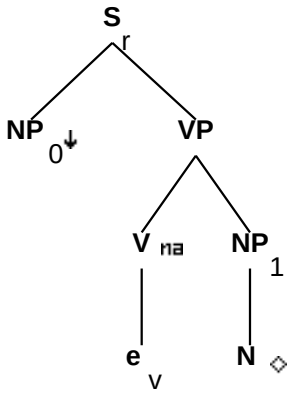
Family "Tnx0N1"

March 5, 2008

1 Tree "alphanx0N1"

1.1 graphe



1.2 comments

```
Tree for predicational NPs:

  John is an author.
  I consider John an author.

<mainv> feature prevents
adjunction of 'do' to auxiliary-headed S
```

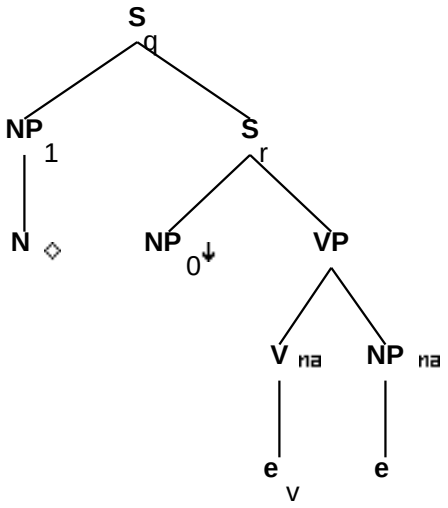
1.3 features

```
S_r.b:<extracted> = -
S_r.b:<inv> = -

S_r.b:<assign-comp> = VP.t:<assign-comp>


S_r.b:<mode> = VP.t:<mode>
S_r.b:<mainv> = VP.t:<mainv>
S_r.b:<comp> = nil
```

```
S_r.b:<tense> = VP.t:<tense>
NP_0:<agr> = S_r.b:<agr>
NP_0:<case> = S_r.b:<assign-case>
NP_0:<wh> = -
NP_1:<case> = acc
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
S_r.b:<passive> = VP.t:<passive>
VP.t:<passive> = -
VP.b:<mode> = nom
VP.b:<assign-case> = acc
VP.b:<compar> = -
N:<agr> = NP_1.b:<agr>
NP_1.b:<case> = N.t:<case>
NP_1.b:<wh> = N.t:<wh>
NP_1.b:<pron> = N.t:<pron>
NP_1.b:<compar> = N.t:<compar>
N.t:<compar> = -
N.t:<const> = NP_1.b:<const>
N.t:<gen> = NP_1.b:<gen>
N.t:<definite> = NP_1.b:<definite>
N.t:<quan> = NP_1.b:<quan>
N.t:<card> = NP_1.b:<card>
N.t:<decreas> = NP_1.b:<decreas>
S_r.b:<control> = NP_0.t:<control>
```

2 Tree "alphaW1nx0N1"

2.1 graphe



2.2 comments

No original comments.

