

Adjectifs épithètes, modifieurs de N

$$\llbracket \textit{tigre} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda x \textit{tigre}(x) \rrbracket^{\mathcal{M},w,g} =$$

Adjectifs épithètes, modifieurs de N

$\llbracket \textit{tigre} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda x \textit{tigre}(x) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les tigres de w .

Adjectifs épithètes, modifieurs de N

$\llbracket \textit{tigre} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda x \textit{tigre}(x) \rrbracket^{\mathcal{M},w,g} = \text{l'ensemble de tous les tigres de } w.$
 $\llbracket \textit{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y \textit{édenté}(y) \rrbracket^{\mathcal{M},w,g} =$

Adjectifs épithètes, modifieurs de N

$\llbracket \textit{tigre} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda x \textit{tigre}(x) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les tigres de w .

$\llbracket \textit{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y \textit{édenté}(y) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les individus édentés de w .

Adjectifs épithètes, modifieurs de N

$\llbracket \text{tigre} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda x \text{ tigre}(x) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les tigres de w .

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y \text{ édenté}(y) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les individus édentés de w .

$\llbracket \text{tigre édenté} \rrbracket^{\mathcal{M},w,g} =$

Adjectifs épithètes, modifieurs de N

$\llbracket \text{tigre} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda x \text{ tigre}(x) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les tigres de w .

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y \text{ édenté}(y) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les individus édentés de w .

$\llbracket \text{tigre édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda x \text{ tigre}(x) \rrbracket^{\mathcal{M},w,g} \cap \llbracket \lambda y \text{ édenté}(y) \rrbracket^{\mathcal{M},w,g} \quad \langle e, t \rangle$

Adjectifs épithètes, modifieurs de N

$\llbracket \text{tigre} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda x \text{tigre}(x) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les tigres de w .

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y \text{édenté}(y) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les individus édentés de w .

$\llbracket \text{tigre édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y [\text{tigre}(y) \wedge \text{édenté}(y)] \rrbracket^{\mathcal{M},w,g}$ $\langle e, t \rangle$

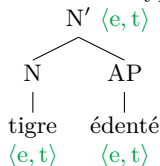
Adjectifs épithètes, modifieurs de N

$\llbracket \text{tigre} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda x \text{tigre}(x) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les tigres de w .

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y \text{édenté}(y) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les individus édentés de w .

$\llbracket \text{tigre édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y [\text{tigre}(y) \wedge \text{édenté}(y)] \rrbracket^{\mathcal{M},w,g}$ $\langle e, t \rangle$

Mais ici les types ne se combinent pas par application fonctionnelle :



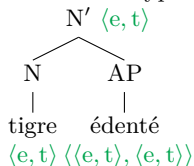
Adjectifs épithètes, modifieurs de N

$\llbracket \text{tigre} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda x \text{ tigre}(x) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les tigres de w .

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y \text{ édenté}(y) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les individus édentés de w .

$\llbracket \text{tigre édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y [\text{tigre}(y) \wedge \text{édenté}(y)] \rrbracket^{\mathcal{M},w,g}$ $\langle e, t \rangle$

Mais ici les types ne se combinent pas par application fonctionnelle :



Si un AP modifie un N, son type devient $\langle \langle e, t \rangle, \langle e, t \rangle \rangle$: c'est une fonction qui attend un N et qui produit un N' ([N AP]).

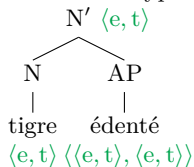
Adjectifs épithètes, modifieurs de N

$\llbracket \text{tigre} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda x \text{ tigre}(x) \rrbracket^{\mathcal{M},w,g}$ = l'ensemble de tous les tigres de w .

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y \text{ édenté}(y) \rrbracket^{\mathcal{M},w,g}$ = l'ensemble de tous les individus édentés de w .

$\llbracket \text{tigre édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y [\text{tigre}(y) \wedge \text{édenté}(y)] \rrbracket^{\mathcal{M},w,g}$ $\langle e, t \rangle$

Mais ici les types ne se combinent pas par application fonctionnelle :



Si un AP modifie un N, son type devient $\langle \langle e, t \rangle, \langle e, t \rangle \rangle$: c'est une fonction qui attend un N et qui produit un N' ($[N \ AP]$).

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda P \lambda y [[P(y)] \wedge \text{édenté}(y)] \rrbracket^{\mathcal{M},w,g}$
 $=$

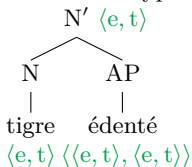
Adjectifs épithètes, modifieurs de N

$\llbracket \text{tigre} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda x \text{ tigre}(x) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les tigres de w .

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y \text{ édenté}(y) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les individus édentés de w .

$\llbracket \text{tigre édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y [\text{tigre}(y) \wedge \text{édenté}(y)] \rrbracket^{\mathcal{M},w,g} \quad \langle e, t \rangle$

Mais ici les types ne se combinent pas par application fonctionnelle :



Si un AP modifie un N, son type devient $\langle \langle e, t \rangle, \langle e, t \rangle \rangle$: c'est une fonction qui attend un N et qui produit un N' $([N \text{ AP}])$.

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda P \lambda y [[P(y)] \wedge \text{édenté}(y)] \rrbracket^{\mathcal{M},w,g}$
 = la fonction qui attend un ensemble (P) et qui retourne l'intersection de $\llbracket P \rrbracket^{\mathcal{M},w,g}$ avec $\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g}$.

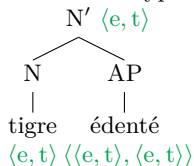
Adjectifs épithètes, modifieurs de N

$\llbracket \text{tigre} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda x \text{ tigre}(x) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les tigres de w .

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y \text{ édenté}(y) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les individus édentés de w .

$\llbracket \text{tigre édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y [\text{tigre}(y) \wedge \text{édenté}(y)] \rrbracket^{\mathcal{M},w,g} \quad \langle e, t \rangle$

Mais ici les types ne se combinent pas par application fonctionnelle :



Si un AP modifie un N, son type devient $\langle \langle e, t \rangle, \langle e, t \rangle \rangle$: c'est une fonction qui attend un N et qui produit un N' $([N \text{ AP}])$.

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda P \lambda y [[P(y)] \wedge \text{édenté}(y)] \rrbracket^{\mathcal{M},w,g}$
 = la fonction qui attend un ensemble (P) et qui retourne l'intersection de $\llbracket P \rrbracket^{\mathcal{M},w,g}$ avec $\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g}$.

$\text{tigre édenté} \rightsquigarrow \llbracket \lambda P \lambda y [[P(y)] \wedge \text{édenté}(y)](\lambda x \text{ tigre}(x)) \rrbracket$

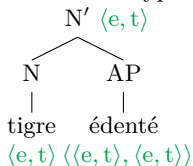
Adjectifs épithètes, modifieurs de N

$\llbracket \text{tigre} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda x \text{ tigre}(x) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les tigres de w .

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y \text{ édenté}(y) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les individus édentés de w .

$\llbracket \text{tigre édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y [\text{tigre}(y) \wedge \text{édenté}(y)] \rrbracket^{\mathcal{M},w,g} \quad \langle e, t \rangle$

Mais ici les types ne se combinent pas par application fonctionnelle :



Si un AP modifie un N, son type devient $\langle \langle e, t \rangle, \langle e, t \rangle \rangle$: c'est une fonction qui attend un N et qui produit un N' ($[N \ AP]$).

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda P \lambda y [[P(y)] \wedge \text{édenté}(y)] \rrbracket^{\mathcal{M},w,g}$
 = la fonction qui attend un ensemble (P) et qui retourne l'intersection de $\llbracket P \rrbracket^{\mathcal{M},w,g}$ avec $\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g}$.

$\text{tigre édenté} \rightsquigarrow \lambda y [\lambda x \text{ tigre}(x)(y)] \wedge \text{édenté}(y)$

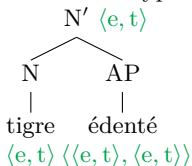
Adjectifs épithètes, modifieurs de N

$\llbracket \text{tigre} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda x \text{tigre}(x) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les tigres de w .

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y \text{édenté}(y) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les individus édentés de w .

$\llbracket \text{tigre édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y [\text{tigre}(y) \wedge \text{édenté}(y)] \rrbracket^{\mathcal{M},w,g}$ $\langle e, t \rangle$

Mais ici les types ne se combinent pas par application fonctionnelle :



Si un AP modifie un N, son type devient $\langle \langle e, t \rangle, \langle e, t \rangle \rangle$: c'est une fonction qui attend un N et qui produit un N' ($[N \ AP]$).

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda P \lambda y [[P(y)] \wedge \text{édenté}(y)] \rrbracket^{\mathcal{M},w,g}$
 = la fonction qui attend un ensemble (P) et qui retourne l'intersection de $\llbracket P \rrbracket^{\mathcal{M},w,g}$ avec $\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g}$.

