Advances in FCO-IM (1); Classification and Qualification: Disconnected and Overlapping Object Type Expressions

Jan Pieter Zwart, Guido Bakema

janpieter.zwart@han.nl, guido.bakema@han.nl
Research and Competence Group Data Architectures & Metadata Management
Academy for Communication and Information technology
HAN University of Applied Science
Ruitenberglaan 26, 6826 CC Arnhem, The Netherlands
Phone: +31-26-3658271. Fax: +31-26-3658126

Abstract. Fully Communication Oriented Information Modeling (FCO-IM) completely and exclusively models the *communication* about a certain universe of discourse (UoD). The essence of building an FCO-IM information model is classifying and qualifying *fact expressions* (user communication in the form of verbalizations of facts by domain experts). The result is *metadata*, stored as an FCO-IM *information grammar* (IG). A visual representation of this IG is an *information grammar diagram* (IGD), but this is just eye candy. We show how disconnected and overlapping object expressions remove a restriction we placed earlier on the form of these fact expressions. Since the communication reflects the mental concepts users have of their UoD, information models depend on these concepts. This leads us to tentatively present a classification of different kinds of semantic equivalence, and to conclude that metamodeling at its best is concerned with finding the most elegant concepts to describe a given UoD.

1 Introduction

Fully Communication Oriented Information Modeling (FCO-IM) [1] is a member of the family of fact oriented modeling (FOM) techniques, as are NIAM, ORM, PSM and others [1, 7, 10, 11]. FCO-IM models the *communication* about a certain UoD completely and exclusively; i.e.: it models the communication, the whole communication, and nothing but the communication. There are two main reasons why we boast it to be '*fully* communication oriented'. Both reasons are briefly explained below, and illustrated in more detail in section 2.

The first reason is that FCO-IM is the only FOM family member to date that completely incorporates the actual verbalizations of facts (*fact expressions*) by domain experts in an information model, in the form of *fact type expressions* (FTEs), which are *predicates* over *object type expressions* (OTEs, which are themselves nominalizations of predicates over object type expressions). From the tuples in the population and these FTEs, the original fact expressions can be regenerated verbatim at any time from an FCO-IM model. An FCO-IM model therefore contains the complete set of fact stating sentence types describing a certain Universe of Discourse (UoD), thereby capturing the 'soft semantics' – the *meaning* of the facts – as well as the 'hard semantics' – the fact types as a set of roles played by object types, and the constraints.

The second reason is that FCO-IM uses a very simple and small set of meta-concepts. The FCO-IM meta-model can be so concise because FCO-IM uses only one means of building a complex integrated information model from a set of elementary fact expressions: the linguistic concept of nominalization [5]. *All object types in FCO-IM are nominalized fact types.* The FCO-IM meta-model is therefore free from the burden to represent object types separately, representing them as parts of sentence types instead. In other words: object types are not modeled independently from the communication, but as inextricable parts of the communication itself; FCO-IM never leaves the communication domain.

In section 2 we illustrate the points made above, by showing that FCO-IM modeling constructs completely follow from the process of *classifying and qualifying* fact expressions. We have shown in [1] that this is true for *all* modeling constructs, including specialization (subtyping), generalization and recursive data structures.

Our book on FCO-IM [1, p.89, remark 5] places a restriction on fact expressions, namely that a set of words that identifies an object (an *object expression*) must be a contiguous (unbroken, connected) sentence part. In section 3, we present a way to remove this restriction (already hinted at in our book), by allowing an object expression to be *disconnected*; i.e.: two or more non-connected sentence parts together identify an object. In this paper we extend this approach by allowing parts of disconnected object expressions to *overlap* (coincide). These disconnected and overlapping object expressions enable us to use all FCO-IM modeling constructs without changing the actual fact expressions of the domain experts, as befits a 'fully communication oriented' information modeling technique.