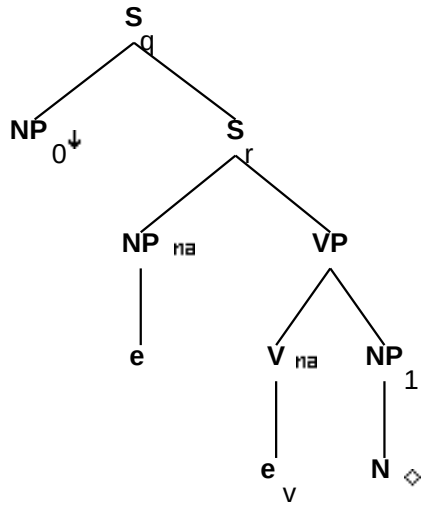
2.3 features

```
S_q.b:<extracted> = +

S_q.b:<inv> = S_r.t:<inv>
S_q.b:<inv> = S_q.b:<invlink>
NP_1.t:<wh> = S_q.b:<wh>
NP_1.b:<wh> = N.t:<wh>
S_r.t:<comp> = nil
S_r.b:<assign-comp> = VP.t:<assign-comp>



S_q.b:<mode> = S_r.t:<mode>
S_q.b:<comp> = nil
S_r.b:<mode> = VP.t:<mode>
S_r.b:<comp> = nil
S_r.b:<inv> = -
NP_0:<agr> = S_r.b:<agr>
NP_0:<case> = S_r.b:<assign-case>
NP_1.b:<compar> = N.t:<compar>
NP_1.b:<agr> = N.t:<agr>
NP_1.b:<case> = N.t:<case>
NP_1.b:<pron> = N.t:<pron>
N.t:<compar> = -
N.t:<const> = NP_1.b:<const>
N.t:<gen> = NP_1.b:<gen>
N.t:<definite> = NP_1.b:<definite>
N.t:<quan> = NP_1.b:<quan>
N.t:<card> = NP_1.b:<card>
N.t:<decreas> = NP_1.b:<decreas>
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
S_r.b:<control> = NP_0.t:<control>
NP.t:<trace> = NP_1:<trace>
NP.t:<agr> = NP_1:<agr>
NP.t:<case> = NP_1:<case>
NP.t:<wh> = NP_1:<wh>
NP.t:<case> = acc
S_r.b:<tense> = VP.t:<tense>
S_r.b:<mainv> = VP.t:<mainv>
S_r.b:<passive> = VP.t:<passive>
VP.t:<passive> = -
VP.b:<mode> = nom
VP.b:<assign-case> = acc
VP.b:<compar> = -
```

S_r.t:<conj> = nil

3 Tree "alphaW0nx0N1"

3.1 graphe



3.2 comments

```
Wh on the subject
 'Who loves Mary'
 'Who has loved Mary'

check the agr equation on  NP0

minidemo:

 The value for tense is passed up from the V node.
 this should be implemented everywhere: I have
 changed only this tree and the regular non-
 question matrix tree
```

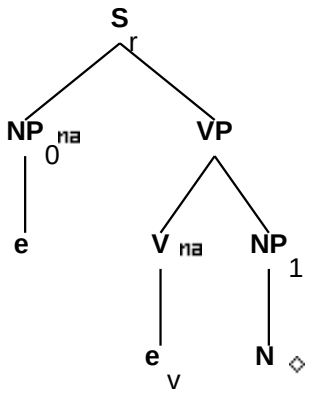
3.3 features

```
S_q.b:<extracted> = +

S_q.b:<inv> = S_r.t:<inv>
S_q.b:<wh> = NP_0.t:<wh>
S_r.t:<comp> = nil
S_r.b:<assign-comp> = VP.t:<assign-comp>
```

```
S_q.b:<comp> = nil
S_q.b:<mode> = S_r.t:<mode>
S_r.b:<mode> = VP.t:<mode>
S_r.b:<comp> = nil
S_r.b:<tense> = VP.t:<tense>
S_r.b:<inv> = -
NP:<trace> = NP_0:<trace>
NP:<agr> = NP_0:<agr>
NP:<case> = NP_0:<case>
NP:<wh> = NP_0:<wh>
NP_0:<wh> = +
NP.t:<agr> = S_r.b:<agr>
NP.t:<case> = S_r.b:<assign-case>
NP_1.b:<compar> = N.t:<compar>
N.t:<compar> = -
N.t:<const> = NP_1.b:<const>
N.t:<gen> = NP_1.b:<gen>
N.t:<definite> = NP_1.b:<definite>
N.t:<quan> = NP_1.b:<quan>
N.t:<card> = NP_1.b:<card>
N.t:<decreas> = NP_1.b:<decreas>
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
VP.b:<mode> = nom
VP.b:<assign-case> = acc
VP.b:<compar> = -
VP.t:<passive> = -
N:<agr> = NP_1.b:<agr>
NP_1.b:<wh> = N.t:<wh>
NP_1.t:<case> = acc
NP_1.b:<case> = N.t:<case>
NP_1.b:<pron> = N.t:<pron>
S_r.t:<conj> = nil
S_r.b:<assign-comp> = inf_nil/ind_nil/ecm
```