$\text{tigre édenté} \rightsquigarrow \lambda y [\text{tigre}(y) \wedge \text{édenté}(y)]$

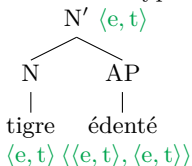
Adjectifs épithètes, modifieurs de N

$\llbracket \text{tigre} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda x \text{ tigre}(x) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les tigres de w .

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y \text{ édenté}(y) \rrbracket^{\mathcal{M},w,g} =$ l'ensemble de tous les individus édentés de w .

$\llbracket \text{tigre édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda y [\text{tigre}(y) \wedge \text{édenté}(y)] \rrbracket^{\mathcal{M},w,g}$ $\langle e, t \rangle$

Mais ici les types ne se combinent pas par application fonctionnelle :



Si un AP modifie un N, son type devient $\langle \langle e, t \rangle, \langle e, t \rangle \rangle$: c'est une fonction qui attend un N et qui produit un N' ($[N \ AP]$).

$\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g} = \llbracket \lambda P \lambda y [[P(y)] \wedge \text{édenté}(y)] \rrbracket^{\mathcal{M},w,g}$
 = la fonction qui attend un ensemble (P) et qui retourne l'intersection de $\llbracket P \rrbracket^{\mathcal{M},w,g}$ avec $\llbracket \text{édenté} \rrbracket^{\mathcal{M},w,g}$.

$\text{tigre édenté} \rightsquigarrow \lambda y [\text{tigre}(y) \wedge \text{édenté}(y)]$

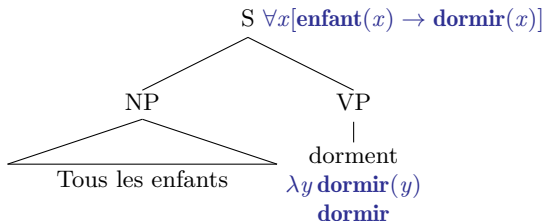
Règle syntaxe-sémantique spéciale :

$$\begin{array}{ccccc}
 N' & \rightarrow & N & AP \\
 [\lambda P \lambda y [[P(y)] \wedge \alpha(y)](\beta)] & \leftarrow & \beta & \alpha
 \end{array}$$

(équivalent à la règle PM de Heim & Kratzer (1997))

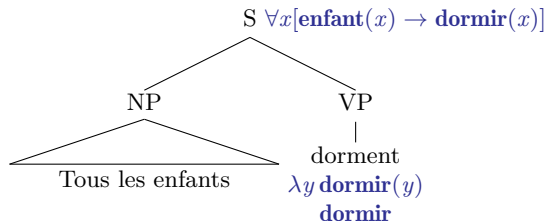
Analyse générale des NP/DP

NP et VP sont dans un bateau



Analyse générale des NP/DP

NP et VP sont dans un bateau

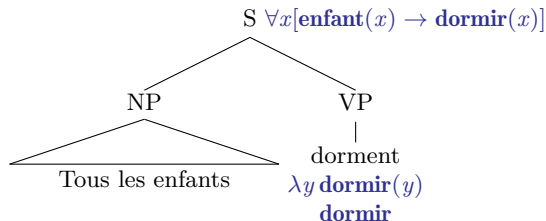


$$S = NP + VP$$

$$NP + VP \rightsquigarrow \forall x[\text{enfant}(x) \rightarrow \text{dormir}(x)]$$

Analyse générale des NP/DP

NP et VP sont dans un bateau

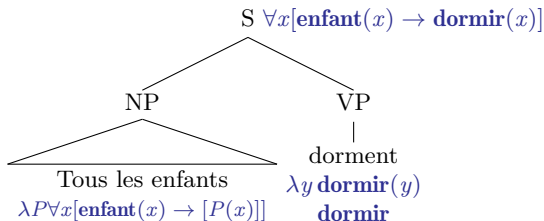


NP = S – VP

NP $\leadsto$ $\forall x[\text{enfant}(x) \rightarrow$ $(x)]$

Analyse générale des NP/DP

NP et VP sont dans un bateau



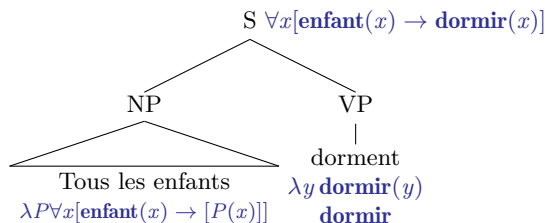
$$NP = S - VP$$

$$NP \rightsquigarrow \lambda P \forall x[\text{enfant}(x) \rightarrow P(x)]$$

 P de type $\langle e, t \rangle$

Analyse générale des NP/DP

NP et VP sont dans un bateau



NP = S – VP

NP $\rightsquigarrow \lambda P \forall x[\text{enfant}(x) \rightarrow P(x)]$ P de type $\langle e, t \rangle$

Le NP est de type $\langle \langle e, t \rangle, t \rangle$, il est la fonction principale de la phrase.

Analyse générale des NP

 $\langle\langle e, t \rangle, t\rangle$

Tous les NP sont de type $\langle\langle e, t \rangle, t\rangle$.

- *Tous les enfants* $\rightsquigarrow \lambda P \forall x [\text{enfant}(x) \rightarrow [P(x)]]$

- *Un enfant* $\rightsquigarrow \lambda P \exists x [\text{enfant}(x) \wedge [P(x)]]$

- *Alice* $\rightsquigarrow \lambda P [P(a)]$

Analyse générale des NP

$\langle\langle e, t \rangle, t\rangle$

Tous les NP sont de type $\langle\langle e, t \rangle, t\rangle$.

(Ensembles de propriétés)

- *Tous les enfants* $\rightsquigarrow \lambda P \forall x [\text{enfant}(x) \rightarrow [P(x)]]$
- *Un enfant* $\rightsquigarrow \lambda P \exists x [\text{enfant}(x) \wedge [P(x)]]$
- *Alice* $\rightsquigarrow \lambda P [P(a)]$
 $[[\lambda P [P(a)]]]^{\mathcal{M}, w, g} \approx$ ensemble de tous les prédicats/propriétés qui s'appliquent à Alice.

Analyse générale des NP

 $\langle\langle e, t \rangle, t\rangle$ Tous les NP sont de type $\langle\langle e, t \rangle, t\rangle$.

(Ensembles de propriétés)

- *Tous les enfants* $\rightsquigarrow \lambda P \forall x [\text{enfant}(x) \rightarrow [P(x)]]$
- *Un enfant* $\rightsquigarrow \lambda P \exists x [\text{enfant}(x) \wedge [P(x)]]$
 $\llbracket \lambda P \exists x [\text{enfant}(x) \wedge [P(x)]] \rrbracket^{\mathcal{M}, w, g} \approx$ ensemble des propriétés qui s'appliquent à un moins un enfant.
- *Alice* $\rightsquigarrow \lambda P [P(\mathbf{a})]$
 $\llbracket \lambda P [P(\mathbf{a})] \rrbracket^{\mathcal{M}, w, g} \approx$ ensemble de tous les prédicats/propriétés qui s'appliquent à Alice.

Analyse générale des NP

 $\langle\langle e, t \rangle, t\rangle$ Tous les NP sont de type $\langle\langle e, t \rangle, t\rangle$.

(Ensembles de propriétés)

- *Tous les enfants* $\rightsquigarrow \lambda P \forall x [\text{enfant}(x) \rightarrow [P(x)]]$
 $[\lambda P \forall x [\text{enfant}(x) \rightarrow [P(x)]]]^{\mathcal{M}, w, g} \approx$ ensemble des propriétés qui s'appliquent à tous les enfants.
- *Un enfant* $\rightsquigarrow \lambda P \exists x [\text{enfant}(x) \wedge [P(x)]]$
 $[\lambda P \exists x [\text{enfant}(x) \wedge [P(x)]]]^{\mathcal{M}, w, g} \approx$ ensemble des propriétés qui s'appliquent à un moins un enfant.
- *Alice* $\rightsquigarrow \lambda P [P(a)]$
 $[\lambda P [P(a)]]^{\mathcal{M}, w, g} \approx$ ensemble de tous les prédicats/propriétés qui s'appliquent à Alice.

Analyse générale des NP

 $\langle \langle e, t \rangle, t \rangle$ Tous les NP sont de type $\langle \langle e, t \rangle, t \rangle$.

(Ensembles de propriétés)

- *Tous les enfants* $\rightsquigarrow \lambda P \forall x [\text{enfant}(x) \rightarrow [P(x)]]$
 $[\lambda P \forall x [\text{enfant}(x) \rightarrow [P(x)]]]^{\mathcal{M}, w, g} \approx$ ensemble des propriétés qui s'appliquent à tous les enfants.
- *Un enfant* $\rightsquigarrow \lambda P \exists x [\text{enfant}(x) \wedge [P(x)]]$
 $[\lambda P \exists x [\text{enfant}(x) \wedge [P(x)]]]^{\mathcal{M}, w, g} \approx$ ensemble des propriétés qui s'appliquent à un moins un enfant.
- *Alice* $\rightsquigarrow \lambda P [P(a)]$
 $[\lambda P [P(a)]]^{\mathcal{M}, w, g} \approx$ ensemble de tous les prédicats/propriétés qui s'appliquent à Alice.

