

# The Logic of Sense and Reference

Reinhard Muskens

Tilburg Center for Logic and Philosophy of Science (TiLPS)

Days 4 and 5

# Overview

- 1 Introduction
- 2 Types and Terms
- 3 Models
- 4 Proofs
- 5 Model Theory
- 6 Applications
- 7 Worlds
- 8 Conclusion

# Intensional Models for the Theory of Types

- When we looked at Thomason's work, we saw that it is possible to work with propositions as primitive entities **within the framework of classical type theory**.
- Today and tomorrow we'll develop an alternative. We will **generalize** standard type theory so that a sense-reference distinction becomes available.
- We shall give up the **Axiom of Extensionality**.
- And we shall generalize the 'general models' of Henkin (1950) in order to make our models match the weaker logic.
- Many desired notions will become available in the generalized logic and logical omniscience problems will disappear.
- See Muskens (2007) for references and technical details.

# Intensionality and Extensionality

- Whitehead and Russell (1913) call a propositional function of propositional functions  $F$  **extensional** if, in modern notation:  

$$\forall XY (\forall \vec{x} (X\vec{x} \leftrightarrow Y\vec{x}) \rightarrow (FX \rightarrow FY)).$$
- (This is slightly generalized to the case where  $X$  and  $Y$  take an arbitrary finite number of arguments, not just one.)
- In a relational version of type theory (Orey 1959; Schütte 1960) the formula above defines the extensionality of **relations of relations**.
- A function of functions that is not extensional Whitehead and Russell call **intensional**.
- The **Axiom of Extensionality** is the statement that all functions of functions (relations of relations) are extensional:  

$$\forall XY (\forall \vec{x} (X\vec{x} \leftrightarrow Y\vec{x}) \rightarrow \forall Z (ZX \rightarrow ZY)).$$

# Leibniz Identity

- Note that, given standard assumptions,  $\forall Z(ZX \rightarrow ZY)$  is equivalent with  $\forall Z(ZX \leftrightarrow ZY)$  (consider  $\lambda X.\neg ZX$ ).
- $\forall Z(ZX \leftrightarrow ZY)$ —‘ $X$  and  $Y$  are indistinguishable’.
- Russell **defined**  $X = Y$  as  $\forall Z(ZX \rightarrow ZY)$ .
- Leibniz identity: identity as indistinguishability.
- Given Russell’s definition (which we will adopt) Extensionality can also be written as  $\forall XY(\forall \vec{x}(X\vec{x} \leftrightarrow Y\vec{x}) \rightarrow X = Y)$ .

# Intensionality and Possible Worlds

- Nowadays the term **intensionality** has acquired a meaning that is different from Whitehead and Russell's. **Intensional** is often used as roughly equivalent to **modal**.
- Attempts at getting true intensionality with the help of possible worlds usually **fail**.
- One reason for this is that if propositions are treated as **sets** of possible worlds, two propositions that are true in the same worlds must be equated.
- We will not use the modal conception of intensionality but will stick to Whitehead and Russell's notion.
- Later on, it will be shown that possible worlds become available as a **result** of 'going intensional'.

# The Dog is Out

- The following two sentences are true in exactly the same circumstances:
- The dog is out if the cat is in  
The cat is out if the dog is in
- So 1) if the meaning of a sentence is identified with the set of circumstances in which it is true, and 2) compositionality holds, and 3) our underlying logic obeys Extensionality, we are committed to the equivalence of:
- Mary believes that **the dog is out if the cat is in**  
Mary believes that **the cat is out if the dog is in**
- But that's wrong: **logical omniscience**.

# Aim

- The theory of types has proved enormously useful in areas such as natural language semantics and automated theorem proving.
- But its models (Henkin 1950) validate Extensionality (both the ‘standard’ models and the ‘general’ ones). For many applications this is **undesirable** (even if there may be workarounds).
- My aim here is to **provide type theory with a wider class of models**, so that intensional predicates of predicates become possible.
- The models should be relatively **simple**.
- Imposing Extensionality as a non-logical axiom should essentially bring us back to the old (general) models.

# On the Shoulders of Giants

- Takahashi (1967) and Prawitz (1968) use a generalization of Henkin's general models in cut-elimination proofs for the Theory of Types.
- These models do not validate Extensionality but their definition depends on Schütte's (1960) 'semi-valuations', essentially sets of sentences.
- Stand-alone models in which Extensionality need not hold are given in Fitting (2002) and Benzmüller, Brown, and Kohlhase (2004).
- But the definitions of these model classes are somewhat complex and involve nonstandard abstract notions of abstraction (Fitting) or application (Benzmüller et al.).
- My work is inspired by Fitting and I'm using the Takahashi / Prawitz construction.

# Sense and Reference

- The models we are about to define are best understood as formalizing Frege's sense/reference distinction.
- In Frege's work a linguistic item **expresses** a **sense** and a sense **determines** a **reference**.
- In our approach an **intension function** sends terms to objects in a relevant domain of interpretation. **Extension functions** then provide these objects with an extension.
- Extension functions need crucially not be injective. **Different** objects can have the **same** extension.

# Types

For the sake of convenience we work with the **relational** types of Orey (1959) and Schütte (1960). Let some finite set  $\mathcal{B}$  of **basic types** be given.