4 Tree "alphaInx0N1"

4.1 graphe



4.2 comments

```
Tree for predicational NPs:

  John is an author.
  I consider John an author.

<mainv> feature prevents
adjunction of 'do' to auxiliary-headed S
```

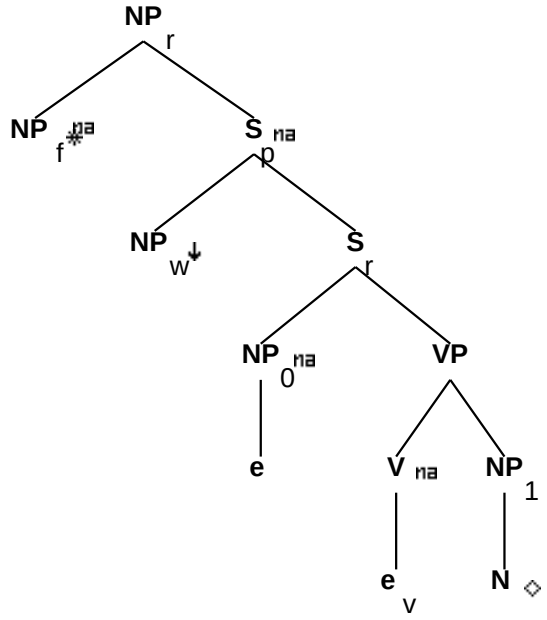
4.3 features

```
S_r.b:<extracted> = -
S_r.b:<inv> = -
S_r.b:<assign-comp> = VP.t:<assign-comp>



S_r.b:<mode> = imp
S_r.b:<mainv> = VP.t:<mainv>
S_r.b:<comp> = nil
S_r.b:<tense> = VP.t:<tense>
NP_0:<agr> = S_r.b:<agr>
NP_0:<case> = S_r.b:<assign-case>
NP_0:<wh> = -
NP_0:<agr pers> = 2
NP_0:<agr 3rdsing> = -
NP_0:<agr num> = plur/sing
NP_0:<case> = nom
NP_1:<case> = acc
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
S_r.b:<passive> = VP.t:<passive>
```

```
VP.t:<passive> = -
VP.t:<tense> = pres
VP.t:<mode> = base
VP.t:<neg> = -
VP.b:<mode> = nom
VP.b:<assign-case> = acc
VP.b:<compar> = -
N:<agr> = NP_1.b:<agr>
NP_1.b:<wh> = N.t:<wh>
NP_1.b:<case> = N.t:<case>
NP_1.b:<pron> = N.t:<pron>
NP_1.b:<compar> = N.t:<compar>
N.t:<compar> = -
N.t:<const> = NP_1.b:<const>
N.t:<gen> = NP_1.b:<gen>
N.t:<definite> = NP_1.b:<definite>
N.t:<quan> = NP_1.b:<quan>
N.t:<card> = NP_1.b:<card>
N.t:<decreas> = NP_1.b:<decreas>
```

5 Tree "betaN0nx0N1"

5.1 graphe



5.2 comments

```
Wh on the subject
 'Who loves Mary'
 'Who has loved Mary'
```

```
check the agr equation on  NP0
```