$[\lambda P [P(a)] (\lambda y \text{ dormir}(y))] \approx$ la propriété de dormir est une des propriétés possédées par Alice.

Analyse générale des NP

 $\langle\langle e, t \rangle, t\rangle$ Tous les NP sont de type $\langle\langle e, t \rangle, t\rangle$.

(Ensembles de propriétés)

- *Tous les enfants* $\rightsquigarrow \lambda P \forall x [\text{enfant}(x) \rightarrow [P(x)]]$
 $[\lambda P \forall x [\text{enfant}(x) \rightarrow [P(x)]]]^{\mathcal{M}, w, g} \approx$ ensemble des propriétés qui s'appliquent à tous les enfants.
- *Un enfant* $\rightsquigarrow \lambda P \exists x [\text{enfant}(x) \wedge [P(x)]]$
 $[\lambda P \exists x [\text{enfant}(x) \wedge [P(x)]]]^{\mathcal{M}, w, g} \approx$ ensemble des propriétés qui s'appliquent à un moins un enfant.
- *Alice* $\rightsquigarrow \lambda P [P(a)]$
 $[\lambda P [P(a)]]^{\mathcal{M}, w, g} \approx$ ensemble de tous les prédicats/propriétés qui s'appliquent à Alice.

 $[\lambda y \text{ dormir}(y)(a)] \approx$ Alice vérifie la propriété de dormir

Analyse générale des NP

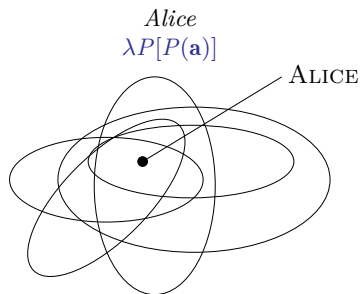
 $\langle \langle e, t \rangle, t \rangle$ Tous les NP sont de type $\langle \langle e, t \rangle, t \rangle$.

(Ensembles de propriétés)

- *Tous les enfants* $\rightsquigarrow \lambda P \forall x [\text{enfant}(x) \rightarrow [P(x)]]$
 $[\lambda P \forall x [\text{enfant}(x) \rightarrow [P(x)]]]^{\mathcal{M}, w, g} \approx$ ensemble des propriétés qui s'appliquent à tous les enfants.
- *Un enfant* $\rightsquigarrow \lambda P \exists x [\text{enfant}(x) \wedge [P(x)]]$
 $[\lambda P \exists x [\text{enfant}(x) \wedge [P(x)]]]^{\mathcal{M}, w, g} \approx$ ensemble des propriétés qui s'appliquent à un moins un enfant.
- *Alice* $\rightsquigarrow \lambda P [P(a)]$
 $[\lambda P [P(a)]]^{\mathcal{M}, w, g} \approx$ ensemble de tous les prédicats/propriétés qui s'appliquent à Alice.

 $\text{dormir}(a) \approx$ Alice appartient à l'ensemble des dormeurs

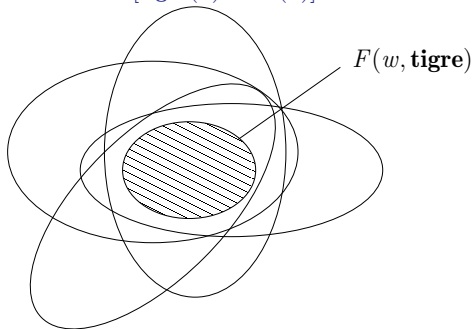
Les NP dénotent des ensembles de propriétés



Ensemble de tous les ensembles qui contiennent ALICE

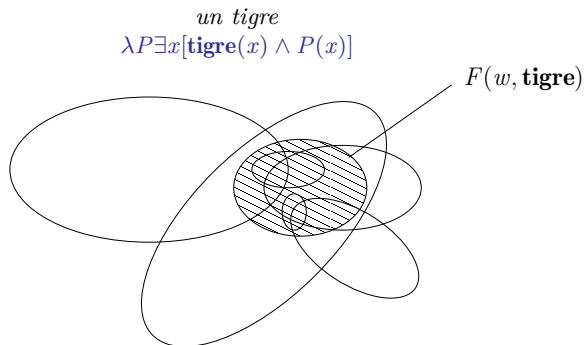
Les NP dénotent des ensembles de propriétés

tous les tigres
 $\lambda P \forall x [\mathbf{tigre}(x) \rightarrow P(x)]$



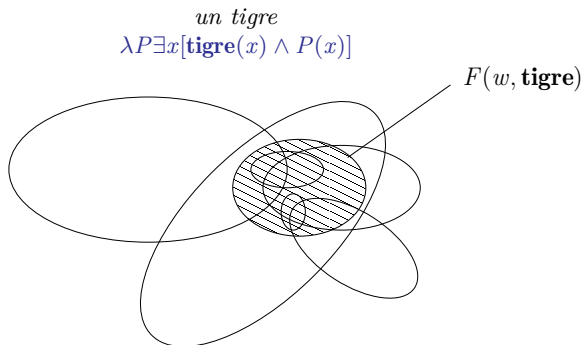
Ensemble de tous les ensembles qui contiennent tous les tigres

Les NP dénotent des ensembles de propriétés



Ensemble de tous les ensembles qui contiennent au moins un tigre

Les NP dénotent des ensembles de propriétés



Ensemble de tous les ensembles qui contiennent au moins un tigre

☞ Les NP ne sont pas référentiels (ils ne sont pas de type e), mais ils ont une dénotation : un ensemble de propriétés extensionnelles.

Déterminants

Les déterminants attendent un N_c ($\langle e, t \rangle$) pour former un NP

Déterminants

Les déterminants attendent un Nc ($\langle e, t \rangle$) pour former un NP qui lui-même attend un VP ($\langle e, t \rangle$) pour former une S.

Déterminants

Les déterminants attendent un Nc ($\langle e, t \rangle$) pour former un NP qui lui-même attend un VP ($\langle e, t \rangle$) pour former une S.

Donc les déterminants sont des fonctions qui renvoient une valeur de vérité si on leur fournit deux propriétés.

Leur type est $\langle \langle e, t \rangle, \langle \langle e, t \rangle, t \rangle \rangle$; ils posent une relation entre 2 ensembles.

Déterminants

Les déterminants attendent un Nc ($\langle e, t \rangle$) pour former un NP qui lui-même attend un VP ($\langle e, t \rangle$) pour former une S.

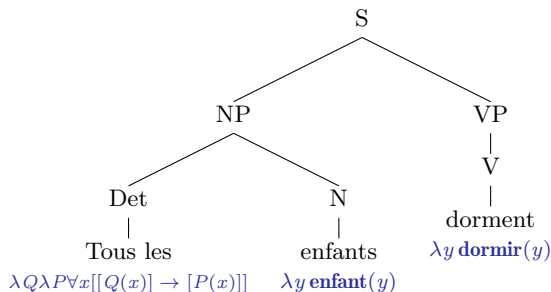
Donc les déterminants sont des fonctions qui renvoient une valeur de vérité si on leur fournit deux propriétés.

Leur type est $\langle \langle e, t \rangle, \langle \langle e, t \rangle, t \rangle \rangle$; ils posent une relation entre 2 ensembles.

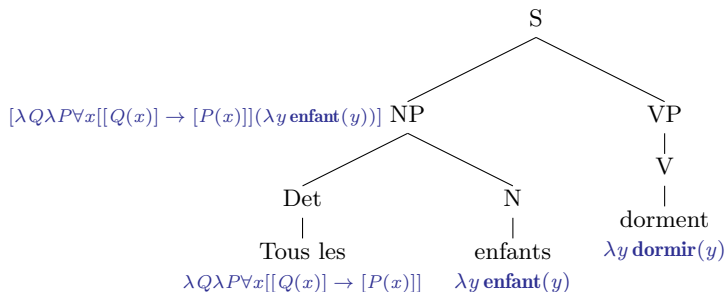
- *tous les* $\rightsquigarrow \lambda Q \lambda P \forall x [[Q(x)] \rightarrow [P(x)]]$;
 - $\llbracket Q \rrbracket^{\mathcal{M}, w, g} \subset \llbracket P \rrbracket^{\mathcal{M}, w, g}$
- *un* $\rightsquigarrow \lambda Q \lambda P \exists x [[Q(x)] \wedge [P(x)]]$;
 - $\llbracket Q \rrbracket^{\mathcal{M}, w, g} \cap \llbracket P \rrbracket^{\mathcal{M}, w, g} \neq \emptyset$
- *le* $\rightsquigarrow \lambda Q \lambda P [P(\iota x [Q(x)])]$;
 - $\llbracket Q \rrbracket^{\mathcal{M}, w, g}$ est un singleton et $\llbracket Q \rrbracket^{\mathcal{M}, w, g} \subset \llbracket P \rrbracket^{\mathcal{M}, w, g}$

