Macau, March 14th

XI UNL School

Day #4

Day #4

- UNLization
- NLization

UNLization (cont’d)

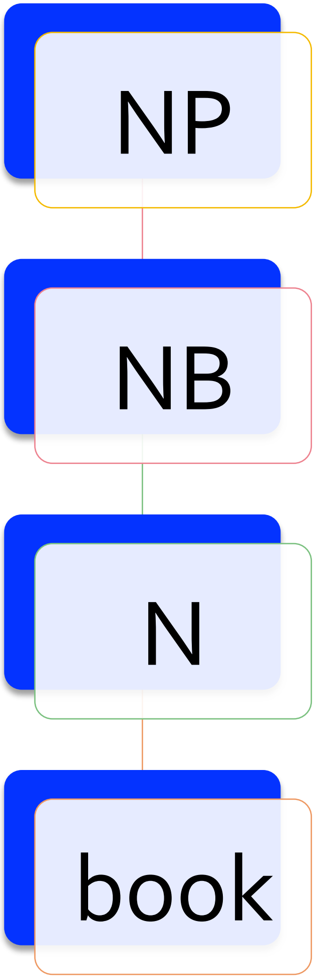
Subcorpus

- 6: book
- 16: my book
- 26: the book on the table
- 36: the train from Paris
- 46: John and Mary
- 56: beautiful car
- 66: the new beautiful expensive car of John
- 76: he arrived
- 86: he killed Mary
- 96: it is beautiful

