Formal Semantics of Natural Language

Philippe de Groote and
Yoad Winter

ESSLLI 2021, Online

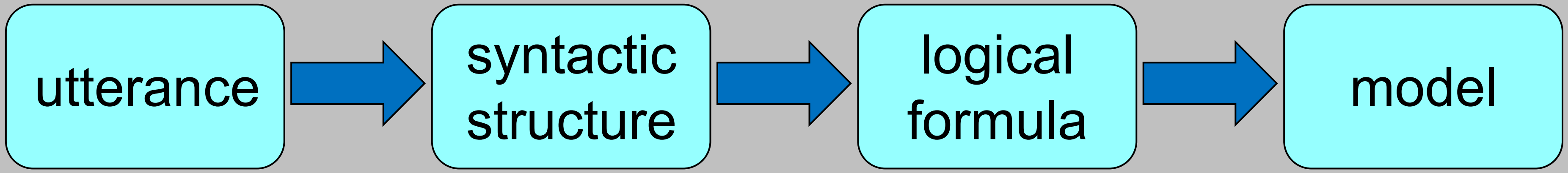
Topic 5

Quantification

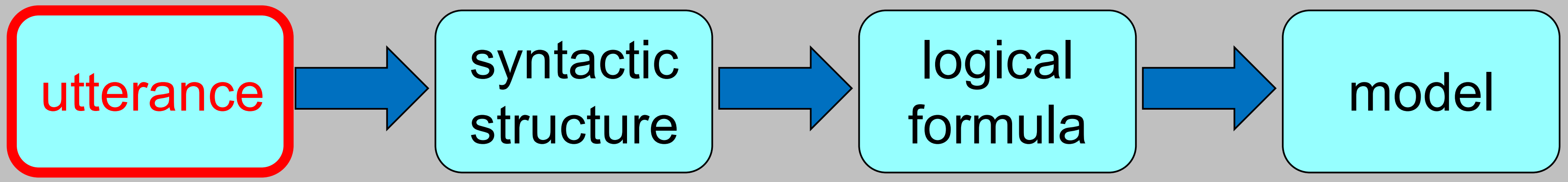
Quantificational expressions

- *Everybody* praised Mary.
- *Everybody but Tina* praised Mary.
- One can find it *everywhere*.
- John *rarely* wears a cap.
- John *most often* wears a cap.
- We are *far from* Beijing.
- There is *a lot of* work to do today.
- *Everybody* needs *some* help *sometimes*.
- *Some* representatives of *every* department in *most* companies saw *a few* samples of *every* product