Q : argument N P : argument VP

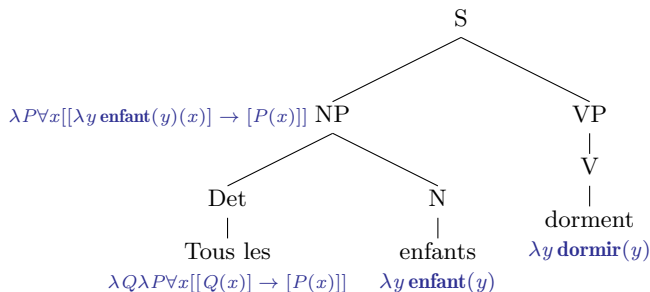
Exemple



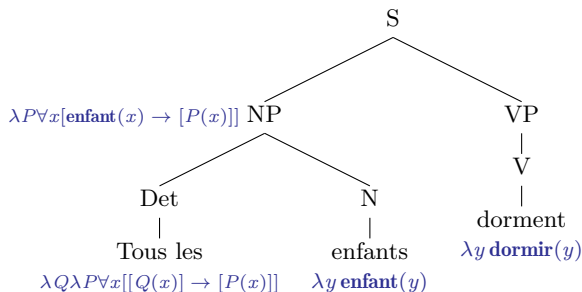
Exemple



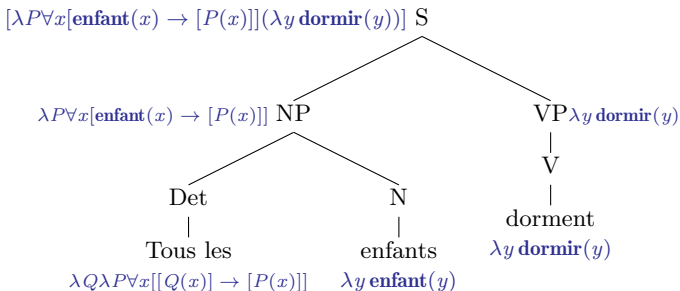
Exemple



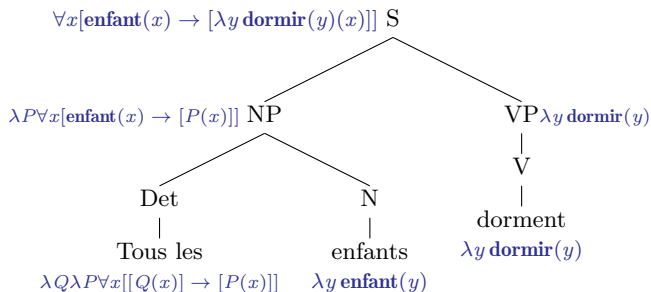
Exemple



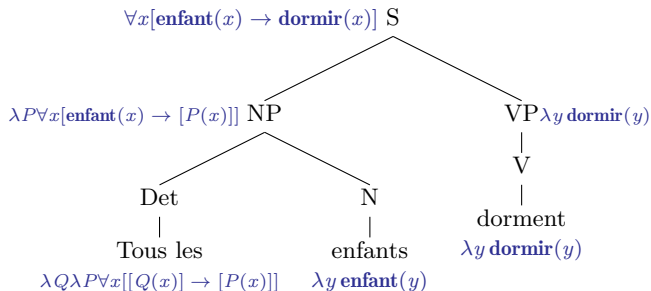
Exemple



Exemple



Exemple

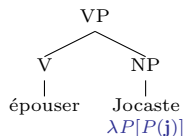


Verbes transitifs

- Les verbes intransitifs sont de type $\langle e, t \rangle$.
- Les verbes transitifs sont-ils de type $\langle e, \langle e, t \rangle \rangle$?

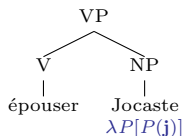
Verbes transitifs

- Les verbes intransitifs sont de type $\langle e, t \rangle$.
- Les verbes transitifs sont-ils de type $\langle e, \langle e, t \rangle \rangle$?
Non, car leur NP complément (objet) est de type $\langle \langle e, t \rangle, t \rangle$.



Verbes transitifs

- Les verbes intransitifs sont de type $\langle e, t \rangle$.
- Les verbes transitifs sont-ils de type $\langle e, \langle e, t \rangle \rangle$?
Non, car leur NP complément (objet) est de type $\langle \langle e, t \rangle, t \rangle$.

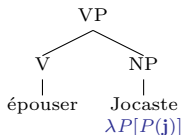


- On veut qu'un VT prenne un complément de type $\langle \langle e, t \rangle, t \rangle$ et produise un VP de type $\langle e, t \rangle$.

Il est donc de type $\langle \underbrace{\langle \langle e, t \rangle, t \rangle}_{NP}, \underbrace{\langle e, t \rangle}_{VP} \rangle$.

Verbes transitifs

- Les verbes intransitifs sont de type $\langle e, t \rangle$.
- Les verbes transitifs sont-ils de type $\langle e, \langle e, t \rangle \rangle$?
Non, car leur NP complément (objet) est de type $\langle \langle e, t \rangle, t \rangle$.



- On veut qu'un VT prenne un complément de type $\langle \langle e, t \rangle, t \rangle$ et produise un VP de type $\langle e, t \rangle$.

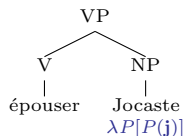
Il est donc de type $\underbrace{\langle \langle e, t \rangle, t \rangle}_{\text{NP}}, \underbrace{\langle e, t \rangle}_{\text{VP}} \rangle$.

$\text{épouser} \rightsquigarrow \lambda Y \lambda x [Y(\lambda y \text{épouser}(x, y))]$

Y de type $\langle \langle e, t \rangle, t \rangle$

Verbes transitifs

- Les verbes intransitifs sont de type $\langle e, t \rangle$.
- Les verbes transitifs sont-ils de type $\langle e, \langle e, t \rangle \rangle$?
Non, car leur NP complément (objet) est de type $\langle \langle e, t \rangle, t \rangle$.



- On veut qu'un VT prenne un complément de type $\langle \langle e, t \rangle, t \rangle$ et produise un VP de type $\langle e, t \rangle$.

Il est donc de type $\langle \underbrace{\langle \langle e, t \rangle, t \rangle}_{\text{NP}}, \underbrace{\langle e, t \rangle}_{\text{VP}} \rangle$.

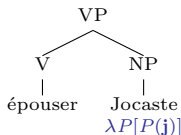
$\text{épouser} \rightsquigarrow \lambda Y \lambda x [Y(\lambda y \text{épouser}(x, y))]$

Y de type $\langle \langle e, t \rangle, t \rangle$

$[_{\text{VP}} \text{épouser } \text{Jocaste}] \rightsquigarrow [\lambda Y \lambda x [Y(\lambda y \text{épouser}(x, y))]](\lambda P[P(j)])]$

Verbes transitifs

- Les verbes intransitifs sont de type $\langle e, t \rangle$.
- Les verbes transitifs sont-ils de type $\langle e, \langle e, t \rangle \rangle$?
Non, car leur NP complément (objet) est de type $\langle \langle e, t \rangle, t \rangle$.



- On veut qu'un VT prenne un complément de type $\langle \langle e, t \rangle, t \rangle$ et produise un VP de type $\langle e, t \rangle$.

Il est donc de type $\underbrace{\langle \langle e, t \rangle, t \rangle}_{NP}, \underbrace{\langle e, t \rangle}_{VP}$.

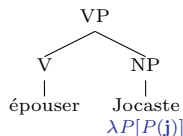
$$\text{épouser} \rightsquigarrow \lambda Y \lambda x [Y(\lambda y \text{épouser}(x, y))]$$

Y de type $\langle \langle e, t \rangle, t \rangle$

$$[_{VP} \text{épouser Jocaste}] \rightsquigarrow \lambda x [\lambda P[P(j)](\lambda y \text{épouser}(x, y))]$$

Verbes transitifs

- Les verbes intransitifs sont de type $\langle e, t \rangle$.
- Les verbes transitifs sont-ils de type $\langle e, \langle e, t \rangle \rangle$?
Non, car leur NP complément (objet) est de type $\langle \langle e, t \rangle, t \rangle$.



- On veut qu'un VT prenne un complément de type $\langle \langle e, t \rangle, t \rangle$ et produise un VP de type $\langle e, t \rangle$.

Il est donc de type $\underbrace{\langle \langle e, t \rangle, t \rangle}_{\text{NP}}, \underbrace{\langle e, t \rangle}_{\text{VP}} \rangle$.

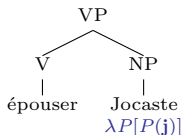
$$\text{épouser} \rightsquigarrow \lambda Y \lambda x [Y(\lambda y \text{épouser}(x, y))]$$

Y de type $\langle \langle e, t \rangle, t \rangle$

$$[_{\text{VP}} \text{épouser} \text{Jocaste}] \rightsquigarrow \lambda x [\lambda y \text{épouser}(x, y)(j)]$$

Verbes transitifs

- Les verbes intransitifs sont de type $\langle e, t \rangle$.
- Les verbes transitifs sont-ils de type $\langle e, \langle e, t \rangle \rangle$?
Non, car leur NP complément (objet) est de type $\langle \langle e, t \rangle, t \rangle$.



