**Vestlandsforsking**
Boks 163, 6851 Sogndal
Tlf. 57 67 61 50
Internett: www.vestforsk.no

**VF-rapport nr. 5/2005**

# Classifications, Taxonomies and Semantic

**Terje Aaberge**

# VF Prosjektrapport

| | |
|---|---|
| **Rapport tittel**<br>Classifications, Taxonomies and Semantic | **Rapportnr.**<br>5/2005 |
| | **Dato**<br>November 2005 |
| | **Gradering**<br>Åpen |
| **Prosjekttittel**<br>Classifications, Taxonomies and Semantic | **Tal sider**<br>29 |
| | **Prosjektnr**<br>5249 |
| **Forskarar**<br>Terje Aaberge | **Prosjektansvarleg**<br>Terje Aaberge |
| **Oppdragsgjevar**<br>Oljeindustriens Landsforening | **Emneord** |

**Samandrag**

We give a brief outline of first order predicate logic, set theory and properties of mappings. These theories are then used to expose the foundations of formal semantics for the description language of a given domain. It is described as a juxtaposition of two languages, the object language used to describe the objects of the domain and the property language that is used to describe the properties of these objects. Both of the languages are based on the syntax of first order predicate logic. We then consider the description of a domain in terms of a hierarchical classification in this framework. We define the notions of ontology and taxonomy, explain how they are related and how a taxonomy imposes a syntactic structure on the object language that mimics a semantic and endows the language with reasoning power.

**Andre publikasjonar frå prosjektet**

| | |
|---|---|
| **ISBN nr 82-428-0254-8**<br>**ISSN**: 0803-4354 | **Pris** |

Revised version 15.01.07

# Semantic, Classes and Taxonomies

Terje Aaberge

*Western Norway Research Institute*
*P.O. Box 216*
*N-6851 Sogndal*
*Norway*
*terje.aaberge@vestforsk.no*

## ABSTRACT

*We expose the foundations of formal semantics for the description language of a given domain. It is described as a juxtaposition of two languages, the object language used to describe the objects of the domain and the property language that is used to describe the properties of these objects. Both of the languages are based on the syntax of first order predicate logic. We then consider the description of a domain in terms of a hierarchical classification in this framework. We define the notions of ontology and taxonomy, explain how they are related and how taxonomies impose logical structures on the object language that mimics a semantic and endows the language with reasoning power.*

# 1 Introduction

Consider the sentences “the water is 5°C” and “5°C is a temperature”. The first sentence is an empirical statement about a system *named* by the word “water”. It is a sentence formulated in the object language for a domain to which water belongs. The word “5°C” is a *predicate* that can apply to water. On the other hand, the sentence “5°C is a temperature” is a statement about the quantity 5°C which refers to a *property* of the water. It is a statement in a property language that applies to the description of properties of the systems of the domain. A description language for the domain is a juxtaposition of these two languages[1]. The aim of this paper is to analyse and describe the semantic structure of this language and to apply it to the definition of a hierarchical classification and the corresponding taxonomy.

The philosophical basis for this work is provided by Tarski [Tarski 1944, 1985] and Wittgenstein [Wittgenstein 1922, 1953]. We apply Tarski’s use of metalanguage to describe the semantic for a language, and we adhere to Wittgenstein’s metaphysical theory and picture theory from Tractatus in the interpretation of what a model is. The picture theory is an expression for the hypothesis that a sentence about reality is true if it pictures an existing state of affairs. More precisely, a sentence *shows* the “logical form” of reality by mirroring it, and it implicitly *claims* that it expresses a statement about reality. In addition, one needs the insights from Philosophical Investigations as a foundation on which to base the description of properties of objects. In Investigations Wittgenstein supplements the standard definition of meaning by also referring to how words and expressions acquire meaning through use; for example, words acquire meaning not only through extension, but also through how they enter into true atomic sentences and valid inferences. These ideas are taken up and partially justified by the results of cognitive linguistics [Croft et al. 2006]. They explain the creative aspects of the mathematical method. Thus, the insights from Tractatus concern the object language and those of Investigations, the property language.

Tarski’s method is to view the relation between the domain of description and the language from the outside. He gives an explicit account of the relation using a second order language, a “meta-language”. The meta-language contains the words and sentences of the language but in addition it contains names for these and predicates characterising them, like meaning and truth. This makes it possible to speak about sentences as well as the

[1] This is also the structure of Description Logic (Baader et al. [2003]) where the two languages are denoted as ABox and TBox . Its origin is the need to make sentences that quantify over the objects and also over the property, though not in the same sentences.

objects and relations they refer to. In particular, he defines the notion of truth as follows (Tarski [1944]):

> "Consider the sentence "*snow is white*". We ask the question under what conditions this sentence is true or false. It seems clear that if we base ourselves on the classical conception of truth, we shall say that the sentence is true if snow is white, and that it is false if snow is not white. Thus if the definition of truth is to conform to our conception, it must imply the following equivalence:
>
> *The sentence "snow is white" is true if, and only if snow is white."*

This sentence is formulated in an object language endowed with an interpretation that gives meaning to the term "snow" and the predicate "white". The standard way of analysing the semantic of a language is to define the interpretation as a couple (D, I) where D denotes a domain[2] and I is an 'interpretation function'[3]. The interpretation function assigns appropriate extensions to the non-logical terms. Thus, if n is a name then I(n) is a system or a set of systems referred to by the same name; if p is a 1-place predicate then I(p) is the set of systems to which the predicate p applies.