Taking stock

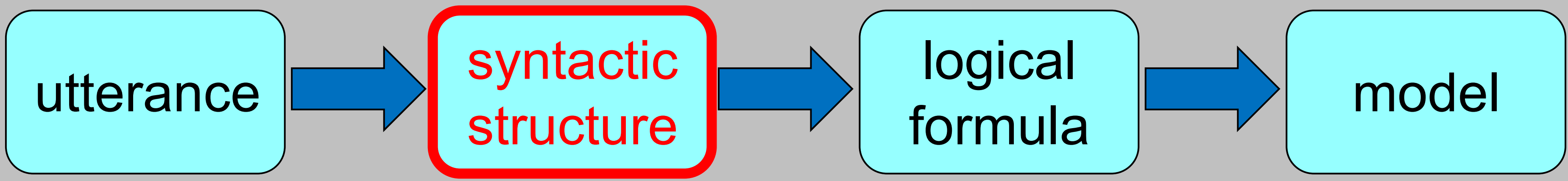


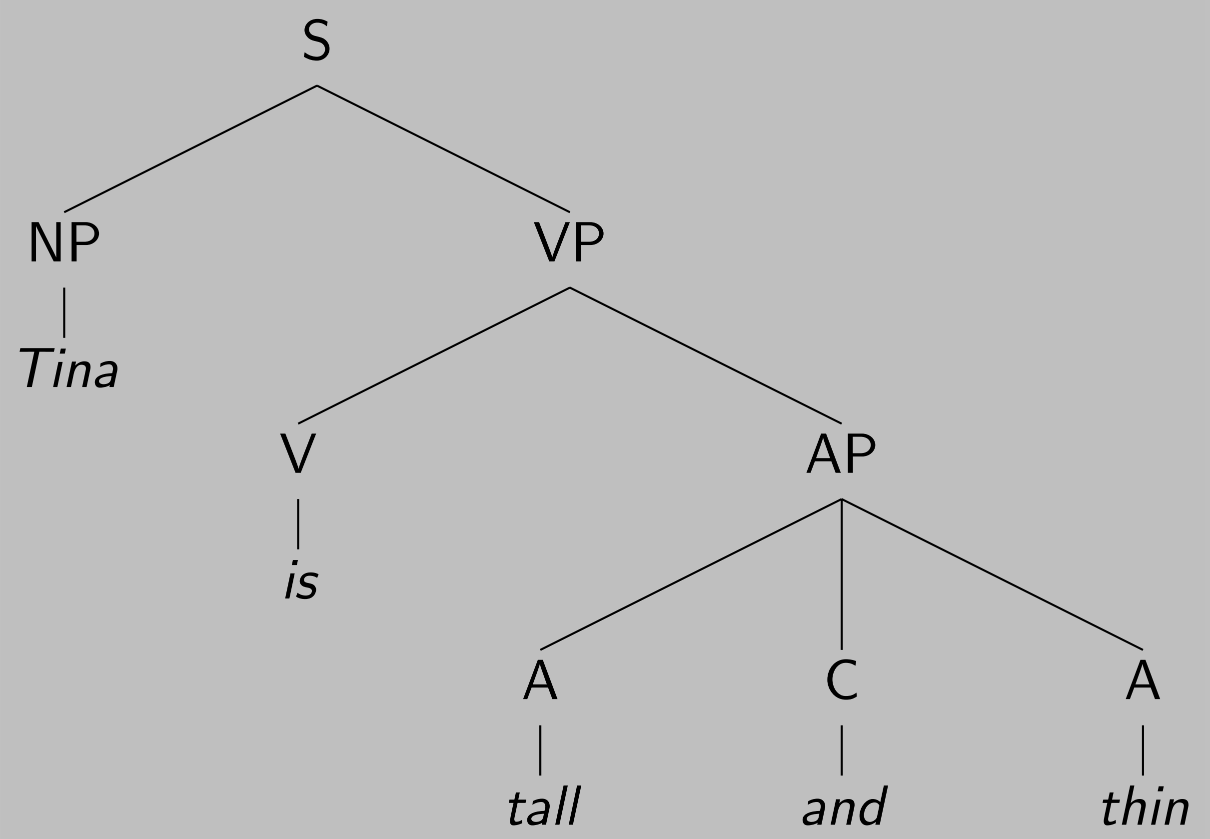
Taking stock



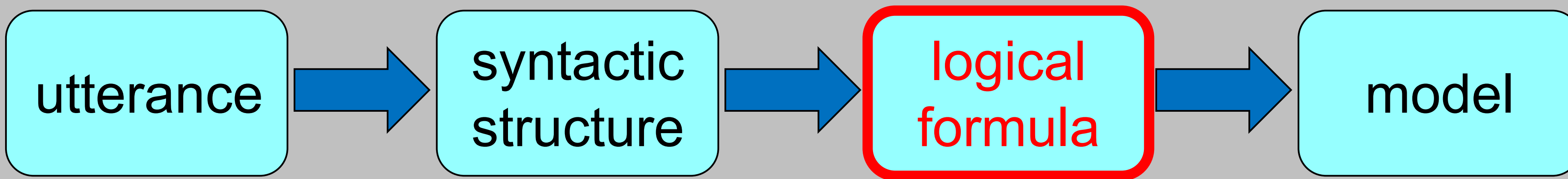
Tina is tall and thin

Taking stock



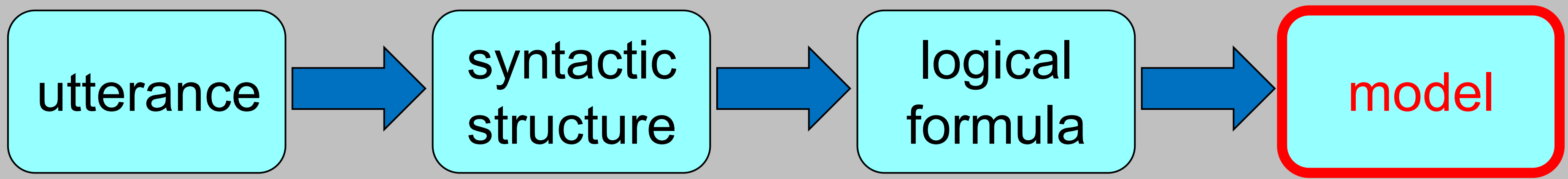


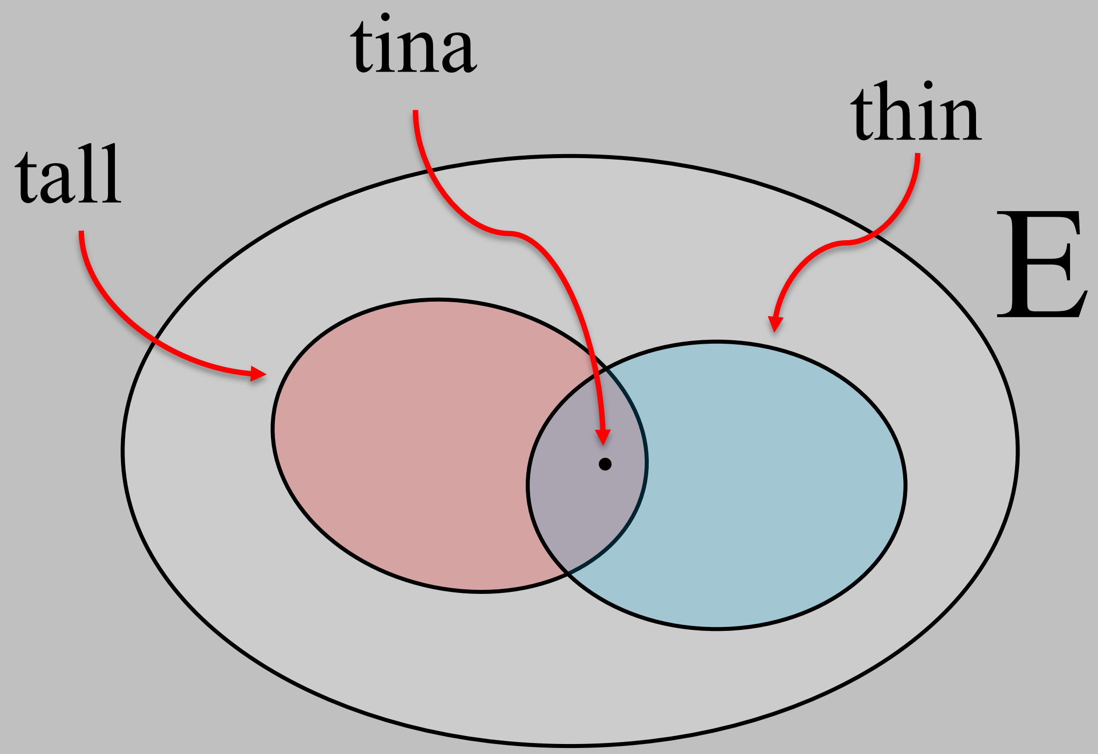
Taking stock



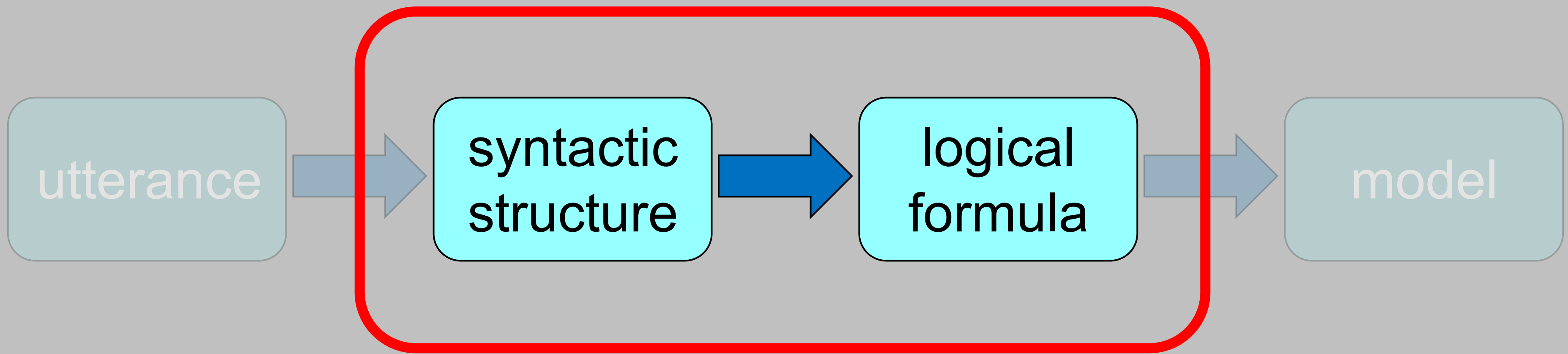
$$
\begin{aligned}
\textsc{is}\,(\textsc{and tall thin})\,\textsc{tina} &= (\lambda p x.\, p\, x)\,(\textsc{and tall thin})\,\textsc{tina} \\
&\to_\beta (\lambda x.\, \textsc{and tall thin}\; x)\,\textsc{tina} \\
&\to_\beta \textsc{and tall thin tina} \\
&= (\lambda p q x.\, (p\, x) \wedge (q\, x))\,\textsc{tall thin tina} \\
&\to_\beta (\lambda q x.\, (\textsc{tall}\; x) \wedge (q\, x))\,\textsc{thin tina} \\
&= (\lambda q x.\, ((\lambda x.\, \mathbf{tall}\, x)\, x) \wedge (q\, x))\,\textsc{thin tina} \\
&\to_\beta (\lambda q x.\, (\mathbf{tall}\, x) \wedge (q\, x))\,\textsc{thin tina} \\
&\to_\beta (\lambda x.\, (\mathbf{tall}\, x) \wedge (\textsc{thin}\; x))\,\textsc{tina} \\
&= (\lambda x.\, (\mathbf{tall}\, x) \wedge ((\lambda x.\, \mathbf{thin}\, x)\, x))\,\textsc{tina} \\
&\to_\beta (\lambda x.\, (\mathbf{tall}\, x) \wedge (\mathbf{thin}\, x))\,\textsc{tina} \\
&\to_\beta (\mathbf{tall}\; \textsc{tina}) \wedge (\mathbf{thin}\; \textsc{tina}) \\
&= (\mathbf{tall}\, \mathbf{tina}) \wedge (\mathbf{thin}\, \mathbf{tina})
\end{aligned}
$$

