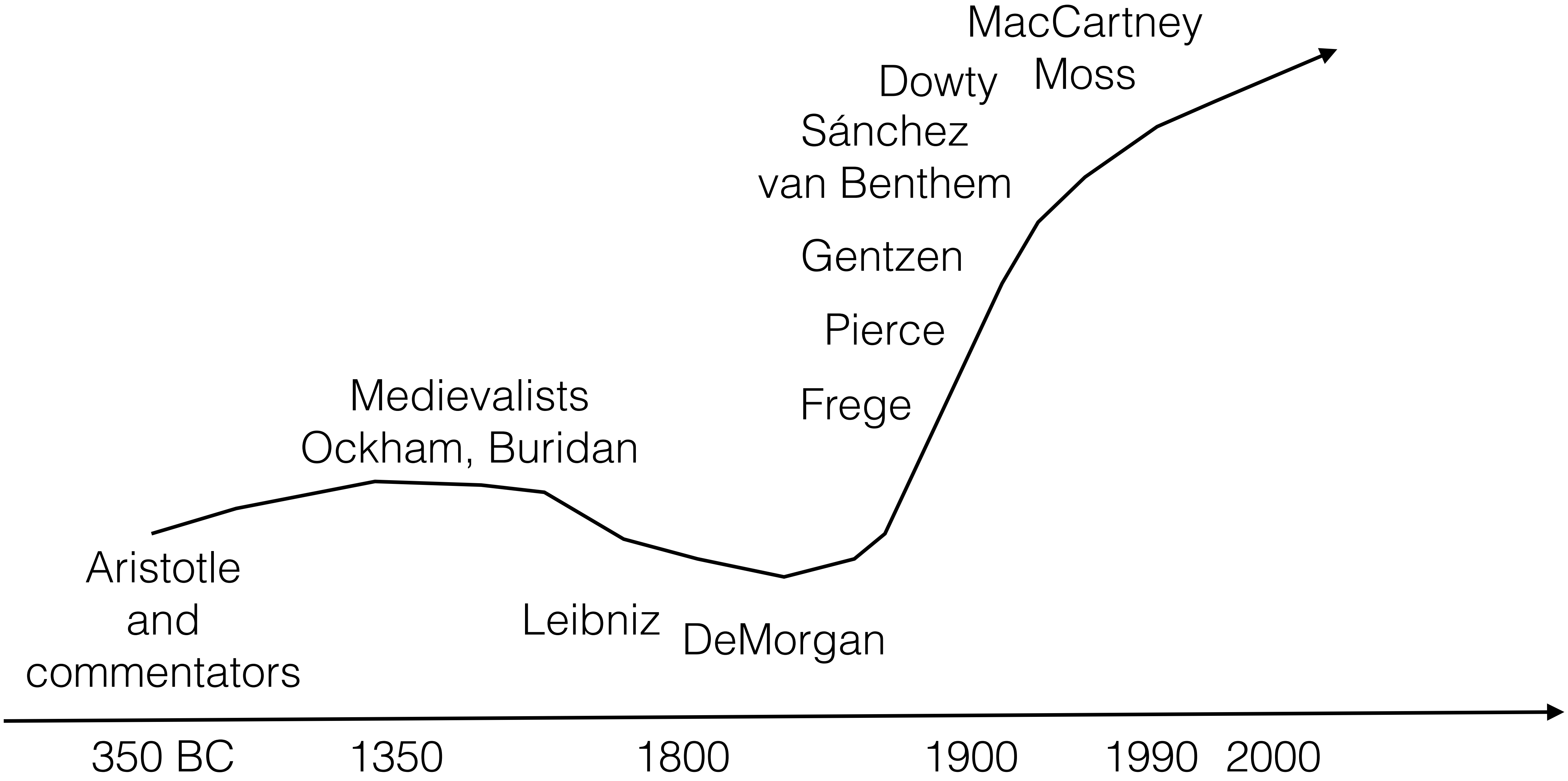
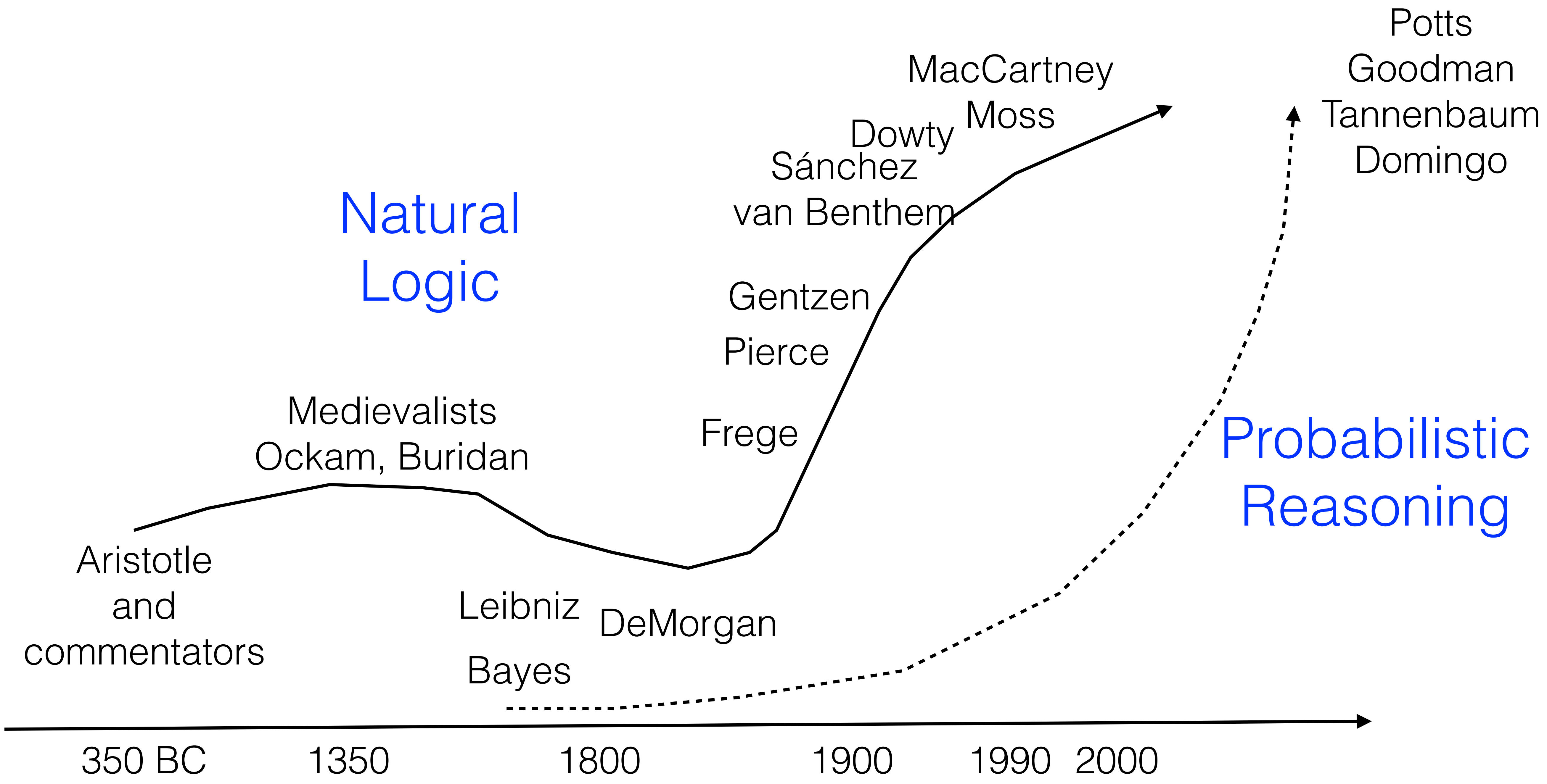
A Brief History of Natural Logic