- On veut qu'un VT prenne un complément de type $\langle \langle e, t \rangle, t \rangle$ et produise un VP de type $\langle e, t \rangle$.

Il est donc de type $\underbrace{\langle \langle e, t \rangle, t \rangle}_{NP}, \underbrace{\langle e, t \rangle}_{VP}$.

$$\text{épouser} \rightsquigarrow \lambda Y \lambda x [Y(\lambda y \text{épouser}(x, y))]$$

Y de type $\langle \langle e, t \rangle, t \rangle$

$$[_{VP} \text{épouser } \text{Jocaste}] \rightsquigarrow \lambda x \text{épouser}(x, j)$$

Le verbe transitif *être*

Two *BEs* or not two *BEs*?

Plusieurs emplois du verbe *être*:

- Transitif (+ NP)
 - Identificationnel
Peter Parker est Spiderman.
 - Prédicatif (+ NP)
Peter Parker est un étudiant.

$\text{être} \rightsquigarrow \lambda X \lambda x [X(\lambda y [x = y])]$

- Copule (+ AP)
Peter Parker est amoureux.

$\text{être} \rightsquigarrow \lambda P \lambda x [P(x)]$

Ambiguïtés de portée

Les NP sont mobiles

Hypothèse

Tout NP *peut* s'interpréter « à gauche » de la phrase où il apparaît.

Ambiguïtés de portée

Les NP sont mobiles

Hypothèse

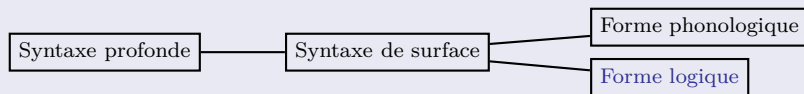
Tout NP *peut* s'interpréter « à gauche » de la phrase où il apparaît.

- (3) a. Tous les élèves ont appris une poésie.
- b. 'Une poésie tous les élèves ont appris.'
- c. 'Tous les élèves une poésie ont appris.'

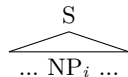
Implémentation syntaxique : *Quantifier Raising*

Les NP sont mobiles

Modèle grammatical « en Y » (e.g. GB)

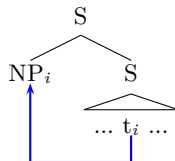


Structure de surface



$\Rightarrow$

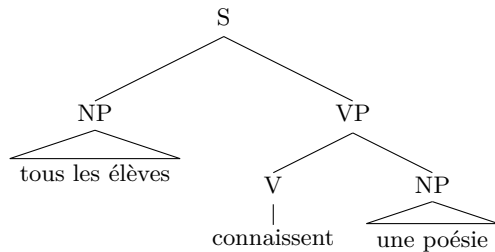
Forme logique



La trace t_i se traduit par une variable (« numérotée ») : $t_i \rightsquigarrow \lambda P[P(x_i)]$.

Quantifier Raising

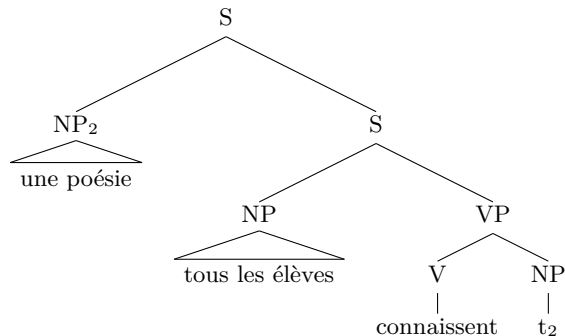
Exemples



Le mouvement est toujours possible, mais jamais obligatoire.

Quantifier Raising

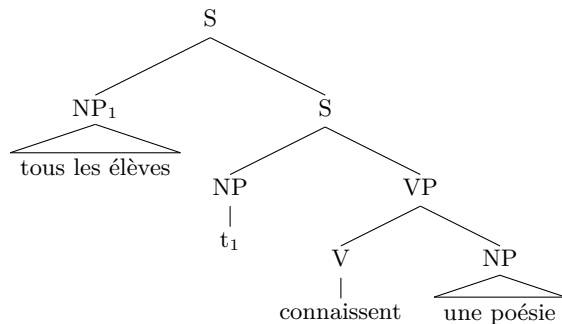
Exemples



Le mouvement est toujours possible, mais jamais obligatoire.

Quantifier Raising

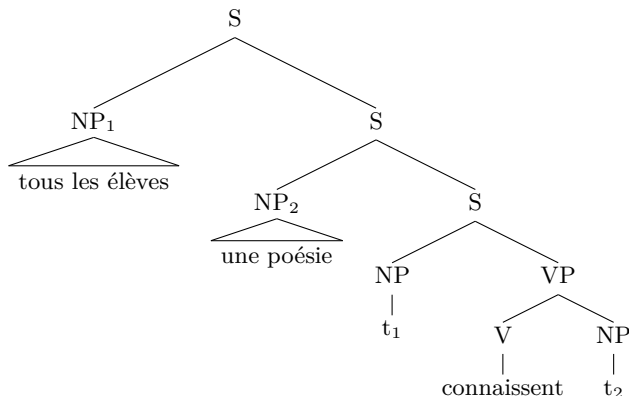
Exemples



Le mouvement est toujours possible, mais jamais obligatoire.

Quantifier Raising

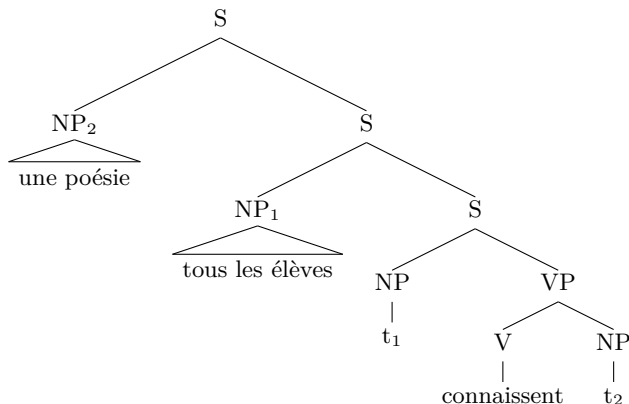
Exemples



Le mouvement est toujours possible, mais jamais obligatoire.

Quantifier Raising

Exemples

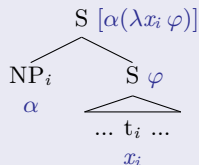


Le mouvement est toujours possible, mais jamais obligatoire.

Interprétation

Les traces sont des variables λ -abstraites (au bon moment)

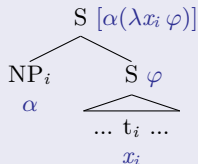
Interprétation des traces (e.g. *QR*)



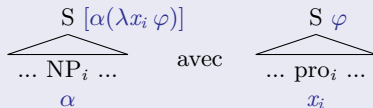
Interprétation

Les traces sont des variables λ -abstraites (au bon moment)

Interprétation des traces (e.g. *QR*)



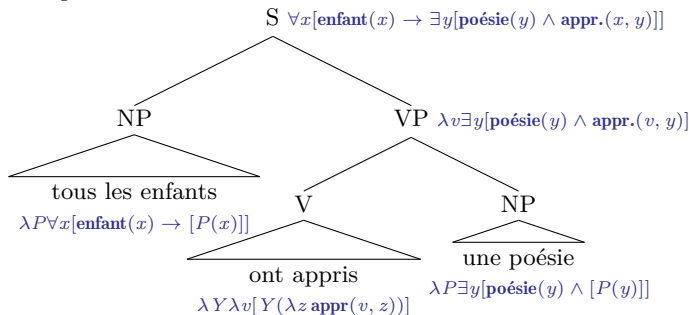
Quantifying in (Montague)



Ambiguïté de portée

Exemple

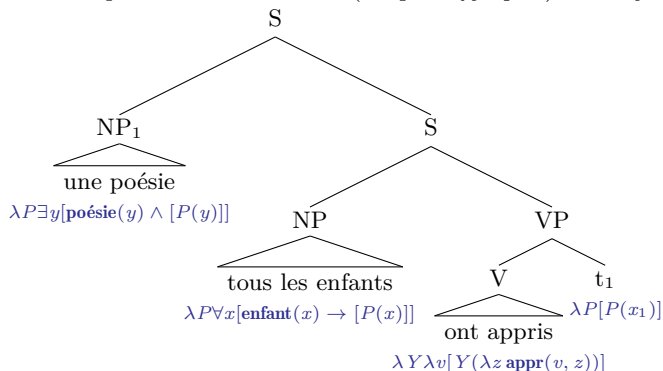
Interprétation des NP *in situ*



Ambiguïté de portée

Exemple

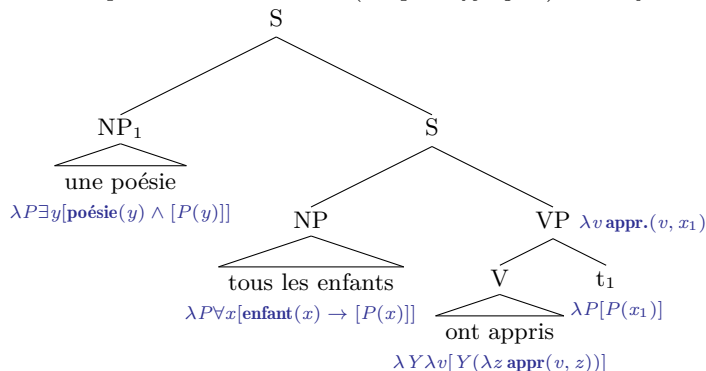
Interprétation avec montée (ou *quantifying in*) de l'objet



