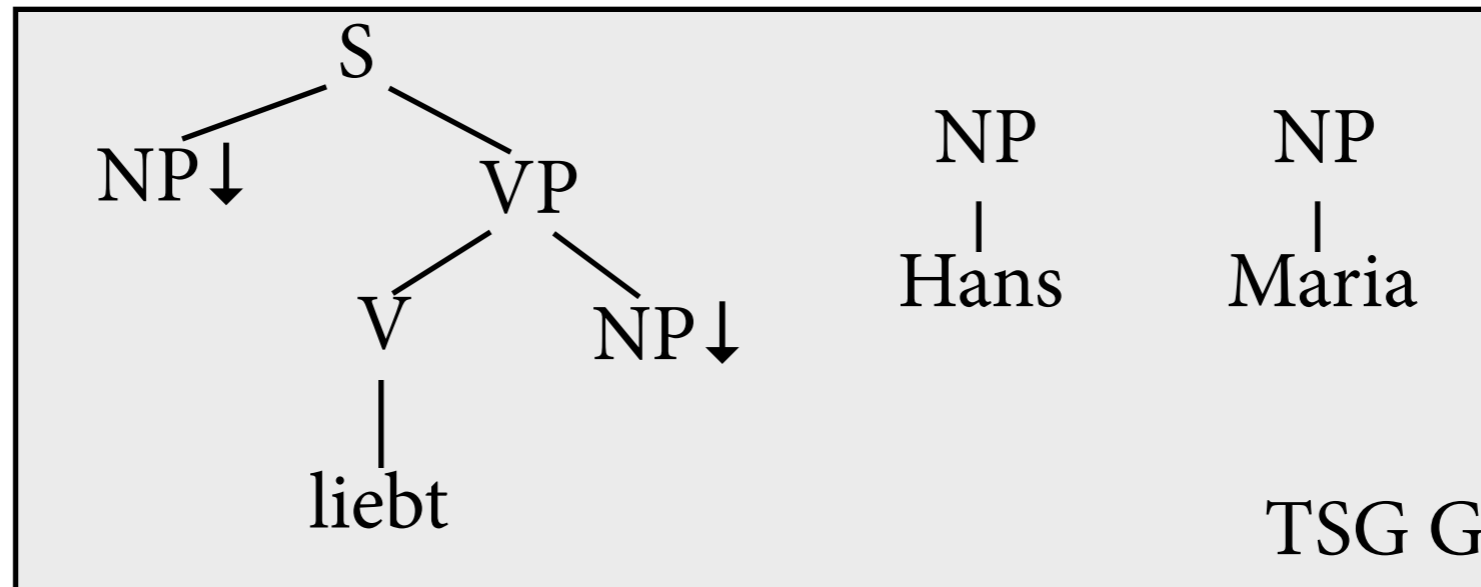
TSGs: Ableitungen

- Ableitungsschritt, $t \Rightarrow t'$:
 - ▶ t ein Baum, der einen Subst.knoten u mit Label A hat
 - ▶ e ein Elementarbaum in TSG mit Wurzel-Label A
 - ▶ t' entsteht aus t durch Ersetzen von u durch e
- Baum t heißt ein *abgeleiteter Baum* der TSG G , wenn t keine Substitutionsknoten enthält und $S \downarrow \Rightarrow^* t$.
- *Baumsprache* $T(G)$ von G :
Menge aller abgeleiteten Bäume.

TSGs: Ein Beispiel

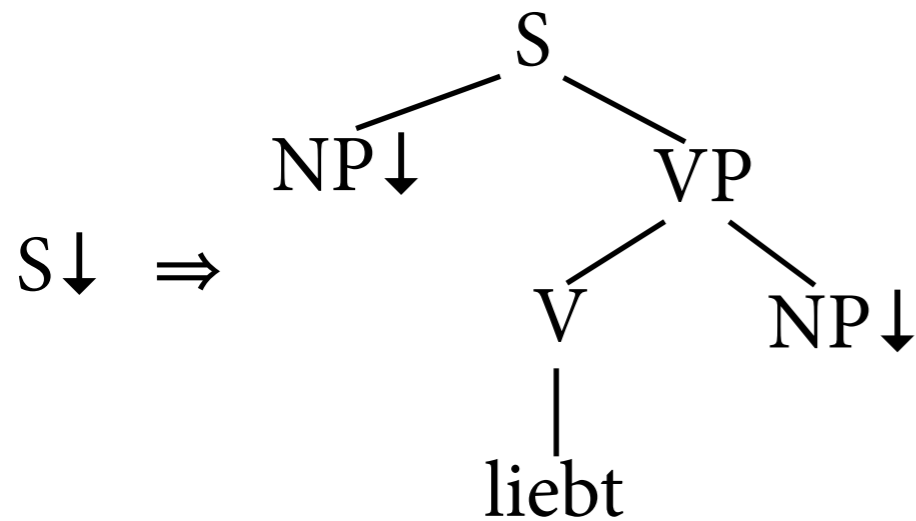
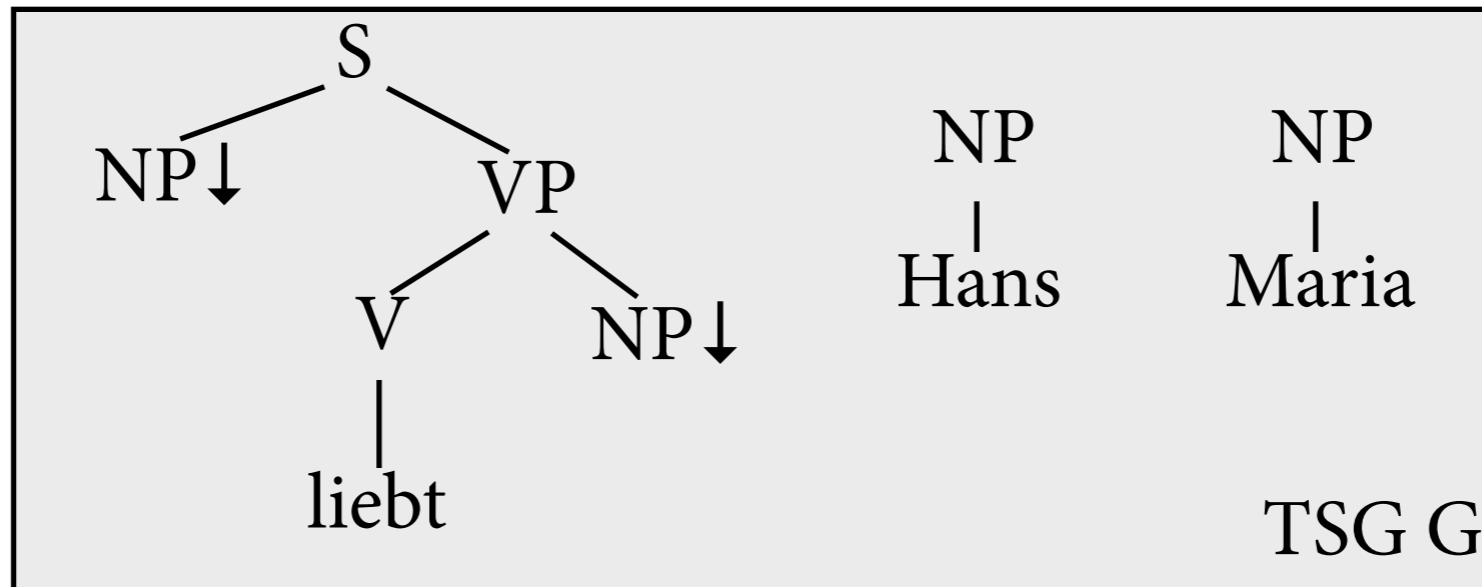


TSGs: Ein Beispiel

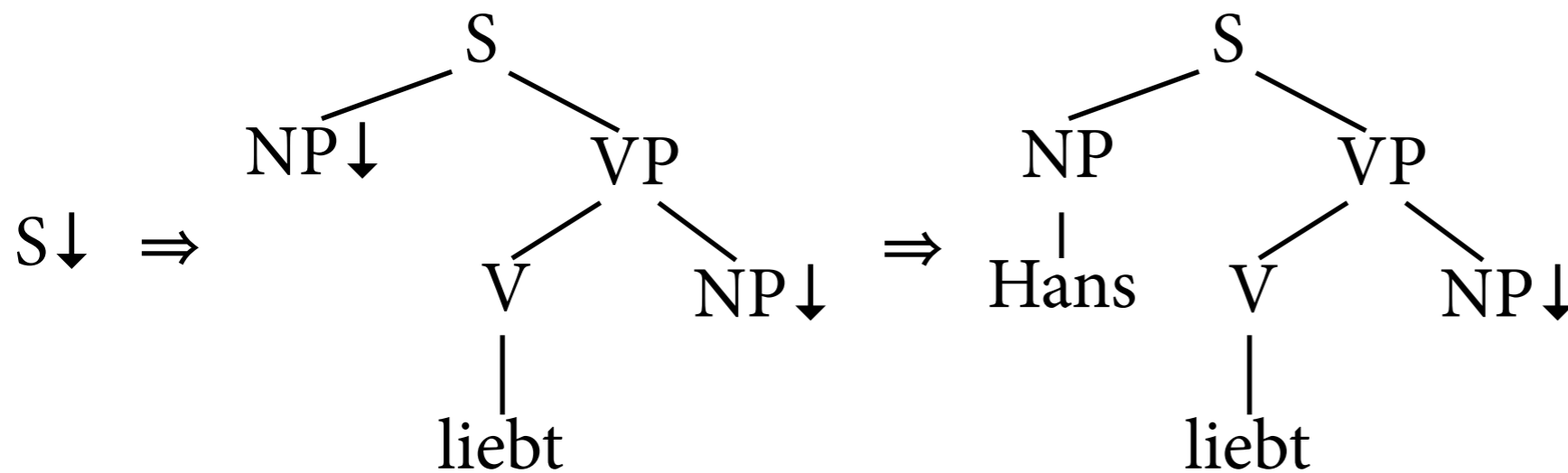
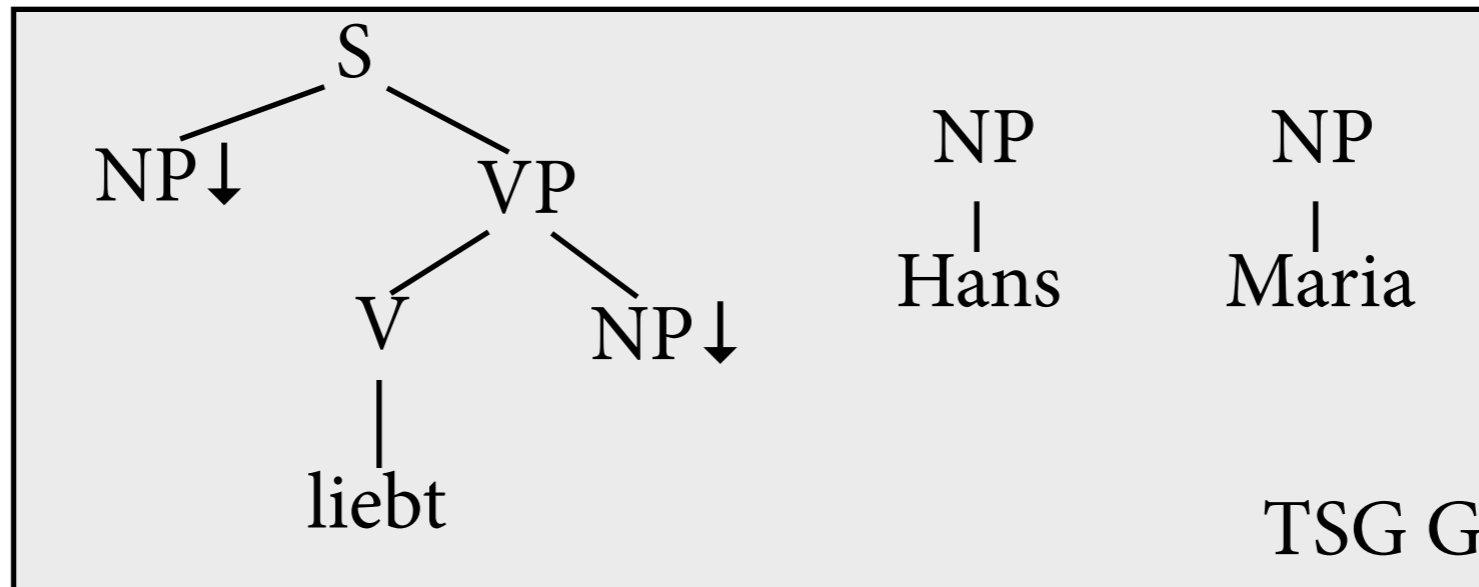


S↓

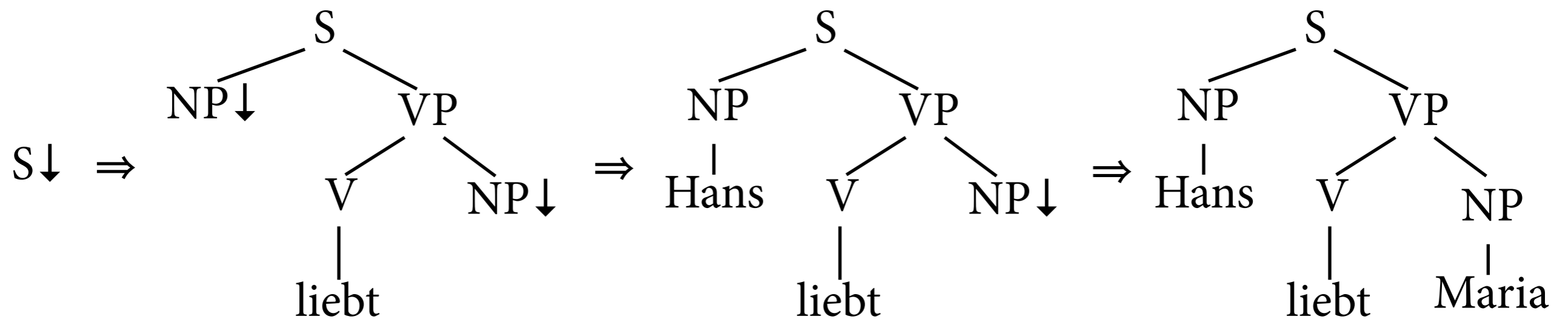
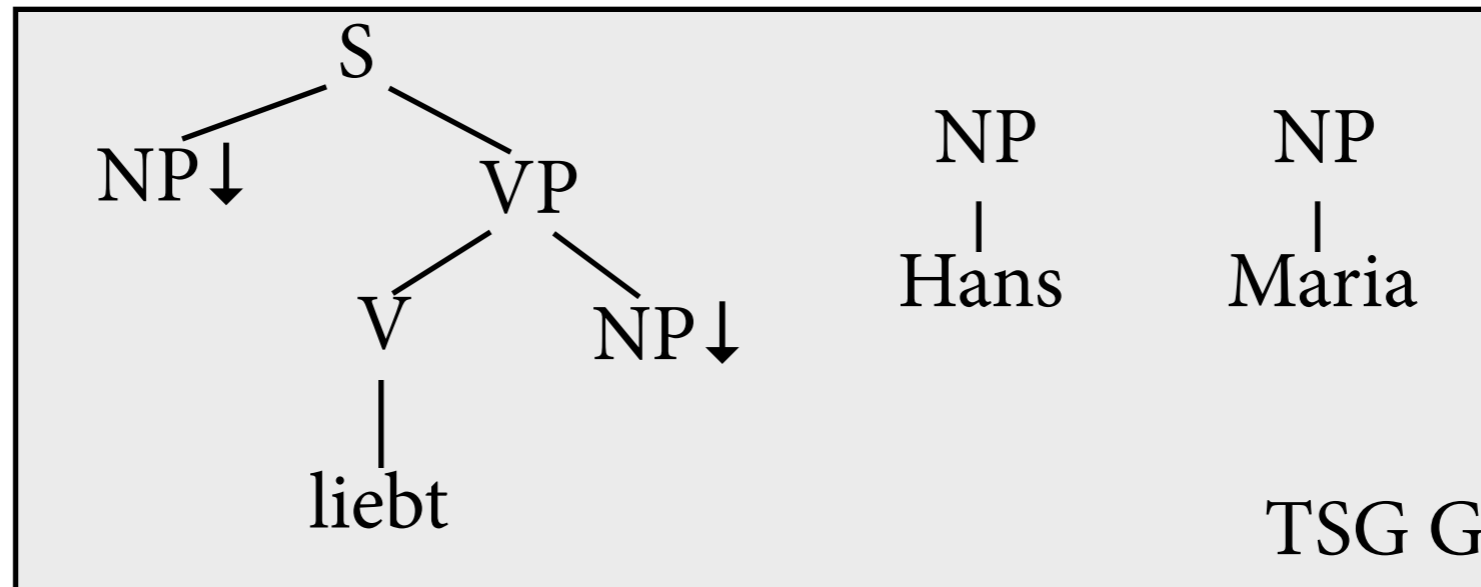
TSGs: Ein Beispiel



