Software Werkstatt

A terminology model approach for defining and managing statistical metadata

Comments to : R. Karge
☎ (49) 30 - 6576 2791

fax (49) 30 - 6576 2801
mail reinhard.karge@run-software.com

Content

1 Introduction

A terminology model is a mean for recording and presenting expert knowledge in a certain area. The terminology model provides both, a method for discovering subject expert knowledge and a structured and formalized way of presenting expert knowledge.

Using terminology models for describing a certain subject field has been practiced since more than five years in different international projects for describing the area of statistical metadata:

- Neuchatel Group for Classifications (1999-2002)
  "*Terminology model for classifications and related concepts*"
  Statistics Sweden, Norway, Switzerland, Denmark, run-Software (Germany)
- METANET – A network of excellence (2000-2003)
  "*Reference model for statistical metadata*"
  EuroStat
- Neuchatel Group for Variables (2003-2005)
  "*Terminology model for statistical variables and related concepts*"
  Statistics Sweden, Norway, Switzerland, Netherlands, Bureau of Labor Statistics USA, run-Software (Germany)

The terminology model is a method for describing expert knowledge within a certain subject field. The terminology model covers two aspects:

- Terminology aspect - relates to the used concepts (term, meaning, usage etc.) in a subject field (e.g. official statistics)
- Model aspect - provides a formal structured presentation of concepts, which can be used for many different purposes.

The goal of a terminology model on the on hand is to provide detailed and high quality definitions and, on the on the hand, to support all phases of an IT application development process. Thus, the terminology model acts also as a sort of interface between subject area experts and IT technicians.

The paper considers the structure of a terminology model based on essential aspects of knowledge representation, i.e. the goal of the terminology model is a structured knowledge representation.

There will be a rather complex structure for terminology models, but many relevant knowledge aspects can be covered by very limited elements of the terminology model. Hence, different terminology model levels will be defined, which can be used for different purposes.

2 Knowledge presentation

It is not the intension of this paper to provide a complete definition of knowledge. Instead, it will focus on certain aspects of knowledge, which are subject of a terminology presentation. Mainly, knowledge aspects described here result from basic principles in human language and terminology structure.

2.1 Knowledge aspects

Human language and concept systems are based on four basic principles. By building classes, human language expresses common features of certain objects, by classifying objects, human language expresses, which objects belong to a class. By describing behavior, human language expresses the way objects may change or interact with other objects. Causalities allow humans to express, when or under which conditions objects will (re)act.

Class

Real world phenomena will be called **objects**, regardless whether objects exist as "things" or just as ideal concepts. Then, a class describes a collection of objects. Classes are associated with names (terms, designations), that relate to the class concept on the one hand and to the class extension in the other hand.

Often, classes are based on intensional definitions by describing, what type of objects a class consists of. But there are also extensional class definitions (especially for conceptual objects), where the elements of a class are listed explicitly, i.e. there are two typical ways of defining classes:

a) extensional definition, which describes the class by referring to objects in the class
b) intensional definition, which provides a conceptual definition in terms of specific object properties for the objects belonging to a class.

Classes defined are not given by nature, but are an expression of human knowledge and the specific view to a certain subject area. Thus, an object may belong to any number of classes, which just express different views to the objects.

There is a high level of standardization when considering object classes defined in human language. Most nouns refer to object classes and have translations in most languages. This is possible only, because there is a common understanding of these classes the terms in each language refer to.

Classification – where objects belong to

Classification is a knowledge process that divides an object collection into different subclasses. Thus, a classification defines a set of classes, which usually defines a complete division of a given set of objects into distinct subsets (subclasses).

The concept of a class in a classification is described as **category**. Thus, classifications are typically described as set of categories, which conceptually define the classes for a classification.

Categories can be defined as abstract categories not referring to a certain set of objects. Then, a category can apply to different object collections (e.g. the category female may apply to a set of persons, animals or nouns). In this case, the category itself has an intensional definition, only, and does not refer directly to a specific class. Only, when applying the category on an object collection, the category creates a subclass. Thus, e.g. borderline definitions are not part of the category definition (because they are dependent on the object collection the category applies on), but of an **applied category**, which describes the application of a category on a certain object set.

> **Example**: *Red* can be considered as category and “red haired persons” as an applied category of red. Wile the category red refers to a very abstract concept, the applied category refers to a class of objects (“red haired persons”).