Taking stock

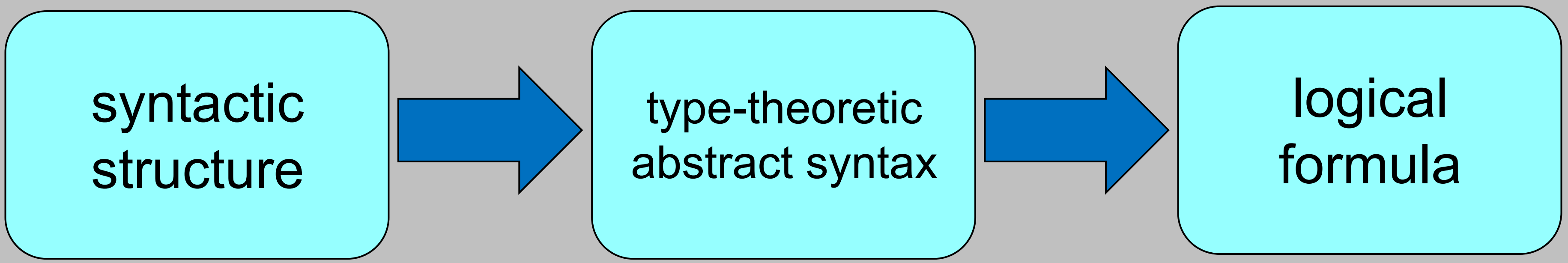




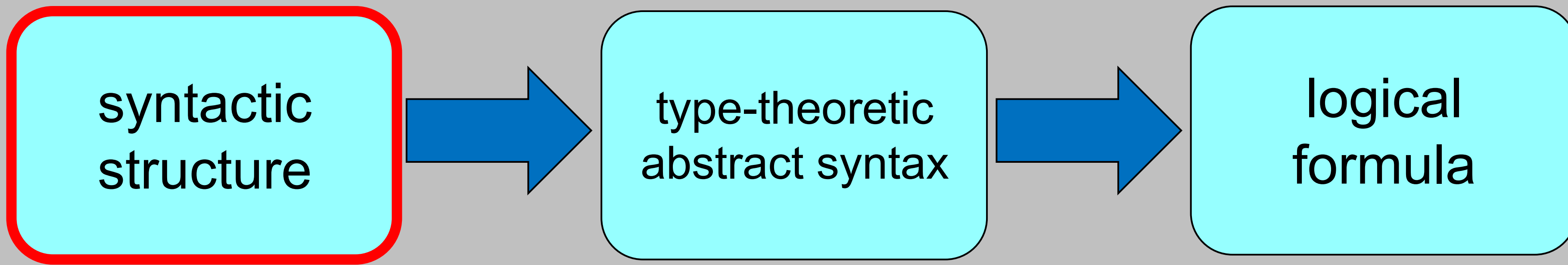
Syntax-semantics interface

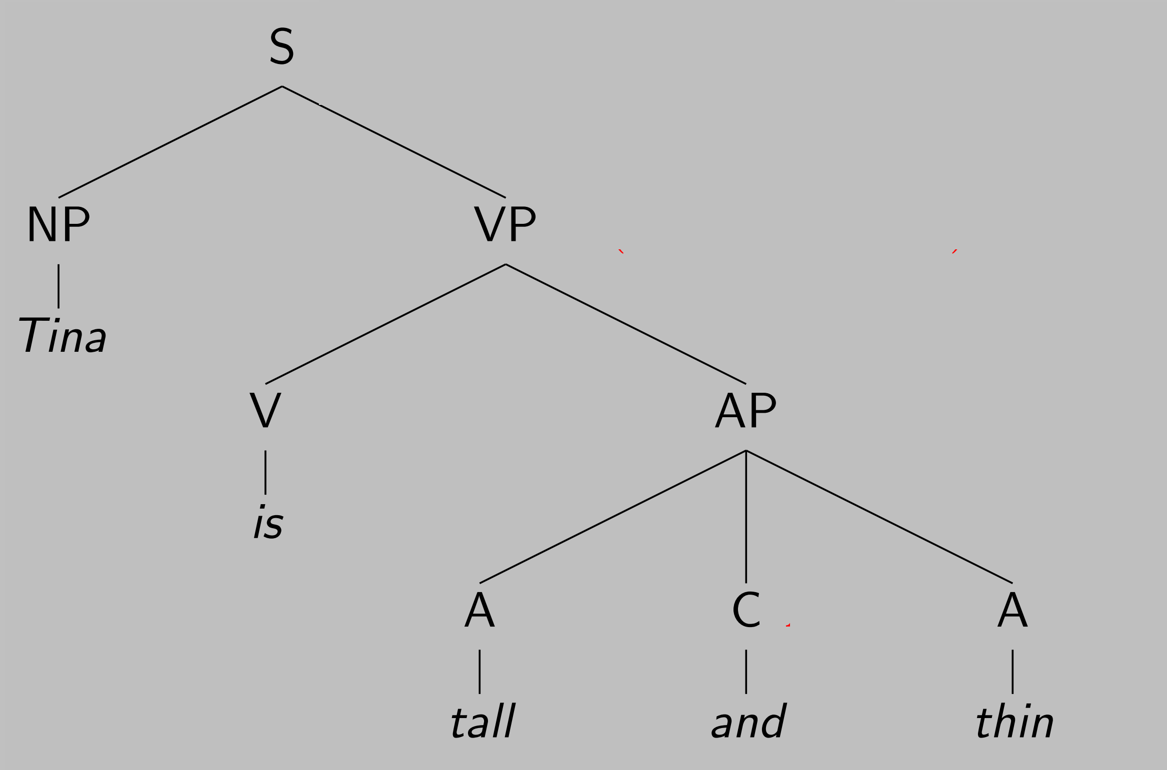


Syntax-semantics interface

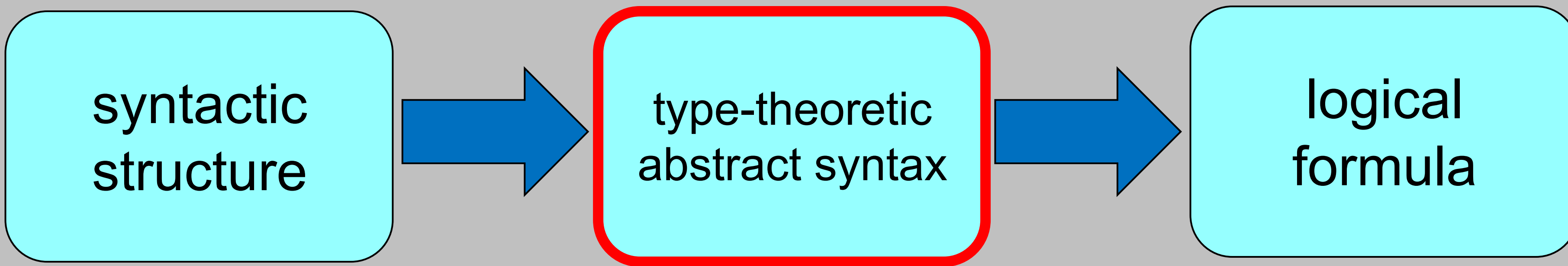


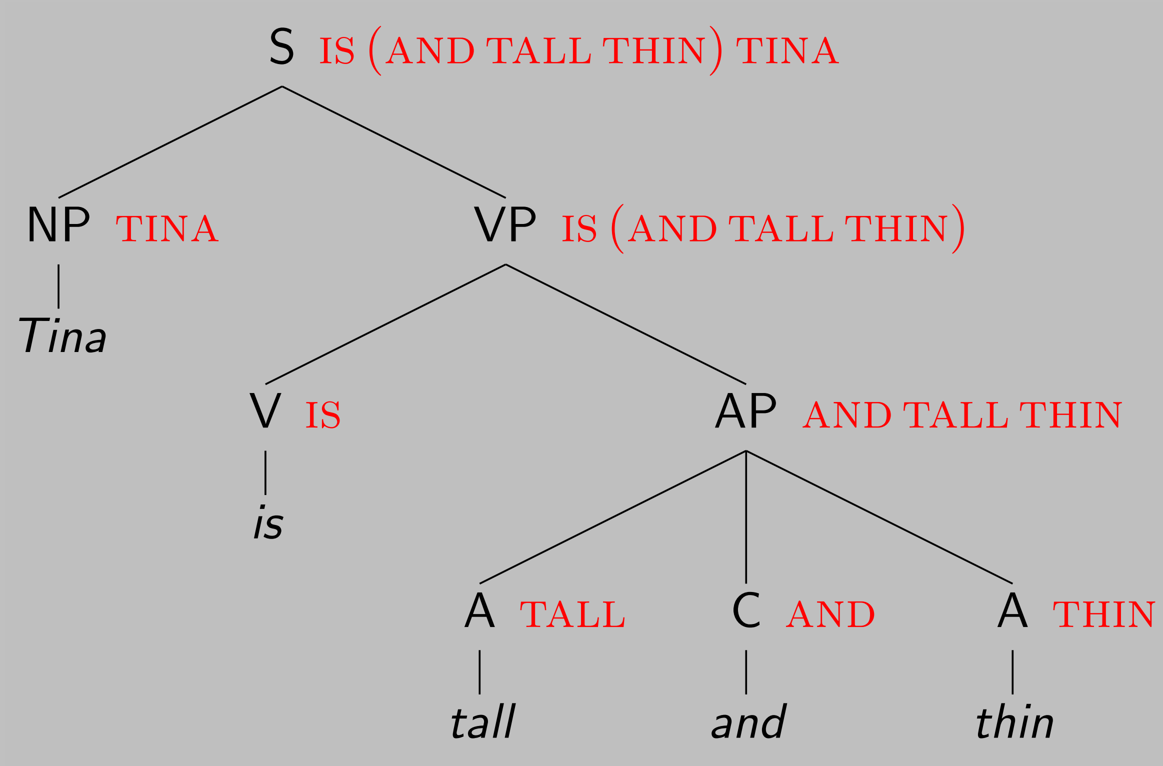
Syntax-semantics interface



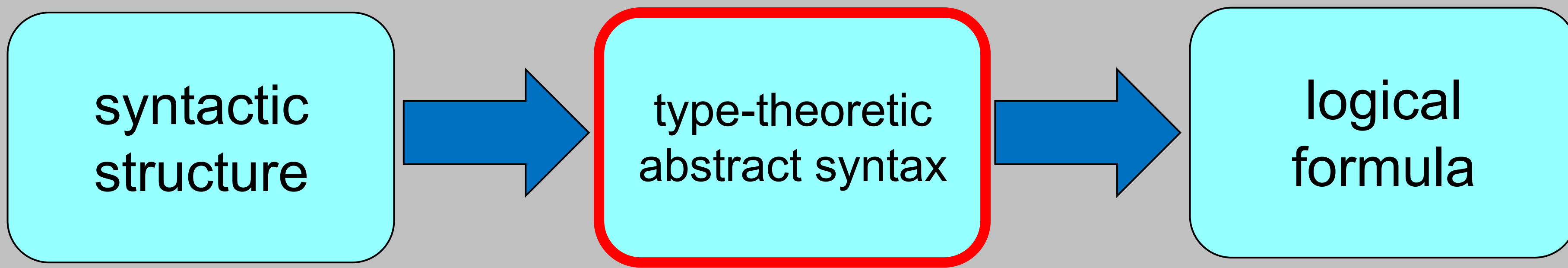


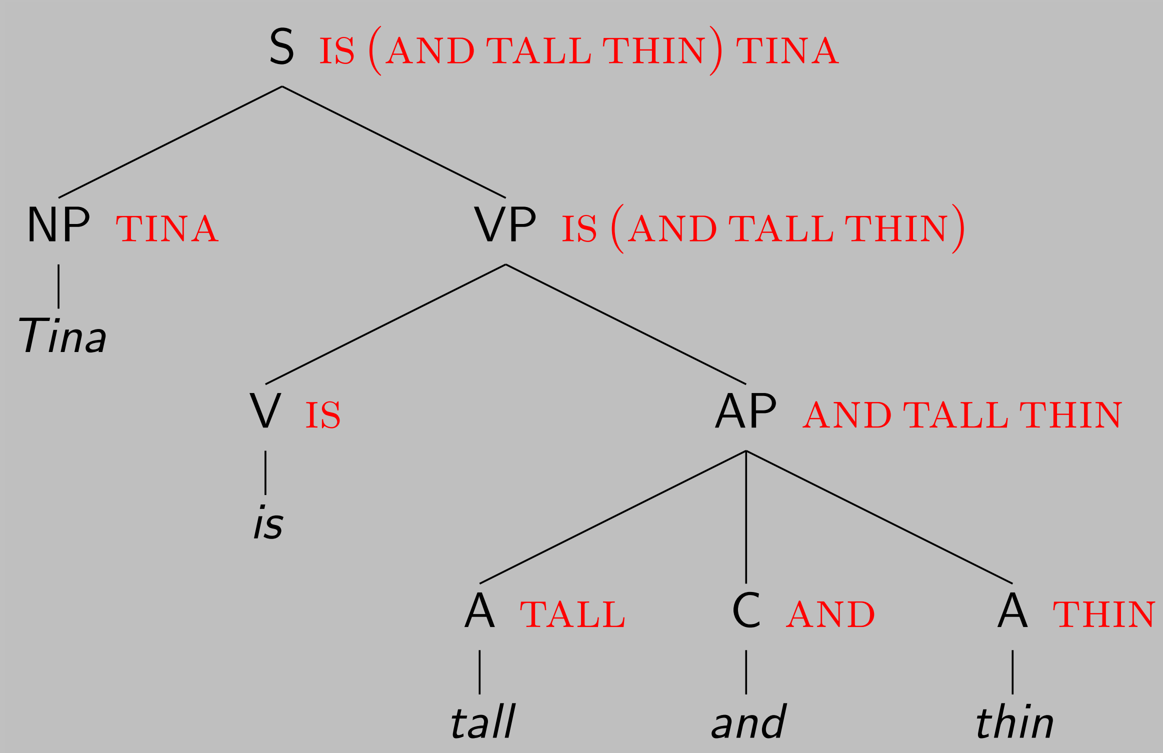
Syntax-semantics interface





Syntax-semantics interface





IS : AP (NP S)

AND : AP (AP AP)

TALL : AP

THIN : AP

TINA : NP

Syntax-semantics interface

$$\begin{aligned}
\text{IS} &: \text{AP (NP S)} \\
\text{AND} &: \text{AP (AP AP)} \\
\text{TALL} &: \text{AP} \\
\text{THIN} &: \text{AP} \\
\text{TINA} &: \text{NP}
\end{aligned}$$

Syntax-semantics interface

$$\textsc{is} : \mathrm{AP}\ (\mathrm{NP}\ \mathrm{S})$$
$$\textsc{and} : \mathrm{AP}\ (\mathrm{AP}\ \mathrm{AP})$$
$$\textsc{tall} : \mathrm{AP}$$
$$\textsc{thin} : \mathrm{AP}$$
$$\textsc{tina} : \mathrm{NP}$$

$$\mathrm{S} := \mathbf{t}$$
$$\mathrm{NP} := \mathbf{e}$$
$$\mathrm{AP} := \mathbf{e}\,\mathbf{t}$$

Syntax-semantics interface

$$\begin{aligned}
\textsc{is} &: \mathrm{AP}\ (\mathrm{NP}\ \mathrm{S}) \\
\textsc{and} &: \mathrm{AP}\ (\mathrm{AP}\ \mathrm{AP}) \\
\textsc{tall} &: \mathrm{AP} \\
\textsc{thin} &: \mathrm{AP} \\
\textsc{tina} &: \mathrm{NP}
\end{aligned}$$