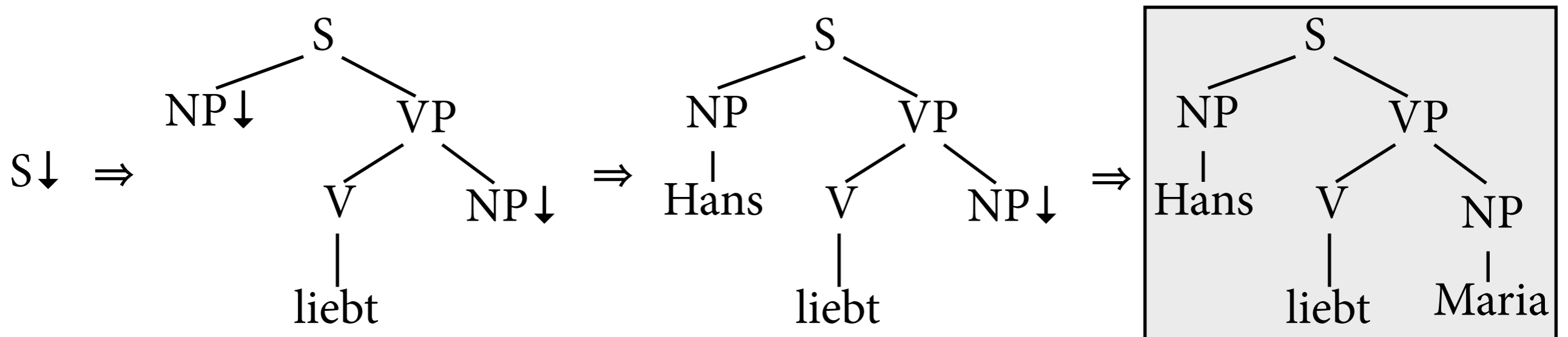
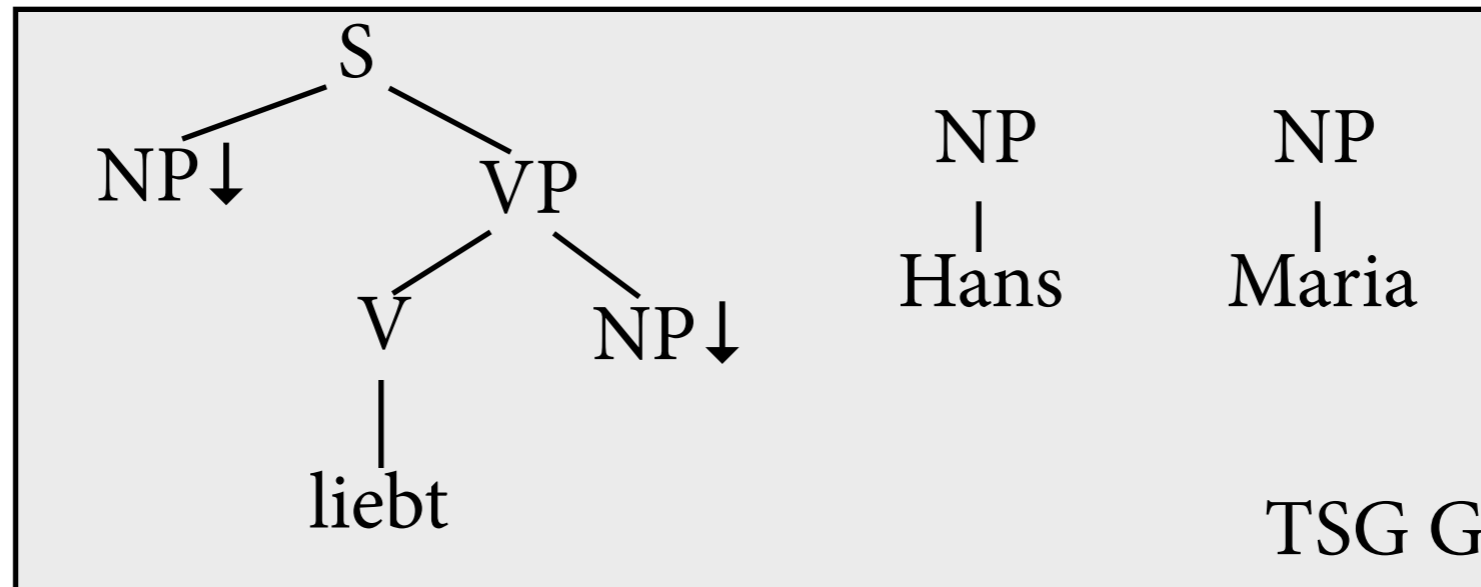
TSGs: Ein Beispiel



TSGs: Ein Beispiel

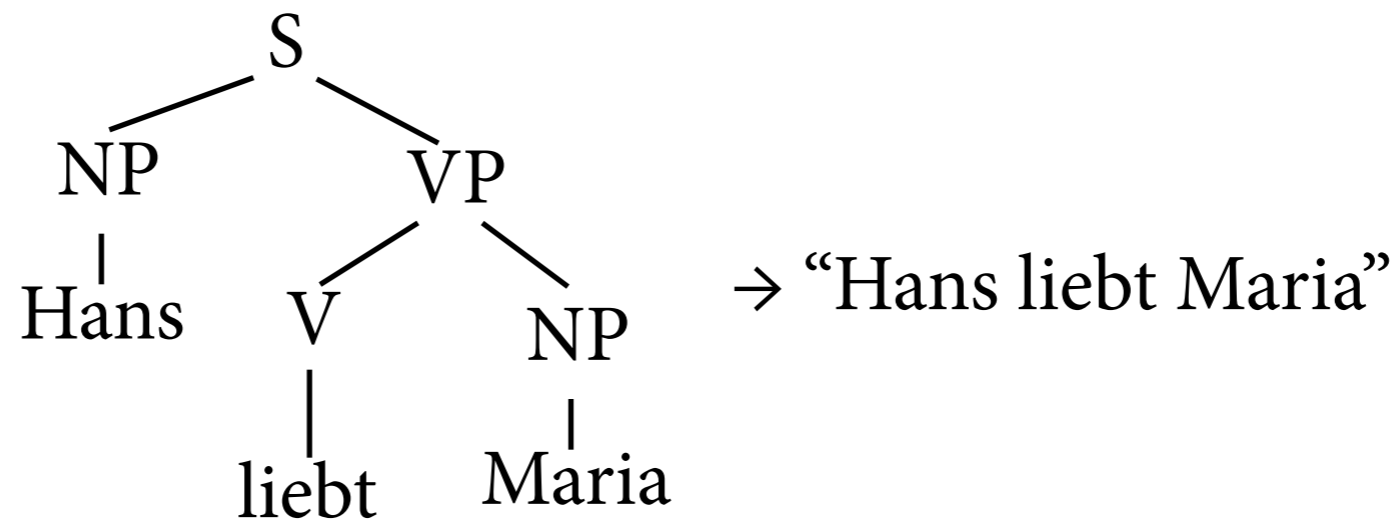


TSGs: Ein Beispiel



TSGs: Stringsprachen

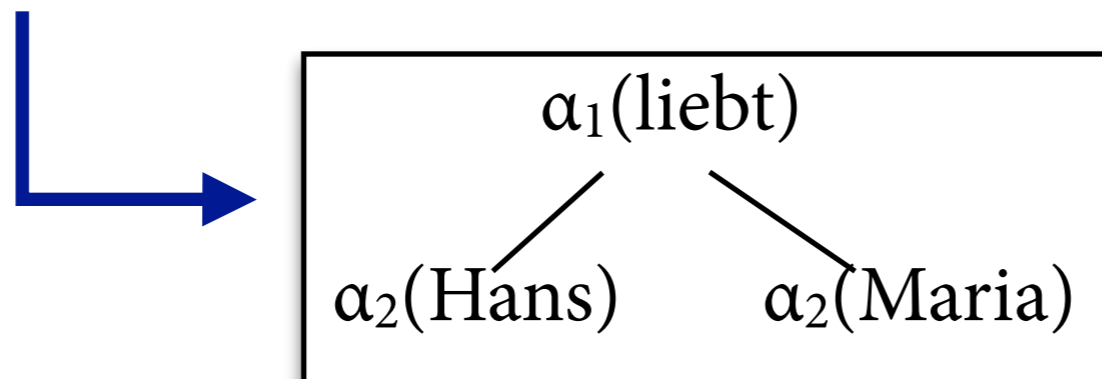
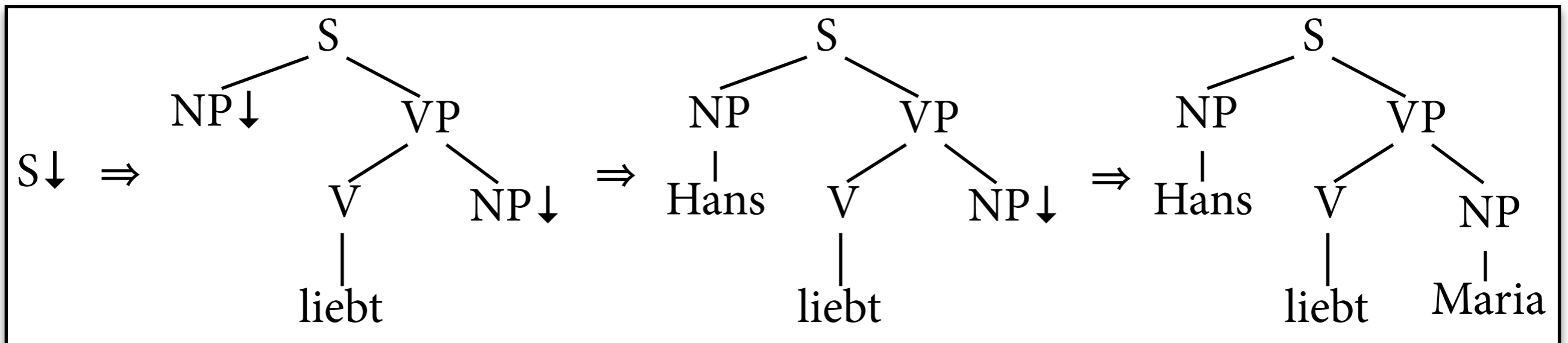
- Zu jedem Baum kann man den *Ertrag* als den String definieren, den man von links nach rechts von den Blättern abliest.



- Stringsprache: $L(G) = \{\text{ertrag}(t) \mid t \in T(G)\}$

Ableitungsbaum

- Ableitungsbaum: Darstellung der Struktur einer TSG-Ableitung.
 - ▶ Knoten: Elementarbäume
 - ▶ Kanten: Substitution



Lexikalisierte TSG

- Elementarbaum heißt lexikalisiert, wenn er einen lexikalischen Anker enthält.
- TSG heißt lexikalisiert, wenn alle Elementarbäume lexikalisiert sind.
- Genügt TSG, um kfGs stark zu lexikalisieren?
 - ▶ d.h.: Gibt es zu jeder kfG eine lexikalisierte TSG, so dass abgeleitete Bäume der TSG = Parsebäume der kfG?

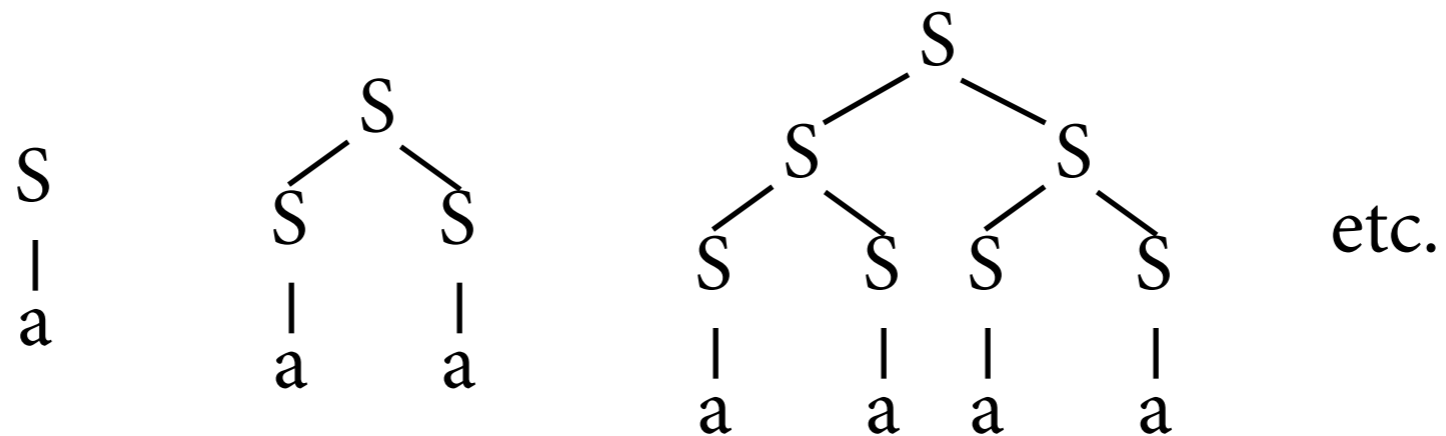
Nein!

- Gegenbeispiel (Schabes):

$$S \rightarrow S S$$

$$S \rightarrow a$$

- Pfad zum minimal tiefen Blatt wird beliebig lang, aber ist in lexikalisierten TSG beschränkt lang.

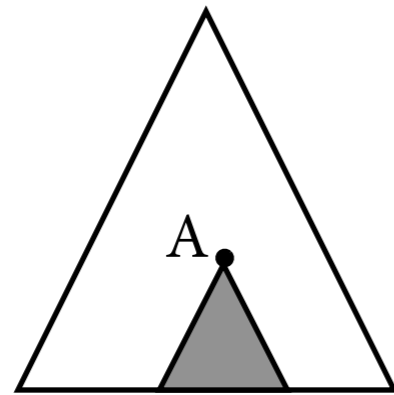


Adjunktion

- Man *kann* Beispiel-kfG lexikalisieren, wenn man eine zweite Operation zulässt: *Adjunktion*.

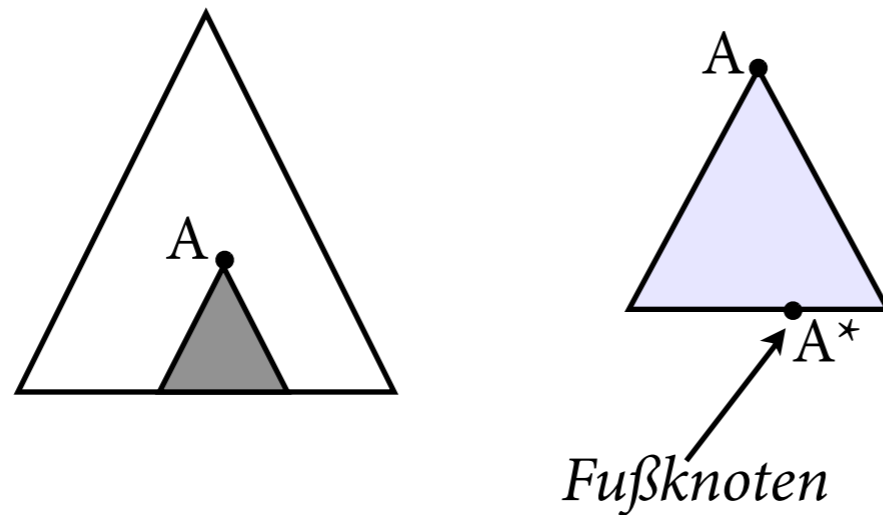
Adjunktion

- Man *kann* Beispiel-kgG lexikalisieren, wenn man eine zweite Operation zulässt: *Adjunktion*.



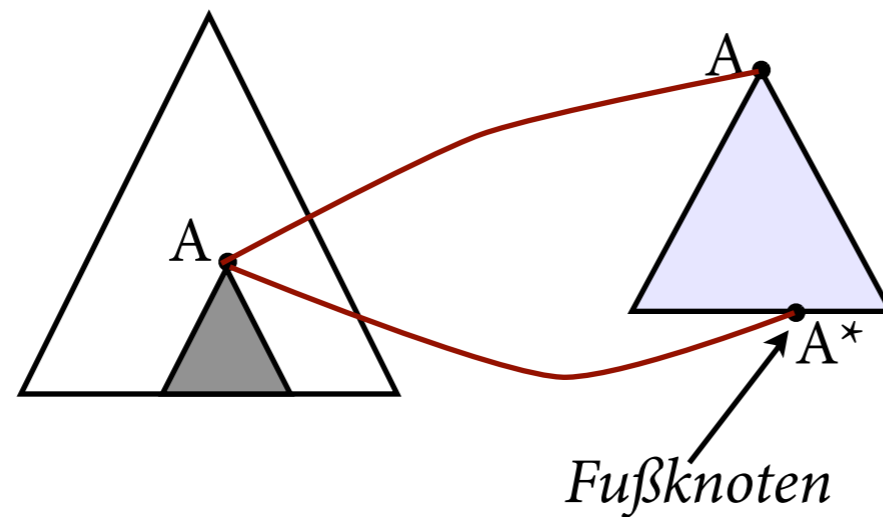
Adjunktion

- Man *kann* Beispiel-kgfG lexikalisieren, wenn man eine zweite Operation zulässt: *Adjunktion*.