The interpretation function I points from the language to the domain. We have chosen to consider the alternative that the 'interpretation function' is pointing the other way, from the domain to the language. This makes it possible to employ the mathematical notion of maps to define interpretations. This choice corresponds to Wittgenstein's picture theory. It is also the point of view that has been adopted in physics and is reflected in the structure of theories of physics (Piron [1973]).

Maps are therefore key elements in the meta-language considered in this paper. A map is used to express the naming of systems. Maps are also used to express assignments of predicates to systems. They appear as symbolic expressions for observations. Together these maps characterise the semantics of a description language for the domain D and they are used to define truth. Maps are also simulating the abstraction of properties from the systems and the naming of the properties in terms of predicates.

It follows from the above outline that considering a sentence we are operating on three levels, the object described, the claim expressed by the

[2] The elements of a domain of description are called systems.
[3] The interpretation function is not a function (or map) in the mathematical sense of a function. It does not necessarily have unique values.

sentence and the characterisation of the claim. This appears more clearly in the following reformulation of Tarski's definition of truth

> *the sentence "snow is white" is true if and only if there exists an object that carries the name "snow" and to which the predicate "white" applies.*

The meta-language therefore also needs symbols for the objects of the domain and for the sentences as objects. Sentences as objects are named by applying quotation marks as in "snow is white". " " thus symbolises a naming map for sentences.

In this paper we restrict ourselves to the study of 1-ary predicates and classifications. We will discuss the case of 2-ary predicates and categorisation in a forthcoming paper.

## 2 The Object Language and Meta-language

The object language L(D;N,P) for a domain D is an interpretation based on D of a first order predicate logic L(N,P). It is defined by sets of maps relating systems of the domain to names and predicates.

Naming is symbolised by a map[4]

$$\nu : \mathrm{D} \to \mathrm{N};\ \mathrm{d} \mapsto \nu(\mathrm{d})$$

from the domain D to the set of names N, that to a system d associates the name n by $\nu(\mathrm{d}) = \mathrm{n}$. $\nu(\mathrm{d}) = \mathrm{n}$ is thus a sentence in a *meta-language* ML(D;N,P). It expresses that the system d is named n.

A fact about a system d is expressed in the description language L(D;N,P) by an *atomic sentence* $\mathrm{p}\,\nu(\mathrm{d})$ that is the association of a predicate p to the name $\mathrm{n} = \nu(\mathrm{d})$ of the system. It reads "n is p". An atomic sentence can also be represented as a map

$$\pi : \mathrm{N} \to \mathrm{P};\ \mathrm{n} \mapsto \pi(\mathrm{n})$$

$\pi(\mathrm{n})$ is thus a synonymous formulation of pn if and only if $\pi(\mathrm{n}) = \mathrm{p}$ since it relates n and p.

[4] See appendix

The naming map for the sentences in a the meta-language is[5]

$$\sigma : \mathrm{L(D;N,P)} \rightarrow \mathrm{S};\ \mathrm{s} \mapsto \sigma(\mathrm{s})$$

Sentences associating the name of a sentence in L(D;N,P) to a predicate characterising the sentence are atomic sentences in the meta-language.

A sentence is characterised by whether it is *true* or not. True t and false f are therefore predicates in the meta-language. The claim that a sentence s is true is expressed by the atomic sentence $\mathrm{t}\sigma(\mathrm{s})$ which itself might be either true or false. This attempt to capture truth thus leads to an infinite regress. However, we may connect truth to the verification or falsification by observation of the claim expressed by the sentence. The map

$$\tau : \mathrm{L(D;N,P)} \rightarrow \{\mathrm{t,f}\}$$

that to a sentence assigns one of the alternatives true or false according to whether what the sentence claims is or is not the case, simulates the procedure of verification. The condition

"$\mathrm{t}\sigma(\mathrm{s})$ is true" if and only if $\tau(\mathrm{s}) = \mathrm{t}$

replaces the infinite regress by referring truth to 'empirical' verification.

If the sentence $\mathrm{p}\nu(\mathrm{d})$ is *true* then the predicate p is said to *apply* to the system d.