Because fact expressions also reflect the mental model domain experts have of a UoD (which object types they perceive, for instance), we will explore the issue of *semantic equivalence* in section 4: what if the same UoD is perceived in different ways (i.e.: different users employ different sets of object types for the same UoD)? We conclude that the ultimate challenge of (meta)modeling is to find the most elegant set of concepts for a given UoD.

In this paper, we will follow the usual FCO-IM procedure of concretely illustrating all matters discussed, using an example Harm van der Lek encountered in a project to model the complete infrastructure of the Dutch Railways [12], the company exploiting the most intensively used train system in the world.

2 Classifying and Qualifying

The process of building an FCO-IM information model from a set of elementary fact expressions consists of two main steps; 1: analysis of these expressions by classification and qualification (*ClaQua* for short), and 2: addition of constraints. Both steps are discussed in detail in the FCO-IM book [1]. In this paper we will only indicate the main points of ClaQua.

We wish to emphasize, that in FCO-IM the ClaQua process is the very essence of the whole modeling process, not just a convenient starting point. Its result is a set of

UoD-related metadata in the form of an *information grammar* (IG), usually represented graphically in an *information grammar diagram* (IGD), but such a diagram is only a visual representation of the underlying contents of the FCO-IM repository, 'eye-candy' if you like. Therefore, an FCO-IM modeling tool such as CaseTalk [4] cannot be a trivial 'drag and drop' drawing tool: it is a sophisticated interface upon the FCO-IM metadata repository, which contains the IG, that supports the information analyst in carrying out the ClaQua process.

Example of ClaQua

Consider the ClaQua of fact expressions FE1 – FE6 below, which express facts about train stations (each station has a unique name), platforms in these stations (each platform has a unique number *per station*) and services available on one or more of these platforms.

FE1: "There is a platform no. 7 in Arnhem station."
FE2: "A bookstall is available on platform 7 in Arnhem station."
FE3: "A payphone is available on platform 7 in Arnhem station."
FE4: "There is a platform no. 3 in Arnhem station."
FE5: "There is a platform no. 3 in Nijmegen station."
FE6: "A restroom is available on platform 3 in Nijmegen station."

Classification (i.e.: assigning things to classes) and qualification (i.e.: giving meaningful names to these classes) are carried out in two stages. In the *first stage* of ClaQua, fact expressions of the same type are grouped into *fact types* and each fact type is given a meaningful name. In the *second stage* of ClaQua, we classify parts of the fact expressions either as an *object expression* (OE, a nominative group that names (identifies) an object in the UoD), or as a *label* (a name, number, code or other lexical reference that does not by itself identify anything). We qualify each underlined part by giving a meaningful name to the corresponding *object type* (the class an OE belongs to) or *label type* (the class a label belongs to).

```
Platform:
FE1  "There is a platform no. 7 in Arnhem station."
FE4  "  "    "  "     "     "   3 "  Arnhem     "  ."
FE5  "  "    "  "     "     "   3 "  Nijmegen   "  ."
                 platform number      Station:O1

                                    'Arnhem station'
                                    'Nijmegen   "  '
                                  station name

Service on Platform:
FE2  "A bookstall is available on platform 7 in Arnhem station."
FE3  "A payphone  "     "      "      "    7 "  Arnhem     "  ."
FE6  "" restroom  "     "      "      "    3 "  Nijmegen   "  ."
        Service:O2                         Platform:O3

       'bookstall'                'platform 7 in Arnhem station'
       'payphone '                '    "   7 "  Arnhem     "  '
       'restroom '                '    "   3 "  Nijmegen   "  '
      service name              platform number    Station:O1(match)
```

Fig. 1. Classification and qualification of several fact expressions.

