



# A nyelvtechnológia alapjai

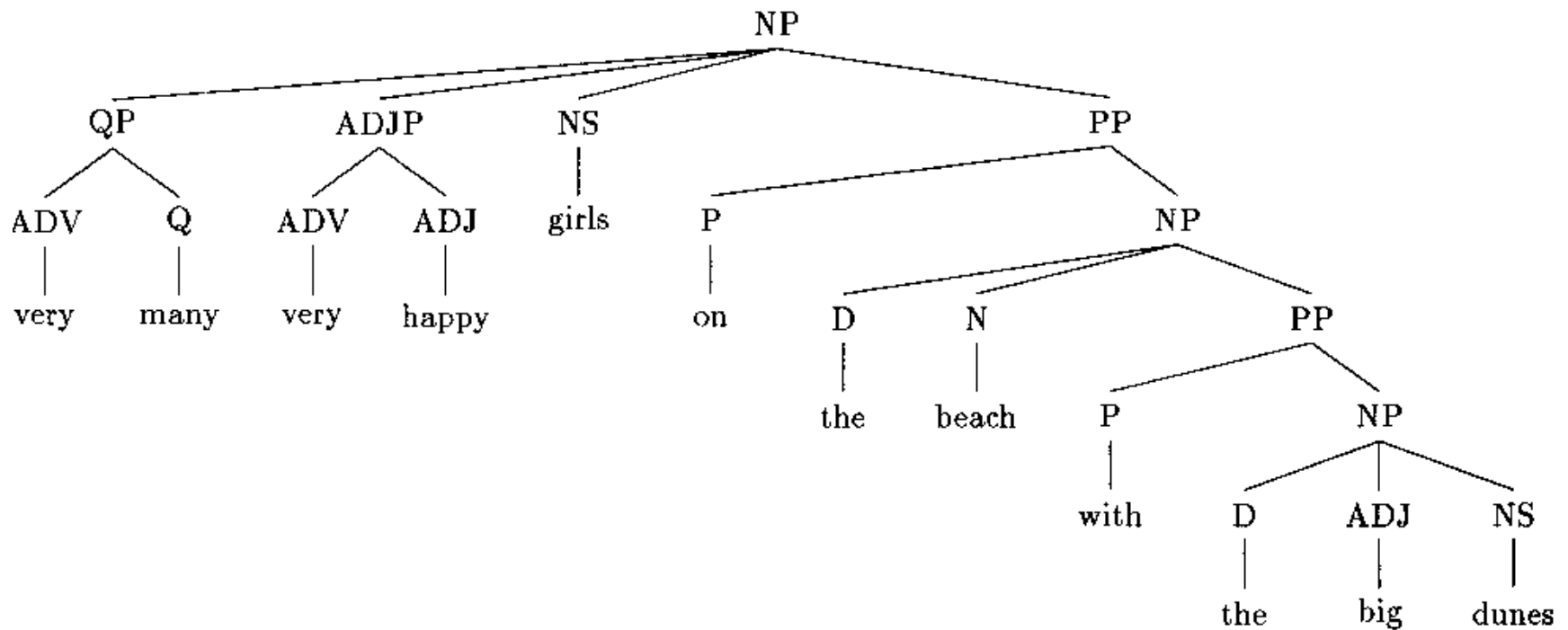
## 5.

### Mondatszintaxis, mondatelemzés



# Szintaktikai reprezentációk

# Szintaktikai szerkezetek reprezentációja fákkal





# A fareprezentáció kettős szerepe

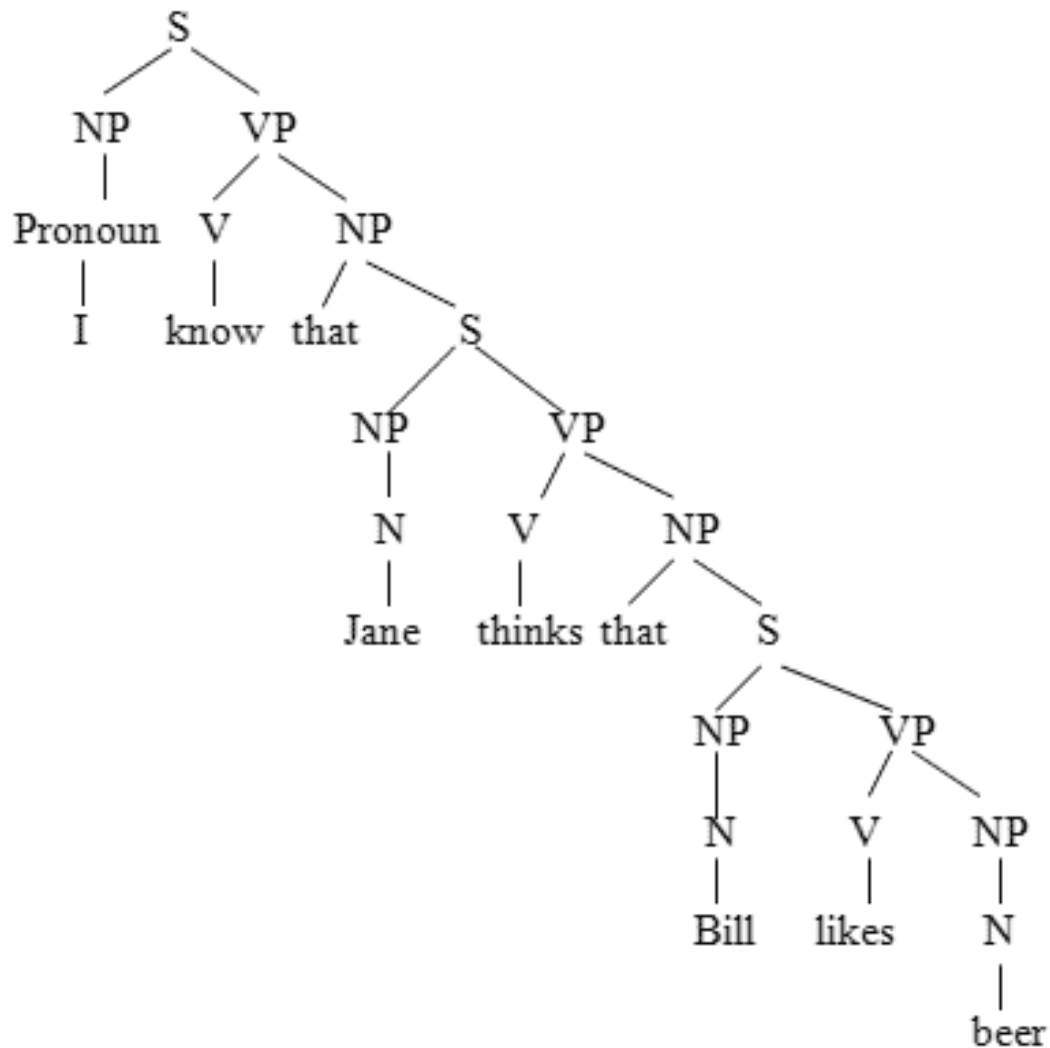
ID: 0 dominance

= mi függ mitől az adott szerkezetben?

LP: linear precedence

= mi mit előzhet meg?  
(az egész nyelvtenban)

## Közvetlen összetevők





## Morfológia és szintaxis határán

*Megbántottam őt.*

$S \rightarrow VP \ PP$

$VP \rightarrow V$

$V \rightarrow megbántottam$

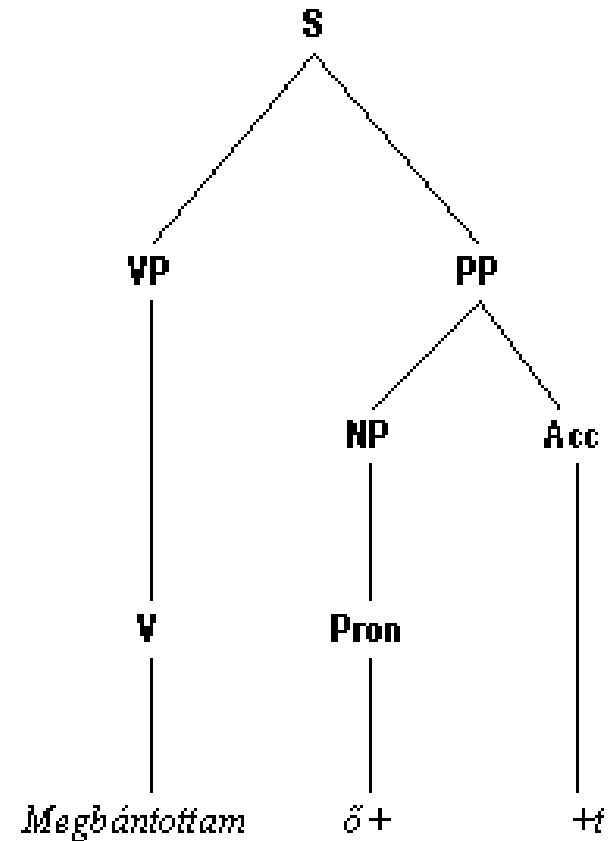
$PP \rightarrow NP \ Acc$

$NP \rightarrow Pron$

*ahol:*

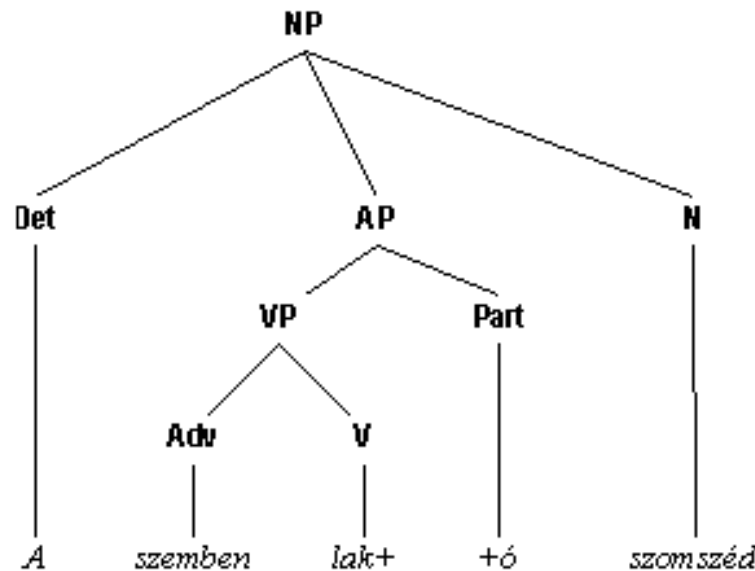
$Pron \rightarrow \text{ő}$

$ACC \rightarrow t \mid at \mid et \mid ot \mid \text{öt}$



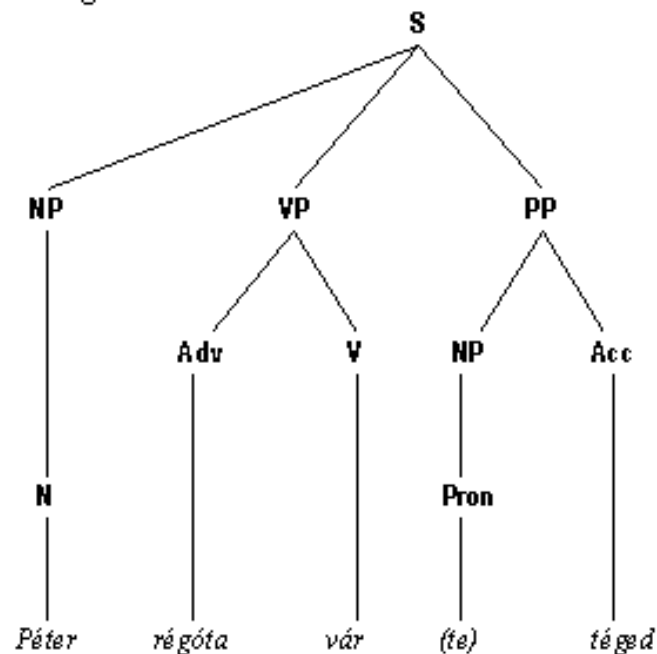
## Projektív fa = címkézett zárójelezés

[NP [Det **A**] [AP [VP [Adv **szemben**] [V **lak**]] [Part **ó**]] [N **szomszéd**]]