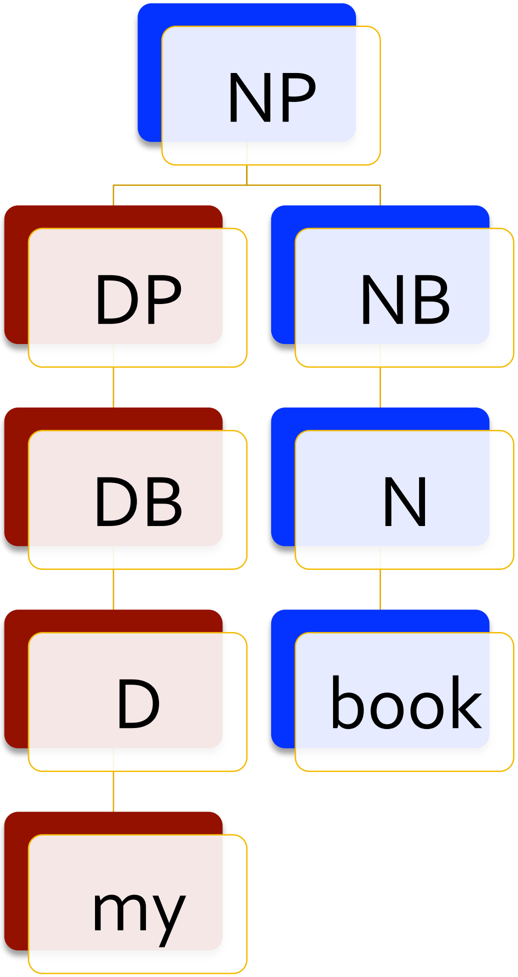
Exercise 12a

Build the x-bar trees for the sentences of the subcorpus

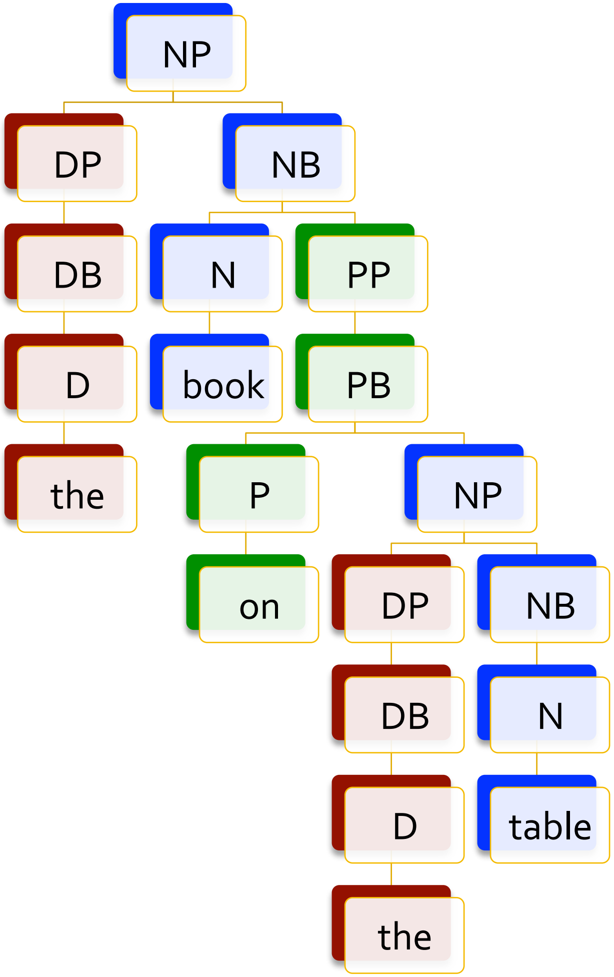
6. book



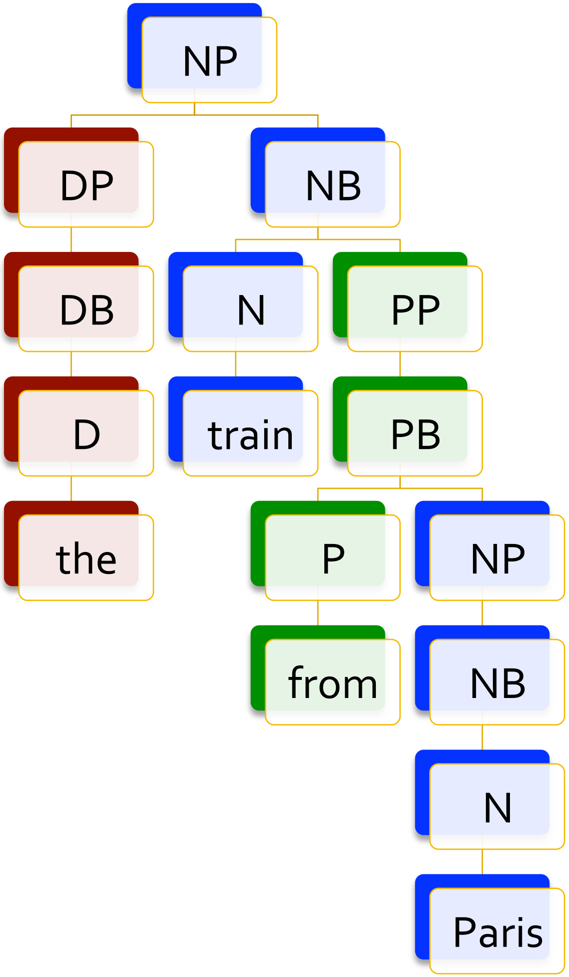
16. my book



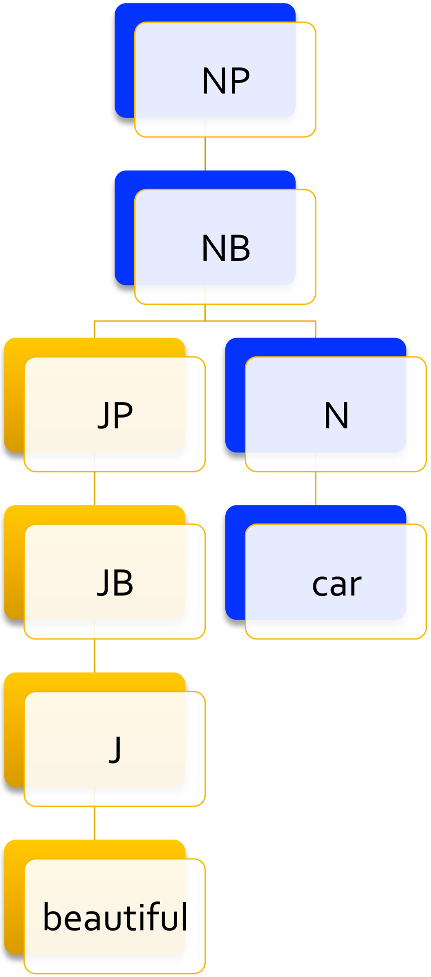
26. the book on the table



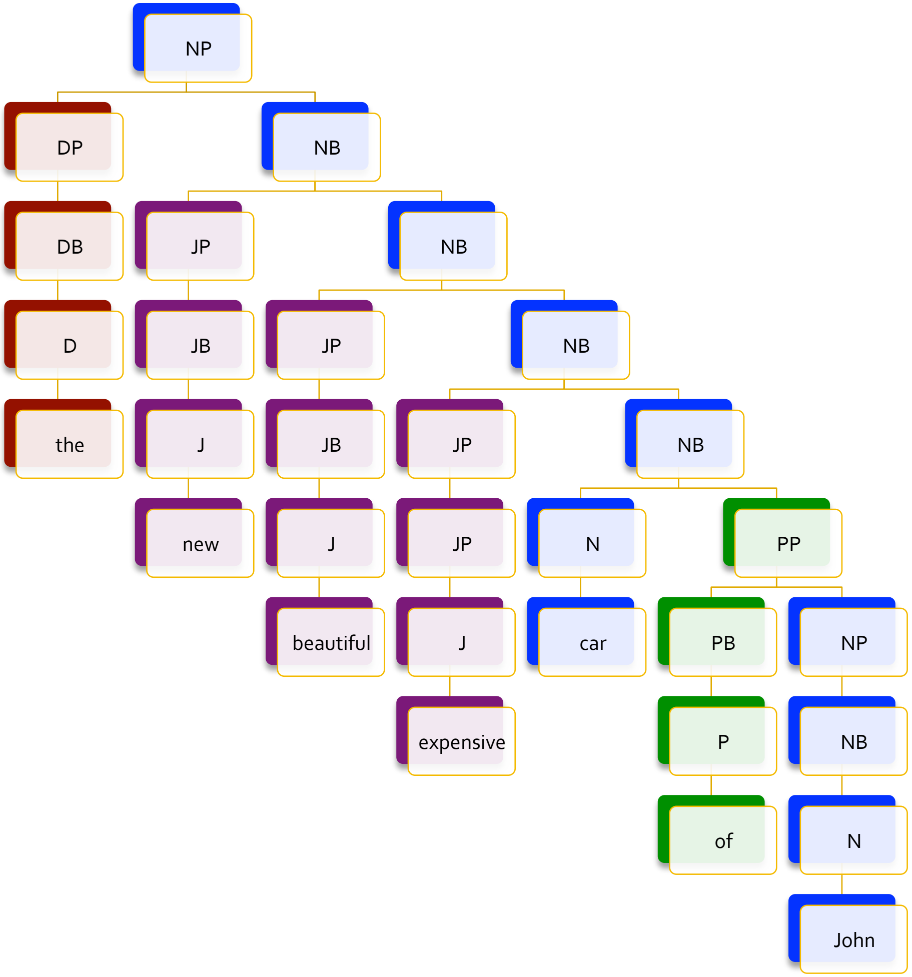
36. the train from Paris



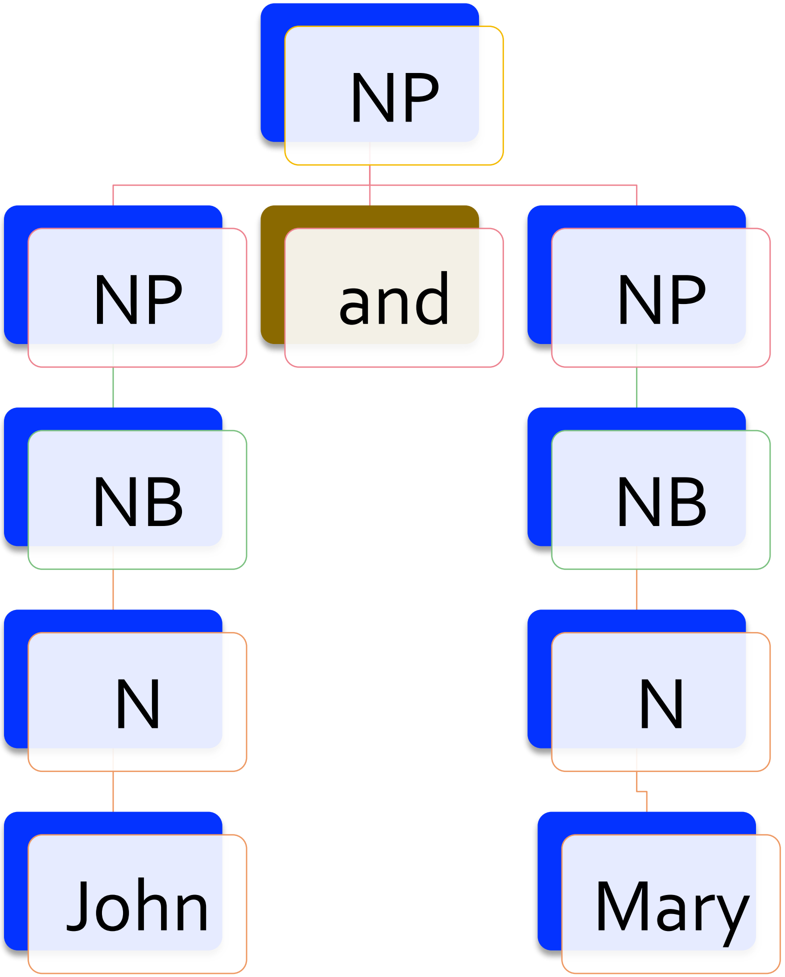
46. beautiful car



56. the new beautiful expensive car of John



