

A case for incorporating vague concepts in formal information modeling

Sander Bosman Theo van der Weide
sanderb@cs.kun.nl tvdw@cs.kun.nl
Computing Science Institute, University of Nijmegen
The Netherlands

Abstract

This paper gives a fundamental overview of the information modeling process in the context of requirements engineering. From this we propose an extension to conventional modeling techniques by introducing so-called vague concepts.

1 Information modeling

Information modeling is one of the main tasks during requirements engineering. Its result is a concise overview of concepts and their relations as they occur in the application domain under consideration (Universe of Discourse, UoD). This overview is called the information structure, and can be seen as a model of the UoD. During information modeling, two roles can be distinguished, referred to as *domain expert* and *system analyst*. The specification of the information structure forms both the basis for and the subject of understanding and communication between domain expert and system analyst.

In this paper we focus on natural language based modeling techniques. Such techniques aim at modeling how the UoD is communicated about; the resulting information structure is called the information grammar. Examples are NIAM (Nijssen and Halpin, 1989) and PSM (Hofstede et al., 1997). From the information grammar the concrete information structures are readily derived. Note that other information modeling techniques (such as UML (Booch et al., 1999) and ER (Chen, 1976)) focus on this concrete information structure, thereby abstracting from the linguistic packing of the information. However, conserving the linguistic packing (information grammar) allows a discussion in terms of the natural concepts in the UoD.

Our aim is to propose the introduction of *vague concepts*: concepts which are recognized as important in the modeling process, but yet have no complete and formal specification of meaning. The intention of the modeling process is to construct a precise (as opposed to vague) specification. During modeling, vague concepts will be subject to further refinement.

Before introducing these vague concepts, we first take a fundamental look at the modeling process as consisting of two main activities: 1) providing domain knowledge, and, 2) processing (modeling) the provided knowledge. For convenience, as mentioned above, we will assume these actions will be performed by two separate persons: domain expert and system analyst. The aim is to work towards a method that lets a person create a formal model by starting with vague informal descriptions, and incrementally making these more formal until a clear, consistent and precise model results. Finally, we draw some conclusions.

2 The modeling process

In this paper, we are interested in the modeling process as an interaction between domain expert and system analyst. In this section, we will first focus on general communication between human beings. Then, we will discuss how information modeling can be seen as finding the information grammar of the communication between domain expert and system analyst.

2.1 Human communication of conceptions

Consider the situation where the domain expert watches the UoD and wants to communicate information about this UoD. As this involves processes that occur inside a human's mind/brain (and are therefore largely unknown), we adopt a rather abstract cognitive model of how this works (see figure 1), based on (Falkenberg et al., 1998).

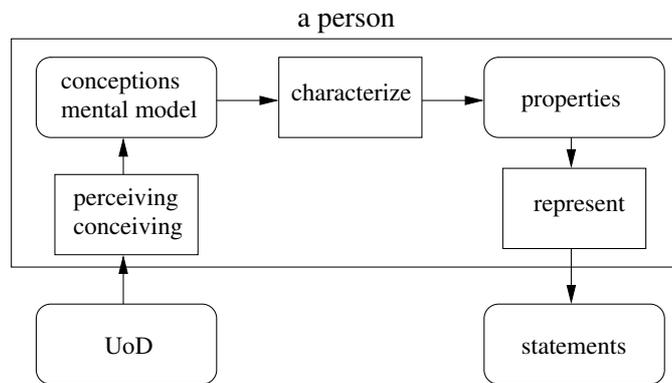


Figure 1: *Communicating conceptions.*

2.1.1 Perceiving and conceiving

We assume the Universe of Discourse to be *perceived* by the domain expert. The resulting perceptions, which initially can be regarded as raw data, are to be interpreted and elaborated on, resulting in *conceptions* as they are stored in the brain. Or, as taken from (Falkenberg et al., 1998):

Assumption 1 *Human beings are able to form conceptions in their minds, as a result of current or past perception, by means of various cognitive or intellectual processes, such as recognition, characterization, abstraction, derivation, and/or inner reflection. The collection of (relatively) stable and (sufficiently) consistent conceptions in a person's mind is called his or her knowledge.*

The collection of conceptions that describes (part of) the UoD is called the *mental model* of this UoD. Each conception can be said to model some aspect of the UoD, at some level of abstraction.