An applied category defines a class, when applying on a certain population. In other words, we could say, the extension of a class can be defined by describing a population (set of objects) and an applied category. On the other hand, applied categories often refer to a specific property of objects in the population (super class), e.g. “color of hair”. In this case, the subclass “red haired persons” can be defined as persons with: color of hair = red. I.e. the category turns into a property value for Person. Since persons of the subclass “red haired persons” have all the same value red for the property “color of hair”, the property becomes a fixed or delineating property.

By referring to a number of categories, each classification has an extension and becomes also a class (of categories) itself. As a class, a classification may correspond to a category on a higher level, which defines a subclass of categories in a larger set of categories. Thus, a set of classification may form a **classification system**, which provides a hierarchical structure of classifications.

Classifications and classification systems are typical examples for extensional class definitions. However, when a classification is a class itself, a classification represents also a type, which is the type of categories, the classification consists of.

Object types and properties – what objects are

Intensional class definitions are usually based on defining objects types. Typing reflects a knowledge aspect based on conceptual class definitions. Object types are defined by selecting a number of fixed and variable object properties, which describe relevant characteristics for the objects of the class. Thus, classes and object types reflect two different perspectives to a set of objects.

Defining object types is based on defining a number of object properties as fixed and variable properties. Properties are typically divided in four categories:

a) generalization – describes a concept, that relates to a more general object type.

b) attribute – describes a fixed or variable characteristic of an object

c) part of – describes the parts an object consists of

d) association – describes the relations to other objects

Those properties are commonly defined in concept systems as concept relations and characteristics (ISO 1087), but also in database modeling (ODMG). There are more links between concepts, i.e. this list of categories is not complete. But those categories are the most relevant property categories.

There is not always a clear distinction between property types, e.g. whether a wheel is a part of a car or associated with a car depends on the view one has to the car. Thus, an object type defines not an object as such, but a certain view to an object, and properties do not describe all the properties of an object, but the relevant properties from a certain perspective.

Each property can be considered as an object, which, again, is associated with an object type, i.e. each property has an object type. Thus, we have alternative type – property – type (or property – type – property) relation, which provides a method of defining a concept system in a consistent way until any level of detail.

Since there might be a number of views to an object, any object can be described by a number of object types.

Rules – how objects are

Object types and properties allow a static definition of objects and object classes. An advanced knowledge approach, however, is to describe, how objects may change, i.e. how objects behave. Typically, the behavior of objects is described as common behavior of objects of a given object class ("birds can fly" or "things can fall").

One approach is describing behavior as rules. A rule describes how objects change from one state to the other or how objects interact with other objects, i.e. changing the state of other objects. Thus, rules are used for

a) Describing constraints or validation rules for objects and properties

b) Describing state transitions, for describing the way how objects change

c) Describing state operations for producing derived information

Defining rules is an advanced approach and not typical for conceptual definitions when starting defining concepts. Later on, it becomes, however, important, because building applications is impossible without knowing the rules, according to which objects in the application behave as an image of reality.

Causalities – when objects are

Describing causalities is an even more advanced approach. Causalities describe the cause or reason, which activates a certain behavior. Describing causalities reduces the human interaction in a system or application, because the application can detect automatically, what is necessary to do and when.

The methods of describing causalities are not yet developed very well. Two ways of describing causalities are:

a) Describing causalities as dependent state transitions, i.e. as state or state transition that causes a reaction (activating a rule). (e.g. when a person has been born (cause) it must be registered in the birth register (reaction).

b) Describing state transitions as consequence of history e.g. sending a mail (reaction) when a person has birthday. In fact, this does not differ in principle from a), but time plays a special role and thus, it becomes useful to consider those causalities separately.

There are not so many areas dealing with causalities compared to static concept definitions, but the interest in causalities is growing.