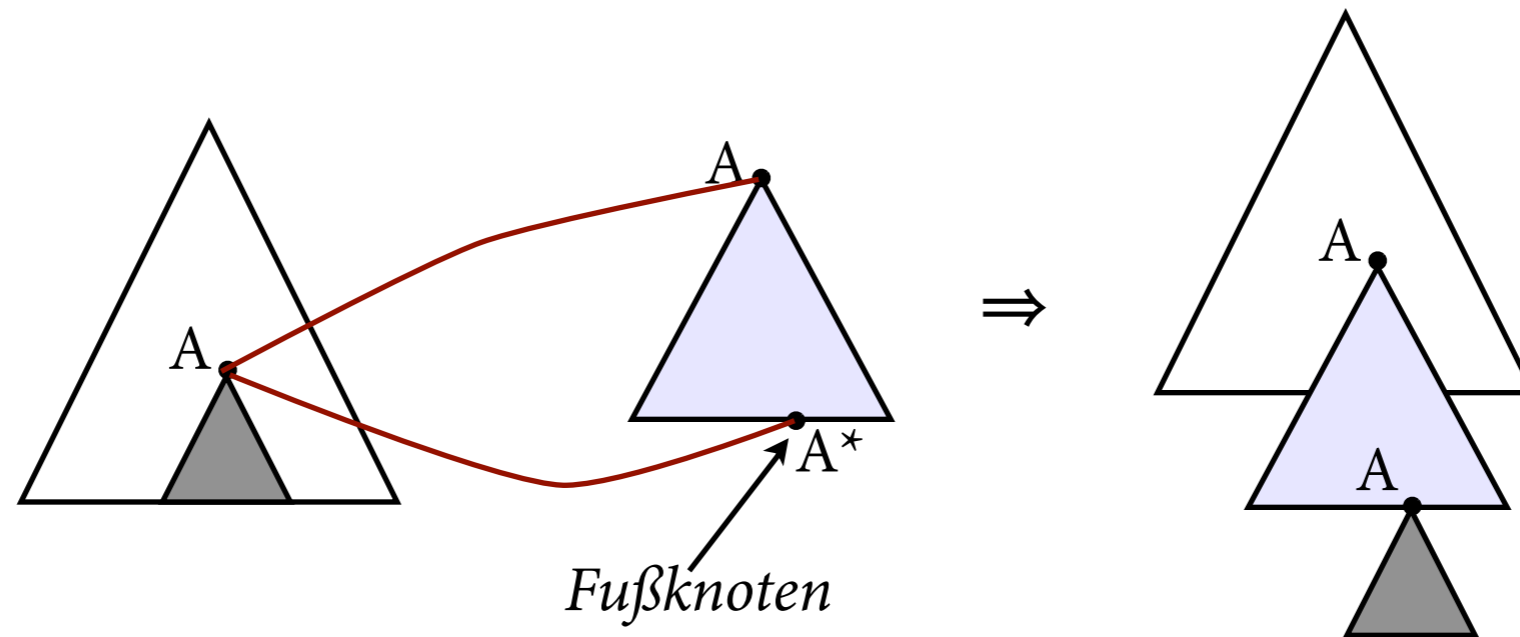
Adjunktion

- Man *kann* Beispiel-kgfG lexikalisieren, wenn man eine zweite Operation zulässt: *Adjunktion*.



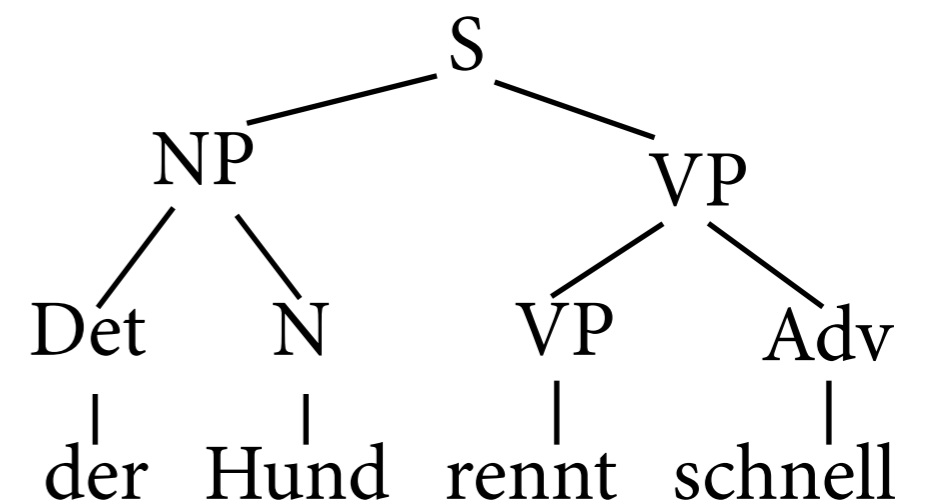
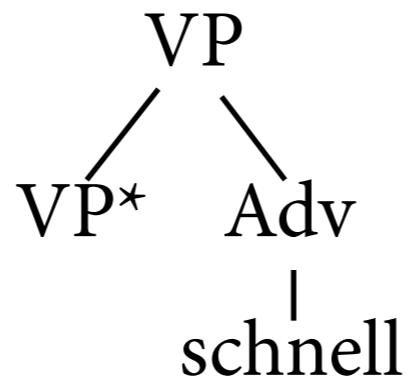
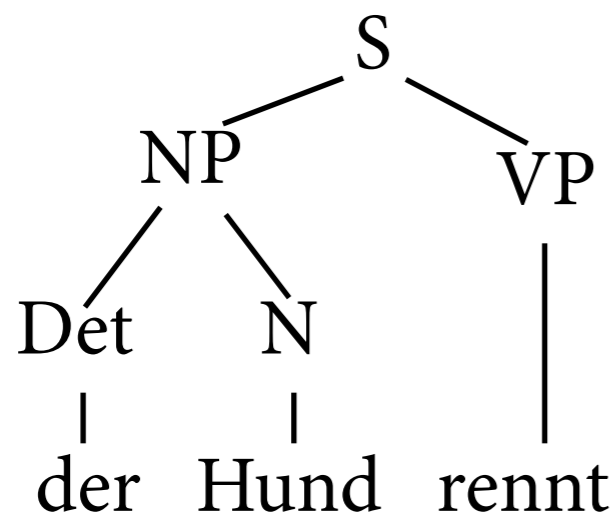
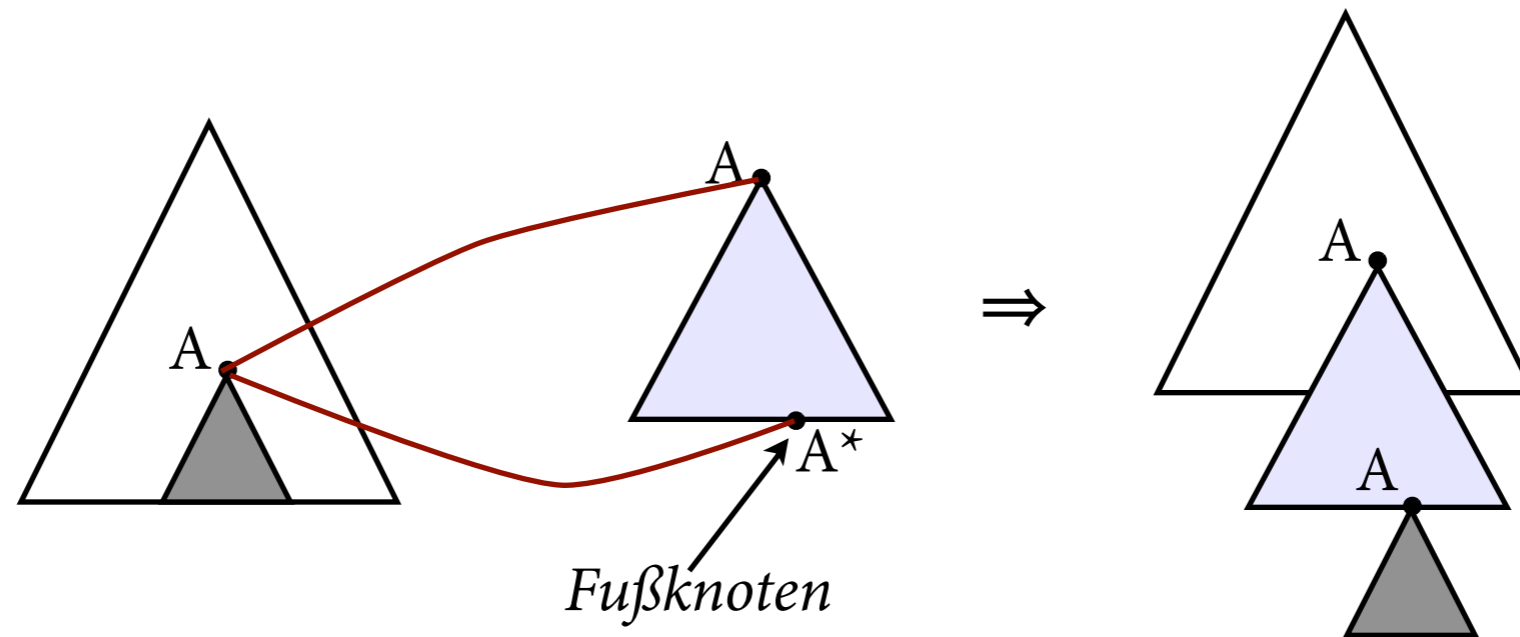
Adjunktion

- Man *kann* Beispiel-kgfG lexikalisieren, wenn man eine zweite Operation zulässt: *Adjunktion*.



Adjunktion

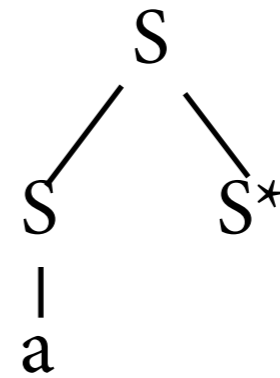
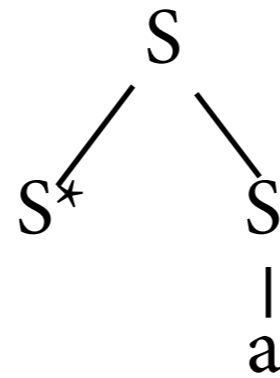
- Man *kann* Beispiel-kfG lexikalisieren, wenn man eine zweite Operation zulässt: *Adjunktion*.



Lexikalisierung mit Adjunktion

- Nehmen wir folgende Bäume:

S
|
a



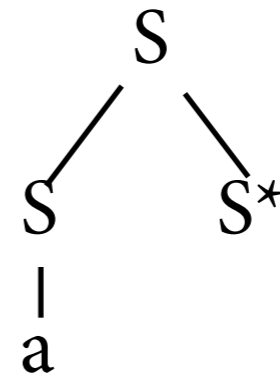
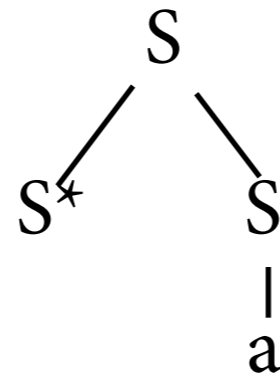
- Jetzt beliebige Parsebäume der kfG bauen:

S
|
a

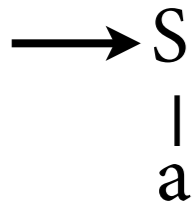
Lexikalisierung mit Adjunktion

- Nehmen wir folgende Bäume:

S
|
a



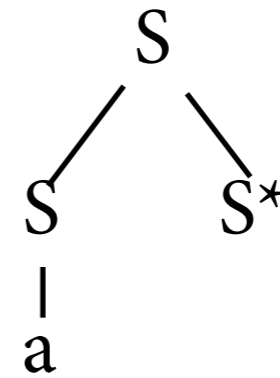
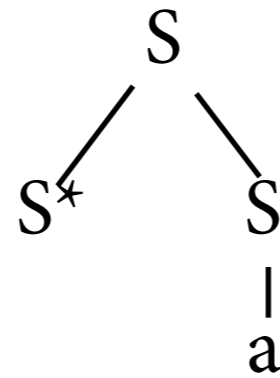
- Jetzt beliebige Parsebäume der kfG bauen:



Lexikalisierung mit Adjunktion

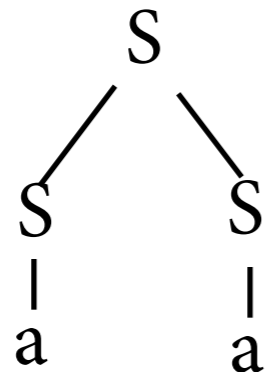
- Nehmen wir folgende Bäume:

S
|
a



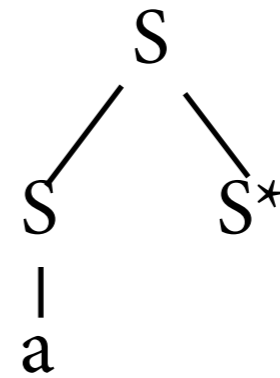
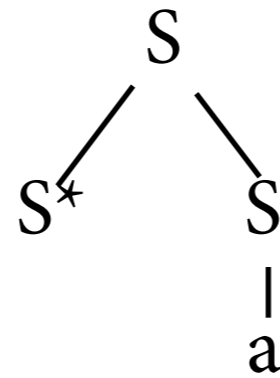
- Jetzt beliebige Parsebäume der kfG bauen:

→ S
|
a

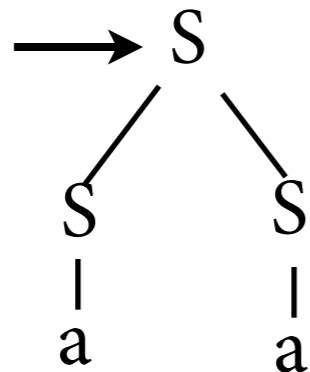
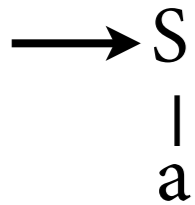


Lexikalisierung mit Adjunktion

- Nehmen wir folgende Bäume:

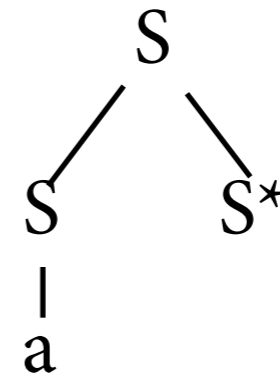
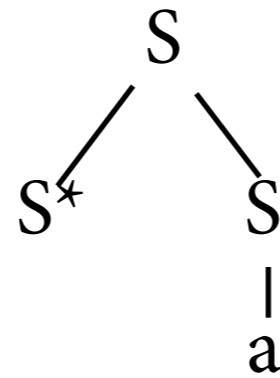


- Jetzt beliebige Parsebäume der kfG bauen:

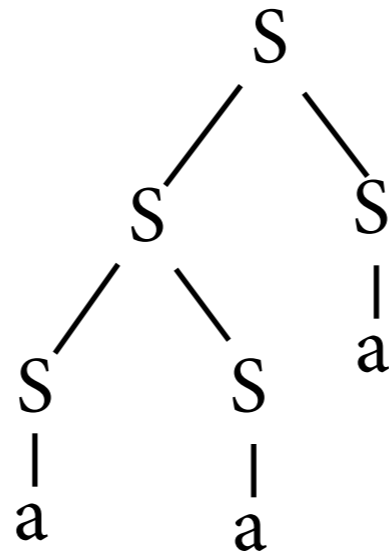
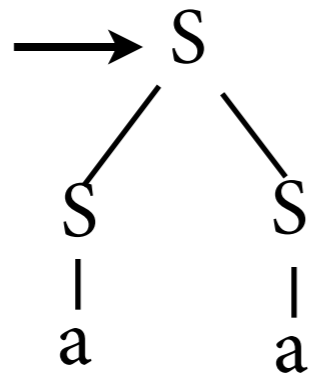
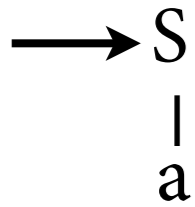


Lexikalisierung mit Adjunktion

- Nehmen wir folgende Bäume:

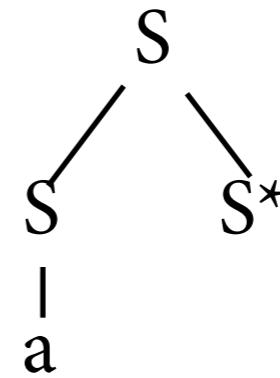
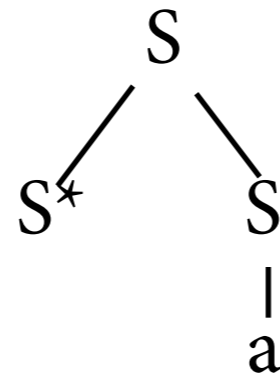


- Jetzt beliebige Parsebäume der kfG bauen:

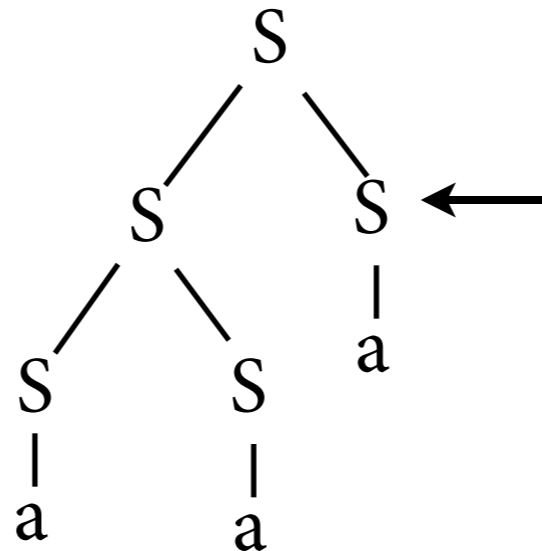
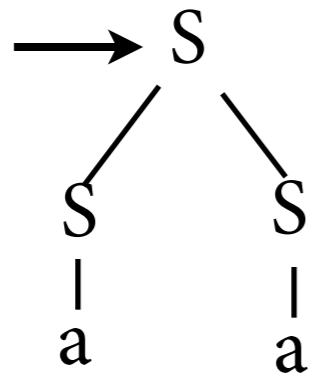
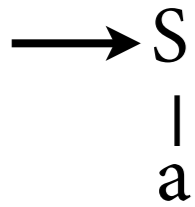


Lexikalisierung mit Adjunktion

- Nehmen wir folgende Bäume:

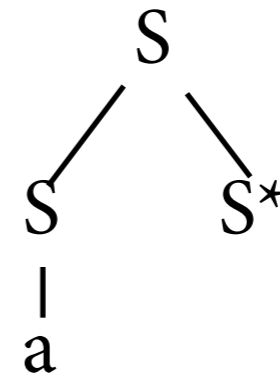
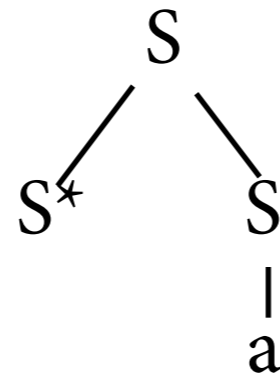


- Jetzt beliebige Parsebäume der kfG bauen:

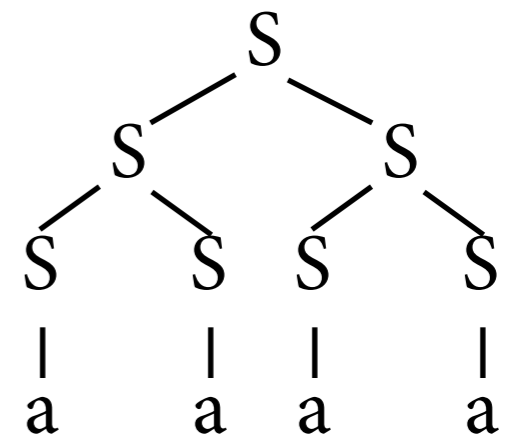
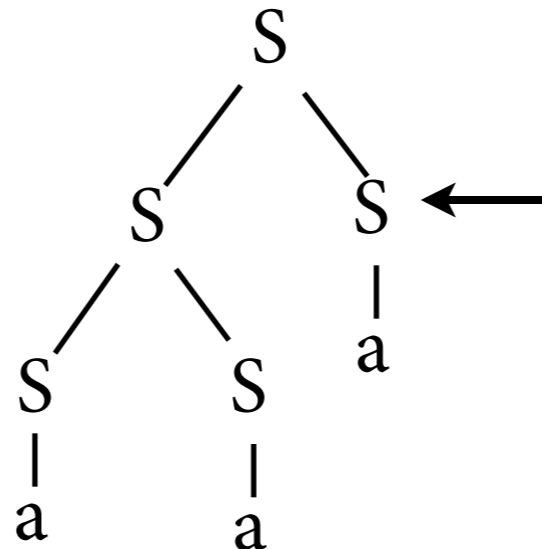
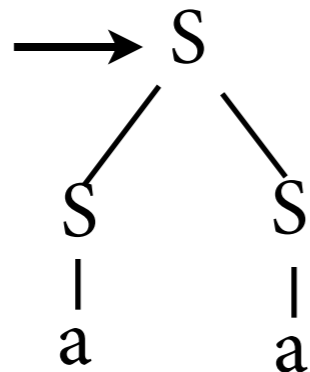
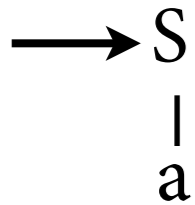


Lexikalisierung mit Adjunktion

- Nehmen wir folgende Bäume:



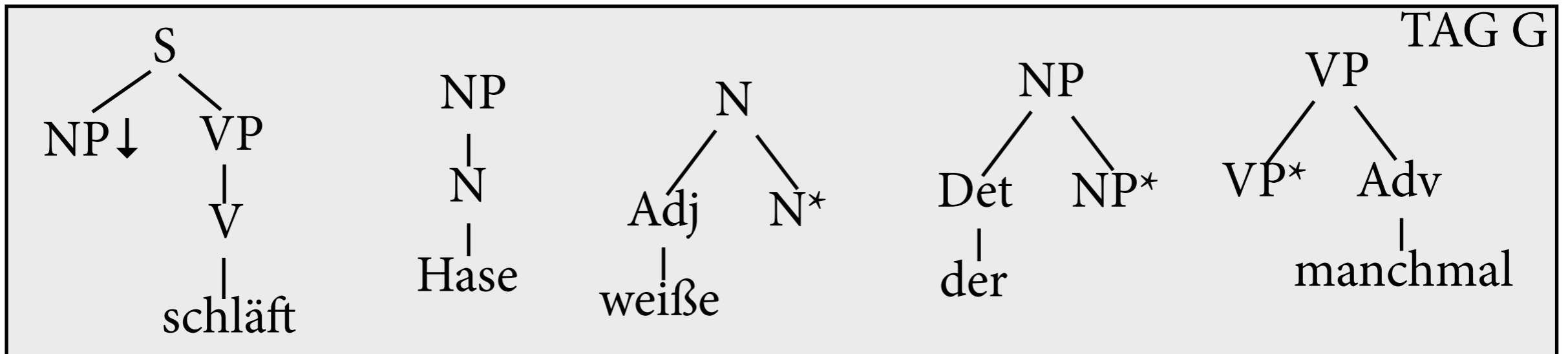
- Jetzt beliebige Parsebäume der kfG bauen:



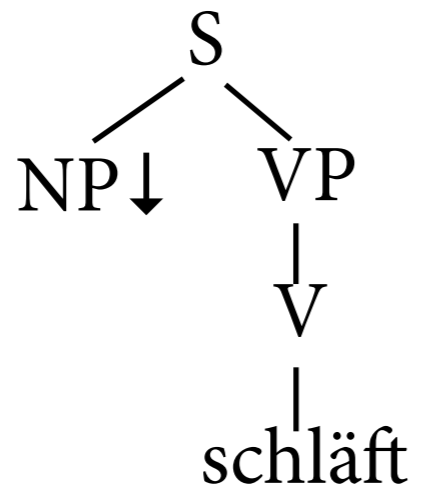
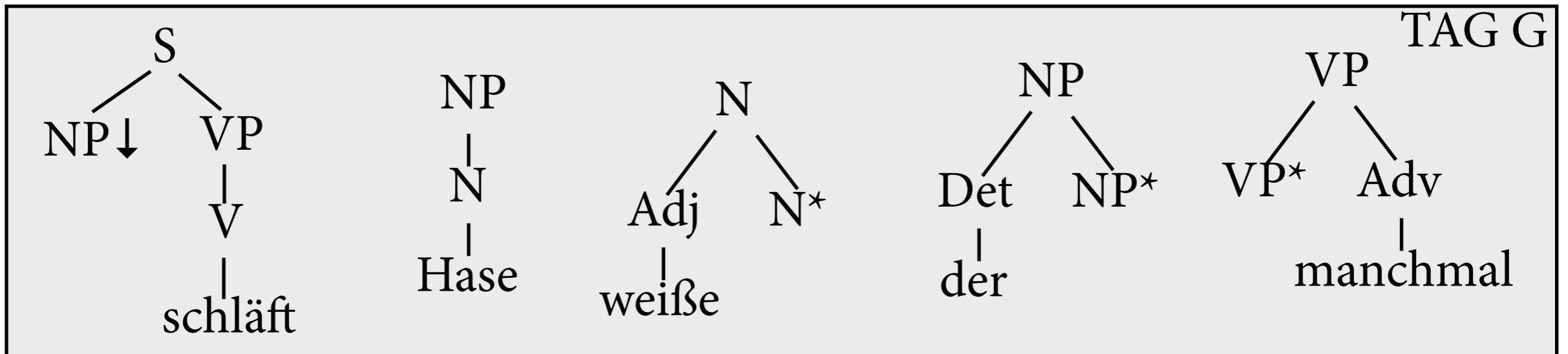
Baumadjunktionsgrammatiken

- *Baumadjunktionsgrammatik* (tree adjoining grammar, TAG): endliche Menge von Elementarbäumen, und zwar
 - ▶ *Initialbäumen*: Elementarbäume wie in TSG
 - ▶ *Auxiliarbäume*: Elementarbäume, in denen genau ein Blatt ein Fußknoten ist
- TAG-Ableitungen: in jedem Schritt
 - ▶ Substitution eines Initialbaums, oder
 - ▶ Adjunktion eines Auxiliarbaums

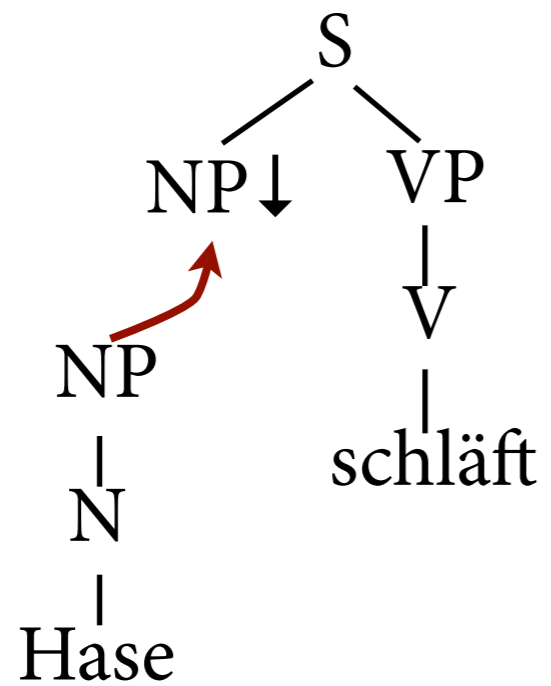
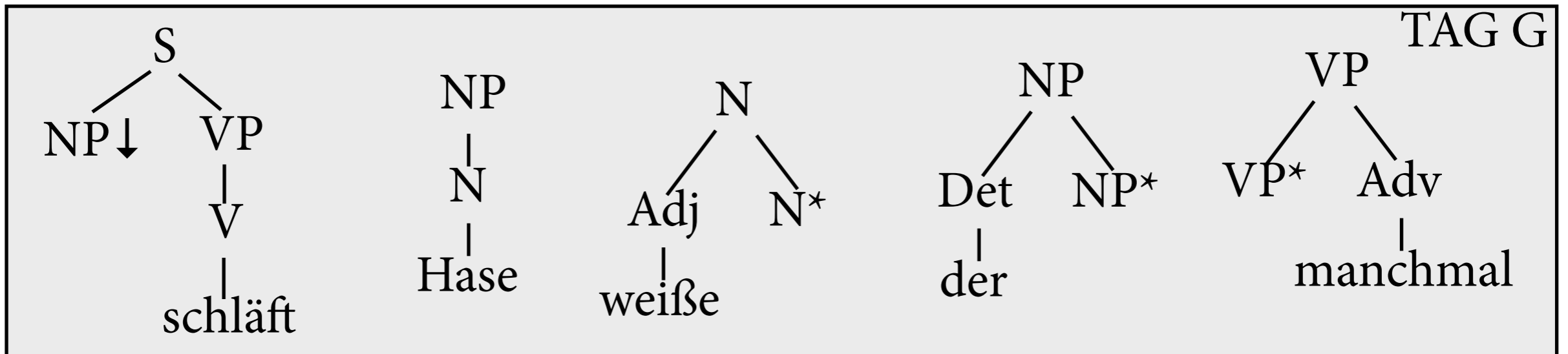
Ein Beispiel



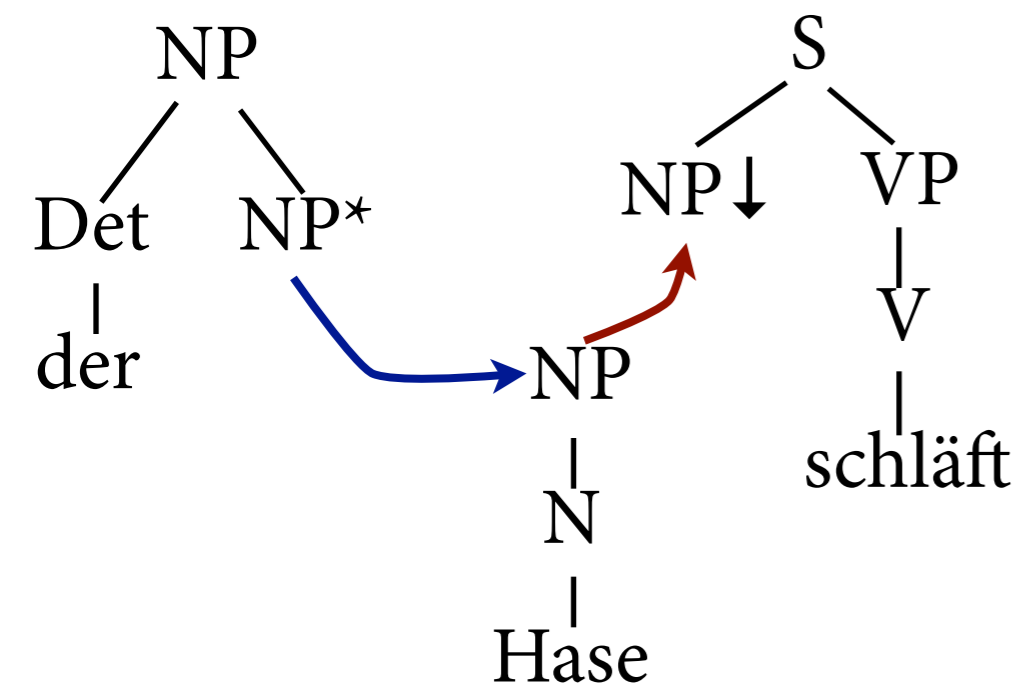
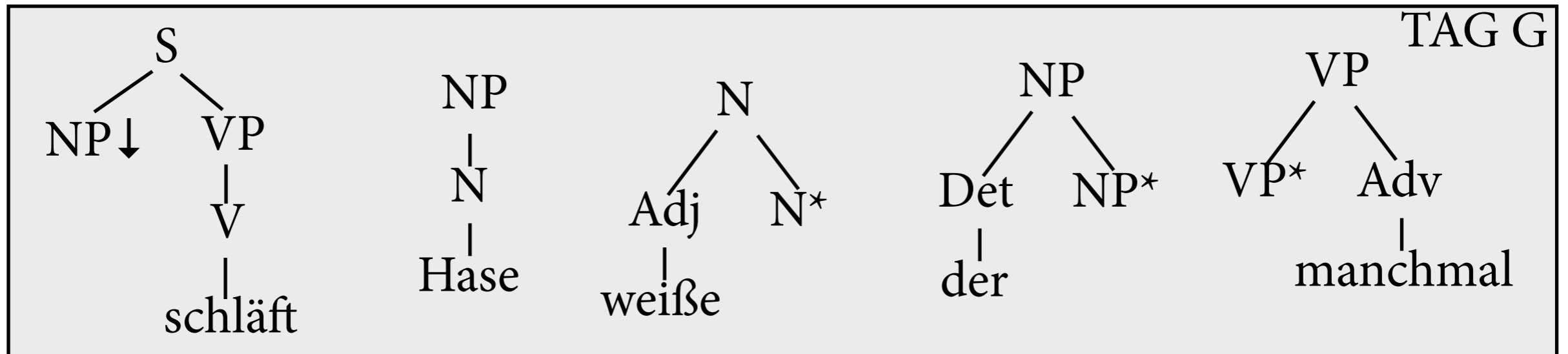
Ein Beispiel