66. John and Mary



NP

NP

and

NP

NB

NB

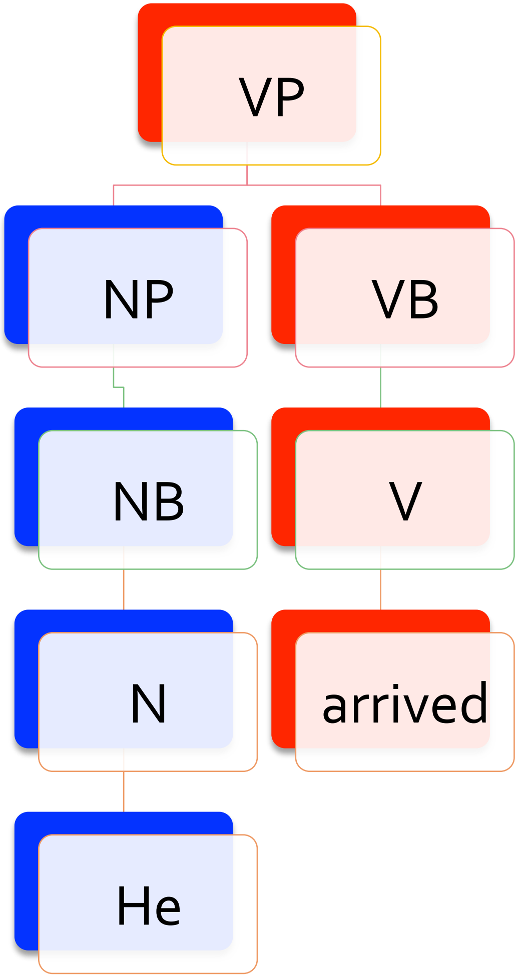
N

N

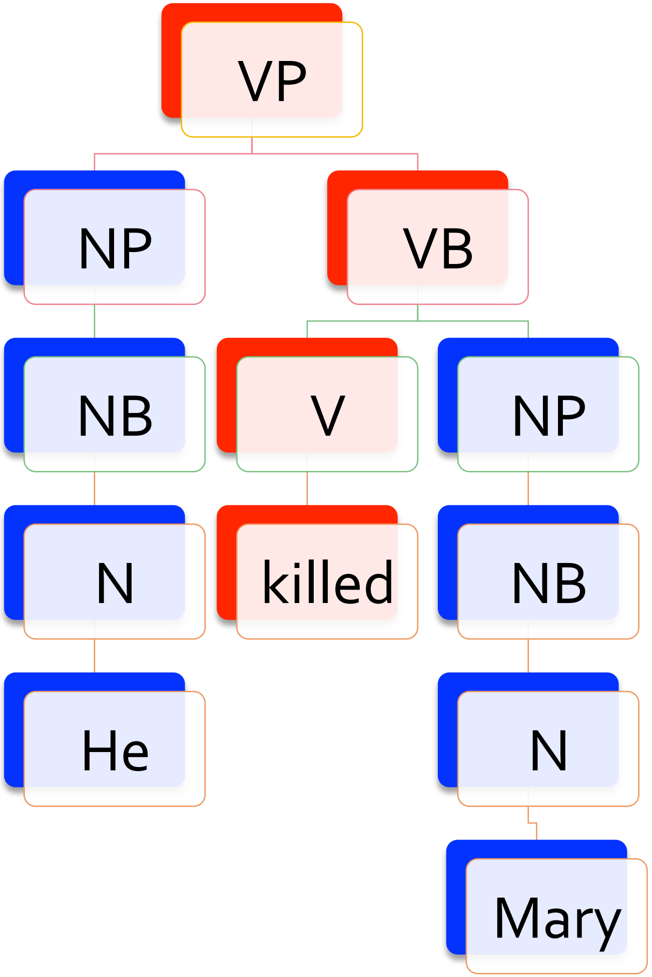
John

Mary

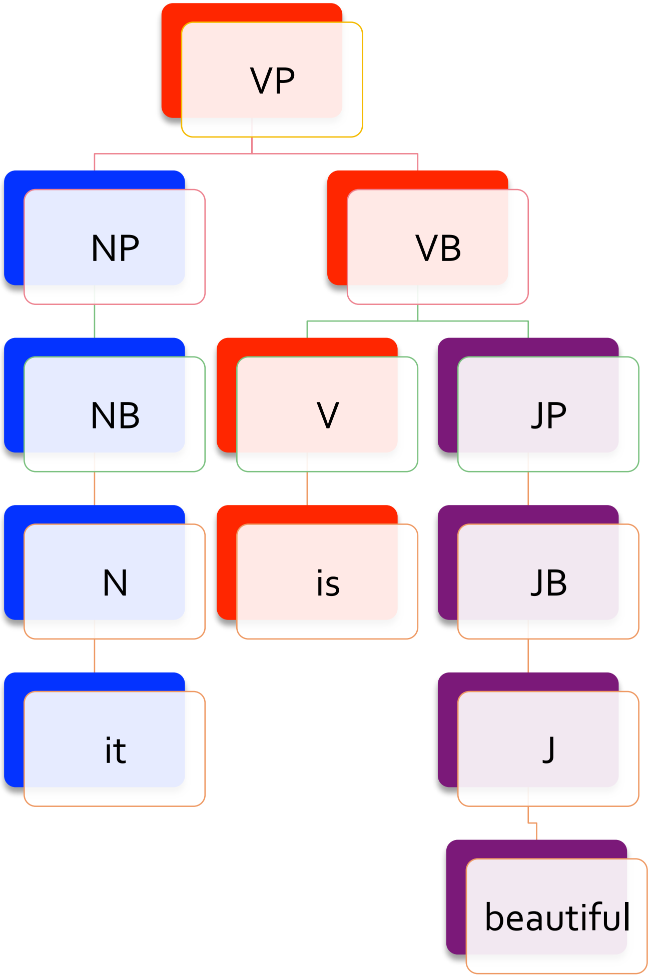
76. He arrived



86. He killed Mary



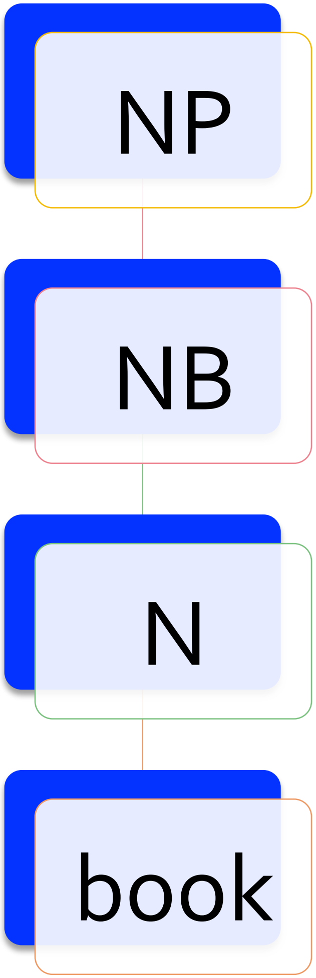
96. It is beautiful



Exercise 12b (15 min)

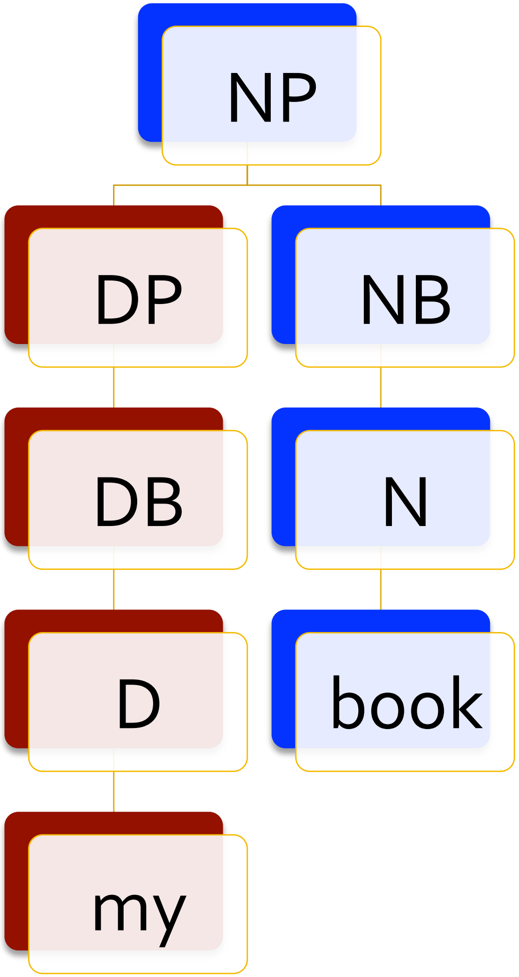
Extract the rules from the trees

6. book



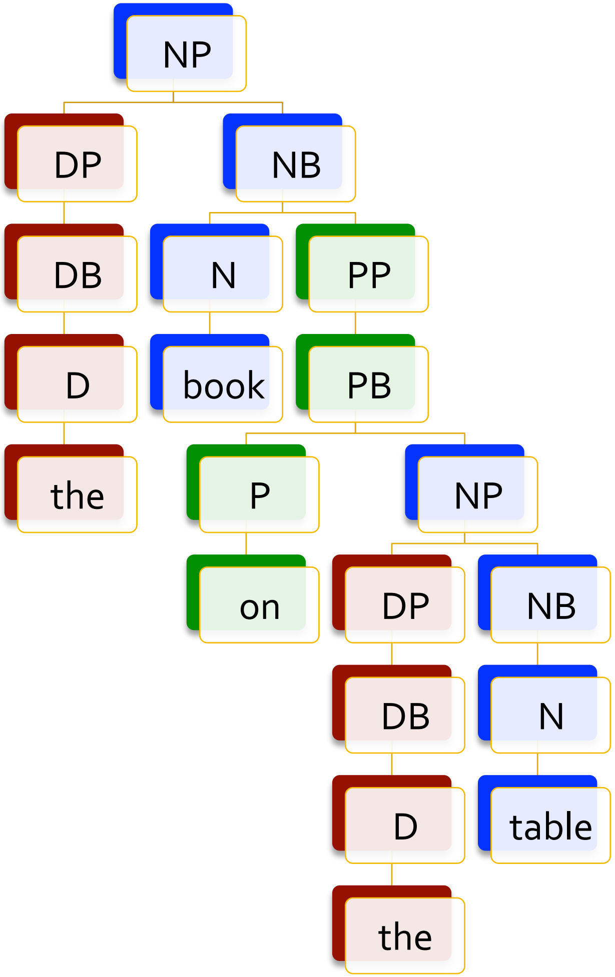
(NP):=(NB);
(NB):=(N);

