The World Is Not Enough
– On Complex Types –

Nikolaos A. Kromidas

German Sport University Cologne
Introduction

- Types with actual realizations
- Types with possible, past, future or no realizations
- We talk about unrealized types.
- My thesis:

  If we talk about unrealized types, then we talk about types which are strictly independent from realizations.
The Main Question

• When talking about realizations: We are talking about concrete objects.
• But: What do we mean when talking about types?
• More exactly: How can we determine types without any reference to realizations?
Why is research on unrealized types relevant?

- In a philosophical sense: Because we want to have a logical consistent and ontological homogeneous typology for all types, whether realized or not.
Why is research on unrealized types relevant?

- In a philosophical sense: Because we want to have a logical consistent and ontological homogeneous typology for all types, whether realized or not.
- In a practical sense: Because we do not want to be confused by ontological commitments when talking about unrealized types.
The Focus on Fictional Types

- The paradigmatic example for unrealized types are fictional types.
- Fictional types have by definition no realizations.
The Focus on Fictional Types

- The paradigmatic example for unrealized types are fictional types.
- Fictional types have by definition no realizations.
- If we can determine fictional types, then we can determine unrealized types.
The Focus on Fictional Types

- The paradigmatic example for unrealized types are fictional types.
- Fictional types have by definition no realizations.
- If we can determine fictional types, then we can determine unrealized types.
- If we can determine unrealized types, then we can determine types independent from realizations.
The Focus on Complex Types

- We cannot define any concept without other concepts, we can define concepts only relative to a given and practiced language (Quine-Duhem-Thesis).

- We can determine types only relative to other types; when determining types, we refer to a given language talking about types.
What is a Complex Type?

- Intuitively, complex types are types that contain types (Reicher 1998).
- The included types are logical parts of the complex type (Reicher 1998).
Complex Types and Logical Parts

• For example the type "word" contains the types “written word”, “read word” and “spoken word” as logical parts.
Complex Types and Logical Parts

• For example the type "word" contains the types “written word”, “read word” and “spoken word” as logical parts.

• Every type could be considered to be complex, because every type contains itself as a logical part (Reicher 1998).
Complex Types and Logical Parts

- For example the type "word" contains the types “written word”, “read word” and “spoken word” as logical parts.

- Every type could be considered to be complex, because every type contains itself as a logical part (Reicher 1998).

- Therefore it is sufficient to focus complex types.
What is a Logical Part?

- A type "B" is complex iff there is a type "A" so that necessarily every realization of “B” is a realization of “A” (Reicher 1998).
What is a Logical Part?

- A type "B" is complex iff there is a type "A" so that necessarily every realization of “B” is a realization of “A” (Reicher 1998).
- Thereby "A" is a logical part of "B".
What is a Logical Part?

- A type "B" is complex iff there is a type "A" so that necessarily every realization of “B” is a realization of “A” (Reicher 1998).
- Thereby "A" is a logical part of "B".
- A RECTANGLE is a logical part of a SQUARE -> Every realization of the type SQUARE is necessarily a realization of the type RECTANGLE.
Reicher's Problem I
The World is not enough

- Reichers definition is co-referential: Complex types determine co-references to realizations.
Reicher's Problem I
The World is not enough

• Reichers definition is co-referential: Complex types determine co-references to realizations.

• Therefore Reicher cannot determine fictional types independently from realizations.
Reicher's Problem I
The World is not enough

• Reichers definition is co-referential: Complex types determine co-references to realizations.
• Therefore Reicher cannot determine fictional types independently from realizations.
• To define complex types, we have to add possible, fictional or drafted worlds of realizations.
Reicher's Problem II

- In the case of fictional types Reicher adds drafted worlds and non-relational presented realizations.
- Reicher says for example that Pegasus is a “non-relational presented” flying-horse in a drafted world of myths.
Reicher's Problem III

- Reicher cannot define *fictional* types, because fictional types cannot have any realizations.
Reicher's Problem III

- Reicher cannot define *fictional* types, because fictional types cannot have any realizations.
- Fictional types cannot have non-relational presented realizations as Reicher asserts.
Reicher's Problem III

• Reicher cannot define *fictional* types, because fictional types cannot have any realizations.