MacCartney
Moss
Dowty
Sánchez
van Benthem
Gentzen
Pierce
Frege
Medievalists
Ockham, Buridan
Aristotle
and
commentators
Leibniz
DeMorgan

350 BC
1350
1800
1900
1990
2000

A Brief History of Reasoning



Natural Logic

Aristotle
and
commentators

Medievalists
Ockam, Buridan

Leibniz

Bayes

DeMorgan

Frege

Pierce

Gentzen

van Benthem

Sánchez

Dowty

MacCartney
Moss

Potts
Goodman
Tannenbaum
Domingo

Probabilistic Reasoning

350 BC
1350
1800
1900
1990
2000

Categorial Grammar

Is built on a recursive definition of types (categories):

Basic types:

e, **t**, **p**

Function types:

If α and β are types, so are α/β and α\β.

For example: **e**, **t** **e\t** **(e\t)/e**

Every well-formed phrase has a type.

Forward Application

α/β

Functor Argument

$$\xrightarrow[\alpha]{\alpha/\beta \qquad \beta} \text{FA}$$

Result

For example:

$$\xrightarrow[\mathbf{e\backslash t}]{\mathbf{(e\backslash t)/e \quad e}} \text{FA}$$

Backward Application

$\alpha\backslash\beta$

Argument Functor

$$\mathrm{BA} \xleftarrow[\alpha]{\beta \qquad \alpha\backslash\beta}$$

Result
on Left

Steedman, Baldridge
CCG

Argument Functor

$$\mathrm{BA} \xleftarrow[\beta]{\alpha \qquad \alpha\backslash\beta}$$

Result
on Top

Lambek Calculus

This is a matter of convention, not a theoretical issue. In the USA you drive on the right side of the road, in the UK on the left.That's just the way it is. In this set of slides we follow the "Result on Top" principle.

Two traditions of CG

- The first, *rule-based*, approach, pioneered by Lyons 1968, Bach 1976, Dowty 1979, among other linguists, and by Lewis 1970 and Geach 1972, among philosophical logicians, starts from the pure CG of Bar-Hillel, and adds rules corresponding to simple operations over categories, such as "wrap" (or commutation of arguments), "type-raising," (which resembles the application of traditional nominative, accusative etc. *case* to NPs etc.) and functional composition.

- The alternative, *deductive*, style of Categorial Grammar, pioneered by van Benthem 1986 and Moortgat 1988 takes as its starting point Lambek's syntactic calculus. The Lambek system embodies a view of the categorial slash as a form of logical implication for which a number of axioms or inference rules define a proof theory. (For example, functional application corresponds to the familiar classical rule of *Modus Ponens* under this view). A number of further axioms give rise to a deductive calculus in which many but not all of the rules deployed by the alternative rule-based generalizations of CG are theorems.

Mark Steedman, Categorial Grammar, p. 2-3

Nondirectional application

$\beta\to\alpha$

Argument Functor

$$\frac{\beta \qquad \beta\to\alpha}{\alpha}$$

Result

Functor Argument

$$\frac{\beta\to\alpha \qquad \beta}{\alpha}$$

Result

Van Benthem, Sánches’ LP,
“Lambek’s system L with Permutation”

Categories

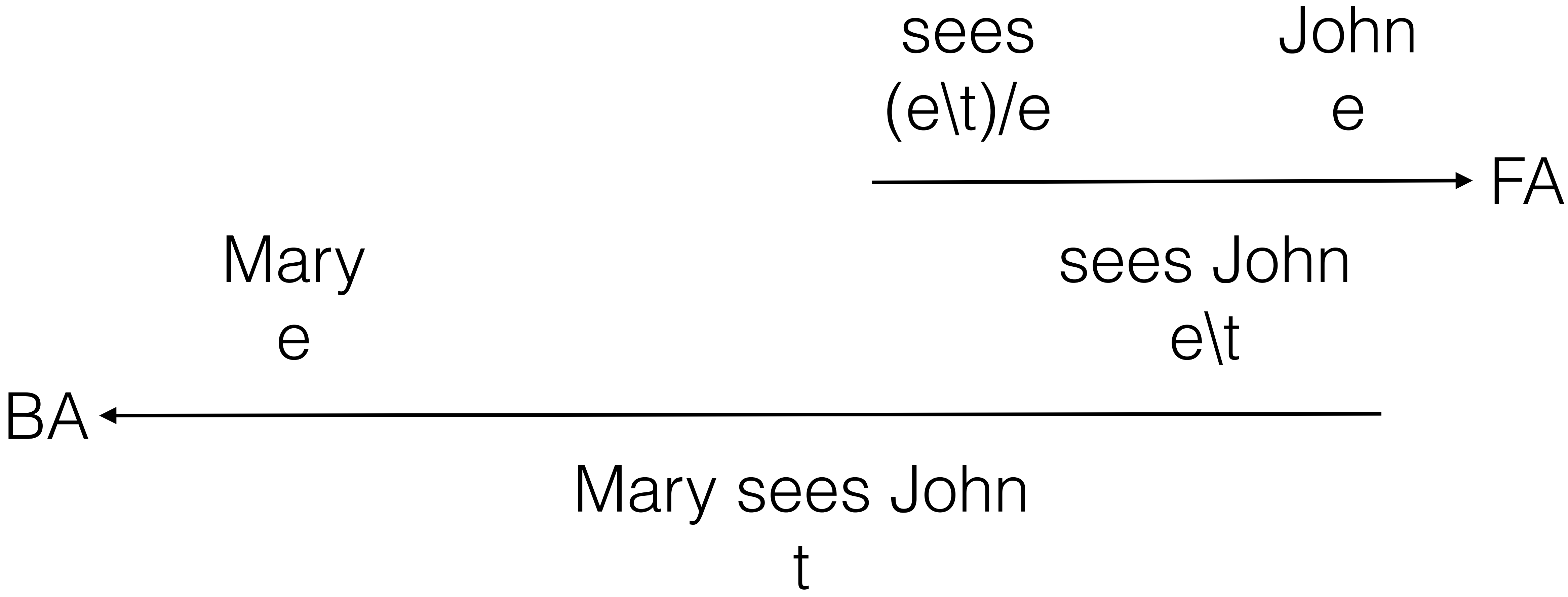
Syntactic Class	Example	Category
Proper Name	John	e
Intransitive Verb	walks sees John	e\t
Sentence	John walks	t
Common Noun	man	p
Adjective	young	p/p
Transitive Verb	sees	(e\t)/e
Adverb	well	(e\t)\(e\t)
Noun Phrase	every man	t/(e\t)
Determiner	every	(t/(e\t))/p

