Names

Reinhard Muskens

Tilburg Center for Logic and Philosophy of Science

Toulouse, The Logic of the Lexicon, 30-2-2013

The Standard Theory

Standard approaches to proper names, based on (Kripke 1971; Kripke 1972), make the following three assumptions.

Carnap Intensionality The semantic values of expressions are (possibly partial) functions from possible worlds to extensions.

Extensionality These functions are identified with their graphs.

Rigidity Names are rigid designators, i.e. their extensions are world-independent.

In particular, the semantic values of names are taken to be constant functions from worlds to entities, possibly undefined for some worlds.

Problems for the Standard Theory I

- The problems for the Standard Theory are well-known.
- On the one hand (Carnap Intensionality +) Extensionality lead to the usual problems of logical omniscience.
- Propositions such as $p \rightarrow q$ and $\neg q \rightarrow \neg p$ will not only be equivalent, but will actually be identified.
- This is wrong, as a person may well believe $p \rightarrow q$ but fail to believe $\neg q \rightarrow \neg p$, and so the two propositions do not have the same properties.
- Rigidity makes things worse, as we shall see on the next slide.

Problems for the Standard Theory II

- Any theory that says that the semantic values of names are constant functions in extension from worlds to entities entails that codesignating names have the same semantic values.
- And hence that codesignating names can be interchanged in any context whatsoever, *salva veritate*.
- But that is not the case:

(1) a. We do not know *a priori* that Hesperus is Phosphorus
    b. We do not know *a priori* that Phosphorus is Phosphorus

- (1a) is asserted in (Kripke 1972, page 308); (1b) is obviously false.

What If We Give Up Extensionality?

Let's have a look at the standard assumptions again:

Carnap Intensionality The semantic values of expressions are functions from possible worlds to extensions.

Extensionality These functions are identified with their graphs.

Rigidity Names are rigid designators, i.e. their extensions are world-independent.

What If We Give Up Extensionality?

Let's have a look at the standard assumptions again:

Carnap Intensionality The semantic values of expressions are functions from possible worlds to extensions.

Extensionality These functions are identified with their graphs.

Rigidity Names are rigid designators, i.e. their extensions are world-independent.

- What happens if we give up Extensionality and let the semantic values of expressions be functions in intension?

What If We Give Up Extensionality?

Let's have a look at the standard assumptions again:

Carnap Intensionality The semantic values of expressions are functions from possible worlds to extensions.

Rigidity Names are rigid designators, i.e. their extensions are world-independent.

- What happens if we give up Extensionality and let the semantic values of expressions be functions in intension?
- Then we would get a theory in which names are rigid designators but the meaning of a name is not determined by its bearer.

What If We Give Up Extensionality?

Let's have a look at the standard assumptions again:

Carnap Intensionality The semantic values of expressions are functions from possible worlds to extensions.

Rigidity Names are rigid designators, i.e. their extensions are world-independent.

- What happens if we give up Extensionality and let the semantic values of expressions be functions in intension?
- Then we would get a theory in which names are rigid designators but the meaning of a name is not determined by its bearer.
- Actually, in a truly intensional theory Carnap Intensionality becomes superfluous.

What If We Give Up Extensionality?

Let's have a look at the standard assumptions again:

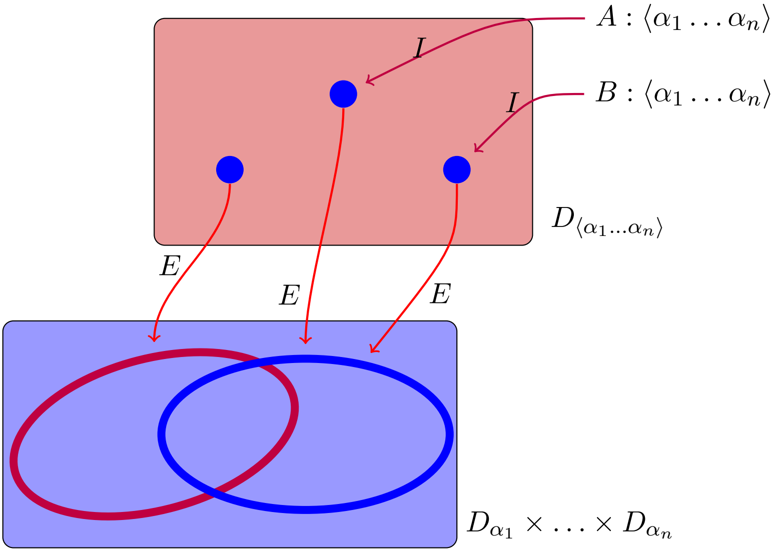
Rigidity Names are rigid designators, i.e. their extensions are world-independent.

- What happens if we give up Extensionality and let the semantic values of expressions be functions in intension?
- Then we would get a theory in which names are rigid designators but the meaning of a name is not determined by its bearer.
- Actually, in a truly intensional theory Carnap Intensionality becomes superfluous.
- And so we can omit that too.