• Fictional types cannot have non-relational presented realizations as Reicher asserts.

• Fictional types are only a special case of unrealized types.
Reicher's Problem III

• Reicher cannot define *fictional* types, because fictional types cannot have any realizations.

• Fictional types cannot have non-relational presented realizations as Reicher asserts.

• Fictional types are only a special case of unrealized types.

• That means in fact: Reicher cannot define unrealized types.
Reicher's Problem III

- Reicher cannot define *fictional* types, because fictional types cannot have any realizations.
- Fictional types cannot have non-relational presented realizations as Reicher asserts.
- Fictional types are only a special case of unrealized types.
- That means in fact: Reicher cannot define unrealized types.
- Therefore Reicher cannot define types independent from realizations.
Reicher's Problem IV

- Reicher allows fictional or drafted worlds and non-relational presented realizations.
- That permission has costs: The danger of inflating ontology arises by adding such fictional objects.
Reicher's Problem IV

- Reicher allows fictional or drafted worlds and non-relational presented realizations.
- That permission has costs: The danger of inflating ontology arises by adding such fictional objects.
- Our world is at risk of being lost in a universe of everything.
James Bond Strategy I

• The James Bond strategy avoids this inflation of ontology.
James Bond Strategy I

- The James Bond strategy avoids this inflation of ontology.
- The technical idea is that we determine types only within a language.
The James Bond strategy avoids this inflation of ontology.

The technical idea is that we determine types only within a language.

To this effect I will use a substitutional semantics as a method to determine types with respect to the truth values of our sentences about types.
The James Bond strategy avoids this inflation of ontology.

The technical idea is that we determine types only within a language.

To this effect I will use a substitutional semantics as a method to determine types with respect to the truth values of our sentences about types.

In a substituional way we are able to use a non-referential technique to talk about types.
The James Bond Strategy II

- Wieckowski developed a recursive and compositional substituional semantics.
- I will use this fine-grained substitutional semantics from Wieckowski (Wieckowski 2003) to model the semantics of types.
My Assumptions I

- Presupposed is a language talking about types.
- Types are considered as universals, that means as abstract entities.
- A language talking about types is committed to an anti-particularist point of view.
- We add nothing new if we assume abstract entities such like senses.
My Assumptions II

- Every character string of a language talking about types has a sense.
- The ontological point: We do not add something ontologically new like non-relational presented realizations.
- When talking about types we are only committed to language we are already committed to.
My Assumptions III

- The main assumption is that types are abstract entities captured epistemologically by linguistic structures that express senses.
- Epistemologically it is to say that types are relative to a fixed sense in a given language.
My Assumptions IV

- Thereby types are ontologically abstract entities which are neither created nor necessary (Husserl's “gebundene Idealitäten”).
- To be neither created nor necessary is not something special: Even our actual world is considered in this way.
The Place of Action

- My thesis in detail: Types are abstract entities captured by Wieckowski models (W-Models).
- Analogous to the way Descartes' coordinate systems serve to identify spaces (Quine 2003).
- Similarly, the instantiation of W-Models fixes senses of a given language.
- A semantically fixed sense in a given language is a so called sense-extension.
Informal Type Determination

- A language talking about types has sense-extensions.
- When talking about types we talk about sense-structures captured by sense-extensions of the very language we are using to talk about types.
- Determination of types: Types are sense-structures captured by sense-extensions.
What is a sense-structure?

- In general a structure is related to relationships of entities.
- That means a structure of abstract entities is related to relationships of abstract entities.
- A sense-structure is related to reflections of senses captured by a Wieckowski sematics.
What is a sense-extension?