Summary

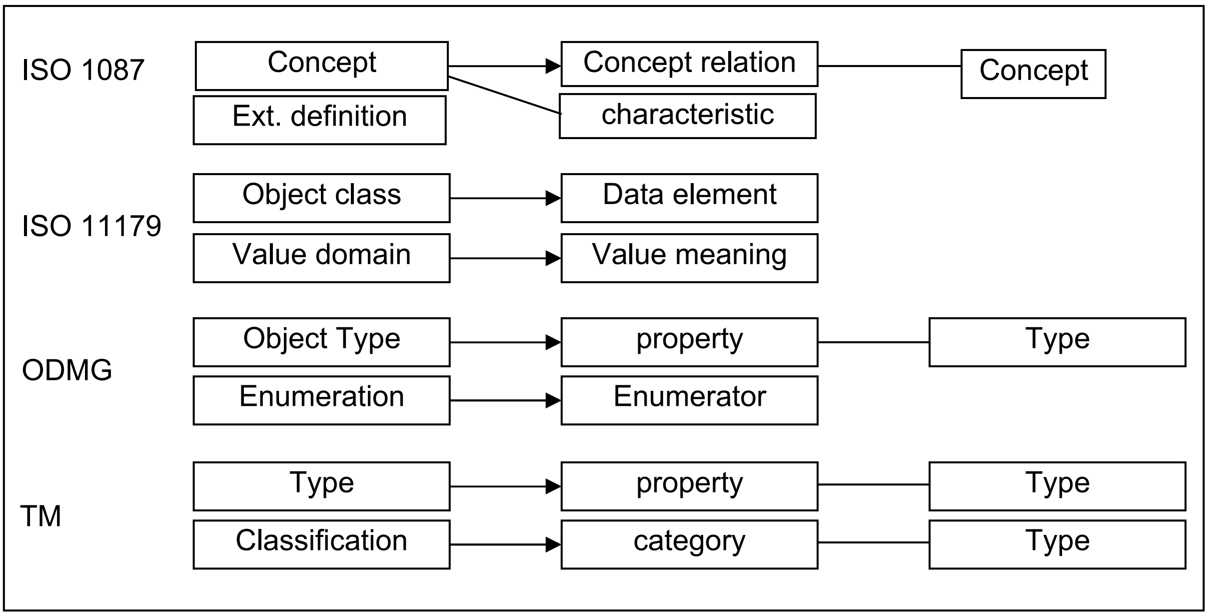
The static knowledge aspects can be described in terms of classes or object types, properties and categories or values. Behavior and causalities describe the dynamic aspects of real world phenomena.

2.2 Standards reflecting knowledge aspects

Defining a terminology model covers basic elements of ISO 1087 concerning the definition of concepts and the relation between concepts. Thus, a terminology reflects a subset of concepts defined in ISO 1087. On the other hand, it is a bit more specific, since it requires a more detailed classification of concepts.

There are some reasons for providing some more specialized concept definitions than provided in ODMG, ISO 1087 and ISO 11179.

- ISO 1087 allows defining characteristics for describing object properties, but it does not consider characteristics as concept or as linked to a concept. One may describe children as characteristic of a person, but there is no link from children to person, telling, that children ARE persons.
- ISO 1087 provides the extensional definition, which corresponds to a classification, but it does not consider categories as concepts and it does not provide a link from a category to a concept.
- Usually, concepts are referred to by names (or terms). Having a large number of concepts, it becomes difficult to define unique names for concepts or names become very long. Dividing general concepts in abstract concepts and related concepts solves this problem.
- The distinction between general and individual concepts is not sufficient, since there are many concepts, which are neither individual nor general concepts according to the definition (temperature, size etc.), because they do not correspond to an object but rather to an object property.
- ISO 11179 provides us with object classes, data elements, value domains and value meanings, but it does not support object relations (generic, association, partitive) and we miss a data element type.
- The ODMG approach provides all the required technical features, but no guidelines for standardized conceptual definitions.



The picture above shows the existing and missing links in different models. ISO 1087 supports the *type – property – type* relation by the *concept – concept relation – concept* link.

ISO 1087 does not provide a good support for defining classifications, except the extensional definition of concepts.

Providing high quality concept definitions requires good experts in the subject field, but also an appropriate method of recording those definitions. ISO 1087 could provide such a frame work, but from the IT aspect, it is not structured in an appropriate way. Nevertheless, we will see that most of the information required is already part of a concept system.

ISO 11179 and ODMG provide data models, i.e. structured definitions, which support both, *type* – *property* relation and *classification* – *category* relation. The *category* – *type* relation is missing in both and ISO 11179 does also not support the *property* – *type* relation.

In principle, the first three models have many things in common. The terminology model refers to the experiences of ISO 1087, ISO 11179 and the ODMG standard for object-oriented database models. The terminology model adds the *classification* – *category* – *type* relation, which is a *type* – *category* – *type* relation, since classifications can be considered as types. Thus, the terminology model mainly describes types, properties and categories and the links between those.

The terminology model refers to terminology objects (or object types) rather than to concepts. Terminology objects add a structure to the concept, which is defined implicitly, only, in a concept system.

