Representation, Meaning and a Concept of Concepts.

Tom de Havas

November 14, 2016

Abstract

One of the primary issues in theories of meaning is the issue of "compositional meaning", how it is that a structure of meaningful elements can be assembled and can have a precise meaning over and above the meanings of the elements of which it is composed. I believe this problem is solved by this paper not just for languages but also other forms of representation.

This paper is not a study of the subtle issues of meaning in modern language, that is a subject for linguists and philosophers, but it rather provides some precise fundamental concepts of representation applicable to languages, signs, diagrams, images and concepts, for people in the field of computing.

It explains exactly how symbols that are assigned explicit meanings can be used to compose systems of symbols that have implicit (or compositional) meaning.

It examines the special case of linear symbol-systems, such as speech or signals on a cable, and amorphous symbol-systems such as biochemical solutions or databases, and how they can be used to represent real-systems with arbitrarily complex topologies. The paper concludes by defining a concept of concepts, a concept being one of the most useful forms of representation.

Contents

1 Prior Work

1.1 Compositionality of Meaning

[Szabó, Zoltán Gendler, "Compositionality", The Stanford Encyclopedia of Philosophy http://plato.stanford.edu/entries/compositionality/]

> The principle of compositionality: The meaning of a complex expression is fully determined by its structure and the meanings of its constituents.
>
> (C') - For every complex expression e in L, the meaning of e in L is determined by the structure of e in L and the meanings of the constituents of e in L.
>
> Questions of structure and constituency are settled by the syntax of L, while the meanings of simple expressions are given by the lexical semantics of L. Compositionality entails (although on many elaborations is not entailed by) the claim that syntax plus lexical semantics determines the entire semantics for L.

However this does not tell us how the meaning comes about from the structure!

1.2 Montague Grammar

[Szabó, Zoltán Gendler, "Compositionality", The Stanford Encyclopedia of Philosophy http://plato.stanford.edu/entries/compositionality/]

> An important principle underlying Montague Grammar is the so called "principle of compositionality". The meaning of a complex expression is a function of the meanings of its parts, and the syntactic rules by which they are combined (Partee and al, 1993) assumes a strict one-to-one correspondence between syntax and corresponding semantic representations;
>
> m(F(e(1),...,e(k))) = G(m(e(1)),...,m(e(k)))
>
> Where;

> e(n) is the nth expression in a sequence of expressions. m(<expression>) is the meaning of <expression>. F(<expression-list>) is a set of syntactic operations on the <expression-list>. G(<meaning-list>) is a set of "semantic partial functions" on the <meaning-list>.

Consider an elementary expression, the word "bigger". It has no meaning in isolation there is no m(bigger). This approach does not help in understanding how for example a visual scene is translated into a textual description.

1.3 Chomsky and Grammar

Chomsky suggests that each sentence in a language has two levels of representation: deep structure and surface structure Deep structure is a direct representation of the semantics underlying the sentence Surface structure is the syntactical representation Deep structures are mapped onto surface structures via transformations.

If one assumes that deep structure is reflected in “parse trees” or “sentence diagrams” one is still left with the same problem of meaning. As far as I can see the transformation from surface grammar to deep grammar is a translation but not an interpretation.

1.4 Semiotics

> “Language is a system of interdependent terms in which the value of each term results solely from the simultaneous presence of the others.” From Course in general linguistics, de Saussure F.
>
> “Where there is choice there is meaning” Jakobson R. and Halle M., Fundamentals of Language [1956]

Many terms do have reality on there own! Choice does not imply meaning.

2 Systems

Bold type indicates that the meaning of a word is being defined.

Definitions:

- A **system** as anything that is analysed[1] as a relationship **over** a union[2] of disjoint **sub-systems**, or is a **component-system** of the analysis.
- A **relationship** over a union of sub-systems, is analysed as a union of **sub-relationships** over sub-unions of the sub-systems, or is a **component-relationship** of the analysis.

  (Component relationships are generally not disjoint and are defined as over sub-unions of the sub-systems because even when a sub-system can be further analysed this does not imply that a relationship over it necessarily can be.)
- A **semi-system** is a system composed of a single component-relationship over a union of sub-systems.

  (A system is generally composed of the union of one or more not-disjoint (overlapping) semi-systems.)

A component is not the same as an element. An element one cannot analyse any further. A component is something that one chooses not to analyse any further.

This definition of a system is chosen to be compatible with current usage of the concept of systems.

Systems can exist without being under a relationship but relationships only exist over systems.

An **entity** is either a system or relationship. There are many cases where what is written applies equally well to either a system or a relationship in which case the word entity is used.