The *meaning* $<\mathrm{n}>$ of a name n is the system that is referred to by the name n,

$$<\mathrm{n}> = \mathrm{d} \text{ if and only if } \mathrm{n} = \nu(\mathrm{d})$$

The *meaning* $<\mathrm{p}>$ of a predicate p is the set of systems to which a predicate applies

$$<\mathrm{p}> = \{\mathrm{d} \mid \tau(\mathrm{p}\nu(\mathrm{d})) = \mathrm{t}\}$$

d is said to *satisfy* the condition $\tau(\mathrm{p}\nu(\mathrm{d})) = \mathrm{t}$ if and only if the equality holds, i.e. "$\mathrm{p}\nu(\mathrm{d})$" is true.

[5] The standard notation for $\sigma(\mathrm{s})$ is "s".

The words “truth”, “meaning” and “satisfaction” belong to the meta-language ML(D;N,P). They are implicitly defined by the above definitions. It is these implicit definitions that together with ‘empirical’ verifications gives them meaning. The source is thus the meaning of “true” as the set of all true sentences of L(D;N,P),

$$<\mathrm{t}>=\{\mathrm{s} \mid \tau(\mathrm{s})=\mathrm{t} \text{ and } \mathrm{s} \in \mathrm{L}(\mathrm{D};\mathrm{N},\mathrm{P})\}$$

We will distinguish between two kinds of 1-place predicates. Examples of predicates of the first kind are the predicates denoting colours, weights, positions etc. Examples of predicates of the second kind are philosopher, human and Norwegian. The predicates of the second kind are definable in terms of predicates of the first kind (see §5). Only predicates of the first kind appear in atomic sentences.

Predicates of the first kind can be classified according to mutual exclusiveness. Two predicates that are not simultaneously applicable to a system belong to the same class. For each class there exists a map

$$\delta : \mathrm{D} \to \mathrm{P};\ \mathrm{d} \mapsto \delta(\mathrm{d})$$

These maps are called *observables*. An observable $\delta$ is uniquely defined by a choice of class as range $R_\delta$ and the condition that $\mathrm{p}=\delta(\mathrm{d})$ for $\mathrm{p} \in \mathrm{R}_\delta$ and where p *applies* to d. In terms of the observables, the meaning of a predicate is therefore

$$<\mathrm{p}>=\{\mathrm{d} \mid \delta(\mathrm{d})=\mathrm{p}\}$$

The observables simulate acts of observations. Each observable is associated with a measuring device and a set of instructions that leads to a result that is represented by one of the predicates in its range. In terms of the observables, truth for atomic sentences is characterised as follows

> “pn” is true if and only if there exists a system d and observable $\delta$ such that $\mathrm{n}=\nu(\mathrm{d})$ and $\mathrm{p}=\delta(\mathrm{d})$

## 3 Operational Definitions

The observables simulate observations and thus express the empirical relations that constitute the interpretation of L(N,P) based on D. Accordingly, they constitute the empirical foundation for the object language and its interpretation. The basis for the interpretation is the global condition of truth of the atomic sentences. This condition is expressed by the commutativity of the diagrams

$$\begin{array}{ccc} & \pi & \\ N & \rightarrow & P \\ \uparrow \nu & & \uparrow \delta \\ D & \equiv & D \end{array} \qquad \text{i.e. } \pi(\nu(\mathrm{d})) = \delta(\mathrm{d}), \quad \forall \mathrm{d} \in \mathrm{D} \tag{1}$$

The maps are related to the particular domain D and its description. Notice that the condition of commutativity fixes the choice of $\pi$ for each $\delta$. For each $\delta$ there is a unique $\pi$ that has the range $R_\delta$.

An observation is not a simple act of seeing. It involves a "measuring" or "observational" device, a manual of instructions for the manipulations of the device, the application of the device according to the manual and the recording and interpretation of the result. These elements are not independent. They constitute a logically consistent wholeness.

The complexity of an actual observation depends on how directly a system or a phenomenon can be observed. However, this should not obscure the fact that even direct observations that appear as automatic and unconscious actions are not simple. Thus, the observation of the number of eggs in a basket consists of moving the sight from one egg to the next and each time counting one more. The final result is interpreted as a number in the system of natural numbers that is kept as a standard of measure.

The determination of the position of a system in space is a slightly more intricate example. It is based on the choice of a reference point, a set of three orthogonal axes crossing at the reference point and the choice of a measure of length. The position of the system is then given by three numbers measuring the number of unit lengths from the reference point to the orthogonal projections of the position of the system on the axes. The measures are determined by lying measuring rods of unit length one after the other without spacing until one reaches the projection point, and then recording the number of rods counted.

For both of the examples the records are numerical values. The observation of colour of a system is an example where this is not the case. The measuring device is then a colour chart where each of the colours is named and the rule of application is to compare the colour of the system with the colours on the colour chart. If, for instance, the colour of the system is identical to the colour named "red" on the chart, then the colour of the system is taken to be red.

In spite of the difference of complexity between these examples it is easy to see that they share some common characteristics. The observations described all involve the use of a scale based on a standard of measure and the result follows from a comparison between a representation of the standard and the system. Moreover, the definition of the standard of measure determines the interpretation of the result. These means are resumed in an operational definition that is an account of the applied standard of measure and the instructions to be performed to make an observation. Each observable is associated with an operational definition. It 'describes' the corresponding measurements.