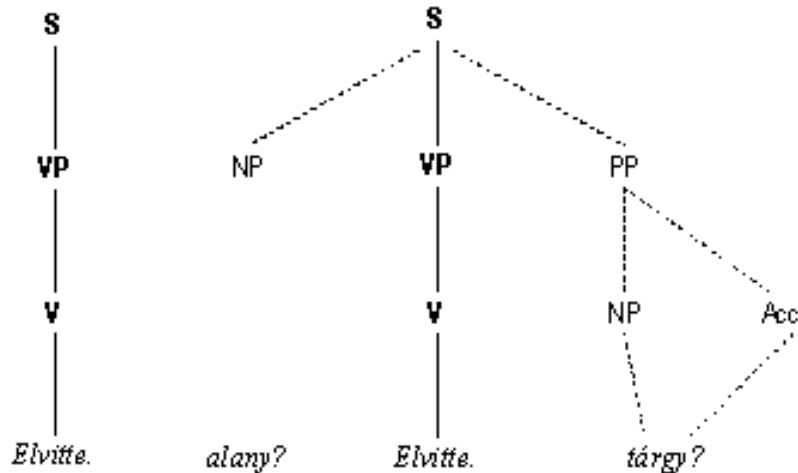


# Problémás esetek: üres csomópontok

*Péter régóta vár téged.*

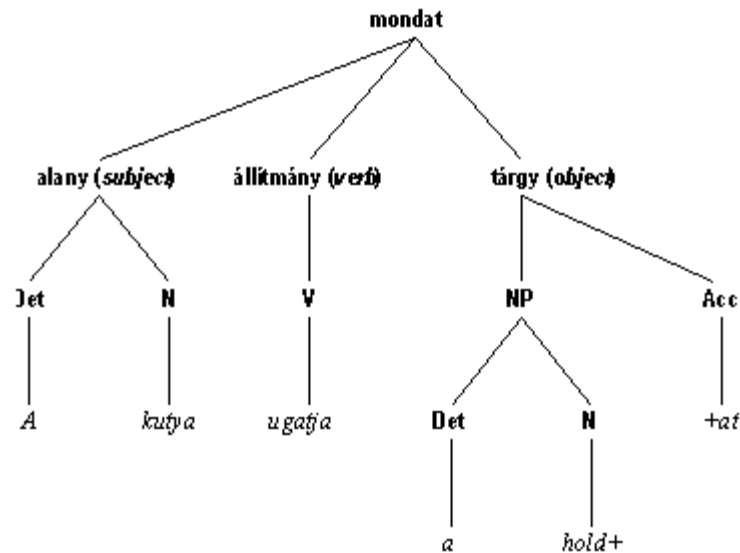


*Elvitte.*

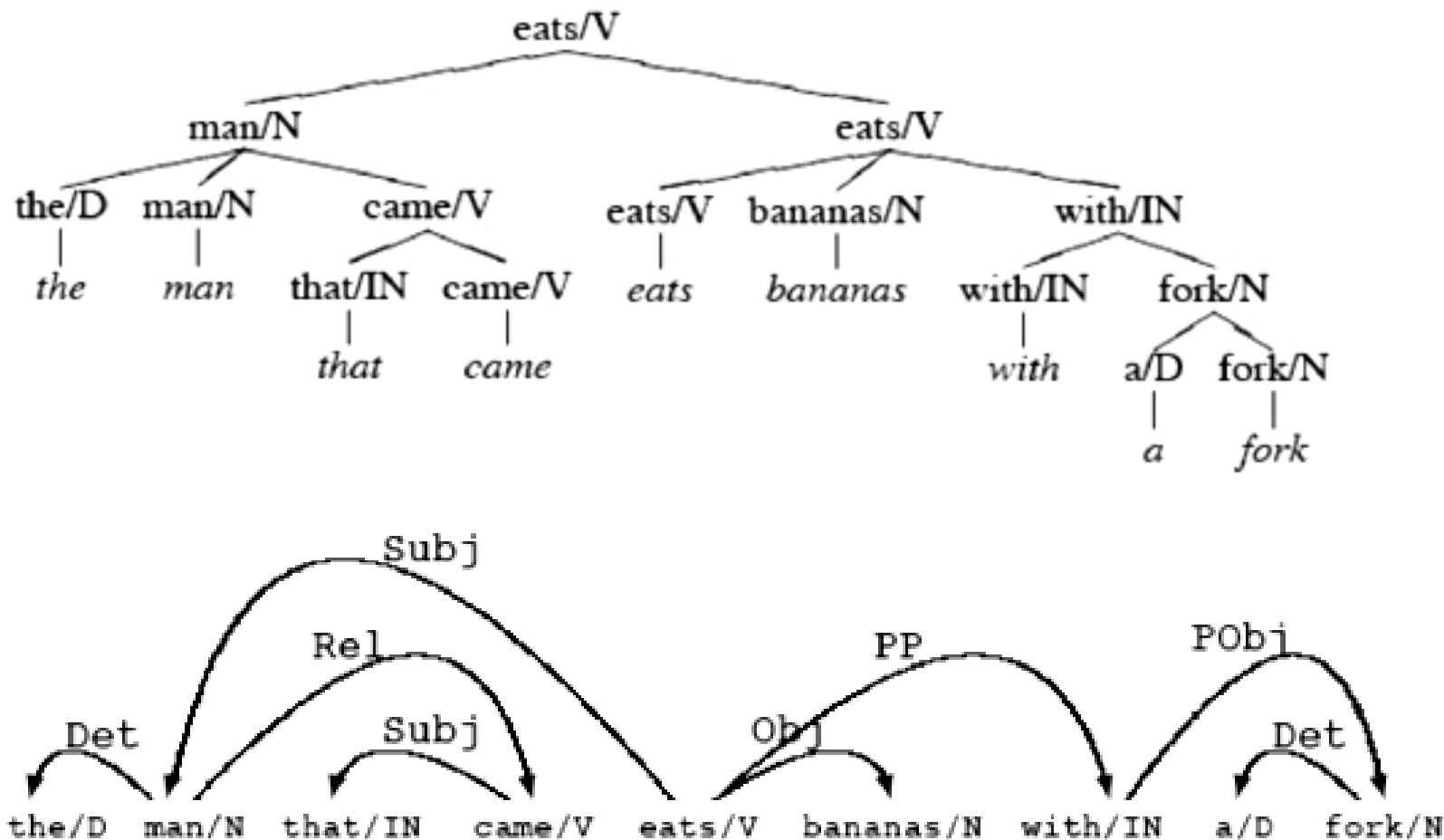




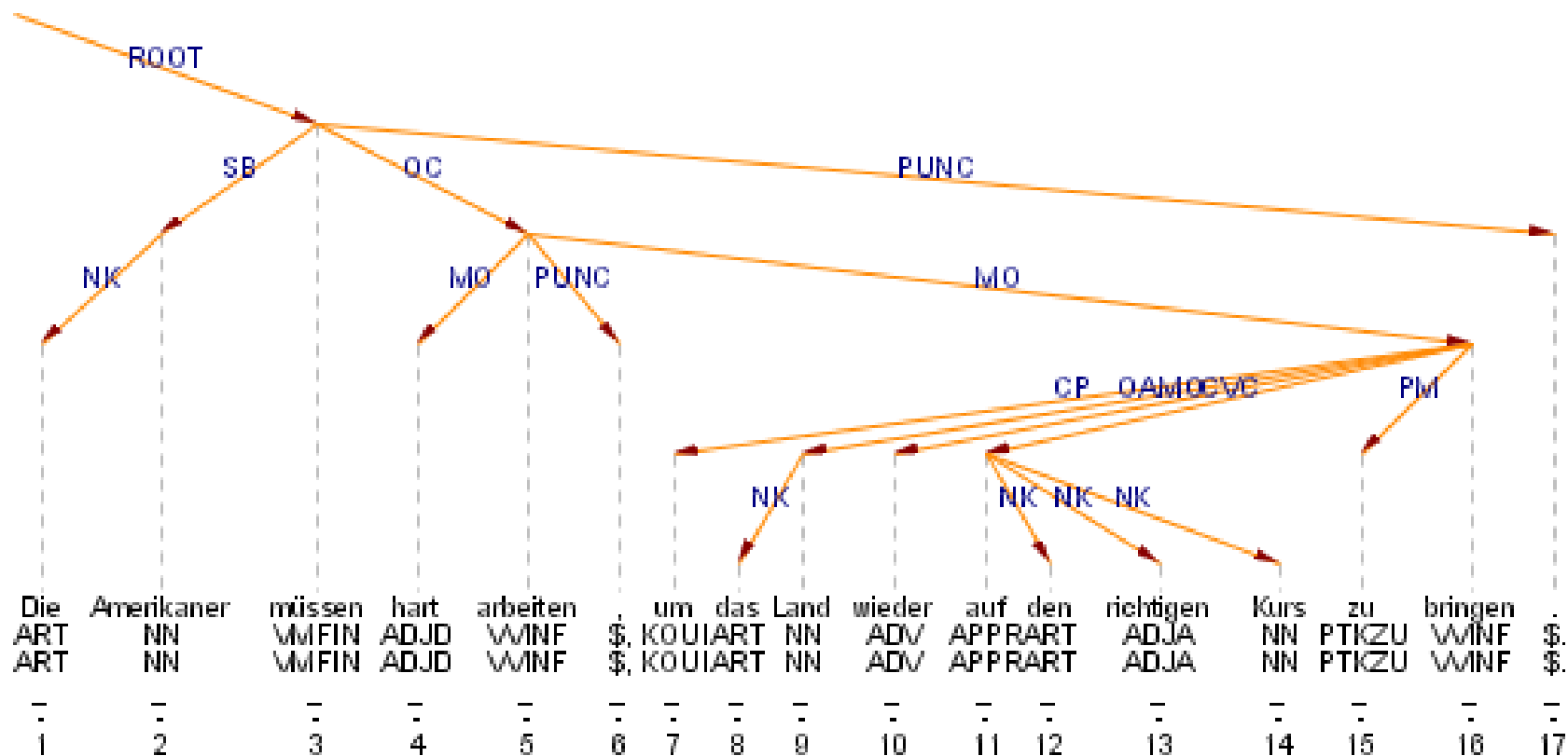
# Funkcionális szerkezet



# Összetevős és függőségi reprezentációk



## Projektív függőségi fa



## A mondat szerkezet-leírások összehasonlítása

- ❑ Közvetlen összetevős nyelvtanok:  
előnyük a magasabb szintű kategóriák bevezetésének lehetősége,  
hátrányuk a szintaktikai viszonyok egy részének „kifejezhetetlensége”
- ❑ Függőségi szerkezet:  
előnyük a szintaktikai függőség kifejezésének lehetősége,  
hátrányuk a magasabb szintű kategóriák kezelhetetlensége
- ❑ Egy elegáns közös megoldás: az X-vonás nyelvtanok