In praxis, it turned out that the terminology model fulfils the requirements of experts as well as the requirements of IT technicians. More than 10 countries referred to the "*Terminology model for classifications and related concepts*" when building their classification systems. An unknown number of countries (e.g. Australia, Croatia, Switzerland) refer to the Reference Model for building metadata systems in national statistical offices.

3 Terminology Model

The goal of a terminology model is knowledge presentation. The terminology model covers the knowledge aspects discussed in the previous chapter and tries to present those in a structured way. The terminology model tries to refer as much as possible to available standards, but for providing a structured knowledge presentation, it needs further extensions.

Those aspects are important for improving definitions but also for matching IT requirements.

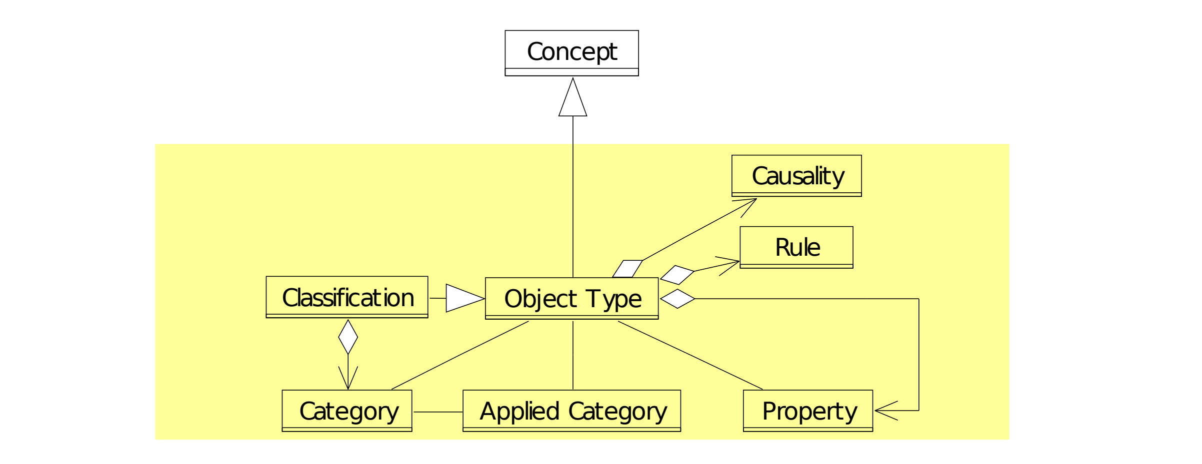


Fig. 1: Concept extensions for terminology model

The following definitions refer to corresponding concepts and terms in referenced standards in the following sequence: (ISO 1087, ISO 11179, ODMG).

Object Type (general concept, object class, type)

Instead of a concept, the terminology model refers to the more specific object type. The object type is a concept in the sense of ISO 1087. In principle, object types are general concepts. Nevertheless, there are types, which do not really have an object extension. In any case, types are not individual concepts.

> **Example**: *Colour* is a concept, but *colour* does not define a class as *person* does. Colour could be an extension definition naming all relevant colours, but it could also be a composite concept consisting of parts RGB.

Object Types consist of extensional and intensional definition. Object types correspond to general concepts.

An object type defines the intension and the extension of an object class. In principle, object types implicitly define a set of objects, which applies to the object type. But an object type may also define an abstract types (as colour), which does not at all refer to object instances.

When an object type describes a class extension, it is usually linked with one or more applied categories that define the borderlines for the objects belonging to this object type within one or more supersets. Since an object type reflects a specific view to a set of similar objects, i.e. there might be several object types applying to the same set of objects, or the object type may apply to a number of (applied) categories.

Classification (extensional definition, conceptual value domain, enumeration)

A classification is an object type that is mainly defined by its extensional definition (this is not the same as the extension defined by a concept, which refers to a set of objects applying to this concept). The objects described in a classification are categories, i.e. the classification as an object type describes the properties of its categories.

The conceptual issue of a classification is to divide a given set of objects into (distinct) subsets, i.e. defining sub classes, which are related to the categories defined in the classification extension.