## 4 The Property Language

### 4.1 Properties

1-place predicates of the first kind refer to *properties* of systems. A property is something in terms of which a system manifests itself and is observed, and by means of which it is characterised and identified. The properties of a system are thus in a natural way 'mentally' separated from the system. The separation is made possible by the fact that the 'same' property is possessed by more than one system.

To an observer a system appears as a collection of properties. The separation is expressed by the commutativity of the following diagrams

$$\begin{array}{ccc} P & \equiv & P \\ \uparrow\delta & & \uparrow\rho \\ D & \underset{\varepsilon}{\rightarrow} & E \end{array} \qquad \text{i.e. } \delta(\mathrm{d}) = \rho(\varepsilon(\mathrm{d})), \quad \forall \mathrm{d} \in D \qquad (2)$$

Here E represents the set of properties of the systems in D, $\varepsilon$ is a map that simulates the mental separation of properties from the systems and $\rho$ stands for a map that to a property associates the predicate referring to it. It is an abstract construction and it is therefore a theoretical task to characterise E. Thus, it might be that E is a natural extension of the set of properties that can be associated to the systems of the domain and that this is reflected in the set of predicates available in the language. The pluralities and the natural numbers are an example.

In the case of coloured systems for example, the condition of commutativity means that if a system appears as red then it possesses the property redness.

It is assumed that each 1-place predicate refers to a unique property.

**4.2 Axiomatic**

The maps $\rho : \mathrm{E} \to \mathrm{P};\, \mathrm{e} \mapsto \rho(\mathrm{e})$ can be considered as naming maps for the properties. But the properties do also possess relations that impose a structure on E and that are expressed as relations among the predicates in P. To describe them we need a set of predicates Q, a logic L(P,Q), a formal system L(E;P,Q) and a formal language L(D,E;P,Q), the property language. The formal system is associated with the diagrams

$$\begin{array}{ccc} & \varphi & \\ \mathrm{P} & \to & \mathrm{Q} \\ \uparrow\rho & & \uparrow\chi \\ \mathrm{E} & \equiv & \mathrm{E} \end{array} \qquad \text{i.e. } \varphi(\rho(\mathrm{e})) = \chi(\mathrm{e}) \qquad (3)$$

that is analogous to diagrams (1).

From the set of sentences in L(E;P,Q) describing E we can select a subset, the axioms, from which all the other sentences, the theorems, can be derived by means of logical inferences (Blanché [1955]). Some of the axioms are directly related to the operational definitions and can be established on the basis of these. They are supplemented by other axioms that are expressing 'properties' of the properties. These are referred to as laws.

The set of axioms contains the "full" information of the structure of E. They are implicit partial definitions of the concepts considered. They do not give the full meaning of these concepts, but limit their range of possible interpretations. Only by choosing a domain D and interpret some of the concepts by means of the commutative diagram

$$\begin{array}{ccccc} & & & \varphi & \\ & & \mathrm{P} & \to & \mathrm{Q} \\ & & \uparrow\rho & & \uparrow\chi \\ D & \to & \mathrm{E} & \equiv & \mathrm{E} \\ & \varepsilon & & & \end{array}$$

will the exact meaning of all the concepts involved be determined. The diagram thus expresses the semantic structure of the property language.

**Remark:** The elements of the diagram (3) separated from any relation to a domain D and considered in their own rights, constitute a mathematical theory. The mathematical theories of geometrical spaces or manifolds are prominent examples A manifold (space) is defined as an abstract entity in terms of a given class of coordinatisations, i.e. by diagrams like

$$\rho : E \to P;\ e \mapsto \rho(e)$$

(Boothby [1973], p 52). D is then the source of an interpretation of the mathematical theory.

**Example: Arithmetic**
The properties of and relations between pluralities are expressed by means of the Peano axioms for arithmetic that read as follows[6]:

1. o is a number[7]
2. the successor[8] of a number is a number
3. no two numbers have the same successor
4. o is not the successor of any number
5. any property of o and of the successor of any number that possesses the property is a property of any number

The word “number” is used as a predicate of the second kind in Q denoting the class of numbers and also as a variable referring to the elements of the class, and the word “successor” is a 2-place predicate in Q. By assumption the axioms describe a structure on E that is mapped by $\rho$ onto the description given by the Peano axioms. The Peano axioms therefore describe the structure of the set of pluralities. The first four axioms are directly abstracted from the operational definition that describes counting. The fifth axiom is purely structural. It expresses the induction law.