A Truly Intensional Type Logic

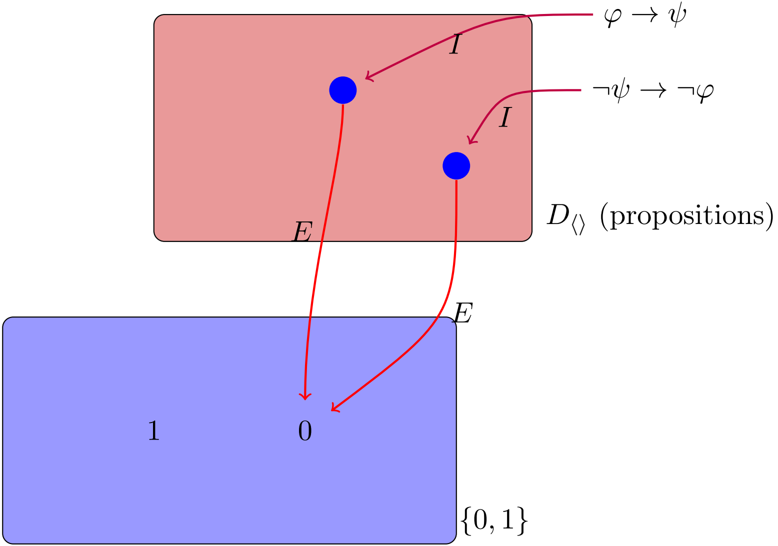
- In the following I will give a very rough sketch of the intensional type logic defined in (Muskens 2007).
- The logic is based on hierarchies of relations, not on hierarchies of functions. For the rest, its language is that of the simply typed λ-calculus.
- In the following pictures intensional models for this language are illustrated.
- A function I sends expressions to their intensions and a function E sends intensions to their extensions (see also Fitting 2002).
- Different intensions can be associated with the same extension and 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



$A : \langle \alpha_1 \ldots \alpha_n \rangle$
$B : \langle \alpha_1 \ldots \alpha_n \rangle$
I
I
$D_{\langle \alpha_1 \ldots \alpha_n \rangle}$
E
E
E
$D_{\alpha_1} \times \ldots \times D_{\alpha_n}$

Propositions



The Logic

- Interpretation in models can be made precise and it turns out that the logic enjoys a generalised complete proof system in which the usual logical operators $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\forall$, $\exists$ and $=$ behave classically.
- But in which the Extensionality axiom is not provable: $\nvdash \forall XY(\forall \vec{x}(X\vec{x} \leftrightarrow Y\vec{x}) \rightarrow \forall Z(ZX \rightarrow ZY))$
- The usual rules for λ-conversion are not present, but can be added consistently.
- A rough and ready characterisation of the logic is: ordinary classical type theory minus Extensionality.
- (If the functions E_α are required to be injective our models essentially become Henkin's generalized models and if additionally surjectivity is required we get full models)

Gentzen Calculus

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

$$\frac{}{\Pi, \varphi \Rightarrow \Sigma, \varphi}\,[R] \qquad\qquad \frac{}{\Pi, \bot \Rightarrow \Sigma}\,[\bot\mathsf{L}]$$

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

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

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

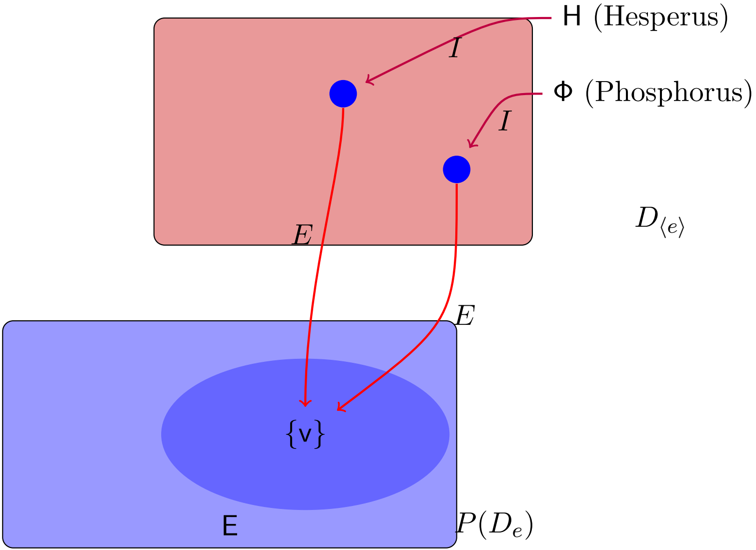
if the constants $\vec{c}$ are fresh

Names I

Given a truly intensional logic such as the one just defined, a theory of names can take the following form.

- Ordinary proper names are predicates.
- They are singular in the sense that their extensions are singletons.
- Meanings are represented by lambda terms and combine with the help of application and type shifters.
- Among the type shifters is Partee's type shifter A (Partee 1986), which we identify here with $\lambda P'P.\exists x(\mathsf{E}x \wedge P'x \wedge Px)$, where E is an existence predicate.