- A sense-extension is determined by a non-referential Wieckowski semantics of a language with senses.
- Technically, a sense-extension is related to semantic constraints of the *associations* within a W-Model.
The James Bond Strategy Summarized

• I will combine Reicher's typology with the fine-grained substitutional semantics from Wieckowski to form a two-semantics typology.

• My aim is to provide a typology which has a non-referential semantics for types and a referential semantics for realizations.
The Technical Equipment

An example of a Wieckowski semantics:

- Let us assume that we want to know if the sentence "James Bond is well-dressed" is true or not.
The Technical Equipment

An example of a Wieckowski semantics:

• Let us assume that we want to know if the sentence "James Bond is well-dressed" is true or not.

• We start with a language talking about James Bond, for example the sentences written by Ian Flemming.
What means association?

- We collect all atomic sentences in which the name "James Bond" or the predicate “... is well-dressed” are mentioned.
What means association?

• We collect all atomic sentences in which the name "James Bond" or the predicate “... is well-dressed” are mentioned.

• We now say that "James Bond" and "... is well-dressed" are associated with these sets of collected sentences.
What means association?

• We collect all atomic sentences in which the name "James Bond" or the predicate “... is well-dressed” are mentioned.

• We now say that "James Bond" and "... is well-dressed" are associated with these sets of collected sentences.

• Such sets of sentences are the associations of constants and predicates.
What means association?

- We can define that the sentence "James Bond is well-dressed" is true in a W-Model iff this very sentence is in the intersection of the set of sentences which are associated with "James Bond" and "... is well-dressed".
What means reflection?

- Insofar we talk only about nominal constants, pure predicates and atomic sentences.
- I am interested in the senses expressed by linguistic objects.
- The semantically associated sets of sentences expresses a reflected sense-extension (Wieckowski 2003).
Admissible W-Semantics I

- We have no consistent semantics, because “S and S'” could be a contradiction although S and S' are valid in a W-Model.
Admissible W-Semantics I

- We have no consistent semantics, because “S and S’” could be a contradiction although S and S' are valid in a W-Model.

- To achieve consistency we have to make the sense-extension admissible.
Admissible W-Semantics I

- We have no consistent semantics, because “S and S'” could be a contradiction although S and S' are valid in a W-Model.
- To achieve consistency we have to make the sense-extension admissible.
- Admissible W-Models can be set up by determining constraints in a way that enables the associations of a constant (or a predicate) to be definitional, consequent or conform.
Admissible W-Semantics II

- The constraints are characterized by:
  - Nominal definitions for the constants
  - Meaning postulates for the predicates.
Admissible W-Semantics II

- The constraints are characterized by:
  - Nominal definitions for the constants
  - Meaning postulates for the predicates.

- Nominal definitions and meaning postulates behave like origins and axes from coordinate systems: There is no right or wrong place to fix them in a given space or language.
Admissible W-Semantics II

- The constraints are characterized by:
  - Nominal definitions for the constants
  - Meaning postulates for the predicates.
- Nominal definitions and meaning postulates behave like origins and axes from coordinate systems: There is no right or wrong place to fix them in a given space or language.
- A fixed W-Model makes a sense-structure accesible like a fixed coordinate system makes a space accesible.
Type Determination

- $T$ is a type iff there is a fixed W-Model with the nominal constant "$T$" (Kromidas 2009).

- $T$ is simple iff there is only one nominal constant, but complex iff there are two or more constants.

- This definition of types is relative to a background language which fixes the constraints – or so to say “the relevant piece of discourse” (Wieckowski 2008).
The Background Language

- Each type is entrenched to a particular part of a specified discourse – by a background language.
- The background language is an agreement on a meta-level: We agree on how we conceptualize (List 2002).
- For example: We agree on how we conceptualize “James Bond”.
Example I

- For instance the sentence "James Bond is well-dressed" is neither referential nor absolute true or false.
- In a given discourse the type “James Bond” could be determined to be well-dressed and that means:
- In that case the sentence “James Bond is well-dressed” is true within a fixed W-Model.
Example II

• For example: To fix “James Bond” in a given discourse, we could agree to collect the books written by Ian Fleming and extract all atomic sentences that mention James Bond.

