Cognitive Requirements for Natural Language Based Conceptual Modeling

P.J.M. Frederiks, Th.P. van der Weide

Department of Information Systems, University of Nijmegen
Toernooiveld 1, NL-6525 ED Nijmegen, The Netherlands
{paulf,tvdw}@cs.kun.nl

Published as: P.J.M. Frederiks and Th.P. van der Weide. Cognitive Requirements for Natural Language Based Conceptual Modeling. Technical Report CSI-R9610, Computing Science Institute, University of Nijmegen, Nijmegen, The Netherlands, June 1996.

Abstract

In this paper we discuss the consequences of a natural language based modeling process for those who are involved in this process, i.e. domain experts and system analysts. For both domain experts and system analysts the cognitive requirements in a natural language based conceptual modeling process are presented as axiom-like requirements.

Keywords: cognitive requirements, natural language based conceptual modeling, natural language

Classification: 68U99 *(AMS-1991)*, H.1.2 *(CR-1991)*

1 Introduction

Many conceptual modeling methods have as point of departure a description in natural language (NL) of the world to be modeled (the so-called *universe of discourse*, or UoD for short). This description is referred to as the *initial specification*, and is provided by so-called *domain experts*. Ideally, this initial specification is a precise specification from which *system analysts* can derive the required information system.

However in practice this is not the case. Using an initial specification as a point of departure for conceptual modeling requires a number of skills for both domain experts and system analysts.

In this paper these skills are investigated and presented as *base axioms* ([17]) in section 3. In order to be able to appreciate these base axioms the pros and cons of using natural language in a conceptual modeling process are presented in section 2. In section 4 it is shown that the base axioms intercept the drawbacks of natural language as a starting point for conceptual modeling. The conclusions are presented in section 5.

2 NL-Based Modeling

In section 2.1, the usage of natural language for conceptual modeling is looked at in a wider context. Furthermore, the major advantages and disadvantages of using natural language for conceptual modeling are discussed in section 2.2. Finally, section 2.3 gives a rough sketch of the process of natural language based conceptual modeling.

2.1 Background

Using natural language for problem specification is not new in the field of computer science. Already in the early seventies syntactic-oriented programming methods, like step-wise refinement, were introduced, see e.g. [7]. The step-wise refinement method verbalizes a problem top-down using refinements and simple control structures like `IF THEN ELSE FI`.

Also a natural language approach to conceptual modeling is not new. The conceptual data modeling techniques (Extended)ER ([2]), NIAM ([18]) and PSM ([14]), and the object-oriented methods SACIS as described in [12] and the KISS method ([15]) are based on such an approach. For these techniques and methods the goal of the modeling process is to derive the grammar that governs the communication within the UoD, the so-called *information grammar* ([18, 10]). The advantage of this approach is that it enables the

system in a language close to natural language.

The information grammar is usually depicted as an information structure diagram. The verbalization of the elements of this diagram forms the base for the terminal symbols for this grammar. A major advantage of the use of a natural language specifications as a starting point for the modeling process is that (intermediate) results can be validated by the domain expert readily, as each intermediate result corresponds to a (partial) grammar for the language spoken in the UoD ([9]).

For more information about using natural language for conceptual modeling the reader is referred to [3, 4, 8, 13, 16].

2.2 Pros and cons of using NL

Natural language has the potential to be a precise specification language *provided* it is used well. But there are not many people who can use natural language in a consistent, non-verbose, expressed on a same level of abstraction, complete, and unambiguous way.

Still, as stated in [19], natural language is the vehicle of our thoughts and their communication. Since good communication between system analyst and domain expert is essential for obtaining the intended information system, the communication between these two partners should be in a common language: natural language. An additional advantage of using natural language for informal specifications is that it saves a translation between the initial specification and informal specification as we assumed the initial specification to be written in natural language. Informal specifications may also give hints on the way in which a user wants to communicate with the information system. A formal specification can never capture the *pragmatics* of a system. As a final argument, the formal specification may very well be paraphrased in natural language which increases the possibility for domain experts to validate the formal specification ([6]). [5] gives more arguments for paraphrasing a conceptual model to natural language.