[1] Analysis is defined to be the process of creating a representation of an entity from the real-entity, and is defined later in the text. This is an unavoidable forward reference.

[2] A union is an arbitrary collection of things. It is unlike a set in that a union of unions is simply the union of its members, where as a set of sets is a more than the union of its members.

Be aware that the use of the prefixes "sub" and "super" simply indicate a level up or down in an analysis hierarchy and therefore anything said about entities (i.e. systems or relationships) is also true for sub-entities and super-entities, sub-sub-entities and super-super-entities etc. etc. This fact is used later in recursive definitions.

2.1 Place

A relationship that composes sub-systems into a system may be non-commutative and so may have identifiable places for the sub-systems within it. Consider the three characters (sub-systems) A, B and C composed into a sequence (system). The sequence ABC is not equivalent to the sequence BAC. The system has identifyable places thanks to its relationship.

In view of this instead of simply saying "a relationship is over a union of sub-systems" one says "a relationship is **over placing** a union of sub-systems" meaning that as well as being over them it may also be providing identifiable places for them.

2.2 Background

In choosing a definition for a system, initially simpler options where chosen:

- It was assumed that a component-relationship could not exist over more than two component-systems.
- It was assumed that a component-relationship could only exist over component-systems but not over sub-systems.
- It was assumed that a component-relationships were commutative, i.e. there could not be an order to the sub-systems they were over.

All of these approaches have proved to be limiting in building a concept of representation. It simply is the case that as human beings we do, at times, represent systems where the component-relationships are over, multiple, placed (i.e. ordered), sub-systems.

The main point that has remained is the idea that component-systems are disjoint while component-relationships do not need to be disjoint.

Because of the complexity of systems that include component-relationships over multiple, placed sub-systems it is often easier for the reader to picture first simpler systems, where some of the above assumptions are made, and consider how the principles introduced might apply in these simpler cases.

3 Symmetry

The **context** of an entity is the complimentary union of entities i.e. all the other systems and relationships that are not part of the entity its self.

3.1 Types

When a union of entities cannot be distinguished by analysis of the entities themselves, but can only be distinguished because they are in different contexts, then they are **equivalent**, and are said to be of the same **type**. A type can be assigned a <type-name>, then every **member** of the union is said to be **of the** type <type-name>.

3.2 Classes

When a union of entities all have an instance of one particular type of sub-entity, then they are **similar**, and are said to be of the same **class**. A class can be assigned a <class-name>, then every **member** of the union is said to be **of the** class <class-name>.

In entities of a given class;

- there will be instances of sub-entity-types that are present in every member of the class, called **constants of the class**, and
- there may be instances of sub-entity-types that may or may not be present in any quantity or may vary in at least one member of the class, called **instancial-variables of the class**.

Where there are no variables then the class is also a type. It should always be born in mind that classes do not have to have variables to be a class.

3.3 Processes

Processes are relationships between systems over time, a relationship between “before" and “after". Time of course is simply a number invented to measure processes. Clocks are machines that perform a process to provide this number, time.

Processes can create, destroy and transform other relationships and systems.

Processes can change the constant values of a system thus changing one class of system into another class of system or can change the variable values thus changing the state of the system but not its class, depending on the class in question.

When there exists a process that can change the value of a variable of an instance of a class then the variable is a **temporal-variable**. The **temporal-value** of a system is often called the **state** of the system.

4 Mediums

Just as one can have a union of entities so also one can have a union of types of entities.

A union of types of systems and the types of relationships[3] that can join them, and the types of processes to create and destroy those relationships, is called a **medium** and in it one can compose many different super-systems, which are said to be **compositions** in the medium.

The process-types to create and destroy relationships are known as the **compositional-process-types** of the medium and should not be mixed up with any relationship-types of the medium that just happen to be process-types them selves.

In talking about the process-types of a medium one generally means the compositional process-types and not just relationship-types of the medium that just happen to be process-types.

Call the system-types, relationship-types and compositional process-types of the medium the component-system-types, component-relationship-types and

[3]These are usually non-process relationships but could be process relationships also.

(compositional) component-process-types of the medium, because they are the components in and from which composition may take place.

Some mediums are **hard** (require substantial force/energy) to change but others are **soft** (easy) to change i.e. bricks and mortar or pen and paper.

Comment: Reality its self may be considered to be a medium.

4.1 Super-Mediums

Given a medium it is easy to compose a super-medium. From the component-system-types, component-relationship-types and component-process-types of the medium, compose a union of super-system-types, super-relationship-types and super-process-types to use as the components of a super-medium.