## X-vonás: összetevők és függőség

- ❑  $S \rightarrow NP VP$
- ❑ Az összetevős szerkezetben az NP és a VP „testvérek”, azaz mindketten az S „gyermekai”, de ezt nem fejezi ki a függőségi leírás
- ❑ Azt viszont a közvetlen összetevős leírás nem fejezi ki, hogy testvérek bár, de nem egyforma súllyal, ui. a VP a szerkezet feje
- ❑ X-vonás szabályként:  $V'' \rightarrow N' V'$
- ❑ Azaz: a  $V''$  a V maximális projekciója, tehát a mondat feje az ige!
- ❑ Csak endocentrikus szerkezetekre!  
(v.ö. exocentrikus)



## Az X-vonás nyelvtanok tulajdonságai

- ❑ Megőrzi mind az összetevős, mind a függőségi leírás előnyeit
- ❑ Szófajok feletti általánosítás:  
$$X = V / N / P / Adj / Adv / Det / Q / \dots$$
- ❑ Az X-nél magasabb szerkezeteket is egységesen lehet vele leírni
- ❑ Megkülönböztethetők a különféle jellegű bővítmények (pl. előtte, mögötte)
- ❑ Szintjei:  
X (= fej, v. X-nullvonás) a lexikális szerkezet  
X' (= X-vonás) a köztes szerkezet  
X'' (= X-kétvonás v. X-max) a maximális szerkezet, a tulajdonképpeni frázis

## Az X-vonás szabályok

- Az X-vonás szerkezet egy X fejből és egy opcionális komplementumból áll:

$$X' \rightarrow X \text{ (COMPLEMENT)}$$

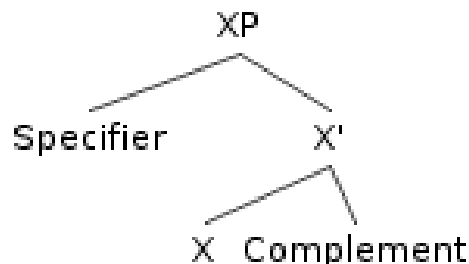
- Az X-vonás ezen kívül opcionálisan állhat egy X-vonás elemből és egy szabad bővítményből (ADJUNCT):

$$(X' \rightarrow X' \text{ ADJUNCT})$$

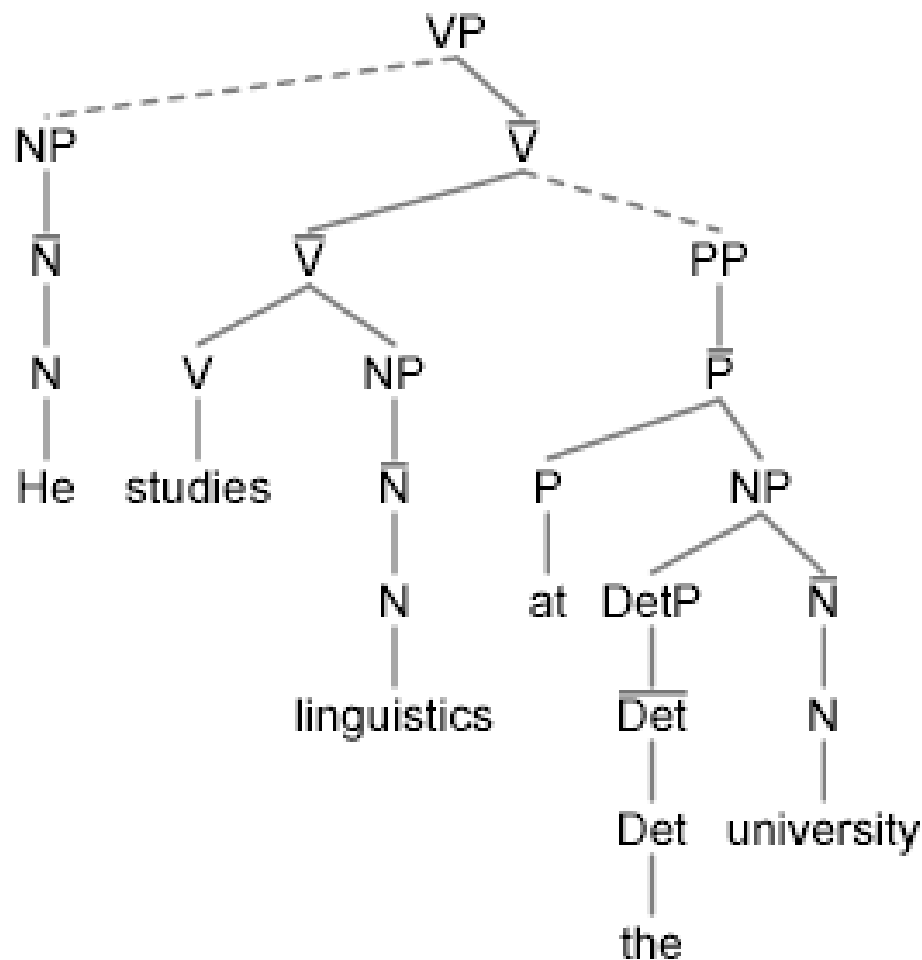
- Egy X-kétvonás (más néven XP)-szerkezet egy opcionális SPECIFIER elemből és egy X-vonásból áll:

$$XP \rightarrow (\text{SPECIFIER}) X'$$

- Az alap XP, azaz az X-vonás frázis tehát:



## Példa egy X-vonás szerkezetre







# Alapvető szintaktikai elemzési módszerek

# Nyelvtan, környezetfüggetlenség, elemzés

- ☐  $G = (N, T, S, P)$ , ahol
  - T: a szintaxisfa levelei a nyelvtan terminális szimbólumai
  - N: a szintaxisfa többi pontja, a nemterminális szimbólumok
  - S: a gyökérelem, a nyelvtan kezdőszimbóluma
  - P: a szabályok
- ☐ Két nyelvtan gyengén ekvivalens, ha ugyanazt a nyelvet ismeri fel, és erősen ekvivalens, ha ugyanazokat a szerkezeteket építi fel
- ☐ Az emberi nyelvekhez a reguláris: kevés környezetfüggő: sok! (?)
- ☐ Környezetfüggetlen (Chomsky 2-es típusú) nyelvtan:
$$A \rightarrow \alpha, \quad \text{ahol } A \in N, \alpha \in (N \cup T)^*$$
- ☐ A szintaxisfa belső részének felépítésére több módszer létezik
- ☐ Ha az S szimbólumból kiindulva építjük fel a szintaxisfát: felülről-lefelé (TD) haladó elemzés
- ☐ Ha a szintaxisfa építése a levelekből kiindulva halad az S szimbólum felé: alulról-felfelé (BU) elemzés

# Egy „hagyományos” mininyelvten angolra

$S \rightarrow NP VP$

$S \rightarrow Aux NP VP$

$S \rightarrow VP$

$NP \rightarrow Det Nominal$

$NP \rightarrow Proper-Noun$

$VP \rightarrow Verb$

$VP \rightarrow Verb NP$

$Nominal \rightarrow Noun$

$Nominal \rightarrow Noun Nominal$

$Nominal \rightarrow Nominal PP$

$PP \rightarrow Prep NP$

$Det \rightarrow that \mid this \mid a$

$Noun \rightarrow book \mid flight \mid meal \mid money$

$Verb \rightarrow book \mid include \mid prefer$

$Aux \rightarrow does$

$Prep \rightarrow from \mid to \mid on$

$Proper-Noun \rightarrow Houston \mid TWA$



# Alulról-felfelé (BU) elemzés

## (Book that flight)

 $S \rightarrow NP VP$  $S \rightarrow Aux NP VP$  $S \rightarrow VP$  $NP \rightarrow Det Nominal$  $Nominal \rightarrow Noun$  $Nominal \rightarrow Noun Nominal$  $NP \rightarrow Proper-Noun$  $VP \rightarrow Verb$  $VP \rightarrow Verb NP$ 

