

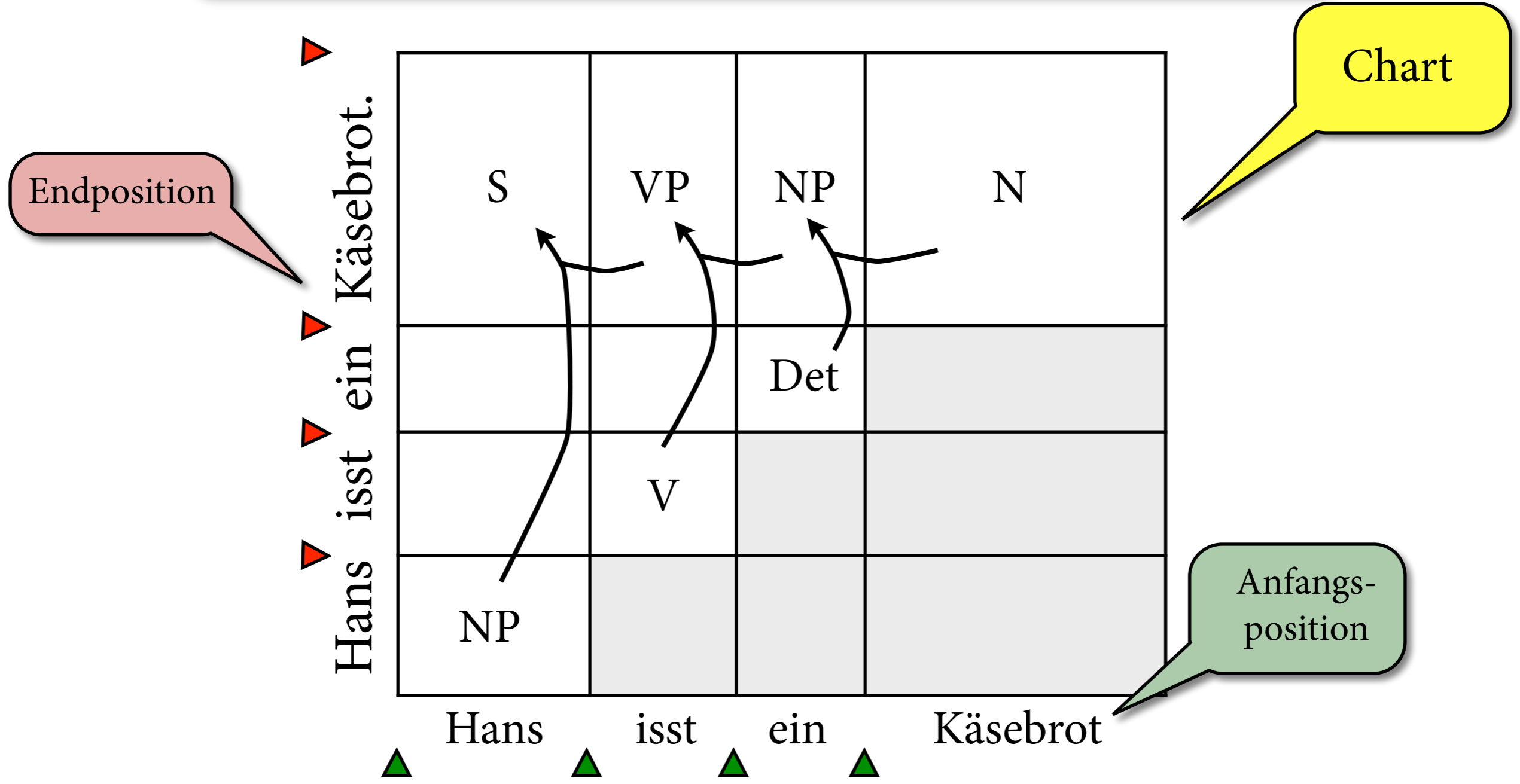
# Der Earley-Parser

Vorlesung “Computerlinguistische Techniken”  
Alexander Koller

07. November 2014

# Der CKY-Parser

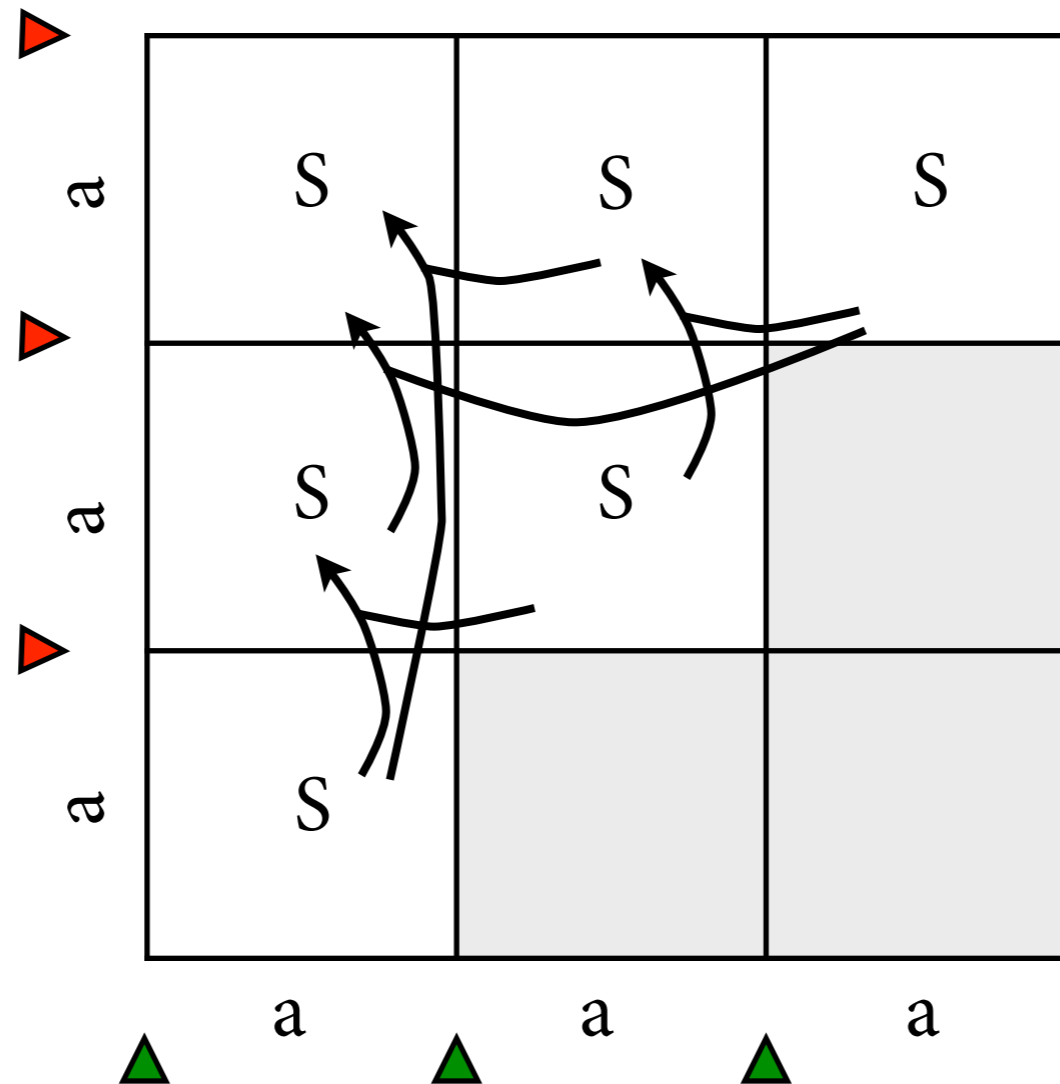
$S \rightarrow NP VP$        $V \rightarrow isst$        $Det \rightarrow ein$   
 $NP \rightarrow Det N$        $NP \rightarrow Hans$        $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$



# Der CKY-Parser

$S \rightarrow SS$

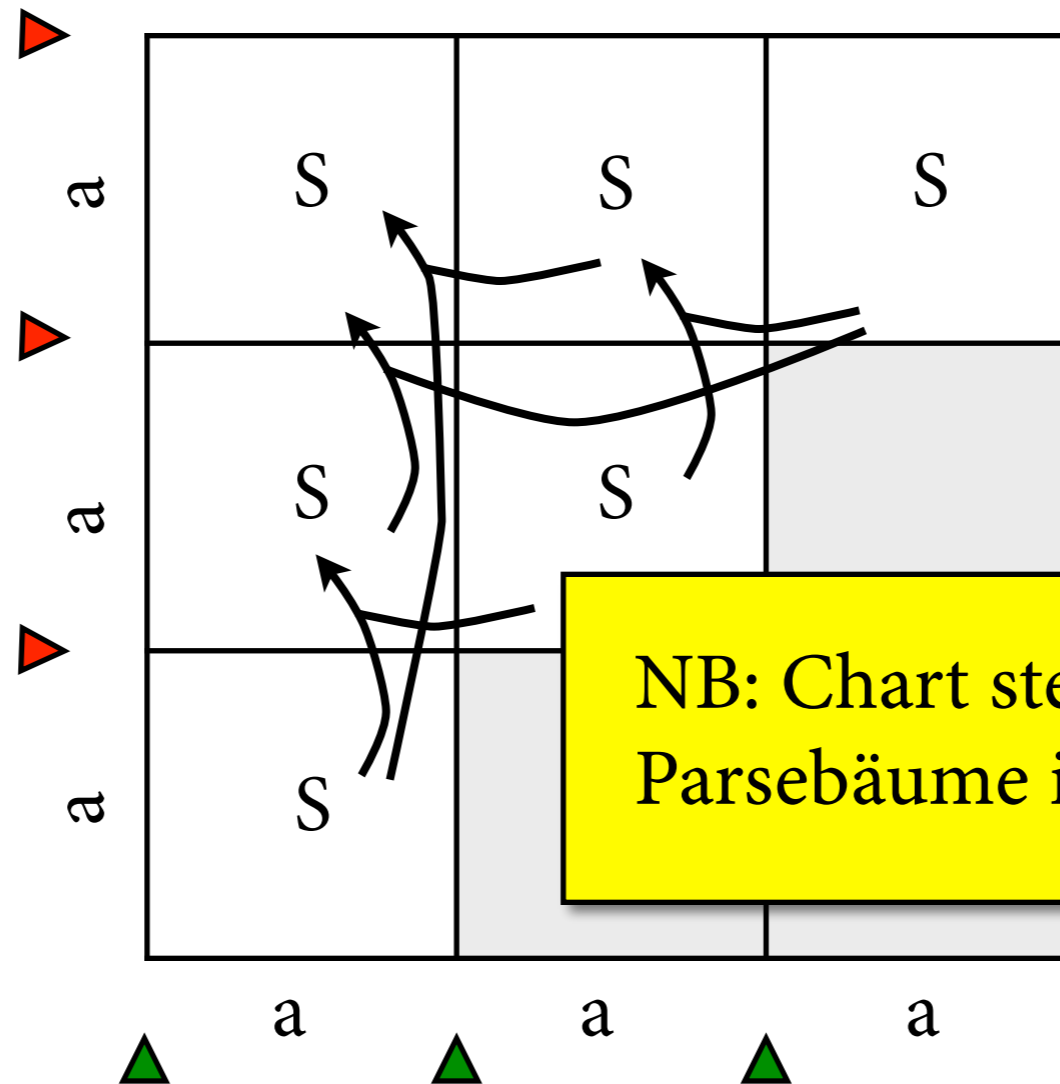
$S \rightarrow a$



# Der CKY-Parser

$S \rightarrow SS$

$S \rightarrow a$



NB: Chart stellt exponentiell viele Parsebäume in kubischem Platz dar!

# Implementierung

- Idee der Datenstruktur:
  - ▶  $\text{chart}[i][k] = \text{Ch}(i+1, k+1)$
  - ▶ chart ist eine Liste von Zeilen der Charts
  - ▶ Jede Zeile ist eine Liste von Mengen von NT-Symbolen.

- Initialisierung:

```
chart = []
```

```
for i in range(n+1):  
    row = []  
    for j in range(n+1):  
        row.append(set())  
    chart.append(row)
```

# Implementierung

```
for i in range(n):                                # Terminalregeln
    for prod in grammar productions(rhs=words[i]):
        chart[i][i+1].add(prod.lhs())
```

```
for width in range(2, n+1):                       # Binaere Regeln
    for i in range(0, n-width+1):
        for j in range(1, width):
            nts1 = chart[i][i+j]
            nts2 = chart[i+j][i+width]
            for nt1 in nts1:
                productions = grammar productions(rhs=nt1)
                for production in productions:
                    if production.rhs()[1] in nts2:
                        chart[i][i+width].add(production.lhs())
```

# Implementierung: Parser

- Für einen *Parser* merkt man sich zu jedem Nichtterminal in einer Chartzelle, auf welche (mehreren) Weisen man es bauen kann.
- `chart[i][k]` ist jetzt Dictionary (statt Menge).
  - ▶ `keys(chart[i][k])`: Nichtterminale  $A$ , die den Teilstring  $w_i \dots w_{k-1}$  abdecken können
  - ▶ Eintrag `chart[i][k][“A”]`: Liste von *Backpointern*, d.h. Tripeln  $(B, C, j)$ , die angeben, dass  $A$  aus  $B$  von  $i$  bis  $j$  plus  $C$  von  $j$  bis  $k$  gebaut werden kann.

# Nicht-binäre Grammatiken

- CKY-Algorithmus verlangt, dass Grammatiken in CNF vorliegen.
  - ▶ insbesondere: Regeln sind alle binär
- Ist das wichtig? Was würde passieren, wenn man CKY mit längeren rechten Seiten verwenden würde?



# CKY: nichtbinäre Version

für alle  $b$  von 2 bis  $n$ :

für alle  $i_1$  von 1 bis  $n-b+1$ :

für alle Produktionen  $A \rightarrow B_1 \dots B_r$ :

$$i_{r+1} = i_1 + b$$

für alle  $i_2$  mit  $i_1 < i_2 < i_{r+1}$ :

für alle  $i_3$  mit  $i_2 < i_3 < i_{r+1}$ :

...

für alle  $i_r$  mit  $i_{r-1} < i_r < i_{r+1}$ :

wenn  $B_j \in C(i_j, i_{j+1})$  für alle  $1 \leq j \leq r$ :

füge  $A$  zu  $C(i, i+b)$  hinzu

# CKY: nichtbinäre Version

Laufzeit?

für alle  $b$  von 2 bis  $n$ :

für alle  $i_1$  von 1 bis  $n-b+1$ :

für alle Produktionen  $A \rightarrow B_1 \dots B_r$ :

$i_{r+1} = i_1 + b$

für alle  $i_2$  mit  $i_1 < i_2 < i_{r+1}$ :

für alle  $i_3$  mit  $i_2 < i_3 < i_{r+1}$ :

...

für alle  $i_r$  mit  $i_{r-1} < i_r < i_{r+1}$ :

wenn  $B_j \in C(i_j, i_{j+1})$  für alle  $1 \leq j \leq r$ :

füge  $A$  zu  $C(i, i+b)$  hinzu

# CKY: nichtbinäre Version

Laufzeit?

für alle  $b$  von 2 bis  $n$ :

für alle  $i_1$  von 1 bis  $n-b+1$ :

für alle Produktionen  $A \rightarrow B_1 \dots B_r$ :

$$i_{r+1} = i_1 + b$$

für alle  $i_2$  mit  $i_1 < i_2 < i_{r+1}$ :

für alle  $i_3$  mit  $i_2 < i_3 < i_{r+1}$ :

...

für alle  $i_r$  mit  $i_{r-1} < i_r < i_{r+1}$ :

wenn  $B_j \in C(i_j, i_{j+1})$  für alle  $1 \leq j \leq r$ :

füge  $A$  zu  $C(i, i+b)$  hinzu

# CKY: nichtbinäre Version

Laufzeit?

für alle  $b$  von 2 bis  $n$ :

für alle  $i_1$  von 1 bis  $n-b+1$ :

für alle Produktionen  $A \rightarrow B_1 \dots B_r$ :

$$i_{r+1} = i_1 + b$$

für alle  $i_2$  mit  $i_1 < i_2 < i_{r+1}$ :

für alle  $i_3$  mit  $i_2 < i_3 < i_{r+1}$ :

...

für alle  $i_r$  mit  $i_{r-1} < i_r < i_{r+1}$ :

wenn  $B_j \in C(i_j, i_{j+1})$  für alle  $1 \leq j \leq r$ :

füge  $A$  zu  $C(i, i+b)$  hinzu

$O(n^{r+1})$  !

# Nichtbinärer CKY-Algorithmus

- CKY-Algorithmus versucht in jedem Schritt,  $r$  Konstituenten zusammenzubauen;  
im worst-case  $O(n^{r+1})$  Kombinationen.
  - ▶ Rang einer Regel = Länge der rechten Seite
  - ▶  $r = \text{Rang der Grammatik} = \max. \text{Rang der Regeln}$
- Wenn Grammatik in CNF ist, ist  $r = 2$ ,  
Laufzeit wird dann  $O(n^3)$ .
- Geht es auch für beliebige Grammatiken in Zeit  $O(n^3)$ ?

# Chomsky-Normalform

- Eine kfG  $G$  heißt in *Chomsky-Normalform*, wenn sie nur zwei Typen von Regeln hat:
  - ▶  $A \rightarrow BC$ , d.h. genau zwei Nichtterminale
  - ▶  $A \rightarrow a$ , d.h. genau ein Terminalsymbol
- Für jede kfG  $G$  kann man eine kfG  $G'$  in CNF mit  $L(G) = L(G')$  berechnen.
- Aber die Parsebäume von  $G$  und  $G'$  sind verschieden.

# CNF: Schritt 1

- Zunächst kümmern wir uns um Auftreten von Terminalsymbolen.
- Für jede Regel der Form  $A \rightarrow \alpha a \beta$ :
  - ▶ führe neue Regel  $A' \rightarrow a$  ein;  $A'$  ist neues NT-Symbol
  - ▶ führe neue Regel  $A \rightarrow \alpha A' \beta$  ein
  - ▶ entferne  $A \rightarrow \alpha a \beta$  aus der Grammatik
- Danach treten Terminale nur noch in Regeln der Form  $A \rightarrow a$  auf.

# CNF: Schritt 2

- Jetzt eliminieren wir Regeln von Rang  $> 2$ .
- Für jede Regel der Form  $A \rightarrow B_1 \dots B_n$ ,  $n > 2$ :
  - ▶ Entferne die Regel aus der Grammatik.
  - ▶ Führe  $n-1$  neue Regeln ein:  
 $A \rightarrow B_1 [A \rightarrow B_1 \bullet B_2 \dots B_n]$   
 $[A \rightarrow B_1 \bullet B_2 \dots B_n] \rightarrow B_2 [A \rightarrow B_1 B_2 \bullet \dots B_n]$   
...  
 $[A \rightarrow B_1 \dots \bullet B_{n-1} B_n] \rightarrow B_{n-1} B_n$
- Danach haben alle Regeln Rang  $\leq 2$ .



# CNF: Schritt 3

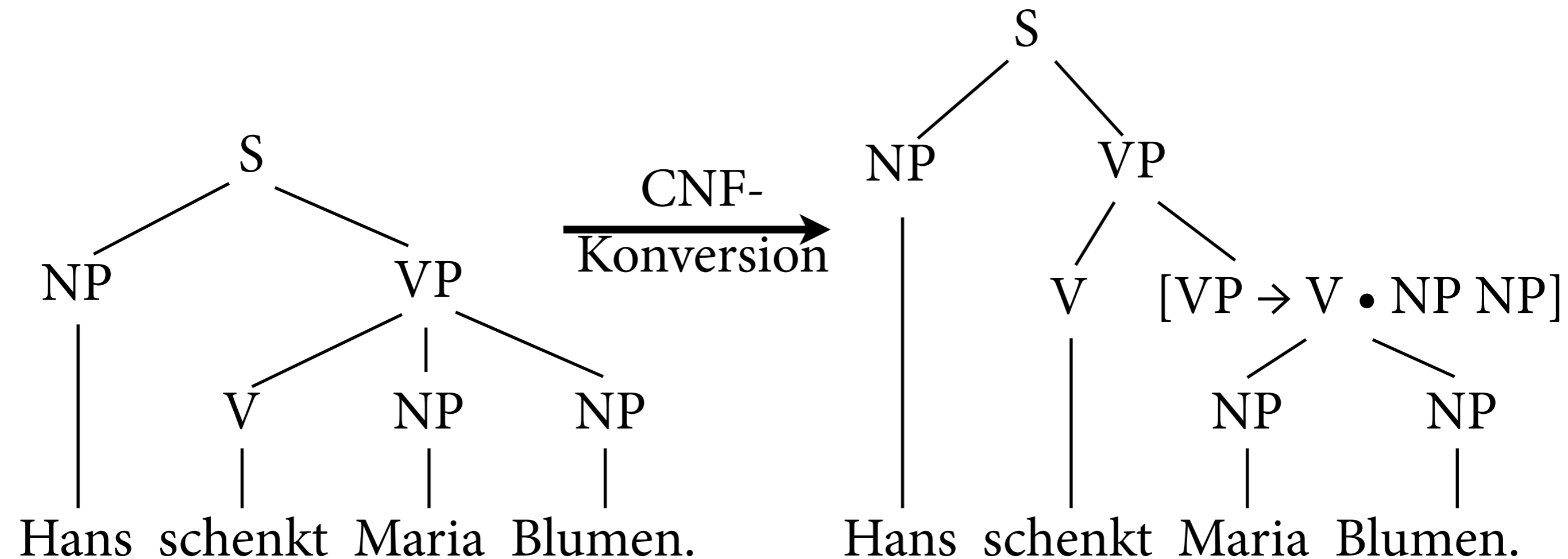
- Zuletzt eliminieren wir Kettenregeln von der Form  $A \rightarrow B$ .
- Für Regeln  $A \rightarrow \alpha B \beta$  und  $B \rightarrow C$ :
  - ▶ füge Regel  $A \rightarrow \alpha C \beta$  hinzu
- Dann lösche alle Kettenregeln.
- Nun ist kfG in CNF. Sprache ist in jedem Schritt gleich geblieben.