16. my book



(NP):=(DP)(NB);
(DP):=(DB);
(DB):=(D);
(NB):=(N);

26. the book on the table



(NP):=(DP)(NB);
(DP):=(DB);
(DB):=(D);
(NB):=(N)(PP);
(PP):=(PB);
(PB):=(P)(NP);
(NB):=(N);

Exercise 12c (15 min)

Invert and order the rules

Example

BEFORE

- (NP):=(DP)(NB);
- (DP):=(DB);
- (DB):=(D);
- (NB):=(N)(PP);
- (PP):=(PB);
- (PB):=(P)(NP);
- (NB):=(N);

AFTER

- (DP)(NB):=(NP);
- (P)(NP):=(PB);
- (N)(PP):=(NB);
- (DB):=(DP);
- (PB):=(PP);
- (D):=(DB);
- (N) :=(NB);

Exercise 12d (15 min)

Create the parsing module

Example

BEFORE

- (DP)(NB):=(NP);
- (P)(NP):=(PB);
- (N)(PP):=(NB);
- (DB):=(DP);
- (PB):=(PP);
- (D):=(DB);
- (N) :=(NB);

AFTER

- (DP,%dp)(NB,%nb):=(NP(%nb;%dp),+XP=NP,+LEX=N);
- (P,%p)(NP,%np):=(PB(%p;%np),+XB=PB,+LEX=P);
- (N,%n)(PP,%pp):=(NB(%n;%pp),+XB=NB,+LEX=N);
- (DB,^DP):=(+DP);
- (PB,^PP):=(+PP);
- (D,^DB):=(+DB);
- (N,^NB) :=(+NB);

Exercise 12e (15 min)

Implement the parsing module in IAN
(i.e., upload the grammar to IAN)

Exercise 12f (15 min)

Implement the parsing module in IAN (i.e., upload the grammar to IAN)

Exercise 12g (15 min)

Test the results

Exercise #13 (Dearborization and Interpretation) 30 min

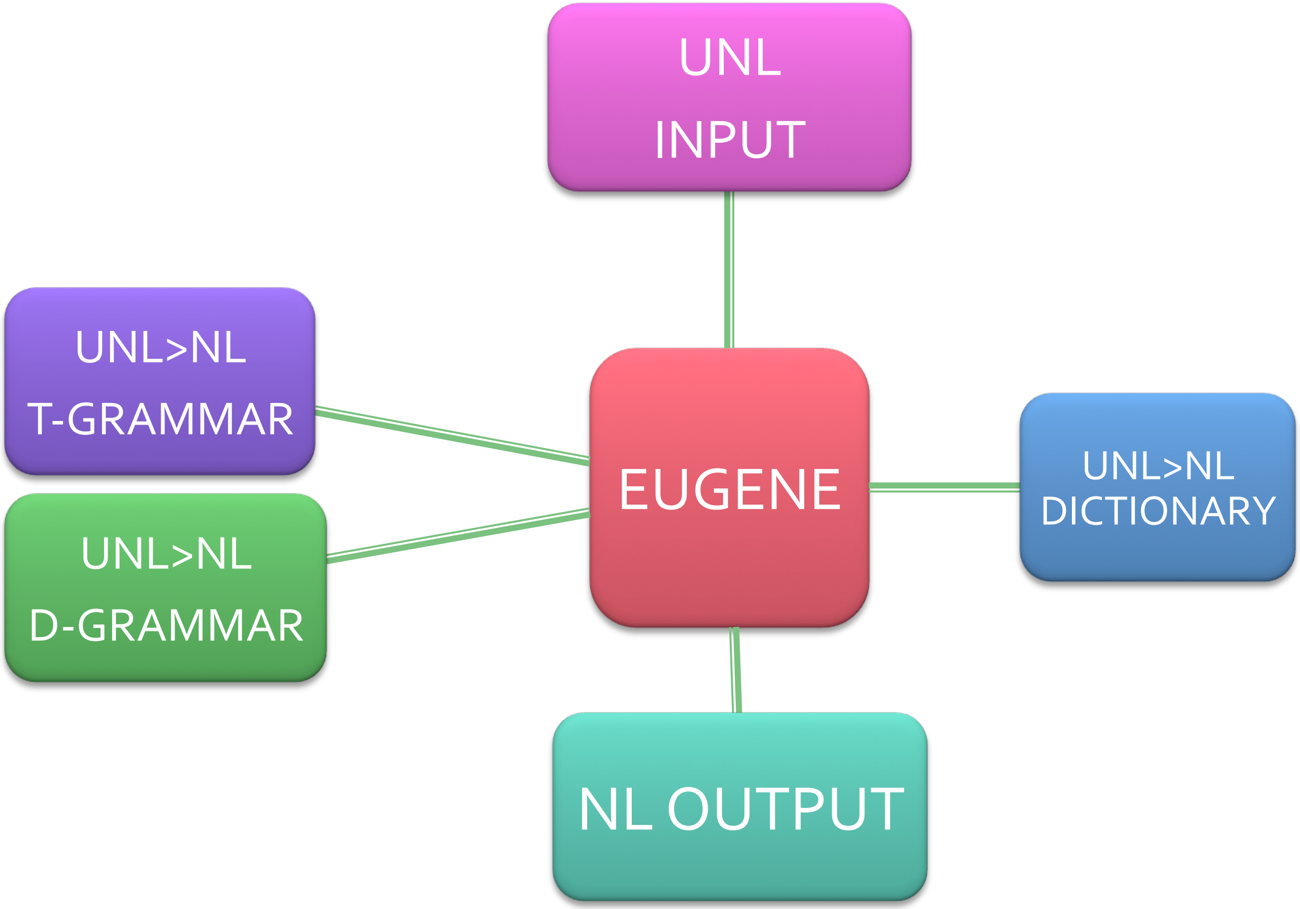
- Dearborize and interpret the subcorpus*.
- Analyze the results and provide the necessary changes.

*In order to dearborize and interpret the subcorpus, enable the default grammar.