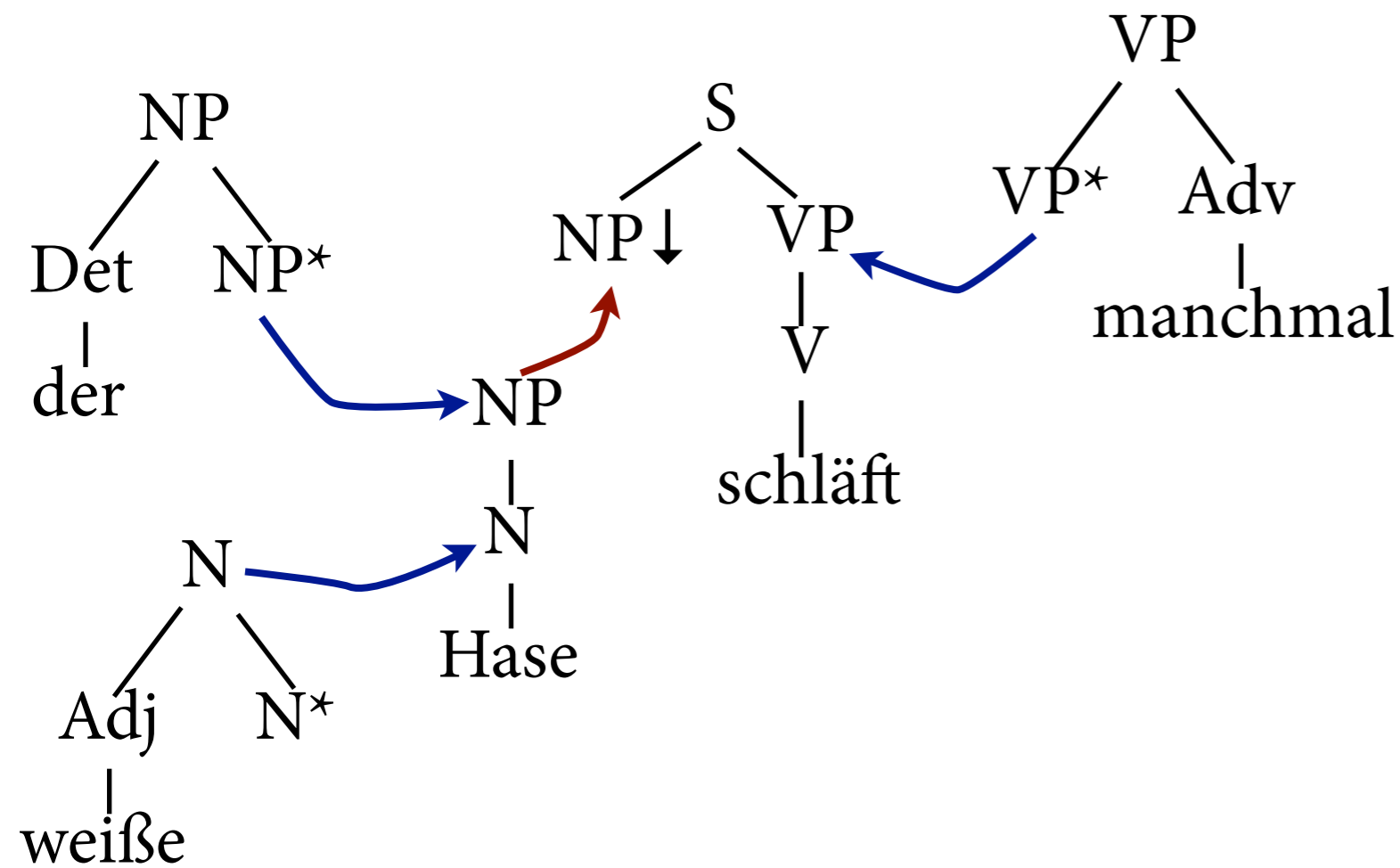
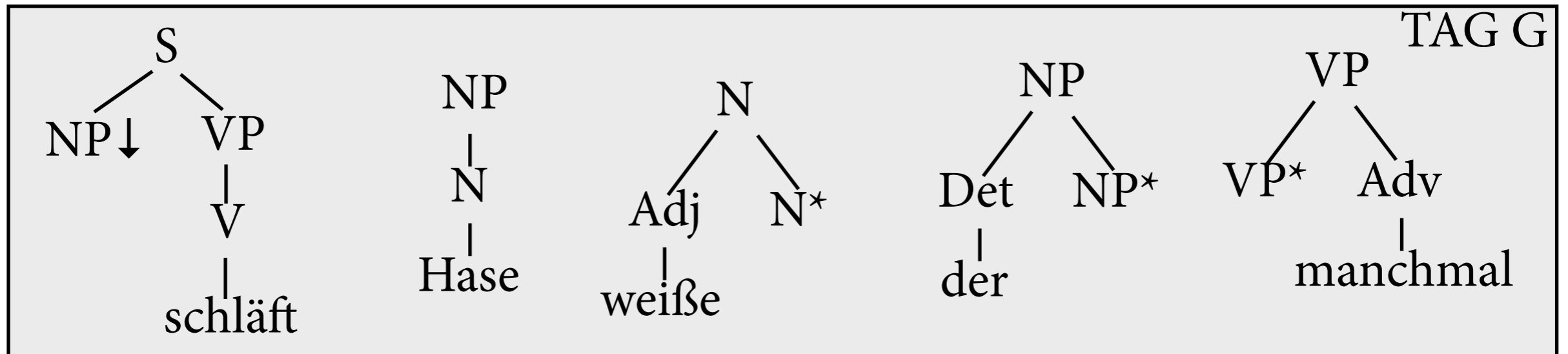
Ein Beispiel



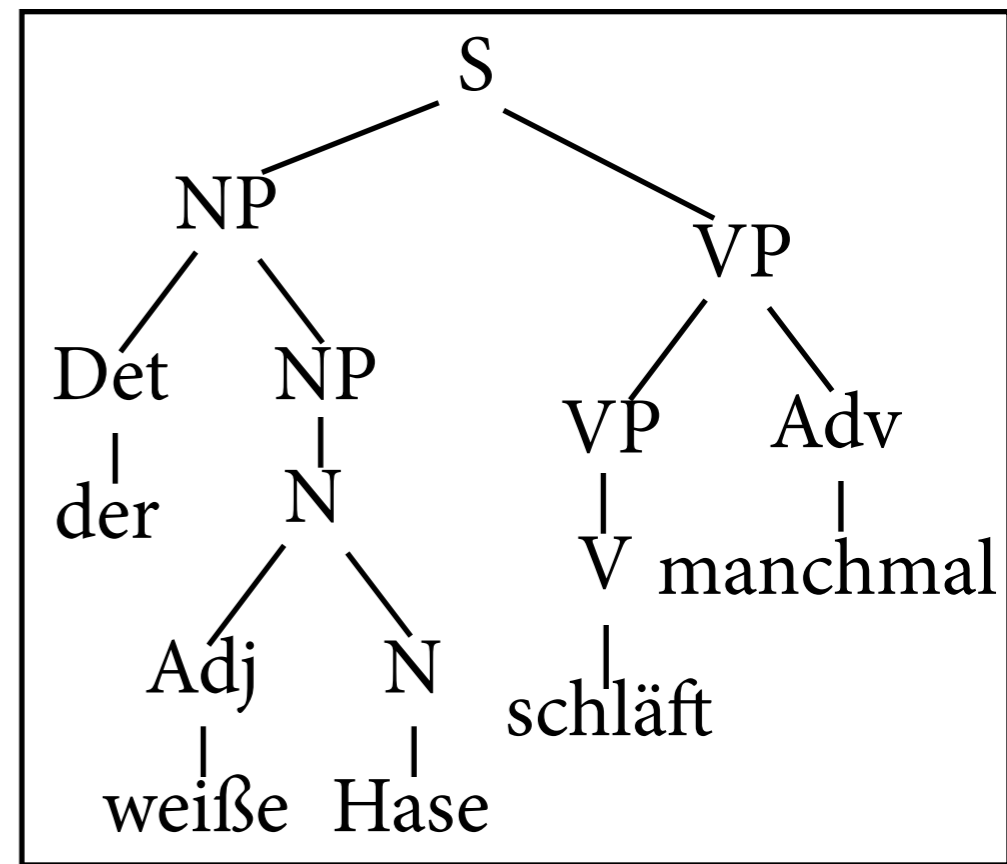
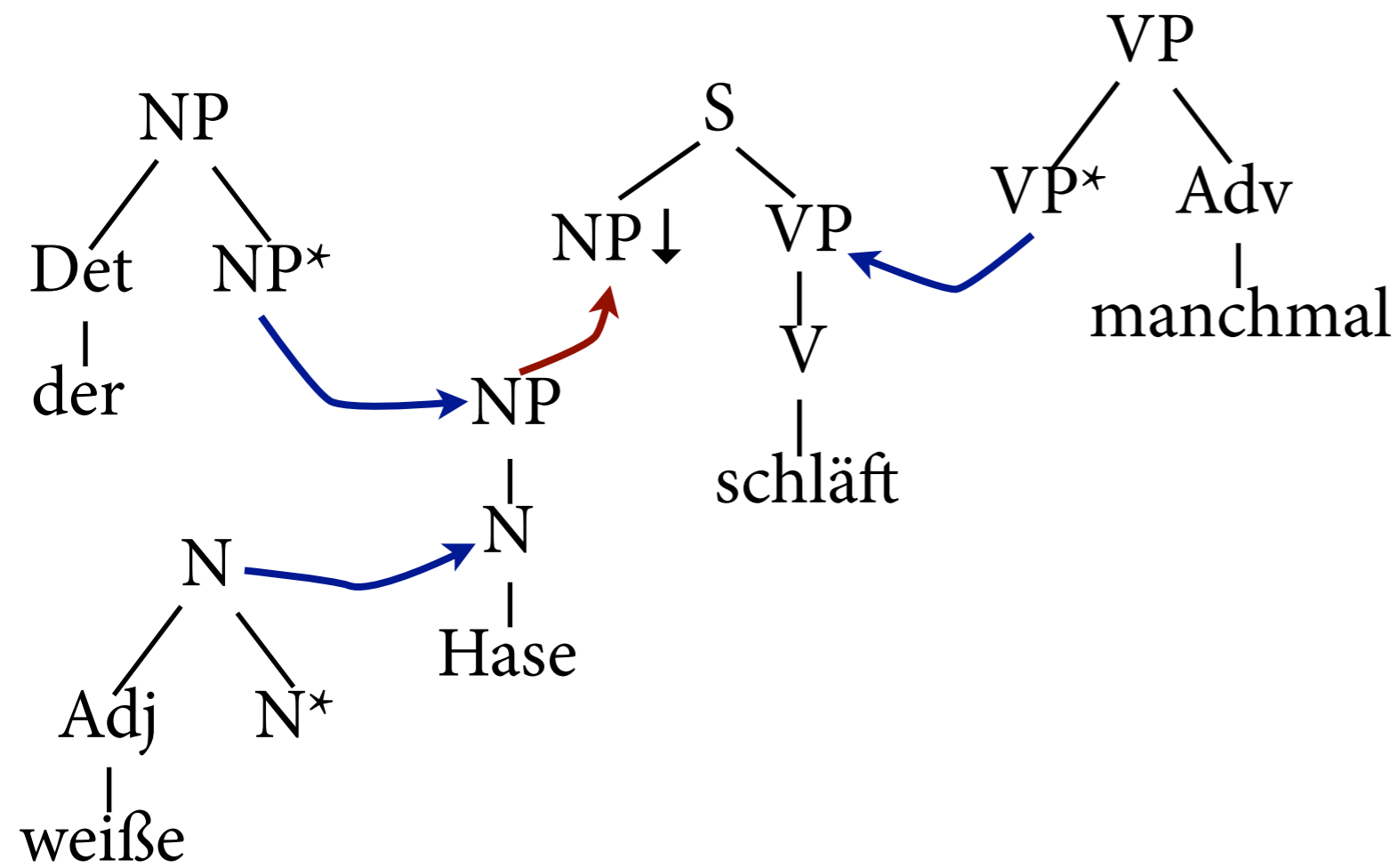
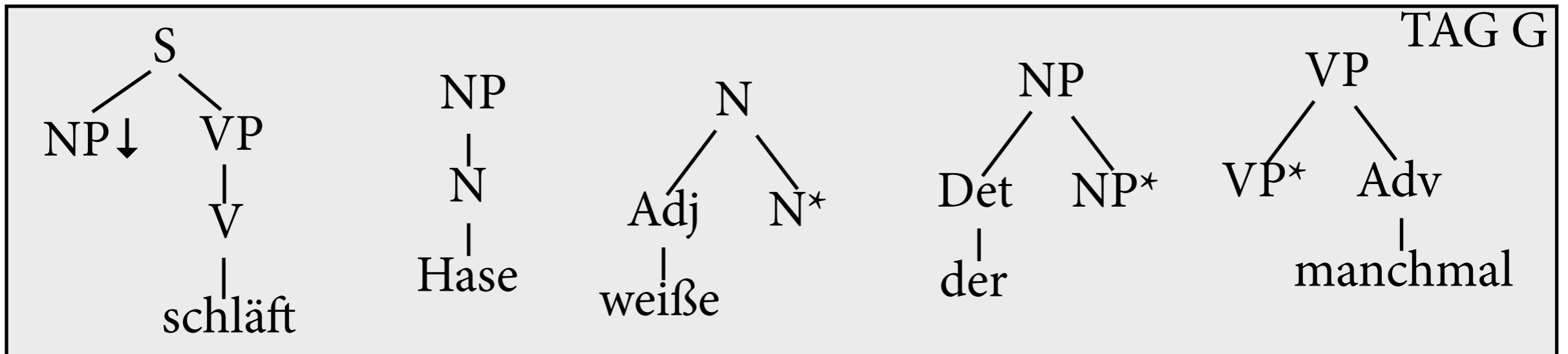
Ein Beispiel



Ein Beispiel

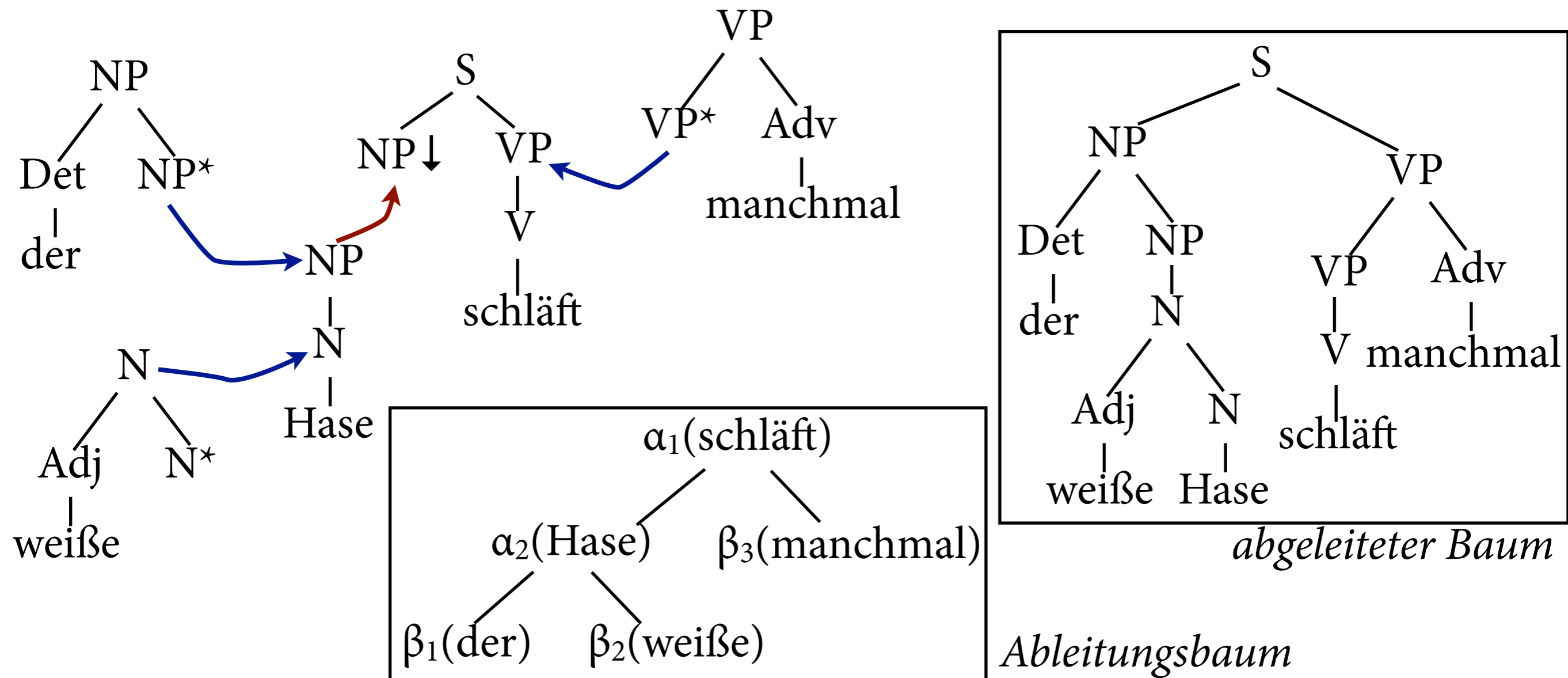
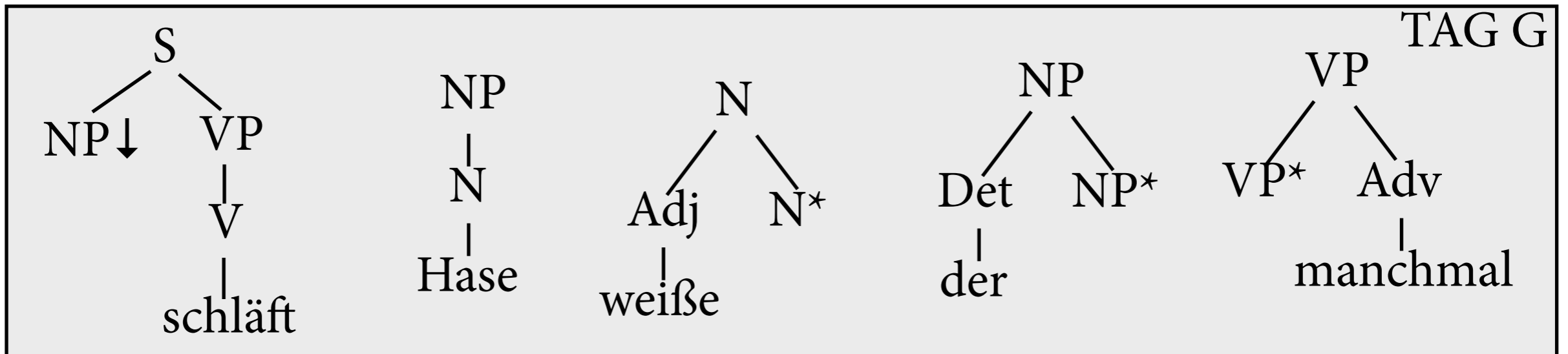


Ein Beispiel



abgeleiteter Baum

Ein Beispiel



Einige Punkte

- Ableitungsbaum: wie in TSG
 - ▶ aber jetzt Adjunktions- und Substitutionskanten unterscheiden
- Adjunktionsconstraints:
 - ▶ kann Knoten in Elementarbaum mit “no adjunction” (NA) markieren: hier darf man nicht adjungieren
 - ▶ Markierung “obligatory adjunction” (OA): hier *muss* man etwas adjungieren
 - ▶ An Fußknoten darf man *nie* adjungieren.

TAG: Linguistische Prinzipien

- Substitution verbindet ein Prädikat mit den *Komplementen*, für die es subkategorisiert.
- Adjunktion verbindet einen Ausdruck mit seinen *Adjunkten (Modifikation)*.
 - ▶ Substantive mit Adjektiven, Artikeln, Relativsätzen usw.
 - ▶ Verben und Sätze mit Adverbien
- Ein Großteil von *Rekursion* kann durch Adjunktion erfasst werden.