Result
on Top

More Categories

Syntactic Class	Example	Category
Intensifier	very	?
Preposition	in	?
Conjunction	and	?
Ditransitive Verb	teach	?
Modal Verb	must	?
Complementizer	that	?

Derivation



sees
(e\t)/e

John
e

FA

Mary
e

sees John
e\t

BA

Mary sees John
t

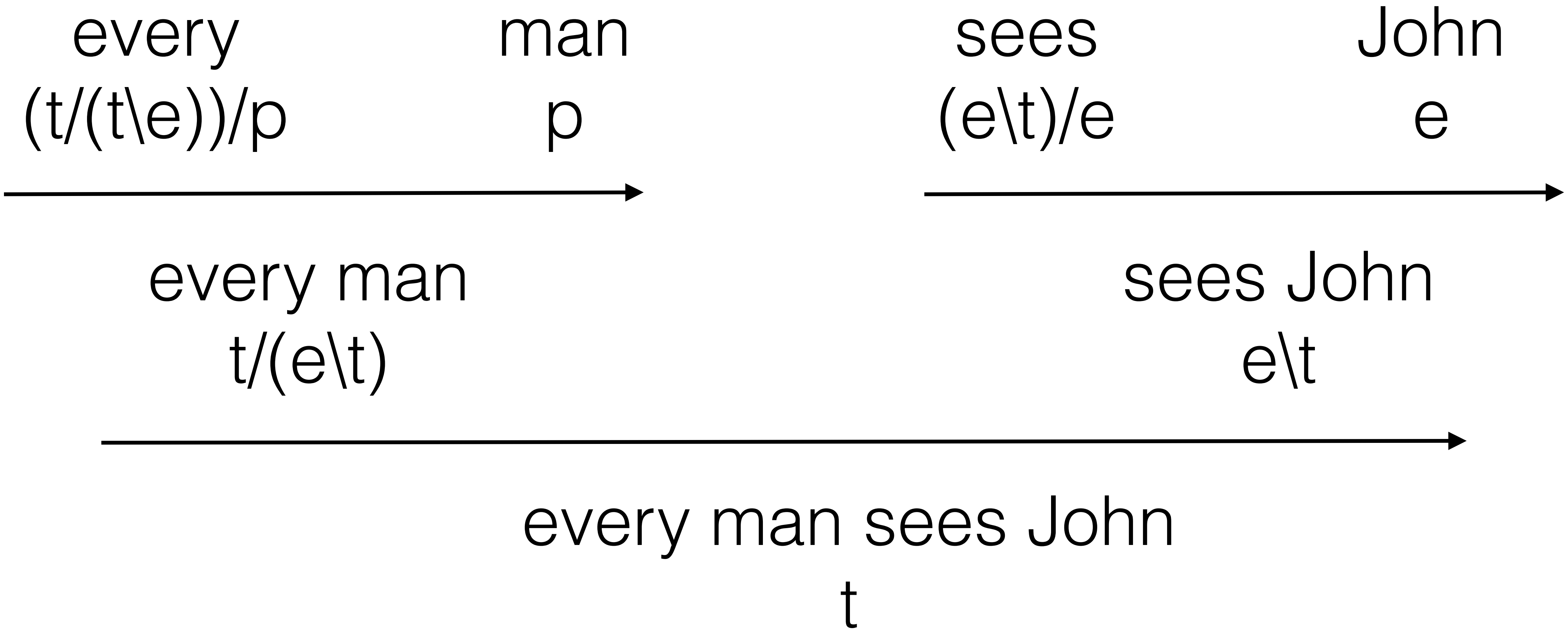
Derivation as Deduction

Premises

$$\text{MP}\;\dfrac{\begin{array}{c}\text{Mary}\\ e\end{array} \qquad\qquad \dfrac{\begin{array}{c}\text{sees}\\ e\to(e\to t)\end{array} \qquad \begin{array}{c}\text{John}\\ e\end{array}}{\begin{array}{c}\text{sees John}\\ e\to t\end{array}}\;\text{MP}}{\begin{array}{c}\text{Mary sees John}\\ t\end{array}}$$

Conclusion

Derivation



every
(t/(t\e))/p

man
p

sees
(e\t)/e

John
e

every man
t/(e\t)