Hesperus and Phosphorus



Names II

- Names are singletons: $\exists x \forall y(\mathsf{N}y \leftrightarrow y = x)$, for all names N.
- (2) a. Zeus $\rightsquigarrow \mathsf{Z}$
  b. Zeus $\rightsquigarrow \lambda P.\exists x(\mathsf{E}x \wedge \mathsf{Z}x \wedge Px)$
  c. Zeus smiles $\rightsquigarrow \exists x(\mathsf{E}x \wedge \mathsf{Z}x \wedge \mathsf{S}x)$
- (3) a. Phosphorus $\rightsquigarrow \Phi$
  b. is Phosphorus $\rightsquigarrow \Phi$
  c. Hesperus is Phosphorus $\rightsquigarrow \exists x(\mathsf{E}x \wedge \mathsf{H}x \wedge \Phi x)$
  d. Hesperus is Hesperus $\rightsquigarrow \exists x(\mathsf{E}x \wedge \mathsf{H}x \wedge \mathsf{H}x)$
- (3c) and the singularity requirement entail that $\forall x(\mathsf{H}x \leftrightarrow \Phi x)$, but $\mathsf{H} = \Phi$ does not follow and (3c) and (3d) can be distinct.
- Hence, it is possible to know (3d) a priori, without knowing (3c) a priori.

Worlds

- Worlds can be viewed as certain properties of propositions.

  W1 $\forall w(\Omega w \to \neg w\bot)$
  W2 $\forall w(\Omega w \to (w(A \subset B) \leftrightarrow \forall \vec{x}(w(A\vec{x}) \to w(B\vec{x}))))$
  W3 $\Omega(\lambda p.p)$

  Here Ω stands for 'is a world'. Some more axioms are necessary.

- W1 and W2 + definitions of logical operators entail

  a. $\forall w(\Omega w \to (w(\neg\varphi) \leftrightarrow \neg(w\varphi)))$
  b. $\forall w(\Omega w \to (w(\varphi \wedge \psi) \leftrightarrow ((w\varphi) \wedge (w\psi))))$
  c. $\forall w(\Omega w \to (w(\forall x\varphi) \leftrightarrow \forall x(w\varphi)))$
  d. $\forall w(\Omega w \to (w(\exists x\varphi) \leftrightarrow \exists x(w\varphi)))$

  'Maximal consistency plus the Henkin property'.

- Write $\Box\varphi$ for $\forall w(\Omega w \to w\varphi)$, '$\varphi$ is globally necessary'.

An Aside

- Introducing worlds as properties of propositions in a truly intensional logic has certain advantages for semantic theory.
- Non-modal sentences can get an interpretation that does not mention possible worlds in any way (explicitly or in the metatheory).
- Zeus smiles $\rightsquigarrow \exists x(\mathsf{E}x \wedge \mathsf{Z}x \wedge \mathsf{S}x)$ (this has type $\langle\rangle$).
- For modal sentences, just add the relevant operators.
- Necessarily, Zeus smiles $\rightsquigarrow \Box\exists x(\mathsf{E}x \wedge \mathsf{Z}x \wedge \mathsf{S}x)$ (also type $\langle\rangle$).
- No need for 'intensionalisation'.

Rigidity

- Global singleton constraint: $\Box\exists x\forall y(\mathsf{N}y \leftrightarrow y = x)$, for all names N.
- Rigidity: $\exists x\Box\forall y(\mathsf{N}y \leftrightarrow y = x)$, for all names N.
- In the presence of Rigidity $\exists x(\mathsf{E}x \wedge \mathsf{H}x \wedge \Phi x)$ entails $\Box(\exists x(\mathsf{H}x \wedge \mathsf{E}x) \to \exists x(\mathsf{E}x \wedge \mathsf{H}x \wedge \Phi x))$
- So if Hesperus is Phosphorus, it is necessary that Hesperus is Phoshorus if Hesperus exists and the usual Kripkean intuitions are formalised.
- But there still is no interchangeability in arbitrary contexts.

Conclusion

- We have given a theory in which names denote rigidly but have a meaning that is not determined by their denotation.
- This shows that there is light between the idea of rigid designation and the idea of direct reference (Millianism).
- The theory does not suffer from the counterintuitive consequences of a prediction that codesignating names can be interchanged in any context.
- The technical move we made consisted in giving up the axiom of Extensionality, using a model theory for type logic that does not validate this axiom.
- This move has independent motivation: it is also good for getting rid of logical omniscience.

References I

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

Kripke, S. (1971).
Identity and Necessity.
In M. Munitz (Ed.), *Identity and Individuation*, pp. 135–164. New York University Press.

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

Muskens, R. (2007).
Intensional Models for the Theory of Types.
Journal of Symbolic Logic 72(1), 98–118.

References II

Partee, B. (1986).
Noun Phrase Interpretation and Type Shifting Principles.
In J. Groenendijk, D. de Jongh, and M. Stokhof (Eds.), *Studies in Discourse Representation and the Theory of Generalized Quantifiers*, pp. 115–143. Dordrecht: Foris.