Ambiguïté de portée

Exemple

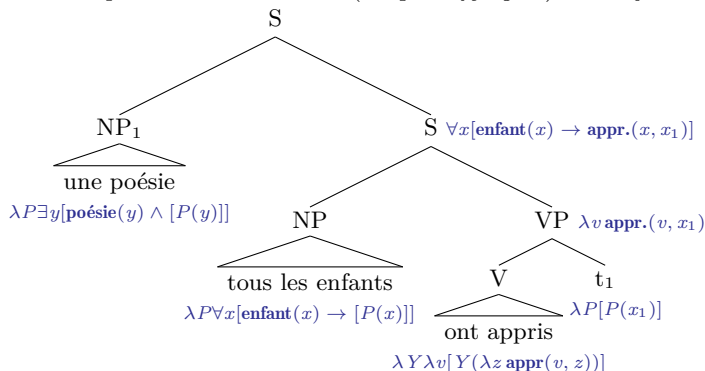
Interprétation avec montée (ou *quantifying in*) de l'objet



Ambiguïté de portée

Exemple

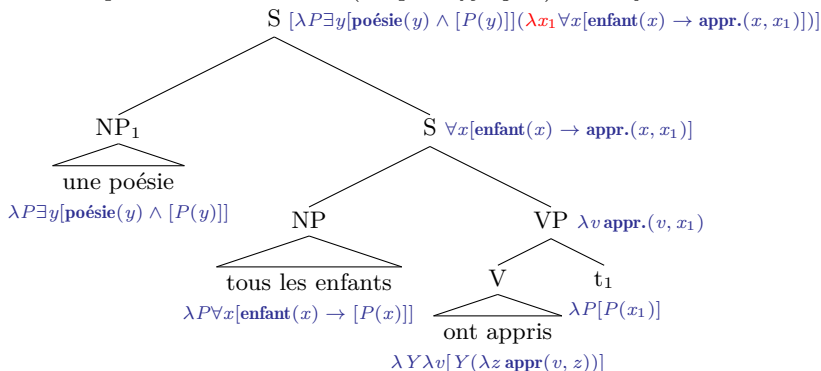
Interprétation avec montée (ou *quantifying in*) de l'objet



Ambiguïté de portée

Exemple

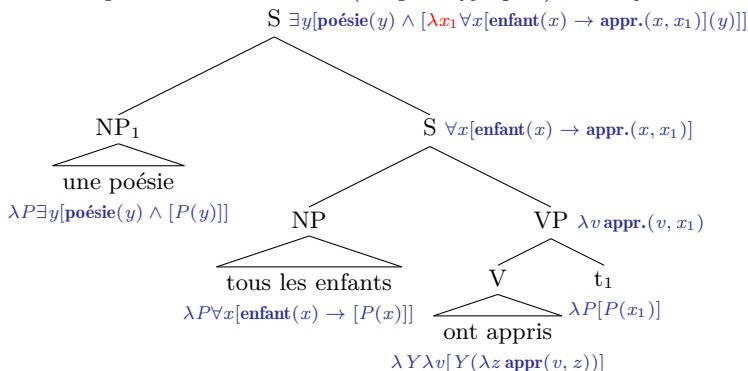
Interprétation avec montée (ou *quantifying in*) de l'objet



Ambiguïté de portée

Exemple

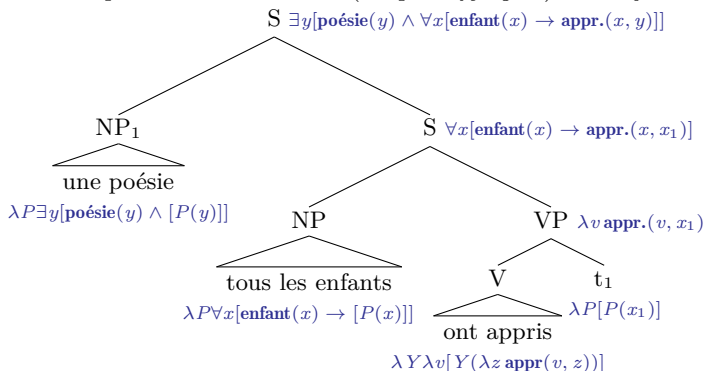
Interprétation avec montée (ou *quantifying in*) de l'objet



Ambiguïté de portée

Exemple

Interprétation avec montée (ou *quantifying in*) de l'objet



De dicto vs. de re

Verbes d'attitudes propositionnelles + subordonnées complétives

Ils dénotent une relation entre une proposition φ ($\langle s, t \rangle$) et un individu x .

Ils sont de type $\langle \langle s, t \rangle, \langle e, t \rangle \rangle$

De dicto vs. de re

Verbes d'attitudes propositionnelles + subordonnées complétives

Ils dénotent une relation entre une proposition φ ($\langle s, t \rangle$) et un individu x .

Ils sont de type $\langle \langle s, t \rangle, \langle e, t \rangle \rangle$

- $\text{croire} \rightsquigarrow \lambda\varphi\lambda x \text{croire}(x, \varphi)$

De dicto vs. de re

Verbes d'attitudes propositionnelles + subordonnées complétives

Ils dénotent une relation entre une proposition φ ($\langle s, t \rangle$) et un individu x .

Ils sont de type $\langle \langle s, t \rangle, \langle e, t \rangle \rangle$

- $\text{croire} \rightsquigarrow \lambda\varphi\lambda x \text{croire}(x, \varphi)$
 $\llbracket \text{croire}(x, \varphi) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi les croyances qu'a $\llbracket x \rrbracket^{\mathcal{M}, w, g}$ dans w
 impliquent φ ;

De dicto vs. de re

Verbes d'attitudes propositionnelles + subordonnées complétives

Ils dénotent une relation entre une proposition φ ($\langle s, t \rangle$) et un individu x .

Ils sont de type $\langle \langle s, t \rangle, \langle e, t \rangle \rangle$

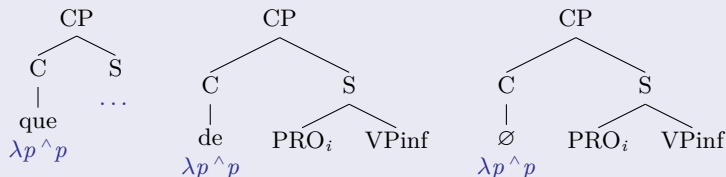
- $\text{croire} \rightsquigarrow \lambda\varphi\lambda x \text{croire}(x, \varphi)$
 $\llbracket \text{croire}(x, \varphi) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi les croyances qu'a $\llbracket x \rrbracket^{\mathcal{M}, w, g}$ dans w impliquent φ ;
- $\text{vouloir} \rightsquigarrow \lambda\varphi\lambda x \text{vouloir}(x, \varphi)$
 $\llbracket \text{vouloir}(x, \varphi) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi l'ensemble de tous les mondes w' compatibles avec les désirs de $\llbracket x \rrbracket^{\mathcal{M}, w, g}$ dans w est inclus dans $\llbracket \varphi \rrbracket^{\mathcal{M}, w, g}$;
 autrement dit : dans w , $\llbracket x \rrbracket^{\mathcal{M}, w, g}$ souhaite que φ devienne vraie (dans (le futur de) w).

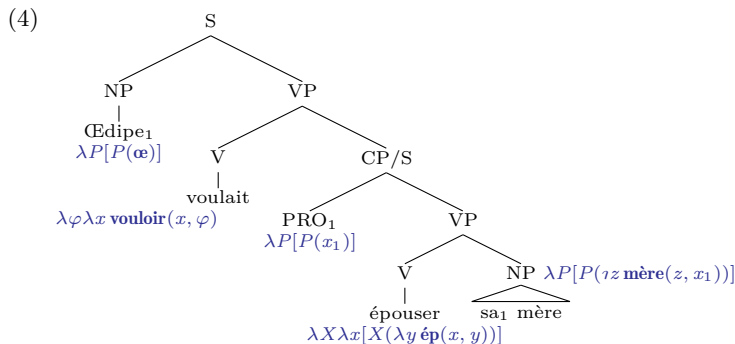
De dicto vs. de re

Complémenteurs et complétives

Les complémenteurs (*que*, *de*) introduisent $\wedge$, pour former une proposition à partir d'une formule.