• Then we determine the definitional part of the sense-extension by using a piece of discourse as the relevant background language in which we agree.
The Mission I

I will give a sketch of my approach as a whole:

- We have a dimension of non-referential semantics for types, and a dimension of referential semantics for realizations.
- By translating a substituional predication of a type into a denotational predication of a concrete object, we determine a so called realization of a type.
The Mission II

• I regard a (complex) type as determined by the reflected sense of a nominal constant in a W-Semantics.

• This sense is associated with a nominal definition, which can be represented by a list containing all definitional information with respect to the complex type (or so to say to the relevant piece of discourse).
The Mission III

- For instance, the nominal definition of the complex type SQUARE could be the list “the polygon having four angles of 90°, the polygon having four equal sides”.

- Notably we are talking about a complex type which has the type RECTANGLE as a logical part.
The Mission IV

- The nominal definition of a logical part is included in the nominal definition of the complex type.
The Mission IV

• The nominal definition of a logical part is included in the nominal definition of the complex type.

• This inclusion is related to the mentioned constraints, and to the agreement on using a particular background language.
The Mission IV

• The nominal definition of a logical part is included in the nominal definition of the complex type.

• This inclusion is related to the mentioned constraints, and to the agreement on using a particular background language.

• For example: The predicate “... is having four equal sides” is a defining predicate of SQUARE and a conform predicate of RECTANGLE.
The Mission V

- Intuitively, the sense-extension of a complex type captures all information which are compatible with the defining predicates for that particular complex type.
The Mission V

• Intuitively, the sense-extension of a complex type captures all information which are compatible with the defining predicates for that particular complex type.

• In this way the sense-extension of a complex type serves to define the truth of our sentences about complex types.
• Intuitively, the sense-extension of a complex type captures all information which are compatible with the defining predicates for that particular complex type.

• In this way the sense-extension of a complex type serves to define the truth of our sentences about complex types.

• Thus, we can talk about complex types without any reference to realizations.
The Happy Ending

- A complex type reflects a sense-extension, and the included types are semantically connected by association and their respective constraints.
The Happy Ending

- A complex type reflects a sense-extension, and the included types are semantically connected by association and their respective constraints.
- Type “A” is a logical part of type “B” iff necessarily the sense-extension of the defining predicates of type “A” is included in the sense-extension of the defining predicates of type “B”.
The Happy Ending

• A complex type reflects a sense-extension, and the included types are semantically connected by association and their respective constraints.

• Type “A” is a logical part of type “B” iff necessarily the sense-extension of the defining predicates of type “A” is included in the sense-extension of the defining predicates of type “B”.

• For instance, the sense-extension of the type RECTANGLE is necessarily included in the sense-extension of the type SQUARE.
A type is determined in a non-referential way by reflection of senses, and technically captured by a non-referential semantics. We do not need any reference to realizations.

Realizations are characterized by translations from a non-referential semantics of abstract types into a referential semantics of concrete objects.
Summary II

- A fictional type is captured by a non-referential semantics which – by definition – cannot be translated into a referential semantics.
- An unrealized type is captured by a non-referential semantics which has no translation.
Summary II

- A fictional type is captured by a non-referential semantics which – by definition – cannot be translated into a referential semantics.
- An unrealized type is captured by a non-referential semantics which has no translation.
- We have a logical consistent and ontological homogeneous two-semantics typology. We avoid an inflation of ontology by using the James Bond strategy.
Summary II

- A fictional type is captured by a non-referential semantics which – by definition – cannot be translated into a referential semantics.
- An unrealized type is captured by a non-referential semantics which has no translation.
- We have a logical consistent and ontological homogeneous two-semantics typology. We avoid an inflation of ontology by using the James Bond strategy.
- The danger of inflating ontology is banned.
References


Thank you for your attention!