(Remember a component is not the same as an element. An element cannot be analysed any further. A component is something that one chooses not to analyse any further.)

Even if the medium has only a limited number of component-types (1s and 0s perhaps), still an infinite number of super-component-types can be composed for the super-medium (The letters of an alphabet can be composed with the medium of ink on paper, but can be used in turn to compose an infinity of types of composition in the super-medium of writing).

Comment: Naturally this suggests that given any medium, including reality, there is the possibility that it is composed from a sub-medium.

4.1.1 Interfaces

In a super-medium;

- Super-relationship-types are just system-types where some of their sub-relationships can be over external systems. Such sub-relationships are said to be **interface-relationships** of the super-relationship. The other component-relationships are said to be **contained-relationships**.
- Super-system-types are just system-types where some of their sub-systems can be under external relationships. Such sub-systems are said

to be **interface-systems** of the super-system. The other sub-systems are said to be **contained-systems**.

- Super-entity-types can be composed which have both interface-systems and interface-relationships and so are **impure** or mixed however they are rarely necessary.

So -

It is not only possible to compose an arbitrary number of types of super-component-system, it is also possible to compose an arbitrary number of types of super-component-relationship, each over any arbitrary union of instances of types of super-system.

The use of the prefixes "sub" and "super" simply indicate a level up or down in an analysis hierarchy. So going up a level -

It is not only possible to compose an arbitrary number of types of component-system, it is also possible to compose an arbitrary number of types of relationship, each over any arbitrary union of instances of types of system.

5 The Meaning Theorem

Theorem: Given any real-medium one can compose another medium (called a symbol-medium) such that for any real-system-type in the real-medium, one can compose a corresponding symbol-system-type in the soft-medium to represent it i.e. mean it.

Mediums in which representations are composed, are said to be **symbolic-mediums** and the mediums that they can represent are are said to be **real-mediums**.

Any symbolic-medium is also a real-mediums but most real-mediums are not symbolic-mediums (they do not represent anything).

5.1 Diagrammatic Representation

Proof:

- FOR EACH real-component-system-type, compose and assign a unique symbol-component-system-type, to **explicitly mean**[4] that real-component-system-type.
- FOR EACH real-component-relationship-type over placing instances of some real-sub-system-types[5], compose and assign a unique symbol-component-relationship-type with corresponding places, so as to over place instances of the corresponding symbol-sub-system-types, to **explicitly mean**[6] that real-component-relationship-type.
- THEN FOR ANY real-system-type, composed of instances of real-component-relationship-types over placing instances of real-sub-system-types (from the component-system-types up), there exists a unique symbol-system-type, composed of instances of the corresponding symbol-component-relationship-types over correspondingly placed instances of the corresponding symbol-sub-system-types (from the component-system-types up), that **implicitly means** that real-system-type.
- SO FOR ANY real-system-type, there exists a unique symbol-system-type, that **means** that real-system-type.

This approach depends on being able to copy the topology of the real-system in the topology of the symbol system i.e. there is a direct correspondence of each real-system-component and its symbol-system-component, and each real-relationship-component and its symbol-relationship-component. this form of representation is called **diagrammatic representation**.

[4]You can point at trees and say the word "tree" as a means of explicitly defining the meaning of the word.

[5]System-types are defined next and in defining them one has also defined sub-system-types which are just system-types at a lower leval in the analysis hierarchy. At the bottom are component-system-types which have already been defined, so all levels of system-type are thus defined.

[6]explicitly defining relationships can be harder than explicitly defining systems such as trees, but the same principles apply.

Note: Although a symbol-system may have an implicit meaning this does not preclude the assignment of an overriding explicit meaning to specific symbol-systems. So the phrase "bone to pick" in "I have a bone to pick with you" has a meaning that overrides its implicit meaning of picking a bone together. "My taxi has arrived", does not represent ownership in the same way as "My floor tiles have arrived". Real live languages are full of this kind of thing.

5.1.1 Typifiers and Identifiers

In diagrammatic representation, as defined above, there is always a topological correspondence between the real-system and the symbol-system it represents. So for every symbol-sub-entity, in a symbol context, the corresponding real-sub-entity, in its real context, can be identified.

However there are non-diagrammatic forms of representation, which are about to be discussed, where the topology of the symbol-system bares no correspondence with the topology of the real-system it represents.

Serial-symbol-systems such as writing and speach, always have a linear structure, one symbol-sub-system follows another i.e. one letter follows another, one sound follows another. There are also symbol-systems that have no topology at all called Amorphous systems!