# Äquivalenz von kfGs

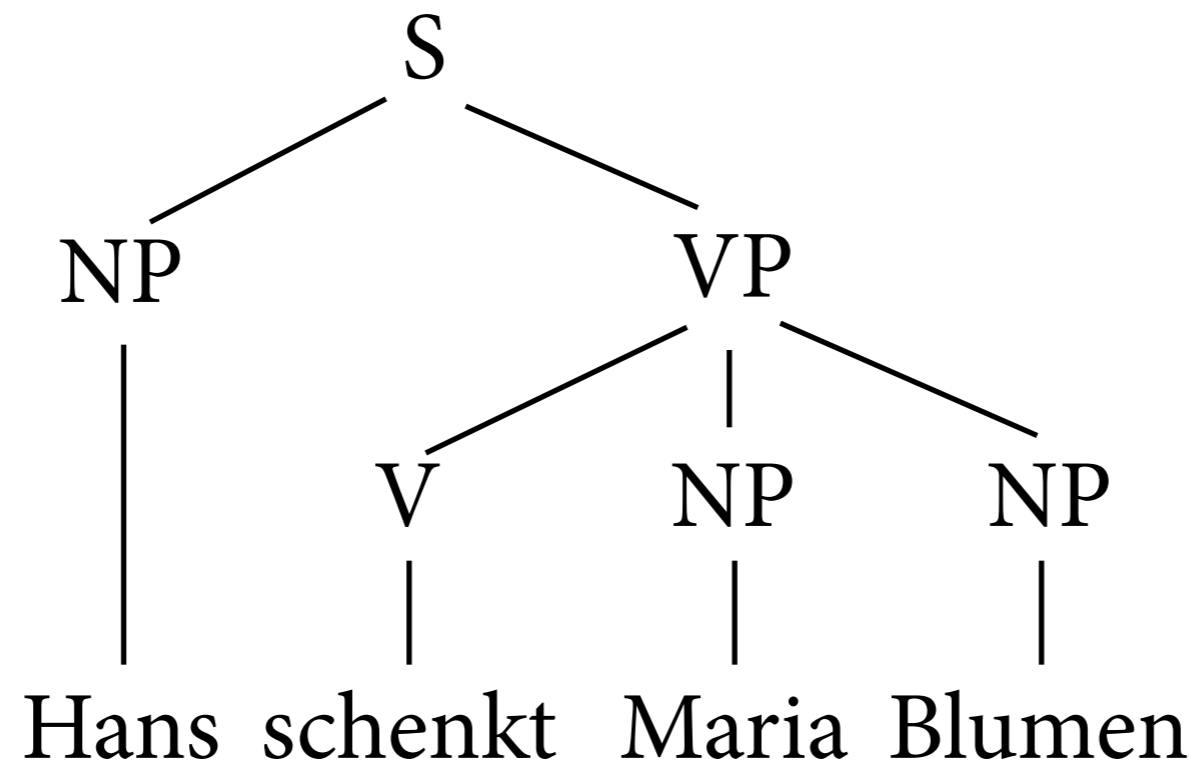
- Zwei kfGs  $G$  und  $G'$  heißen *schwach äquivalent*, wenn  $L(G) = L(G')$ .
  - ▶ d.h. Wortproblem von  $G$  und  $G'$  gleich
- Zwei kfGs  $G$  und  $G'$  heißen *stark äquivalent*, wenn sie jedem String die gleichen Parsebäume zuweisen.
  - ▶ d.h. Parsingproblem von  $G$  und  $G'$  gleich

# CNF: Parsebäume

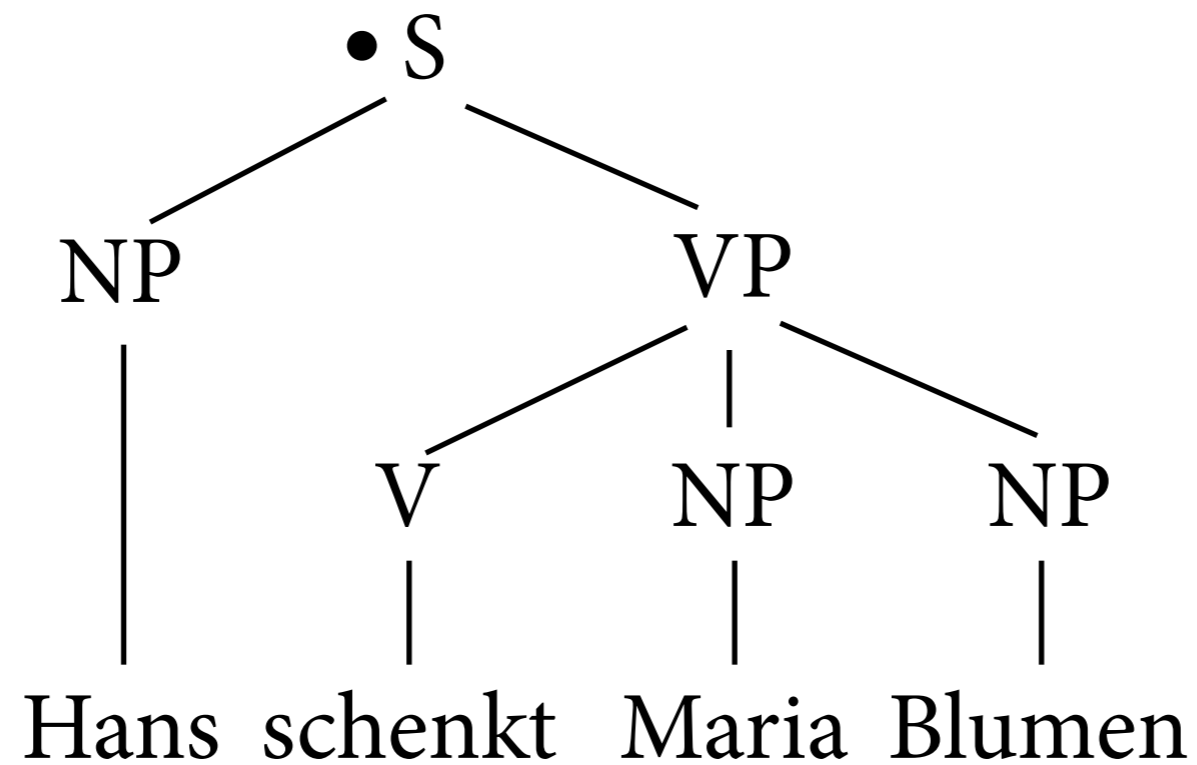
- CNF einer kfG ist schwach, aber nicht stark äquivalent.



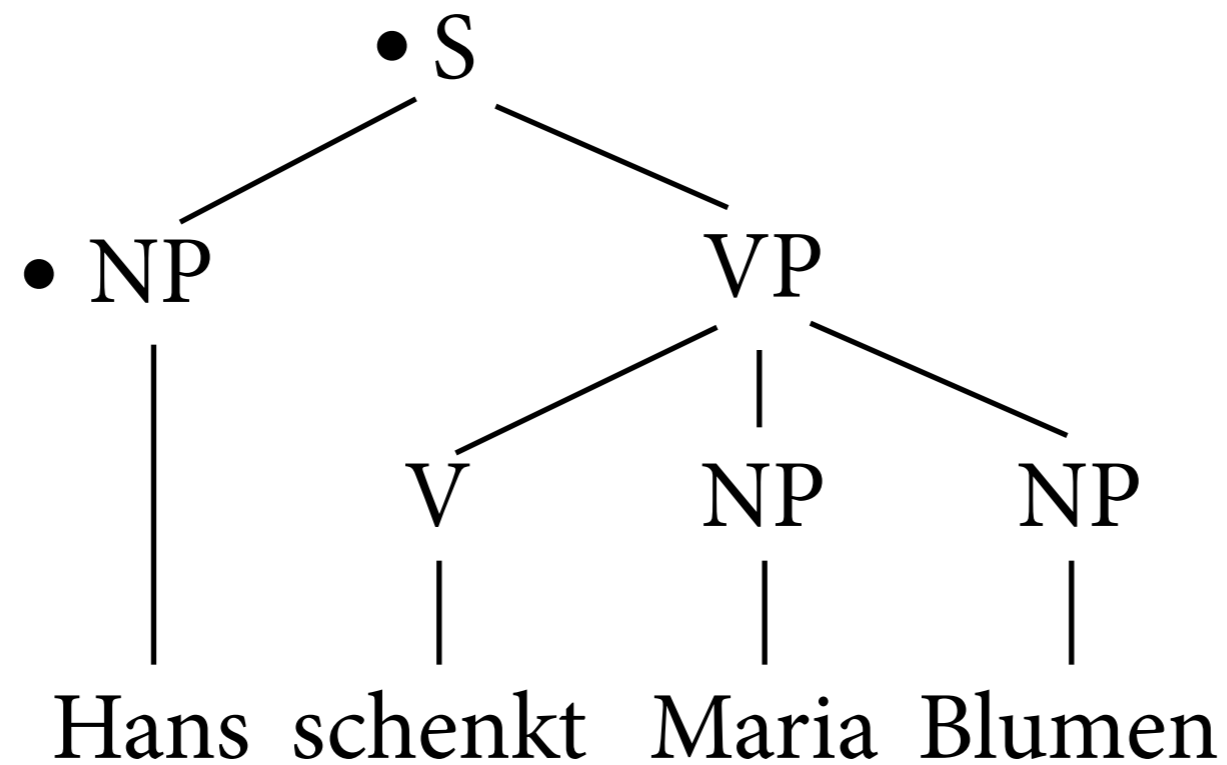
# Parsebaum schrittweise bauen?



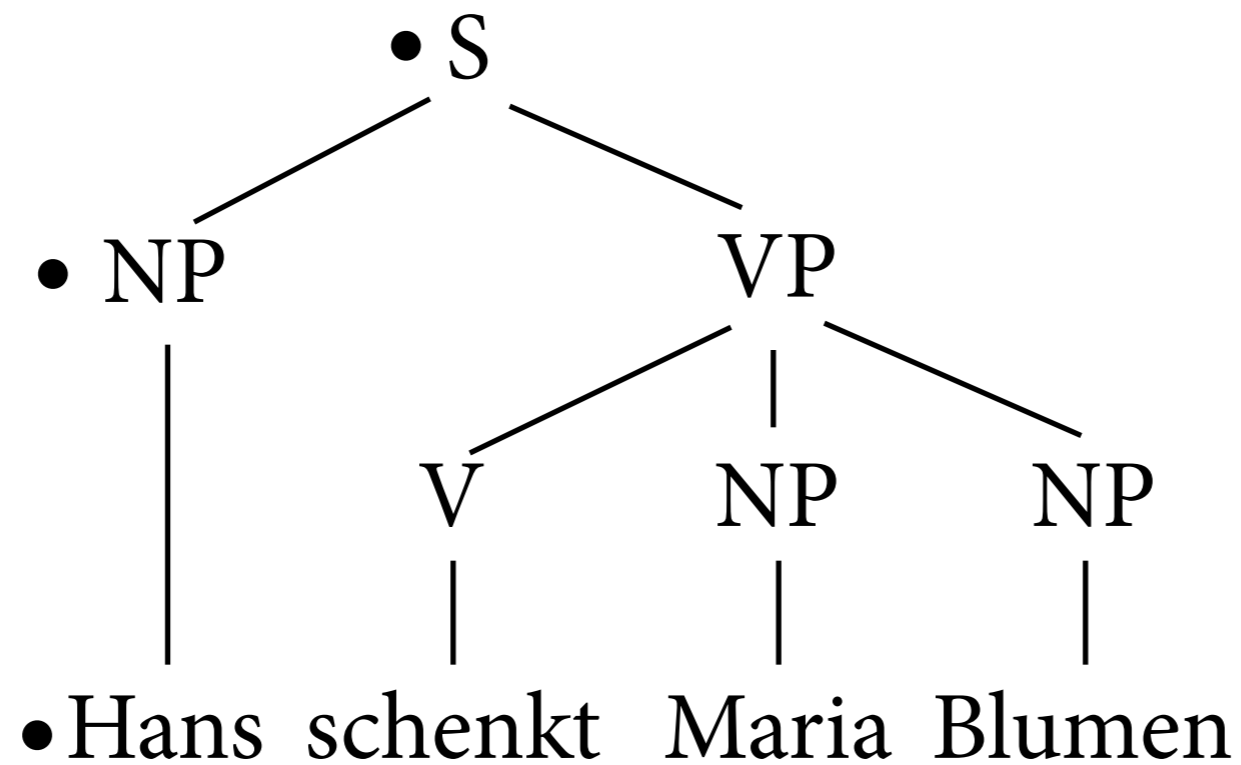
# Parsebaum schrittweise bauen?



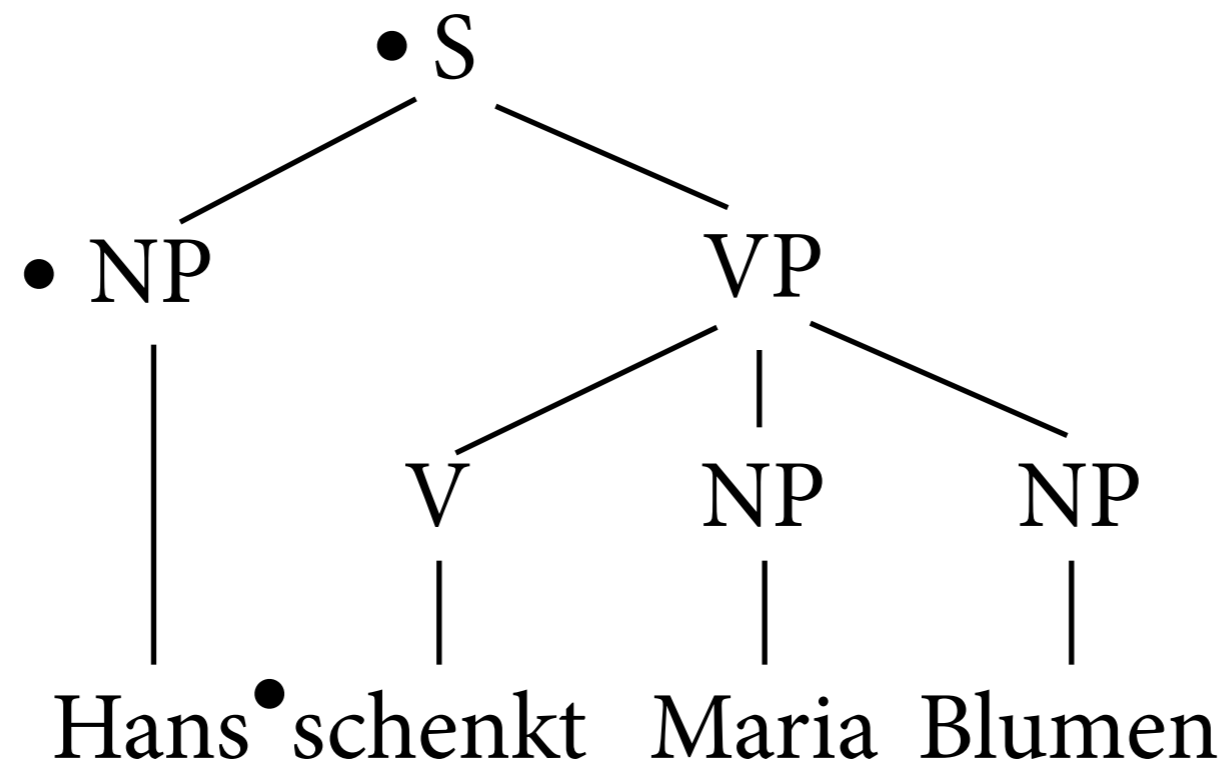
# Parsebaum schrittweise bauen?