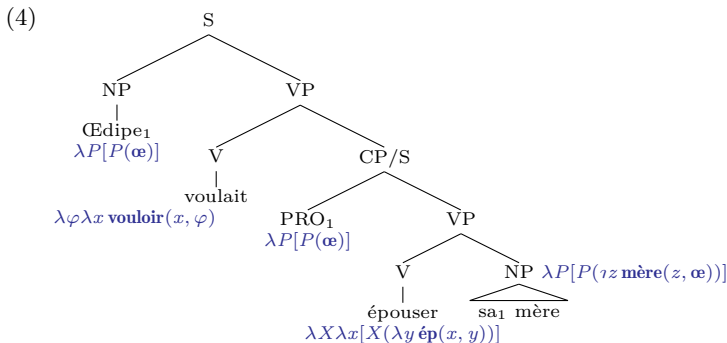
Ils sont donc de type $\langle t, \langle s, t \rangle \rangle$:





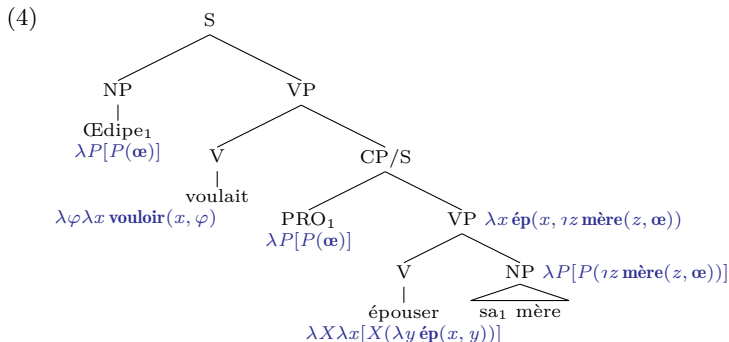
De dicto vs. de re

Œdipe voulait épouser sa mère (de dicto)



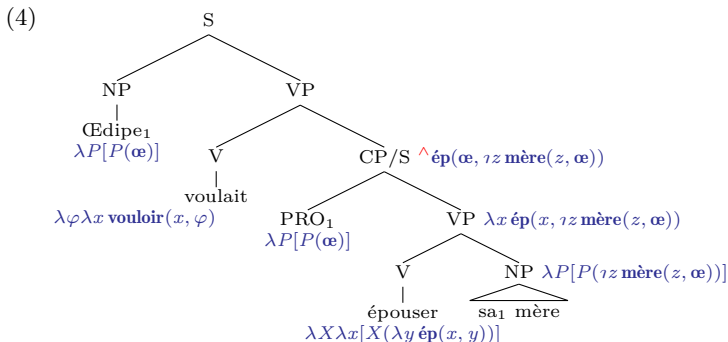
De dicto vs. de re

Œdipe voulait épouser sa mère (de dicto)



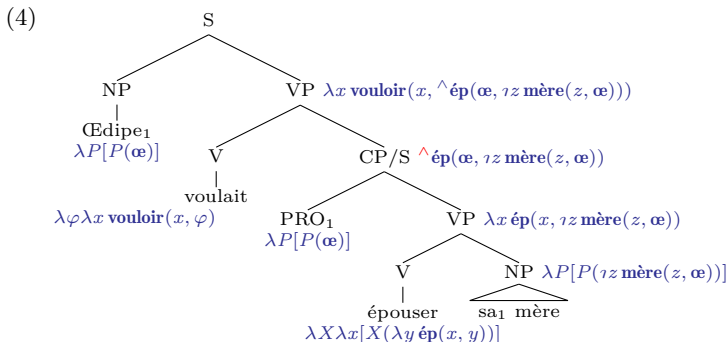
De dicto vs. de re

Œdipe voulait épouser sa mère (de dicto)



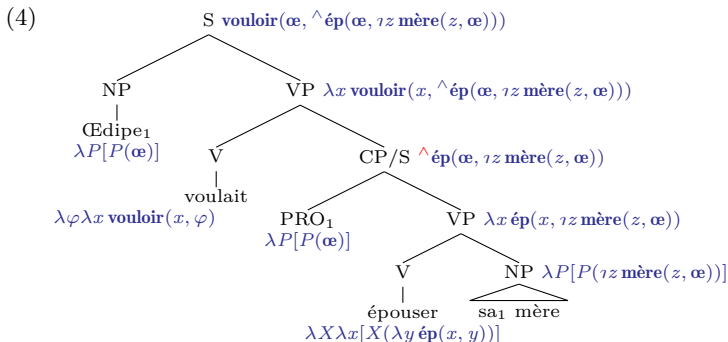
De dicto vs. de re

Œdipe voulait épouser sa mère (de dicto)



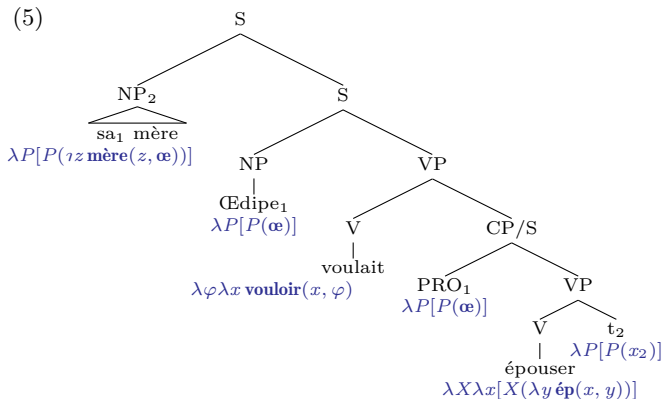
De dicto vs. de re

Œdipe voulait épouser sa mère (de dicto)



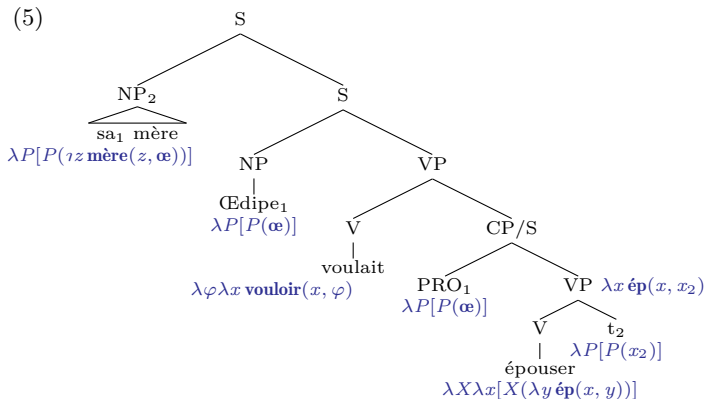
De dicto vs. de re

Œdipe voulait épouser sa mère (de re)



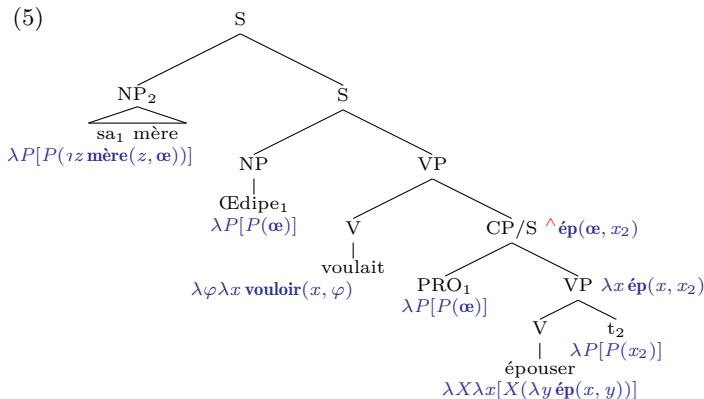
De dicto vs. de re

Œdipe voulait épouser sa mère (de re)



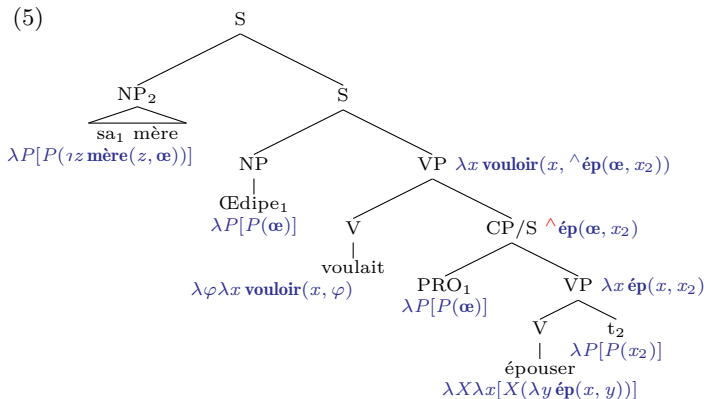
De dicto vs. de re

Edipe voulait épouser sa mère (de re)