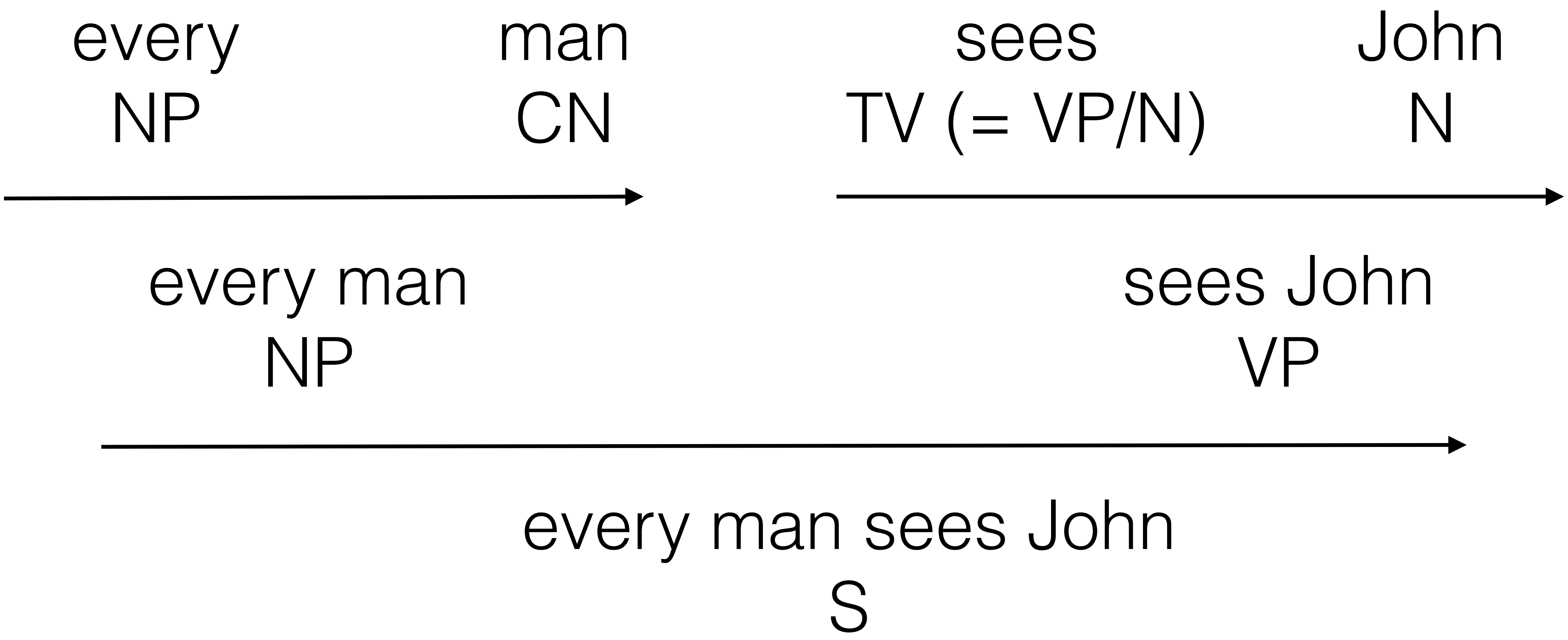
sees John
e\t

every man sees John
t

Alternative Notations

Syntactic Class	Example	Category
Proper Name	John	N
Intransitive Verb	walks	VP (= N\S)
Sentence	John walks	S
Common Noun	man	CN
Adjective	young	A (= CN/CN)
Transitive Verb	sees	TV (= VP/N)
Adverb	well	ADV (= VP\VP)
Noun Phrase	every man	NP (= S/VP)
Determiner	every	D (= NP/CN)

Derivation



every
NP

man
CN

sees
TV (= VP/N)

John
N

every man
NP

sees John
VP

every man sees John
S

John sees every man
?

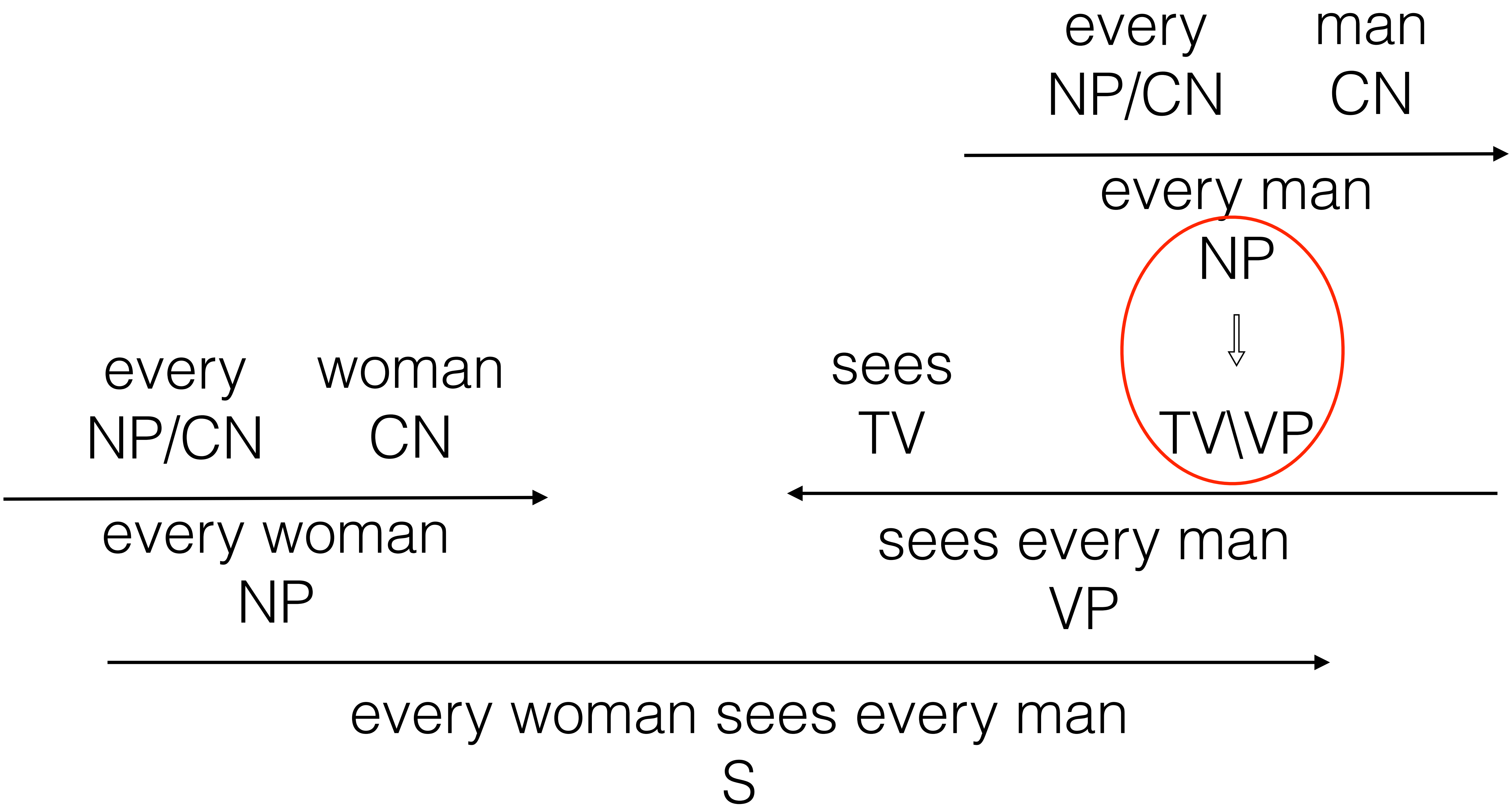
Division (“Geach Rule”)

Sánches evokes the rule of division:

Whenever an expression has the type β/α, it also has the type (α/γ)\(γ\β) for any type γ. Applied to the type of noun phrases (= **t/(e\t)**), the effect is to assign to noun phrases also the type **((e\t)/e)\(e\t)** where e plays the role of γ.