## Definition

The set  $\mathcal{T}$  of **types** is the smallest set of strings such that

- ①  $\mathcal{B} \subseteq \mathcal{T}$
- ② If  $\alpha_1, \dots, \alpha_n \in \mathcal{T}$  ( $n \geq 0$ ) then  $\langle \alpha_1 \dots \alpha_n \rangle \in \mathcal{T}$

All types that are not basic are called **complex**.

# Relational Types and Functional Types

- There are several ways in which type hierarchies can be built up. Here we consider **relational** hierarchies.
- $\langle \alpha_1 \dots \alpha_n \rangle$  is the type of  $n$ -place relations that take arguments of type  $\alpha_k$  in their  $k$ -th argument place.
- A **functional** type  $\alpha \rightarrow \beta$  (or just  $\alpha\beta$ ) stands for functions taking arguments of type  $\alpha$  and yielding values of type  $\beta$ .
- Moses Schönfinkel noted that there is a **correspondence** between relational and functional types:  

$$\langle \beta \alpha_1 \dots \alpha_n \rangle \approx \beta \rightarrow \langle \alpha_1 \dots \alpha_n \rangle.$$
- $\langle \rangle \approx t$ , the type of propositions/truth values;  
 for a type such as  $\langle ee \rangle$ :  

$$\langle ee \rangle \approx e \rightarrow \langle e \rangle \approx e \rightarrow (e \rightarrow \langle \rangle) \approx e \rightarrow (e \rightarrow t)$$

# Terms

**Terms** are built up from typed constants and variables with the help of application and abstraction. There are two **logical** constants,  $\perp$  and  $\subset$ . The official definition runs as follows.

## Definition

Let  $\mathcal{L}$  be a language. Define sets  $T_\alpha^\mathcal{L}$  of **terms** of  $\mathcal{L}$  of type  $\alpha$ , for each  $\alpha \in \mathcal{T}$ , as follows.

- ①  $\mathcal{L}_\alpha \subseteq T_\alpha^\mathcal{L}$  and  $\mathcal{V}_\alpha \subseteq T_\alpha^\mathcal{L}$  for each  $\alpha \in \mathcal{T}$
- ②  $\perp \in T_{\langle \rangle}^\mathcal{L}$
- ③ If  $A \in T_{\langle \alpha_1 \alpha_2 \dots \alpha_n \rangle}^\mathcal{L}$  and  $B \in T_{\alpha_1}^\mathcal{L}$ , then  $(AB) \in T_{\langle \alpha_2 \dots \alpha_n \rangle}^\mathcal{L}$
- ④ If  $A \in T_{\langle \alpha_2 \dots \alpha_n \rangle}^\mathcal{L}$  and  $x \in \mathcal{V}_{\alpha_1}$ , then  $(\lambda x.A) \in T_{\langle \alpha_1 \alpha_2 \dots \alpha_n \rangle}^\mathcal{L}$
- ⑤ If  $A \in T_\alpha^\mathcal{L}$  and  $B \in T_\alpha^\mathcal{L}$  then  $(A \subset B) \in T_{\langle \rangle}^\mathcal{L}$ , if  $\alpha$  is complex

## Abbreviations

Having few logical constants is nice if you want to prove something using term induction, but not so nice if you actually want to use the logic. But the usual logical operators are definable with the help of  $\perp$  and  $\subset$ :

### Definition

Write

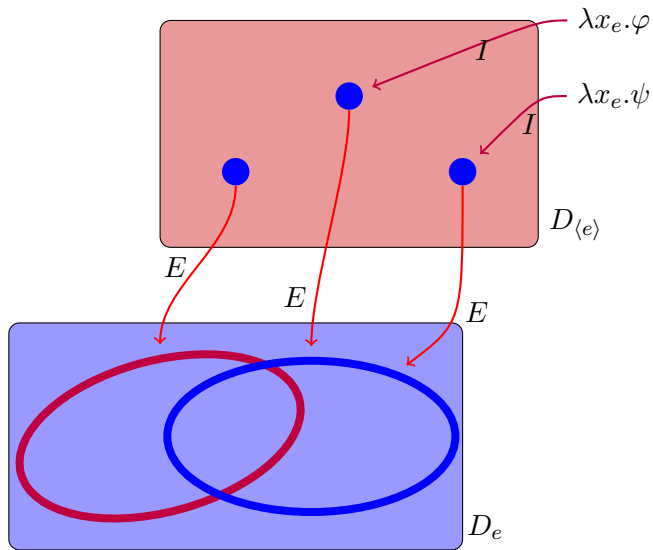
$$\begin{aligned} \varphi \rightarrow \psi & \text{ for } \varphi \subset \psi, \\ \top & \text{ for } \perp \rightarrow \perp, \\ \forall x \varphi & \text{ for } (\lambda x. \top) \subset (\lambda x. \varphi), \text{ and} \\ A_\alpha = B_\alpha & \text{ for } \forall x_{\langle \alpha \rangle} (xA \rightarrow xB). \end{aligned}$$

Other operators are obtained as usual.