$$\begin{aligned}
\mathrm{S} &:= \mathbf{t} \\
\mathrm{NP} &:= \mathbf{e} \\
\mathrm{AP} &:= \mathbf{e}\,\mathbf{t}
\end{aligned}$$

$$\begin{aligned}
\textsc{tina} &:= \mathbf{tina} \\
\textsc{tall} &:= \lambda x.\, \mathbf{tall}\, x \\
\textsc{thin} &:= \lambda x.\, \mathbf{thin}\, x \\
\textsc{and} &:= \lambda pqx.\, (p\, x) \wedge (q\, x) \\
\textsc{is} &:= \lambda px.\, p\, x
\end{aligned}$$

Noun phrases (naive interpretation)

Noun phrases (naive interpretation)

Abstract syntax:

$$\begin{aligned} \textsc{Tina} &: \mathrm{NP} \\ \textsc{Mary} &: \mathrm{NP} \\ \textsc{praised} &: \mathrm{NP}\ \mathrm{NP}\ \mathrm{S} \end{aligned}$$

Noun phrases (naive interpretation)

Abstract syntax:

$$\begin{aligned} \textsc{Tina} &: \mathrm{NP} \\ \textsc{Mary} &: \mathrm{NP} \\ \textsc{praised} &: \mathrm{NP}\;\mathrm{NP}\;\mathrm{S} \end{aligned}$$

Semantic interpretation:

Noun phrases (naive interpretation)

Abstract syntax:

$$\begin{aligned} \textsc{Tina} &: \mathrm{NP} \\ \textsc{Mary} &: \mathrm{NP} \\ \textsc{praised} &: \mathrm{NP}\,\mathrm{NP}\,\mathrm{S} \end{aligned}$$

Semantic interpretation:

$$\begin{aligned} \mathrm{NP} &:= \mathbf{e} \\ \mathrm{S} &:= \mathbf{t} \end{aligned}$$

Noun phrases (naive interpretation)

Abstract syntax:

$$\begin{aligned} \textsc{Tina} &: \mathrm{NP} \\ \textsc{Mary} &: \mathrm{NP} \\ \textsc{praised} &: \mathrm{NP}\,\mathrm{NP}\,\mathrm{S} \end{aligned}$$

Semantic interpretation:

$$\begin{aligned} \mathrm{NP} &:= \mathbf{e} \\ \mathrm{S} &:= \mathbf{t} \end{aligned}$$

$$\begin{aligned} \textsc{Tina} &:= \mathbf{tina} \\ \textsc{Mary} &:= \mathbf{mary} \\ \textsc{praised} &:= \lambda xy.\,\mathbf{praised}\,y\,x \end{aligned}$$

where:

$$\begin{aligned} \mathbf{tina}, \mathbf{mary} &: \mathbf{e} \\ \mathbf{praised} &: \mathbf{e}\,\mathbf{e}\,\mathbf{t} \end{aligned}$$

Quantified noun phrases

- Tina praised Mary.
- Everybody praised Mary.
- Nobody praised Mary.
- Tina praised somebody.
- Everybody praised somebody.
- Everybody ran.

Generalized quantifiers & type raising

Generalized quantifiers & type raising

- Remember that **(e t) t** is the type of the logical constants **all** (∀) and **exists** (∃).

Generalized quantifiers & type raising

- Remember that **(e t) t** is the type of the logical constants **all** (∀) and **exists** (∃).
- Every term of type **(e t) t** is called a generalized quantifier.

Generalized quantifiers & type raising

- Remember that **(e t) t** is the type of the logical constants **all** (∀) and **exists** (∃).
- Every term of type **(e t) t** is called a generalized quantifier.
- Semantically, a generalized quantifier corresponds to a set of sets of entities.