# Parsebaum schrittweise bauen?

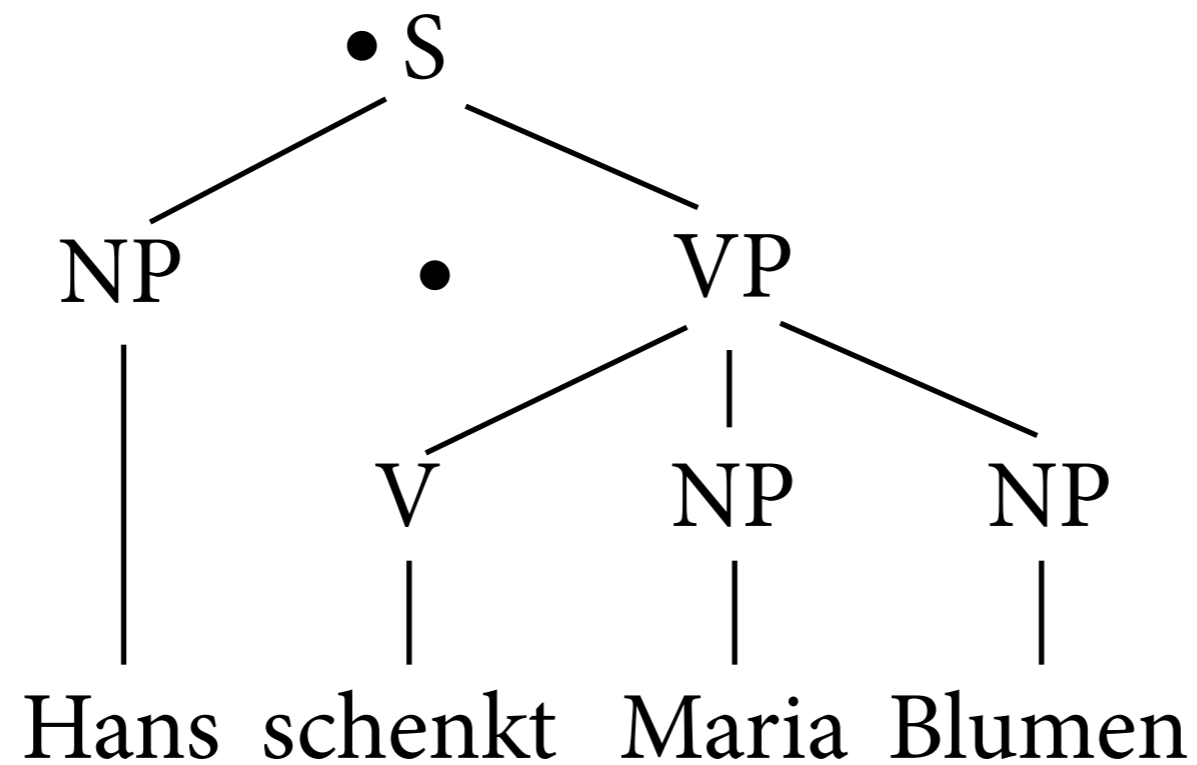


# Parsebaum schrittweise bauen?

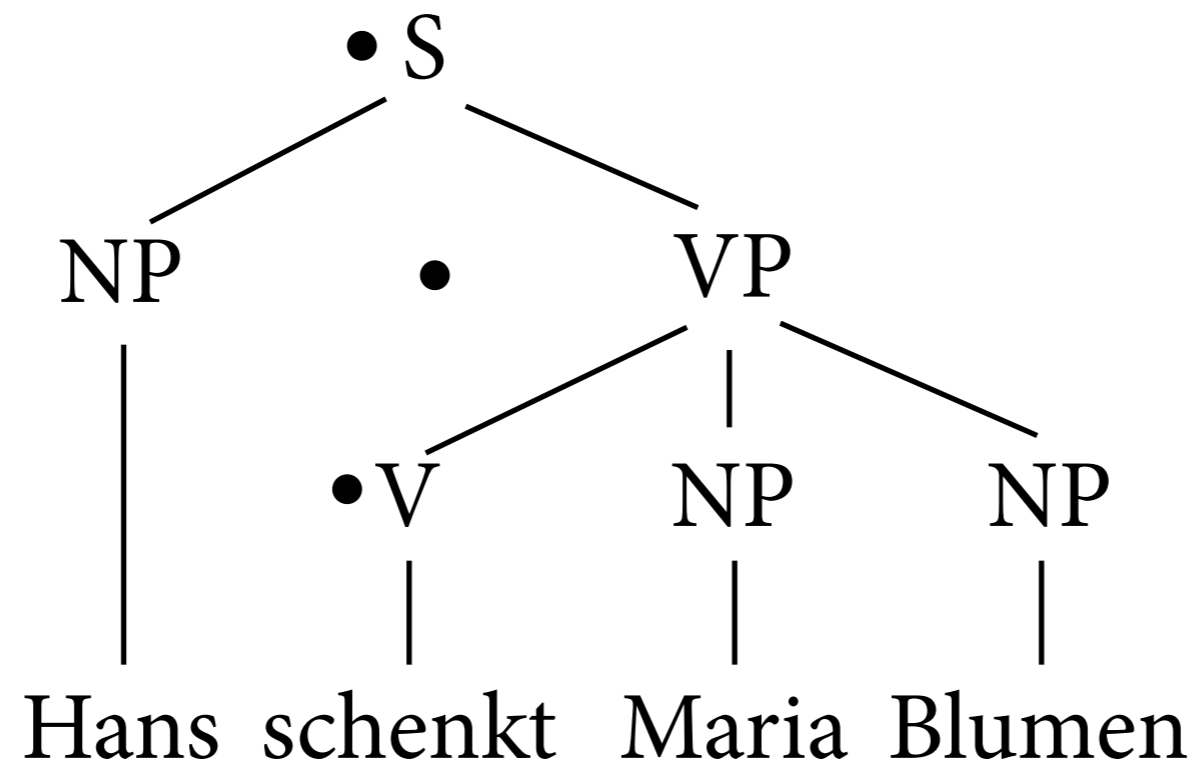




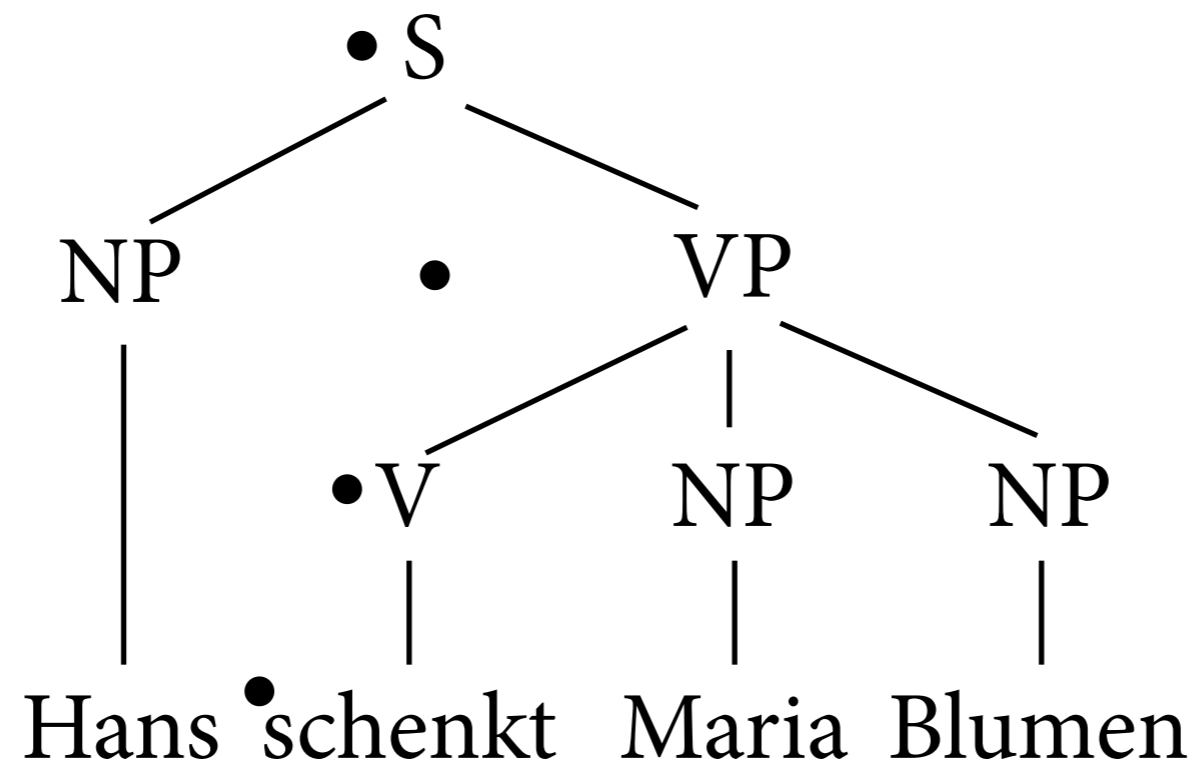
# Parsebaum schrittweise bauen?



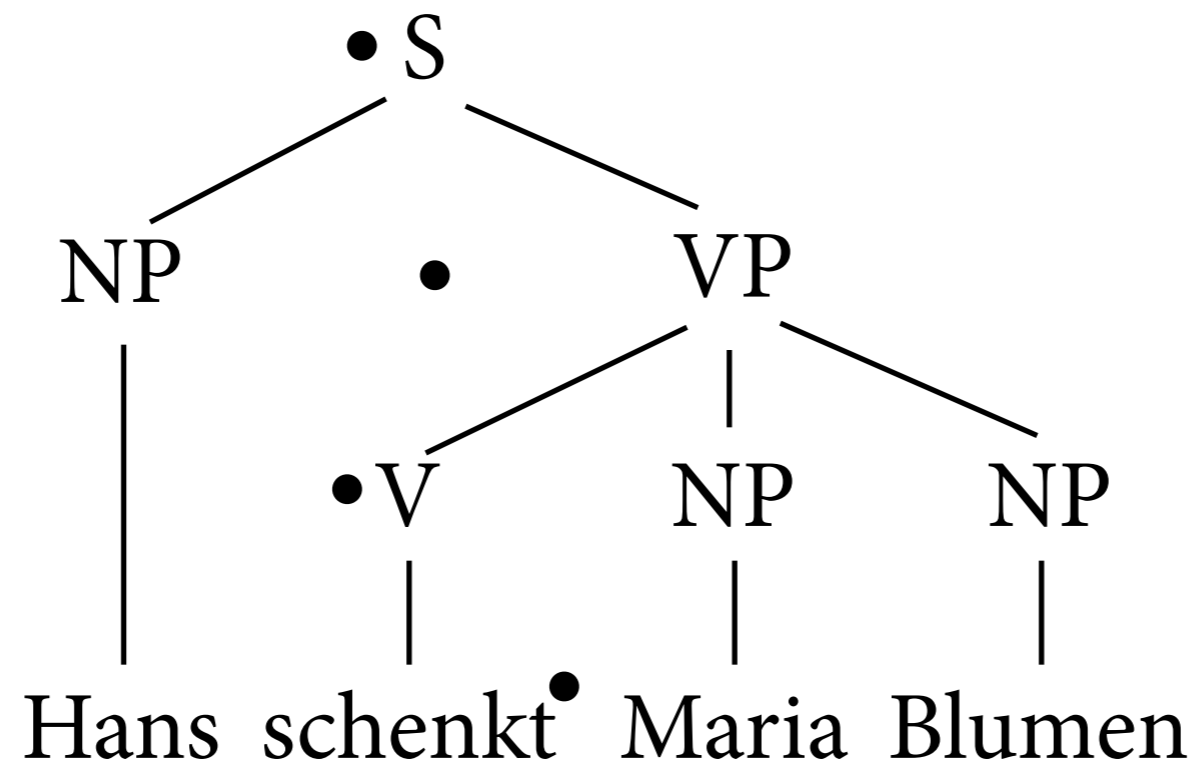
# Parsebaum schrittweise bauen?



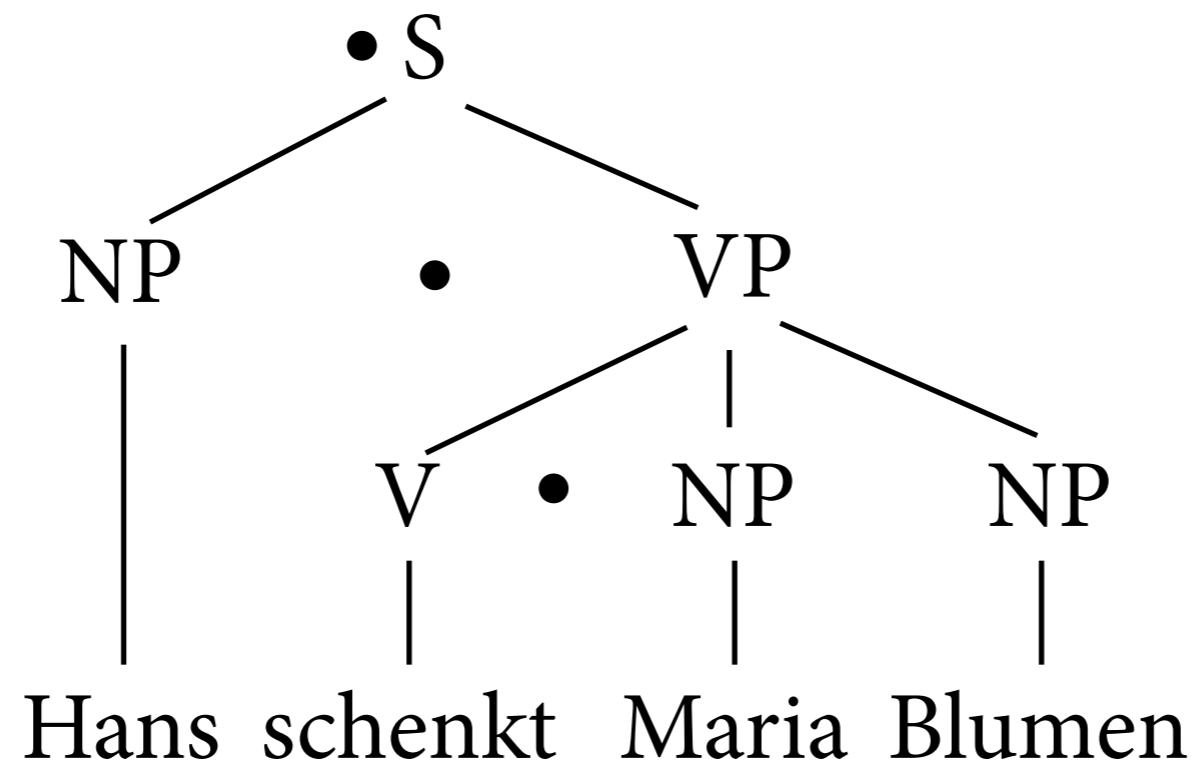
# Parsebaum schrittweise bauen?



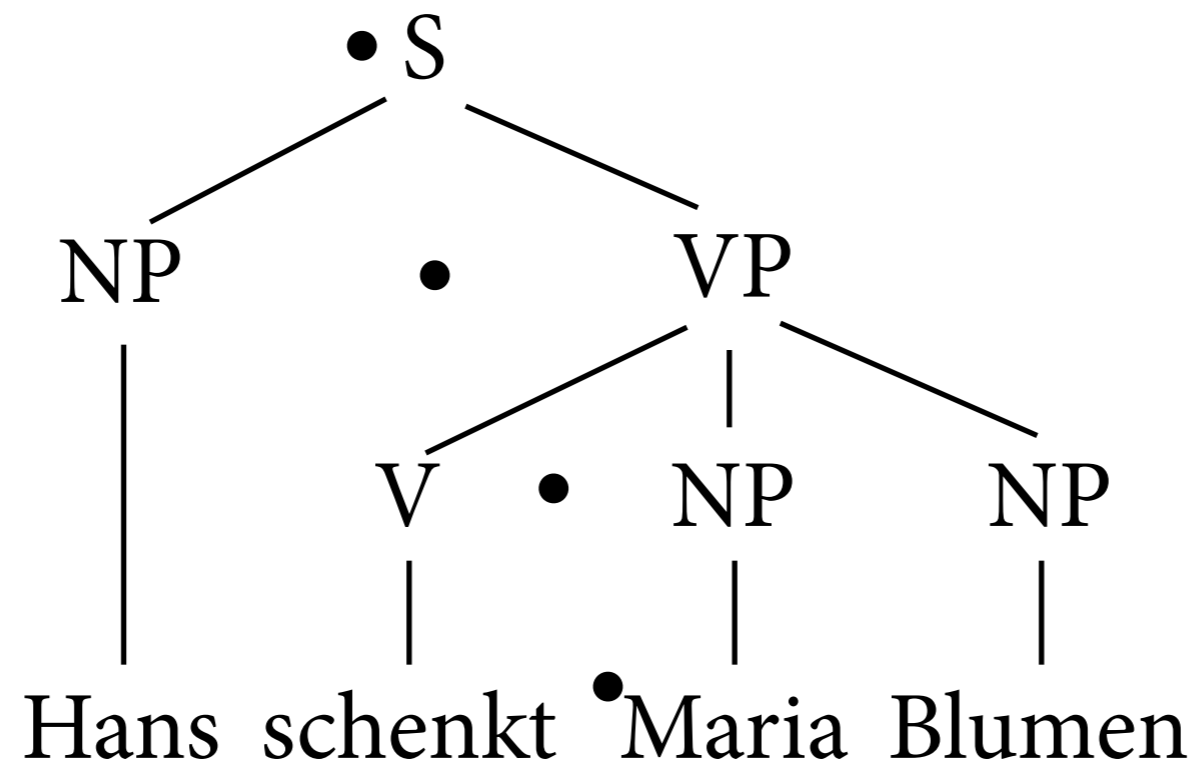
# Parsebaum schrittweise bauen?



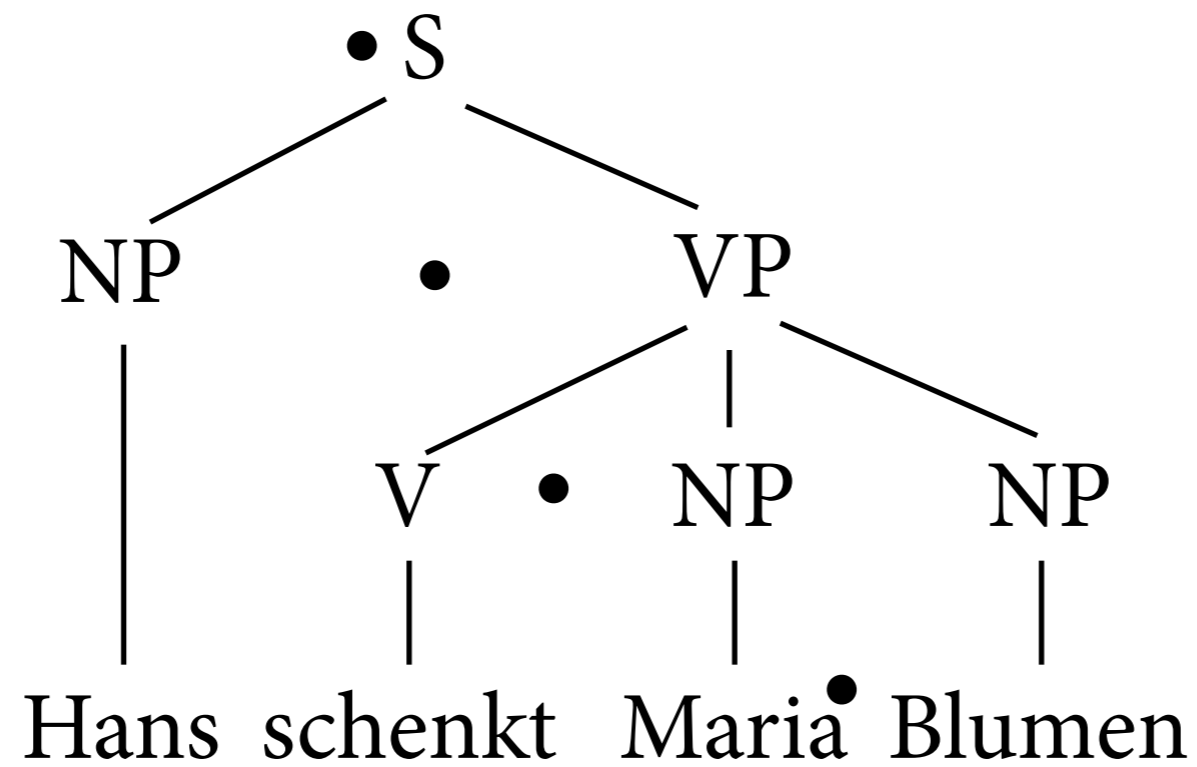
# Parsebaum schrittweise bauen?



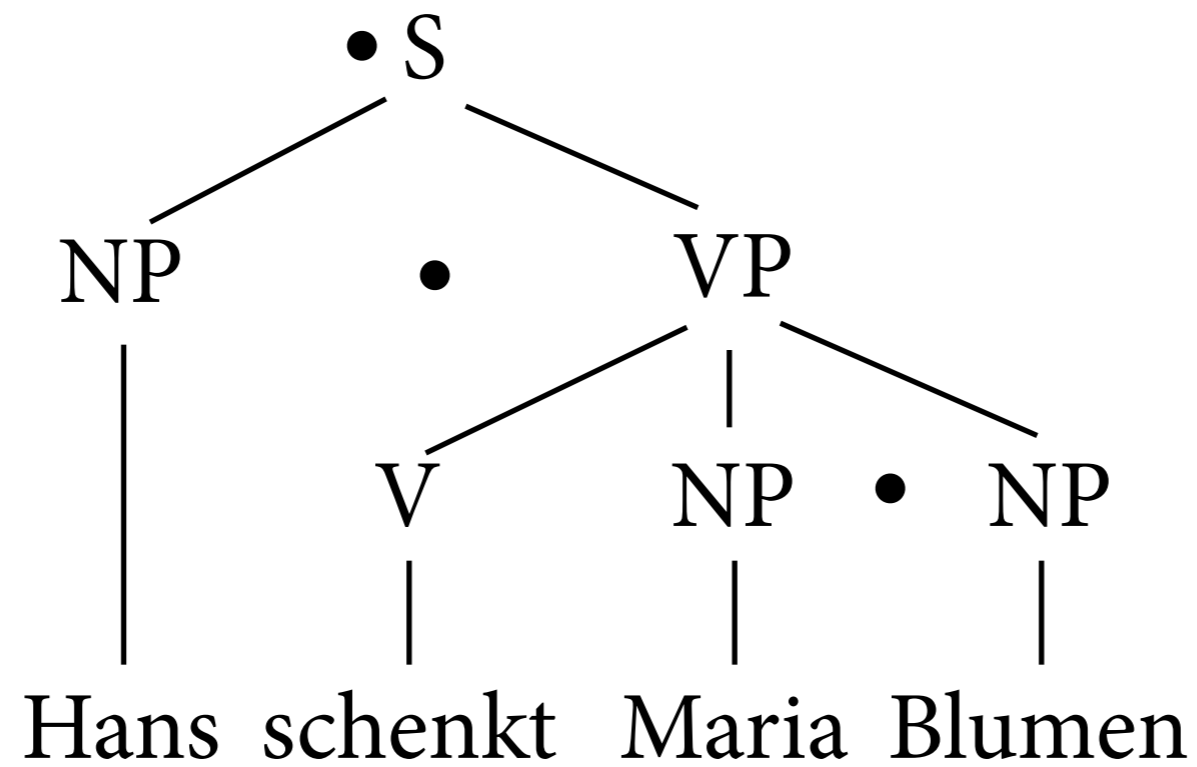
# Parsebaum schrittweise bauen?