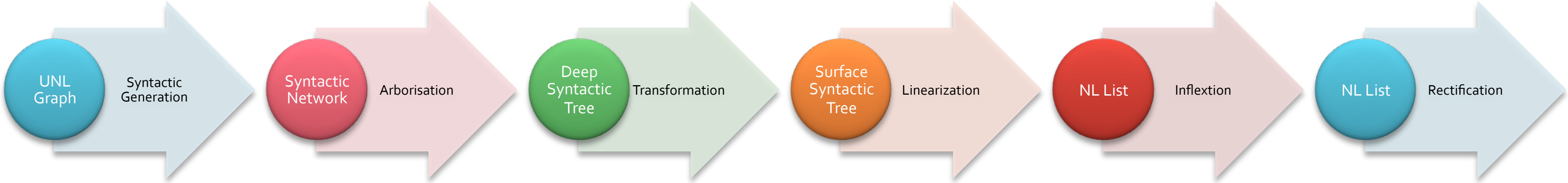
NLization

EUGENE

(dEep-to-sUrface GENErator)



NLization

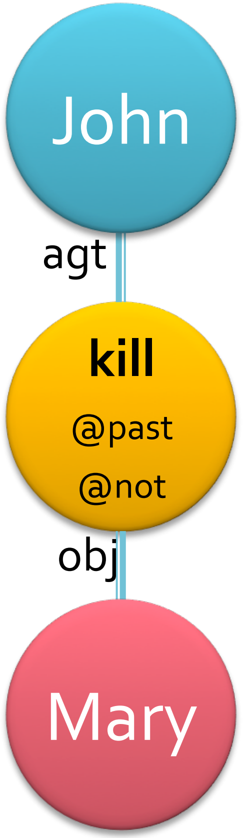

UNL Graph
Syntactic Generation
Syntactic Network
Arborisation
Deep Syntactic Tree
Transformation
Surface Syntactic Tree
Linearization
NL List
Inflextion
NL List
Rectification

Syntactic Generation

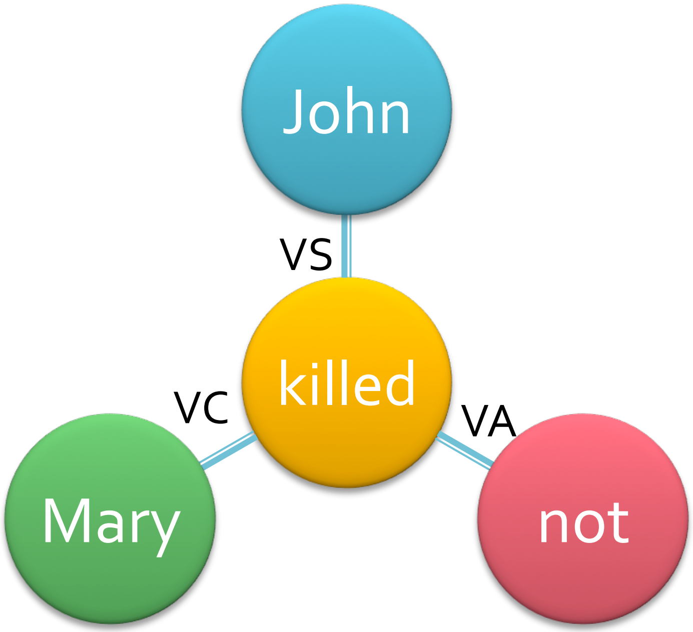
Syntactic Generation

semantic network > syntactic network

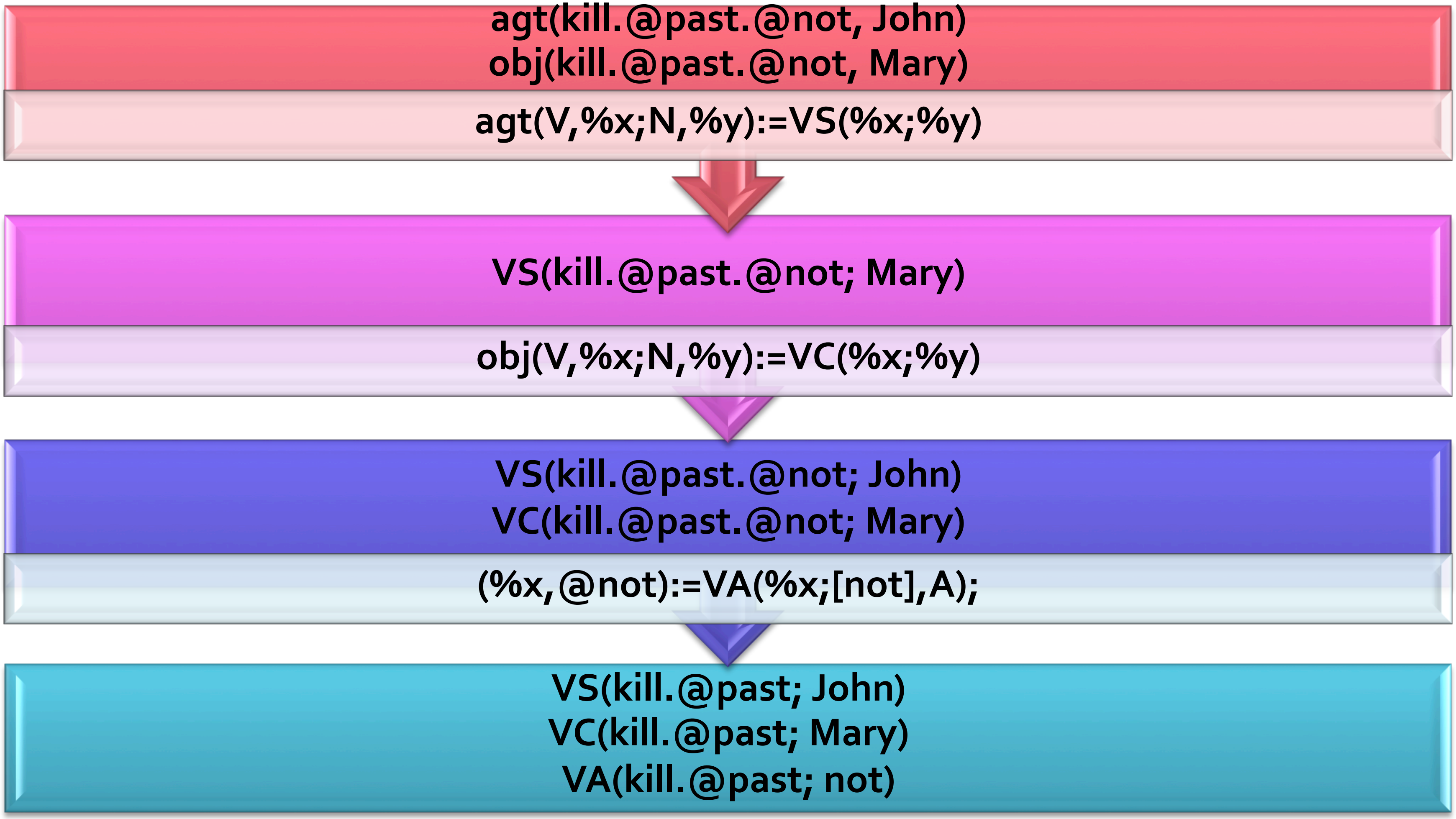
SEMANTIC NETWORK



SYNTACTIC NETWORK



Example



agt(kill.@past.@not, John)
obj(kill.@past.@not, Mary)

agt(V,%x;N,%y):=VS(%x;%y)

VS(kill.@past.@not; Mary)

obj(V,%x;N,%y):=VC(%x;%y)

VS(kill.@past.@not; John)
VC(kill.@past.@not; Mary)

(%x,@not):=VA(%x;[not],A);

VS(kill.@past; John)
VC(kill.@past; Mary)
VA(kill.@past; not)