Generalized quantifiers & type raising

- Remember that **(e t) t** is the type of the logical constants **all** ($\forall$) and **exists** ($\exists$).
- Every term of type **(e t) t** is called a generalized quantifier.
- Semantically, a generalized quantifier corresponds to a set of sets of entities.
- The expected meaning of "*everybody ran*" might be captured by the following formula:

$$\forall x.\,(\mathbf{human}\ x) \rightarrow (\mathbf{ran}\ x)$$

Generalized quantifiers & type raising

- Remember that $(\mathbf{e}\ \mathbf{t})\ \mathbf{t}$ is the type of the logical constants **all** ($\forall$) and **exists** ($\exists$).
- Every term of type $(\mathbf{e}\ \mathbf{t})\ \mathbf{t}$ is called a generalized quantifier.
- Semantically, a generalized quantifier corresponds to a set of sets of entities.
- The expected meaning of "*everybody ran*" might be captured by the following formula:

$$\forall x.\, (\mathbf{human}\ x) \longrightarrow (\mathbf{ran}\ x)$$

- Accordingly, the following $\boldsymbol{\lambda}$-term, which is of type $(\mathbf{e}\ \mathbf{t})\ \mathbf{t}$, is a good candidate for the interpretation of "*everybody*":

$$\boldsymbol{\lambda} p.\, \forall x.\, (\mathbf{human}\ x) \longrightarrow (p\ x)$$

Generalized quantifiers & type raising

Syntax/semantics interface:

$$\begin{aligned}
\textsc{tina} &: \mathrm{NP} \\
\textsc{mary} &: \mathrm{NP} \\
\textsc{everybody} &: \mathrm{NP} \\
\textsc{somebody} &: \mathrm{NP} \\
\textsc{ran} &: \mathrm{NP}\,\mathrm{S} \\
\textsc{praised} &: \mathrm{NP}\,\mathrm{NP}\,\mathrm{S}
\end{aligned}$$

Semantic interpretation:

$$\begin{aligned}
\mathrm{NP} &:= (\mathbf{e}\,\mathbf{t})\,\mathbf{t} \\
\mathrm{S} &:= \mathbf{t}
\end{aligned}$$

$$\begin{aligned}
\textsc{tina} &:= ? \\
\textsc{mary} &:= ? \\
\textsc{everybody} &:= \lambda k.\,\forall x.\,(\mathbf{human}\,x) \to (k\,x) \\
\textsc{somebody} &:= \lambda k.\,\exists x.\,(\mathbf{human}\,x) \wedge (k\,x) \\
\textsc{ran} &:= ? \\
\textsc{praised} &:= ?
\end{aligned}$$

Proper names as generalized quantifiers

Proper names as generalized quantifiers

- The interpretations of TINA and MARY must be of type **(e t) t.**

Proper names as generalized quantifiers

- The interpretations of TINA and MARY must be of type **(e t) t.**
- Semantically, it means that we must characterize an entity using a set of sets of entities.

Proper names as generalized quantifiers

- The interpretations of TINA and MARY must be of type **(e t) t.**
- Semantically, it means that we must characterize an entity using a set of sets of entities.
- $\{\, S \in \mathcal{P}(\mathrm{E}) : \textbf{tina} \in S \,\}$

Proper names as generalized quantifiers

- The interpretations of TINA and MARY must be of type **(e t) t.**
- Semantically, it means that we must characterize an entity using a set of sets of entities.
- $\{ S \in \mathcal{P}(\mathrm{E}) : \mathbf{tina} \in S \}$
- $\boldsymbol{\lambda} S_{(\mathbf{e\ t})}.\ S\ \mathbf{tina}$

Applying type-raising to verb arguments

Applying type-raising to verb arguments

- The syntactic type of RAN is (NP S), the semantic interpretation of NP is **(e t) t**, and the one of S is **t**. Accordingly the type of the interpretation of RAN must be **((e t) t) t**

Applying type-raising to verb arguments

- The syntactic type of RAN is (NP S), the semantic interpretation of NP is **(e t) t**, and the one of S is **t**. Accordingly the type of the interpretation of RAN must be **((e t) t) t**

- RAN := $\boldsymbol{\lambda} s.\ s\ (\boldsymbol{\lambda} x.\ \mathbf{ran}\ x)$

Applying type-raising to verb arguments

- The syntactic type of RAN is (NP S), the semantic interpretation of NP is **(e t) t**, and the one of S is **t**. Accordingly the type of the interpretation of RAN must be **((e t) t) t**

- RAN := $\boldsymbol{\lambda} s.\, s\, (\boldsymbol{\lambda} x.\, \mathbf{ran}\ x)$

- Similarly, the type of the interpretation of PRAISED must be **((e t) t) ((e t) t) t**.

Applying type-raising to verb arguments

- The syntactic type of RAN is (NP S), the semantic interpretation of NP is **(e t) t**, and the one of S is **t**. Accordingly the type of the interpretation of RAN must be **((e t) t) t**
- RAN := $\boldsymbol{\lambda} s.\ s\ (\boldsymbol{\lambda} x.\ \textbf{ran}\ x)$
- Similarly, the type of the interpretation of PRAISED must be **((e t) t) ((e t) t) t**.
- PRAISED := $\boldsymbol{\lambda} os.\ s\ (\boldsymbol{\lambda} x.\ o\ (\boldsymbol{\lambda} y.\ \textbf{praised}\ x\ y))$

Generalized quantifiers & type raising

Syntax/semantics interface:

$$\begin{aligned}
\textsc{tina} &: \mathrm{NP} \\
\textsc{mary} &: \mathrm{NP} \\
\textsc{everybody} &: \mathrm{NP} \\
\textsc{somebody} &: \mathrm{NP} \\
\textsc{ran} &: \mathrm{NP}\,\mathrm{S} \\
\textsc{praised} &: \mathrm{NP}\,\mathrm{NP}\,\mathrm{S}
\end{aligned}$$

Semantic interpretation:

$$\begin{aligned}
\mathrm{NP} &:= (\mathbf{e}\,\mathbf{t})\,\mathbf{t} \\
\mathrm{S} &:= \mathbf{t}
\end{aligned}$$

$$\begin{aligned}
\textsc{tina} &:= \lambda k.\, k\,\mathbf{tina} \\
\textsc{mary} &:= \lambda k.\, k\,\mathbf{mary} \\
\textsc{everybody} &:= \lambda k.\, \forall x.\, (\mathbf{human}\,x) \to (k\,x) \\
\textsc{somebody} &:= \lambda k.\, \exists x.\, (\mathbf{human}\,x) \wedge (k\,x) \\
\textsc{ran} &:= \lambda s.\, s\,(\lambda x.\, \mathbf{ran}\,x) \\
\textsc{praised} &:= \lambda o.\, \lambda s.\, s\,(\lambda x.\, o\,(\lambda y.\, \mathbf{praised}\,x\,y))
\end{aligned}$$

Generalized quantifiers & type raising

Tina praised somebody.

Generalized quantifiers & type raising

Tina praised somebody.