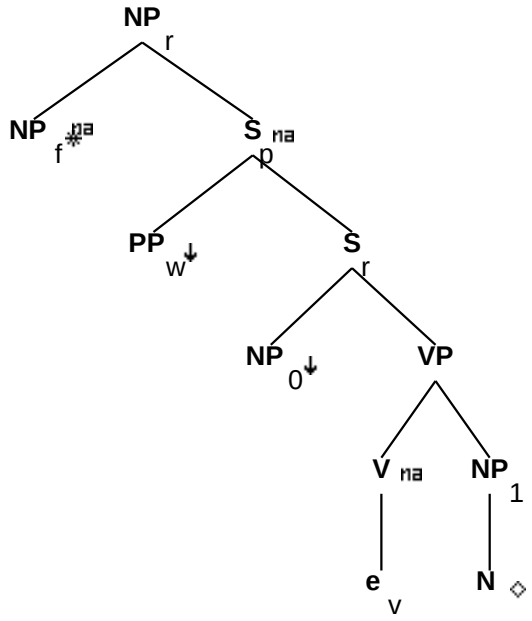
5.3 features

```
S_r.b:<assign-comp> = VP.t:<assign-comp>
```

```
S_r.t:<mode> = ind/inf
S_r.b:<comp> = nil
S_r.b:<mode> = VP.t:<mode>
S_r.b:<tense> = VP.t:<tense>
S_r.t:<inv> = -
NP_0.t:<agr> = S_r.b:<agr>
NP_0.t:<case> = S_r.b:<assign-case>
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
VP.b:<mode> = nom
VP.b:<assign-case> = acc
VP.b:<agr> = NP_1.t:<agr>
VP.b:<compar> = -
NP_1.t:<wh> = -
NP_1.b:<case> = N.t:<case>
NP_1.b:<agr> = N.t:<agr>
NP_1.b:<wh> = N.t:<wh>
NP_1.b:<pron> = N.t:<pron>
NP_1.b:<compar> = N.t:<compar>
N.t:<compar> = -
NP_r.b:<wh> = NP_f.t:<wh>
NP_r.b:<agr> = NP_f.t:<agr>
NP_r.b:<case> = NP_f.t:<case>
N.t:<const> = NP_1.b:<const>
N.t:<gen> = NP_1.b:<gen>
N.t:<definite> = NP_1.b:<definite>
N.t:<quan> = NP_1.b:<quan>
N.t:<card> = NP_1.b:<card>
N.t:<decreas> = NP_1.b:<decreas>
S_r.t:<conj> = nil

NP_w.t:<trace> = NP_0.b:<trace>
NP_w.t:<case> = NP_0.b:<case>
NP_w.t:<agr> = NP_0.b:<agr>
NP_w.t:<wh> = +
S_r.t:<comp> = nil
NP_r.b:<rel-clause> = +
NP_f.b:<case> = nom/acc
NP_r.b:<pron> = NP_f.t:<pron>
```

6 Tree "betaNpxnx0N1"

6.1 graphe



6.2 comments

```
Tree for predicational NPs:

  John is an author.
  I consider John an author.

<mainv> feature prevents
adjunction of 'do' to auxiliary-headed S
```

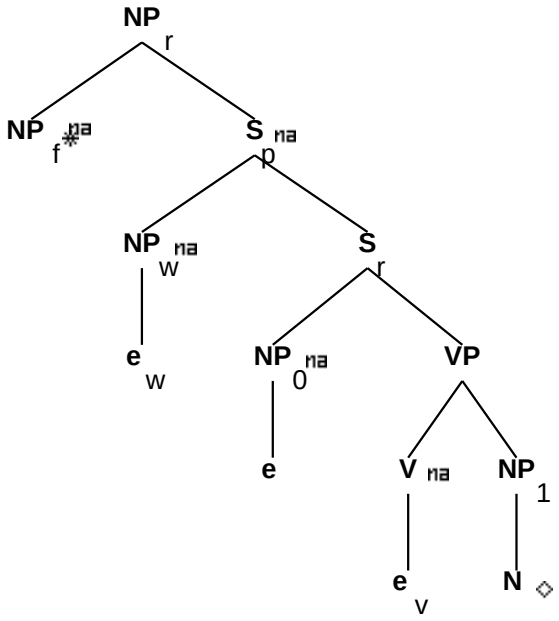
6.3 features

```
S_r.b:<extracted> = -
S_r.b:<inv> = -

S_r.b:<assign-comp> = VP.t:<assign-comp>


VP.b:<compar> = -
S_r.b:<mode> = VP.t:<mode>
S_r.b:<mainv> = VP.t:<mainv>
S_r.b:<comp> = nil
S_r.b:<tense> = VP.t:<tense>
NP_0:<agr> = S_r.b:<agr>
NP_0:<case> = S_r.b:<assign-case>
NP_0:<wh> = -
```

```
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
S_r.b:<passive> = VP.t:<passive>
VP.t:<passive> = -
VP.b:<mode> = nom
VP.b:<assign-case> = acc
N:<agr> = NP_1.b:<agr>
NP_1.t:<wh> = -
NP_1.b:<case> = N.t:<case>
NP_1.b:<pron> = N.t:<pron>
NP_1.b:<wh> = N.t:<wh>
NP_1.b:<compar> = N.t:<compar>
N.t:<compar> = -
N.t:<const> = NP_1.b:<const>
N.t:<gen> = NP_1.b:<gen>
N.t:<definite> = NP_1.b:<definite>
N.t:<quan> = NP_1.b:<quan>
N.t:<card> = NP_1.b:<card>
N.t:<decreas> = NP_1.b:<decreas>
S_r.b:<control> = NP_0.t:<control>
S_r.t:<inv> = -
PP_w.t:<wh> = +
NP_r.b:<wh> = NP_f.t:<wh>
NP_r.b:<agr> = NP_f.t:<agr>
NP_r.b:<case> = NP_f.t:<case>
NP_f.b:<case> = acc/nom
S_r.t:<comp> = nil
NP_r.b:<rel-clause> = +
NP_f.b:<case> = nom/acc
NP_r.b:<pron> = NP_f.t:<pron>
```