Further examples

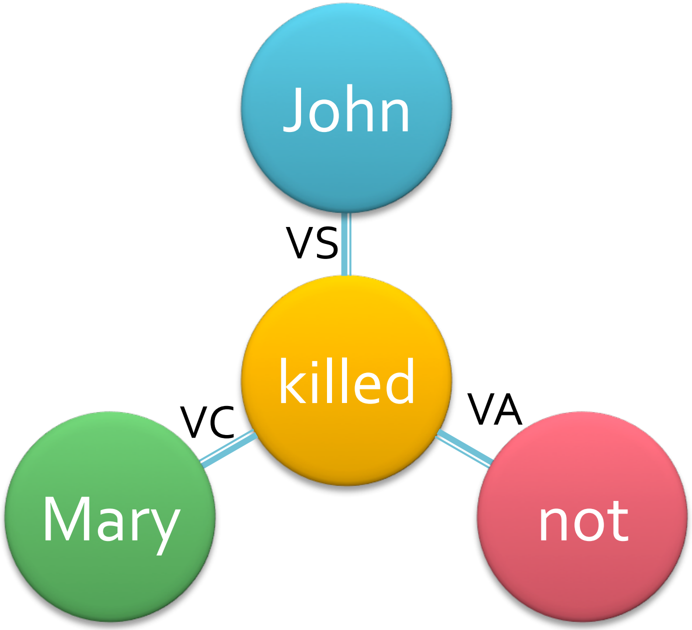
- Relations
  - agt(%x,V;%y,N):=VS(%x,PER=%y;%y,+NOM);
  - tim(%x,V;%y,N):=VA(%x;PC([in];%y));
- Attributes
  - (%x,N,@def):=NS(%x,-@def;%y,?[[]],?LEX=D,?POS=ART);
  - (%x,@future):=IC(%y,[will],LEX=V,POS=AUX,PER=%x;%x,-@future,+INF,-PER);

Arborisation

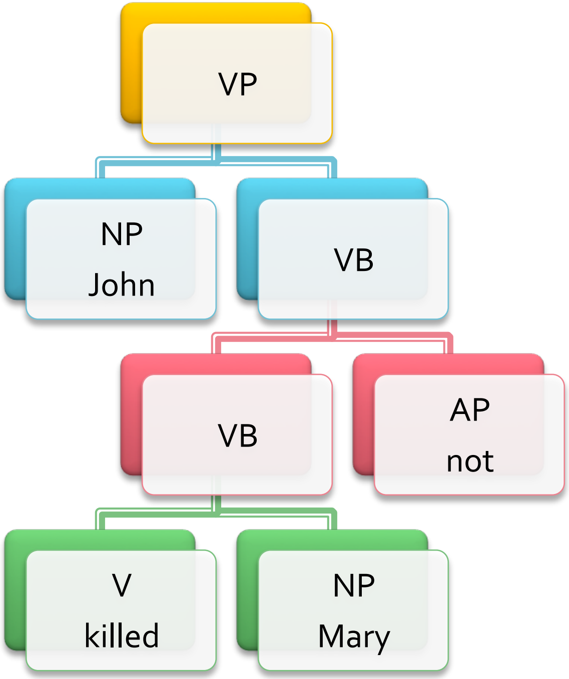
Arborisation

syntactic network > syntactic tree

NETWORK STRUCTURE



TREE STRUCTURE

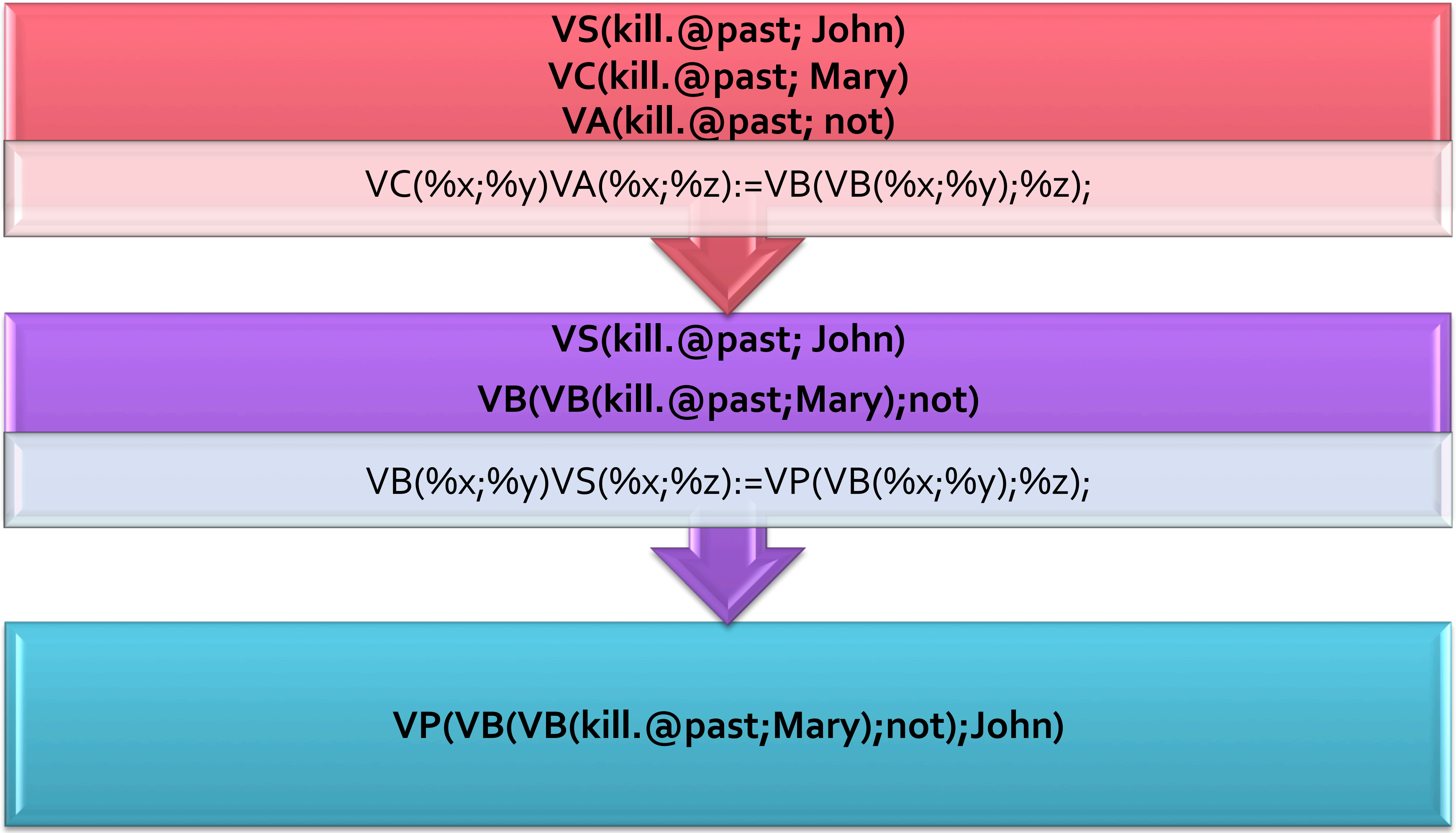


Arborisation

Example of rules

- VC(%x;%y)VC(%x;%z):=VB(VB(%x;%y);%z);
- VC(%x;%y)VA(%x;%z):=VB(VB(%x;%y);%z);
- VA(%x;%y)VA(%x;%z):=VB(VB(%x;%y);%z);
- VB(%x;%y)VC(%x;%z):=VB(VB(%x;%y);%z);
- VB(%x;%y)VA(%x;%z):=VB(VB(%x;%y);%z);
- VC(%x;%y):=VB(%x;%y);
- VA(%x;%y):=VB(%x;%y);
- VB(%x;%y)VS(%x;%z):=VP(VB(%x;%y);%z);
- VS(%x;%y):=VP(%x;%y);

Example



VS(kill.@past; John)
VC(kill.@past; Mary)
VA(kill.@past; not)

VC(%x;%y)VA(%x;%z):=VB(VB(%x;%y);%z);

VS(kill.@past; John)

VB(VB(kill.@past;Mary);not)

VB(%x;%y)VS(%x;%z):=VP(VB(%x;%y);%z);