2.3 Conceptual modeling process

Figure 1 shows a simplified view on the process of natural language based conceptual modeling. In order to initialize the modeling process the domain expert must provide the system analyst an initial natural language specification. Therefore the so-called *principle of universal linguistic expressibility* ([1]):

> *made explicit in a verbal way.*

which may be concretized by the *telephone heuristic* ([18]):

> *Explain your observations tot a non-expert via a telephone.*

is a presupposition for this modeling process. The principle of universal linguistic expressibility excludes common sensorial experiences, i.e. common sense reality. The domain expert and system analyst have no other experience in common than that which can be verbally expressed. The telephone heuristic provides the domain expert a way to obtain an informal specification.

The process of obtaining a initial natural language specification is called *elicitation*. The primary task of the system analyst is to map a sample sentence of the initial natural language specification on concepts of a particular conceptual modeling technique. This is referred to in figure 1 by the arrow labeled *modeling*. The conceptual model on its turn is translated to natural language sentences in order to be validated by the domain expert. This translation is called *paraphrasing*.

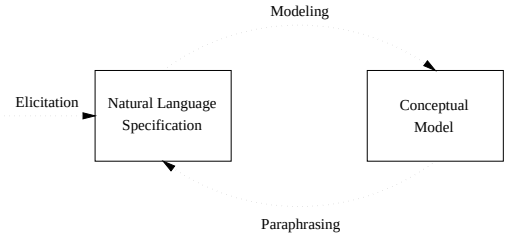

Figure 1: Natural language based conceptual modeling

This way of working is most effective when performed as an iterative process as it is impracticable to elicit a complete and correct initial natural language specification at once. The base axioms in the next section focus on a justification for this way of working, showing that it maximizes the pros and minimizes the cons of using natural language as a specification language. We argue that the iterative process eventually leads to the intended conceptual model.

3 Base Axioms

The rationale behind natural language based conceptual modeling is a scheme of base axioms describing

analysts. For more information see [17]. Roughly, a domain expert can be characterized as someone with (1) superior detail-knowledge and (2) minor capabilities for abstraction. The characterization of a system analyst is direct opposite.

An example of a base axiom is the *completeness base axiom*:

> *Domain experts can provide a complete set of significant sample sentences.*

As trivial as this axiom might seem, its impact is rather high: the axiom is the foundation for correctness of the information system, and provides insight in the requirements for those who communicate the domain to the system analyst. As the completeness base axiom is in practice a too strong requirement for a domain expert this axiom is weakened into the *provision base axiom*:

[D1] *Domain experts can provide any number of significant sample sentences.*

As this base axiom does not state that a domain expert can provide a *complete* set of significant examples by a single request from the system analyst, some more base axioms are necessary that describe aspects of the communication between domain experts and system analysts.

A prerequisite for conceptual modeling is that sentences are elementary, i.e. not splitable without loss of information. As a system analyst is not assumed to be familiar with the semantics of sample sentences, it is up to the domain expert to judge about splitting. This is expressed by the *splitting base axiom*:

[D2] *Domain experts can split sample sentences into elementary sentences.*

During the analysis process, the information grammar is constructed in a number of steps. After each step, a provisional information grammar is obtained, which can be used in the subsequent steps to communicate with domain experts. First a description of the model so far can be presented to the domain experts for validation. In the second place, the system analyst may confront the domain expert with a sample state of the UoD for validation. The goal of the system analyst might be to detect a specific constraint, or to explore a subtype hierarchy. This is based on the *validation base axiom*:

[D3] *Domain experts can validate a description of their application domain.*

check completeness by suggesting new sample sentence to the domain expert. This is based on the *significance base axiom*:

[D4] *Domain experts can judge the significance of a sample sentence.*

Besides studying the cognitive identity of domain experts it is necessary to investigate the cognitive identity of system analysts.