These systems overcome the the loss of the corresponding topologies by using multiple instances of a given symbol-sub-entity-type to mean just one instance of the corresponding real-sub-entity-type. This solves the problem of the different topologies but introduces a problem of ambiguity. If there is ever more than one occurrence of a given real-sub-entity-type in the context of the real-system being represented it becomes unclear which one is being represented.

The problem is that an instance of a symbol-entity-type is only a **typifier** it only indicates the existence of an instance of the real-entity-type but does not identify the instance its self. To the typifier must be added an **identifier**, that uniquely identifies which particular instances of the real-entity-type within the real-system is represented. This allows unambiguous representation.

To illustrate the problem there are two red blocks, a green block and a blue block. The statement "The red block is on top of the green block and the blue block is on top of the red block" is ambiguous in what it represents. "Which *red block?"*

By identifying the red blocks the ambiguity is resolved. "The first red block is on top of the green block and the blue block is on top of the second red block", or the other alternative, "The first red block is on top of the green block and the blue block is on top of the first red block"

So by appending an identifier to any instance of a symbol-entity-type that is not unique, in a given context, ambiguities are resolved.

5.2 Amorphous Representation

Amorphous representations have no topology at all. Where as diagrammatic representations mimic the topology of the original. Amorphous representations simply consist of symbols in no particular arrangement i.e. with no relationship between them, no topology!

Examples exist in both biological systems, where solutions contain mixtures of biological free floating compounds, and database systems where records can move freely in a table without effecting the meaning.

Amorphous representation depends on the ability to form mixtures of compounds, the mixture-types are used as symbol-system-types to represent real-system-types. These words are borrowed from chemistry and their meanings extended so;

- **mixture** means, a union of unrelated sub-systems, while
- **compound** means a union of related sub-systems.

Let us examine general amorphous representation in the context of "The Meaning Theorem".

As highlighted above any symbol-component-compound may well need to include an identifier to avoid ambiguities if it is not unique. It is probably best to assume identifiers are always included as it is unlikely that a large system does not have multiple instances of any given entity-type.

- FOR THE real-system-type, to refer to it, arbitrarily compose and **explicitly assign** a unique symbol-component-compound-type.
- FOR EACH real-sub-system-type, to refer to it, arbitrarily compose and **explicitly assign** a unique symbol-component-compound-type.
- FOR EACH real-component-relationship-type over some real-sub-systems, to refer to that real-component-relationship-type, compose and **explicitly assign** a unique symbol-component-compound-type.

  AND FOR EACH place in a real-component-relationship-type, to refer to that place, compose if necessary, and **explicitly assign** a unique (in the context of that real-component-relationship-type) symbol-component-compound-type.
- THEN FOR ANY real-system-type, composed of real-sub-relationships over real-sub-systems, to represent that real-system-type, there **implicitly exists** a unique type of symbol-mixture in which each compound represents one real-sub-system in one place under one real-sub-relationship of the one real-system-type. Each compound being made up as follows;

  1. a symbol-component-compound and identifier referring to the real-system-type, combined with
  2. a symbol-component-compound and identifier referring to one real-sub-system, combined with
  3. a symbol-component-compound and identifier referring to one real-sub-relationship, combined with
  4. a symbol-component-compound and identifier referring to the real-placement.

So the mixture of these compounds represent the real-system but only to the level of its sub-systems, which are themselves not defined at this stage.

Each sub-system must be defined by a mixture of its own, composed according to the above process. The sub-system mixtures can then be mixed together with the mixture above, in order to make a mixture that defines the system but only to the level of its sub-sub-systems.

This process can be repeated down to sub-sub-sub-systems and so on to the level required.

So with the appropriate mixture any system can be represented down to component level as required.

5.3 Serial Representation

Serialisation of representation is essential for most forms of communication including spoken language, writing and electrical communication. In these types of communication symbol-systems are composed of sequences of symbol-sub-systems.

A **sequence** is a composition of sub-systems joined by one type of directed-dyadic-relationship ("follows"), so as to form a chain.

How can a super-medium be composed out of a sequential-medium?

As highlighted before any component-sequence may need to include an identifier-sequence to avoid ambiguities.

- FOR EACH type and instance of real-component-system, to reference that real-component-system, compose and **explicitly assign** a unique symbol-component-sequence-type.
- FOR EACH type and instance of real-component-relationship over some real-component-systems, to reference that real-component-relationship, compose and **explicitly assign** a unique symbol-component-sequence-type.
- FOR EACH type and instance of real-system, to reference that real-system, arbitrarily compose and **explicitly assign** a unique symbol-component-sequence-type.
- THEN FOR ANY type and instance of real-system, composed of real-component-relationships over real-sub-systems, to represent that real-system, there **implicitly exists** a unique symbol-sequence-type, composed of a symbol-component-sequence referencing the real-system; followed by each symbol-component-sequence referencing each real-component-relationship, each of which is followed by its symbol-component-sequences referencing each real-sub-system, it is over.