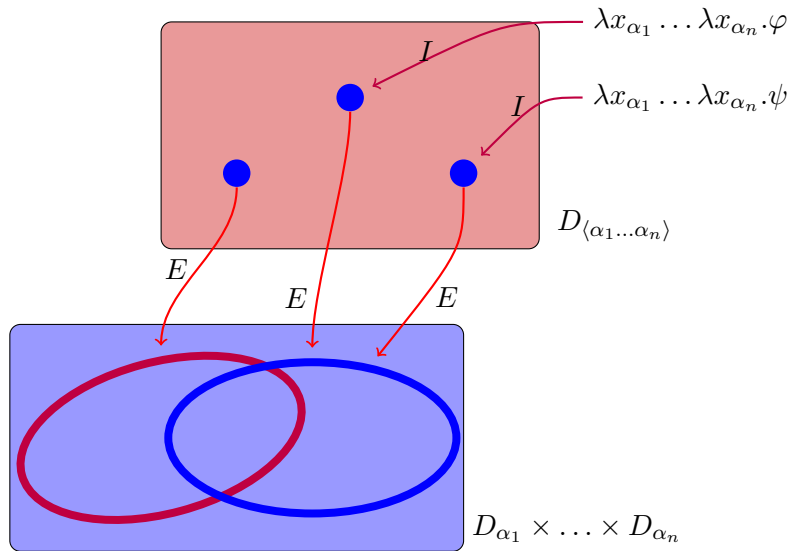
## Models: the Basic Idea

- Before we turn to the definition of our models (‘intensional models’), I’ll explain the basic idea with the help of a series of **pictures**.
- It is essential that the sense (or intension) of an expression is **not** identified with its extension.
- In the following pictures, a function  $I$  sends expressions to their senses.
- And a function  $E$  sends senses to their denotations/extensions.
- **Different** senses could be associated with **the same** denotation.
- It can even be the case that expressions that get associated with the same extensions in **all** models, are associated with different senses in some.

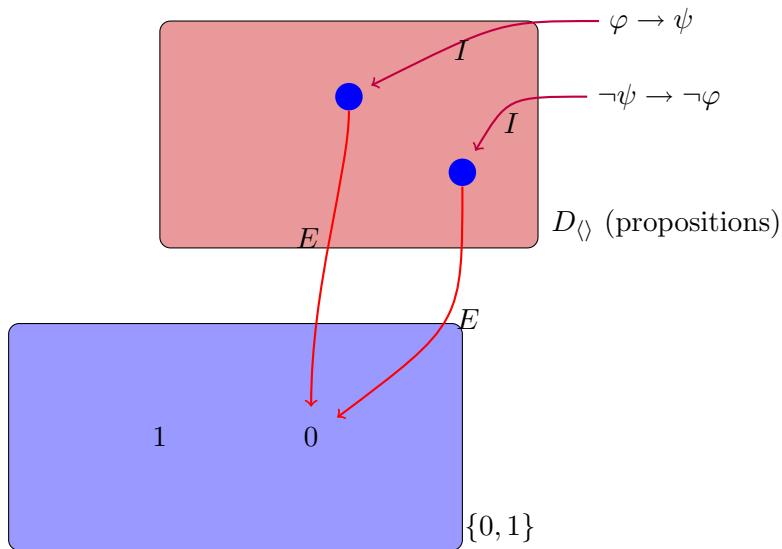
## A Picture



## More Generally



# Propositions



# Domains and Assignments

- A **collection of domains** will be a set  $\{D_\alpha \mid \alpha \in \mathcal{T}\}$ , each of whose elements is non-empty.
- The idea is that the objects in relational domains are **senses** (i.e. relations-in-intension). **Extensions** will typically be set theoretic constructs from the elements of our domains.
- The extension of an element of  $D_{\langle\alpha_1 \dots \alpha_n\rangle}$  is a subset of  $D_{\alpha_1} \times \dots \times D_{\alpha_n}$ .
- **Assignments** are defined as usual.  $\mathcal{A}_D$  is the set of all assignments for  $D$ .

# The Rest of the Construction

- We now need to define our **intension functions**  $I$  and our **extension functions**  $E$  and define the notion of an **intensional model**.
- The constraints on the first will be very light and those on the second will be virtually non-existent.
- But the structures that will be obtained will only be called intensional models if the usual Tarski definition of semantic values goes through.
- This follows the construction of (Henkin 1950).

# Intension Functions

## Definition

An **intension function** for a collection of domains  $D = \{D_\alpha \mid \alpha \in \mathcal{T}\}$  and a language  $\mathcal{L}$  is a function  $I: \mathcal{A}_D \times T^{\mathcal{L}} \rightarrow \cup D$  such that

- ①  $I(a, A) \in D_\alpha$ , if  $A$  is of type  $\alpha$
- ②  $I(a, x) = a(x)$ , if  $x$  is a variable
- ③  $I(a, A) = I(a', A)$ , if  $a$  and  $a'$  agree on all variables free in  $A$
- ④  $I(a, A\{x := B\}) = I(a[I(a, B)/x], A)$ , if  $B$  is free for  $x$  in  $A$

No more restrictions on intension functions!