PRAISED SOMEBODY TINA

Generalized quantifiers & type raising

Tina praised somebody.

$$\textsc{praised somebody tina}$$

$$= \quad (\lambda o.\, \lambda s.\, s\, (\lambda x.\, o\, (\lambda y.\, \mathbf{praised}\, x\, y)))\ \textsc{somebody tina}$$

Generalized quantifiers & type raising

Tina praised somebody.

$$
\begin{aligned}
&\textsc{praised}\ \textsc{somebody}\ \textsc{tina} \\
=\quad &(\lambda o.\, \lambda s.\, s\,(\lambda x.\, o\,(\lambda y.\, \mathbf{praised}\, x\, y)))\ \textsc{somebody}\ \textsc{tina} \\
\to_\beta\quad &(\lambda s.\, s\,(\lambda x.\, \textsc{somebody}\,(\lambda y.\, \mathbf{praised}\, x\, y)))\ \textsc{tina}
\end{aligned}
$$

Generalized quantifiers & type raising

Tina praised somebody.

$$
\begin{aligned}
&\textsc{praised somebody tina}\\
&\quad = \quad (\lambda o.\, \lambda s.\, s\,(\lambda x.\, o\,(\lambda y.\, \mathbf{praised}\,x\,y)))\ \textsc{somebody tina}\\
&\quad \to_\beta \quad (\lambda s.\, s\,(\lambda x.\, \textsc{somebody}\,(\lambda y.\, \mathbf{praised}\,x\,y)))\ \textsc{tina}\\
&\quad \to_\beta \quad \textsc{tina}\,(\lambda x.\, \textsc{somebody}\,(\lambda y.\, \mathbf{praised}\,x\,y))
\end{aligned}
$$

Generalized quantifiers & type raising

Tina praised somebody.

$$\begin{aligned}
&\textsc{praised somebody tina}\\
&\quad = \quad (\lambda o.\, \lambda s.\, s\,(\lambda x.\, o\,(\lambda y.\, \mathbf{praised}\, x\, y)))\ \textsc{somebody tina}\\
&\quad \to_\beta \quad (\lambda s.\, s\,(\lambda x.\, \textsc{somebody}\,(\lambda y.\, \mathbf{praised}\, x\, y)))\ \textsc{tina}\\
&\quad \to_\beta \quad \textsc{tina}\,(\lambda x.\, \textsc{somebody}\,(\lambda y.\, \mathbf{praised}\, x\, y))\\
&\quad = \quad (\lambda k.\, k\, \mathbf{tina})\,(\lambda x.\, \textsc{somebody}\,(\lambda y.\, \mathbf{praised}\, x\, y))
\end{aligned}$$

Generalized quantifiers & type raising

Tina praised somebody.

$$
\begin{aligned}
&\textsc{praised somebody tina} \\
&\quad = \quad (\lambda o.\, \lambda s.\, s\,(\lambda x.\, o\,(\lambda y.\, \mathbf{praised}\, x\, y)))\ \textsc{somebody tina} \\
&\quad \to_\beta \quad (\lambda s.\, s\,(\lambda x.\, \textsc{somebody}\,(\lambda y.\, \mathbf{praised}\, x\, y)))\ \textsc{tina} \\
&\quad \to_\beta \quad \textsc{tina}\,(\lambda x.\, \textsc{somebody}\,(\lambda y.\, \mathbf{praised}\, x\, y)) \\
&\quad = \quad (\lambda k.\, k\, \mathbf{tina})\,(\lambda x.\, \textsc{somebody}\,(\lambda y.\, \mathbf{praised}\, x\, y)) \\
&\quad \to_\beta \quad (\lambda x.\, \textsc{somebody}\,(\lambda y.\, \mathbf{praised}\, x\, y))\ \mathbf{tina}
\end{aligned}
$$

Generalized quantifiers & type raising

Tina praised somebody.

$$\textsc{praised}\ \textsc{somebody}\ \textsc{tina}$$

$$\begin{aligned}
&= (\lambda o.\,\lambda s.\,s\,(\lambda x.\,o\,(\lambda y.\,\mathbf{praised}\,x\,y)))\ \textsc{somebody}\ \textsc{tina} \\
&\to_\beta (\lambda s.\,s\,(\lambda x.\,\textsc{somebody}\,(\lambda y.\,\mathbf{praised}\,x\,y)))\ \textsc{tina} \\
&\to_\beta \textsc{tina}\,(\lambda x.\,\textsc{somebody}\,(\lambda y.\,\mathbf{praised}\,x\,y)) \\
&= (\lambda k.\,k\,\mathbf{tina})\,(\lambda x.\,\textsc{somebody}\,(\lambda y.\,\mathbf{praised}\,x\,y)) \\
&\to_\beta (\lambda x.\,\textsc{somebody}\,(\lambda y.\,\mathbf{praised}\,x\,y))\,\mathbf{tina} \\
&\to_\beta \textsc{somebody}\,(\lambda y.\,\mathbf{praised}\,\mathbf{tina}\,y)
\end{aligned}$$

Generalized quantifiers & type raising

Tina praised somebody.

$$
\begin{aligned}
&\textsc{praised somebody tina}\\
&= (\lambda o.\, \lambda s.\, s\, (\lambda x.\, o\, (\lambda y.\, \mathbf{praised}\, x\, y)))\ \textsc{somebody tina}\\
&\to_\beta (\lambda s.\, s\, (\lambda x.\, \textsc{somebody}\, (\lambda y.\, \mathbf{praised}\, x\, y)))\ \textsc{tina}\\
&\to_\beta \textsc{tina}\, (\lambda x.\, \textsc{somebody}\, (\lambda y.\, \mathbf{praised}\, x\, y))\\
&= (\lambda k.\, k\, \mathbf{tina})\, (\lambda x.\, \textsc{somebody}\, (\lambda y.\, \mathbf{praised}\, x\, y))\\
&\to_\beta (\lambda x.\, \textsc{somebody}\, (\lambda y.\, \mathbf{praised}\, x\, y))\, \mathbf{tina}\\
&\to_\beta \textsc{somebody}\, (\lambda y.\, \mathbf{praised}\, \mathbf{tina}\, y)\\
&= (\lambda k.\, \exists x.\, (\mathbf{human}\, x) \wedge (k\, x))\, (\lambda y.\, \mathbf{praised}\, \mathbf{tina}\, y)
\end{aligned}
$$

Generalized quantifiers & type raising

Tina praised somebody.

$$
\begin{aligned}
&\textsc{praised somebody tina}\\
&= \quad (\lambda o.\, \lambda s.\, s\,(\lambda x.\, o\,(\lambda y.\, \mathbf{praised}\,x\,y)))\ \textsc{somebody tina}\\
&\to_\beta \quad (\lambda s.\, s\,(\lambda x.\, \textsc{somebody}\,(\lambda y.\, \mathbf{praised}\,x\,y)))\ \textsc{tina}\\
&\to_\beta \quad \textsc{tina}\,(\lambda x.\, \textsc{somebody}\,(\lambda y.\, \mathbf{praised}\,x\,y))\\
&= \quad (\lambda k.\, k\,\mathbf{tina})\,(\lambda x.\, \textsc{somebody}\,(\lambda y.\, \mathbf{praised}\,x\,y))\\
&\to_\beta \quad (\lambda x.\, \textsc{somebody}\,(\lambda y.\, \mathbf{praised}\,x\,y))\,\mathbf{tina}\\
&\to_\beta \quad \textsc{somebody}\,(\lambda y.\, \mathbf{praised}\,\mathbf{tina}\,y)\\
&= \quad (\lambda k.\, \exists x.\, (\mathbf{human}\,x) \wedge (k\,x))\,(\lambda y.\, \mathbf{praised}\,\mathbf{tina}\,y)\\
&\to_\beta \quad \exists x.\, (\mathbf{human}\,x) \wedge ((\lambda y.\, \mathbf{praised}\,\mathbf{tina}\,y)\,x)
\end{aligned}
$$