In effect, this says that in addition to being of type VP (= e\t), noun phrases also have the type TV\VP (= **((e\t)/e)\(e\t))**

Derivation with the "Geach Rule"



every NP/CN man CN

every man NP ⇩ TV\VP

every NP/CN woman CN

every woman NP

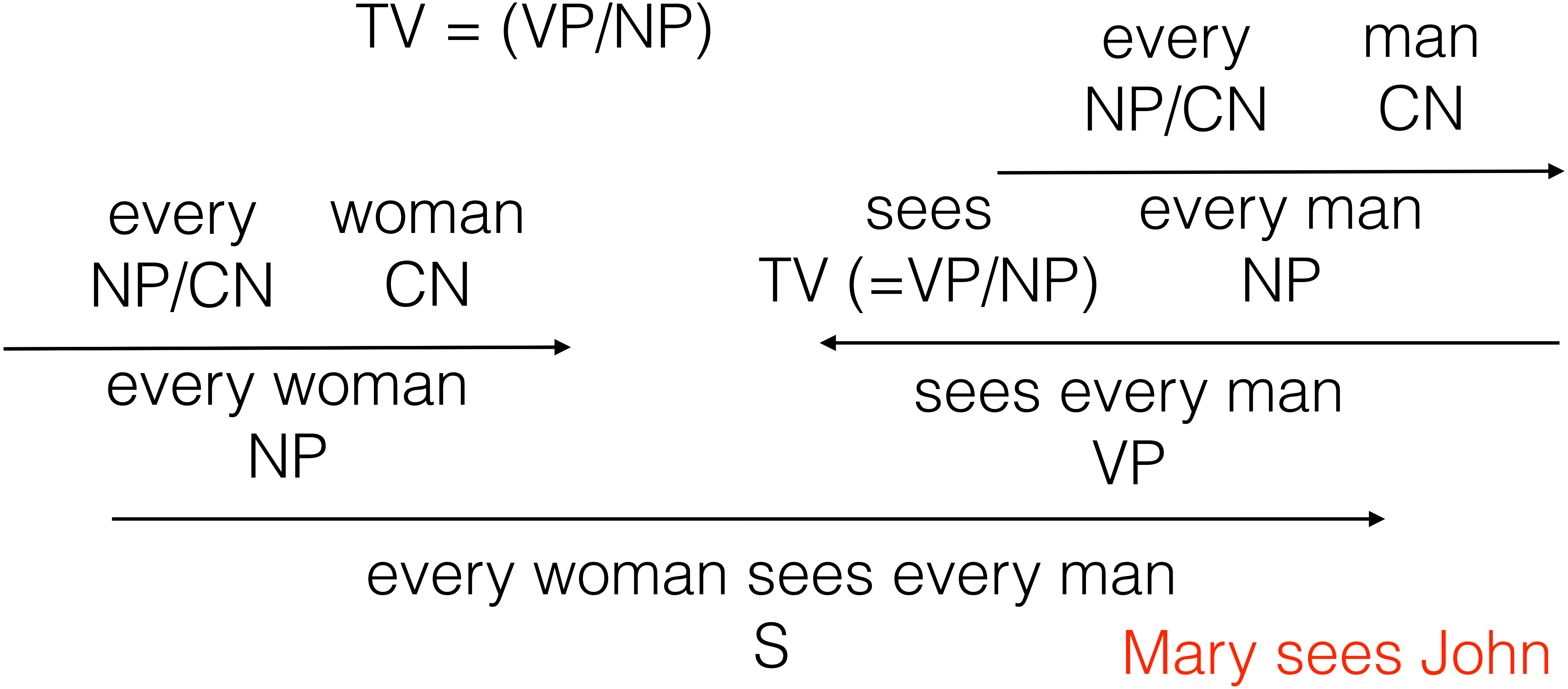
sees TV

sees every man VP

every woman sees every man S

Another solution

Make transitive verbs look for the object
NP rather than the other way round,
replace (e\t)/e by ((e\t)/(t/(e\t))
TV = (VP/NP)



every NP/CN   man CN → every man NP

every NP/CN   woman CN → every woman NP

sees TV (=VP/NP)   every man NP ← sees every man VP

every woman NP   sees every man VP → every woman sees every man S

Mary sees John
?

Type Raising ("Montague Rule")

Whenever an expression has the type α, it also has the type of a functor that takes as its argument a functor of type α\β or β/α and yields the result that this functor would return when applied to α.

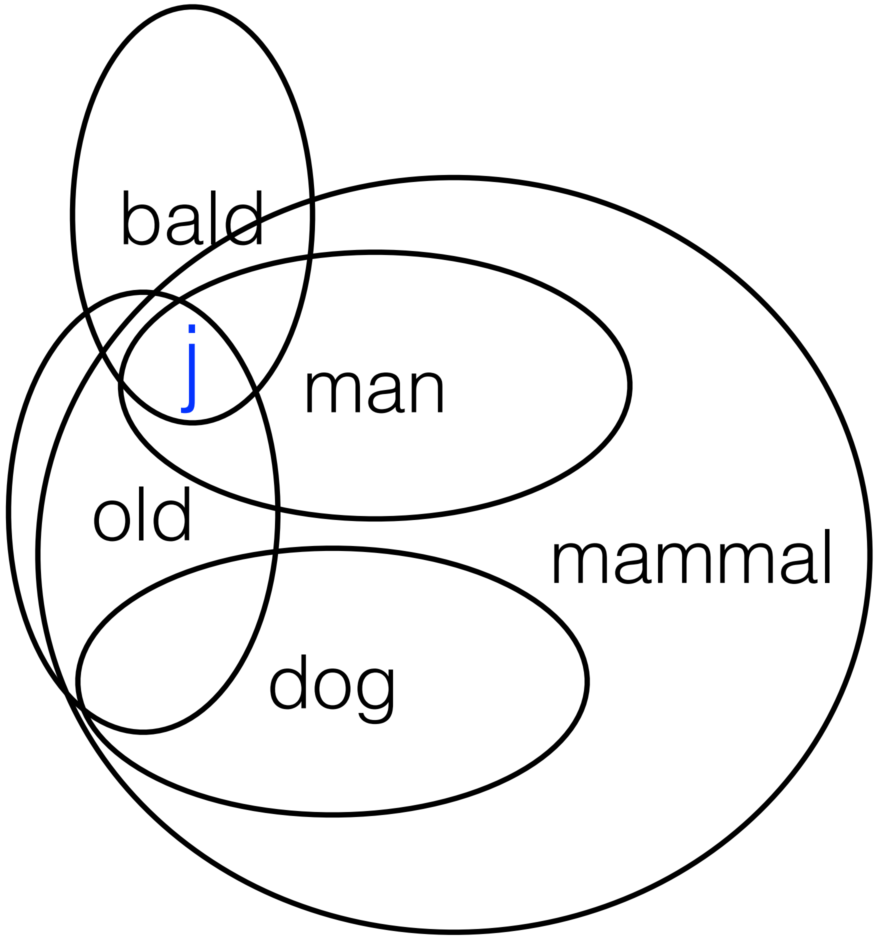
$$\alpha \Rightarrow \beta/(\alpha\backslash\beta)$$