# Extension Functions

## Definition

Let  $D = \{D_\alpha \mid \alpha \in \mathcal{T}\}$  be a collection of domains and let  $\alpha = \langle \alpha_1 \dots \alpha_n \rangle$  be a complex type. An **extension function** of type  $\alpha$  for  $D$  is a function

$$E_\alpha: D_\alpha \rightarrow \mathcal{P}(D_{\alpha_1} \times \dots \times D_{\alpha_n})$$

A **collection of extension functions** for  $D$  is a set  $\{E_\alpha \mid \alpha \in \mathcal{T} \setminus \mathcal{B}, E_\alpha \text{ is an extension function of type } \alpha \text{ for } D\}$ .

Note that  $E_{\langle \rangle}: D_{\langle \rangle} \rightarrow \mathcal{P}(\{\emptyset\}) = \{0, 1\}$

# Generalized Frames

## Definition

A **generalized frame** for the language  $\mathcal{L}$  is a triple  $\langle D, I, E \rangle$  such that  $D$  is a collection of domains,  $I$  is an intension function for  $D$  and  $\mathcal{L}$ , and  $E$  is a collection of extension functions for  $D$ .

This is all still very general, but the next definition will give some more bite to the construction.

# Intensional Models

- Intensional models will be generalized frames for which  $E_\alpha(I(a, A))$  is always the “correct” value.
- In the interest of readability, we will write  $V_\alpha(a, A)$ , or even  $V(a, A)$ , for  $E_\alpha(I(a, A))$ .

## Definition

A generalized frame  $\langle D, I, E \rangle$  for  $\mathcal{L}$  is an **intensional model** for  $\mathcal{L}$  if

- 1  $V(a, \perp) = 0$
- 2  $V(a, AB) = \{\langle \vec{d} \rangle \mid \langle I(a, B), \vec{d} \rangle \in V(a, A)\}$
- 3  $V(a, \lambda x_\beta. A) = \{\langle d, \vec{d} \rangle \mid d \in D_\beta \text{ and } \langle \vec{d} \rangle \in V(a[d/x], A)\}$
- 4  $V(a, A \subset B) = 1 \iff V(a, A) \subseteq V(a, B)$

# Relations as Functions

- The treatment of abstraction and application corresponds closely to the usual treatment in functional hierarchies (but is ‘uncurried’).
- $\langle \beta \alpha_1 \dots \alpha_n \rangle \approx \beta \rightarrow \langle \alpha_1 \dots \alpha_n \rangle$
- Any  $n + 1$  place relation  $R$  corresponds to a unary function  $F$  such that  $F(d) = \{ \langle \vec{d} \rangle \mid \langle d, \vec{d} \rangle \in R \}$  (the “first slice function” of  $R$ —we have discussed it on Day 2).
- $V(a, AB) = \{ \langle \vec{d} \rangle \mid \langle I(a, B), \vec{d} \rangle \in V(a, A) \} = F(I(a, B))$ , where  $F$  is the function corresponding to  $V(a, A)$ .
- $V(a, \lambda x_\beta. A) = \{ \langle d, \vec{d} \rangle \mid d \in D_\beta \text{ and } \langle \vec{d} \rangle \in V(a[d/x], A) \}$  corresponds to the function  $F$  such that  $F(d) = V(a[d/x], A)$  for each  $d \in D_\beta$ .

# Intensional Models: Some Further Remarks

- Note that the **extension** of a term  $AB$  depends on the **extension** of  $A$  and the **intension** of  $B$ .
- This means that  $AB$  and  $AB'$  can have different extensions even if  $B$  and  $B'$  have the same extension.

# Do (Interesting) Intensional Models Exist?

- We can define whatever we like, but is there anything corresponding to our definition?
- That there are intensional models is clear:
  - Full (“standard”) models (with  $I$  the usual valuation function and the  $E_\alpha$  identity functions) are intensional models.
  - Henkin’s generalized models are intensional models.
- But are there interesting cases, in which Extensionality fails?
- A **model existence** theorem tells us that there are.

# Proofs: Summary

- There is a **complete proof system** for the logic in which the usual logical operators  $\neg$ ,  $\wedge$ ,  $\vee$ ,  $\rightarrow$ ,  $\forall$ ,  $\exists$  and  $=$  behave **classically**.
- But in which Extensionality is not provable.
- The usual rules for  $\lambda$  are not present, but can be added consistently.
- Since our  $\lambda$ -expressions are (essentially) the standard ones a rough and ready characterisation of the logic then becomes: **ordinary classical type theory minus Extensionality**.

## Gentzen Calculus

$$\frac{\Pi \Rightarrow \Sigma}{\Pi' \Rightarrow \Sigma'} [W], \quad \text{if } \Pi \subseteq \Pi', \Sigma \subseteq \Sigma'$$

$$\frac{}{\Pi, \varphi \Rightarrow \Sigma, \varphi} [R]$$

$$\frac{}{\Pi, \perp \Rightarrow \Sigma} [\perp L]$$

$$\frac{\Pi, A\{x := B\}\vec{C} \Rightarrow \Sigma}{\Pi, (\lambda x.A)B\vec{C} \Rightarrow \Sigma} [\lambda L]$$

if  $B$  is free for  $x$  in  $A$

$$\frac{\Pi \Rightarrow \Sigma, A\{x := B\}\vec{C}}{\Pi \Rightarrow \Sigma, (\lambda x.A)B\vec{C}} [\lambda R]$$