A first axiom which is applicable for system analysts is the so-called *consistency base axiom:*

[A1] *System analysts can validate a set of sample sentences for consistency.*

A system analyst is expected to make abstractions from detailed information provided by domain experts. By having more instances of some sort of sentence, the system analyst will get an impression of the underlying *sentence structure* and its appearances. The sentence structures all together form the structure of the information grammar. This is addressed in the *abstraction base axiom:*

[A2] *System analysts can abstract sentence structures from a set of sample sentences.*

As abstraction of sentences is based on the existence of a number of concrete sample sentences the system analyst must be able to generate new sample sentences which are validated by the domain experts, i.e. the *generation base axiom*:

[A3] *System analysts can generate new sample sentences.*

Finally, a system analyst must be able to match sentence structures found with the abstraction base axiom with the concepts of a particular conceptual modeling technique. This is expressed by the *modeling base axiom*:

[A4] *System analysts can match sentence structures with concepts of a modeling technique.*

4 Impact of Base Axioms

In section 2, a number of disadvantages of using natural language for conceptual modeling were encountered. Summarizing, natural language is hard to use:

2. in a *non-verbose* way,
3. in an *unambiguous* way,
4. in a *consistent* way, and
5. on a same level of *abstraction*.

In this section we will make it plausible that the base axioms for domain experts and systems analysts, introduced in section 3, can reduce the impact of the above mentioned disadvantages of using natural language.

Base axiom D1 overcomes the problem with respect to the completeness of natural language specifications. The axiom states that domain experts can provide any number of significant sample sentences. Assumed that each UoD can be described by a finite number of structurally different sample sentences, the probability of missing some sentence structure will decrease with each sample sentence generated. The process of providing sample sentences by domain experts is triggered, controlled and guided by the system analyst, as stated in base axiom A3.

Specifications in natural language tend to be verbose. Complex (verbose) sentences will be offered to the domain expert for splitting (base axiom D2) and relevance judgment (base axiom D4). A natural language specification may also be verbose by a large number of sample sentences. This is anticipated by the skill of system analysts to abstract from superfluous sample sentences (see base axiom A2).

An often raised problem of natural language is ambiguity, i.e. sentences with the same sentence structure having a different meaning. In order to detect ambiguities, the system analyst may offer the domain expert alternative formulations of sentences for validation (base axioms A3 and D3). On the other hand, a system analyst may also elicit a domain expert for further explanation by asking to provide alternative formulations or more sample sentences with respect to the original ambiguous sample sentence (base axiom D1).

In base axiom A1 it is stated that a system analyst is equipped with the ability to verify a natural language specification for consistency. Just like the entire conceptual modeling process, consistency checking of natural language specifications has an iterative character. Furthermore, consistency checking requires interaction with the domain expert as a system analyst may have either a request for more sample sentences (base axiom D1), or a request to validate new sample sentences (base axioms A3, D3 and D4).

ten written on several levels of abstraction, for example

> *The Rolling Stones record the song Paint it Black.*

As a system analyst has no detail knowledge, and thus also no knowledge at the instance level, a prerequisite for abstraction is typing of instances. Some instances will be typed by the domain expert (the song *Paint it Black*) while others are untyped (*The Rolling Stones*). This may be resolved by applying a typing mechanism to untyped instances (see [11]). Typed sentences can be presented to the domain expert for validation (base axiom D3).

5 Conclusions

This paper has described advantages and disadvantages of using natural language for conceptual modeling. A number of base axioms have been presented, for both domain experts and system analysts, which describe skills domain experts and system analysts should have in order to perform this natural language based way of conceptual modeling. It has been made plausible that the disadvantages of using natural language in the conceptual modeling process are weakened by the base axioms.

Further research is required in order to investigate the consequences of this approach for education of both domain experts and system analysts.

Acknowledgment

The authors would like to thank Eduard Hoenkamp and Kees Koster for their constructive criticism and suggestions on earlier versions of this paper.

References

[1] P.W. Adriaans. *Language Learning from a Categorial Perspective.* PhD thesis, University of Amsterdam, Amsterdam, The Netherlands, 1992.

[2] E. Buchholz, H. Cyriaks, A. Düsterhöft, H. Mehlan, and B. Thalheim. Applying a Natural Language Dialogue Tool for Designing Databases. In *Proceedings of the First Workshop on Applications of Natural Language to Databases (NLDB'95)*, pages 119–133, Versailles, France, June 1995.

of Linguistics on Conceptual Models: Consistency and Understandability. In *Proceedings of the First Workshop on Applications of Natural Language to Databases (NLDB'95)*, pages 183–197, Versailles, France, June 1995.