$$\alpha \Rightarrow (\beta/\alpha)\backslash\beta$$

In particular, type raising optionally 'lifts' proper names to NPs.

$$\mathbf{e \Rightarrow t/(e\backslash t)}$$

Semantics of N⇒NP



$[\![\text{John}_{\text{NP}}]\!]$ ε $D_{(e\rightarrow t)\rightarrow t}$

the set of John_{N}'s properties

The type raising rule that lifts proper names to noun phrases has a semantic counterpart. The interpretation of *John*$_{N}$ is a particular individual, *j*, the interpretation of *John*$_{NP}$ is the set of his properties (= all and only the sets that *j* is a member of).

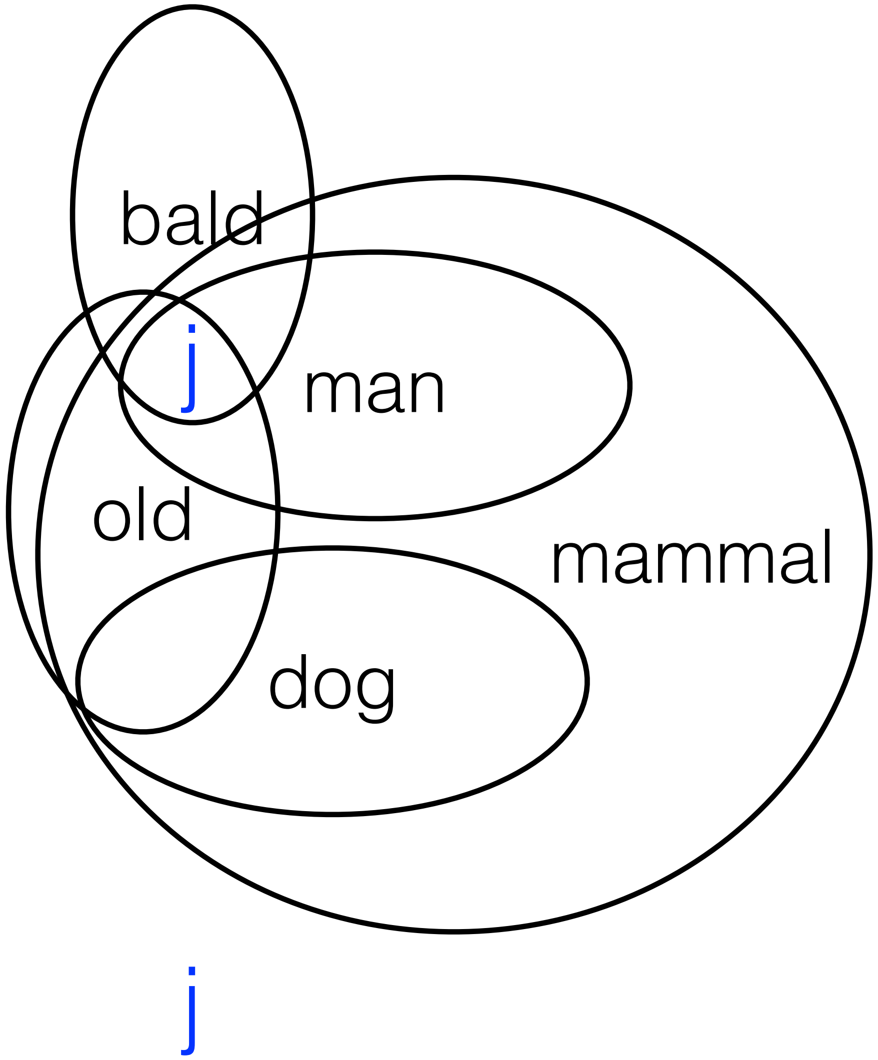
This idea is often attributed to Montague but...