if  $B$  is free for  $x$  in  $A$

$$\frac{\Pi, B\vec{C} \Rightarrow \Sigma \quad \Pi \Rightarrow \Sigma, A\vec{C}}{\Pi, A \subset B \Rightarrow \Sigma} [\subset L]$$

$$\frac{\Pi, A\vec{c} \Rightarrow \Sigma, B\vec{c}}{\Pi \Rightarrow \Sigma, A \subset B} [\subset R]$$

if the constants  $\vec{c}$  are fresh

# Derivable Rules: Classical Logic

$$\frac{}{\Pi \Rightarrow \Sigma, \top} [\top R]$$

$$\frac{\Pi, \psi \Rightarrow \Sigma \quad \Pi \Rightarrow \Sigma, \varphi}{\Pi, \varphi \rightarrow \psi \Rightarrow \Sigma} [\rightarrow L]$$

$$\frac{\Pi, \varphi \Rightarrow \Sigma, \psi}{\Pi \Rightarrow \Sigma, \varphi \rightarrow \psi} [\rightarrow R]$$

$$\frac{\Pi, \varphi\{x := A\} \Rightarrow \Sigma}{\Pi, \forall x \varphi \Rightarrow \Sigma} [\forall L]$$

$$\frac{\Pi \Rightarrow \Sigma, \varphi\{x := c\}}{\Pi \Rightarrow \Sigma, \forall x \varphi} [\forall R]$$

where  $c$  is fresh

$$\frac{\Pi, A \doteq B \Rightarrow \Sigma, \varphi\{x := A\}}{\Pi, A \doteq B \Rightarrow \Sigma, \varphi\{x := B\}} [=L]$$

$$\frac{}{\Pi \Rightarrow \Sigma, A = A} [=R]$$

where  $A \doteq B$  is  $A = B$  or  $B = A$

# Axiomatic Extensions: $\beta\eta$ Reduction

- This is still a **relatively weak** system.
- Write  $A \stackrel{E}{=} B$  for  $A \subset B \wedge B \subset A$ . We **do** have
  - $(\alpha')$   $\lambda x.A \stackrel{E}{=} \lambda y.A\{x := y\}$ , if  $y$  is free for  $x$  in  $A$ ;
  - $(\beta')$   $(\lambda x.A)B \stackrel{E}{=} A\{x := B\}$ , if  $B$  is free for  $x$  in  $A$ ;
  - $(\eta')$   $\lambda x.Ax \stackrel{E}{=} A$ , if  $x$  is not free in  $A$ .
- But  $\stackrel{E}{=}$  cannot be replaced with  $=$  here. Therefore, one may want to add all universal closures of instantiations of
  - $(\alpha)$   $\lambda x.A = \lambda y.A\{x := y\}$ , if  $y$  is free for  $x$  in  $A$ ;
  - $(\beta)$   $(\lambda x.A)B = A\{x := B\}$ , if  $B$  is free for  $x$  in  $A$ ;
  - $(\eta)$   $\lambda x.Ax = A$ , if  $x$  is not free in  $A$ .

# Elementary Model Theory

- A **model existence** theorem can be proved with the following consequences:
  - (Generalized Compactness) If  $\Gamma$  is i-valid then some finite  $\Gamma_0 \subseteq \Gamma$  is i-valid.
  - (Generalized Löwenheim–Skolem) If  $\Gamma$  is not i-valid then  $\Gamma$  is refutable by a countable normal intensional model.
  - (Generalized Completeness)  $\Gamma$  is i-valid iff  $\Gamma$  is provable.
  - (Cut elimination) If  $\Pi, \varphi \vdash \Sigma$  and  $\Pi \vdash \Sigma, \varphi$  then  $\Pi \vdash \Sigma$ .
- The proof uses techniques from Takahashi (1967) and Prawitz (1968).

# Applying the Logic to Natural Language Semantics

- The logic we have defined is intensional, but its syntax is essentially that of the usual extensional type theory.
- This means it can be used to translate natural language in the usual Montagovian way.
- The result will be an intensional semantics, even if we do not work with possible worlds.
- The following slides will give an example.

## Some Lexical Translations

WORD	TRANSLATION	WORD	TRANSLATION
if	$\lambda p_{\langle \rangle} \lambda q_{\langle \rangle} . p \rightarrow q$	man	$man_{\langle e \rangle}$
no	$\lambda P'_{\langle e \rangle} \lambda P_{\langle e \rangle} . \neg \exists x_e (P' x \wedge P x)$	unicorn	$unicorn_{\langle e \rangle}$
some	$\lambda P'_{\langle e \rangle} \lambda P_{\langle e \rangle} . \exists x_e (P' x \wedge P x)$	runs	$run_{\langle e \rangle}$
every	$\lambda P'_{\langle e \rangle} \lambda P_{\langle e \rangle} . \forall x_e (P' x \rightarrow P x)$	laughs	$laugh_{\langle e \rangle}$
loves	$\lambda Q_{\langle \langle e \rangle \rangle} \lambda x_e . Q(\lambda y_e . love_{\langle ee \rangle} x y)$	Bill	$bill_{\langle \langle e \rangle \rangle}$
is	$\lambda Q_{\langle \langle e \rangle \rangle} \lambda x_e . Q(\lambda y_e . x = y)$	Ann	$ann_{\langle \langle e \rangle \rangle}$
knows	$\lambda p_{\langle \rangle} \lambda x_e . know_{\langle e \rangle} x p$	Tully	$tully_{\langle \langle e \rangle \rangle}$
believes	$\lambda p_{\langle \rangle} \lambda x_e . believe_{\langle e \rangle} x p$	Cicero	$cicero_{\langle \langle e \rangle \rangle}$