Category (partitive concept, value meaning, enumerator)

The concept for each sub class is defined as category, which define the meaning of each sub class (associating categories with codes is more a technical issue). A category can be linked to a number of object types for three purposes:

a) defining one or more divisions for the subclass, in which case the category refers to one or more subordinated classifications.

b) referring to an applied category, which refers to an object type that describes the objects in the subclass.

c) Referring to an object type, which describes the properties of a category

> **Example**: A sex classification may divide persons in male and female persons (applied categories). Now, female can be linked to another classification, which divides females in those who have born children and those, who did not. On the other hand, the category female should refer to a general concept of woman, which defined the objects in the subclass female persons and which has an additional characteristic: number of born children. Finally, categories in the sex classification can be described as having the properties title and code, which are defined in the object type the sex classification stands for.

Type links from categories are not supported in any of the mentioned standards, even though they contain essential information about the category. Only data ware house models (e.g. CWM) support a sort of type links for categories.

Categories are very specific and describe individual concepts.

Applied category (none, none, none)

An applied category describes the way a category applies to a certain population of objects. The categories itself are neutral concepts, which may apply on any set of objects (even though the difference between a category and an applied category is often not evident).

Applied categories become necessary for describing the borderlines, which allow associating objects of a given population with a category).

> **Example**: A sex classification, which consists of categories male, female and neutral, may apply on a set of persons, but also on a set of nouns. The borderline definition for female persons (applied category) and female nouns (applied category) will differ essentially. There will be essential differences also between German female nouns and Spanish female nouns, which both define different applied categories for nouns in different languages.

An applied category refers to an object type, which describes the objects classified by the applied category. The object type associated with an applied category may be a specialisation of the object type of the population.

Often, there is no distinction between a category and an applied category, which is not necessary, when the category applies on only one object type.

Property (characteristic/concept relation, data element concept, property)

The intension of an object type is defined by a number of properties, which play different roles in the object type definition. Properties can be

a) generalizations (generic relation)
b) attributes (characteristics)
c) part of (partitive relation)
d) association (association relation)

In the sense of ISO 1087, properties describe concept relations and characteristics. ISO 11179 refers to TM properties as indifferent data element concept.

Each property refers to a type, that tells us, what the property IS:

> **Example**: Asking a person, how her or his children are, implies that these children might be grown up people or even grand parents. Thus, children are probably persons and not really children.

The terminology model considers properties as concepts defined in a limited scope (object type). Thus, considering properties as related concepts, i.e. concepts, which play a role only within this specific object type definition, names for properties need to be unique only within an object type definition. This makes it a lot easier to find appropriate terms for these properties.

In fact, properties are object types, which describe an object class which corresponds to the objects referred to by a property. Thus, properties are similar to applied categories and there are often situations, where it becomes difficult to distinguish between a property and an applied category.

> **Example**: Persons living in a certain region can be classified by living in this region, i.e. the region acts as an applied category (people living in Germany). On the other hand, one can consider Germany as an object instance of type country. Persons are associated via the property “country”, which provides an association to the country, where the person is living..

Thus, one may express applied categories as properties or reverse, depending on the perspective one has to a certain knowledge area. The difference is that properties provide ad hoc classifications, while categories are conceptually pre-defined.

Rule (none, none, method)

A rule is a concept that describes a certain behavioral aspect of an object. Rules can be defined for object types, but also for properties. Any number of rules can be defined for on object type or property (which is also an object type).

Rules can be defined for different purposes:

a) A consistency rule describes a condition for a certain concept, which cannot be violated (e.g. age of a person is greater than or equal to 0). Sometimes, consistency rules are similar to value domains, i.e. value domains provide specific consistency rules.

b) A state transition rule describes the way objects behave to itself or other objects (e.g. how the egg behaves, when it falls with a certain speed to the ground). State transition rules describe the way objects change and objects interact with other objects.

c) A state operation rule describes how information can be derived from one or more objects (getting the total income of all German inhabitants).

The behavior described by a rule is typical for the object type or property, i.e. rules are general concepts.

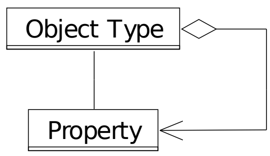
Causality (causal relation, none, exception)

The methods for describing causalities are rather limited in the standards referenced above. One may describe causalities as concepts with a causal relation, but there is nothing specific for describing cause or reason and reaction. Exception is also a very weak feature for describing causalities. The current terminology model does also not involve causalities so far.