# Parsebaum schrittweise bauen?

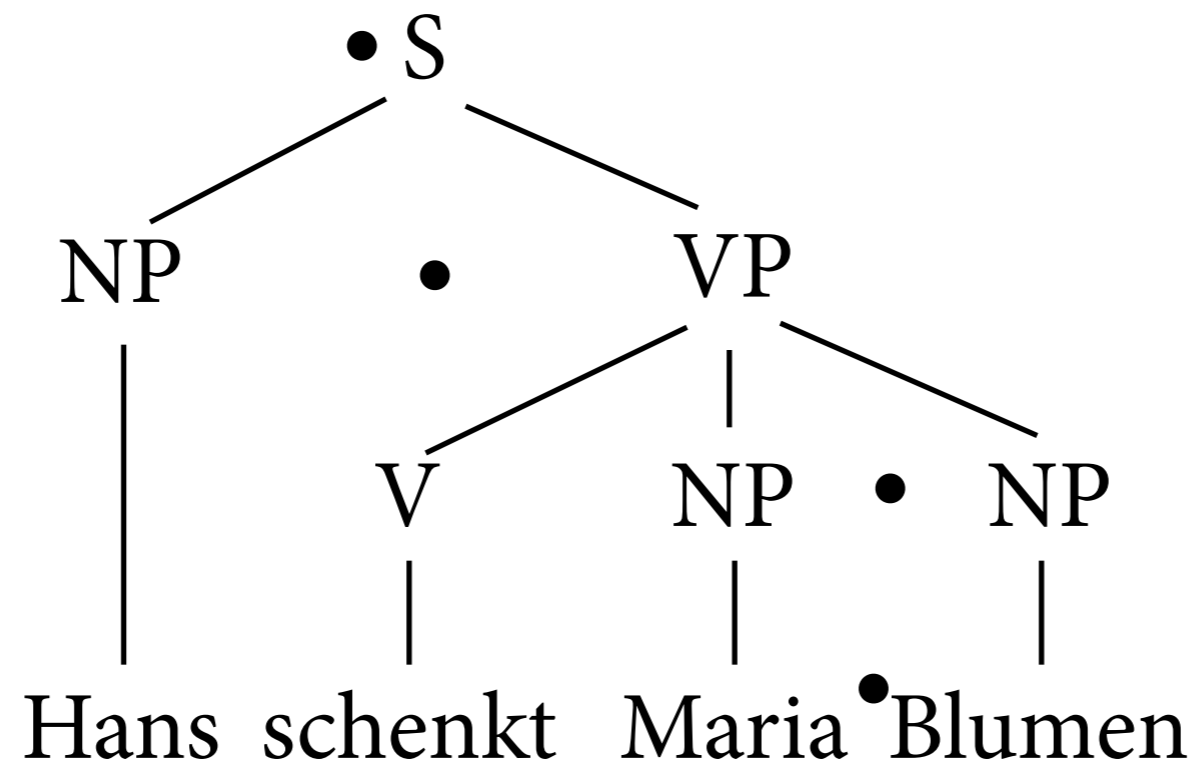


# Parsebaum schrittweise bauen?

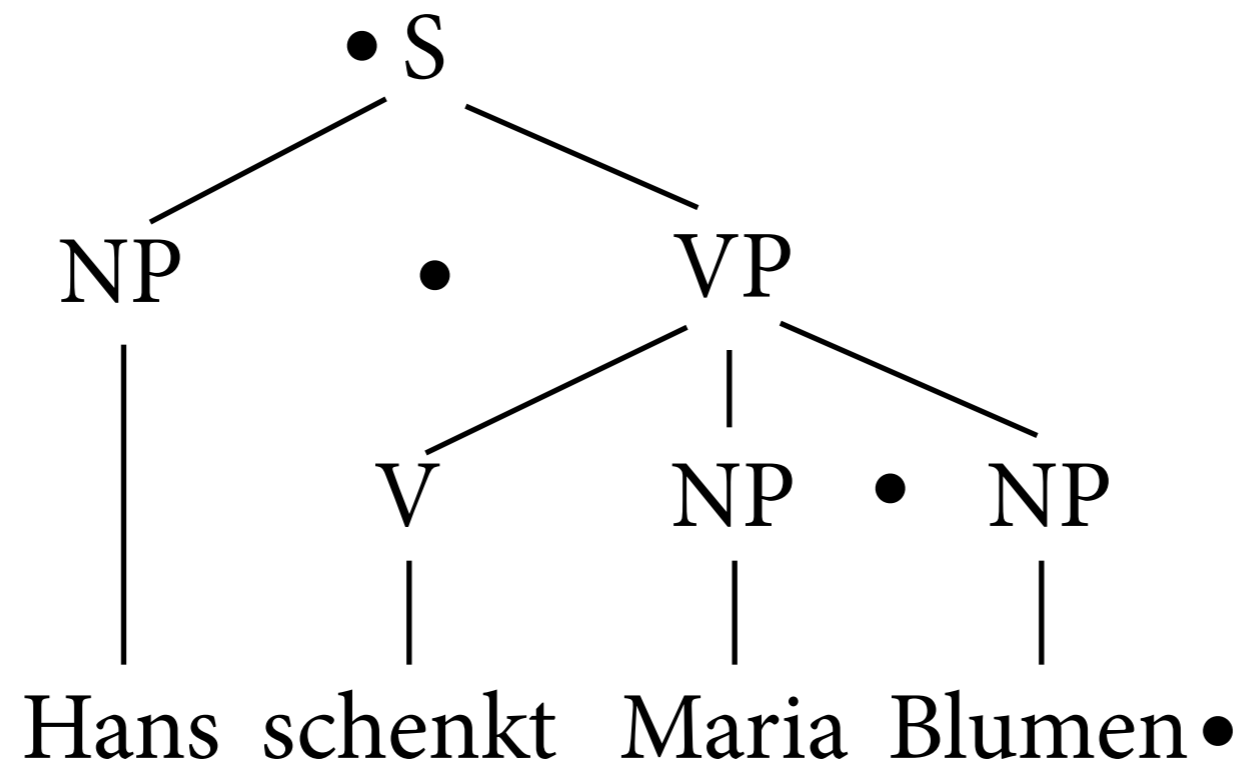




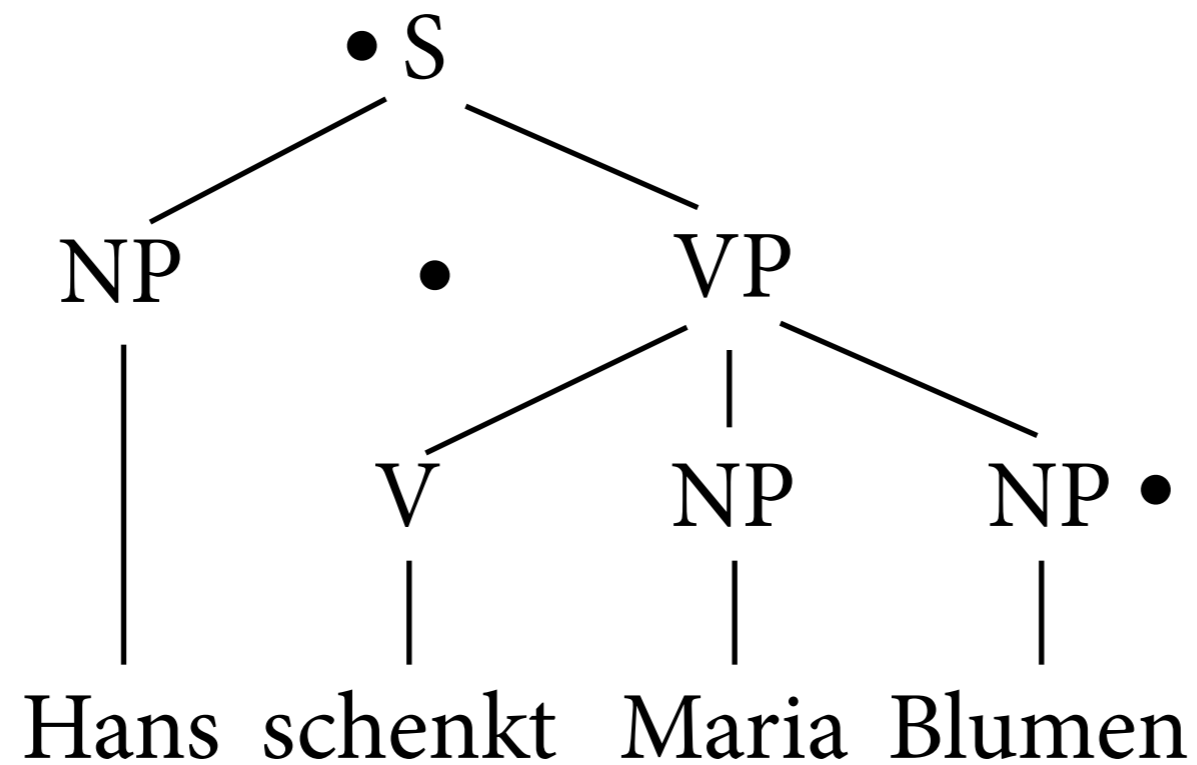
# Parsebaum schrittweise bauen?



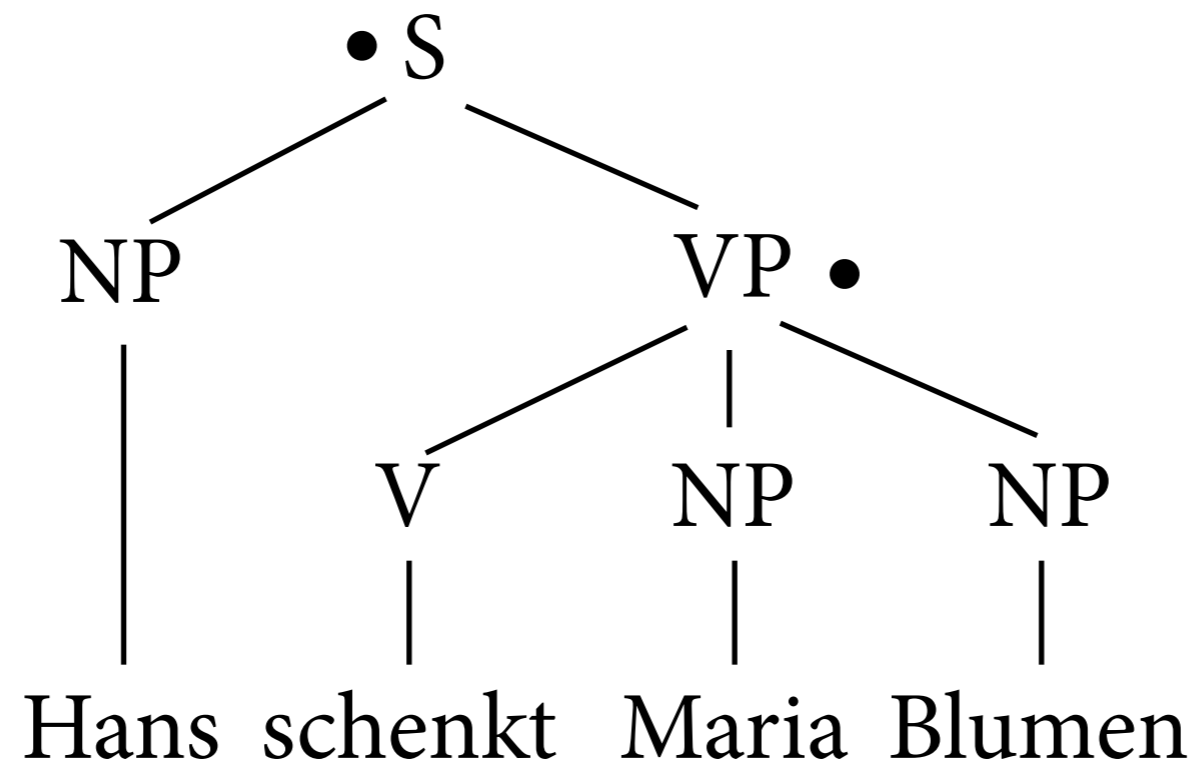
# Parsebaum schrittweise bauen?



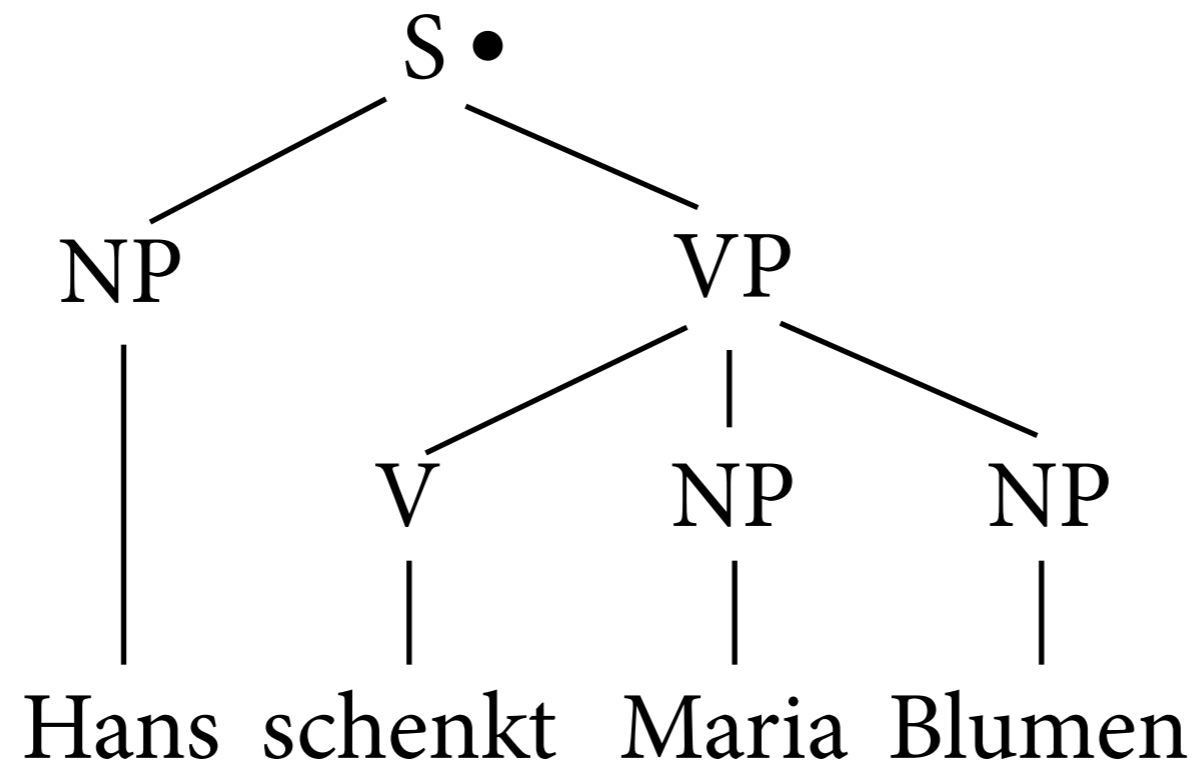
# Parsebaum schrittweise bauen?



# Parsebaum schrittweise bauen?

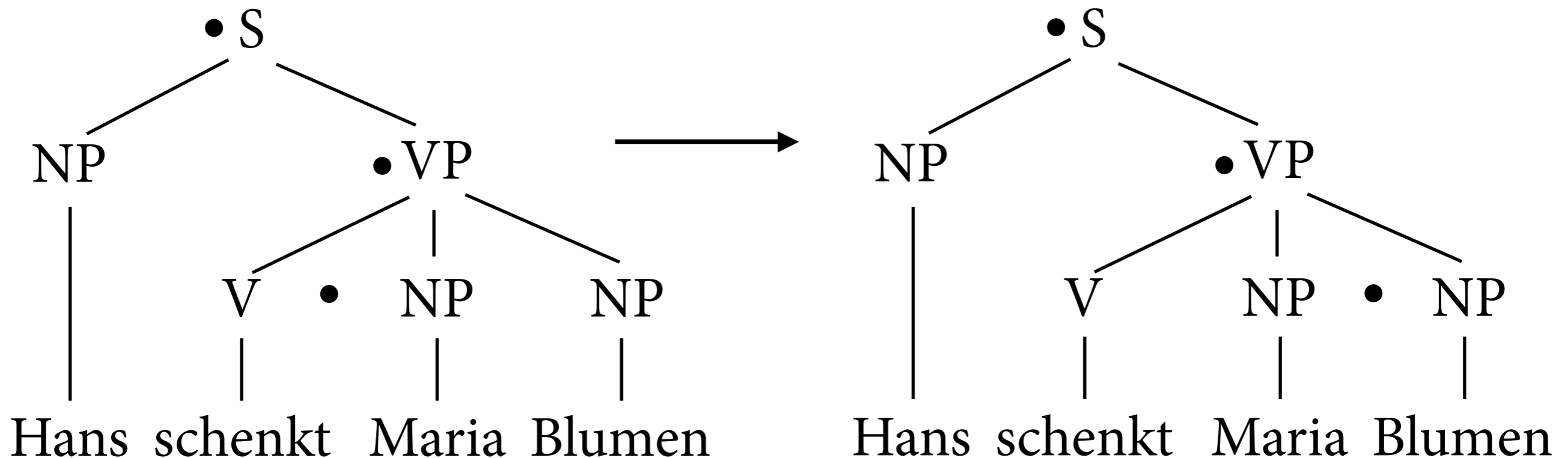


# Parsebaum schrittweise bauen?

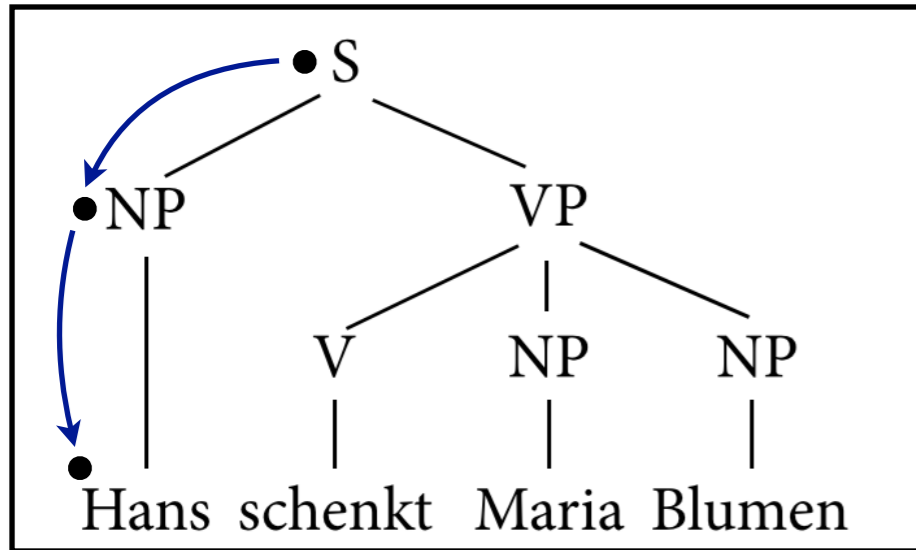


# Nicht-binäre Regeln

- Algorithmus auf dieser Grundlage könnte direkt aus nicht-CNF-Grammatik den Originalbaum berechnen.
- Könnte auch in kubischer Zeit laufen, weil er immer nur zwei Teilstrings auf einmal kombiniert:

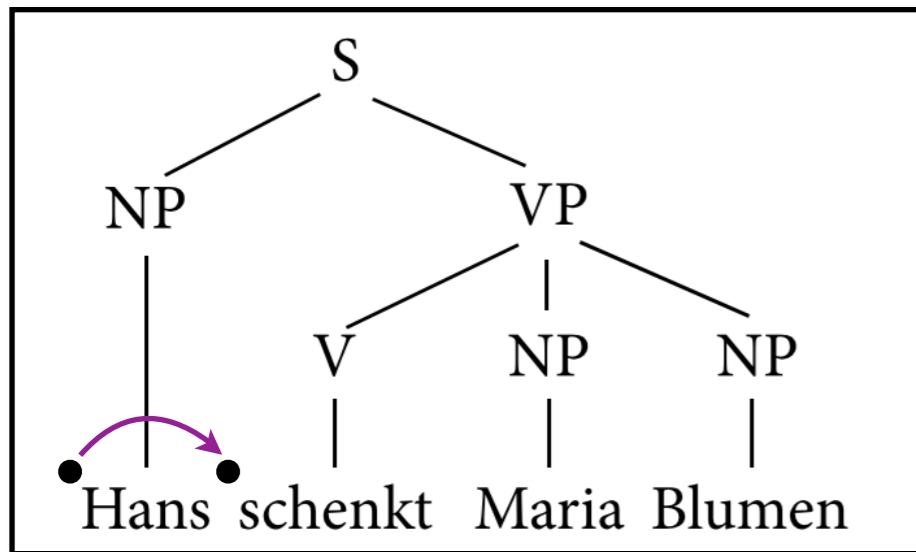


# Operationen



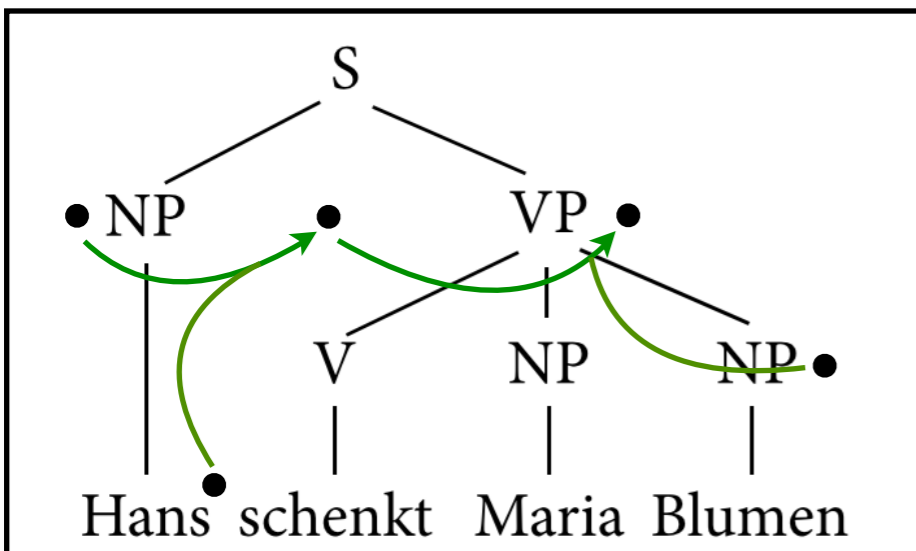
*Predict:*

Will ein S ab Stringposition  $i$  bauen,  
habe Regel  $S \rightarrow NP VP$ :  
kann versuchen, NP ab  $i$  zu bauen.