**Remark:** The axioms give a definition of the elements of the class named “number” in terms of the 2-place predicate “successor”. It is only a partial definition of the elements of the class number. Not only the sequence of natural numbers satisfies the Peano axioms but also sequences like 0, 2, 4, 6, …. or 1, 1/2, 1/4, …. (Russell [1919]). It is only if we interpret the concepts of number in concordance with an operational definition of $\delta$ that their meaning is unambiguously determined

### 4.3 Observables
The maps $\delta$ are representing operational definitions and each of them

[6]The complete set of Peano axioms also contains the definitions of addition and multiplication.
[7] 0 is the name for an element of E. The word “number” is a class name.
[8] Successor refers to a relation between the elements in E. It is a 2-place predicate of Q.

symbolises the kind of observations that are realised by its application. The range of an observable $\delta$ is the set $R_\delta \subset P$ of predicates that are denoting the possible results of the observations applying the corresponding operational definitions. By the commutativity of the diagrams (1) and (2), each $\delta$ defines unique $\pi$ and $\rho$ with which it shares the range $R_\delta$. Together they express the association of a property to a system by observation. It is therefore natural to call these maps observables and use the same name to denote each of the observables $\rho, \pi$ and $\delta$ that share range.

Each 1-place predicate denotes a unique property and moreover, a map can only have one value for each argument, it thus follows that a system can only be assigned one predicate from the range $R_\delta \subset P$ of an observable. On the other hand, properties in E can be classified according to their possible assignments. Two properties belong to the same class if and only if one and the same system cannot possess them (at a given moment of time). A system cannot be red and green. Red and green therefore belong to the same set, the set of colours. The corresponding set of predicates $R_c$, denoting the colours, is by assumption the range of an observable, the observable “colour”, and thus associated with an operational definition. Other examples of observables are age, mass, position and velocity.

The “classification” of 1-place predicates in terms of observables is not a partition. A set of observables leading to partitioning classification of the predicates is said to be complete. The complementary observables to a complete set must necessarily have functional relations to observables in the set. Some of the functional relations depend on the system considered and their specification is then called a *model* of the system.

## 5 Descriptions, Classes, Ontologies and Taxonomies

A description language for a given domain is the juxtaposition of an object language and a property language. Because of their association the observables $\delta, \pi, \rho$ constitute a bridge between the object language and the property language with the observable $\delta$ as the central pillar. The diagram

$$\begin{array}{ccccccc} & \pi & & & & \varphi & \\ N & \rightarrow & P & \equiv & P & \rightarrow & Q \\ \uparrow \nu & & \uparrow \delta & & \uparrow \rho & & \uparrow \chi \\ D & \equiv & D & \rightarrow & E & \equiv & E \\ & & & \varepsilon & & & \end{array} \tag{4}$$

i.e. the composition of the diagrams (1), (2) and (3), expresses the semantic structure of the description language.

Though their function differs the observables in a triple are denoted by the same 'name'. "Colour" is an example of that. Thus, while $\delta$, by $\delta(d) = \text{red}$ associates the colour red to a system d, $\pi(n) = \text{red}$ stands for the sentence "n is red" and $\rho(e) = \text{red}$ is the statement that claims that red is the manifestation of a property e that is redness. The observation that a system is red expressed by the sentence "n is red", is therefore to be interpreted as expressing that the system whose name is n possesses the *property* redness. This interpretation is justified by the commutativity of the diagram (4).

**5.1 Description**

A domain D is described by a set of observables, i.e. $\{\delta^1, \delta^2, ...., \delta^n\}$. The description $\mathcal{D}$(d) of a system d is the sentence $p^1\nu(d) \wedge p^2\nu(d) \wedge ... \wedge p^n\nu(d)$. It is the conjunction of all the atomic sentences associating a predicate that applies the system to its name $\nu(d)$, i.e. such that $\delta^1(d) = p^1$ and $\delta^2(d) = p^2$ and … and $\delta^n(d) = p^n$. Accordingly,

$$\mathcal{D}(d) = p^1\nu(d) \wedge p^2\nu(d) \wedge ... \wedge p^n\nu(d)$$

is a description of the system d if and only if

$$\tau(p^1\nu(d) \wedge p^2\nu(d) \wedge ... \wedge p^n\nu(d)) = t$$

$\mathcal{D}$(d) is therefore also a definition of the word $n = \nu(d)$ denoting the name of d.

A description based on all the observables is usually redundant. Because of relations between the observables there are, in general, more observables then is needed to distinguish a system from other systems of the domain. A minimal set that is sufficient for the identification of systems is said to be complete, i.e. if the set

$$\begin{aligned}
&\{d \mid \tau(p^1\nu(d) \wedge p^2\nu(d) \wedge ... \wedge p^n\nu(d)) = t\} = \\
&\{d \mid \delta^1(d) = p^1 \ \& \ \delta^2(d) = p^2 \ \&...\& \ \delta^n(d) = p^n\} = \\
&\langle p^1 \rangle \cap \langle p^2 \rangle \cap ... \cap \langle p^n \rangle
\end{aligned}$$

contains at most one system whatever are the values $p^1, p^2, ..., p^n$ chosen for the observables $\delta^1, \delta^2, ...., \delta^n$. Notice that the above formulae express equivalent definitions of the same set.