Book that flight

Noun Det Noun  
| | |  
Book that flight

Verb Det Noun  
| | |  
Book that flight

NOM NOM  
| |  
Noun Det Noun  
| | |  
Book that flight

NOM  
|  
Verb Det Noun  
| | |  
Book that flight

NP  
| |  
NOM NOM  
| |  
Noun Det Noun  
| | |  
Book that flight

VP NOM  
| |  
Verb Det Noun  
| | |  
Book that flight

NP  
| |  
Verb Det Noun  
| | |  
Book that flight

VP NP  
| |  
Verb Det NOM  
| | |  
Book that flight

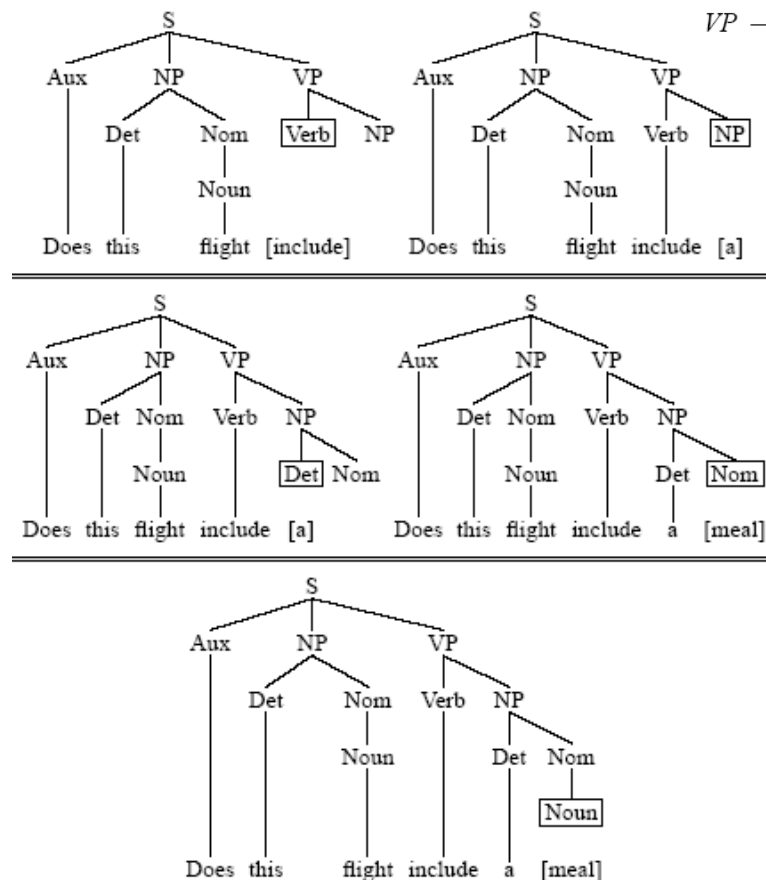
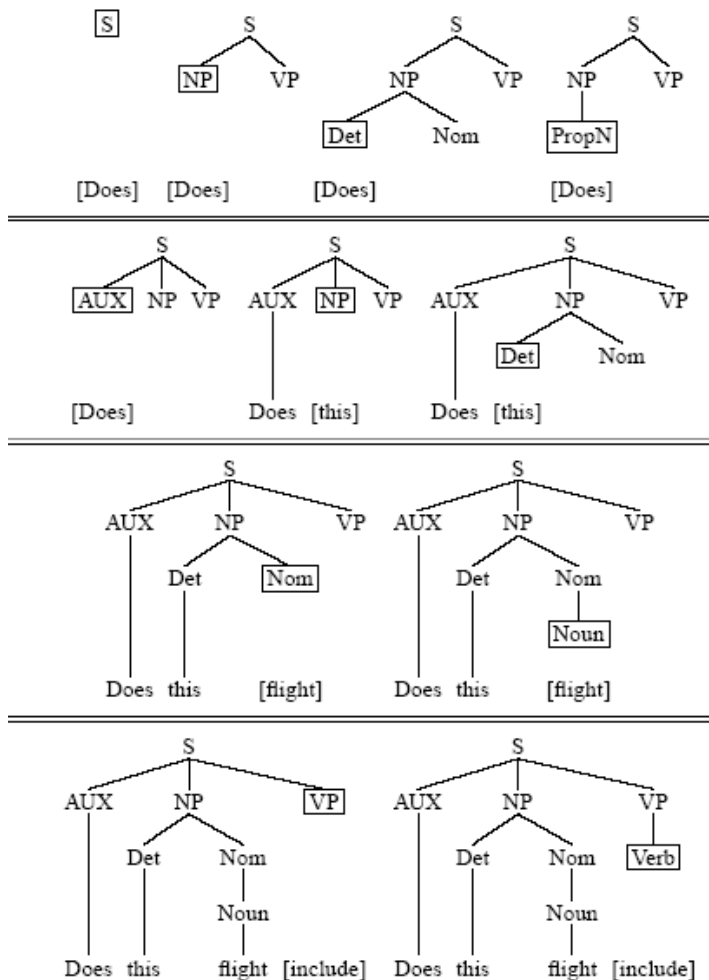
VP NP  
| |  
Verb Det NOM  
| | |  
Book that flight

S  
|  
VP NP  
| |  
Verb Det Nom  
| | |  
Book that flight

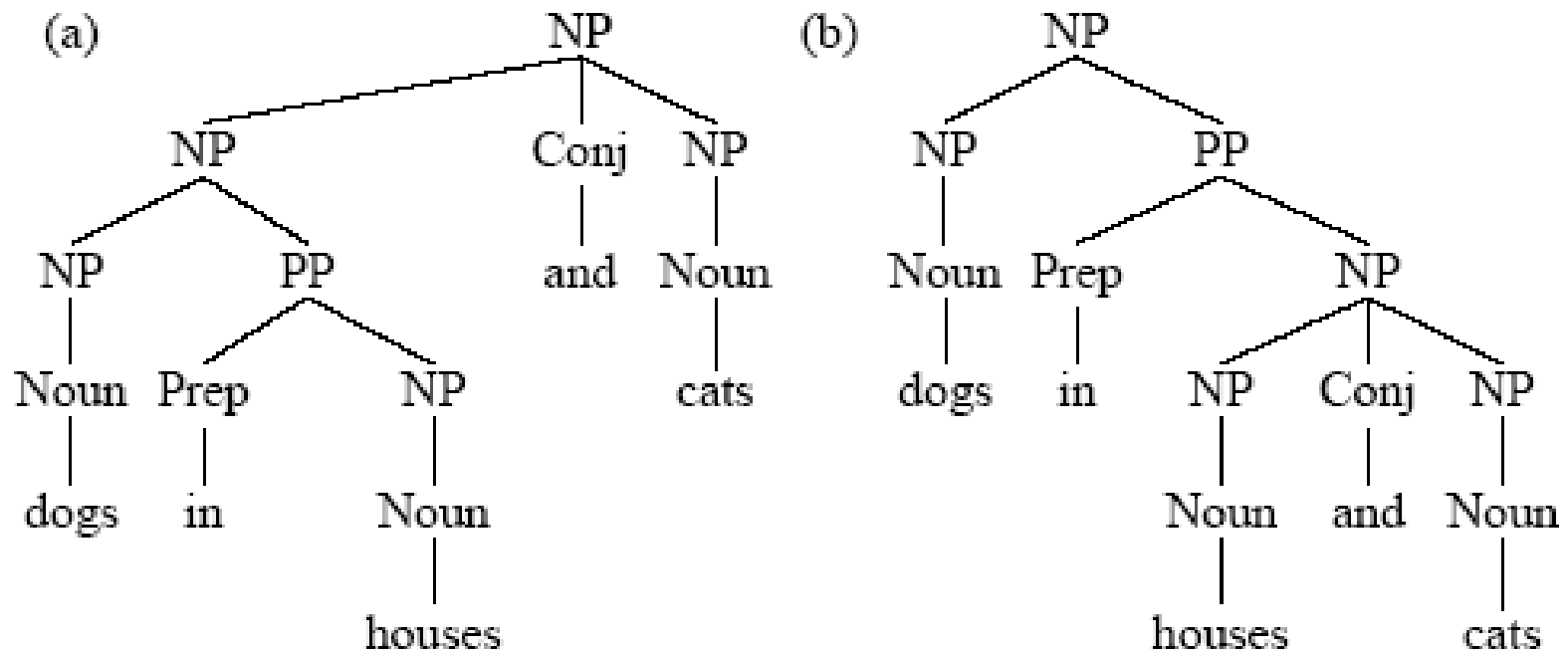
# Felülről-lefelé (TD) elemzés

(Does this flight include a meal?)

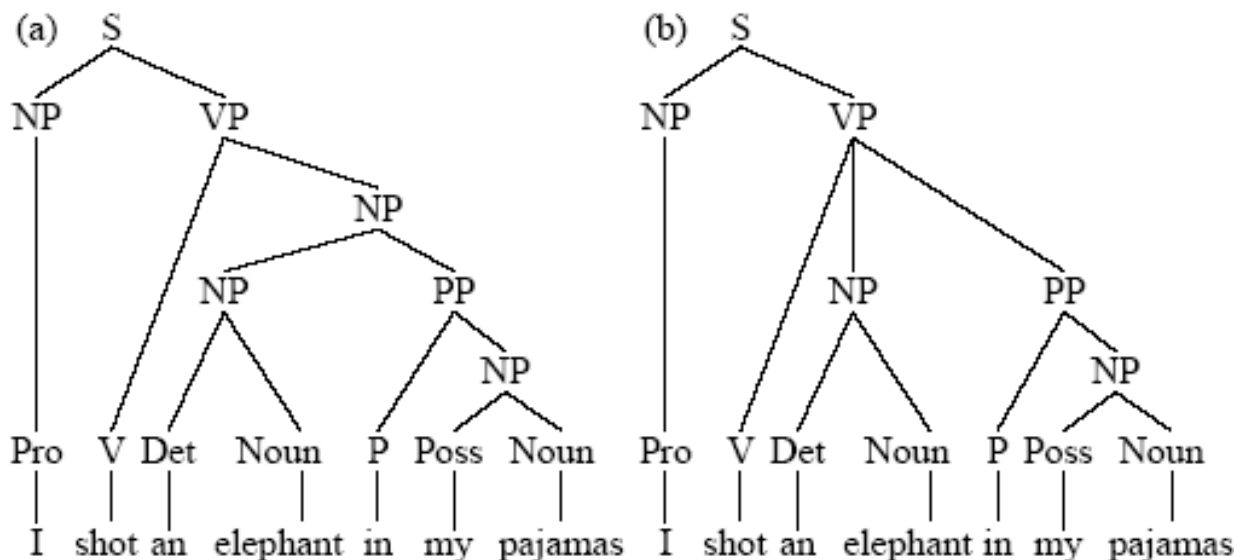
$S \rightarrow NP VP$   
 $S \rightarrow Aux NP VP$   
 $S \rightarrow VP$   
 $NP \rightarrow Det Nominal$   
 $Nominal \rightarrow Noun$   
 $Nominal \rightarrow Noun Nominal$   
 $NP \rightarrow Proper-Noun$   
 $VP \rightarrow Verb$   
 $VP \rightarrow Verb NP$



## Egy örök nyelvi probléma: hová tartozik a PP?



$S \Rightarrow$ 
 $\begin{array}{c} S \\ \swarrow \quad \searrow \\ NP \quad VP \end{array}$ 
 $\Rightarrow$ 
 $\begin{array}{c} S \\ \swarrow \quad \searrow \\ NP \quad VP \\ \swarrow \quad \searrow \\ NP \quad PP \end{array}$ 
 $\Rightarrow$ 
 $\begin{array}{c} S \\ \swarrow \quad \searrow \\ NP \quad VP \\ \swarrow \quad \searrow \\ NP \quad PP \\ \swarrow \quad \searrow \\ NP \quad PP \end{array}$ 
 $\dots$



## Egy példa a balrekurzióra

$n > 1$  esetén a balrekurziót természetes nyelvekben meglehetősen nehéz megérteni:

$$\begin{aligned} S &\rightarrow NP VP \\ NP &\rightarrow RelPron S \end{aligned}$$

0: *A fiú elment.*

1: *A fiú,  
akit a barátom meghívott,  
elment.*

2: *A fiú,  
akit a barátom,  
akiről a kollégám mesélt,  
meghívott,  
elment.*

3: *A fiú,  
akit a barátom,  
akiről a kollégám,  
akivel egy iskolába jártam,  
mesélt,  
meghívott,  
elment.*



## Egy példa a jobbrekurzióra

A nyelvekben a jobbrekurzió ( $VP \rightarrow PP VP$ ) mélysége nem jelent problémát az (emberi) feldolgozásban:

*„az agyag ölelő karjai közül kibontakozni akaró  
kocsikerék rettentő nyikorgásától megriadt juhászkutya  
bundájába kapaszkodó kullancs kidülledt félszeméből  
alácseppenő könnycseppben visszatükröződő holdvilág  
fényétől illuminált rablólovagvár felvonóhídjából kiálló  
vasszegek kohéziós erejének hatása”*

(Fehér G.)

# A jobbrekurziós alárendelés természete

1: A karok ölelnek.

→ ölelő karok

2: A kocsikerék ki akar bontakozni (vhonnan).

→ (vhonnan) kibontakozni akaró kocsikerék

3: A juhászkutya megriadt (vmitől).

→ (vmitől) megriadt juhászkutya

4: A kullancs kapaszkodik (vmibe).

→ (vmibe) kapaszkodó kullancs

5: A könnyecsepp alácseppen (vhonnan).

→ (vhonnan) alácseppenő könnyecsepp

6: A holdvilág visszatükröződik (vmiben).

→ (vmiben) visszatükröződő holdvilág

7: A rablólovagvár illuminált (vmitől).