7 Tree "betaNc0nx0N1"

7.1 graphe



7.2 comments

```
Wh on the subject
 'Who loves Mary'
 'Who has loved Mary'

check the agr equation on  NP0
```

7.3 features

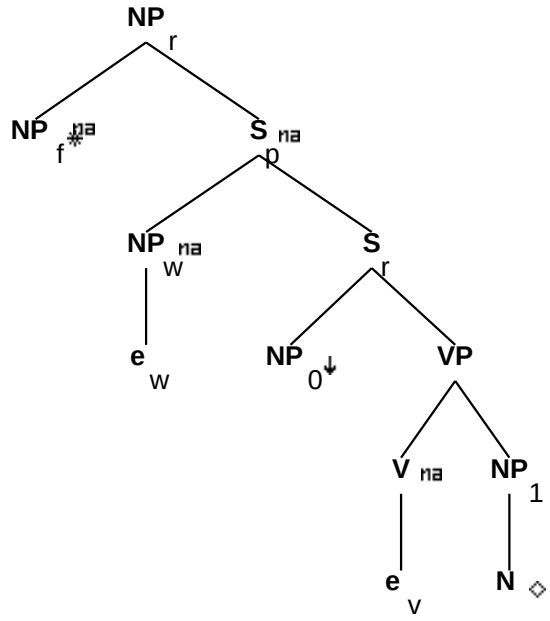
```
S_r.b:<assign-comp> = VP.t:<assign-comp>
```

```
S_r.b:<comp> = nil
S_r.b:<mode> = VP.t:<mode>
S_r.b:<tense> = VP.t:<tense>
S_r.t:<inv> = -
NP_0.t:<agr> = S_r.b:<agr>
NP_0.t:<case> = S_r.b:<assign-case>
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
VP.b:<mode> = nom
VP.b:<assign-case> = acc
VP.b:<agr> = NP_1.t:<agr>
```

```
VP.b:<compar> = -
NP_1.t:<wh> = -
NP_1.b:<case> = N.t:<case>
NP_1.b:<agr> = N.t:<agr>
NP_1.b:<wh> = N.t:<wh>
NP_1.b:<pron> = N.t:<pron>
NP_1.b:<compar> = N.t:<compar>
N.t:<compar> = -
NP_r.b:<wh> = NP_f.t:<wh>
NP_r.b:<agr> = NP_f.t:<agr>
NP_r.b:<case> = NP_f.t:<case>
N.t:<const> = NP_1.b:<const>
N.t:<gen> = NP_1.b:<gen>
N.t:<definite> = NP_1.b:<definite>
N.t:<quan> = NP_1.b:<quan>
N.t:<card> = NP_1.b:<card>
N.t:<decreas> = NP_1.b:<decreas>
S_r.t:<conj> = nil

NP_w.t:<trace> = NP_0.b:<trace>
NP_w.t:<case> = NP_0.b:<case>
NP_w.t:<agr> = NP_0.b:<agr>
NP_r.b:<rel-clause> = +
S_r.t:<mode> = inf/ger/ind
S_r.t:<nocomp-mode> = inf/ger
VP.t:<assign-comp> = that/ind_nil/inf_nil/ecm
S_r.b:<nocomp-mode> = S_r.t:<nocomp-mode>
S_r.b:<nocomp-mode> = S_r.b:<mode>
NP_f.b:<case> = nom/acc
NP_r.b:<pron> = NP_f.t:<pron>
```