**5.2 Classes**

It is necessary to distinguish between two kinds of observables. This is a result of the problem encountered when one wants to describe change and that is illustrated by the following statement

> *change do not exist, because if something changes than it is no longer the same and we cannot say that anything has changed*

This semantic problem was a central theme in Greek philosophy. One of their solutions, which have become a basis for physics, is to distinguish between two kinds of properties, properties that do not change in time and thus serves to identify the system and properties that change. The latter are called state properties. The properties of the systems are therefore also classified as identification and state properties. The state properties form a space called the state space of the systems. Classifications are usually made only with respect to the identification properties, in terms of the corresponding observables. Class names are then often used to denote the systems because the set of actions associated with a system (its function) depend on the class thus defined, even if their actualisation depend on the state of the system.

We will illustrate this kind of description with an example, the game of chess. The domain is the chess board on which there are 8x8 squares each of which can potentially hold one of 32 pieces of the game (the *systems* of the domain) and 32 positions extending the board that can hold one piece each. The identification observables for the systems are colour C with values black and white, and form F with values pawn, rook, knight, bishop, king and queen. The state properties are the 96 positions of the extended board denoted by (-2,A), …, (10,H) and the set of possible moves on the board, i.e. the set of translations. The state observables are $P_f$ and $P_r$ that measure the position of a piece on the board, and $T_f$ and $T_r$ that measures the moves of the pieces, i.e. the translations along the ranks and files respectively. $P_f$ has values (A, …, H), $P_r$ has values (-2, …, 10), $T_f$ has values (0, ..,7) and $T_r$ has values (0,..,10), i.e. they count the of steps a piece is moved.

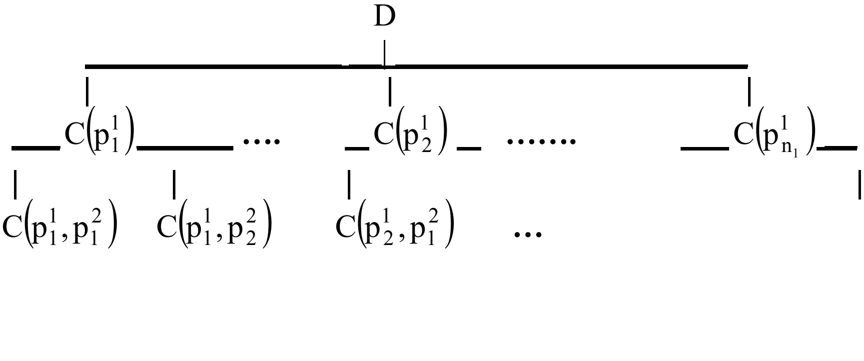
The pieces of the game are classified according to the identification observables and the classes named accordingly, black king, white pawn etc. The possible moves of the pieces are determined by their class. A white pawn, for example, can only move forward by one step if the path is free and diagonally by one step if it encounters a black piece at that place which can then be beaten, i.e. moved to one of the positions on the extension of the board. We notice that a given piece is completely identified at any time by giving the actual value for each of the pair of observables $P_f$ and $P_r$ or $T_f$ and $T_r$ that applies to the piece. The state of a piece is thus represented by the couple (move, position).

The classification procedure can be formalised. Thus, consider an independent set of identification observables, say $\{\delta^1, \delta^2, ...., \delta^n\}$. Let $\delta^j$ be one of the observables and let $p^j_{i_j}$, $i_j = 1, 2, 3, .., n_j$ denote its values. Then

$$C\left(p^j_{i_j}\right) = \left\{d \mid \delta^j(d) = p^j_{i_j} \ \& \ d \in D\right\}, \ i_j = 1, 2, 3, .., n_j$$

is a classification of D with respect to the observable $\delta^j$. It is characterised by the fact that all the elements of a class $C\left(p^j_{i_j}\right)$ share the property denoted by $p^j_{i_j}$ and that it is a partition of D; i.e. the classes are mutually independent and their union is the whole of D. By repeating the same procedure for each of the classes with another observable, one constructs the next level of a hierarchical classification. The procedure can be continued recursively until the set of observables is exhausted. If there are n observables, there will be n+1 levels, the top level being the domain D.

The classes can be arranged in a two dimensional hierarchy, a lattice, by inclusion. The hierarchy depends on two choices, of a complete set of independent observables and on the order in which these are applied. Assuming that these choices are made, the hierarchy can be represented graphically by




The vertical lines express inclusion of sets. Each of the classes below a horizontal line is a partition of the class above.

Class is a ‘property’ of a collection of systems of the domain. The name of a class is thus an element of P. It is a predicate of the first kind in the object language and a name in the property language.