*Scan:*

Will "Hans" ab Stringposition  $i$  bauen,  
an Position  $i$  steht tatsächlich  $i$ :  
habe erfolgreich "Hans" von  $i$  bis  $i+1$  geparst.



*Complete:*

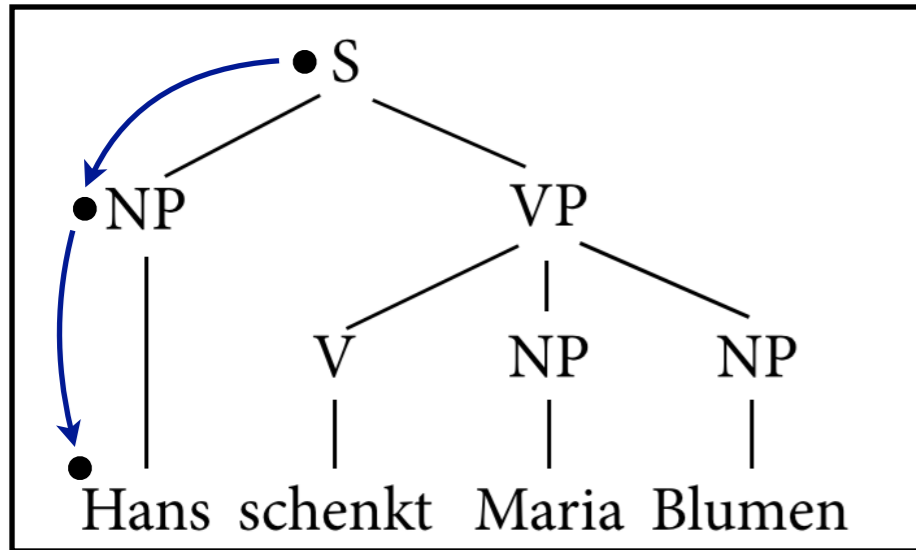
Will S ab Position  $i$  mit Regel  $S \rightarrow NP VP$   
bauen. Habe von  $i$  bis  $j$  eine NP-Regel  
vollständig geparst. Deshalb habe ich in  
 $S \rightarrow NP VP$  Fortschritt gemacht.

# Der Earley-Parser

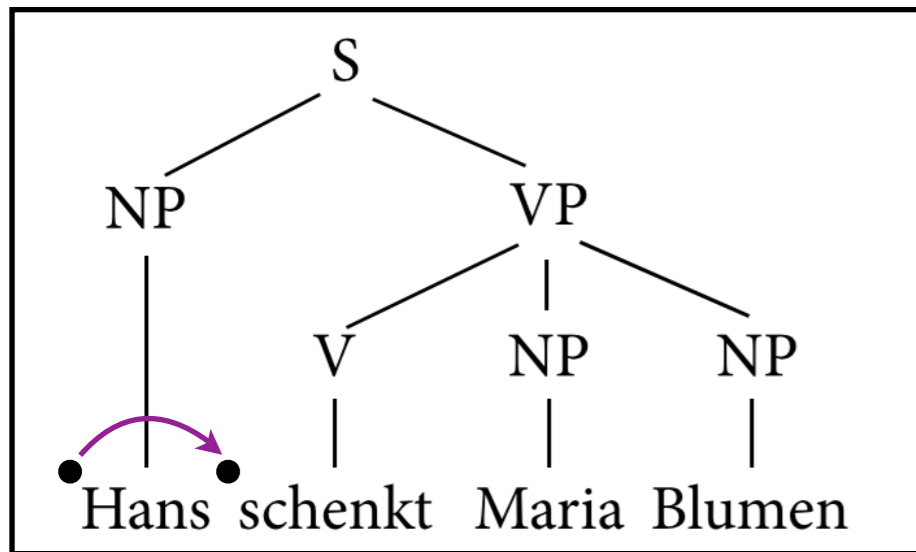
- Items haben die Form  $[i, A \rightarrow \alpha \bullet B \beta, k]$ :
  - ▶ Item repräsentiert einen Versuch, eine A-Konstituente ab Stringposition  $i$  zu lesen
  - ▶ indem als erste Regel  $A \rightarrow \alpha B \beta$  angewendet wird
  - ▶ bis vor Position  $k$  sind wir schon gekommen, dabei haben wir ein  $\alpha$  gelesen
  - ▶ d.h.:  $A \Rightarrow \alpha B \beta \Rightarrow^* w_i \dots w_{k-1} B \beta$  (wie vorhin)
- Drei Regeln, um Items zu manipulieren: Predict, Scan, Complete.



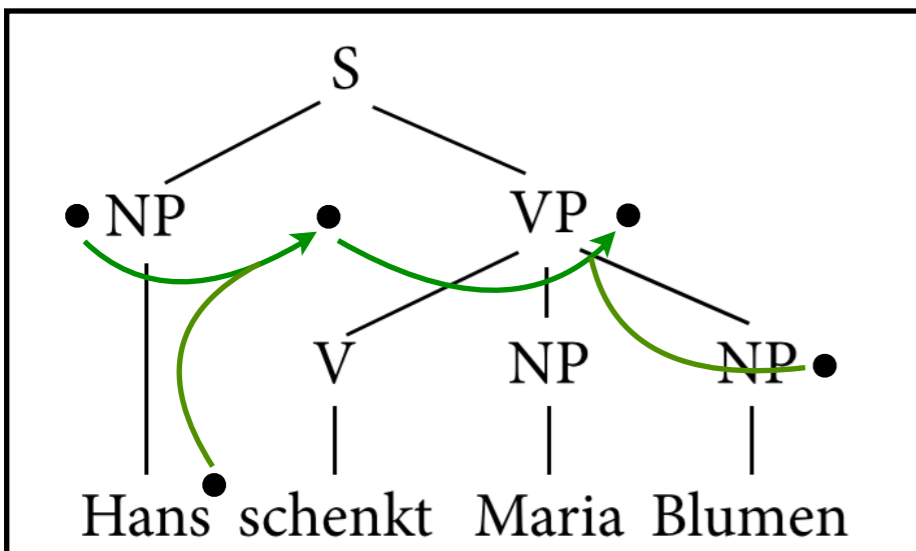
# Operationen



$$\frac{[i, A \rightarrow \alpha \cdot B \beta, k] \quad B \rightarrow \gamma \text{ ist Regel}}{[k, B \rightarrow \cdot \gamma, k]} \quad \text{Predict}$$

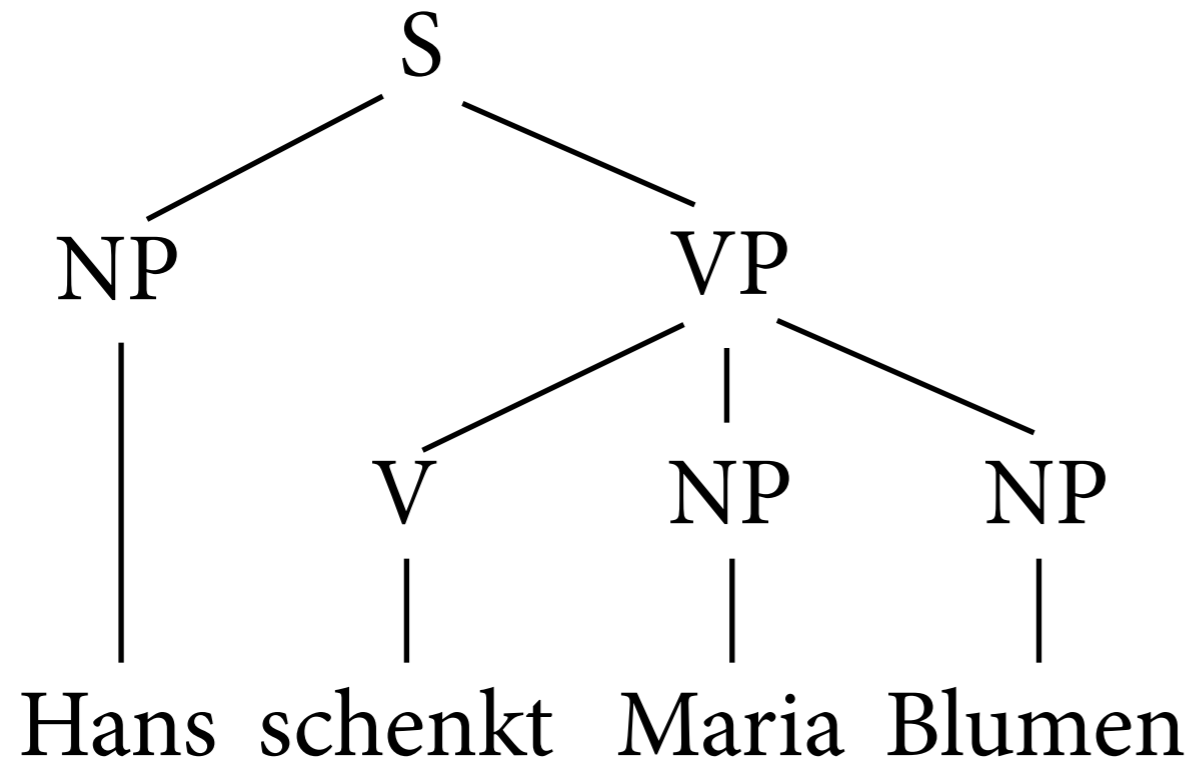


$$\frac{[j, A \rightarrow \alpha \cdot w_i \beta, i]}{[j, A \rightarrow \alpha w_i \cdot \beta, i+1]} \quad \text{Scan}$$



$$\frac{[i, A \rightarrow \alpha \cdot B \beta, j] \quad [j, B \rightarrow \gamma \cdot, k]}{[i, A \rightarrow \alpha B \cdot \beta, k]} \quad \text{Complete}$$

# Earley-Parser im Beispiel

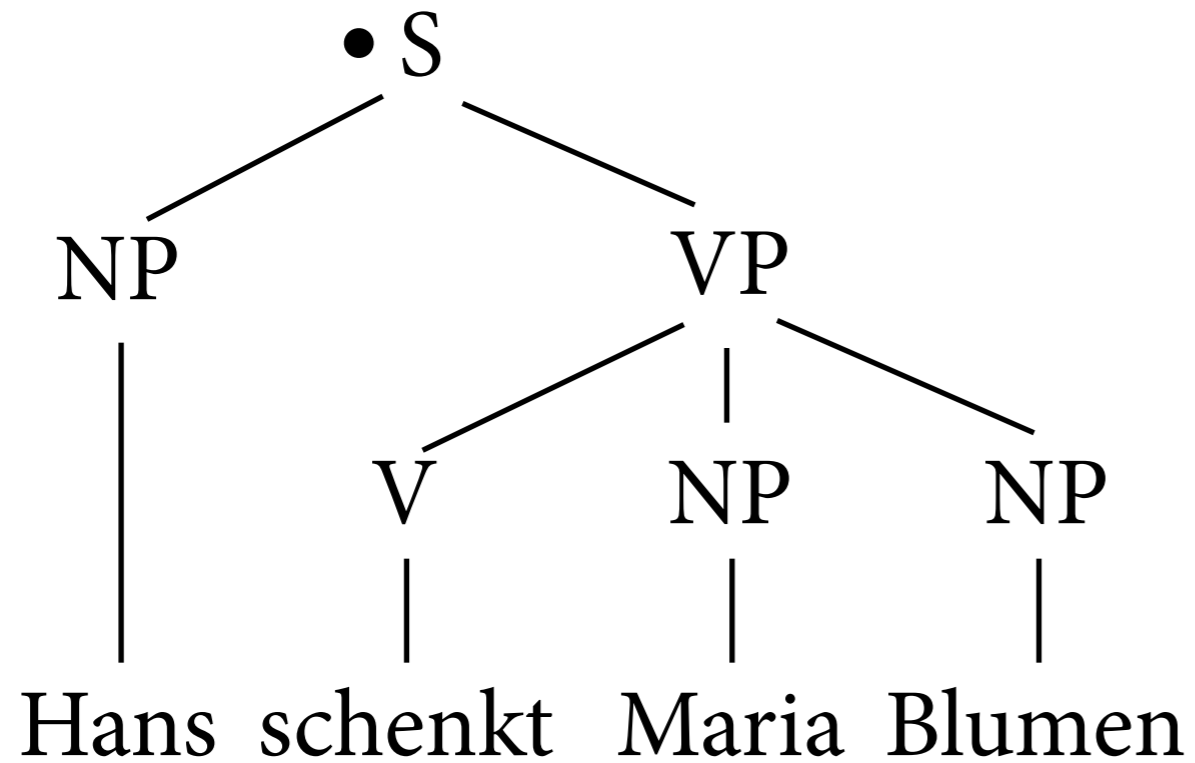


$$\frac{[i, A \rightarrow \alpha \bullet B \beta, k] \quad B \rightarrow \gamma \text{ ist Regel}}{[k, B \rightarrow \bullet \gamma, k]} \quad (\text{P})$$

$$\frac{[j, A \rightarrow \alpha \bullet w_i \beta, i]}{[j, A \rightarrow \alpha w_i \bullet \beta, i+1]} \quad (\text{S})$$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, j] \quad [j, B \rightarrow \gamma \bullet, k]}{[i, A \rightarrow \alpha B \bullet \beta, k]} \quad (\text{C})$$

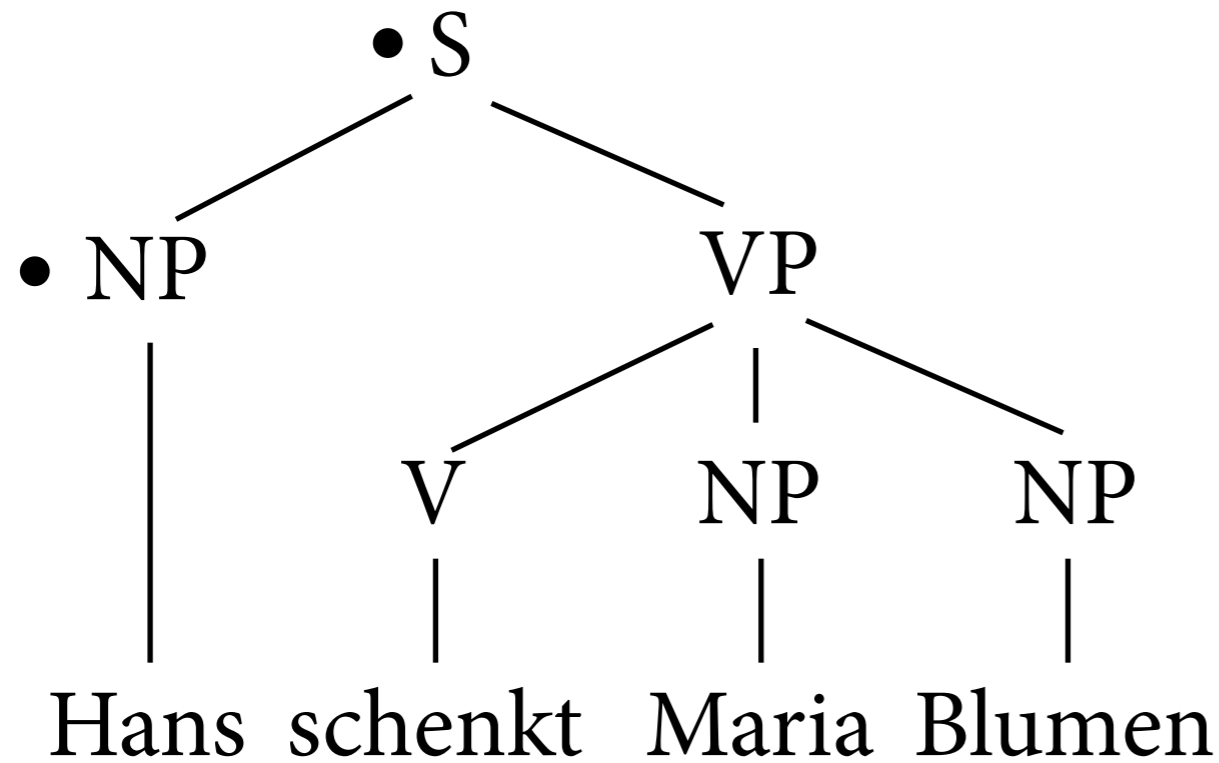
# Earley-Parser im Beispiel



$[1, S' \rightarrow \bullet S, 1]$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, k] \quad B \rightarrow \gamma \text{ ist Regel}}{[k, B \rightarrow \bullet \gamma, k]} \quad (\text{P}) \quad \frac{[j, A \rightarrow \alpha \bullet w_i \beta, i]}{[j, A \rightarrow \alpha w_i \bullet \beta, i+1]} \quad (\text{S}) \quad \frac{[i, A \rightarrow \alpha \bullet B \beta, j] \quad [j, B \rightarrow \gamma \bullet, k]}{[i, A \rightarrow \alpha B \bullet \beta, k]} \quad (\text{C})$$

# Earley-Parser im Beispiel



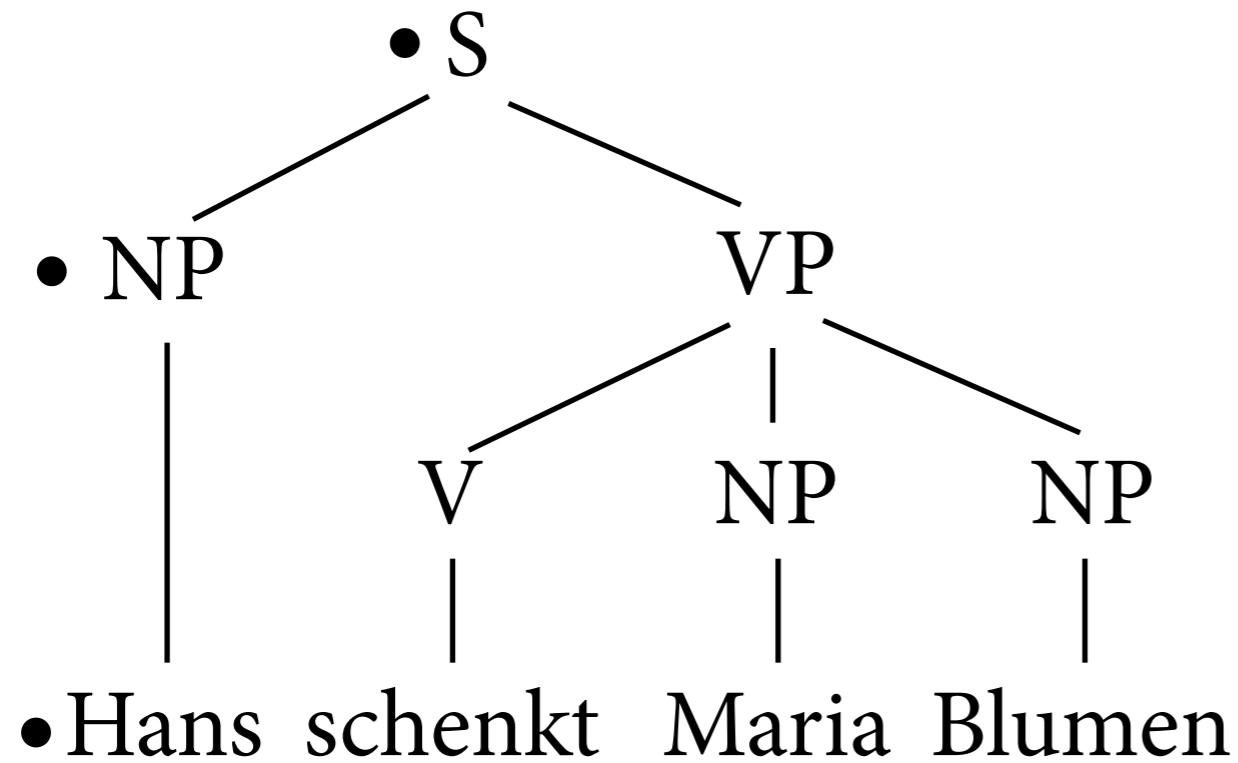
[1, S' → • S, 1]  
 [1, S → • NP VP, 1]