→ (vmitől) illuminált rablólovagvár

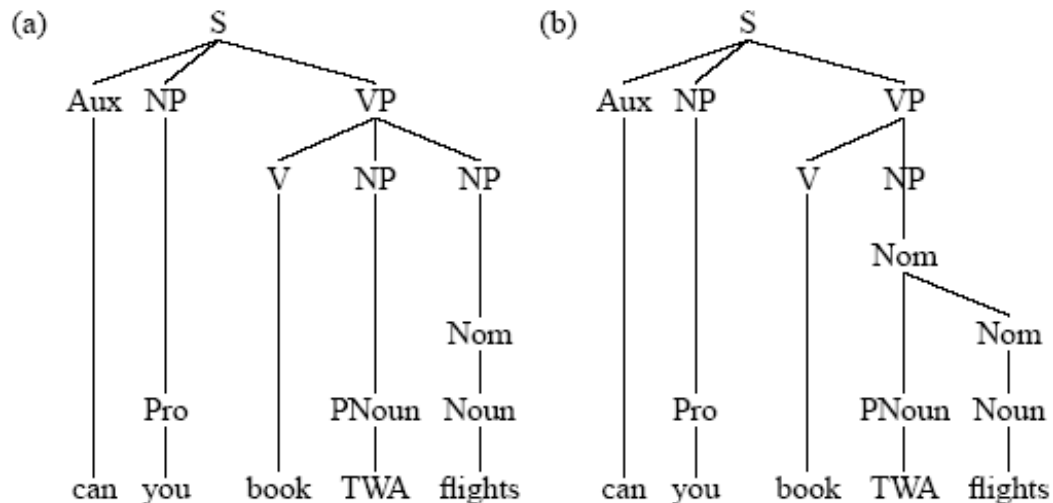
8: A vasszegek kiállnak (vhonnan).

→ (vhonnan) kiálló vasszegek

## A TD- és a BU-stratégiák összehasonlítása

- ☐ TD  
nem fordulhat elő, hogy ne mondat-szimbólummal végződő elemzéssel töltsük az időt, és  
nincs olyan részszerkezet, melynek ne lenne helye a végső fában
- ☐ BU:  
nem fordulhat elő, hogy a bemenettel inkonzisztens szabályokkal kellene töltenie az időt, és  
nincs olyan részszerkezet, mely ne lenne kompatibilis a bemenet valamely részével
- ☐ Optimális megoldás: valamiféle kombináció, pl.  
BU-elemzés előfordulási valószínűségekkel;  
TD-elemzés „BU-jóslással”

# A valószínűség bevonása az elemzésbe



Egy megoldás valószínűségek használatával:

Rules	P	Rules	P
S → Aux NP VP	.15	S → Aux NP VP	.15
NP → Pro	.40	NP → Pro	.40
VP → V NP NP	.05	VP → V NP	.40
NP → Nom	.05	NP → Nom	.05
NP → PNoun	.35	Nom → PNoun Nom	.05
Nom → Noun	.75	Nom → Noun	.75
Aux → Can	.40	Aux → Can	.40
NP → Pro	.40	NP → Pro	.40
Pro → you	.40	Pro → you	.40
Verb → book	.30	Verb → book	.30
PNoun → TWA	.40	Pnoun → TWA	.40
Noun → flights	.50	Noun → flights	.50

## Balsarok-elemzés

B-t A balsarkának nevezzük, ha van A-nak B-vel kezdődő levezetése

$$S \rightarrow NP VP$$
$$S \rightarrow Aux NP VP$$
$$S \rightarrow VP$$
$$NP \rightarrow Det Nominal$$
$$Nominal \rightarrow Noun$$
$$Nominal \rightarrow Noun Nominal$$
$$NP \rightarrow Proper-Noun$$
$$VP \rightarrow Verb$$
$$VP \rightarrow Verb NP$$

Ennek a nyelvtannak a balsarok-táblája az alábbi:

Szófaj	Balsarok
<i>S</i>	<i>Det, Proper-Noun, Aux, Verb</i>
<i>NP</i>	<i>Det, Proper-Noun</i>
<i>Nominal</i>	<i>Noun</i>
<i>VP</i>	<i>Verb</i>

## A balsarok-elemzés működése

Szófaj	Balsarok
<i>S</i>	<i>Det, Proper-Noun, Aux, Verb</i>
<i>NP</i>	<i>Det, Proper-Noun</i>
<i>Nominal</i>	<i>Noun</i>
<i>VP</i>	<i>Verb</i>

Tehát:

a „*Does this flight include a meal?*” esetében a

$$S \rightarrow NP VP$$

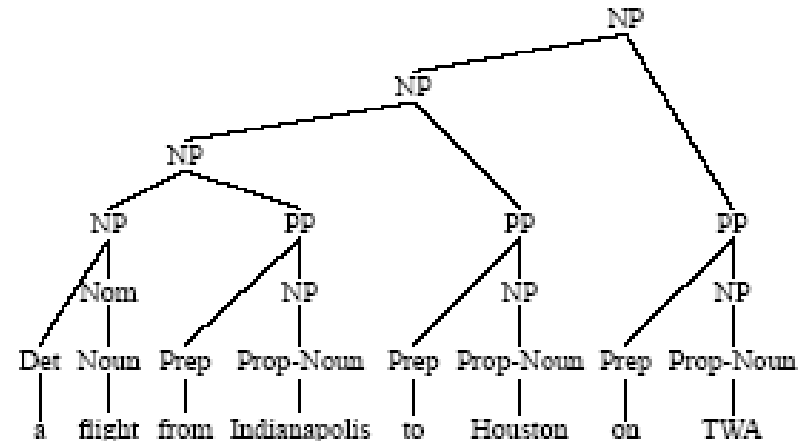
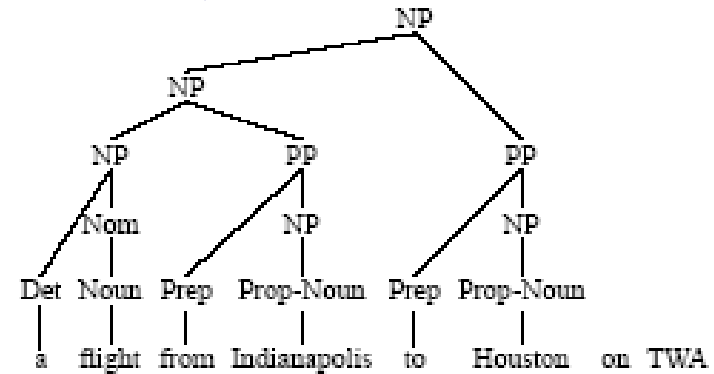
$$S \rightarrow Aux NP VP$$

$$S \rightarrow VP$$

három szabályból TD csak a középső jöhet szóba

a *does* Aux volta miatt!

(visszalépéses TD-elemzés)



1: on TWA, a flight from Indianapolis to Houston on TWA

## Mondatelemzés Earley-algoritmussal

- ❑ Earley-algoritmus: párhuzamos TD-elemzés
- ❑ BU-párja a Cocke – Young – Kasami-algoritmus (CYK)
- ❑ Exponenciális idejű problémát polinomiálisra redukál
- ❑ Azaz: dinamikus programozáson alapul
- ❑ Lényege:  
a repetitív részszerkezet-kezelés eliminálása
- ❑ Magja:  
egy  $n+1$  elemű táblázat és az ún. pontozott szabály



## A pontozott szabály

Ha az elemzéssel a bemenet  $j$ -edik betűjéig jutottunk el, miközben megállapítottuk, hogy léteznek olyan

$$p, q, r, t \in (NUT)^*$$

szavak, melyekre

$$S \Rightarrow^* pAq$$

$$p \Rightarrow^* a_1 \dots a_i$$

$$q \Rightarrow^* a_{i+1} \dots a_j$$

$$A \rightarrow rt \in P$$

akkor ezt a körülményt fejezzük ki azzal, hogy az

$$A \rightarrow r \bullet t$$

az  $\langle i, j \rangle$  indexű pontozott szabály

## A pontozós elemzési lépések fajtái

$S \rightarrow \bullet VP$

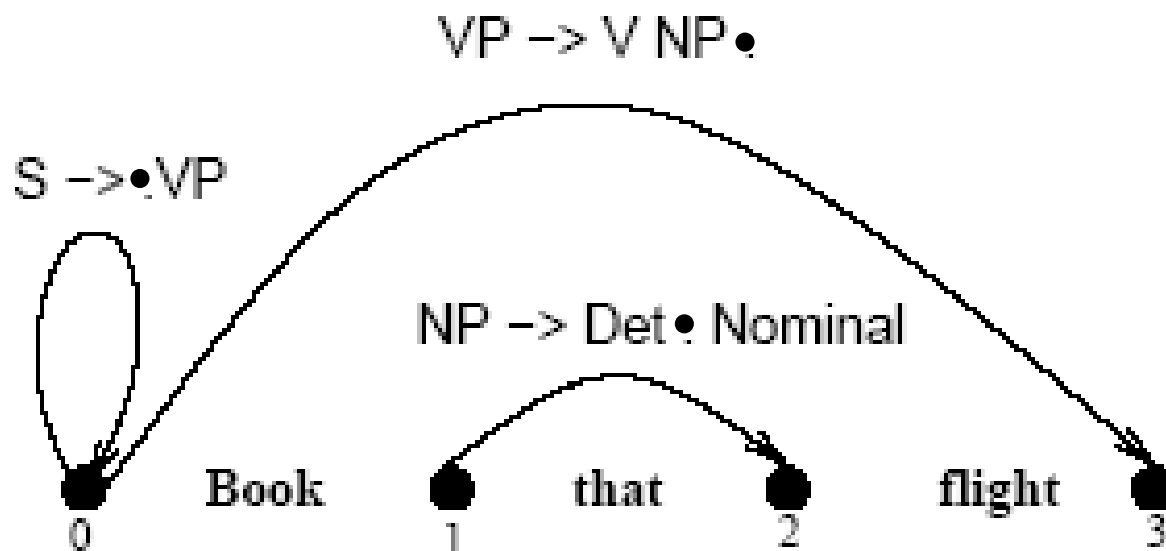
$NP \rightarrow \text{Det} \bullet \text{Nominal}$

$VP \rightarrow V NP \bullet$

jósló (predictor)

elemző (scanner)

befejező (completer)



# Az Earley-algoritmus

```
function EARLEY-PARSE(words, grammar) returns chart
  ENQUEUE( $(\gamma \rightarrow \bullet S, [0, 0])$ , chart[0])
  for  $i \leftarrow$  from 0 to LENGTH(words) do
    for each state in chart[i] do
      if INCOMPLETE?(state) and
        NEXT-CAT(state) is not a part of speech then
        PREDICTOR(state)
      elseif INCOMPLETE?(state) and
        NEXT-CAT(state) is a part of speech then
        SCANNER(state)
      else
        COMPLETER(state)
    end
  end
  return(chart)

procedure PREDICTOR( $(A \rightarrow \alpha \bullet B \beta, [i, j])$ )
  for each  $(B \rightarrow \gamma)$  in GRAMMAR-RULES-FOR( $B$ , grammar) do
    ENQUEUE( $(B \rightarrow \bullet \gamma, [j, j])$ , chart[j])
  end

procedure SCANNER( $(A \rightarrow \alpha \bullet B \beta, [i, j])$ )
  if  $B \subset$  PARTS-OF-SPEECH(word[j]) then
    ENQUEUE( $(B \rightarrow \text{word}[j], [j, j + 1])$ , chart[j+1])

procedure COMPLETER( $(B \rightarrow \gamma \bullet, [j, k])$ )
  for each  $(A \rightarrow \alpha \bullet B \beta, [i, j])$  in chart[j] do
    ENQUEUE( $(A \rightarrow \alpha B \bullet \beta, [i, k])$ , chart[k])
  end

procedure ENQUEUE(state, chart-entry)
  if state is not already in chart-entry then
    PUSH(state, chart-entry)
  end
```