j

$[\![\text{John}_{\text{N}}]\!]$ ε D_e

the individual

Semantics of N⇒NP



$[\![\text{John}_{\text{N}}]\!] \; \varepsilon \; D_{e}$

the individual

$[\![\text{John}_{\text{NP}}]\!] \; \varepsilon \; D_{(e\to t)\to t}$

the set of John$_{\text{N}}$'s properties

$[\![\text{man}]\!] \; \varepsilon \; D_{e\to t}$

the set of men

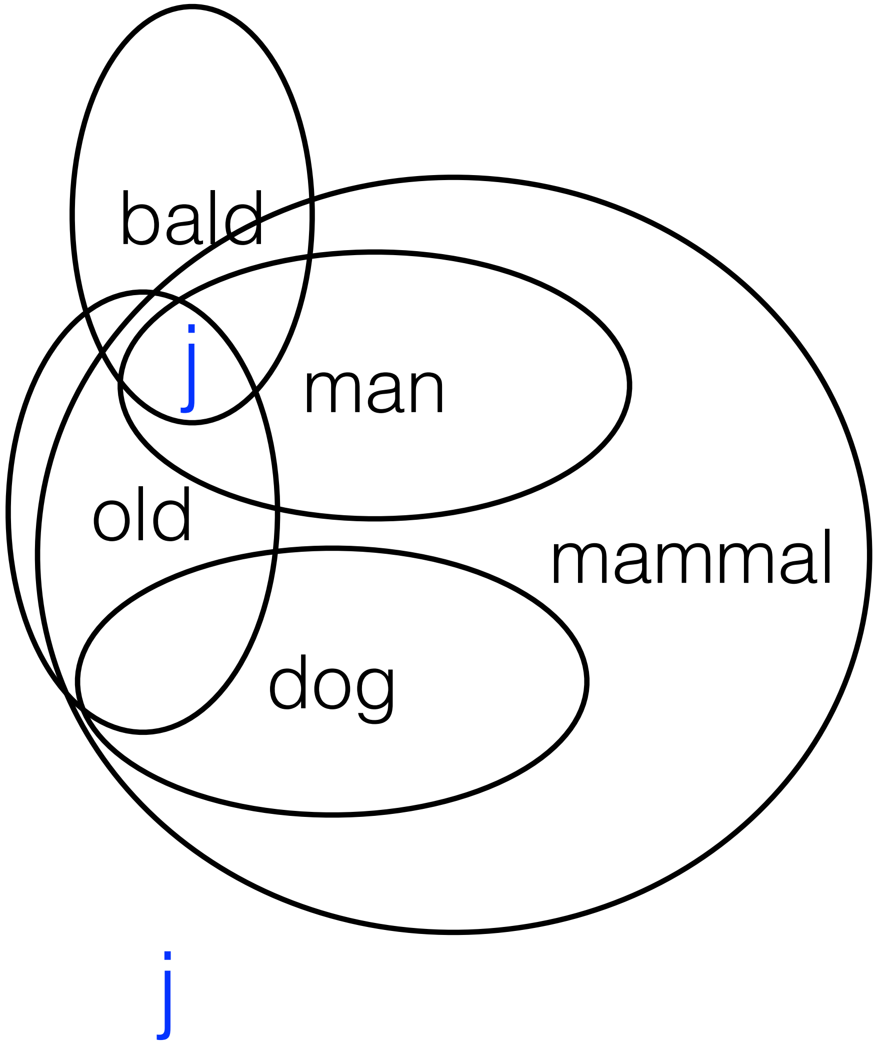
$[\![\text{man}]\!]([\![\text{John}_{\text{N}}]\!]) = 1$

$[\![\text{dog}]\!]([\![\text{John}_{\text{N}}]\!]) = 0$

$[\![\text{John}_{\text{NP}}]\!]([\![\text{man}]\!]) = 1$

$[\![\text{John}_{\text{NP}}]\!]([\![\text{dog}]\!]) = 0$

A Quote from Antiquity



“Bottom level e should be the type of proper names. But Basil of Caesarea (4th century AD, in Sorabji *Sourcebook* III p. 227):

The names [of particular men] are not actually signifiers of substances, but of the distinctive properties which characterise the individual.

Basil may be relying on earlier Stoic sources, but this view was accepted by Porphyry and his successors. It had the effect of preventing the use of type e until Frege.”

Wildfrid Hodges *Why modern logic took so long to arrive: Three lectures*, p. 74 http://wilfridhodges.co.uk/history11.pdf

Did Frege invent “Type Lowering?”

St. Basil

“I greatly respect Wilfrid Hodges as a mathematical logician and a historian [---] And yet, I sometimes feel that his strong claims about history are not uncontroversial, and based on personal views.”

“I would not attach much significance to this claim about St. Basil: just let it go.”

Van Benthem, p.c. (2011-07-15)

http://en.wikipedia.org/wiki/Basil_of_Caesarea

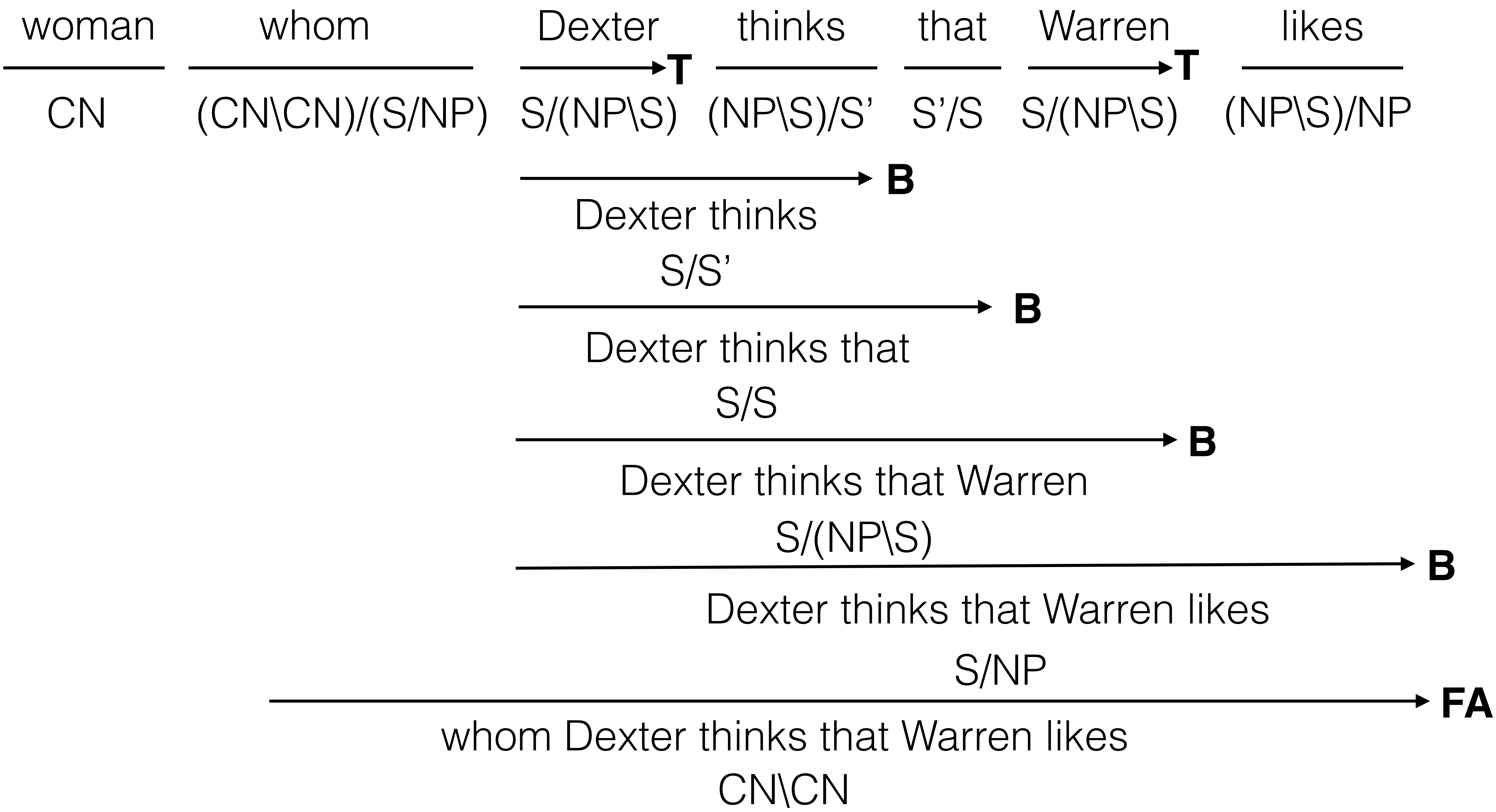
Composition

In addition to function application, Lambek calculus and categorial grammar employ function composition.