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, k] \quad B \rightarrow \gamma \text{ ist Regel}}{[k, B \rightarrow \bullet \gamma, k]} \quad (\text{P})$$

$$\frac{[j, A \rightarrow \alpha \bullet w_i \beta, i]}{[j, A \rightarrow \alpha w_i \bullet \beta, i+1]} \quad (\text{S})$$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, j] \quad [j, B \rightarrow \gamma \bullet, k]}{[i, A \rightarrow \alpha B \bullet \beta, k]} \quad (\text{C})$$

# Earley-Parser im Beispiel



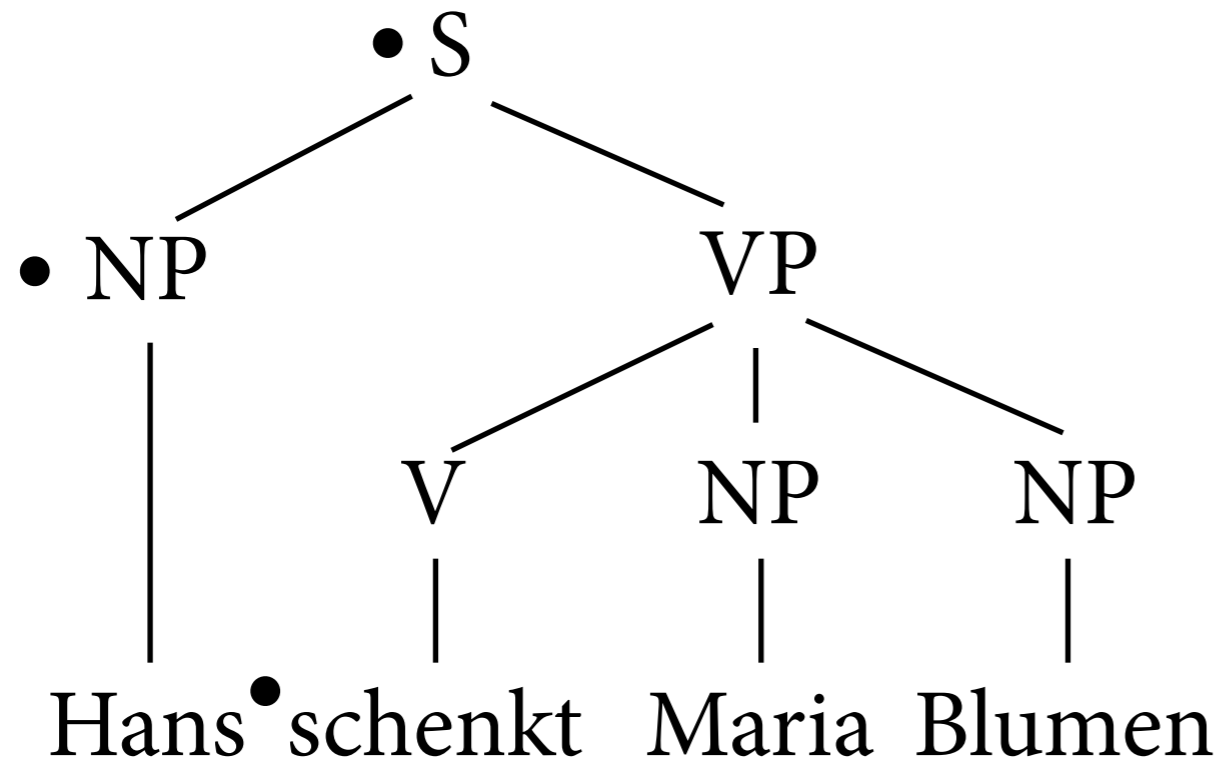
$[1, S' \rightarrow \bullet S, 1]$   
 $[1, S \rightarrow \bullet NP VP, 1]$   
 $[1, NP \rightarrow \bullet Hans, 1]$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, k] \quad B \rightarrow \gamma \text{ ist Regel}}{[k, B \rightarrow \bullet \gamma, k]} \quad (P)$$

$$\frac{[j, A \rightarrow \alpha \bullet w_i \beta, i]}{[j, A \rightarrow \alpha w_i \bullet \beta, i+1]} \quad (S)$$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, j] \quad [j, B \rightarrow \gamma \bullet, k]}{[i, A \rightarrow \alpha B \bullet \beta, k]} \quad (C)$$

# Earley-Parser im Beispiel



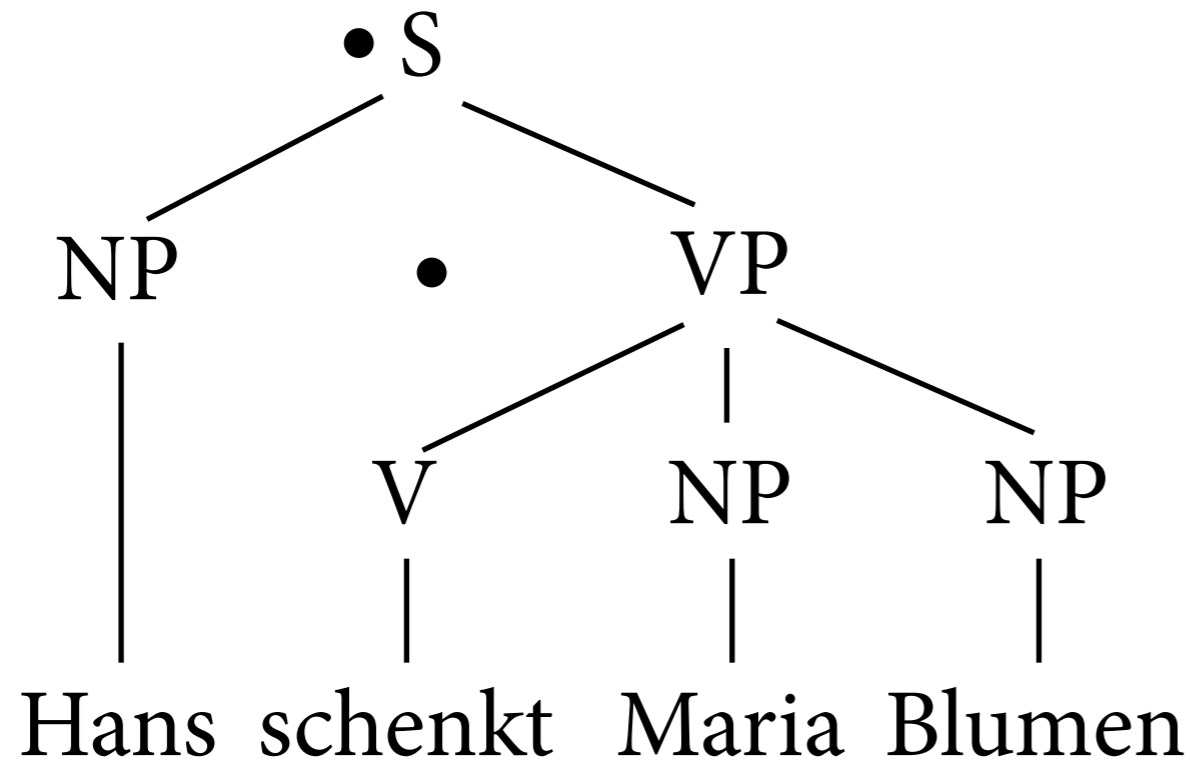
$[1, S' \rightarrow \bullet S, 1]$   
 $[1, S \rightarrow \bullet NP VP, 1]$   
 $[1, NP \rightarrow \bullet Hans, 1]$   
 $[1, NP \rightarrow Hans \bullet, 2]$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, k] \quad B \rightarrow \gamma \text{ ist Regel}}{[k, B \rightarrow \bullet \gamma, k]} \quad (\text{P})$$

$$\frac{[j, A \rightarrow \alpha \bullet w_i \beta, i]}{[j, A \rightarrow \alpha w_i \bullet \beta, i+1]} \quad (\text{S})$$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, j] \quad [j, B \rightarrow \gamma \bullet, k]}{[i, A \rightarrow \alpha B \bullet \beta, k]} \quad (\text{C})$$

# Earley-Parser im Beispiel



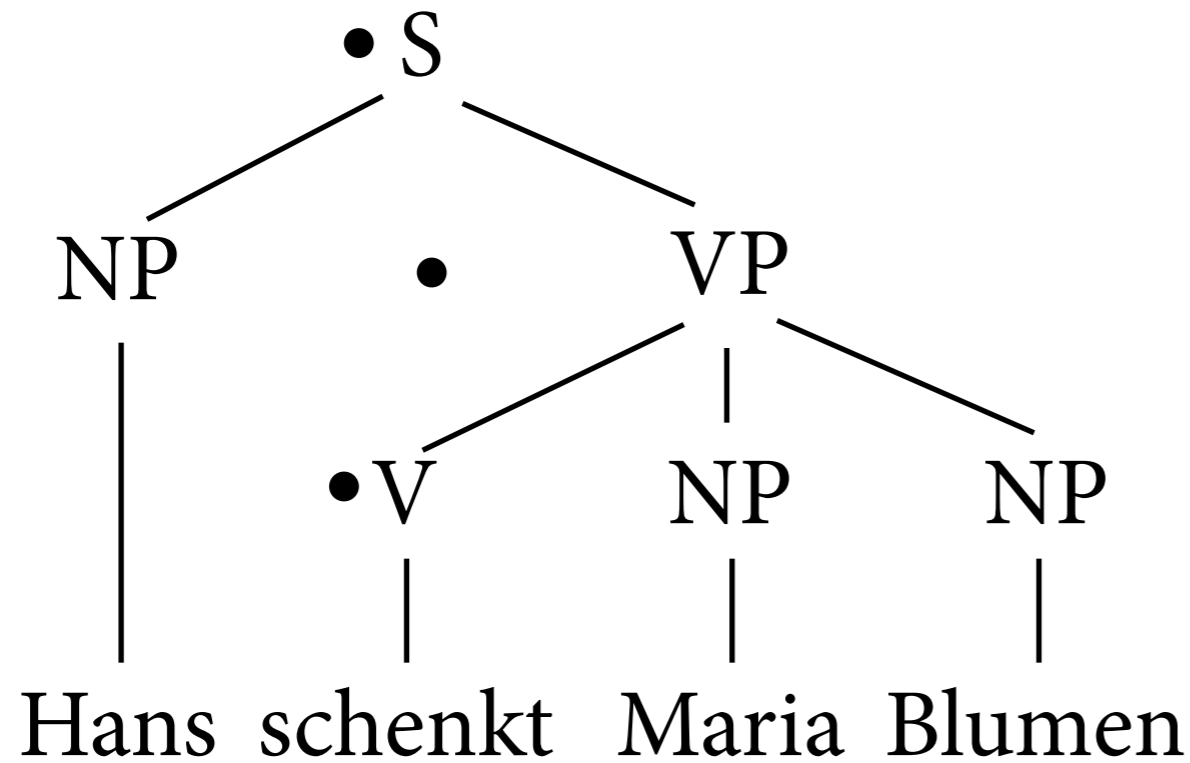
$[1, S' \rightarrow \bullet S, 1]$   
 $[1, S \rightarrow \bullet NP VP, 1]$   
 $[1, NP \rightarrow \bullet Hans, 1]$   
 $[1, NP \rightarrow Hans \bullet, 2]$   
 $[1, S \rightarrow NP \bullet VP, 2]$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, k] \quad B \rightarrow \gamma \text{ ist Regel}}{[k, B \rightarrow \bullet \gamma, k]} \quad (\text{P})$$

$$\frac{[j, A \rightarrow \alpha \bullet w_i \beta, i]}{[j, A \rightarrow \alpha w_i \bullet \beta, i+1]} \quad (\text{S})$$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, j] \quad [j, B \rightarrow \gamma \bullet, k]}{[i, A \rightarrow \alpha B \bullet \beta, k]} \quad (\text{C})$$

# Earley-Parser im Beispiel



$[1, S' \rightarrow \bullet S, 1]$   
 $[1, S \rightarrow \bullet NP VP, 1]$   
 $[1, NP \rightarrow \bullet Hans, 1]$   
 $[1, NP \rightarrow Hans \bullet, 2]$   
 $[1, S \rightarrow NP \bullet VP, 2]$   
 $[2, VP \rightarrow \bullet V NP NP, 2]$

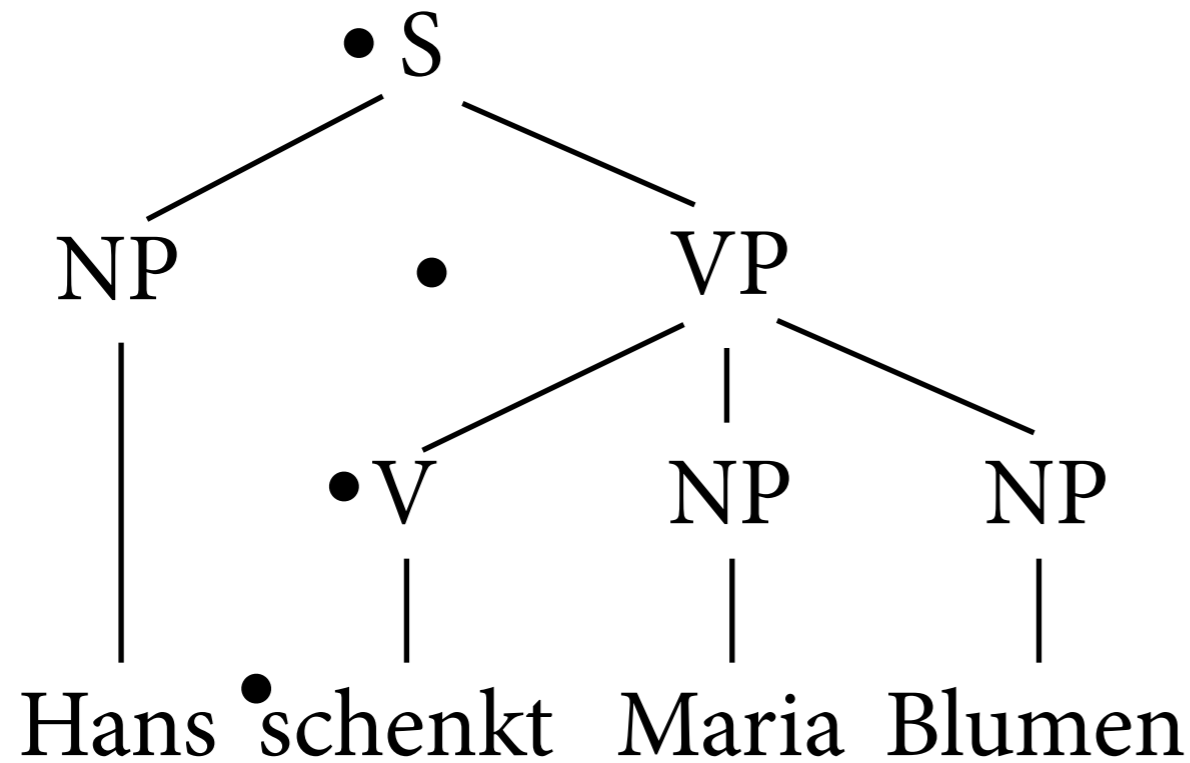
$$\frac{[i, A \rightarrow \alpha \bullet B \beta, k] \quad B \rightarrow \gamma \text{ ist Regel}}{[k, B \rightarrow \bullet \gamma, k]} \quad (P)$$

$$\frac{[j, A \rightarrow \alpha \bullet w_i \beta, i]}{[j, A \rightarrow \alpha w_i \bullet \beta, i+1]} \quad (S)$$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, j] \quad [j, B \rightarrow \gamma \bullet, k]}{[i, A \rightarrow \alpha B \bullet \beta, k]} \quad (C)$$



# Earley-Parser im Beispiel



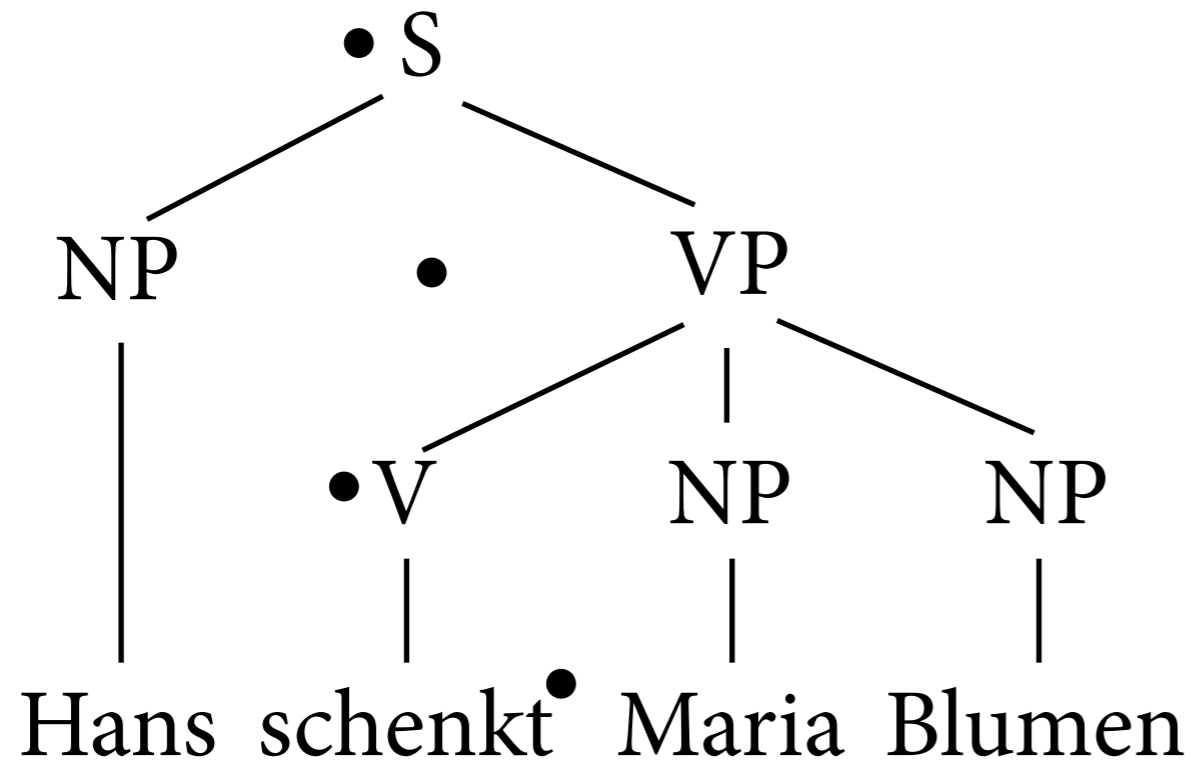
$[1, S' \rightarrow \bullet S, 1]$   
 $[1, S \rightarrow \bullet NP VP, 1]$   
 $[1, NP \rightarrow \bullet Hans, 1]$   
 $[1, NP \rightarrow Hans \bullet, 2]$   
 $[1, S \rightarrow NP \bullet VP, 2]$   
 $[2, VP \rightarrow \bullet V NP NP, 2]$   
 $[2, V \rightarrow \bullet schenkt, 2]$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, k] \quad B \rightarrow \gamma \text{ ist Regel}}{[k, B \rightarrow \bullet \gamma, k]} \quad (P)$$

$$\frac{[j, A \rightarrow \alpha \bullet w_i \beta, i]}{[j, A \rightarrow \alpha w_i \bullet \beta, i+1]} \quad (S)$$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, j] \quad [j, B \rightarrow \gamma \bullet, k]}{[i, A \rightarrow \alpha B \bullet \beta, k]} \quad (C)$$