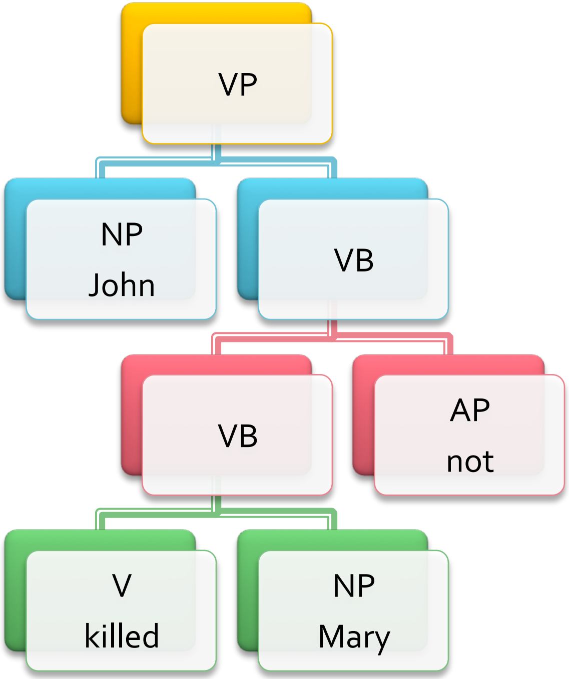
VP(VB(VB(kill.@past;Mary);not);John)

Transformation

Transformation

syntactic tree (deep) > syntactic tree (surface)

DEEP STRUCTURE



VP

NP John

VB

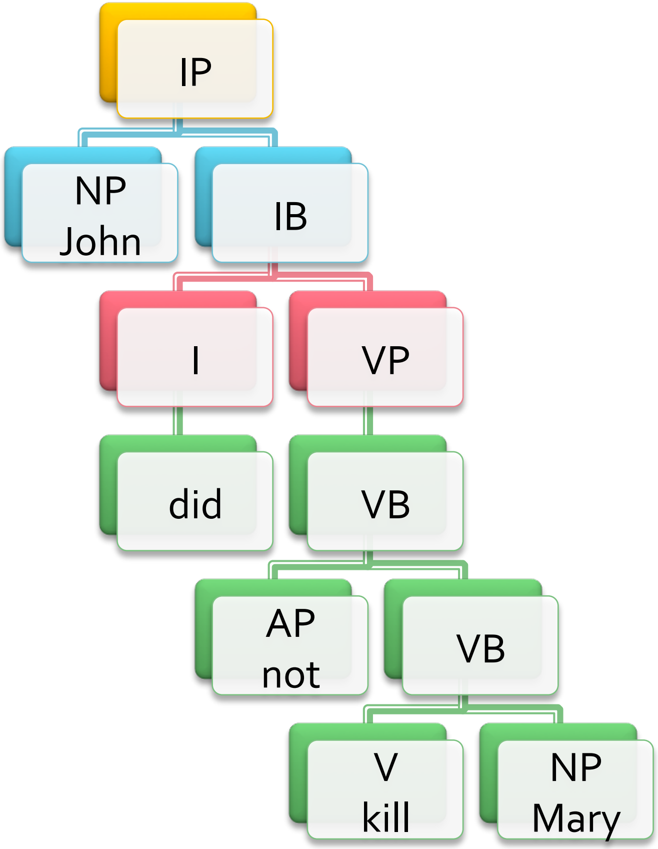
VB

AP not

V killed

NP Mary

SURFACE STRUCTURE



IP

NP John

IB

I

VP

did

VB

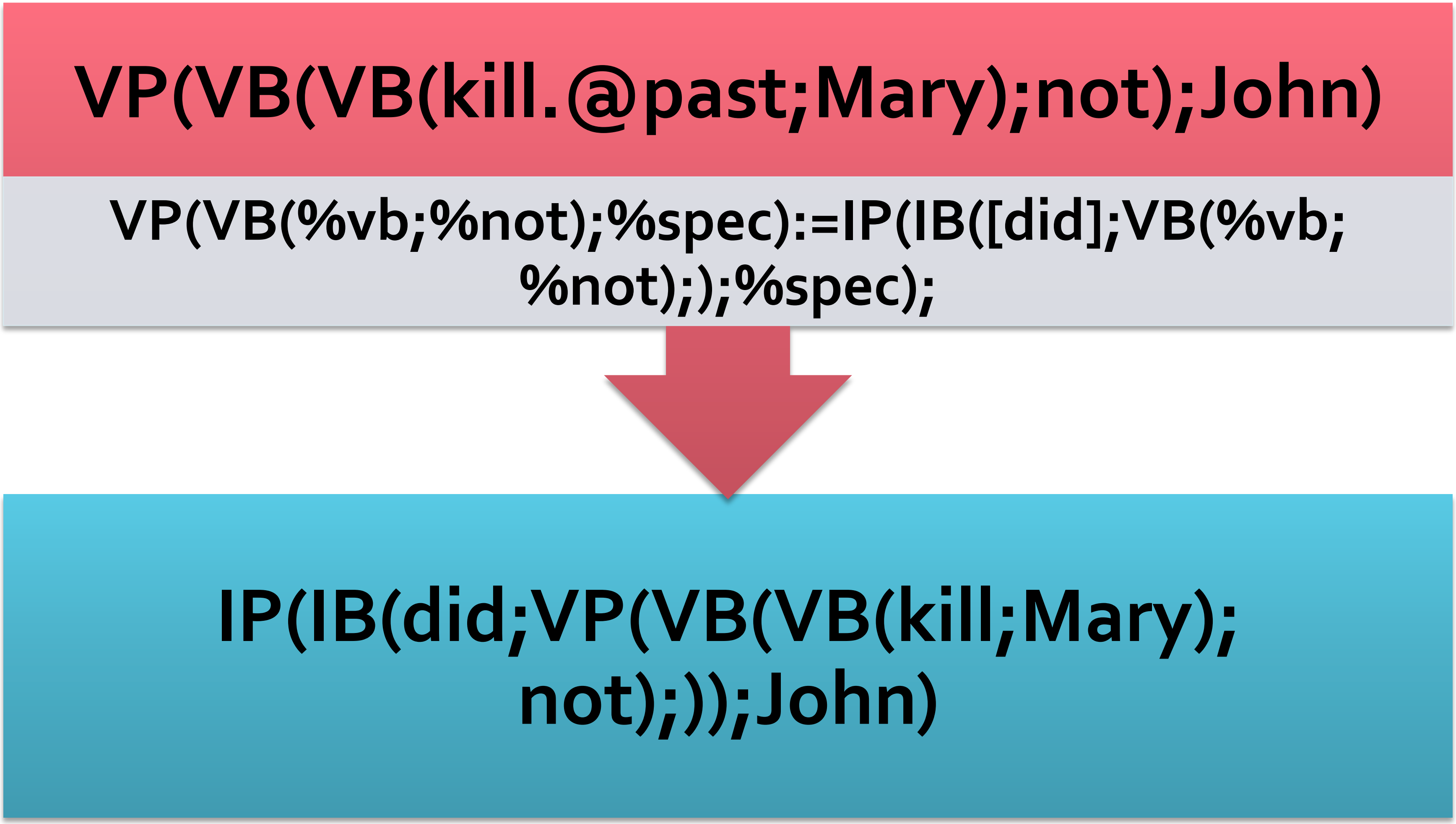
AP not

VB

V kill

NP Mary

Example



VP(VB(VB(kill.@past;Mary);not);John)

VP(VB(%vb;%not);%spec):=IP(IB([did];VB(%vb; %not););%spec);