3.1 Terminology model parts

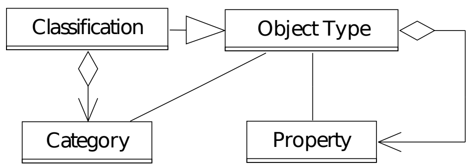
The terminology model is divided in different parts, which refer to different phases when developing a terminology model. Moreover, each part can be described on different levels (core level and extended level).

3.1.1 Static terminology model



The static terminology model mainly describes the *type - property – type* relations. A relevant concept of an expert area is considered as type, which has properties. Properties refer to types again, which then must be described by its properties etc. Such static models have been developed with the Neuchatel Group Classification and Variable model.

Even though this is a very simple definition of the world, the static terminology provides a good conceptual definition of many relevant concepts on the one hand, and the base for generating database models and other technical information on the other hand.

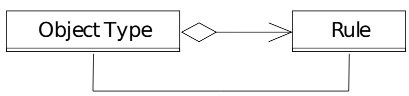


A small extension of the static terminology model includes classification s, which are often necessary for defining categories of object classes defined in the model.

The properties in the model should be described according to its categories, since it makes a relevant difference, whether a company is a person or has person. In the case a company is a person the company inherits all the properties that are defined for the person.

For simplification, the terminology model does not differ between categories and applied categories. Usually, on the level ob knowledge representation, categories and applied categories are identical. Still, it is possible to describe applied categories separately, when it becomes necessary, i.e. when a classification can apply on object sets of different types.

3.1.2 Behavioral terminology model



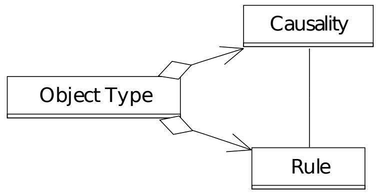
The behavioral terminology model becomes necessary, when going to develop an application. Nevertheless, the behavioral aspect is a conceptual aspect which is known only by the experts. Thus the terminology model provides the behavioral part for describing behavior of objects of a given class. The semantics follow the common rules, i.e. more special object types (concepts) inherit the behavior from more general object types (concepts).

Since rules or behavior may generate an output (result), rules can be linked with an object type, which defines the output.

Behavior or rules are usually defined for objects within a class. The terminology model allows describing affected objects (side effects) of a behavior, but there is no explicit support for describing cooperative behavior. This problem can be solved by defining composite object types or parameters, which may influence the behavior.

3.1.3 Dynamic terminology model (planed for the future)

The dynamic terminology model will include causalities. Causalities are described as complex state transitions or time (as cause) and a reaction (behavior).



3.2 Terminology model object types

Defining a terminology model means defining the class and classification concepts for a subject field. There is a core level for defining a terminology model and an extended definition level. Characteristics for the extended definition level have a grey background.

As shown in Fig.1, the terminology model considers all entities in the model as concepts. Thus, the core of the model is the concept definition.

The following sections describe the details of the terminology model. The definition of the terminology model is a practical example for a terminology model, i.e. the terminology model definition is a terminology model for the terminology model.

3.2.1 Object type

The object type is a specialization of the ISO 1087 concept definition. An object type is either complex (containing one or more properties) or elementary (not containing properties). Text or number are typical elementary object types.

An object type can be a classification. Properties are not considered as object types but as properties, only.

Property	Description
concept	The object type inherits the characteristics from the concept. [--> Concept]
properties	List of related concepts that describe the object properties for the concept. [--> Property]
rules	Any number of rules can be defined for an object type for describing consistency, state transition and state operation rules. [--> Rule]
extension	The extension of a class concept describes the objects corresponding to the class concept.

3.2.2 Property

A property is a concept that defines a characteristic or concept relation for the object type it belongs to. The designations for properties and its synonyms must be unique within the scope of the object type. Properties do not have characteristics, but usually, they refer to an object type, which describes the characteristics for the property[1].

Properties may have different types, which reflect the role of a property within the object type.

Property	Description
concept	The property inherits the characteristics from the concept. [--> Concept]
type	Object type that defines the properties or categories for the property. [--> Object Type]
rules	Any number of rules can be defined for a property for describing consistency, state transition and state operation rules. [--> Rule]
relation type	defines the type of the relation between the property and the referenced object type. [--> Property Types]
importance	Defines the importance of the property. [--> Importance Levels]

3.2.3 Concept

The concept defines “a unit of knowledge created by a unique combination of characteristics” (ISO 1087). Here, the concept is defined as a set of characteristics, which are used to describes the different concept types within the terminology model. The terminology model is using a subset of characteristics defined in ISO 1087. Concepts are identified by terms and may have synonyms. Terms for concepts and synonyms must be unique in the subject area the concept belongs to or within an object type.