8 Tree "betaNcnx0N1"

8.1 graphe



8.2 comments

```
Tree for predicational NPs:

  John is an author.
  I consider John an author.

<mainv> feature prevents
adjunction of 'do' to auxiliary-headed S
```

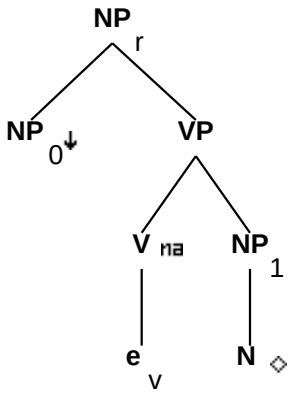
8.3 features

```
S_r.b:<extracted> = -
S_r.b:<inv> = -

S_r.b:<assign-comp> = VP.t:<assign-comp>


S_r.b:<mode> = VP.t:<mode>
S_r.b:<mainv> = VP.t:<mainv>
S_r.b:<comp> = nil
S_r.b:<tense> = VP.t:<tense>
NP_0:<agr> = S_r.b:<agr>
NP_0:<case> = S_r.b:<assign-case>
NP_0:<wh> = -
S_r.b:<agr> = VP.t:<agr>
```

```
S_r.b:<assign-case> = VP.t:<assign-case>
S_r.b:<passive> = VP.t:<passive>
VP.t:<passive> = -
VP.b:<mode> = nom
VP.b:<assign-case> = acc
VP.b:<compar> = -
N:<agr> = NP_1.b:<agr>
NP_1.t:<wh> = -
NP_1.b:<case> = N.t:<case>
NP_1.b:<pron> = N.t:<pron>
NP_1.b:<wh> = N.t:<wh>
NP_1.b:<compar> = N.t:<compar>
N.t:<compar> = -
N.t:<const> = NP_1.b:<const>
N.t:<gen> = NP_1.b:<gen>
N.t:<definite> = NP_1.b:<definite>
N.t:<quan> = NP_1.b:<quan>
N.t:<card> = NP_1.b:<card>
N.t:<decreas> = NP_1.b:<decreas>
S_r.b:<control> = NP_0.t:<control>
NP_r.b:<wh> = NP_f.t:<wh>
NP_r.b:<agr> = NP_f.t:<agr>
NP_r.b:<case> = NP_f.t:<case>
NP_f.b:<case> = acc/nom
S_r.t:<inv> = -
S_r.t:<mode> = ind/inf
S_r.t:<nocomp-mode> = ind
VP.t:<assign-comp> = that/for/ind_nil
S_r.b:<nocomp-mode> = S_r.t:<nocomp-mode>
S_r.b:<nocomp-mode> = S_r.b:<mode>
NP_r.b:<rel-clause> = +
NP_f.b:<case> = nom/acc
NP_r.b:<pron> = NP_f.t:<pron>
```

9 Tree "alphaGnx0N1"

9.1 graphe



9.2 comments

```
Gerund Tree for predicational NPs:

...John('s) being an author...
```

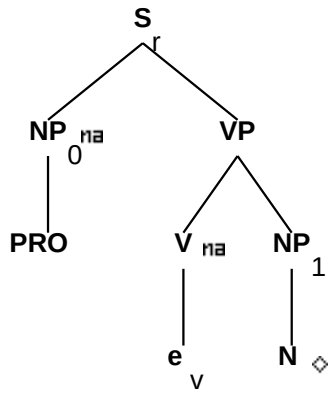
9.3 features

```
NP_0:<wh> = NP_r.b:<wh>
VP.t:<mode> = ger
NP_r.b:<case> = nom/acc
NP_r.b:<agr num> = sing
NP_r.b:<agr pers> = 3
NP_r.b:<agr 3rdsing> = +
VP.b:<mode> = nom
VP.b:<assign-case> = acc
VP.b:<compar> = -
N:<agr> = NP_1.b:<agr>
NP_1.b:<wh> = N.t:<wh>
NP_1.b:<case> = N.t:<case>
NP_1.b:<pron> = N.t:<pron>
NP_1.b:<compar> = N.t:<compar>
N.t:<compar> = -
N.t:<const> = NP_1.b:<const>
N.t:<gen> = NP_1.b:<gen>
N.t:<definite> = NP_1.b:<definite>
N.t:<quan> = NP_1.b:<quan>
N.t:<card> = NP_1.b:<card>
N.t:<decreas> = NP_1.b:<decreas>



NP_r.b:<gerund> = +
```