In figure 1 above, the results of both stages are shown. In the first stage, fact expressions FE1, FE4 and FE5 were assigned to fact type Platform, and FE2, FE3 and FE6 to fact type Service on Platform. The ditto marks (") indicate identical and unchanging sentence parts. A double underlining means we classify the sentence part as an object expression, and a single underlining means we classify the sentence part as a label. The names of the corresponding object types and label types are written below the underlining. The word 'match' in the analysis of fact type Service on Platform indicates that object type Station has already been defined (in the analysis of fact type Platform) and is therefore already known in the FCO-IM repository. Detailed rules and guidelines for how to carry out the ClaQua process are given in [1].

This example illustrates that ClaQua essentially adds grammatical information to the FCO-IM information grammar (IG), which is stored in an FCO-IM repository, such as names for fact types, object types and label types, predicates (the sentence parts without underlining) and so on. For an example of a simplified version of such a repository, see [2]. For the present full version, see [4].

Information Grammar Diagram (IGD)

An FCO-IM information grammar diagram (IGD) is just a human-friendly view on the contents of the FCO-IM information grammar (IG, the repository). In practice, diagrams are also used as a graphical interface for extending the population, adding constraints and editing, but the real changes are made only in the repository. The diagram in figure 2 graphically shows the results of the ClaQua process given textually in figure 1. Fact types consist of one or more numbered rectangles (called *roles*), solid circles depict object types and dotted circles represent label types. For further details see [1]. Uniqueness constraints [1, section 3.2] (arrows) and totality constraints 9 [1, section 3.4] (fat dots) have been added as well. These constraints are stored as meta-facts in the repository as well.

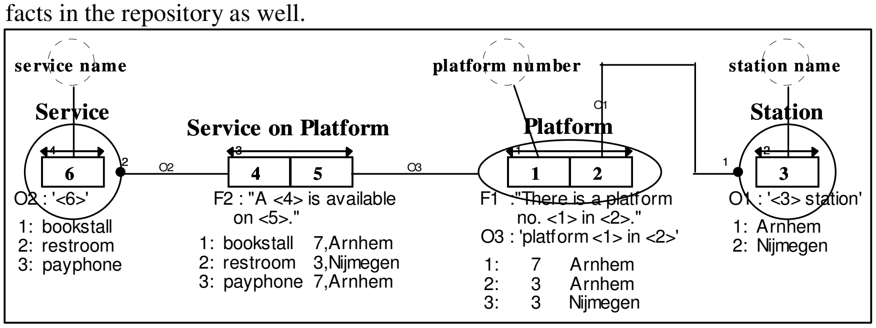

Fig. 2. FCO-IM diagram of the result of the ClaQua process, with constraints added.

Note that the *fact type* Platform is used as an *object type* in fact type Service on Platform. In FCO-IM, *all* object types are objectified (or nominalized) versions of fact types [1, section 2.4].

However, formally FCO-IM does not model objects themselves, but only the way these objects are identified in the communication. An object expression (OE) is a nominative group, i.e.: a sentence part that names (identifies) something in the UoD. FCO-IM therefore needs no distinction between objects and references to objects, since the objects themselves are not modeled.

3 Disconnected and Overlapping Object Expressions

In our book on FCO-IM [1, p.89, remark 5], we placed a restriction on fact expressions, namely that a set of words that identifies an object must be a contiguous sentence part (one unbroken section of the expression). Here we show how this limitation can be removed. Consider the following two alternative fact expressions from the Dutch Railways UoD, with a part of their classification and qualification:

```
FE2:  "A bookstall is available on platform 7 in Arnhem station."
         Service:O2                          Platform:O3
FE2d: "In Arnhem station, a bookstall is available on platform 7."
          Platform:O3.2     Service:O2            Platform:O3.1
```

In FE2, object expression ‘platform 7 in Arnhem station’ is a contiguous sentence part, belonging to O3. Figure 2 shows an FCO-IM diagram that contains it. In FE2d, this is split in two disconnected sentence parts ‘in Arnhem station’, belonging to O3.2, and ‘platform 7’, belonging to O3.1. Figure 3 shows an FCO-IM diagram that contains this disconnected object type expression. Note that the placeholder for role 5 in the fact type expression F2 is split in two parts <5.1> and <5.2>. The sentence can be regenerated from the diagram [1, section 2.6], using the substitution rules indicated in the diagram (O3.1→5.2 and O3.2→5.1). We leave it to the reader to regenerate from figure 3 fact expressions FE3d and FE6d, analogous to FE2d and corresponding to FE3 and FE6.