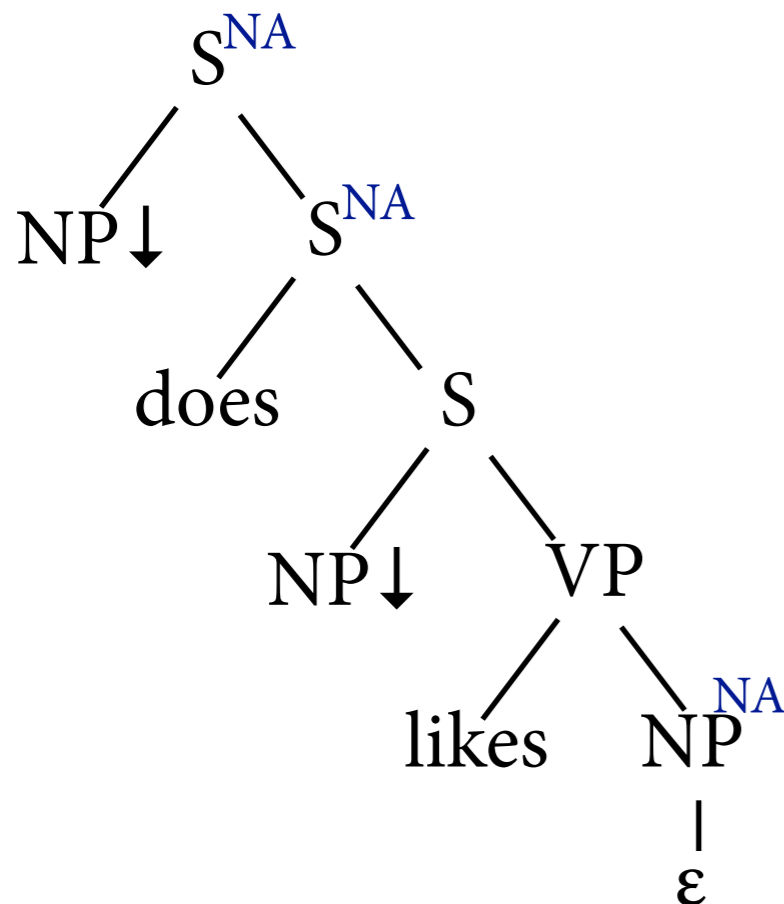
Fernabhängigkeiten

- Fernabhängigkeiten (z.B. wh-Bewegung):
 - ▶ “Bewegung” innerhalb eines Elementarbaums
 - ▶ “bewegtes” wh-Wort wird durch Adjunktion immer weiter vom Verb weggedrückt
 - ▶ NB: Bewegung ist nur Motivation für konkreten Elementarbaum; in TAG-Ableitungen wird nichts bewegt!

“Who does Peter say Mary thinks John likes”

Fernabhängigkeiten

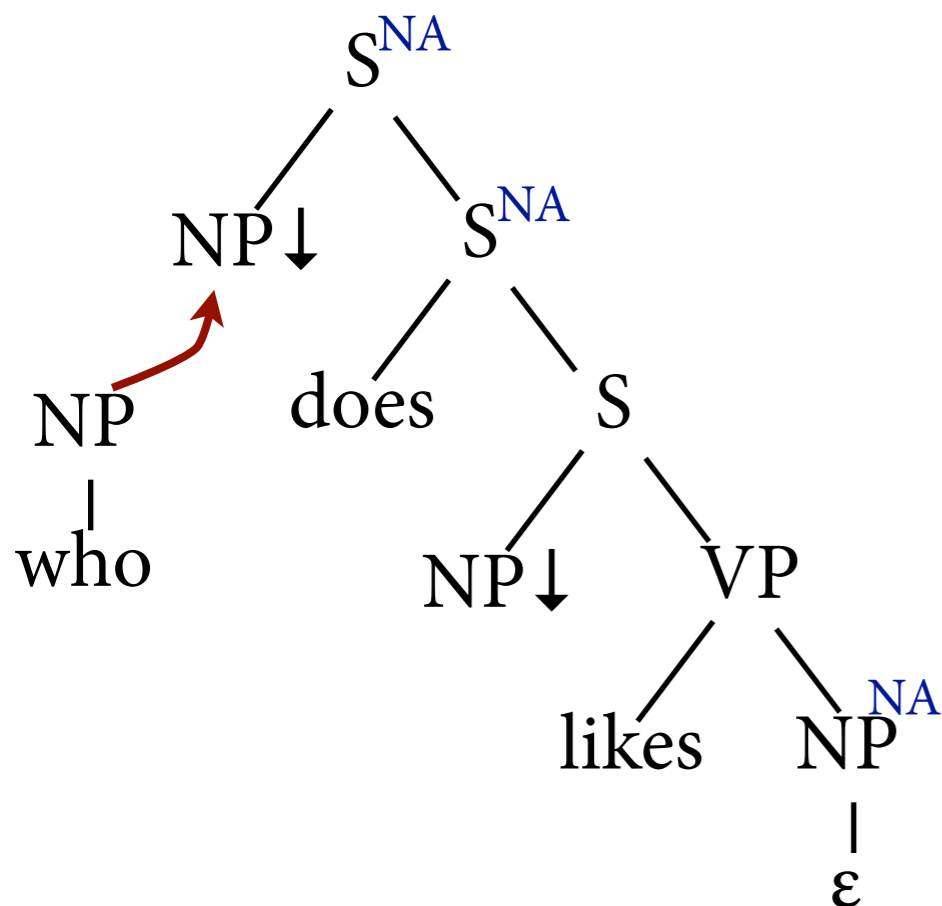
- Fernabhängigkeiten (z.B. wh-Bewegung):
 - ▶ “Bewegung” innerhalb eines Elementarbaums
 - ▶ “bewegtes” wh-Wort wird durch Adjunktion immer weiter vom Verb weggedrückt
 - ▶ NB: Bewegung ist nur Motivation für konkreten Elementarbaum; in TAG-Ableitungen wird nichts bewegt!



“Who does Peter say Mary thinks John likes”

Fernabhängigkeiten

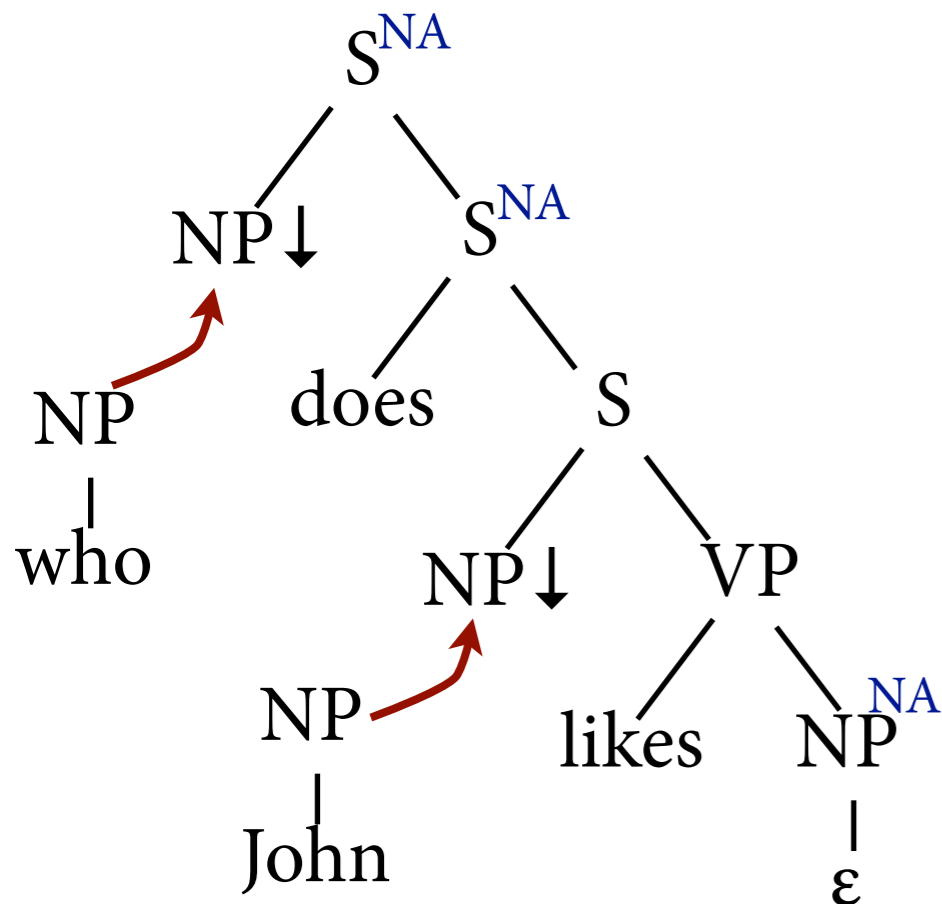
- Fernabhängigkeiten (z.B. wh-Bewegung):
 - ▶ “Bewegung” innerhalb eines Elementarbaums
 - ▶ “bewegtes” wh-Wort wird durch Adjunktion immer weiter vom Verb weggedrückt
 - ▶ NB: Bewegung ist nur Motivation für konkreten Elementarbaum; in TAG-Ableitungen wird nichts bewegt!



“Who does Peter say Mary thinks John likes”

Fernabhängigkeiten

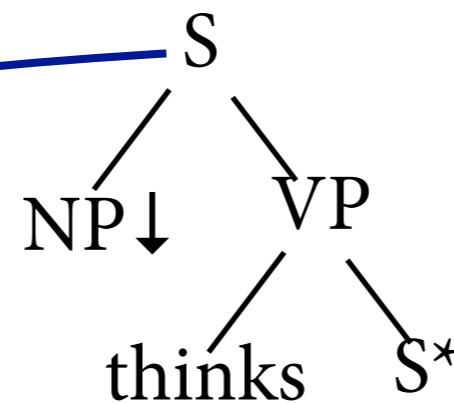
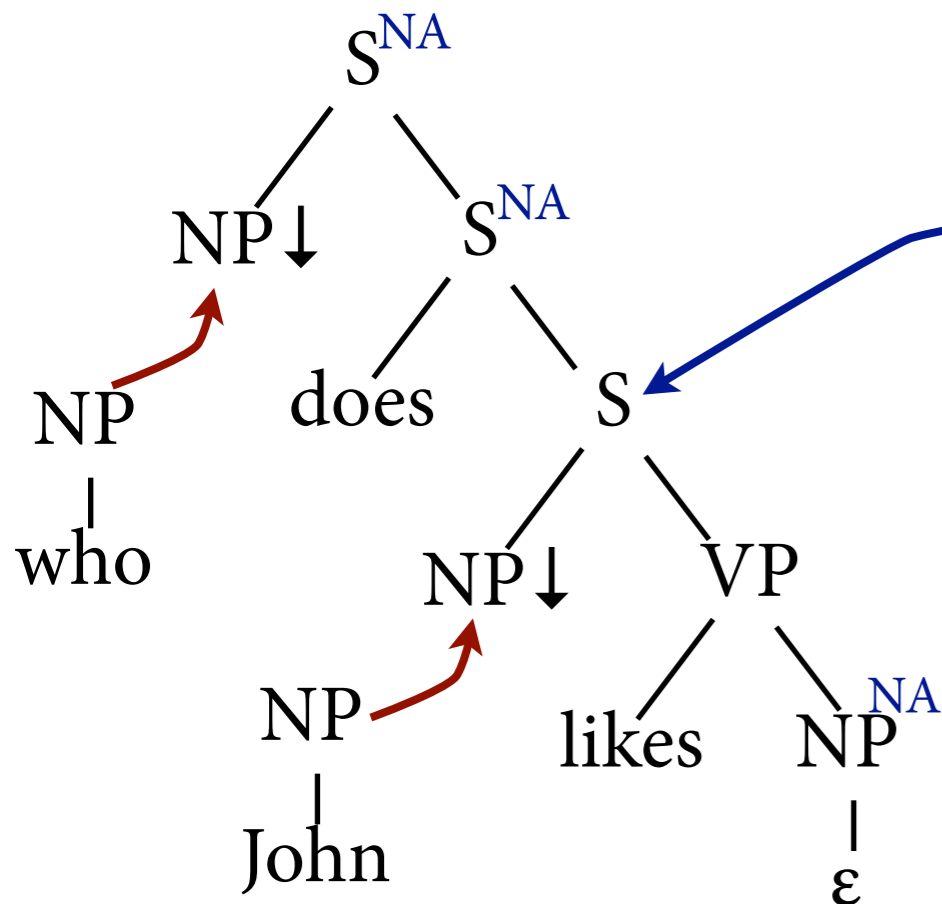
- Fernabhängigkeiten (z.B. wh-Bewegung):
 - ▶ “Bewegung” innerhalb eines Elementarbaums
 - ▶ “bewegtes” wh-Wort wird durch Adjunktion immer weiter vom Verb weggedrückt
 - ▶ NB: Bewegung ist nur Motivation für konkreten Elementarbaum; in TAG-Ableitungen wird nichts bewegt!



“Who does Peter say Mary thinks John likes”

Fernabhängigkeiten

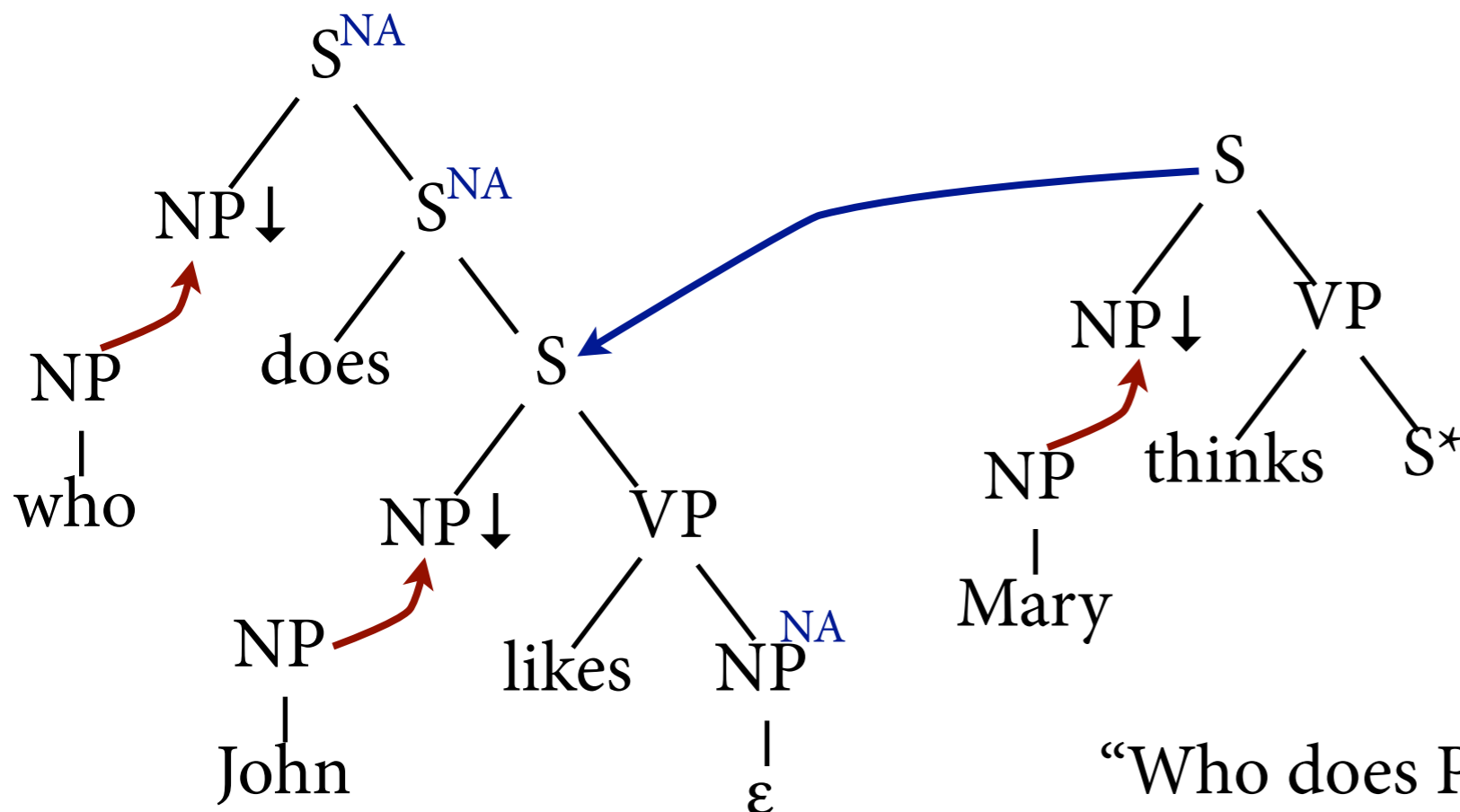
- Fernabhängigkeiten (z.B. wh-Bewegung):
 - ▶ “Bewegung” innerhalb eines Elementarbaums
 - ▶ “bewegtes” wh-Wort wird durch Adjunktion immer weiter vom Verb weggedrückt
 - ▶ NB: Bewegung ist nur Motivation für konkreten Elementarbaum; in TAG-Ableitungen wird nichts bewegt!