We do not assume that mental models are complete and consistent representations of the UoD. More typically, a mental model has a level of completeness and consistency *good enough* for its use.

2.1.2 Characterizing

To communicate about a mental model, we assume the domain expert derives relevant properties from conceptions in accordance with some goal (e.g., in response to a question of the system analyst).

Derivation of properties may also be influenced by the characteristics of the communication channel on which the properties will be represented. For example, in direct communication such

as face-to-face speech it makes sense to communicate 'small' properties, allowing for interruption. When writing a document, a person will focus on properties on a higher level of abstraction.

2.1.3 Representing

Properties need to be represented in some language on a medium, in order to be communicated. A represented property is called a *statement*.

A common way for humans to construct statements is to formulate them in natural language. It has been postulated that verbal communication still dominates these other styles of communication (the '*telephone heuristic*', (Nijssen and Halpin, 1989)). However, true as this may be, verbal communication is not always the most effective communication modality. This is why we allow graphical elements to be part of the communication.

Since the creation of representation takes time, it is possible for the UoD and its mental model perceived by the domain expert to change while communicating statements. Suddenly a sequence of valid statements may become invalid. The communicating person then has to 'invalidate' the invalid statements.

2.1.4 Interpreting representations

On the other end of a communication channel, someone can perceive and conceive communicated statements. This on its turn creates conceptions, in the listener's mind, being the *interpretation* of the statements. We assume interpretation follows the opposite way of characterizing and representing:

1. the person creates a sequence of properties in his mind, by perceiving and conceiving the statements.
2. the person tries to form a mental model of these properties, trying to 'make sense' of the properties. Thereby he will take the (probable) goals and context of the communicating party into account.

The two steps are not necessarily performed sequentially, as humans have a limited short term memory for storing statements.

2.2 The informal specification

In the remainder of this paper we will consider a domain expert communicating with a system analyst, who in turn creates a formal model of the UoD. This setting is shown in figure 2.

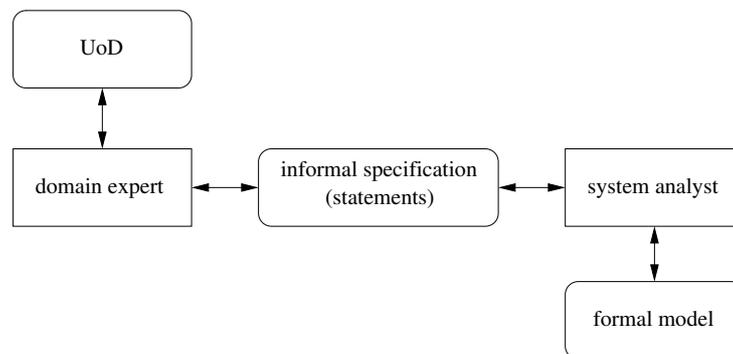


Figure 2: *Setting*

As discussed in the previous section, the domain expert perceives the UoD and has a mental model of it. The expertise of the domain expert consists of having thorough knowledge of the

UoD, and being able to communicate this model. The domain expert does not need to have the skills of providing a well abstracted model (Frederiks and van der Weide, 2003).

The domain expert communicates statements about the UoD, to be interpreted by the system analyst. The system analyst, in turn, can respond with remarks (e.g., questions). This results in a sequence of statements, referred to as the *informal specification IS*:

$$IS = [s_1, r_1, s_2, r_2, \dots]$$

where each s is a statement from the domain expert, and each r a response by the system analyst.

Goal of this communication is to create an informal specification that represents the mental model of the domain expert. We assume transitivity of the relation between UoD and mental model of the domain expert, and the relation between this mental model and the informal specification. This allows us to view the informal specification as a model of the UoD.

2.3 Modeling by communication

Modeling is the process of creating a model. There are many definitions of what a model is. For example, the definition used in (Falkenberg et al., 1998) is:

A model is a purposely abstracted, clear, precise and unambiguous conception.

We will use the term in the following way:

Definition 2.1 (Model)

1. *A model is a description of the UoD seen from some perspective. The perspective is defined by the model's goal: it defines the aspects of the UoD it should describe and the abstraction level.*
2. *A model can be abstract or physical. A mental model can be regarded abstract, while a description on paper is a physical model.*
3. *A model of some UoD can be used as substitute for that UoD, allowing to derive propositions valid in that UoD (perspective), represented as statements via the representation mechanism encapsulated in the model.*

Using a model as substitute for a UoD is the main reason for creating a model, as the model provides more insight in the UoD. It allows the creation of a 'shadow' UoD, which can be questioned and examined more efficiently than the 'real' UoD.