Let $o$ be a naming map for the classes in a hierarchical classification

$$o : H \to P; C \mapsto o(C)$$

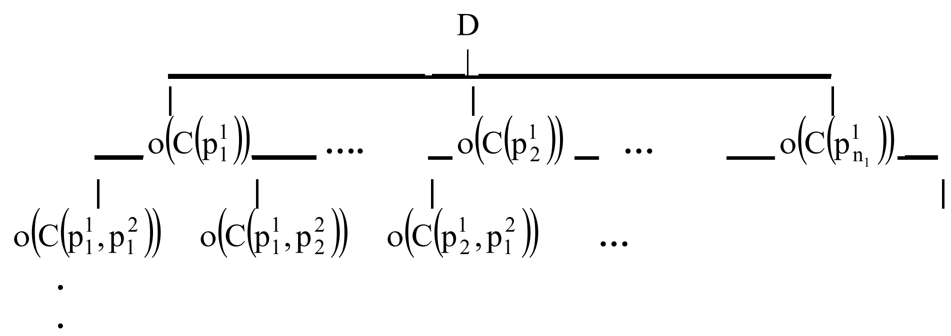
where H is the set of classes. The *ontological definition* of the name $o\left(C\left(p^1, p^2, ..., p^m\right)\right)$ is

$$o\left(C\left(p^1, p^2, ..., p^m\right)\right) \equiv p^1 x \wedge p^2 x \wedge ... \wedge p^m x$$

The word “class” is a predicate in the property language and the sentence “$o(C)$ is a class” is by construction a true sentence in the property language. The classes have only one observable, their cardinality, i.e. the number of elements they contains.

**5.3 Ontologies and Taxonomies**

The class names are naturally arranged according to the classification hierarchy. The predicates denoting the classes are serving as node titles,




The taxonomy endowed with the *ontological definitions* for the variable names is a linguistic representation of the classification. The ontological definitions impose a syntactic structure on the taxonomy that mirrors the class inclusion relations and create semantic relations between the taxonomy titles. There is two important notions, family and heritage, that characterise these relations. Titles on the same level belong to the same family and a title denoting a class C inherit meaning from titles denoting superclasses of C.

The position of a title in the taxonomy considered in its own terms, therefore limits the possible meaning of the titles without completely determining them. Titles on a higher level have a less precise meaning than titles on a lower level. The structure defined by the taxonomy endows the object language with reasoning power by means of the tautologies

$$\tau\left(p_i^1 n \wedge p_2^2 n \wedge \ldots \wedge p_i^m n \rightarrow p_i^1 n \wedge p_i^2 n \wedge \ldots \wedge p_i^{m'} n\right) = t \quad \text{for } m > m'$$

This mirrors the semantic relation

$$\text{"if } d \in C\left(p_i^1, p_i^2, .., p_i^m\right) \text{ then } d \in C\left(p_i^1, p_i^2, .., p_i^{m'}\right) \text{ for m>m'"}$$

i.e. if $n = \nu(d)$ then the sentence "$p_i^1 n \wedge p_i^2 n \wedge \ldots \wedge p_i^m n$" is true and "$p_i^1 n \wedge p_i^2 n \wedge \ldots \wedge p_i^{m'} n$" will also be true.

A node title is the name of a corresponding class. By convention the same word is also used to denote any element in a class. It is then a variable name and referred to as a variable over the class. The convention can be improved by using lower case and capital first letters to distinguish between the use of a word as a variable name or as a predicate of the second kind.

Using this convention, the sentences “dog is a Predator”, “spaniel is a Dog” and ”spaniel is Predator” are examples that show that titles higher in the taxonomy have more precise meaning then titles the titles lower in the taxonomy. In fact, the syllogism

all dogs are Predators
all spaniels are Dogs
all spaniels are Predators

is automatically valid.

One may also use relations to determine classes or categories by satisfaction. If r is a 2-ary predicate in a language based on D and n is the name of a given system, then

$$\{d \mid r(n, \nu(d) \;\&\; d \in D)\}$$

denotes the class of all the systems that has the relation r to the one named n.

## 6 Concluding Remarks

The above account of the semantic of a *description language* for a domain D is a partial formalisation. It has been described as the juxtaposition of two languages each of which is based on the syntax of first order predicate logic. The language has a well-defined alphabet; a set of formation rules and at least some axioms can be established on the basis of our description. The description has been made in a meta-language employing the resources of first order predicate logic, set theory and the theory of mappings. A formal study of the semantic of the description language is thus possible applying first order predicate logic. This reveals relations between syntactic and semantic relations that do not appear in our study. Its study is outside the scope of the present paper.

The class of languages to which our description applies is general enough to include all kind of knowledge representation languages, also scientific theories like those of physics. The study has, however, mainly been motivated by problems encountered in modelling systems and processes that are to be implemented as computer models.

Sowa [2000] lists five abstraction levels for modelling[9]

1. *Linguistic*. The level of arbitrary concepts, words and expressions of natural languages
2. *Conceptual*. The level of semantic relations, linguistic roles, objects and actions
3. *Epistemological*. A level for defining concepts types with subtypes, inheritance and structuring relations
4. *Logical*. Symbolic logic with its propositions, predicates, variables, quantifiers and Boolean operators
5. *Implementational*. The level of data structures such as atoms, pointers, lists and other programming notions

The first four levels lists the development that takes place during the construction of models in any science, from simple ideas about the constitution and behaviour of a system to more substantial ones and at the end a mathematical model that can be tested.