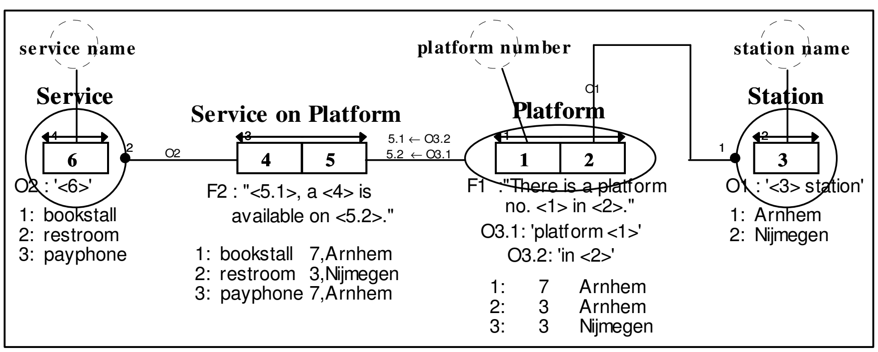

Fig. 3. FCO-IM diagram with disconnected object type expression.

Finally, consider the following fact expressions, encountered in practice during the large project to model the entire infrastructure of the Dutch Railways in 1992 [2].

FEo1: “There is a platform no.7 in Arnhem station.”
FEo2: “There is a track no.12 in Arnhem station.”
FEo3: “In Arnhem station, platform 7 is next to track 12.”

To identify a train track, both the station name and the track number are required, similar to platform identification. In a verbalization expressing which platform is next to which track however, the station name is mentioned only once, with tacit understanding that the same station name applies to both objects. We can model this in FCO-IM using disconnected object type expressions that *overlap*:

```
FEo3:     "In Arnhem station, platform 7 is next to track 12."
            Platform:O3.2    Platform:O3.1
                                                                   "
               Track:O4.2                                  Track:O4.1
```

Object type expression parts O3.2 and O4.2 coincide, causing O3 and O4 to overlap.

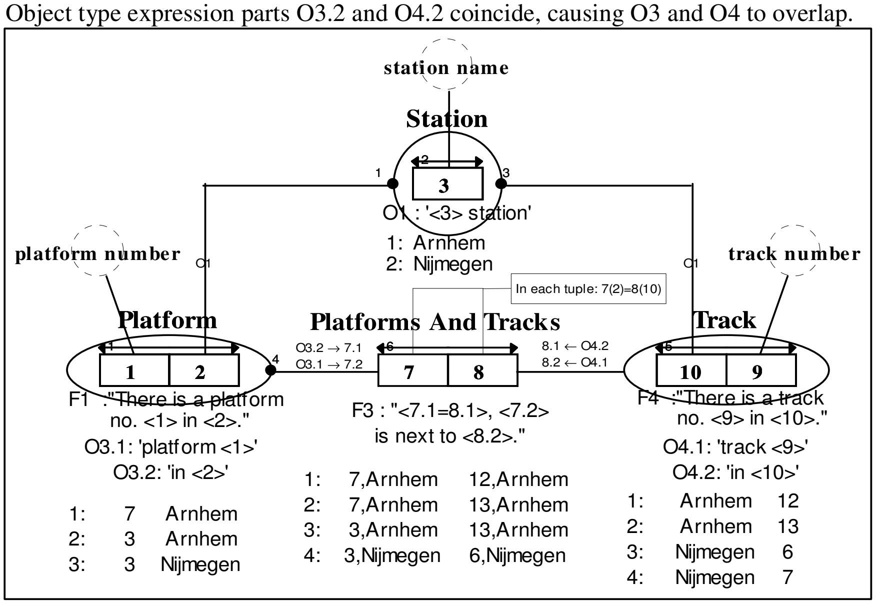

Fig. 4. FCO-IM diagram with disconnected and overlapping object type expressions.