Example 1 Consider an organization that maintains a compact disc library (e.g., a hospital that has a music collection for their internal radio programs). To facilitate searching for CDs on keyword, the organization decides to construct a music database. For the purpose of searching CDs on keyword, the database is a model that acts as a substitute for the physical CD collection.

2.3.1 Information language

The goal of the system analyst is to create a complete and consistent formal model, based only on the communication between her and a domain expert. The system analyst is not assumed to have direct knowledge of the UoD, but her expertise is to make a well abstracted and complete formal model from the informal specification (Frederiks and van der Weide, 2003).

Let the information language be the set of possible statements the domain expert can give about the UoD, which are relevant for his perspective on the UoD. Now we can describe the modeling goal for the system analyst as follows:

The creation of a formal model by the system analyst is equivalent to the finding of the information language. The information language has been found when the system analyst can produce all statements that the domain expert could have communicated.

To be able to talk about intermediate stages in the modeling process, we relax the necessary properties of a formal model. Modeling starts with an empty formal model. As statements are communicated, the intermediate formal model grows towards the final formal model.

The formal model is finished when it can produce the information language. Until then, some required statement may not be generated by the model. Alternatively, a generated statement may not be interpretable by the domain expert.

2.3.2 Information grammar

An extensional model is an explicit listing of all the statements in an information language. The informal specification may be seen as an *incomplete* extensional model.

An extensional model has some disadvantages:

- It takes a long (and maybe infinite) time to communicate all statements in the information language. In practice, listing is infeasible for non-trivial UoDs.
- When the UoD changes, or the conceptions of this UoD, changes in the information language must be communicated. It may be a difficult task to determine which old statements are to be replaced by new statements.

For these reasons, we limit formal models to be *intensional* models, containing the *structure* of the information language. This structure is called the *information grammar*. From the information grammar, all statements in the information language can be generated. The information grammar does not have the disadvantages of the extensional model: it is much more compact, and may even describe infinite information languages.

The task of the system analyst can now be described in further detail:

The task of the system analyst is to create an intensional formal model from the informal specification. This involves finding structure in the informal specification, as well as obtaining or inducing additional information that is not part of the informal specification.

The need to obtain or induce additional information is a direct result of the assumption that the informal specification may be an *incomplete* extensional model. Although induction may be performed by the system analyst, we assume information about the structure of the information language can be obtained from the domain expert:

Assumption 2 A domain expert can provide statements about the structure of the information language he uses.

Note that this does not imply the domain expert can *directly* give the complete structure of the information language. Typically, the domain expert will give 'pieces of the puzzle' that are analyzed and combined by the system analyst.

We can now distinguish two types of statements communicated by the domain expert:

1. *Example statements*, elements of the information language.
2. *Structural statements*, which specify their structure.

Example 2 Consider the 'multiplication' domain. The following informal specification may be given by a domain expert:

```
s1: 5*3=15
s2: 8*11=88
s3: The grammar of these sentences reads:
    S := INT "*" INT "=" INT
    INT := [0-9]+
s4: the 3rd integer is the multiplication of the first two
s5: numbers range from [1..100]
```

The first two statements are example statements, while the others are structural statements. Both types of statements are accepted by the system analyst as input for creating a formal model. Note that the system analyst will have to obtain or induce more information than is available in the informal specification, in order to create a working calculator.

3 Vague concepts

The communication pattern between domain expert and analyst is the way in which the dialogue between them takes place. Typically, the domain expert provides details about the UoD, whereas the system analyst asks questions in order to trigger the domain expert to provide new or revised information.

The communicative behavior of the system analyst is determined by the task to construct a complete and consistent formal model (Veldhuijzen van Zanten et al., 2003). The system analyst may exhibit the following two extreme types of behavior: awaiting and strict.

3.1 Awaiting behavior

An awaiting system analyst waits for the domain expert to produce an initial description of the UoD. The system analyst will interpret the description and create a formal model from it.

This behavior has several disadvantages:

- The domain expert typically is not able to provide a complete description of the UoD without feedback and elaboration.
- Questions by the system analyst can be posed only after the description is finished. When answering a question causes reconsidering the view on the UoD, the rest of the description may be useless.

In short, there is a lack of interaction and direct feedback.