Generalized quantifiers & type raising

Tina praised somebody.

$$
\begin{aligned}
&\textsc{praised}\ \textsc{somebody}\ \textsc{tina} \\
=\quad & (\lambda o.\, \lambda s.\, s\, (\lambda x.\, o\, (\lambda y.\, \mathbf{praised}\, x\, y)))\ \textsc{somebody}\ \textsc{tina} \\
\to_\beta\quad & (\lambda s.\, s\, (\lambda x.\, \textsc{somebody}\, (\lambda y.\, \mathbf{praised}\, x\, y)))\ \textsc{tina} \\
\to_\beta\quad & \textsc{tina}\, (\lambda x.\, \textsc{somebody}\, (\lambda y.\, \mathbf{praised}\, x\, y)) \\
=\quad & (\lambda k.\, k\, \mathbf{tina})\, (\lambda x.\, \textsc{somebody}\, (\lambda y.\, \mathbf{praised}\, x\, y)) \\
\to_\beta\quad & (\lambda x.\, \textsc{somebody}\, (\lambda y.\, \mathbf{praised}\, x\, y))\, \mathbf{tina} \\
\to_\beta\quad & \textsc{somebody}\, (\lambda y.\, \mathbf{praised}\, \mathbf{tina}\, y) \\
=\quad & (\lambda k.\, \exists x.\, (\mathbf{human}\, x) \wedge (k\, x))\, (\lambda y.\, \mathbf{praised}\, \mathbf{tina}\, y) \\
\to_\beta\quad & \exists x.\, (\mathbf{human}\, x) \wedge ((\lambda y.\, \mathbf{praised}\, \mathbf{tina}\, y)\, x) \\
\to_\beta\quad & \exists x.\, (\mathbf{human}\, x) \wedge (\mathbf{praised}\, \mathbf{tina}\, x)
\end{aligned}
$$

Nouns & Determiners

Syntax/semantics interface:

$$\begin{aligned} \textsc{tina} &: \mathrm{NP} \\ \textsc{mary} &: \mathrm{NP} \\ \textsc{everybody} &: \mathrm{NP} \\ \textsc{somebody} &: \mathrm{NP} \\ \textsc{man} &: \mathrm{N} \\ \textsc{woman} &: \mathrm{N} \\ \textsc{every} &: \mathrm{N}\ \mathrm{NP} \\ \textsc{some} &: \mathrm{N}\ \mathrm{NP} \\ \textsc{ran} &: \mathrm{NP}\ \mathrm{S} \\ \textsc{praised} &: \mathrm{NP}\ \mathrm{NP}\ \mathrm{S} \end{aligned}$$

Semantic interpretation:

$$\begin{aligned} \mathrm{N} &:= \mathbf{e}\,\mathbf{t} \\ \mathrm{NP} &:= (\mathbf{e}\,\mathbf{t})\,\mathbf{t} \\ \mathrm{S} &:= \mathbf{t} \end{aligned}$$

Nouns & Determiners

Semantic interpretation:

$$
\begin{aligned}
\text{TINA} &:= \lambda k.\, k\, \mathbf{tina} \\
\text{MARY} &:= \lambda k.\, k\, \mathbf{mary} \\
\text{EVERYBODY} &:= \lambda k.\, \forall x.\, (\mathbf{human}\, x) \to (k\, x) \\
\text{SOMEBODY} &:= \lambda k.\, \exists x.\, (\mathbf{human}\, x) \wedge (k\, x) \\
\text{MAN} &:= \lambda x.\, \mathbf{man}\, x \\
\text{WOMAN} &:= \lambda x.\, \mathbf{woman}\, x \\
\text{EVERY} &:= \lambda n.\, \lambda m.\, \forall x.\, n\, x \to m\, x \\
\text{SOME} &:= \lambda n.\, \lambda m.\, \exists x.\, n\, x \wedge m\, x \\
\text{RAN} &:= \lambda s.\, s\, (\lambda x.\, \mathbf{ran}\, x) \\
\text{PRAISED} &:= \lambda o.\, \lambda s.\, s\, (\lambda x.\, o\, (\lambda y.\, \mathbf{praised}\, x\, y))
\end{aligned}
$$

where:

$$\mathbf{woman}, \mathbf{man} : \mathbf{e}\, \mathbf{t}$$

Determiners as binary generalized quantifiers

Determiners as binary generalized quantifiers

- The type of the interpretations of EVERY and SOME is **(e t) (e t) t**.

Determiners as binary generalized quantifiers

- The type of the interpretations of EVERY and SOME is **(e t) (e t) t**.
- Every term of type **(e t) (e t) t** is called a binary generalized quantifier.

Determiners as binary generalized quantifiers

- The type of the interpretations of EVERY and SOME is **(e t) (e t) t**.
- Every term of type **(e t) (e t) t** is called a binary generalized quantifier.
- Semantically, a binary generalized quantifier corresponds to a relation between two sets of entities.

Determiners as binary generalized quantifiers

SOME A B	$A \cap B \neq \emptyset$
EVERY A B	$A \subset B$
NO A B	$A \cap B = \emptyset$
(AT-LEAST n) A B	$\|A \cap B\| \geq n$
(AT-MOST n) A B	$\|A \cap B\| \leq n$
(EXACTLY n) A B	$\|A \cap B\| = n$
MOST A B	$\|A\| \leq 2 \times \|A \cap B\|$

Scope ambiguities

Scope ambiguities

Every man praised a woman

Scope ambiguities

Every man praised a woman

$$\forall x.\mathbf{man}\,x \to (\exists y.\mathbf{woman}\,y \land \mathbf{praised}\,x\,y)$$
$$\exists y.\mathbf{woman}\,y \land (\forall x.\mathbf{man}\,x \land \mathbf{praised}\,x\,y)$$

Scope ambiguities

Every man praised a woman

$$\forall x.\mathbf{man}\, x \to (\exists y.\mathbf{woman}\, y \wedge \mathbf{praised}\, x\, y)$$
$$\exists y.\mathbf{woman}\, y \wedge (\forall x.\mathbf{man}\, x \wedge \mathbf{praised}\, x\, y)$$

Subject wide scope:

$$\textsc{praised} = \lambda o.\, \lambda s.\, s\,(\lambda x.\, o\,(\lambda y.\, \mathbf{praised}\, x\, y))$$

Object wide scope:

$$\textsc{praised}_{\textsc{ows}} = \lambda o.\, \lambda s.\, o\,(\lambda y.\, s\,(\lambda x.\, \mathbf{praised}\, x\, y))$$