Figure 4 shows an FCO-IM diagram that contains these disconnected and overlapping object type expressions. Note how the overlap is indicated by the role placeholders in the fact type expression: <7.1=8.1>, which together with the substitution rules (O3.2→7.1 and O4.2→8.1) enables us to regenerate FEo3 from the diagram. There are obviously many metaconstraints that must be satisfied if an FCO-IM grammar with such constructs is to be well-formed, but these will be discussed in a future paper on the changes that must be made to the FCO-IM metamodel (the FCO-IM repository) to incorporate disconnected and overlapping object type expressions.

Please note the constraint indicated in figure 4 on roles 7 and 8, which requires the two station names in every tuple to be identical. A *tuple* is one of the numbered text lines with labels written below the fact types in the diagram. In words: in every tuple, the part of the population of role 7 that comes from role 2 must be equal to the part of the population of role 8 that comes from role 10. This is an example of what we call a *strict* (i.e.: per tuple) *equality constraint* on *subroles* (subroles are parts of roles coming from other roles higher up in the nominalization tree). Subroles are also very interesting for our current work on a new metamodel for FCO-IM.

When a transformation to another data modeling technique is made (for instance to a relational schema), then the transformation algorithm will ensure that no redundant structures (for instance identical table columns) will arise. Note that there is no redundancy here at the conceptual level.

As frequently happens as soon as a new phenomenon is recognized, it seems to crop up everywhere. We encounter overlapping identification parts in fact expressions in many contexts (university teaching programs, holiday planning on a campsite, hiking trails in nature reserves, etc.).

4 Semantic Equivalence

We conclude this paper with a tentative discussion of kinds of *semantic equivalence* (SE). Two different information models are said to be semantically equivalent if (in some sense) they – correctly – describe the same UoD. We are naturally led to consider this issue because it is the user communication about a UoD that is modeled in FCO-IM, and the ClaQua process brings out object types perceived by the domain experts, as we showed in section 2. Different domain experts may regard the same UoD differently however, which may result in different models for the same UoD. Indeed, it is fundamentally impossible for any information modeling technique to model the UoD itself, we can only model the users' perception(s) of the UoD.

We distinguish between several kinds of semantic equivalence, as depicted in figure 5. Some of these kinds are well-known and well-understood (e.g. SE of the 1st and 2nd kind, see below), but we hope that this first attempt to provide a framework will help to further our understanding of this intriguing subject.