# Earley-Parser im Beispiel



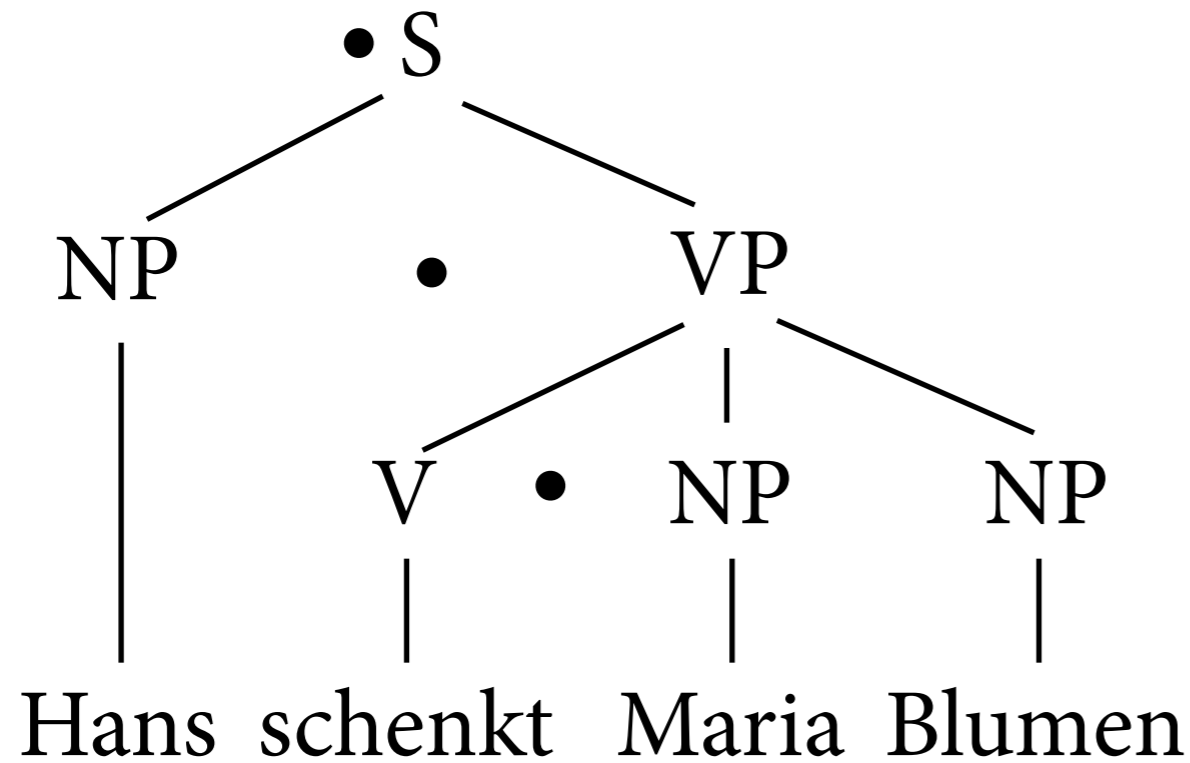
[1,  $S' \rightarrow \bullet S$ , 1]  
 [1,  $S \rightarrow \bullet NP VP$ , 1]  
 [1,  $NP \rightarrow \bullet Hans$ , 1]  
 [1,  $NP \rightarrow Hans \bullet$ , 2]  
 [1,  $S \rightarrow NP \bullet VP$ , 2]  
 [2,  $VP \rightarrow \bullet V NP NP$ , 2]  
 [2,  $V \rightarrow \bullet schenkt$ , 2]  
 [2,  $V \rightarrow schenkt \bullet$ , 3]

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, k] \quad B \rightarrow \gamma \text{ ist Regel}}{[k, B \rightarrow \bullet \gamma, k]} \quad (P)$$

$$\frac{[j, A \rightarrow \alpha \bullet w_i \beta, i]}{[j, A \rightarrow \alpha w_i \bullet \beta, i+1]} \quad (S)$$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, j] \quad [j, B \rightarrow \gamma \bullet, k]}{[i, A \rightarrow \alpha B \bullet \beta, k]} \quad (C)$$

# Earley-Parser im Beispiel



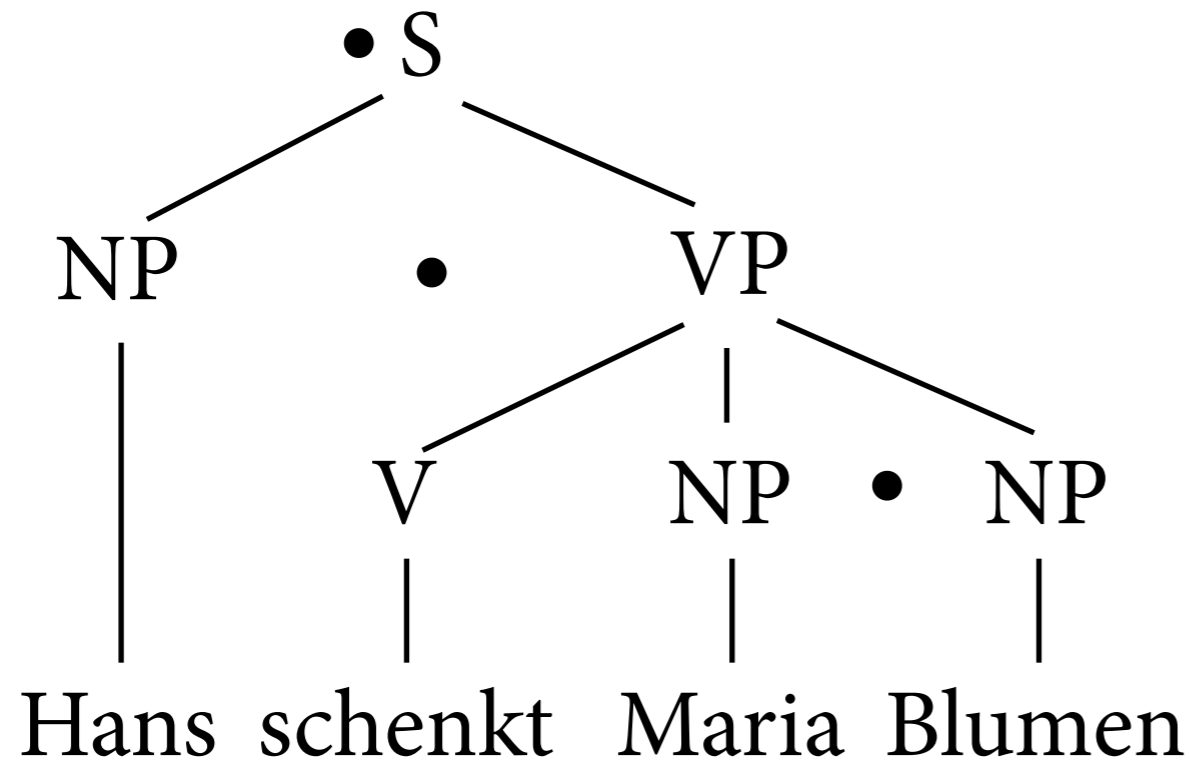
$[1, S' \rightarrow \bullet S, 1]$   
 $[1, S \rightarrow \bullet NP VP, 1]$   
 $[1, NP \rightarrow \bullet Hans, 1]$   
 $[1, NP \rightarrow Hans \bullet, 2]$   
 $[1, S \rightarrow NP \bullet VP, 2]$   
 $[2, VP \rightarrow \bullet V NP NP, 2]$   
 $[2, V \rightarrow \bullet schenkt, 2]$   
 $[2, V \rightarrow schenkt \bullet, 3]$   
 $[2, VP \rightarrow V \bullet NP NP, 3]$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, k] \quad B \rightarrow \gamma \text{ ist Regel}}{[k, B \rightarrow \bullet \gamma, k]} \quad (P)$$

$$\frac{[j, A \rightarrow \alpha \bullet w_i \beta, i]}{[j, A \rightarrow \alpha w_i \bullet \beta, i+1]} \quad (S)$$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, j] \quad [j, B \rightarrow \gamma \bullet, k]}{[i, A \rightarrow \alpha B \bullet \beta, k]} \quad (C)$$

# Earley-Parser im Beispiel



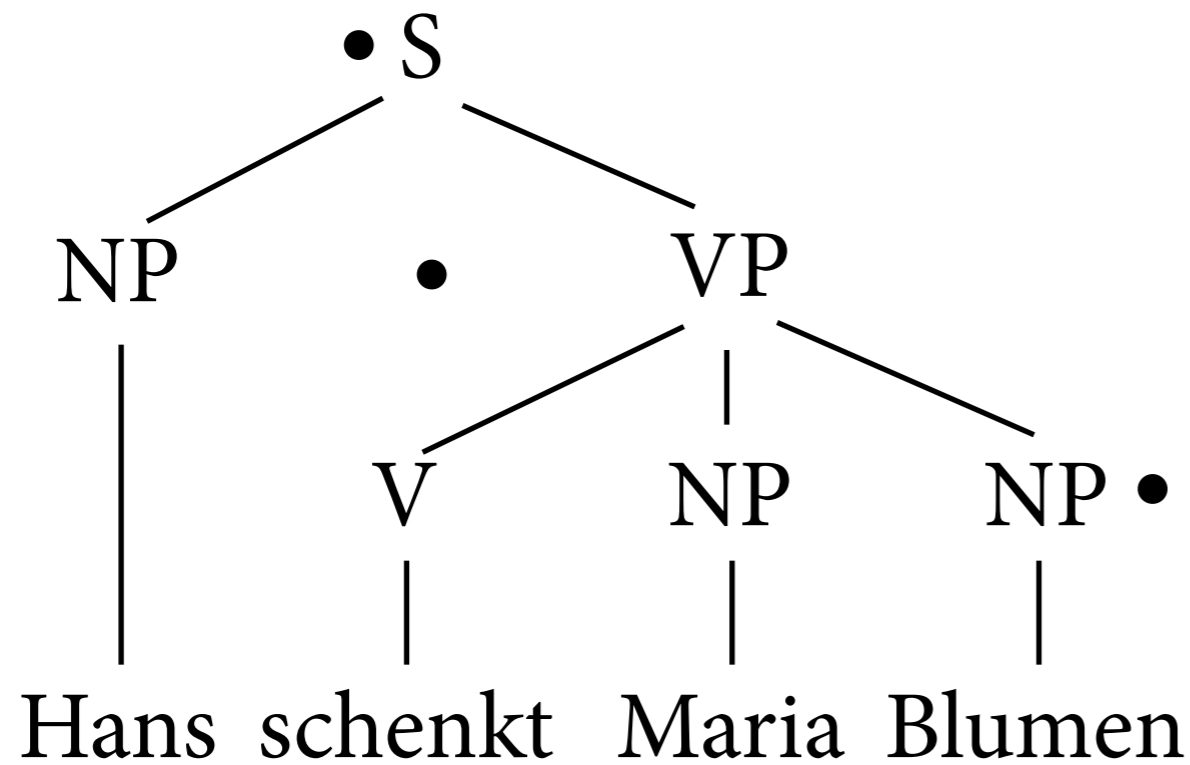
[1,  $S' \rightarrow \bullet S$ , 1]  
 [1,  $S \rightarrow \bullet NP VP$ , 1]  
 [1,  $NP \rightarrow \bullet Hans$ , 1]  
 [1,  $NP \rightarrow Hans \bullet$ , 2]  
 [1,  $S \rightarrow NP \bullet VP$ , 2]  
 [2,  $VP \rightarrow \bullet V NP NP$ , 2]  
 [2,  $V \rightarrow \bullet schenkt$ , 2]  
 [2,  $V \rightarrow schenkt \bullet$ , 3]  
 [2,  $VP \rightarrow V \bullet NP NP$ , 3]  
 ... [2,  $VP \rightarrow V NP \bullet NP$ , 4]

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, k] \quad B \rightarrow \gamma \text{ ist Regel}}{[k, B \rightarrow \bullet \gamma, k]} \quad (P)$$

$$\frac{[j, A \rightarrow \alpha \bullet w_i \beta, i]}{[j, A \rightarrow \alpha w_i \bullet \beta, i+1]} \quad (S)$$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, j] \quad [j, B \rightarrow \gamma \bullet, k]}{[i, A \rightarrow \alpha B \bullet \beta, k]} \quad (C)$$

# Earley-Parser im Beispiel



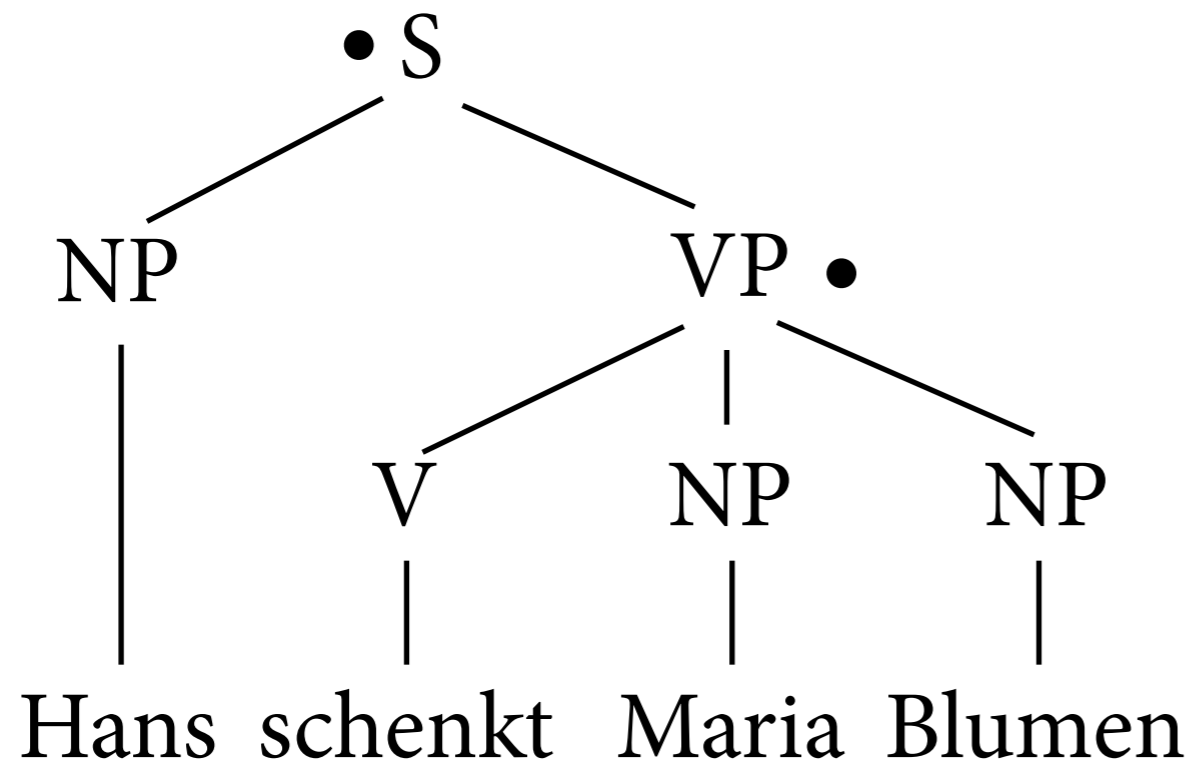
$[1, S' \rightarrow \bullet S, 1]$   
 $[1, S \rightarrow \bullet NP VP, 1]$   
 $[1, NP \rightarrow \bullet Hans, 1]$   
 $[1, NP \rightarrow Hans \bullet, 2]$   
 $[1, S \rightarrow NP \bullet VP, 2]$   
 $[2, VP \rightarrow \bullet V NP NP, 2]$   
 $[2, V \rightarrow \bullet schenkt, 2]$   
 $[2, V \rightarrow schenkt \bullet, 3]$   
 $[2, VP \rightarrow V \bullet NP NP, 3]$   
 $\dots [2, VP \rightarrow V NP \bullet NP, 4]$   
 $\dots [2, VP \rightarrow V NP NP \bullet, 5]$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, k] \quad B \rightarrow \gamma \text{ ist Regel}}{[k, B \rightarrow \bullet \gamma, k]} \quad (P)$$

$$\frac{[j, A \rightarrow \alpha \bullet w_i \beta, i]}{[j, A \rightarrow \alpha w_i \bullet \beta, i+1]} \quad (S)$$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, j] \quad [j, B \rightarrow \gamma \bullet, k]}{[i, A \rightarrow \alpha B \bullet \beta, k]} \quad (C)$$

# Earley-Parser im Beispiel



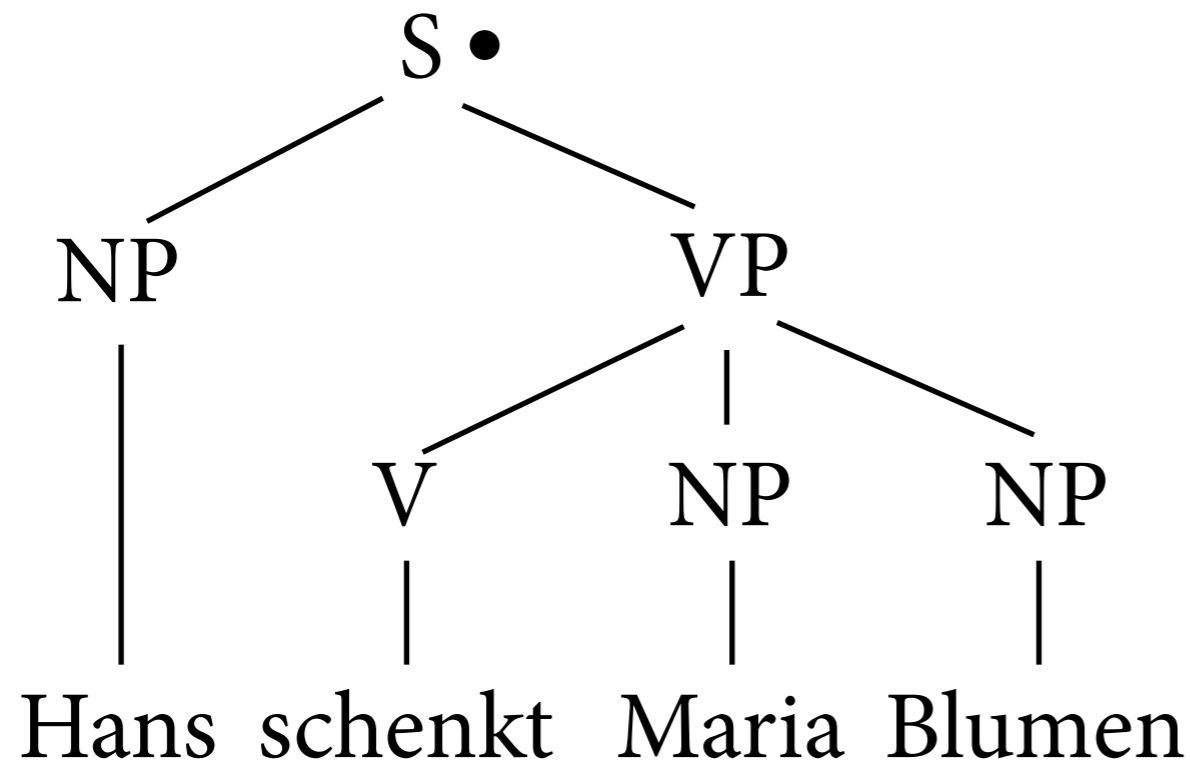
$[1, S' \rightarrow \bullet S, 1]$   
 $[1, S \rightarrow \bullet NP VP, 1]$   
 $[1, NP \rightarrow \bullet Hans, 1]$   
 $[1, NP \rightarrow Hans \bullet, 2]$   
 $[1, S \rightarrow NP \bullet VP, 2]$   
 $[2, VP \rightarrow \bullet V NP NP, 2]$   
 $[2, V \rightarrow \bullet schenkt, 2]$   
 $[2, V \rightarrow schenkt \bullet, 3]$   
 $[2, VP \rightarrow V \bullet NP NP, 3]$   
 $\dots [2, VP \rightarrow V NP \bullet NP, 4]$   
 $\dots [2, VP \rightarrow V NP NP \bullet, 5]$   
 $\dots [1, S \rightarrow NP VP \bullet, 5]$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, k] \quad B \rightarrow \gamma \text{ ist Regel}}{[k, B \rightarrow \bullet \gamma, k]} \quad (P)$$

$$\frac{[j, A \rightarrow \alpha \bullet w_i \beta, i]}{[j, A \rightarrow \alpha w_i \bullet \beta, i+1]} \quad (S)$$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, j] \quad [j, B \rightarrow \gamma \bullet, k]}{[i, A \rightarrow \alpha B \bullet \beta, k]} \quad (C)$$

# Earley-Parser im Beispiel



[1,  $S' \rightarrow \bullet S$ , 1]  
 [1,  $S \rightarrow \bullet NP VP$ , 1]  
 [1,  $NP \rightarrow \bullet Hans$ , 1]  
 [1,  $NP \rightarrow Hans \bullet$ , 2]  
 [1,  $S \rightarrow NP \bullet VP$ , 2]  
 [2,  $VP \rightarrow \bullet V NP NP$ , 2]  
 [2,  $V \rightarrow \bullet schenkt$ , 2]  
 [2,  $V \rightarrow schenkt \bullet$ , 3]  
 [2,  $VP \rightarrow V \bullet NP NP$ , 3]  
 ... [2,  $VP \rightarrow V NP \bullet NP$ , 4]  
 ... [2,  $VP \rightarrow V NP NP \bullet$ , 5]  
 ... [1,  $S \rightarrow NP VP \bullet$ , 5]  
 ... [1,  $S' \rightarrow S \bullet$ , 5]

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, k] \quad B \rightarrow \gamma \text{ ist Regel}}{[k, B \rightarrow \bullet \gamma, k]} \quad (P)$$

$$\frac{[j, A \rightarrow \alpha \bullet w_i \beta, i]}{[j, A \rightarrow \alpha w_i \bullet \beta, i+1]} \quad (S)$$

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, j] \quad [j, B \rightarrow \gamma \bullet, k]}{[i, A \rightarrow \alpha B \bullet \beta, k]} \quad (C)$$

# Der Earley-Algorithmus

- Beginne mit Chart, die nur das Item  $[1, S' \rightarrow \bullet S, 1]$  enthält ( $S$  ist Startsymbol;  $S'$  ist frisches “Super-Startsymbol”).
- Wende Predict, Scan, Complete an, so lange es geht.
- Wenn dabei  $[1, S' \rightarrow S \bullet, n+1]$  abgeleitet wurde, war Wort in der Sprache; sonst nicht.