Konkrét implementációk:

[https://en.wikipedia.org/wiki/Earley\\_parser#Implementations](https://en.wikipedia.org/wiki/Earley_parser#Implementations)

# Példa az Earley-elemzésre

(a circle touches a square)

S → NP VP  
NP → Det N  
VP → VT NP  
VP → VI PP  
PP → P NP

Det → a  
N → circle|square|triangle  
VT → touches  
VI → is  
P → above|below

	a	circle	touches	a	square
□ → .S	scanned	scanned	scanned	scanned	scanned
predicted	□ Det → a.	1 N → circle.	2 VT → touches.	3 Det → a.	4 N → triangle.
□ S → .NP VP	completed	completed	completed	completed	completed
□ NP → .Det N	□ NP → Det.N	□ NP → Det N.	2 VP → VT.NP	3 NP → Det.N	4 NP → Det N.
□ Det → .a	predicted	□ S → NP.VP	predicted	predicted	3 VP → VT NP.
	1 N → .circle	predicted	3 NP → .Det N	5 N → .circle	□ S → NP VP.
	1 N → .square	2 VP → .VT NP	3 Det → .a	4 N → .square	□ → S.
	1 N → .triangle	2 VP → .VI PP		4 N → .triangle	
		2 VT → .touches			
		2 VI → .is			
State set 0	1	2	3	4	5



# Az unifikációs szintaxis

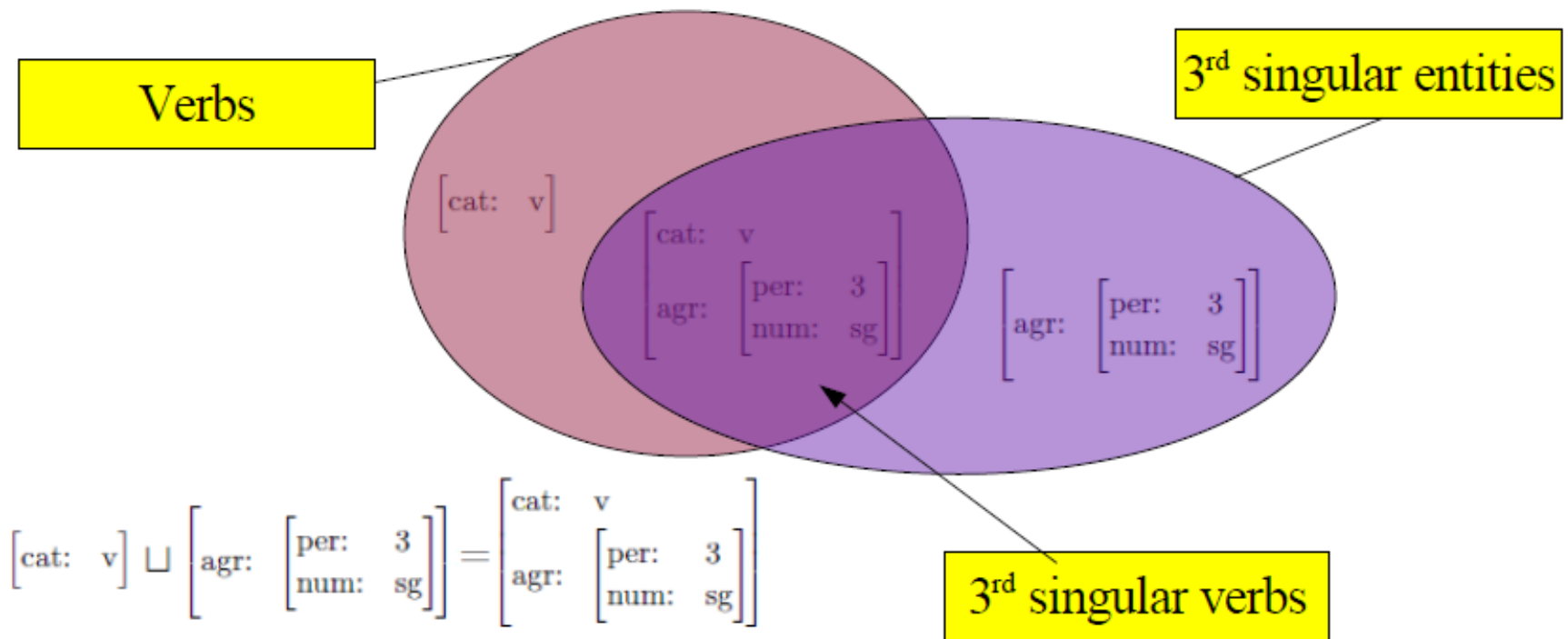


## Jegyszerkezetek

- ☐ A kategóriák (szófajok) nem atomiak, tovább alosztályozhatók: *VPto*, *Sthat*, *Non3sgAux*, *3sgNP*, *tV*, *NPmass*, ...
- ☐ Az efféle jegyek hierarchiába rendezhetők
- ☐ A szabályok így szabályosztályokká válnak
- ☐ Jegy-érték párok
- ☐ Jegyek: nyelvtani tulajdonságok, értékek: atomi szimbólumok és jegyszerkezetek
- ☐ Jegy-ösvény: nem egyetlen jegynek van értéke, hanem az ösvénynek
- ☐ Például: *[egyeztetés[szám]] = Plur*
- ☐ Fej-jegyek és láb-jegyek

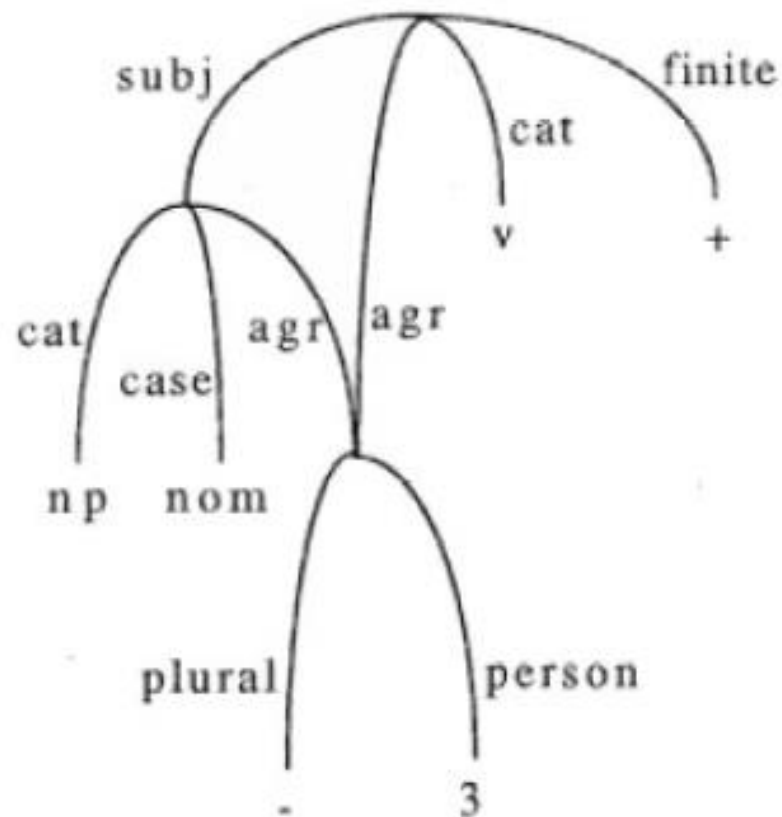
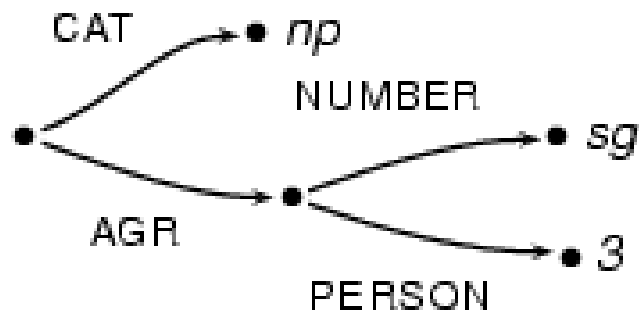
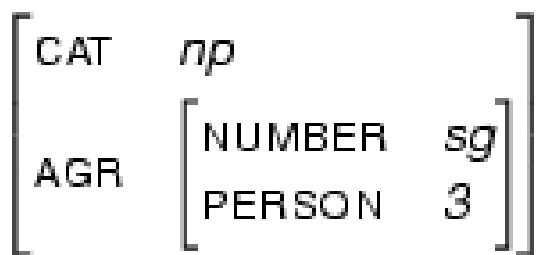
# Diszjunkt jegy-érték párok unifikációja

Az információk unifikációja a denotációk metszete:



Berthold Crysmann tananyagából ([http://www.coli.uni-saarland.de/courses/msc-prep-06/slides/VL\\_PrepSyntaxUBG.pdf](http://www.coli.uni-saarland.de/courses/msc-prep-06/slides/VL_PrepSyntaxUBG.pdf))

## Jegyszerkezetek DAG-ként





# A jegyszerkezetek reprezentációja

Az unifikációs rész bemutatásához Berthold Crysmann tananyagát használtam fel ([http://www.coli.uni-saarland.de/courses/msc-prep-06/slides/VL\\_PrepSyntaxUBG.pdf](http://www.coli.uni-saarland.de/courses/msc-prep-06/slides/VL_PrepSyntaxUBG.pdf))