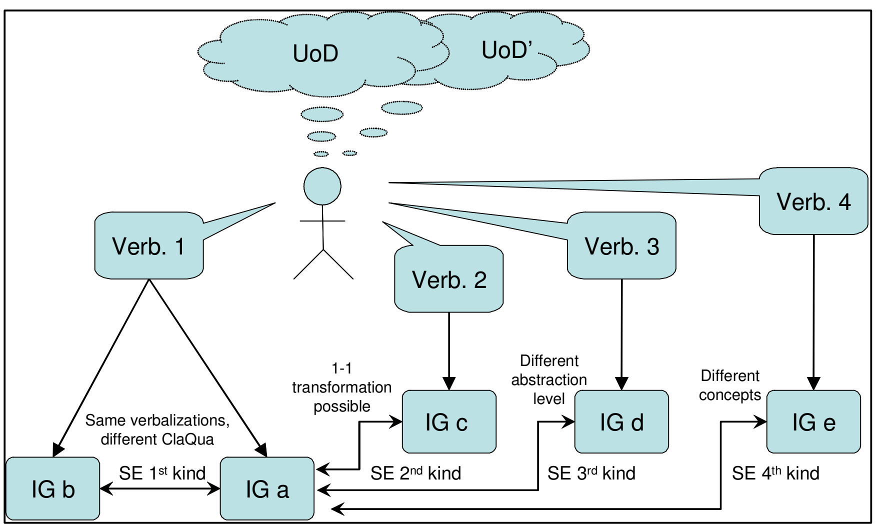

Fig. 5. Several kinds of semantic equivalence

Semantic Equivalence of the First Kind

Exactly the same facts can lead to different models in NIAM and ORM (entity type to fact type conversion, nesting and flattening, conceptual schema equivalence) [10]. This carries over into FCO-IM as well, where these differences in models result from different ways to ClaQua the *same* fact expressions. We have shown [1, section 2.7] that a difference in the *first* stage of ClaQua leads to different grammars that can be transformed into each other by a role-to-object transformation (entity type to fact type conversion), whereas a difference in the *second* stage of ClaQua leads to grammars that can be transformed into each other by a nominalization - denominalization transformation (nesting or flattening).

We consider two FCO-IM grammars IGa and IGb (see figure 5) as semantically equivalent of the first kind, if they model exactly the same fact expressions but are different because of one or both of these transformations.

Semantic Equivalence of the Second Kind

Consider the following two fact expressions, verbalizing the same fact (length of the same item) in different ways:

FE7a: "The length of item 123 is 5.44 cm."
FE7c: "The length of item 123 is 2.14 inch."

In an FCO-IM grammar IGa containing FE7a, there will be an object type Length identified by the number of centimeters, but in a grammar IGc containing FE7c, there will be a different object type Length, identified by the number of inches. These two grammars can be transformed into each other, using the additional information that 1.00 inch = 2.54 cm.

We consider two FCO-IM grammars IGa and IGc (see figure 5) as semantically equivalent of the second kind, if they model different fact expressions, but that there exists a one-to-one transformation that can transform all fact expressions of IGa into those of IGc and vice versa.

Semantic Equivalence of the Third Kind

Consider the following fact expressions, which are verbalizations of facts from the same UoD in different ways:

FE8a1: "The school is closed on 27/12/2006." FE8a2: "The school is closed on 28/12/2006." FE8a3: "The school is closed on 29/12/2006."
FE8d: "The school is closed in the period from 27/12/2006 through 29/12/2006."

fact expression FE8d uses concepts on a *different abstraction level* compared to those used by fact expressions FE8a1-FE8a3. An FCO-IM grammar IGd containing FE8d will have an object type Period which is absent in a grammar IGa containing the other fact expressions (and which cannot be created by a transformation from semantic equivalence of the first kind). Both grammars will have an object type Day. It is possible to define a transformation between the different sets of concepts, however, but not one-to-one.

We consider two FCO-IM grammars IGa and IGd (see figure 5) as semantically equivalent of the third kind, if they model different fact expressions using concepts on different abstraction levels, as described above.

Semantic Equivalence of the Fourth Kind

Finally, it is possible to make two sets of fact expressions for the same UoD, which use two quite different sets of concepts, such that a transformation between the two sets seems illusory. As a well-known example consider the replacement of Newton's by Einstein's description of gravity. We have encountered such a paradigm shift a few times in practice. For instance, Paulus Bakx [3] succeeded in making a very elegant model for an enterprise content management system for maintaining law versions for the Dutch Ministry of Justice, by inventing a novel way of looking at law articles, versions of articles, etc. The domain experts were surprised to find their UoD so beautifully formulated. This legislation case is discussed in more detail in [6].

Indeed, FCO-IM itself is a case in point: by replacing a redundant set of metaconcepts used in ERM - to a lesser extent also present in NIAM, PSM and ORM - with a more *elegant* (in the mathematical sense of the word) set of metaconcepts [8], the FCO-IM metamodel became both smaller and more powerful; for instance, it proved to be possible to easily generate ERM models, UML class diagrams and dimensional datawarehouse models from FCO-IM models [9].