Models are subjected to the condition of faithful representation of the systems they picture. Computer models also serve an additional purpose. They are tools by means of which human operators not knowing anything

[9] The list is originally due to Ron Brachman. We have listed the levels in the opposite direction.

about computers and programming might accomplish mundane tasks. Computer models are thus also subjected to the normative judgement of how easy it is to perform these tasks. The user interface must give direct access to all important operations. Because the user interface reflects the logical model that is implemented this puts additional conditions on the modelling. The semantic basis for the model must correspond to that of the intended users. For developers to make workable models it is thus necessary to know how the logical, syntactic and semantic structures of language are related. The motivation of this paper has been to give an account of this based on Wittgensteins picture theory which leads to the extensive use of maps and Kant's distinction between "Ding an sich" and "Ding für mich" from which the distinction between object language and property language follows. Models are formulated in both languages. The taxonomy of classification of the systems of a domain and the descriptions of the systems are models in the object language. A taxonomy, possibly supplemented by relations between the systems, provides a model of the domain, and the description of a system a model of the system. On the other hand, the Peano axioms provide a model, or ontology, of the natural numbers[10]. This is a model in the property language. Most computer models are models of abstract systems. They are thus formulated in the property language. We will deal with this case in a forthcoming paper.

## Appendix: Maps

Interpretation is based on correspondence, equality and identification. By equality, one means same meaning. It is expressed by the sign "=", an example is "1+2 = 3". Identification also refers to symbolic representation. To express this one employs the sign "≡". This sign means "identical to" or "defined by". Correspondence is represented by maps between the set of elements that are known and the set of elements that are to be interpreted. The correspondence is symbolised by sign "→" and the relation between the elements by the sign "↦".

A map *f* from a set *A* to a set *B* that associates elements in *A* to elements in *B* is thus expressed by

$$f : A \to B; a \mapsto f(a) = b$$

*A* is called the *domain* for *f* and *B* the *target*. The set $\{f(a) \mid a \in A\} \subset B$ is the *range* of *f*. The set $\{a \mid f(a) = b\} \subset A$ is called the *inverse picture* of *b* in

[10] The use of the word model in model theory is the opposite of our use here. In model theory one would say that the natural numbers is a model of the Peano axioms.

*A*. If $\{f(a) \mid a \in A\} = B$, then *f* is said to be *onto*. *f* is into if the cardinal number for $\{a \mid f(a) = b\}$ is 0 or 1 for all *b*. A map *f* that is into and onto is *one-to-one*. The map has then a unique inverse denoted $f^{-1} : B \to A$. A necessary condition for the existence of one-to-one maps between two sets *A* and *B* is that they have the same cardinal number.

Maps can be composed. If

$$f : A \to B;\ a \mapsto f(a) = b$$

and

$$g : B \to C;\ b \mapsto f(b) = c$$

are maps from *A* to *B* and *B* to *C*, then

$$g \circ f : A \to C;\ a \mapsto g(f(a)) = c$$

is a map from *A* to *C*.

Compositions of maps can be represented by diagrams, for example

$$\begin{array}{ccc} & g & \\ B & \to & C \\ \uparrow f & & \uparrow h \\ A & \equiv & A \end{array}$$

The diagram is said to be commutative if $g \circ f = h$, i.e. $g(f(a)) = h(a)$ for all *a* in *A*.

Mathematical sets are sets with a structure, i.e. that there exists relations between the elements. This structure is reflected in the properties of the maps that preserve the structure. The study of mathematical structures can thus also be referred to the properties of maps.

**ACKNOWLEDGMENT**

I am indebted to The Norwegian Oil Industry Association (OLF) for partial support.

## References

Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P. eds. [2003]: *'The Description Logic Handbook, Theory, Implementation and Applications'*, Cambrigde: *Cambrigde University press*

Blanché, R. [1955]: *'L'axiomatique*', Paris: *Presses universitaires de France*

Boothby, W. M. [1973]: '*An Introduction to Differentiable Manifolds and Riemannian Geometry',* New York: *Academic Press*

Piron, C. [1973]: *'Foundations of Quantum Physics'*, Boston: *Benjamin Inc.*

Russell, B. [1919]: *'Introduction to Mathematical Philosophy'*, London: *Routledge*

Sowa, John E., Knowledge Representation: Logical, Philosophical and Computational Foundations, Brooks/Cole (Pacific Grove 2000)

Suppes, P. [1972]: *'Axiomatic Set Theory'.* New York: *Dover Publications Inc.*

Tarski, A. [1944]: *'The Semantic Conception of Truth and the Foundations of Semantic', Philosophy and Phenomenological Research 4, pp. 341-75*

Tarski, A. [1985]: '*Logic, Semantic, Metamatematics (second edition)*' Indianapolis: *Hackett Publishing Company*

Wittgenstein, L. [1922]: 'Tractatus Logico – Philosophicus', London: Routledge