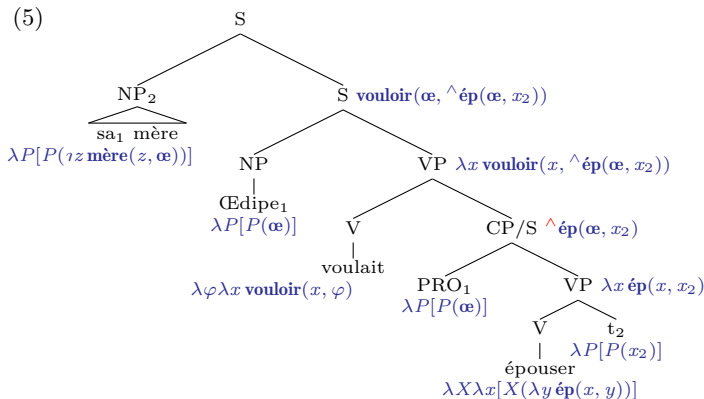
De dicto vs. de re

Edipe voulait épouser sa mère (de re)



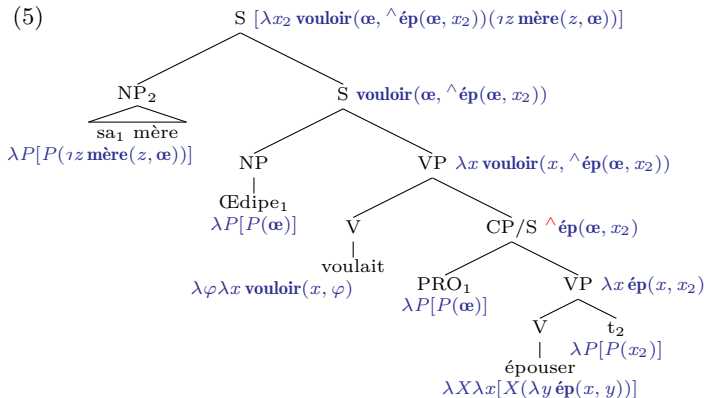
De dicto vs. de re

Edipe voulait épouser sa mère (de re)



De dicto vs. de re

Edipe voulait épouser sa mère (de re)



Lecture de dicto

Détail

- $\llbracket \text{ép}(\text{œ}, \text{mère}(z, \text{œ})) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi celle qui est la mère d'œ. dans w épouse œ. dans w .

Lecture de dicto

Détail

- $\llbracket \text{ép}(\alpha, \lambda z \text{ mère}(z, \alpha)) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi celle qui est la mère d'Æ. dans w épouse Æ. dans w .
- $\llbracket ^\wedge \text{ép}(\alpha, \lambda z \text{ mère}(z, \alpha)) \rrbracket^{\mathcal{M}, w, g} =$ l'ensemble de tous les mondes w' où celle qui est la mère d'Æ. dans w' épouse Æ.

Lecture de dicto

Détail

- $\llbracket \text{ép}(\alpha, \text{mère}(z, \alpha)) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi celle qui est la mère d'Æ. dans w épouse Æ. dans w .
- $\llbracket \text{^ép}(\alpha, \text{mère}(z, \alpha)) \rrbracket^{\mathcal{M}, w, g} =$ l'ensemble de tous les mondes w' où celle qui est la mère d'Æ. dans w' épouse Æ.
- $\llbracket \text{vouloir}(\alpha, \text{^ép}(\alpha, \text{mère}(z, \alpha))) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi dans w Æ. souhaite que w appartienne à l'ensemble de tous les mondes w' où celle qui est la mère d'Æ. dans w' épouse Æ.

Lecture de re

Détail

- $\llbracket \text{ép}(\alpha, x_2) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi $\mathbb{E}. \text{épouse } g(x_2) \text{ dans } w$.

Lecture de re

Détail

- $\llbracket \text{ép}(\mathfrak{e}, x_2) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi $\mathfrak{E}$. épouse $g(x_2)$ dans w .
- $\llbracket \wedge \text{ép}(\mathfrak{e}, x_2) \rrbracket^{\mathcal{M}, w, g} =$ l'ensemble de tous les mondes w' où $\mathfrak{E}$. épouse $g(x_2)$.

Lecture de re

Détail

- $\llbracket \text{ép}(\alpha, x_2) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi $\mathbb{E}$. épouse $g(x_2)$ dans w .
- $\llbracket \text{^ép}(\alpha, x_2) \rrbracket^{\mathcal{M}, w, g}$ = l'ensemble de tous les mondes w' où $\mathbb{E}$. épouse $g(x_2)$.
- $\llbracket \text{vouloir}(\alpha, \text{^ép}(\alpha, x_2)) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi dans w $\mathbb{E}$. souhaite que w appartienne à l'ensemble de tous les mondes w' où $\mathbb{E}$. épouse $g(x_2)$.

Lecture de re

Détail

- $\llbracket \text{ép}(\alpha, x_2) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi $\mathbb{E}$. épouse $g(x_2)$ dans w .
- $\llbracket \text{^ép}(\alpha, x_2) \rrbracket^{\mathcal{M}, w, g}$ = l'ensemble de tous les mondes w' où $\mathbb{E}$. épouse $g(x_2)$.
- $\llbracket \text{vouloir}(\alpha, \text{^ép}(\alpha, x_2)) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi dans w $\mathbb{E}$. souhaite que w appartienne à l'ensemble de tous les mondes w' où $\mathbb{E}$. épouse $g(x_2)$.
- $\llbracket \lambda x_2 \text{ vouloir}(\alpha, \text{^ép}(\alpha, x_2)) \rrbracket^{\mathcal{M}, w, g}$ = l'ensemble de tous les individus qu' $\mathbb{E}$. souhaite épouser dans w .

Lecture de re

Détail

- $\llbracket \text{ép}(\alpha, x_2) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi $\mathbb{E}$. épouse $g(x_2)$ dans w .
- $\llbracket \wedge \text{ép}(\alpha, x_2) \rrbracket^{\mathcal{M}, w, g} =$ l'ensemble de tous les mondes w' où $\mathbb{E}$. épouse $g(x_2)$.
- $\llbracket \text{vouloir}(\alpha, \wedge \text{ép}(\alpha, x_2)) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi dans w $\mathbb{E}$. souhaite que w appartienne à l'ensemble de tous les mondes w' où $\mathbb{E}$. épouse $g(x_2)$.
- $\llbracket \lambda x_2 \text{ vouloir}(\alpha, \wedge \text{ép}(\alpha, x_2)) \rrbracket^{\mathcal{M}, w, g} =$ l'ensemble de tous les individus qu' $\mathbb{E}$. souhaite épouser dans w .
- $\llbracket \iota z \text{ mère}(z, \alpha) \rrbracket^{\mathcal{M}, w, g} =$ celle qui est la mère d' $\mathbb{E}$. dans w .

Lecture de re

Détail

- $\llbracket \text{ép}(\alpha, x_2) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi $\mathbb{E}$. épouse $g(x_2)$ dans w .
- $\llbracket \wedge \text{ép}(\alpha, x_2) \rrbracket^{\mathcal{M}, w, g} =$ l'ensemble de tous les mondes w' où $\mathbb{E}$. épouse $g(x_2)$.
- $\llbracket \text{vouloir}(\alpha, \wedge \text{ép}(\alpha, x_2)) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi dans w $\mathbb{E}$. souhaite que w appartienne à l'ensemble de tous les mondes w' où $\mathbb{E}$. épouse $g(x_2)$.
- $\llbracket \lambda x_2 \text{ vouloir}(\alpha, \wedge \text{ép}(\alpha, x_2)) \rrbracket^{\mathcal{M}, w, g} =$ l'ensemble de tous les individus qu' $\mathbb{E}$. souhaite épouser dans w .
- $\llbracket [\lambda x_2 \text{ vouloir}(\alpha, \wedge \text{ép}(\alpha, x_2))](\iota z \text{ mère}(z, \alpha)) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi $\mathbb{E}$. souhaite épouser dans w celle qui est la mère d' $\mathbb{E}$. dans w .

Lecture de re

Détail

- $\llbracket \text{ép}(\alpha, x_2) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi $\mathbb{E}$. épouse $g(x_2)$ dans w .
- $\llbracket \wedge \text{ép}(\alpha, x_2) \rrbracket^{\mathcal{M}, w, g} =$ l'ensemble de tous les mondes w' où $\mathbb{E}$. épouse $g(x_2)$.
- $\llbracket \text{vouloir}(\alpha, \wedge \text{ép}(\alpha, x_2)) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi dans w $\mathbb{E}$. souhaite que w appartienne à l'ensemble de tous les mondes w' où $\mathbb{E}$. épouse $g(x_2)$.
- $\llbracket \lambda x_2 \text{ vouloir}(\alpha, \wedge \text{ép}(\alpha, x_2)) \rrbracket^{\mathcal{M}, w, g} =$ l'ensemble de tous les individus qu' $\mathbb{E}$. souhaite épouser dans w .
- $\llbracket [\lambda x_2 \text{ vouloir}(\alpha, \wedge \text{ép}(\alpha, x_2))](\iota z \text{ mère}(z, \alpha)) \rrbracket^{\mathcal{M}, w, g} = 1$ ssi $\mathbb{E}$. souhaite épouser dans w celle qui est la mère d' $\mathbb{E}$. dans w .

Ici la β -réduction est interdite, car x_2 est dans la portée de $\wedge$

Référence