An Experience using Context in Translation from System's Requirements to Conceptual Model

Nara Martini Bigolin[*] *and Patrick Brezillon*
LAFORIA - IBP - University of Paris VI
4, Place Jussieu - 75252 PARIS cedex 05
Tel. 33 1 44 27 70 10
Fax. 33 1 44 27 70 00
{bigolin,brezillon}@laforia.ibp.fr

Abstract

In this paper, we use context to simplify the translation from system's requirements expressed in natural language to conceptual model (Entity-Relationship Model). This translation is made by a tool called Designer Assistant. The goal of the system is to identify entities, attributes and relationships from requirements expressed in natural language (Portuguese Language in our case) of a database system. For example, the application ***post-graduation evaluation*** *has several entities as* ***note****,* ***student****,* ***discipline****, etc. The entity* ***studen****t has attributes as* ***Department****. However, in another application as* ***university's organization****,* ***Department*** *is an entity, because in this context it is a relevant information, thus the same item can be an attribute or an entity depending on the context of the application. Our system distinguishes the nature of such an attribute (contextual and contextualized respectively in the example above) in an application. The contextual information is stored in dictionaries with the words of the Portuguese Language.*

Keywords: natural language, artificial intelligence, context, database design, translation.

1 Introduction

The context has been recently the main topic of workshops (Brézillon and Abu-Hakima 1993, 1995) and symposium (Buvac 1995, Strat and Mundy 1995). However, the notion of context is not yet clearly defined in the literature and each author presents a personal definition. McCarthy in logic domain (McCarthy and Buvac 1994) introduces context as an abstract mathematical entity with properties useful in artificial intelligence. Farwell and Helmreich (1995) in natural language processing domain define context as an intelligent agent's view of the world that consists of the entire range of facts, regularities, inferences, and hypotheses. In translation system, Maier (1995) presents context as a multi-dimensional context models of a speech translation system, which has been derived from constraints given by various sub-components of the system.

In the design of a database, we define a **context** as the delimitation of a domain, that allows to restrict the possible solution-space of a problem. For example, consider the item *Department* in two applications, *post-graduation evaluation* and *university's organization*. The first application, ***post-graduation evaluation*** has several entities as *note*, *student*, *discipline*, etc. The entity *student* has attributes as *Department*. However, in another application as ***university's organization***, *Department* is an entity, which is a relevant information in this context. Thus, the same item (department) can be an attribute or an entity depending on the application context.

In our formalization, *department* is explicitly contextualized in the context of the *university's organization*, because it is a relevant information. Conversely it is let implicit in the context of a *post-graduation evaluation* and thus has the status of contextual information.

[*] Supported by CNPq, Brazil

Context has been applied in different areas such as problem solving (Giunchiglia, 1993; Edlund, Weise and Lewis, 1995) natural language (Farwell and Helmreich, 1995; IJCAI-95), database (Hian Goh, Madnick, and Siegel, 1994). For example, Zarri (1995) uses techniques associated to the context for translating natural language into a framework called NKRL (Narrative Knowledge Representation Language) technology.

Our purpose is to translate system's requirements expressed in natural language (Portuguese language) into a formal description. The formal description is presented in terms of the Entity-Relationship (E-R) Model during the process of database system design. The process, which is described here, concerns the second step among the following ones: requirement analysis, conceptual design, logic design and physical design of the database system design (Teorey and Fry, 1982).

The *requirement analysis* is the most poorly defined, difficult, and time-consuming step of the whole process. However, it is the most important because the majority of subsequent design decisions are based on this step. At this first step, the major task concerns the collecting of information content and processing of requirements from all the identified and potential users of the database. The contextual techniques can be used to delimit the domain knowledge of the application (Paton, 1993).

In the step of *conceptual design*, the system's requirements collected in the first step and expressed in natural language are translated into a formal model. The result is a *conceptual schema*. Recently, semi-automatic tools are used by the designer to bring support during this step without the notion of context (Storey, 1993). However, contextual techniques play an important role, being used to eliminate ambiguities and delimit the domain. For example, according to application *post-graduation evaluation* we can determine that *notes* and *disciplines* belong to same context (post-graduation evaluation) by using contextual rules and contextual information stored in a dictionary.

The two last steps in the database-system design concern the implementation design and the selection of the physical database structure. The result is a complete, implementable database structure. In these last steps, the context is fixed, i.e., it is the application itself.

The quality of the resulting database structure depends on the design methodology, the design techniques used in the methodology steps and the validity of the information requirements. Traditionally developed by designer or semi-automatic tools, the step of conceptual design is very important.

The goal of this work is to show how context-based techniques can be useful when elaborating the conceptual schema during database system design. The context-based technique that is used in our system is a set of contextual rules. These rules handle contextual information that is stored in dictionaries.

Section 2 discusses some related works on context in linguistics. Section 3 presents the architecture of our system that considers context during the translation from requirements, expressed in natural language, to E-R Model. Section 4 shows context-based techniques. Section 5 gives an experiment to validate our work. We conclude with a summary and directions for future research in Section 6.