## Some Assumptions

- We will assume the  $(\alpha)$ ,  $(\beta)$  and  $(\eta)$  axioms, so that we have **full lambda conversion**.
- There will also be some **meaning postulates** for **names**:
  - $\forall P(ann P \leftrightarrow Pa)$
  - $\forall P(bill P \leftrightarrow Pb)$
  - $\forall P(tully P \leftrightarrow Pt)$
  - $\forall P(cicero P \leftrightarrow Pc)$
  - etc.
- Names will be discussed a bit more shortly.

## A Fragment and its Translation

- We need a fragment of natural language to translate:
- All words in the previous table are syntactic structures and whenever  $X$  and  $Y$  are syntactic structures,  $[XY]$  is also a syntactic structure.
- $X \rightsquigarrow A$ , if  $X$  is a word and  $A$  is its translation in the previous table;
- If  $X \rightsquigarrow A$  and  $Y \rightsquigarrow B$  then  $[XY] \rightsquigarrow AB$  if  $AB$  is a well-formed term and  $[XY] \rightsquigarrow BA$  if  $BA$  is well-formed.
- “Shake’n bake semantics” (Emmon Bach). Lots of syntactic structures will be gibberish, but then, lots of these will also be **uninterpretable**.
- We are only interested in cases where we do get an interpretation.

# Consequences I

The following form–meaning pairs are obtained:

- a  $[[[\text{no man}]\text{laughs}][\text{if}[[\text{some unicorn}]\text{runs}]]] \rightsquigarrow \exists x(\text{unicorn } x \wedge \text{run } x) \rightarrow \neg \exists x(\text{man } x \wedge \text{laugh } x)$
- b  $[[[\text{no unicorn}]\text{runs}][\text{if}[[\text{some man}]\text{laughs}]]] \rightsquigarrow \exists x(\text{man } x \wedge \text{laugh } x) \rightarrow \neg \exists x(\text{unicorn } x \wedge \text{run } x)$
- c  $[[\text{every man}][\text{knows}[[[\text{no man}]\text{laughs}][\text{if}[[\text{some unicorn}]\text{runs}]]]]] \rightsquigarrow \forall y(\text{man } y \rightarrow \text{know } y (\exists x(\text{unicorn } x \wedge \text{run } x) \rightarrow \neg \exists x(\text{man } x \wedge \text{laugh } x)))$
- d  $[[\text{every man}][\text{knows}[[[\text{no unicorn}]\text{runs}][\text{if}[[\text{some man}]\text{laughs}]]]]] \rightsquigarrow \forall y(\text{man } y \rightarrow \text{know } y (\exists x(\text{man } x \wedge \text{laugh } x) \rightarrow \neg \exists x(\text{unicorn } x \wedge \text{run } x)))$

a and b co-entail; c and d **do not**.

## Consequences II: Names

- a Tully runs  $\rightsquigarrow$   $tully\ run \stackrel{E}{=} run\ t$
- b Tully is Cicero  $\rightsquigarrow$   $tully(\lambda x.cicero(\lambda y.x = y)) \stackrel{E}{=} t = c$
- c Cicero runs  $\rightsquigarrow$   $cicero\ run \stackrel{E}{=} run\ c$
- d Ann believes Tully runs  $\rightsquigarrow$   $ann(\lambda x.believe\ x(tully\ run)) \stackrel{E}{=} believe\ a(tully\ run)$
- e Ann believes Cicero runs  $\rightsquigarrow$   $ann(\lambda x.believe\ x(cicero\ run)) \stackrel{E}{=} believe\ a(cicero\ run)$

c follows from a + b; e **does not follow** from b + d.

# Names

- If **names** are treated as constants of individual type  $e$ , the usual Hesperus–Phosphorus, Cicero–Tully, Dr Lauben–Gustav Lauben problems are **not** solved.
- *believe a (run t)* and  $t = c$  together do entail *believe a (run c)*.
- The Extensionality axiom does not talk about individuals at all; so getting rid of it should not be expected to change the logic of individual constants.
- So we must either introduce a sense-reference distinction for individuals into the logic,
- Or we should treat names as dependent upon predicates somehow.
- I think the latter course is most attractive. Recall Quine's **Primacy of Predicates** in *Word and Object* (Quine 1960).

## Names as Descriptions

- Another way to treat names as being dependent upon predicates is to adopt the classical **description theory of names** (Frege, Russell, Quine). For example:
- Translate ‘Cicero’ as  $\lambda P.\exists x(Cx \wedge Px)$ , adopting a meaning postulate  $\forall xy((Cx \wedge Cy) \rightarrow x = y)$ . Treat other names similarly.
- The predicate  $C$  stands for ‘being Cicero’ or ‘Ciceronize’ in Quine’s way.
- *Cicero is Tully* now translates as **coextensionality** of  $C$  and  $T$ —‘being Tully’.
- Nice Russellian treatment of nonexistence statements: *Zeus doesn’t exist* can go to  $\neg\exists xZx$ —nothing Zeuses.