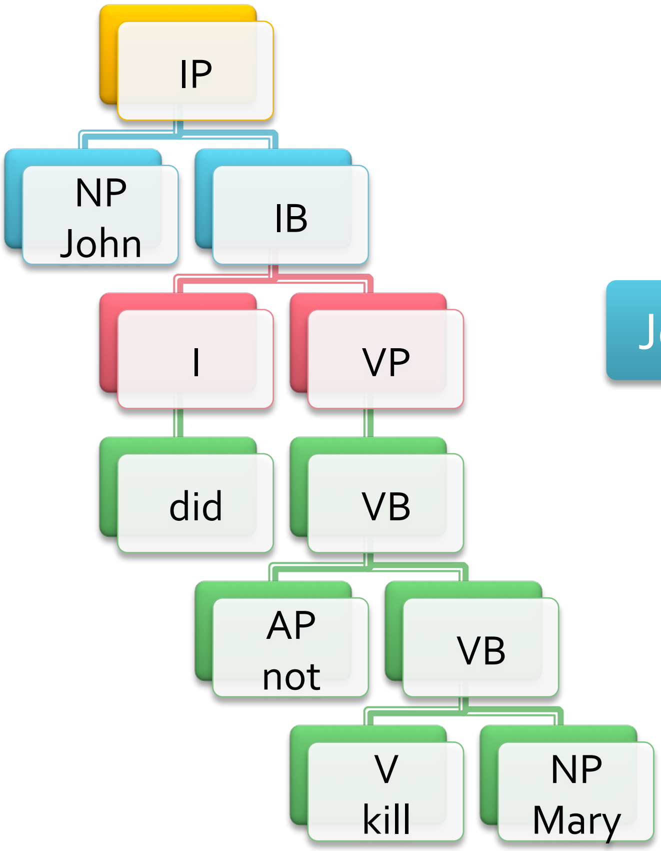
IP(IB(did;VP(VB(VB(kill;Mary); not););John)

Linearization

Linearization

tree > list

TREE STRUCTURE



IP

NP John

IB

I

VP

did

VB

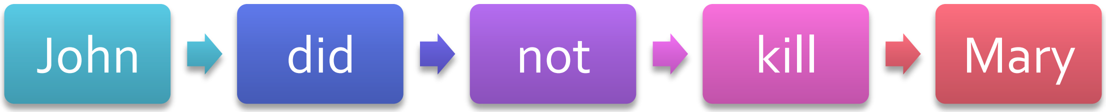
AP not

VB

V kill

NP Mary

LIST STRUCTURE



John

did

not

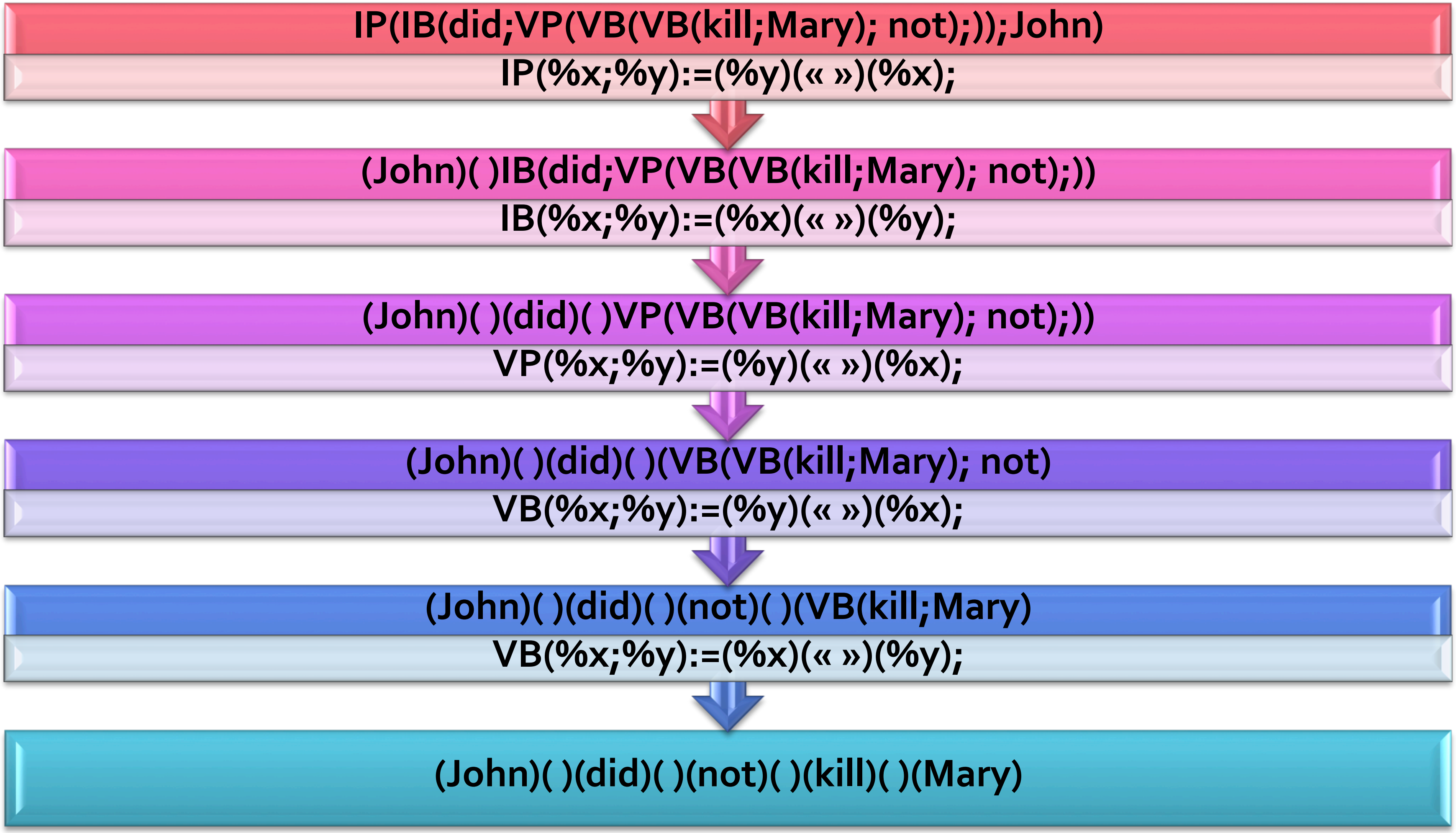
kill

Mary

Linearization
Examples of rules

- PP(%x;%y,BEF):=(%y)(" ")(%x);
- PP(%x;%y,IBEF):=(%y)(%x);
- PP(%x;%y,AFT):=(%x)(" ")(%y);
- PP(%x;%y,IAFT):=(%x)(%y);
- PP(%x;%y):=(%y)(" ")(%x);

Example



IP(IB(did;VP(VB(VB(kill;Mary); not);));John)

IP(%x;%y):=(%y)(« »)(%x);

(John)( )IB(did;VP(VB(VB(kill;Mary); not);))

IB(%x;%y):=(%x)(« »)(%y);

(John)( )(did)( )VP(VB(VB(kill;Mary); not);))

VP(%x;%y):=(%y)(« »)(%x);

(John)( )(did)( )(VB(VB(kill;Mary); not)

VB(%x;%y):=(%y)(« »)(%x);

(John)( )(did)( )(not)( )(VB(kill;Mary)

VB(%x;%y):=(%x)(« »)(%y);

(John)( )(did)( )(not)( )(kill)( )(Mary)

Inflection

Morphological generation

- Inflectional Grammar
  - (%x,M2):=(%x,-M2,+FLX(SNG:=0>""; PLR:=0>"s";));
- Triggering inflectional rules
  - (%x,^inflected,FLX):=(%x,!FLX,+inflected);

Rectification

- Eliminate excessive blank spaces
  - (BLK,%x)(BLK,%y):=(%x);
- Punctuation
  - (^PUT,%x)(STAIL,%y):=(%x)([.])(%y);
- Contraction
  - ([de])(BLK)([le]):=([du]);
  - ([a])(BLK,%x)("/[aeiou].*",%y):=([an])(%x)(%y);

Exercises

Exercise #14 (30 min)

- Download, from www.unlweb.net/wiki/UCA1:
  - The UNL-ENG dictionary
  - The default dictionary
  - The UNL-ENG t-grammar
  - The UNL-ENG d-grammar
  - The UNL-NL default grammar
- Upload these files to EUGENE
- Run EUGENE and analyze the results

Exercise #15

- Develop the resources necessary to generate the sentences -6 (6,16,26,36,...) from the corpus UCA1.