3.2 Strict behavior

When displaying strict behavior, the system analyst wants to be able to interpret and understand a statement as complete as possible directly after it has been communicated. This implies:

- The domain expert has to explain the strict syntactical structure of the sentence.
- The domain expert must be able to explain how the concepts are related within the sentence, and how they are related to the current formal model maintained by the system analyst.
- The domain expert must not introduce inconsistencies, as this violates the consistency of the formal model.

A new sentence has to fit nicely in the current formal model, otherwise the system analyst will try to revise the formal model immediately, or has to refuse to incorporate the statement.

3.3 Towards allowing vague concepts

It is preferable in most cases to have a behavior that is somewhere in between the two given extremes. This is what people generally seem to do in practice: when the meaning of a sentence is not directly clear, and the sentence seems to be non crucial, we wait a little hoping that later statements will provide clues about how to interpret this unclear sentence. If this takes too long, or the misunderstanding becomes too crucial, we ask questions for further specification, hoping to get enough clues to proceed (Hoppenbrouwers, 2003, page 64).

We introduce *vague concepts* as a means for obtaining the behavior sketched above. Vague concepts are concepts or concept structures that are probably important for the final formal model,

but for some reason do not fit into the current formal model. They have to be remembered, and opportunities have to be awaited or created which allow the concepts to become part of the final formal model.

Example 3 Suppose a domain expert gives the following statement:

```
s1: John lives in Madrid
```

Assuming the word structure is recognized, the statement is accepted as instance of a yet anonymous object type. The dialog continues:

```
r1: accept
s2: Mary lives in Madrid
```

Statement s_2 can not be generated from the existing information grammar. However, s_1 and s_2 may be generalized into

```
{John | Mary} lives in Madrid
```

suggesting the introduction of a special object type for John and Mary.

```
r2: accept
r3: a bicycle is a transportation means
```

Suppose our modeling technique required information structures to be completely connected, then there is no way to integrate this new statement. Displaying strict behavior, the system analyst has to reject the sentence, and the domain expert will have to offer the sentence s_3 at a later, more adequate moment.

```
r3: reject
```

If vague concepts would be allowed, the system analyst would not reject statement s_3 . Instead, he would accept it, waiting for (or creating new) opportunities to incorporate the statement into the formal model.

4 Conclusion and further research

This paper discussed formal information modeling, based on analysis of the communication between domain expert and system analyst. The result of this information modeling is a formal model that can be said to be a model of the UoD with a certain level of completeness and consistency. We argued that in order to obtain desirable communication behavior of the system analyst, we need to deal with vague concepts.

In our future research we will try to further develop a theory concerning vague concepts in formal information modeling, as well as describe ways in which vague concepts can incrementally be made precise as part of the final formal model.

References

- Booch, G., Rumbaugh, J., and Jacobson, I. (1999). *The Unified Modelling Language Used Guide*. Addison-Wesley, Reading, Massachusetts.
- Chen, P. (1976). The entity-relationship model: Toward a unified view of data. *ACM Transactions on Database Systems*, 1(1):9–36.
- Falkenberg, E., Hesse, W., Lindgreen, P., Nilsson, B., Oei, J., Rolland, C., Stamper, R., Van Assche, F., Verrijn-Stuart, A., and Vos, K., editors (1998). *A Framework of Information Systems Concepts*. IFIP WG 8.1 Task Group FRISCO.

-
- Frederiks, P. and van der Weide, T. (2003). Information modeling: the process and the required competencies of its participants. Technical report, Department of Information Systems, University of Nijmegen. submitted.
- Hofstede, A. t., Lippe, E., and Weide, T. v. d. (1997). Applications of a Categorical Framework for Conceptual Data Modeling. *Acta Informatica*, 34(12):927–963.
- Hoppenbrouwers, S. (2003). *Freezing Language; Conceptualisation Processes across ICT-Supported Organisations*. PhD thesis, University of Nijmegen.
- Nijssen, G. and Halpin, T. (1989). *Conceptual Schema and Relational Database Design: a fact oriented approach*. Prentice-Hall, Sydney, Australia.
- Veldhuijzen van Zanten, G., Hoppenbrouwers, S., and Proper, H. (2003). System development as a rational communicative process. In *Proceedings of the 7th World Multiconference on Systemics, Cybernetics and Informatics*, volume XVI, pages 126–130, Orlando, Florida.