## Names as Descriptions: Kripke's View

- In the 1970s the description theory of names came under attack from the side of various philosophers who advocated the view of John Stuart Mill that the meaning of a name is its **bearer**.
- Kripke (1972) argues that the name **Aristotle** cannot mean **the student of Plato and teacher of Alexander** (Frege's example). Otherwise, **Aristotle taught Alexander** would be a necessarily true statement, while it is a contingent statement, of course.
- But that line of attack does not seem to work against Quine's treatment of names. It seems perfectly defensible to say that **Aristotle Aristotelized** indeed is a necessarily true (be it rather artificial) statement.
- Note that in many languages names can indeed take the form of descriptions: *der Hans*, *el Juan*.

## Names: Yet Another Way

- Names were treated as predicates of type  $\langle\langle e \rangle\rangle$  and were connected with objects of type  $e$  via **meaning postulates**.
- Since all predicates are intensional within the present logic, the result was an **intensional** treatment of names.
- Another possibility: Give names type  $e$  and predicates like *dances* and *adores* types  $\langle e \rangle$  and  $\langle ee \rangle$ . But consider  $e$  to be an **abbreviation** of a complex type  $\langle 0 \rangle$ .
- This gives **intensionality** in type  $e$ , but also lays the foundation for a treatment of **plurality** (Bartsch, Hoeksema, Landman).
- Singular individuals will have singleton extensions, plural individuals will have extensions with cardinality  $> 1$ .
- We will not follow up this possibility now.

## Intermediate Conclusion

- It seems that with a generalization of type logic that does not validate Extensionality it is possible to get rid of logical omniscience and similar problems.
- A provision is that names are analysed in terms of predicates in one way or another—but there are several ways to do just that.
- But what about **modality**? Is it possible to reobtain possible worlds and the usual modal operators?
- This will be the topic of our next slides.

# Worlds Regained

- Our logic is just classical higher order logic without Extensionality.
- It could be made into a higher order **modal** logic by introducing an extra basic domain  $s$  of **possible worlds** and introducing **modal operators** in the obvious way. This possibility was explored in Muskens (2006).
- But introducing worlds as primitives is in fact unnecessary. Worlds can be obtained as certain objects of type  $\langle\langle\rangle\rangle$ , i.e. **properties of propositions**.
- Similar constructions in Wittgenstein's *Tractatus*, Carnap's *Meaning and Necessity*, and recent work by Carl Pollard.
- The idea of the construction is even already present in **Leibniz' Theodicy**.

## Two Postulates

- Assume  $(\alpha)$ ,  $(\beta)$  and  $(\eta)$  and introduce a constant  $\Omega$  of type  $\langle\langle\langle\rangle\rangle\rangle$ .  
Intuitive interpretation: ‘is a world’. Postulate:
- (W1)  $\forall w(\Omega w \rightarrow \neg w \perp)$
- (W2)  $\forall w(\Omega w \rightarrow (w(A \subset B) \leftrightarrow \forall \vec{x}(w(A\vec{x}) \rightarrow w(B\vec{x}))))$
- Here are some consequences:
- $\forall w(\Omega w \rightarrow (w(\neg\varphi) \leftrightarrow \neg(w\varphi)))$
- $\forall w(\Omega w \rightarrow (w(\varphi \wedge \psi) \leftrightarrow ((w\varphi) \wedge (w\psi))))$
- $\forall w(\Omega w \rightarrow (w(\forall x\varphi) \leftrightarrow \forall x(w\varphi)))$
- $\forall w(\Omega w \rightarrow (w(\exists x\varphi) \leftrightarrow \exists x(w\varphi)))$
- ‘Maximal consistent sets with the Henkin (witness) property’.

# The Actual World

- It is natural to introduce a constant @ for the **actual world** and to postulate:
- $\Omega @$   
 $\forall p_{\diamond} (@p \leftrightarrow p)$
- In models satisfying Extensionality, @ is the **only** world.

## Two More Postulates

- Since worlds are of type  $\langle\langle\rangle\rangle$  it is possible to iterate and form propositions  $w_1\varphi$ ,  $w_2(w_1\varphi)$ ,  $w_3(w_2(w_1\varphi))$  and so on.
- Is it acceptable that these differ in truth value? Preliminary answer: No.
- **(W3)**  $\forall ww'((\Omega w \wedge \Omega w') \rightarrow (w(w'\varphi) \leftrightarrow (w'\varphi)))$
- Worldhood itself presumably should also be invariant:
- **(W4)**  $\forall w(\Omega w \rightarrow \forall w'(\Omega w' \leftrightarrow w(\Omega w')))$

# Accessibility

- If  $R$  is of type  $\langle\langle\rangle\rangle$  it can be interpreted as a predicate ‘is accessible’.
- $\lambda w \lambda w'. w(Rw')$  will then play the role of an accessibility relation.
- The usual relational properties (transitivity, reflexivity, euclideaness, ...) can then either be stipulated or, depending on the choice of  $R$ , be shown to hold.
- For example, the universal accessibility relation  $\lambda w \lambda w'. w(\Omega w')$  is easily seen to be an equivalence relation on the set of worlds in view of (W4).
- Other accessibility relations can be constructed as well.