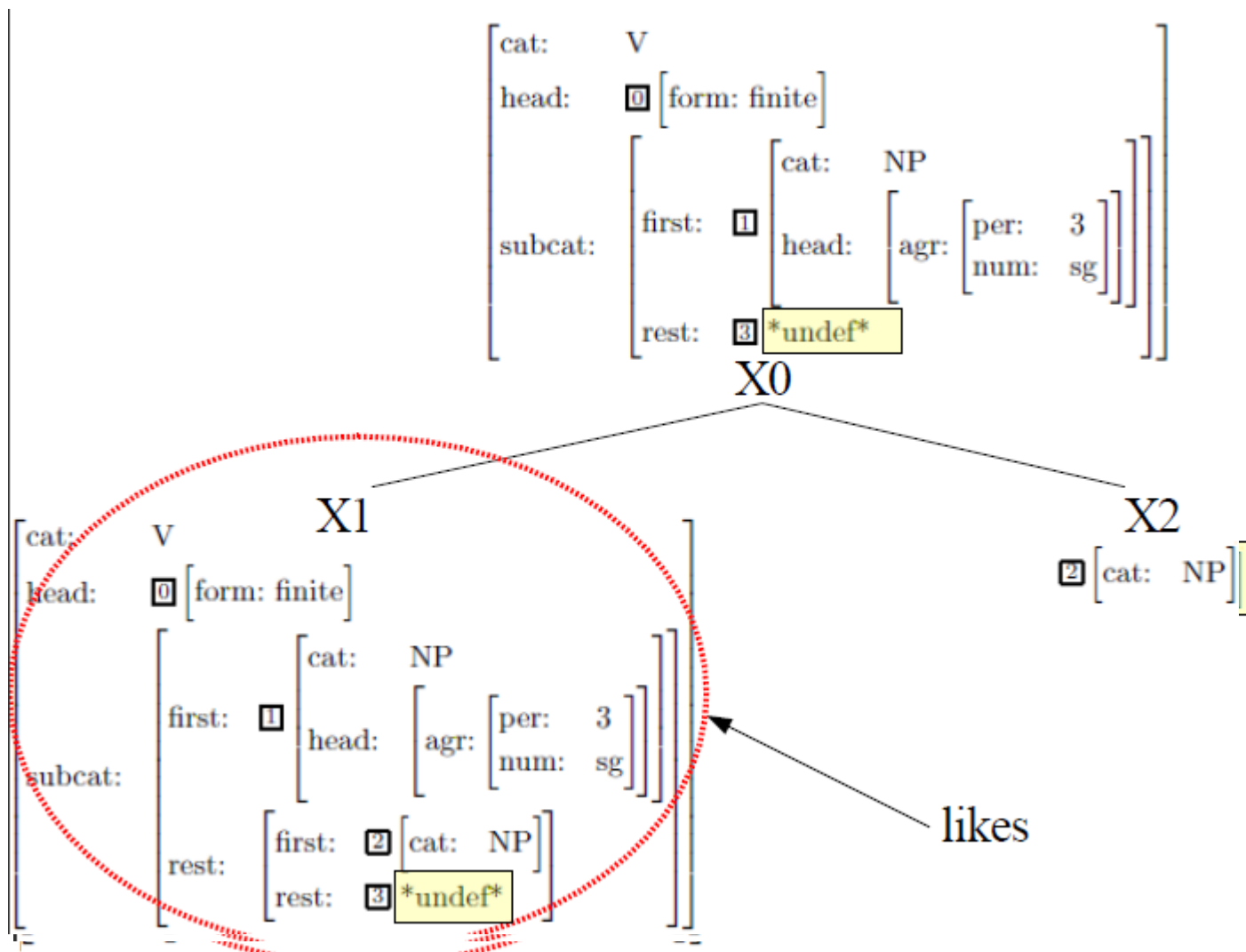
- ❑ Ösvényekre vonatkozó egyenletekkel:
  - $\langle \text{cat} \rangle = v$
  - $\langle \text{finite} \rangle = +$
  - $\langle \text{agr:plural} \rangle = -$
  - $\langle \text{agr:person} \rangle = 3$
  - $\langle \text{subj:cat} \rangle = np$
  - $\langle \text{subj:case} \rangle = \text{nom}$
  - $\langle \text{subj:agr} \rangle = \langle \text{agr} \rangle$

- ❑ Jegy-érték mátrixokkal:

$$\left[ \begin{array}{l} \text{cat:} \\ \text{finite:} \\ \text{agr:} \\ \text{subj:} \end{array} \begin{array}{l} v \\ + \\ \boxed{1} \left[ \begin{array}{l} \text{plural:} \quad - \\ \text{per:} \quad 3 \end{array} \right] \\ \left[ \begin{array}{l} \text{cat:} \quad np \\ \text{case:} \quad \text{nom} \\ \text{agr:} \quad \boxed{1} \left[ \begin{array}{l} \end{array} \right] \end{array} \right] \end{array} \right]$$

# A jegyszerkezet mint szabály

Berthold Crysmann tananyagából ([http://www.coli.uni-saarland.de/courses/msc-prep-06/slides/VL\\_PrepSyntaxUBG.pdf](http://www.coli.uni-saarland.de/courses/msc-prep-06/slides/VL_PrepSyntaxUBG.pdf))



# Az unifikációs Earley-algoritmus

```
function EARLEY-PARSE(words, grammar) returns chart
  ENQUEUE( $(\gamma \rightarrow \bullet S, [0, 0], dag_\gamma)$ , chart[0])
  for  $i \leftarrow$  from 0 to LENGTH(words) do
    for each state in chart[i] do
      if INCOMPLETE?(state) and
        NEXT-CAT(state) is not a part of speech then
        PREDICTOR(state)
      elseif INCOMPLETE?(state) and
        NEXT-CAT(state) is a part of speech then
        SCANNER(state)
      else
        COMPLETER(state)
    end
  end
  return(chart)

procedure PREDICTOR( $(A \rightarrow \alpha \bullet B \beta, [i, j], dag_A)$ )
  for each  $(B \rightarrow \gamma)$  in GRAMMAR-RULES-FOR(B, grammar) do
    ENQUEUE( $(B \rightarrow \bullet \gamma, [j, j], dag_B)$ , chart[j])
  end

procedure SCANNER( $(A \rightarrow \alpha \bullet B \beta, [i, j], dag_A)$ )
  if  $B \in$  PARTS-OF-SPEECH(word[j]) then
    ENQUEUE( $(B \rightarrow word[j], [j, j+1], dag_B)$ , chart[j+1])

procedure COMPLETER( $(B \rightarrow \gamma \bullet, [j, k], dag_B)$ )
  for each  $(A \rightarrow \alpha \bullet B \beta, [i, j], dag_A)$  in chart[j] do
    if new-dag  $\leftarrow$  UNIFY-STATES( $dag_B, dag_A, B$ )  $\neq$  Fails!
      ENQUEUE( $(A \rightarrow \alpha B \bullet \beta, [i, k], new-dag)$ , chart[k])
  end

procedure UNIFY-STATES(dag1, dag2, cat)
  dag1-cp  $\leftarrow$  COPYDAG(dag1)
  dag2-cp  $\leftarrow$  COPYDAG(dag2)
  UNIFY(FOLLOW-PATH(cat, dag1-cp), FOLLOW-PATH(cat, dag2-cp))

procedure ENQUEUE(state, chart-entry)
  if state is not subsumed by a state in chart-entry then
    PUSH(state, chart-entry)
  end
```

## További kiterjesztések

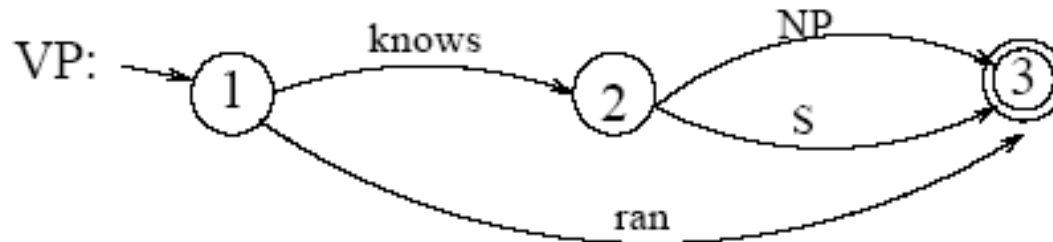
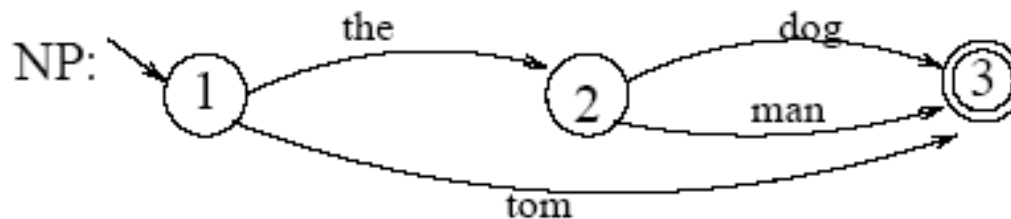
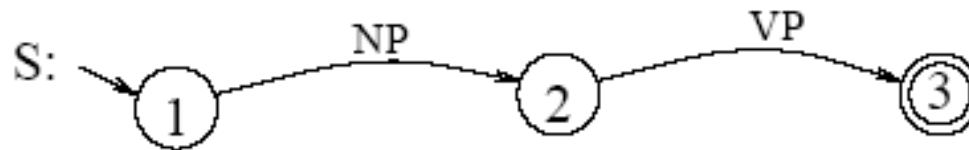
- ☐ Negáció
- ☐ Diszjunkció
- ☐ Ösvény-egyenlőtlenség
- ☐ Halmaz-értékű jegyek
- ☐ Jegyszerkezet-leíró metanyelv



# A véges állapotú eszközök kiterjesztései szintaktikai elemzéshez

# RTN

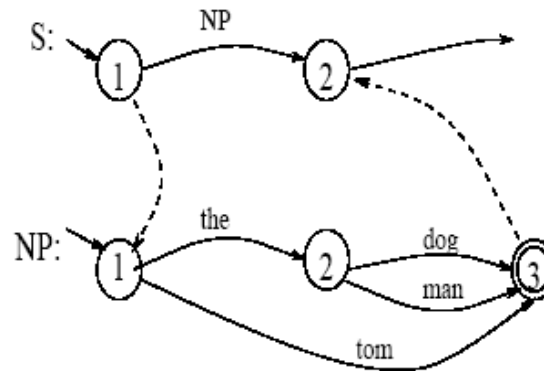
## (Recursive Transition Network)



# RTN (kiegészítések a FSA-hoz)

A szokásos FSA működtetésén túl figyelni kell:

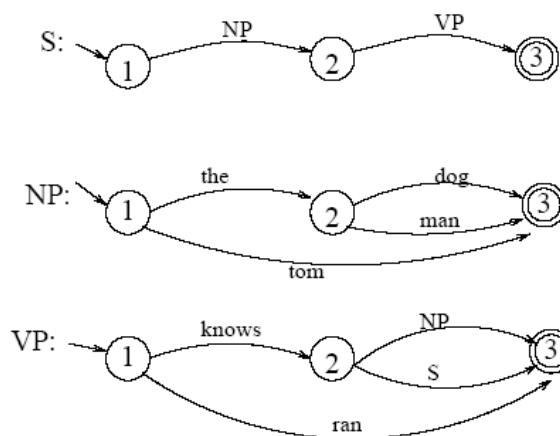
- az aktuális bemeneti pozíciót,
- az aktuális állapotot és
- hogy hova kell visszamenni:



- összegezve:  
veremkezelés kell

# RTN

(a *Thew man ran* mondat elemzésének menete)



Input	State	Return
<i>the man ran</i>	S:1	—
<i>the man ran</i>	NP:1	S:2
<i>man ran</i>	NP:2	S:2
<i>ran</i>	NP:3	S:2
<i>ran</i>	S:2	—
<i>ran</i>	VP:1	S:3
	VP:3	S:3
	S:3	—



## Az RTN összefoglalása

- ☐ Az RTN egymást hívó FSA-k hálózata: az élek címkéi közt megjelenik más FSA-k „neve”
- ☐ A FSA (a reguláris nyelvek)  $O(n)$  idő alatt elemezhető
- ☐ Az RTN valójában egy veremautomata, azaz környezet-független nyelvek elemzésére is alkalmas
- ☐ Az RTN-nel tehát  $O(n^3)$  elemzési idő garantálható