[4] M.A. Collignon and Th.P. van der Weide. An Information Analysis Method Based on PSM. In G.M. Nijssen, editor, *Proceedings of NIAM-ISDM*. NIAM-GUIDE, September 1993. pp. 22.

[5] H. Dalianis. A method for validating a conceptual model by natural language discourse generation. In P. Loucopoulos, editor, *Proceedings of the Fourth International Conference CAiSE'92 on Advanced Information Systems Engineering*, volume 593 of *Lecture Notes in Computer Science*, pages 425–444, Manchester, United Kingdom, 1992. Springer-Verlag.

[6] C.F. Derksen, P.J.M. Frederiks, and Th.P. van der Weide. Paraphrasing as a Technique to Support Object-Oriented Analysis. Technical Report CSI-R9603, Computing Science Institute, University of Nijmegen, Nijmegen, The Netherlands, January 1996. (accepted for Natural Language to Databases '96).

[7] E. W. Dijkstra. *A Discipline of Programming*. Prentice-Hall, Englewood Cliffs, New Jersey, 1976.

[8] L. Dunn and M. Orlowska. A Natural Language Interpreter for the Construction of Conceptual Schemas. In B. Steinholz, A. Sølvberg, and L. Bergman, editors, *Proceedings of the Second Nordic Conference CAiSE'90 on Advanced Information Systems Engineering*, volume 436 of *Lecture Notes in Computer Science*, pages 175–194, Stockholm, Sweden, 1990. Springer-Verlag.

[9] P.J.M. Frederiks, C.H.A. Koster, and Th.P. van der Weide. Validation of Object-Oriented Analysis Models using Informal Language. Technical Report CSI-R9609, Computing Science Institute, University of Nijmegen, Nijmegen, The Netherlands, May 1996.

[10] P.J.M. Frederiks and Th.P. van der Weide. From a File-Oriented View to an Object-Oriented View. Technical Report CSI-R9601, Computing Science Institute, University of Nijmegen, Nijmegen, The Netherlands, January 1996.

damentals for Object-Oriented Analysis. Technical Report, Computing Science Institute, University of Nijmegen, Nijmegen, The Netherlands, September 1996. (in preparation).

[12] I. Graham. *Object-oriented Methods*. Addison-Wesley, Reading, Massachusetts, 1994.

[13] A.H.M. ter Hofstede, H.A. Proper, and Th.P. van der Weide. Grammar Based Information Modelling. Technical Report CSI-R9414, Submitted for publication, Computing Science Institute, University of Nijmegen, Nijmegen, The Netherlands, October 1994.

[14] A.H.M. ter Hofstede and Th.P. van der Weide. Expressiveness in conceptual data modelling. *Data & Knowledge Engineering*, 10(1):65–100, February 1993.

[15] G. Kristen. *Object Orientation, the KISS Method: From Information Architecture to Information System*. Addison-Wesley, Reading, Massachusetts, 1994.

[16] L. Mich and R. Garigliano. A Linguistic Approach to the Development of Object Oriented Systems using the NL System LOLITA. In E. Bertino and S. Urban, editors, *Proceedings of the International Symposium, ISOOMS '94: Object-Oriented Methodologies and Systems*, volume 858 of *Lecture Notes in Computer Science*, pages 371–386, Palermo, Italy, September 1994. Springer-Verlag.

[17] G.M. Nijssen. An Axiom and Architecture for Information Systems. In E. D. Falkenberg and P. Lindgreen, editors, *Information System Concepts: An In-depth Analysis*, pages 157–175. North-Holland/IFIP, Amsterdam, The Netherlands, 1989.

[18] G.M. Nijssen and T.A. Halpin. *Conceptual Schema and Relational Database Design: a fact oriented approach*. Prentice-Hall, Sydney, Australia, 1989.

[19] W. Quine. *Word and object – Studies in communication*. The Technology Press of the Massachusetts Institute of Technology, Cambridge, Massachusetts, 1960.