Property	Description
designation	The name is a single word or group of words that identifies the concept.
syn.: term	*Example*: *Classification Version*
Definition	A description or definition of the named concept.
synonyms	List of synonyms that can be used instead of the concept name.
example	One or more examples describing the defined concept. Examples are marked by italic letters. *Example: This is an example for an example.*
Usage	Use cases for the concept

[1] This sounds a little bit artificial: Why should one not define the characteristics of the property immediately in place. This is possible, too, but it seems to be a better way defining those in a separate object type, since this allows reusing the set of properties for other properties.

Property	Description
Rationale	Allows describing the reasons and the motivation for defining the concept
provenance	Contains remarks about the ownership and history of the concept
source	Source of the concept when being defined in an other place or area
external references	References to external sources
Internal references	References to concepts in this subject field

3.2.4 Classification

The classification is an object type, which provides a schema for dividing a given class of objects into subclasses. A classification contains a number of categories, and it may also contain properties the categories consist of.

Defining classifications becomes necessary, when conceptually objects are divided into categories.

Property	Description
concept	The classification inherits the characteristics from the object type. [--> Object Type]
categories	List of categories that describe the subclass extensions. [--> Category]

3.2.5 Category

A category defines the extension and borderlines for a subclass of a given object set (class). When the defined subclass has an intensional concept beside its extensional definition, the category refers to the corresponding class concept. When the sub class has one or more divisions into subclasses again, it may refer to one or more classifications.

Property	Description
concept	The category inherits the characteristics from the concept. [--> concept]
extension	The extension type describes the object set, which is defined by the category. There are many different ways for defining the extension, but typically, it is defined by describing borderlines. Example: People, that earn between 2000 and 3000 EURO per month.
sub-divisions	One or more subordinated classifications can be provided, which define s subsequent division of the class into subclasses.. [--> Classification]

Property	Description
type	Object type, which describe the intensional concept of the category, i.e. the intensional definition of objects in the class defined by the category. [--> Object Type]

3.2.6 Rule

A rule defines an object type or property specific behaviour. Rules can defined for different purposes as consistency check or getting derived information.

Property	Description
concept	The rule is a concept and inherits the characteristics from the concept. [--> concept]
type	The rule may provide a result, which defines one or more objects of a given type. [--> Object Type]
rule type	defines the purpose of the rule [--> Rule Types]

3.2.7 Property types

Property types define the relation between a comprehensive concept (object type) and its properties. Designations for categories have been chosen according to corresponding concepts in ISO 1087.

Category	Description
characteristic	A related concept is a characteristic, when it can be considered as integral part of the concept. The type for a characteristic can be an object type or a classification. *Example: Size for person (each person has a size).*
generic	A related concept describes a generic relation between the defined object type and the object type referenced by the property. The type for a generic property is always an object type (and no classification) Note, that a generic property is not the generic concept itself, but refers to the generic concept via type, i.e. the generic property represents the generic relation.
partitive	A partitive property defines a part or a list of parts belonging to the object type defining this property. A partitive property defines the relation to a partitive concept (and not a partitive concept itself). The partitive property always refers to an object type.. Example: '*Front wheels' and 'back wheels' are partitive properties of 'car'. Both refer to the object type 'wheel'*.

Category	Description
association	An association property defines a relation to one or a list of objects of a given type[2]. An association property defines an association relation between two concepts. The type of an association property is always an object type.

3.2.8 Importance levels

The importance level is a classification that describes the importance categories.

Category	Description
essential	The characteristic must be defined for the concept
optional	The characteristic can be omitted.

3.2.9 Rule types

Rule types define different purposes of rules. This list is not complete, but defines relevant rule types as known so far.

Category	Description
consistency	A consistency rule describes a condition, which cannot be violated.
transition	A transition rule defines the way an object or related objects change under certain circumstances.
operation	An operation rule defined a way of deriving information from a single or a set of objects.

[2] It is not necessary, that associated objects belong to the same type, but it makes things a little bit easier.

4 From Terminology Model to Database

The goal of building terminology model are detailed definitions for a subject field, but also a base for building a data model based on the terminology model. According to our experience, statistical experts are very satisfied with terminology model definitions, since the method requires clear and detailed definitions and it reflects the role of related concepts in different scopes (class concepts).