It is therefore our opinion, that the challenge in information modeling and particularly in metamodeling lies in trying to find a set of concepts that captures a UoD as elegantly as possible.

We consider two FCO-IM grammars IGa and IGe (see figure 4) as semantically equivalent of the fourth kind, if they model a UoD using different sets of concepts.

Where does semantic equivalence end?

The distinctions between the different kinds of semantic equivalence described above are not sharp, and may well be just differences in degree on a more or less continuous scale. Some people may even be of the opinion that in the case of semantic equivalence of the fourth (or even the third) kind, we are really considering different UoD's, which is why we indicated a UoD′ in figure 5. We intend to explore this topic further in the future, in the context of our research in the area of modeling patterns [6].

5 Conclusion

Classification and qualification (ClaQua) is at the heart of FCO-IM information modeling, and all modeling constructs follow from how this process is carried out.

This ClaQua process also elucidates the mental concepts of the domain experts, which are reflected in the object types being determined. Consequently, the most difficult part of drawing up an information model in a FOM technique (FCO-IM, ORM, PSM, NIAM or other communication oriented modeling technique) is to formulate good fact expressions. High-level information analysts must therefore be capable of helping domain experts to verbalize their facts. We often find in practice that domain experts are not fully conscious of their own mental concepts, a well-known problem in other areas, such as Artificial Intelligence. Such an analyst should be able to deploy the different kinds of semantic equivalence discussed above in any UoD (in consultation with the domain experts, of course).

We have shown how the capability of FCO-IM to model the final verbalizations exactly as they are pronounced can be extended by introducing disconnected and overlapping object type expressions. A corollary of this is, that it will no longer be necessary to reformulate any fact expression if an IGD needs to be changed because it does not pass the nominalization test [1, section 3.3.2]. This is not only needed to stake the claim of being 'fully communication oriented', but we are sure domain experts will welcome this in practice as well, since they no longer will be asked to rephrase their expressions because of limitations in the modeling technique.

References

1 Bakema, Guido; Zwart, Jan Pieter; Lek, Harm van der: Fully Communication Oriented Information Modeling (FCO-IM), 2002. The book can be downloaded for free from http://www.casetalk.com/php/index.php?FCO-IM%20English%20Book.

2 Bakema, G.P.; Zwart, J.P.C; Lek, H. van der: Fully Communication Oriented NIAM, NIAM-ISDM 1994 Conference, Working Papers, Albuquerque, USA (1994).

3 Bakx, Paulus, personal communication, 2005.

4 BCP Software: CaseTalk, FCO-IM modeling tool, see http://www.CaseTalk.com.

5 Brown K, Encyclopedia of Language & Linguistics (2006), Elsevier, London.

6 Fazat Nur Azizah; Guido Bakema: Data Modeling Patterns Using Fully Communication Oriented Information Modeling (FCO-IM), OTM Workshops(2) 2006: 1221-1230.

7 Halpin, Terry: Information Modeling and Relational Databases, Morgan Kaufmann Publishers, ISBN-13: 978-1-55860-672-2, ISBN-10:1-55860672-6.

8 Lek, H. van der: On the structure of an Information Grammar, NIAM-ISDM 1993 Conference, Working Papers, Utrecht (1993).

9 Manoku, Elton; Bakema, Guido: Integrated Tool support for Datawarehouse Design, Journal of Conceptual Modeling, issue 34, January 2005.

10 Nijssen, G.M.; Halpin, T.A.: Conceptual Schema and Relational Database Design, Prentice Hall, 1989, ISBN 0-7248-0151-0.

11 Proper, H.A: A theory for Conceptual Modelling of Evolving Application Domains, PhD Thesis (1994), Nijmegen, Netherlands, ISBN 90-9006849-X.

12 VPT (Vervoer Per Trein (Transport By Train)) project, NS (Nederlandse Spoorwegen (Dutch Railways)), 1992.