## ATN (az RTN bővítése)

### ÉLCÍMKÉK:

WRD \*, CAT \*, PUSH \*, POP, JUMP \*

### ÉRTÉKEK:

GETR, \*, QUOTE, GETF, BUILDQ \*, APPEND

### TESZTEK:

T, EQ, AND, OR, NOT

### AKCIÓK:

SETR, TO

# Az ATN-formalizmus

## (hagyományosan és XML-ben)

```
(
  (s (push np t (setr np *) (to s_np)))
  (s_np (push vp t (setr vp *) (to s_vp)))
  (s_vp (pop (buildq (s + +) np vp) t))
  (np (cat det t (setr det (buildq (det *))) (to np_det))
    (cat npr t (setr np (buildq (np (npr *)))) (to np_np)))
  (np_det(cat n t (setr np (buildq (np (det +) (noun *))) det)
    np_np))
  (np_np (jump np_np1 t (setr pps ())))
  (np_np1 (push pp t (setr pps (append (getr pps) *))(to
    np_np1))
    (jump np_end t))
  (np_end (pop (buildq (np + (pps +)) np pps) t))
  (vp (cat verb t (setr verb *) (setr pps ())) (to vp_v))
  (vp_v (push np t (setr obj *) (to vp_np)))
  (vp_np (push pp t (setr pps (append (getr pps) *))(to vp_
    (jump vp_np1 t))
  (vp_np1 (pop (buildq (vp (verb +) + +) verb obj pps) t))
  (pp (cat prep t (setr prep *) (to pp_p)))
  (pp_p (push np t (setr np *) (to pp_pp)))
  (pp_pp (pop (buildq (pp (p +) +) prep np) t))
)
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE grammar SYSTEM "gram.dtd">

<grammar>

  <network name="S">
    <state name="S.START">
      <PUSH dest="NP">
        <SETR name="NP" value="*"/>
        <JUMP dest="S.2"/>
      </PUSH>
    </state>
    <state name="S.2">
      <CAT name="v" dest="S.END"/>
    </state>
    <state name="S.END">
      <POP>
        <BUILDQ>(S(NP(+)(VP(v(+))))), NP, *</BUILDQ>
      </POP>
    </state>
  </network>

  <network name="NP">
    <state name="NP.START">
      <CAT name="det" dest="NP.2">
        <SETR name="DET" value="*"/>
      </CAT>
      <JUMP dest="NP.END"/>
    </state>
    <state name="NP.2">
      <CAT name="n" dest="NP.END"/>
    </state>
    <state name="NP.END">
      <POP>
        <BUILDQ>(det(+))(n(+)), DET, *</BUILDQ>
      </POP>
    </state>
  </network>

</grammar>
```



# A számítógépes szemantika felé

## Mire elég a szintaxis?

INPUT: Jane sold a book to Bill

MEANING:

```
(S (NP (NOUN Jane))
  (VP (VERB sold)
    (NP (DET a)
      (NOUN book))
    (PP (PREP to)
      (NP
        (NOUN Bill))))))
```



## Lehet, hogy többet érne a „jelentés”?

INPUT: Jane sold a book to Bill

MEANING:

(sell	(agent	Jane)
	(object	a book)
	(counter-agent	Bill)
	(tense	past) )

## Sok mondat - egy jelentés

Jane sold a book to Bill.

*and*

A book was sold by Jane to Bill.

*and*

Bill was sold a book by Jane.

*and*

A book was sold to Bill by Jane.



(sell	(agent	Jane)
	(object	a book)
	(counter-agent	Bill)
	(tense	past) )

## A szintaktikai szerepek és a jelentés viszonya

(sell (agent SUBJECT)  
(object DIRECT OBJECT)  
(counter-agent  
INDIRECT OBJECT  
or  
PREP. OBJECT "to"))



## Hasonló mondat - különböző jelentés

Mom baked for 3 hours

The pie baked for 3 hours

(baked (agent Mom)  
(duration 3 hours))

(baked (object pie)  
(duration 3 hours))



# Többértelműség

## A többértelműség megjelenései

### A szavak szintjén

- ☐ Homonímia (azonos alakúság): *vár<sup>1</sup>, vár<sup>2</sup>*
- ☐ Poliszémia (többértelműség): *vírus<sup>1</sup>, vírus<sup>2</sup>*
- ☐ Szinonímia (rokon értelműség): *humoros, vicces, tréfás, mókás, poénos, mulatságos, kacagtató, mulattató, ...*

### A szóalaktan szintjén

- ☐ Homonímia (azonos alakúság): *nemzet+é<sup>1</sup>+t, nemzet+é<sup>2</sup>+t*
- ☐ Poliszémia (többértelműség): *úr+nak<sup>1</sup> (szolgál), úr+nak<sup>2</sup> (születik)*

### A mondat szintjén

- ☐ Szerkezeti többértelműségek: *Láttalak a teraszon ülve.*

### A szöveg szintjén

- ☐ A közlés szándéka, a kommunikációs helyzet, a beszélő és a hallgató viszonya befolyásolja az üzenet jelentését:  
*De szeretlek!*

## Mire jó a jelentéstani elemzés?

- ❑ A többértelműség egyik fontos forrása a jelentések nem egyértelmű volta
- ❑ A mondszerkezet ismerete ennek eldöntéséhez még nem nyújt elég információt
- ❑ Az üzenet kibocsátója szempontjából egyértelmű üzenet tele lehet jelentésbeli többértelműségekkel, melyeket az üzenet vevőjének kell feloldani
- ❑ Ilyen viszont nincs:  
*Miért fejt több ember keresztretjtvényt, mint szenet?*  
*(... és tehenet?)*
- ❑ A legfontosabb célok:  
(1) feloldani a jelentéstani többértelműségeket, és  
(2) reprezentálni a mögöttes tartalmat

## A nem egyértelmű jelentések kezelése

- ❑ Az agyunk optimalizál, egyensúlyt tart, ui.
  - ha sok a többértelműség, akkor kisebb lesz a lexikon, ezért kevesebb memória kell, de bonyolultabb lesz a feldolgozás
  - ha kevesebb a többértelműség, akkor nagyobb lesz a lexikon, ezért lassabb lesz a tanulás, de könnyebb lesz az értelmezés
- ❑ Az emberi nyelvekben gyakori a többértelműség, mégis ritkán okoz gondot egy-egy fogadott jelsorozat értelmezése
- ❑ A számítógép számára szinte teljesen kezelhetetlen a többértelműség - ezt kell megoldani!



# Jelentés-egyértelműsítés

## Jelentés-egyértelműsítés (WSD)

### ***Word sense disambiguation:***

egy szóalak jelentésének kiválasztása, egy előre megadott (lehetséges jelentés-) halmazból

- a lehetséges jelentések szótárból
- osztályozási modellek alkalmazhatók
- előre egyértelműsített példák szükségesek

### ***Word sense discrimination:***

egy adott szóalak különböző használati eseteit (=jelentéseit) elkülöníteni, anélkül, hogy a lehetséges jelentéseket kívülről megadnánk (és így címkézett példáink sem lehetnek)

- felügyelet nélküli statisztikai modellek

## A WSD két fő iránya

### Minden szövegbeli szóra:

- a cél az összes szóalak egyértelműsítése, folyó szövegben
- nagyon kevés erőforrás áll rendelkezésre
- olyan mintákat kell tanulni, melyek függetlenek az adott szóalaktól
- nincs kielégítő megoldás még
- gyakorlatban ez lenne jól használható

### A szöveg egyes szavaira:

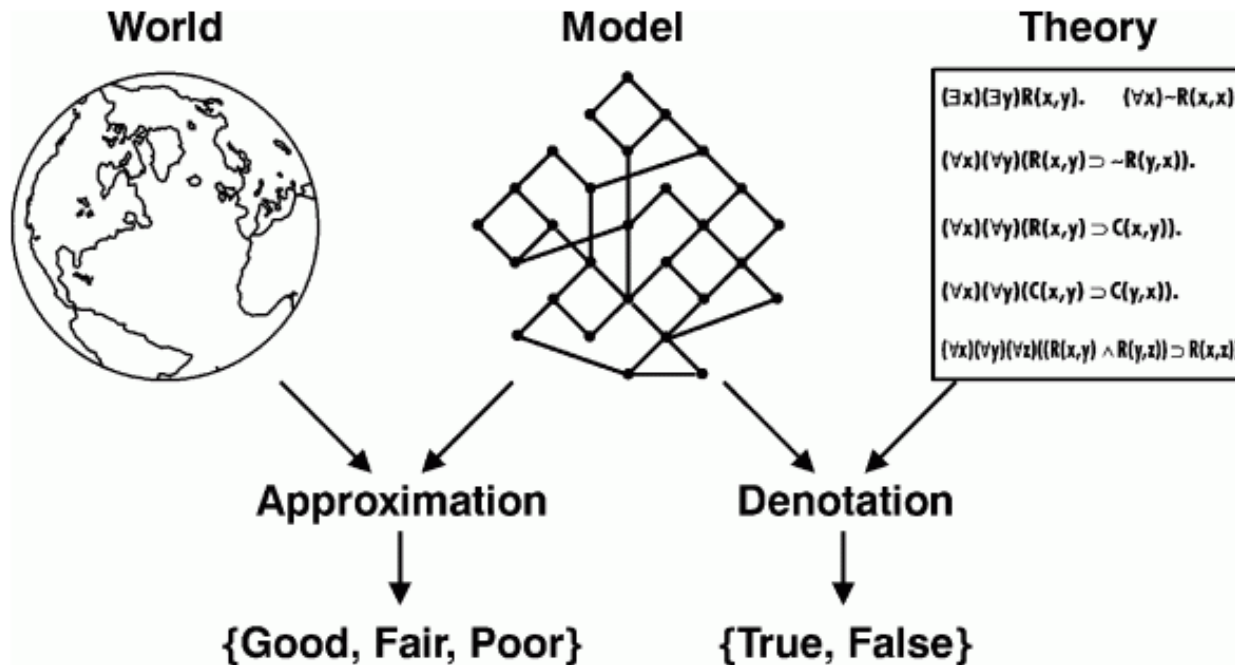
- a cél bizonyos szóalakok egyértelműsítése, minden szóalakra önálló modell
- több erőforrás van, és olcsóbban előállítható 1-1 új szóalakra
- az adott szóra jellemző mintákat tanulhatunk, könnyebb feladat
- tűrhető (nem jó!) megoldások vannak
- gyakorlatban ritkán használható





# Jelentésábrázolás

## Világ - modell - elmélet



## Jelentés vagy világismeret?

- (1) *Péter megvette a könyvet.*
- (2) *Péter megvette Az ember tragédiáját.*
- (3) *Vett könyvet Péter?*

A világismeretet tárolni kell:  
mi micsoda, és  
milyen viszonyban van a többi ismert dologgal?

## Világismeret-reprezentáció

*Péter megvette Az ember tragédiáját.*

(elad (agent ember17)

(object AET3791)

(recipient ember35)

(tense múlt))

AET3791: „*Az ember tragédiája*” könyv egy példánya

ember35 neve: Péter

ember17: (jelenleg) nem ismerjük