“Who does Peter say Mary thinks John likes”

Fernabhängigkeiten

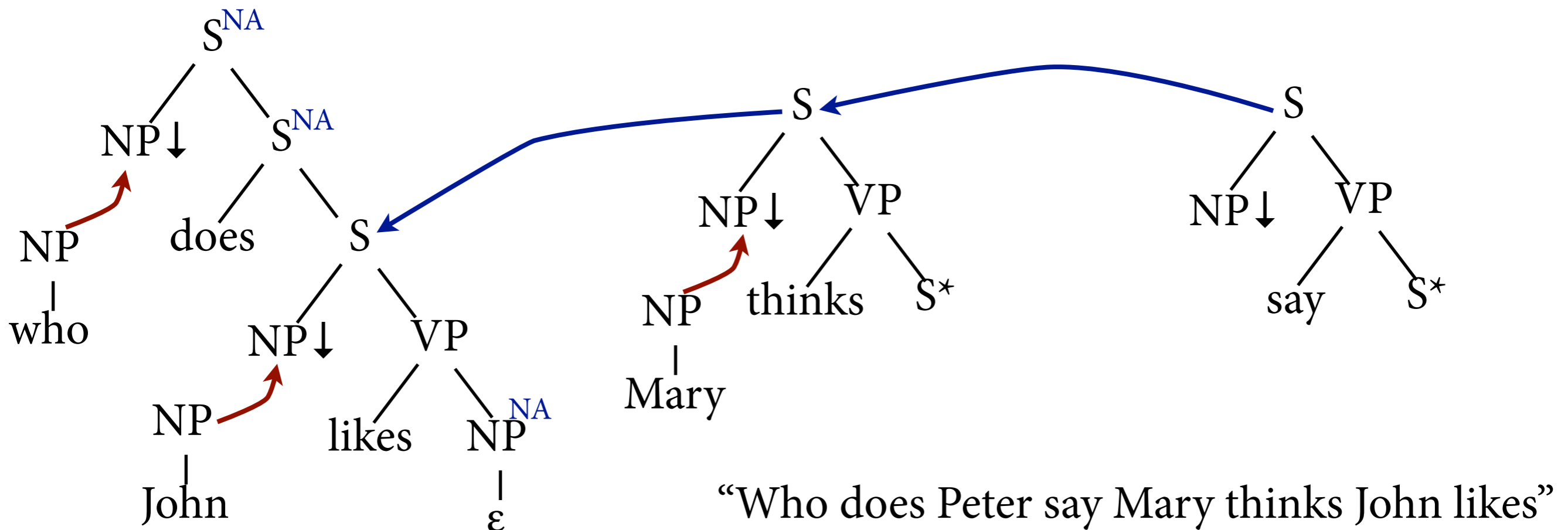
- Fernabhängigkeiten (z.B. wh-Bewegung):
 - ▶ “Bewegung” innerhalb eines Elementarbaums
 - ▶ “bewegtes” wh-Wort wird durch Adjunktion immer weiter vom Verb weggedrückt
 - ▶ NB: Bewegung ist nur Motivation für konkreten Elementarbaum; in TAG-Ableitungen wird nichts bewegt!



“Who does Peter say Mary thinks John likes”

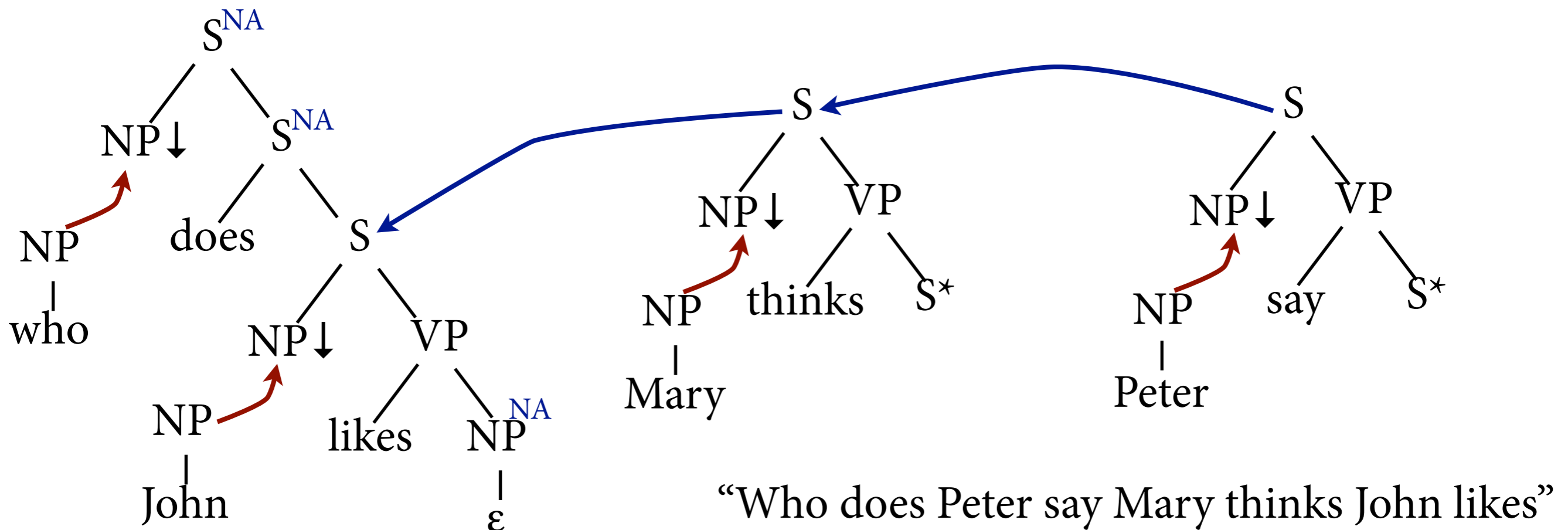
Fernabhängigkeiten

- Fernabhängigkeiten (z.B. wh-Bewegung):
 - ▶ “Bewegung” innerhalb eines Elementarbaums
 - ▶ “bewegtes” wh-Wort wird durch Adjunktion immer weiter vom Verb weggedrückt
 - ▶ NB: Bewegung ist nur Motivation für konkreten Elementarbaum; in TAG-Ableitungen wird nichts bewegt!



Fernabhängigkeiten

- Fernabhängigkeiten (z.B. wh-Bewegung):
 - ▶ “Bewegung” innerhalb eines Elementarbaums
 - ▶ “bewegtes” wh-Wort wird durch Adjunktion immer weiter vom Verb weggedrückt
 - ▶ NB: Bewegung ist nur Motivation für konkreten Elementarbaum; in TAG-Ableitungen wird nichts bewegt!



Extended domain of locality

- Lokalität: Informationen, die in einem einzigen Grammatikeintrag vorliegen.
 - ▶ vom Grammatikentwickler als Einheit gedacht
 - ▶ vom Parser als Einheit verarbeitet
- KfG: einzelne Produktionsregel.
- TAG: Elementarbaum, d.h. viel größere Einheit.

Extended domain of locality

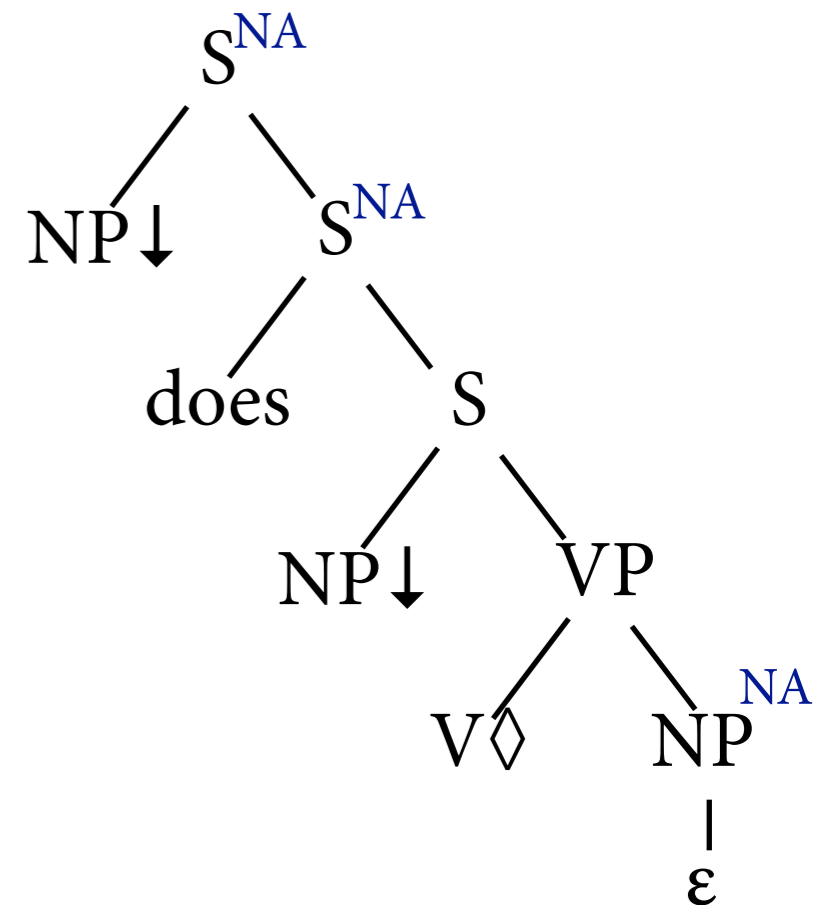
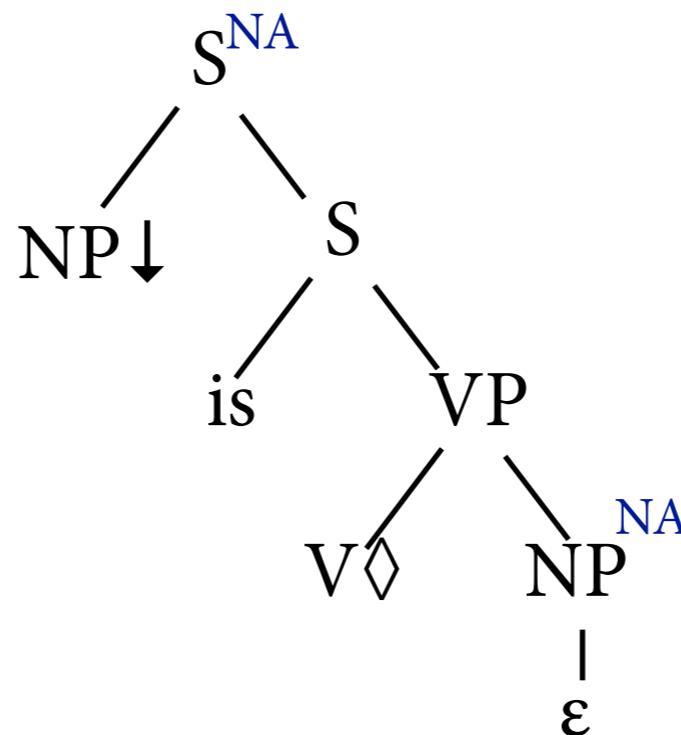
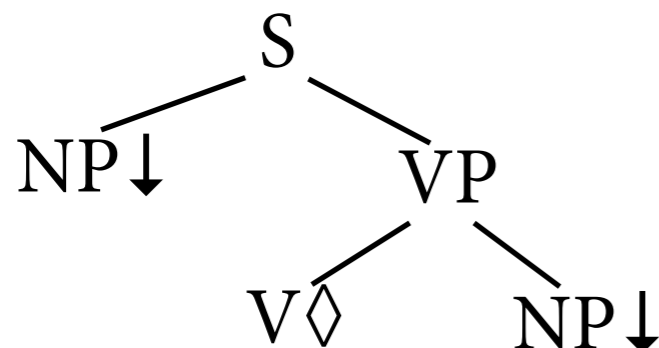
- Einheit Elementarbaum:
 - ▶ enthält Substitutionsknoten für alle Komplemente des lexikalischen Ankers
 - ▶ d.h. alles, was für semantische Interpretation des Wortes nötig ist
 - ▶ “Bewegungen” innerhalb des Elementarbaums können vorausberechnet und lexikalisiert werden; Parser muss deshalb nicht mehr drüber nachdenken.
- Das erleichtert Grammatikentwicklung und Parsing.

Grammatik-Organisation

- Elementarbäume können zu *Baumfamilien* zusammengefasst werden.
 - ▶ Zweck: verschiedene syntaktische Konfigurationen fürs gleiche Wort kommen immer wieder vor.
 - ▶ Ein Wort kann allem Bäumen der gleichen Familie zugleich zugeordnet werden.

Transitive Verben

(z.B. V = like/likes/liked):

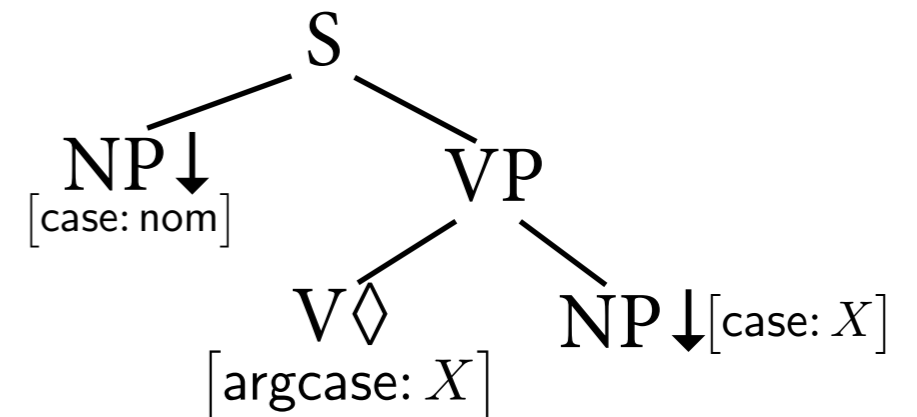
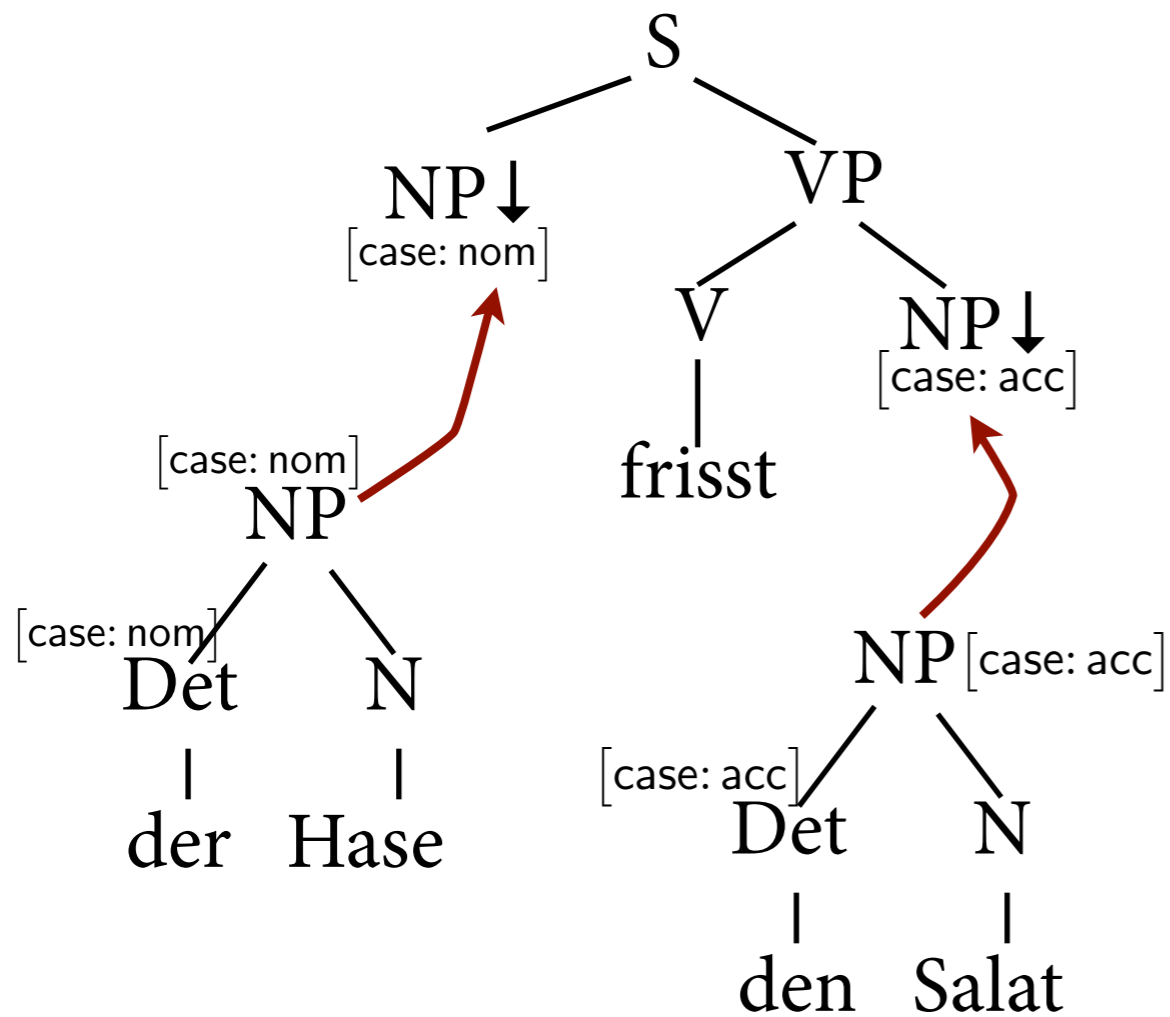


Features

- Merkmale (Features): zusätzliche Werte an Nichtterminale anheften und per Unifikation abgleichen.
 - ▶ ermöglicht kompakte Darstellung von Kongruenz usw.
- Letzte Woche: kfGen mit Features.
 - ▶ geht es auch mit TAG?