NP_0:<case> = acc/gen

10 Tree "alphanx0N1-PRO"

10.1 graphe



10.2 comments

Tree for predicational NPs w/ PRO subject

John wants [PRO to be an author].

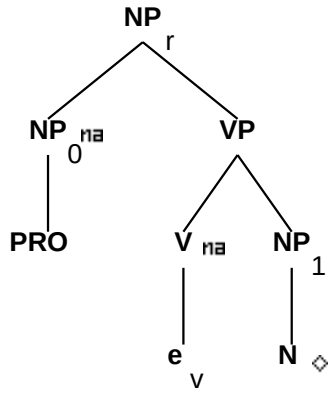
10.3 features

S_r.b:<extracted> = -
S_r.b:<inv> = -
S_r.b:<assign-comp> = VP.t:<assign-comp>
S_r.b:<mode> = VP.t:<mode>
S_r.b:<mainv> = VP.t:<mainv>
S_r.b:<comp> = nil
S_r.b:<tense> = VP.t:<tense>
S_r.b:<assign-case> = NP_0.t:<case>
NP_0:<agr> = S_r.b:<agr>
NP_0:<wh> = -
NP_0.t:<case> = none
NP_1:<case> = acc
S_r.b:<agr> = VP.t:<agr>
S_r.b:<passive> = VP.t:<passive>
VP.t:<passive> = -
VP.b:<mode> = nom
VP.b:<assign-case> = acc
VP.b:<compar> = -
N:<agr> = NP_1.b:<agr>
NP_1.b:<case> = N.t:<case>
NP_1.b:<wh> = N.t:<wh>
NP_1.b:<pron> = N.t:<pron>

NP_1.b:<compar> = N.t:<compar>
N.t:<compar> = -
N.t:<const> = NP_1.b:<const>
N.t:<gen> = NP_1.b:<gen>
N.t:<definite> = NP_1.b:<definite>
N.t:<quan> = NP_1.b:<quan>
N.t:<card> = NP_1.b:<card>
N.t:<decreas> = NP_1.b:<decreas>
S_r.b:<control> = NP_0.t:<control>
VP.t:<mode> = inf/ger

11 Tree "alphaGnx0N1-PRO"

11.1 graphe



11.2 comments

Gerund Tree w/ PRO subject for predicational NPs:

[PRO being an author] is important to John.

11.3 features

NP_0:<wh> = NP_r.b:<wh>
NP_0.t:<case> = none
NP_0.t:<wh> = -
VP.t:<mode> = ger
NP_r.b:<case> = nom/acc
NP_r.b:<agr num> = sing
NP_r.b:<agr pers> = 3
NP_r.b:<agr 3rdsing> = +
VP.b:<mode> = nom
VP.b:<assign-case> = acc
VP.b:<compar> = -
N:<agr> = NP_1.b:<agr>
NP_1.b:<wh> = N.t:<wh>

```
NP_1.b:<case> = N.t:<case>
NP_1.b:<pron> = N.t:<pron>
NP_1.b:<compar> = N.t:<compar>
N.t:<compar> = -
N.t:<const> = NP_1.b:<const>
N.t:<gen> = NP_1.b:<gen>
N.t:<definite> = NP_1.b:<definite>
N.t:<quan> = NP_1.b:<quan>
N.t:<card> = NP_1.b:<card>
N.t:<decreas> = NP_1.b:<decreas>
NP_r.b:<gerund> = +
```