Moreover, IT technicians benefit a lot from terminology models. In case of statistical metadata, the Neuchatel Group terminology models as well as the reference model have been used to build a statistical knowledge base Bridge, an exhaustive metadata system used in several statistical offices in Europe.

One of the important ideas for using a terminology model as base for a data model is, that users can use designations defined in the terminology model for accessing data in the knowledge base or metadata system, i.e. a view must be provided to the database, that corresponds exactly to the terminology model. Therefore, a semantic standard is derived from the terminology model, which creates technical designations from designations in the terminology model. The semantic standard becomes the base for generating a semantic interface, which maps the items of the terminology model to database items (properties/attributes, objects/relations).

4.1 From Terminology Model to Semantic Standard

Communication on technical level requires a semantic interface, because technical names differ from names in the terminology model (e.g. spaces are not allowed in designations). To be able to communicate between different technical (syntax) standards, it makes sense to define a semantic standard based on the terminology model, which fits to several syntax standards (database model).

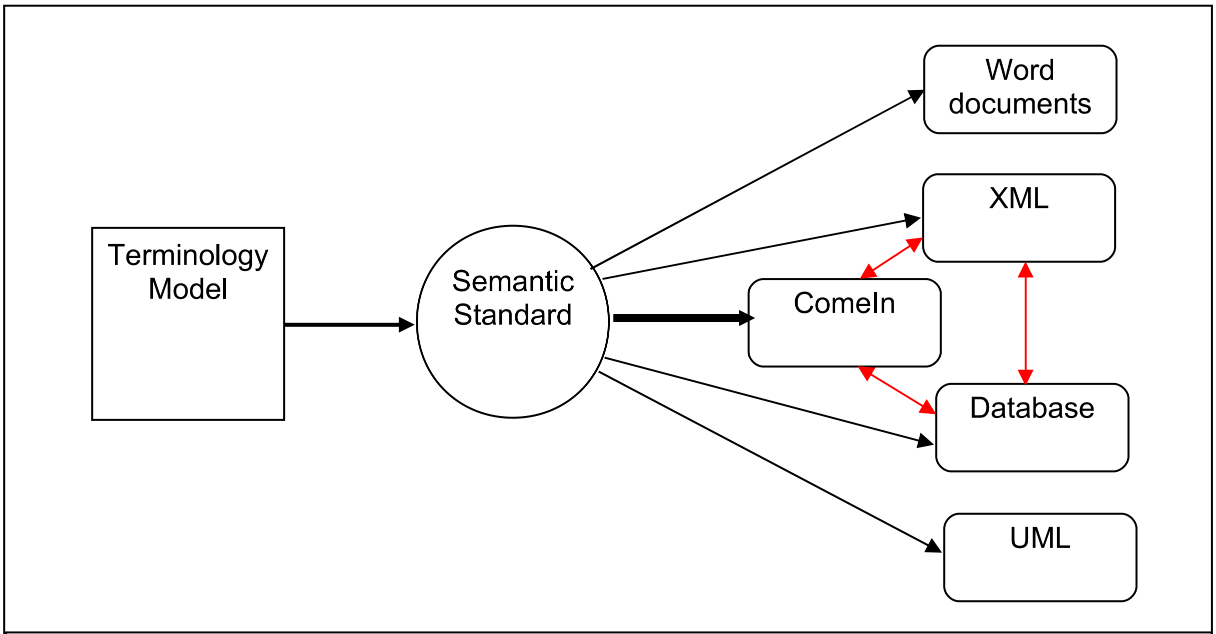


Fig. 2: Model transformation

The rules for creating a semantic standard from a terminology model are very simple, since the only problem are spaces in designations of the terminology model. Hence, the semantic interface can be generated completely from the terminology model.

4.2 From Semantic Standard to Databases and Applications

The semantic standard contains some more characteristics, which must be filled after generating it from the terminology model. But even without adding information, one can already create a UML model, an XML schema or documentation.

After adding cardinalities, a database model can be generated. The other way is, defining mapping rules between the semantic standard and the database. Finally, a technical view or a semantic interface (ComeIn) can be defined, which allows accessing data in the database using designations as defined in the semantic standard.

Again, both, subject matter experts and IT technicians benefit a lot from the terminology model. Subject matter experts got an implementation, which corresponds 100% to their requirements. IT technicians can use the terminology model to generate system and application documentation, which can be understood by subject matter experts, who have delivered an extensive part of the text.