Forward composition:
If X is of type α/β and Y is of type β/γ
XY is of type α/γ.

$$\dfrac{\begin{matrix} X & & Y \\ \alpha/\beta & & \beta/\gamma \end{matrix}}{\begin{matrix} X\,Y \\ \alpha/\gamma \end{matrix}} \rightarrow \mathbf{B}$$

Steedman example



woman	whom	Dexter	thinks	that	Warren	likes
		T			T	
CN	(CN\CN)/(S/NP)	S/(NP\S)	(NP\S)/S’	S’/S	S/(NP\S)	(NP\S)/NP

B

Dexter thinks

S/S’

B

Dexter thinks that

S/S

B

Dexter thinks that Warren

S/(NP\S)

B

Dexter thinks that Warren likes

S/NP

FA

whom Dexter thinks that Warren likes

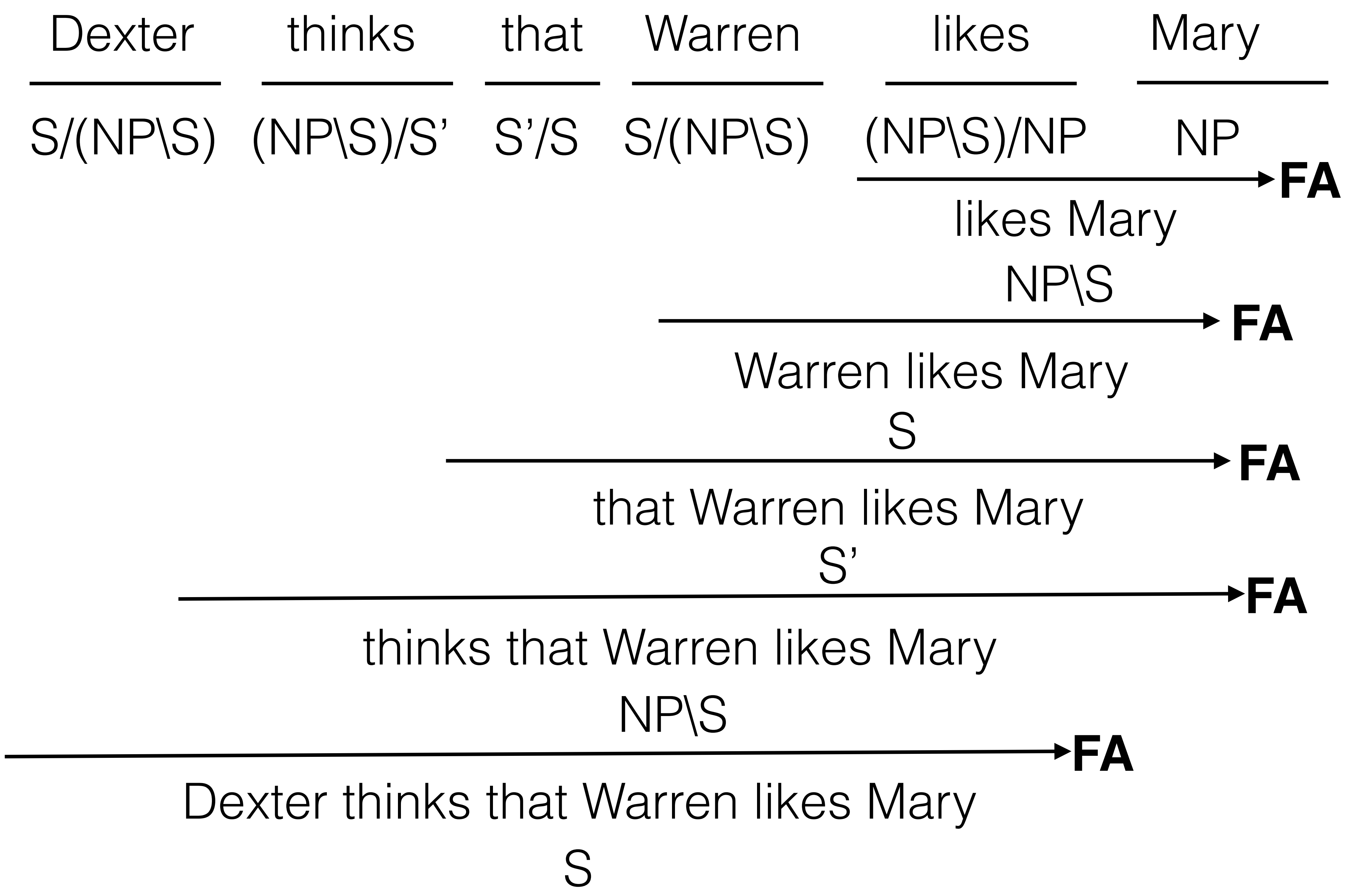
CN\CN

Successive Type Raisings

Dexter		Dexter		Dexter		Dexter
e	⇒	t/(e\t)	=	NP	⇒	S/(NP\S)
						t/((t/(e\t))\t)

Note: Here as well as on the previous slide we use the convention that the result of a backward oriented function category is above the backslash ,\S, rather than to the left of it, S\, as Steedman has it.

Alternative Derivations



Dexter	thinks	that	Warren	likes	Mary
S/(NP\S)	(NP\S)/S’	S’/S	S/(NP\S)	(NP\S)/NP	NP

FA

likes Mary

NP\S

FA

Warren likes Mary

S

FA

that Warren likes Mary

S’

FA

thinks that Warren likes Mary

NP\S

FA

Dexter thinks that Warren likes Mary

S

This is a feature, not a bug, in CCG!

What we are aiming towards

Parts of Sentences	Example	Syntactic Category	Semantic Type
Proper Name	John	e	
Intransitive Verb	walks	VP	e→t
Common Noun	man	CN	e→t
Transitive Verbs	sees	TV	e→(e→t) e→e→t
Adverbs	well	VP\VP	(e→t)→(e→t)
Noun Phrases	every man	NP	(e→t)→t
Determiner	every	NP/CN	(e→t)→((e→t)→t)