# Modal Operators

- Write  $[R]$  for  $\lambda p \forall w ((\Omega w \wedge R w) \rightarrow w p)$ .
- $[R]\varphi$  reduces  $\forall w ((\Omega w \wedge R w) \rightarrow w \varphi)$ .
- Write  $\langle R \rangle$  for  $\lambda p. \neg [R] \neg p$
- It can be shown (using W1-W4) that:  

$$\forall w' (\Omega w' \rightarrow (w'([R]\varphi) \leftrightarrow \forall w ((\Omega w \wedge w'(R w)) \rightarrow w \varphi))$$

# Implicit Belief from Explicit Belief I

- Thus far we have used an operator  $believe_{\langle e \rangle}$  expressing a relation of **explicit belief** between a person and a proposition.
- We can use this relation to **construct** a modal operator corresponding to **implicit belief**.
- Ann will be our paradigmatic agent. Consider the following property  $B_a$  of worlds.
- $\lambda w. \forall p ((believe\ ap \leftrightarrow w(believe\ ap)) \wedge (believe\ ap \rightarrow wp))$
- A world  $w$  is accessible in this sense if 1) Ann believes exactly the same things in  $w$  as she believes in the actual world and 2) everything Ann believes in the actual world is true in  $w$ .

# Implicit Belief from Explicit Belief II

- It is somewhat of an idealization to stipulate that  $\lambda w \lambda w'. w(B_a w')$  is serial (Ann could believe inconsistent things), but if we do, we get

$$[B_a]\varphi \rightarrow \langle B_a \rangle \varphi$$

- And the definition of  $B_a$  automatically gives transitivity and euclideaness of  $\lambda w \lambda w'. w(B_a w')$ . So the following schemata can be obtained as well.

$$[B_a]\varphi \rightarrow [B_a][B_a]\varphi$$

$$\langle B_a \rangle \varphi \rightarrow [B_a]\langle B_a \rangle \varphi$$

- So, the definition of  $B_a$  plus the stipulation that  $\lambda w \lambda w'. w(B_a w')$  is serial give the usual **D45** axioms to  $[B_a]$ .

## No Intensionalization

- In the usual approach the possibility of having modal operators requires all translations, even translations of sentences that do not contain any modal operators, to be parametrized for possible worlds.
- E.g.  $\lambda i. \exists x(\textit{unicorn } xi \wedge \textit{run } xi) \rightarrow \neg \exists x(\textit{man } xi \wedge \textit{laugh } xi)$
- Here that is not necessary. It is okay to work with the simpler  $\exists x(\textit{unicorn } x \wedge \textit{run } x) \rightarrow \neg \exists x(\textit{man } x \wedge \textit{laugh } x)$
- (We also do not have a parameter for possible worlds at the metalevel, as in standard modal logic.)
- Modalities can still be added:  
 $[B_a](\exists x(\textit{unicorn } x \wedge \textit{run } x) \rightarrow \neg \exists x(\textit{man } x \wedge \textit{laugh } x))$

# Conclusion

- The Theory of Types can be provided with a class of models in which Frege's distinction between sense and reference is formalized.
- These models generalize Henkin's models and as a class do not validate Extensionality.
- Problems with logical omniscience will go away.
- Adopting Extensionality as a non-logical axiom will essentially bring us back to the original Henkin models.
- Possible worlds and possible situations can be constructed as sets of propositions.
- No “intensionalisation” necessary.

# References I

Benzmüller, C., C. E. Brown, and M. Kohlhase (2004).  
Higher Order Semantics and Extensionality.  
*Journal of Symbolic Logic* 69.

Fitting, M. (2002).  
*Types, Tableaus, and Gödels God*.  
Dordrecht: Kluwer Academic Publishers.

Henkin, L. (1950).  
Completeness in the Theory of Types.  
*Journal of Symbolic Logic* 15, 81–91.

Kripke, S. (1972).  
Naming and Necessity.  
In D. Davidson and G. Harman (Eds.), *Semantics of Natural Language*, pp.  
253–355. Dordrecht: Reidel.

## References II

Muskens, R. (2006).

Higher Order Modal Logic.

In P. Blackburn, J. van Benthem, and F. Wolter (Eds.), *Handbook of Modal Logic*,  
Studies in Logic and Practical Reasoning. Dordrecht: Elsevier.

Muskens, R. (2007).

Intensional Models for the Theory of Types.

*Journal of Symbolic Logic* 72(1), 98–118.

Orey, S. (1959).

Model Theory for the Higher Order Predicate Calculus.

*Transactions of the American Mathematical Society* 92, 72–84.

Prawitz, D. (1968).

Hauptsatz for Higher Order Logic.

*Journal of Symbolic Logic* 33(3), 452–457.

Quine, W. (1960).

*Word and Object*.

MIT Press.

## References III

Schütte, K. (1960).

Syntactical and Semantical Properties of Simple Type Theory.

*Journal of Symbolic Logic* 25(4), 305–326.

Takahashi, M. (1967).

A Proof of Cut-elimination Theorem in Simple Type Theory.

*Journal of the Mathematical Society of Japan* 19(4), 399–410.

Whitehead, A. N. and B. Russell (1913).

*Principia Mathematica*.

Cambridge University Press.