Alternative, äquivalente Formulierung: Start-Items sind Items der Form  $[1, S \rightarrow \bullet \gamma, 1]$  für jede Regel  $S \rightarrow \gamma$ ; Ziel-Items sind alle  $[1, S \rightarrow \gamma \bullet, n+1]$ .



# Der Earley-Parser

$S \rightarrow NP VP$      $V \rightarrow \text{isst}$      $\text{Det} \rightarrow \text{ein}$   
 $NP \rightarrow \text{Det } N$      $NP \rightarrow \text{Hans}$      $N \rightarrow \text{Käse}$   
 $VP \rightarrow V NP$      $N \rightarrow \text{brot}$

Punkt-  
position

Predict  
Scan  
Complete

Hans isst ein K.brot

Hans	isst	ein	Käse <b>br</b> ot

Anfangs-  
position

# Der Earley-Parser

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Punkt-  
position

Predict  
Scan  
Complete

Hans isst ein K.brot

$1, S' \rightarrow \bullet S, 1$			
Hans	isst	ein	Käsebro

Anfangs-  
position

# Der Earley-Parser

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Punkt-  
position

Predict  
Scan  
Complete

Hans isst ein K.brot

$1, S' \rightarrow \bullet S, 1$ $1, S \rightarrow \bullet NP VP, 1$			
Hans	isst	ein	Käsebro

Anfangs-  
position

# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Predict  
Scan  
Complete

Hans isst ein K.brot

$1, S' \rightarrow \bullet S, 1$ $1, S \rightarrow \bullet NP VP, 1$ $1, NP \rightarrow \bullet Hans, 1$			
Hans	isst	ein	Käsebro

Anfangs-  
position

# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Predict  
Scan  
Complete

Hans isst ein K.brot

1, $NP \rightarrow Hans \bullet$ , 2			
1, $S' \rightarrow \bullet S$ , 1 1, $S \rightarrow \bullet NP VP$ , 1 1, $NP \rightarrow \bullet Hans$ , 1			
Hans	isst	ein	Käsebro

Anfangs-  
position

# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Predict  
Scan  
Complete

Hans isst ein K.brot

<p>1, NP <math>\rightarrow</math> Hans •, 2 1, S <math>\rightarrow</math> NP • VP, 2</p>			
<p>1, S' <math>\rightarrow</math> • S, 1 1, S <math>\rightarrow</math> • NP VP, 1 1, NP <math>\rightarrow</math> • Hans, 1</p>			
Hans	isst	ein	Käsebro

Anfangs-  
position

# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Predict  
Scan  
Complete

Hans isst ein K.brot

<p>1, NP <math>\rightarrow</math> Hans •, 2 1, S <math>\rightarrow</math> NP • VP, 2</p>	<p>2, VP <math>\rightarrow</math> • V NP, 2</p>		
<p>1, S' <math>\rightarrow</math> • S, 1 1, S <math>\rightarrow</math> • NP VP, 1 1, NP <math>\rightarrow</math> • Hans, 1</p>			
Hans	isst	ein	Käsebro

Anfangs-  
position

# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Predict  
Scan  
Complete

Hans isst ein K.brot

<p>1, NP <math>\rightarrow</math> Hans •, 2 1, S <math>\rightarrow</math> NP • VP, 2</p>	<p>2, VP <math>\rightarrow</math> • V NP, 2 2, V <math>\rightarrow</math> • isst, 2</p>		
<p>1, S' <math>\rightarrow</math> • S, 1 1, S <math>\rightarrow</math> • NP VP, 1 1, NP <math>\rightarrow</math> • Hans, 1</p>			
Hans	isst	ein	Käsebro

Anfangs-  
position



# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Predict  
Scan  
Complete

Hans isst ein K.brot

	2, $V \rightarrow isst \bullet$ , 3		
1, $NP \rightarrow Hans \bullet$ , 2 1, $S \rightarrow NP \bullet VP$ , 2	2, $VP \rightarrow \bullet V NP$ , 2 2, $V \rightarrow \bullet isst$ , 2		
1, $S' \rightarrow \bullet S$ , 1 1, $S \rightarrow \bullet NP VP$ , 1 1, $NP \rightarrow \bullet Hans$ , 1			
Hans	isst	ein	Käsebro

Anfangs-  
position

# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Predict  
Scan  
Complete

Hans isst ein K.brot

	<p>2, <math>V \rightarrow isst \bullet</math>, 3                  2, <math>VP \rightarrow V \bullet NP</math>, 3</p>		
<p>1, <math>NP \rightarrow Hans \bullet</math>, 2                  1, <math>S \rightarrow NP \bullet VP</math>, 2</p>	<p>2, <math>VP \rightarrow \bullet V NP</math>, 2                  2, <math>V \rightarrow \bullet isst</math>, 2</p>		
<p>1, <math>S' \rightarrow \bullet S</math>, 1                  1, <math>S \rightarrow \bullet NP VP</math>, 1                  1, <math>NP \rightarrow \bullet Hans</math>, 1</p>			
Hans	isst	ein	Käsebro

Anfangs-  
position

# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Predict  
Scan  
Complete

Hans isst ein K.brot

	<p>2, <math>V \rightarrow isst \bullet</math>, 3</p> <p>2, <math>VP \rightarrow V \bullet NP</math>, 3</p>	<p>3, <math>NP \rightarrow \bullet Det N</math>, 3</p>	
	<p>1, <math>NP \rightarrow Hans \bullet</math>, 2</p> <p>1, <math>S \rightarrow NP \bullet VP</math>, 2</p>	<p>2, <math>VP \rightarrow \bullet V NP</math>, 2</p> <p>2, <math>V \rightarrow \bullet isst</math>, 2</p>	
	<p>1, <math>S' \rightarrow \bullet S</math>, 1</p> <p>1, <math>S \rightarrow \bullet NP VP</math>, 1</p> <p>1, <math>NP \rightarrow \bullet Hans</math>, 1</p>		
	Hans	isst	ein Käsebro

Anfangs-  
position

# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow \text{isst}$      $\text{Det} \rightarrow \text{ein}$   
 $NP \rightarrow \text{Det } N$      $NP \rightarrow \text{Hans}$      $N \rightarrow \text{Käse}$   
 $VP \rightarrow V NP$      $N \rightarrow \text{brot}$

Predict  
Scan  
Complete

Hans isst ein K.brot

	<p>2, <math>V \rightarrow \text{isst} \bullet</math>, 3</p> <p>2, <math>VP \rightarrow V \bullet NP</math>, 3</p>	<p>3, <math>NP \rightarrow \bullet \text{Det } N</math>, 3</p> <p>3, <math>\text{Det} \rightarrow \bullet \text{ein}</math>, 3</p>	
	<p>1, <math>NP \rightarrow \text{Hans} \bullet</math>, 2</p> <p>1, <math>S \rightarrow NP \bullet VP</math>, 2</p>	<p>2, <math>VP \rightarrow \bullet V NP</math>, 2</p> <p>2, <math>V \rightarrow \bullet \text{isst}</math>, 2</p>	
	<p>1, <math>S' \rightarrow \bullet S</math>, 1</p> <p>1, <math>S \rightarrow \bullet NP VP</math>, 1</p> <p>1, <math>NP \rightarrow \bullet \text{Hans}</math>, 1</p>		
	Hans	isst	ein Käse <b>bro</b> t

Anfangs-  
position

# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Predict  
Scan  
Complete

Hans isst ein K.brot

		3, $Det \rightarrow ein \bullet, 4$	
	2, $V \rightarrow isst \bullet, 3$ 2, $VP \rightarrow V \bullet NP, 3$	3, $NP \rightarrow \bullet Det N, 3$ 3, $Det \rightarrow \bullet ein, 3$	
1, $NP \rightarrow Hans \bullet, 2$ 1, $S \rightarrow NP \bullet VP, 2$	2, $VP \rightarrow \bullet V NP, 2$ 2, $V \rightarrow \bullet isst, 2$		
1, $S' \rightarrow \bullet S, 1$ 1, $S \rightarrow \bullet NP VP, 1$ 1, $NP \rightarrow \bullet Hans, 1$			
Hans	isst	ein	Käsebro

Anfangs-  
position

# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Predict  
Scan  
Complete

Hans isst ein K.brot

		<p>3, <math>Det \rightarrow ein \bullet</math>, 4</p> <p>3, <math>NP \rightarrow Det \bullet N</math>, 4</p>	
	<p>2, <math>V \rightarrow isst \bullet</math>, 3</p> <p>2, <math>VP \rightarrow V \bullet NP</math>, 3</p>	<p>3, <math>NP \rightarrow \bullet Det N</math>, 3</p> <p>3, <math>Det \rightarrow \bullet ein</math>, 3</p>	
Hans	<p>1, <math>NP \rightarrow Hans \bullet</math>, 2</p> <p>1, <math>S \rightarrow NP \bullet VP</math>, 2</p>	<p>2, <math>VP \rightarrow \bullet V NP</math>, 2</p> <p>2, <math>V \rightarrow \bullet isst</math>, 2</p>	
	<p>1, <math>S' \rightarrow \bullet S</math>, 1</p> <p>1, <math>S \rightarrow \bullet NP VP</math>, 1</p> <p>1, <math>NP \rightarrow \bullet Hans</math>, 1</p>		
Hans	isst	ein	Käsebro

Anfangs-  
position

# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Predict  
Scan  
Complete

Hans isst ein K.brot

		<p>3, <math>Det \rightarrow ein \bullet</math>, 4</p> <p>3, <math>NP \rightarrow Det \bullet N</math>, 4</p>	<p>4, <math>N \rightarrow \bullet kb</math>, 4</p>
	<p>2, <math>V \rightarrow isst \bullet</math>, 3</p> <p>2, <math>VP \rightarrow V \bullet NP</math>, 3</p>	<p>3, <math>NP \rightarrow \bullet Det N</math>, 3</p> <p>3, <math>Det \rightarrow \bullet ein</math>, 3</p>	
	<p>1, <math>NP \rightarrow Hans \bullet</math>, 2</p> <p>1, <math>S \rightarrow NP \bullet VP</math>, 2</p>	<p>2, <math>VP \rightarrow \bullet V NP</math>, 2</p> <p>2, <math>V \rightarrow \bullet isst</math>, 2</p>	
	<p>1, <math>S' \rightarrow \bullet S</math>, 1</p> <p>1, <math>S \rightarrow \bullet NP VP</math>, 1</p> <p>1, <math>NP \rightarrow \bullet Hans</math>, 1</p>		
Hans	isst	ein	Käsebro

Anfangs-  
position

# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Predict  
Scan  
Complete

Hans isst ein K.brot

			4, $N \rightarrow kb \bullet, 5$
		3, $Det \rightarrow ein \bullet, 4$ 3, $NP \rightarrow Det \bullet N, 4$	4, $N \rightarrow \bullet kb, 4$
	2, $V \rightarrow isst \bullet, 3$ 2, $VP \rightarrow V \bullet NP, 3$	3, $NP \rightarrow \bullet Det N, 3$ 3, $Det \rightarrow \bullet ein, 3$	
1, $NP \rightarrow Hans \bullet, 2$ 1, $S \rightarrow NP \bullet VP, 2$	2, $VP \rightarrow \bullet V NP, 2$ 2, $V \rightarrow \bullet isst, 2$		
1, $S' \rightarrow \bullet S, 1$ 1, $S \rightarrow \bullet NP VP, 1$ 1, $NP \rightarrow \bullet Hans, 1$			
Hans	isst	ein	Käsebro

Anfangs-  
position



# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Predict  
Scan  
Complete

Hans isst ein K.brot

		3, $NP \rightarrow Det N \bullet$ , 5	4, $N \rightarrow kb \bullet$ , 5
		3, $Det \rightarrow ein \bullet$ , 4 3, $NP \rightarrow Det \bullet N$ , 4	4, $N \rightarrow \bullet kb$ , 4
	2, $V \rightarrow isst \bullet$ , 3 2, $VP \rightarrow V \bullet NP$ , 3	3, $NP \rightarrow \bullet Det N$ , 3 3, $Det \rightarrow \bullet ein$ , 3	
1, $NP \rightarrow Hans \bullet$ , 2 1, $S \rightarrow NP \bullet VP$ , 2	2, $VP \rightarrow \bullet V NP$ , 2 2, $V \rightarrow \bullet isst$ , 2		
1, $S' \rightarrow \bullet S$ , 1 1, $S \rightarrow \bullet NP VP$ , 1 1, $NP \rightarrow \bullet Hans$ , 1			
Hans	isst	ein	Käsebro

Anfangs-  
position

# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Predict  
Scan  
Complete

Hans isst ein K.brot

	2, $VP \rightarrow V NP \bullet$ , 5	3, $NP \rightarrow Det N \bullet$ , 5	4, $N \rightarrow kb \bullet$ , 5
		3, $Det \rightarrow ein \bullet$ , 4 3, $NP \rightarrow Det \bullet N$ , 4	4, $N \rightarrow \bullet kb$ , 4
	2, $V \rightarrow isst \bullet$ , 3 2, $VP \rightarrow V \bullet NP$ , 3	3, $NP \rightarrow \bullet Det N$ , 3 3, $Det \rightarrow \bullet ein$ , 3	
Hans	1, $NP \rightarrow Hans \bullet$ , 2 1, $S \rightarrow NP \bullet VP$ , 2	2, $VP \rightarrow \bullet V NP$ , 2 2, $V \rightarrow \bullet isst$ , 2	
	1, $S' \rightarrow \bullet S$ , 1 1, $S \rightarrow \bullet NP VP$ , 1 1, $NP \rightarrow \bullet Hans$ , 1		
	Hans	isst	ein
			Käsebro

Anfangs-  
position

# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Predict  
Scan  
Complete

Hans isst ein K.brot

1, $S \rightarrow NP VP \bullet$ , 5	2, $VP \rightarrow V NP \bullet$ , 5	3, $NP \rightarrow Det N \bullet$ , 5	4, $N \rightarrow kb \bullet$ , 5
		3, $Det \rightarrow ein \bullet$ , 4 3, $NP \rightarrow Det \bullet N$ , 4	4, $N \rightarrow \bullet kb$ , 4
	2, $V \rightarrow isst \bullet$ , 3 2, $VP \rightarrow V \bullet NP$ , 3	3, $NP \rightarrow \bullet Det N$ , 3 3, $Det \rightarrow \bullet ein$ , 3	
1, $NP \rightarrow Hans \bullet$ , 2 1, $S \rightarrow NP \bullet VP$ , 2	2, $VP \rightarrow \bullet V NP$ , 2 2, $V \rightarrow \bullet isst$ , 2		
1, $S' \rightarrow \bullet S$ , 1 1, $S \rightarrow \bullet NP VP$ , 1 1, $NP \rightarrow \bullet Hans$ , 1			
Hans	isst	ein	Käsebro

Anfangs-  
position

# Der Earley-Parser

Punkt-  
position

$S \rightarrow NP VP$      $V \rightarrow isst$      $Det \rightarrow ein$   
 $NP \rightarrow Det N$      $NP \rightarrow Hans$      $N \rightarrow Käsebro$   
 $VP \rightarrow V NP$

Predict  
Scan  
Complete

Hans isst ein K.brot

1, $S \rightarrow NP VP \bullet$ , 5 1, $S' \rightarrow S \bullet$ , 5	2, $VP \rightarrow V NP \bullet$ , 5	3, $NP \rightarrow Det N \bullet$ , 5	4, $N \rightarrow kb \bullet$ , 5
		3, $Det \rightarrow ein \bullet$ , 4 3, $NP \rightarrow Det \bullet N$ , 4	4, $N \rightarrow \bullet kb$ , 4
	2, $V \rightarrow isst \bullet$ , 3 2, $VP \rightarrow V \bullet NP$ , 3	3, $NP \rightarrow \bullet Det N$ , 3 3, $Det \rightarrow \bullet ein$ , 3	
1, $NP \rightarrow Hans \bullet$ , 2 1, $S \rightarrow NP \bullet VP$ , 2	2, $VP \rightarrow \bullet V NP$ , 2 2, $V \rightarrow \bullet isst$ , 2		
1, $S' \rightarrow \bullet S$ , 1 1, $S \rightarrow \bullet NP VP$ , 1 1, $NP \rightarrow \bullet Hans$ , 1			
Hans	isst	ein	Käsebro

Anfangs-  
position

# Implementierung

- Wir verwalten Items, mit denen wir noch nichts gemacht haben, in einer *Agenda*.
  - ▶ Agenda ist eine Queue von Items: Nimm Items vom Anfang weg, füge neue Items am Ende hinzu.
- Alle Items, die wir jemals gesehen haben, schreiben wir in die Chart.
  - ▶ Eine Menge  $S_i$  von Items für jede Punktposition  $i$ .
  - ▶ Jedes  $S_i =$  eine Zeile auf der vorherigen Folie.

# Implementierung

- So lange es geht: nimm erstes Item  $[i, A \rightarrow \alpha \bullet \beta, k]$  von der Agenda.
  - ▶ Wenn  $\beta$  leer ist (Punkt am Schluss), dann wende Complete mit Items in  $S_i$  an, die auf ein  $A$  warten.
  - ▶ Wenn erstes Zeichen in  $\beta$  Terminalsymbol ist, dann wende Scan an.
  - ▶ Wenn erstes Zeichen in  $\beta$  Nichtterminalsymbol  $B$  ist, dann wende Predict mit allen Regeln für  $B$  an.
  - ▶ Schreibe ein neues Item nur dann auf Agenda, wenn es noch nicht in Chart steht.
- Agenda wird irgendwann leer; dann testen, ob  $[1, S' \rightarrow S \bullet, n+1]$  in  $S_{n+1}$  steht oder nicht.

# Laufzeit

- Implementierung schreibt jedes Item nur einmal auf die Agenda.

- Wie viele Items gibt es? Am flexibelsten Complete:

$$\frac{[i, A \rightarrow \alpha \bullet B \beta, j] \quad [j, B \rightarrow \gamma \bullet, k]}{[i, A \rightarrow \alpha B \bullet \beta, k]} \text{ Complete}$$

- Es gibt  $O(n^3)$  Instanzen von Complete. Daher gesamte Laufzeit nur  $O(n^3)$ .
- Für manche Klassen von kfGen ist Earley sogar  $O(n^2)$  oder  $O(n)$ .

# Top-down und bottom-up

- Earley-Parser verbindet Elemente von top-down und bottom-up Parsern.

	Iteration über Regeln	Iteration über String-Zerlegungen
Recursive Descent	top-down	top-down
CKY	bottom-up	bottom-up
Earley	top-down (Predict)	bottom-up (Scan, Complete)



# Zusammenfassung

- CKY hat Nachteile:
  - ▶ nur für Grammatiken in CNF definiert
  - ▶ nichtbinäre Version hat Laufzeit  $O(n^{r+1})$
- Earley-Parser:
  - ▶ wendet beim Parsing CNF-Konversion an
  - ▶ und kontrolliert unnötige Arbeit durch Top-Down-Prediction
  - ▶ Laufzeit  $O(n^3)$  für beliebige kfGs
  - ▶ Korrektheit: zeigen wir nächste Woche allgemeiner.