Then followed by symbol-sequences representing each of the real-sub-systems, down to real-component-system level.

So with the appropriate sequence any system can be represented down to any chosen level.

5.4 Analysis and Synthesis

Analysis - The process of creating a symbol composition that means a subject composition from that subject composition, is called analysis.

Synthesis - The process of creating a subject composition that represented be a symbol composition from that symbol composition, is called synthesis.

The process of converting from one representation to another is **translation**, while analysis and synthesis are **interpretation** processes.

6 Signs

Symbol-components are explicitely assigned to mean real-components, but signs have meanings that are not explicitly defined and nor are they implicitely defined i.e. built of explicitly defined components.

So how are the meanings of signs defined? In two ways.

- The sign-system possesses some similarity with a subject-system and so an interpreter invokes an image of the subject-system. (i.e. silhouette of school children on road signs.)
- The sign-system is a part of a subject-system and so an interpreter invokes an image of the complete subject-system. (i.e. dark clouds are a sign that there may be a storm.)

Signs can be used as symbols and symbol-systems can be composed of both symbol-components and sign-components.

It is important to understands that signs are very dependent on the type of interpreter. For example a silhouette of a bone might be instantly recognisable to a human but a dog would not notice it. However a tiny smell that would go unnoticed by a human would be a clear sign to a dog, of a bone.

It seems reasonable speculate that the origins of spoken and written language may be in signs such as cave paintings or imitation of sounds.

7 Context Dependent Meaning

Symbol-system only composed of symbol-components, are called **literal** symbol-system.

There are symbolic-mediums that are entirely literal but in many common symbolic-mediums such as spoken language it is the case that the meaning of a symbol-system can be different depending on its context i.e. the reality that surrounds it. Thus the literal symbol-system is only part of the entire symbol-system that is interpreted. Interpreters interpret a symbol-system composed of the symbol-components, sign-components and sign-relationships of its surrounding reality and symbol/sign-component-relationships. The literal symbol-system is just the symbol-sub-system composed only of symbol-components-systems and symbol-relationship-components without any signs.

To formalise this it is simply a matter of including appropriate statements into the already formulated "The Meaning Theorem". They are included in italics below.

The Extended Meaning Theorem;-

- FOR EACH real-component-system-type, compose and **explicitly assign** a corresponding symbol-component-system-type.
- *FOR EACH real-component-system-type, there may exist for a given interpreter a corresponding sign-component-system-type.*
- FOR EACH real-component-relationship-type over some of the real-sub-systems (at the lowest level they are real-component-systems), compose and **explicitly assign** a corresponding symbol-component-relationship-type to be over the corresponding symbol-sub-systems (at the lowest level they are symbol-component-systems).
- *FOR EACH real-component-relationship-type over some of the real-sub-systems (at the lowest level they are real-component-systems), there may exist for a given interpreter a corresponding sign-component-relationship-type to be over the corresponding sign/symbol-sub-systems (at the lowest level they are sign/symbol-component-systems).*
- THEN FOR EACH real-system-type, composed of a union of real-component-relationships over real-sub-systems, there **implicitly exists** a corresponding symbol-system-type, composed of the correspond-

ing symbol/sign-component-relationships over the corresponding symbol/sign-sub-systems; recursively down to component-system level.

The last of these statements may also be expressed as the two statements below;-

- THEN FOR EACH real-semi-system-type, composed of a real-component-relationships over real-sub-systems, there **implicitly exists** a corresponding symbol/sign-semi-system-type, composed of a symbol/sign-component-relationship over symbol/sign-sub-systems, recursively down until the sub-systems are actually component-systems.
- THEN FOR EACH real-system-type, composed of a union of real-semi-systems, there **implicitly exists** a corresponding symbol/sign-system-type, composed of the union of symbol/sign-semi-systems.

  There is a one to one correspondence between the real instances and symbol/sign instances of anything.

Thus the principles can be extended to include context dependent meaning and use of signs in the composition of the symbol system.

8 A Concept of Concepts

Define a concept of a real-system as a representation that has types of symbolic-processes, that can be interpreted in order to change a symbolic representation of the real-system, in correspondence, with the way real-processes change that real-system.

A concept of a system, gives the interpreter access to representations of that system when the system its self is not being experienced i.e. the capability of inference, to know the future or past, of that system or know about it when it is somewhere else.