Erster Ansatz

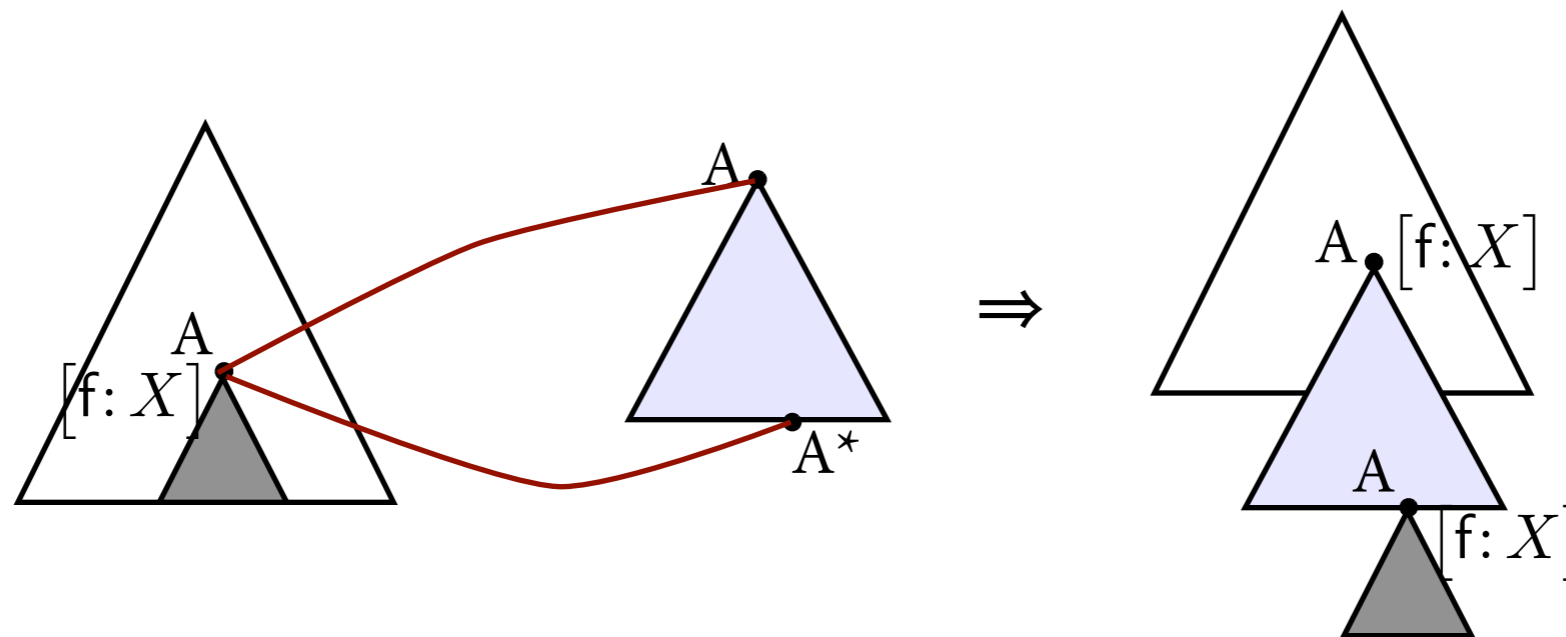
- Erste Idee: Knoten von Elementarbäumen mit Feature-Strukturen dekorieren.



Im Lexikon spezifizieren:
frisst: V, argcase=acc
hilft: V, argcase=dat
usw.

Ein Problem

- Was passiert mit der Featurestruktur an einem Knoten, wenn man dort adjungiert?
 - ▶ X mit FS bei Wurzel unifizieren? Oder mit FS bei Fußknoten?
 - ▶ oder mit beiden? (Konsequenz: $FS(\text{Wurzel}) = FS(\text{Fußknoten})$)



Bessere Intuition:

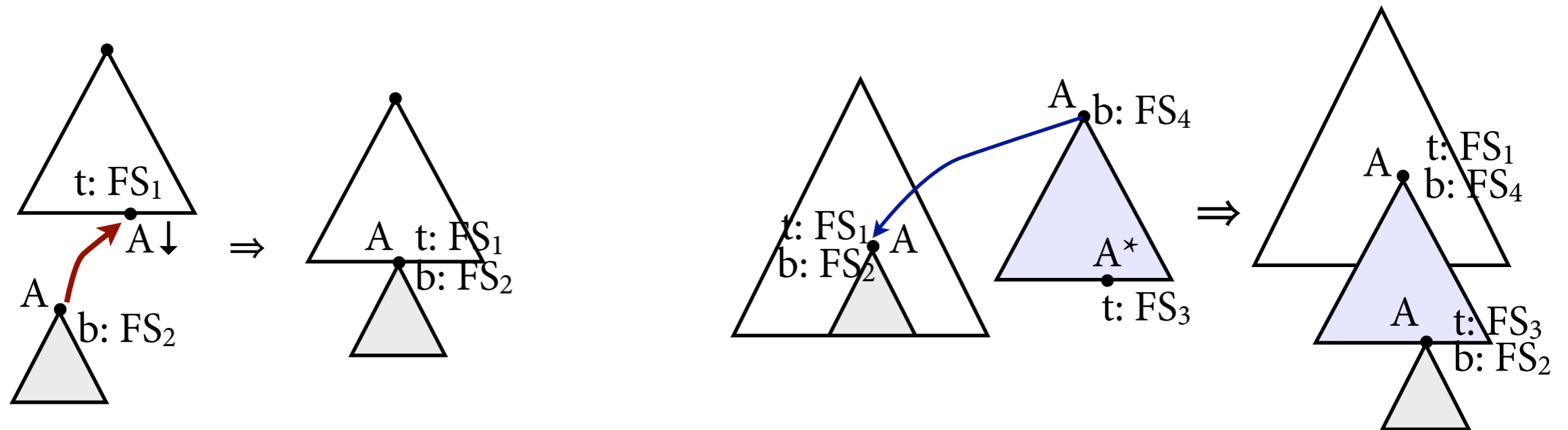
Adjunktion bricht einen Knoten in zwei Hälften auf.

Zweiter Ansatz (FTAG)

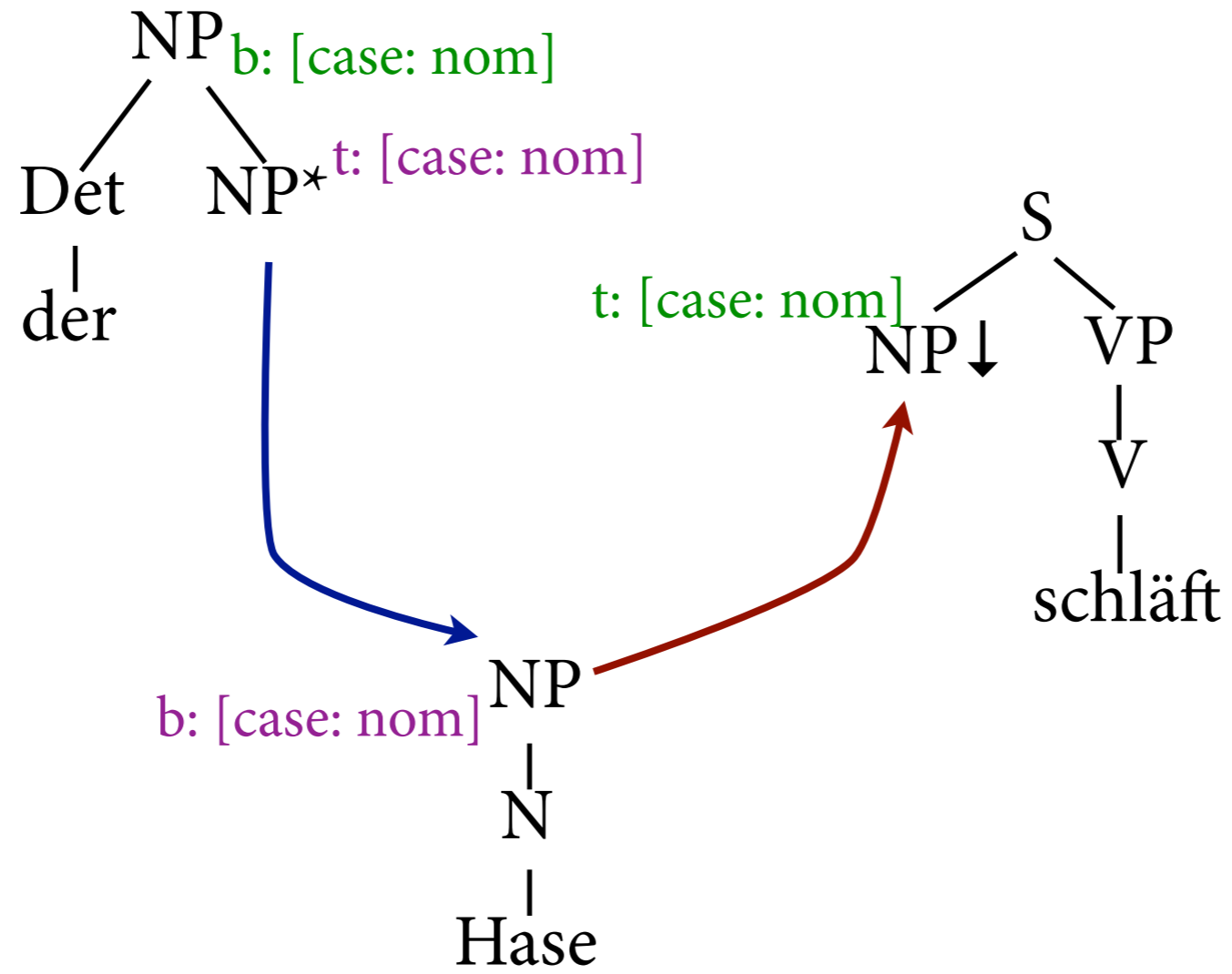
- In Feature-TAG trägt jeder Knoten *zwei* Featurestrukturen: für obere und untere Hälfte des Knotens.
 - ▶ Wurzeln haben nur eine untere FS
 - ▶ Blätter haben nur eine obere FS
 - ▶ innere Knoten haben obere & untere FS
- Am Ende der Ableitung werden obere und untere FS jedes Knotens unifiziert.
 - ▶ wenn das nicht geht, schlägt Ableitung fehl
 - ▶ zwischendurch dürfen FSen nicht-unifizierbar sein

Operationen von FTAG

- Substitution von t in u :
 - ▶ kombiniere obere FS von u mit FS der Wurzel von t
- Adjunktion von t in u :
 - ▶ kombiniere obere FS von u mit FS der Wurzel von t
 - ▶ kombiniere untere FS von u mit FS des Fußknotens von t

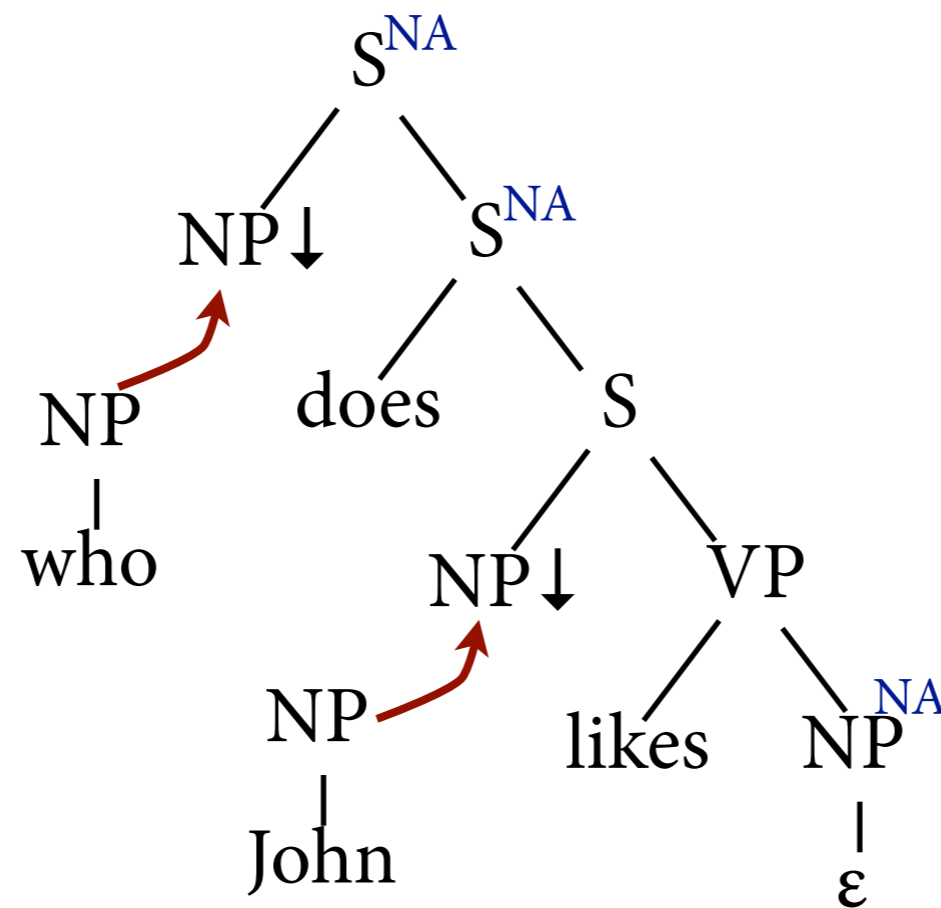


Beispiel



Kontrolle von Fernabhängigkeiten

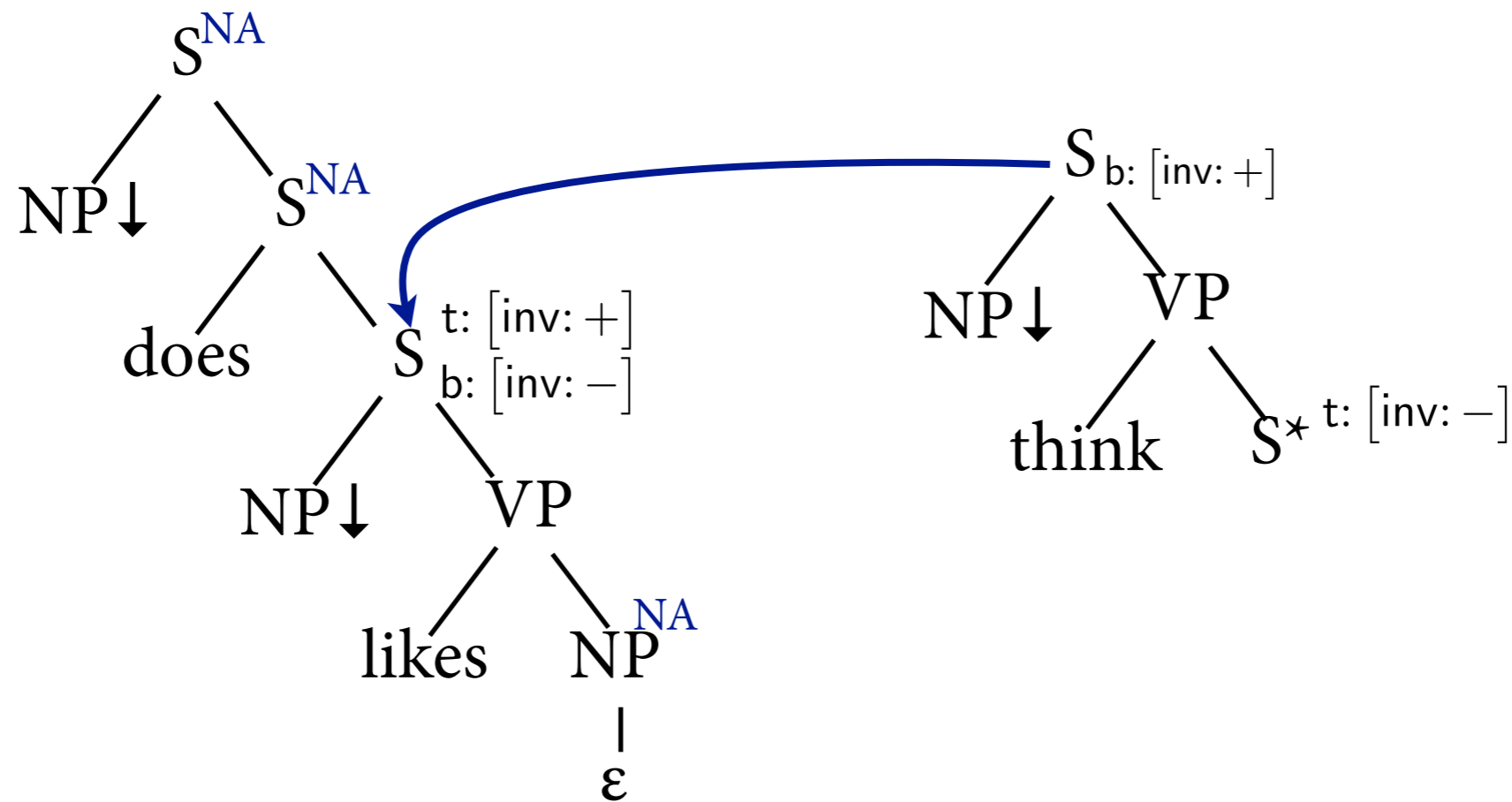
- Problem in unserer Analyse vorhin: “who does John likes” wird als grammatisch akzeptiert.



- Wie erzwingen wir, dass ein einbettendes Verb adjungiert werden *muss*?

Kontrolle von Fernabhängigkeiten

- Lösung: inkompatible Features werden durch Adjunktion aufgelöst.



Zusammenfassung

- TAG: Elementarbäume durch Substitution und Adjunktion kombinieren.
 - ▶ Motivation 1: starke Lexikalisierung von kfGen
 - ▶ Motivation 2: Elementarbaum = linguistisch angenehme Lokalisierungsdomäne
- Features in TAG:
 - ▶ an obere und untere Hälften von Knoten anhängen
 - ▶ Koindizierung innerhalb des gleichen Elementarbaums
 - ▶ elegante Kontrolle von Fernabhängigkeiten