2 Related Works

There has been a lot of research on the nature of context in natural language and its role in natural language processing. In this section, we present researches that use context to help at translation from natural language to formal language.

Burstein and Kaplan (1995) employ a practical approach for extracting from existing technologies the aspects that can be practically used to reduce the costs of analyzing the natural language responses to constructed-response items. For this, they developed two methods that use context as an important basis for analysis procedures.

Dybkjaer et al (1995) view context as a collection of aspects or contextual elements each of which may be defined and analyzed with respect to its specific contribution to dialogue understanding. They developed a system-directed dialogue for a spoken language dialogue system and on first experiments towards achieving mixed-initiative dialogue.

Maier (1995) shows how the notion of context has been refined to fulfill the requirements posed by a natural language processing system. It describes the context model of the speech translation system VERBMOBIL, which has been derived from constraints given by various sub-components of the system. In VERBMOBIL the context is stored in the dialogue memory that is incrementally constructed by the dialogue processing component. However, this context model handles only a specific domain.

Zarri (1995) adopts a very pragmatic understanding of the words knowledge represented in two contexts (internal/static and external/dynamic). The internal/static context includes the knowledge sources (morphological, grammatical and syntactical resources, conceptual dictionaries) inside the system before the launch of the translation operations. The external/dynamic context represents the contribution of the ongoing translation to the growth of the knowledge already possessed by the system. The aim of this system is to translate from natural language documents to framework called NKRL (Narrative Knowledge Representation Language) technology.

The advantage of our approach is the direct translation, independently of the domain, from system's requirements expressed in natural language into formal language. This is possible thanks to large dictionaries with contextual information and appropriated heuristic rules. The others approaches described above are interesting, but they do not focus on the same points that ours, i.e., formalizing of requirements expressed in natural language. In section 3, we present a brief description of our system and contextual rules are discussed in section 4.

3 Architecture of our system

This section describes the architecture of a tool developed to help the human designer in the task of database system development. It is composed of a morphological parser, a syntactical parser, a base of heuristic rules and contextual rules, dictionaries and a translator as showed in Figure 1.

3.1 Morphological parser

The morphological parser analyzes morphologically a natural language sentence using a dictionary and models that match and classify the words (Bigolin et al 1991). The output of the parser is the set of words from the original sentence, each one with its morphological characteristics.

3.2 Dictionaries

Our system has two dictionaries: ***a syntactical dictionary*** and **a morpho-contextual dictionary** (Bigolin, 1994). The former stores the syntactical information on the words. The latter has three parts that store: the words in Portuguese, the morphological information (substantives, verbs, articles, etc.) of these words and contextual information (for example, publications and books belong to same context).

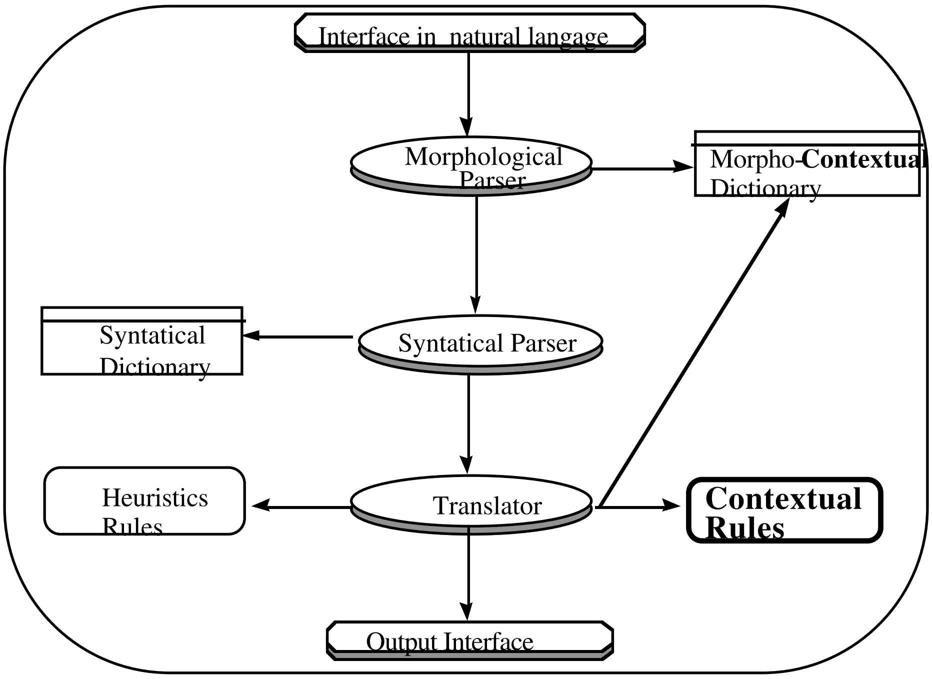

Figure 1 : Designer Assistant

Words are recorded in two different structures: a radical structure, which contains radicals of words and a suffix one, which contains suffixes of words. This structure form is a R-Tree, which is very efficient both for access and storage (Bigolin, 1991). The structure stores a letter in each node and the last letter of each word ends with two pointers: one towards a structure containing the *morphological information* and the other to a structure containing the *contextual information* of the words (see the figure 2).

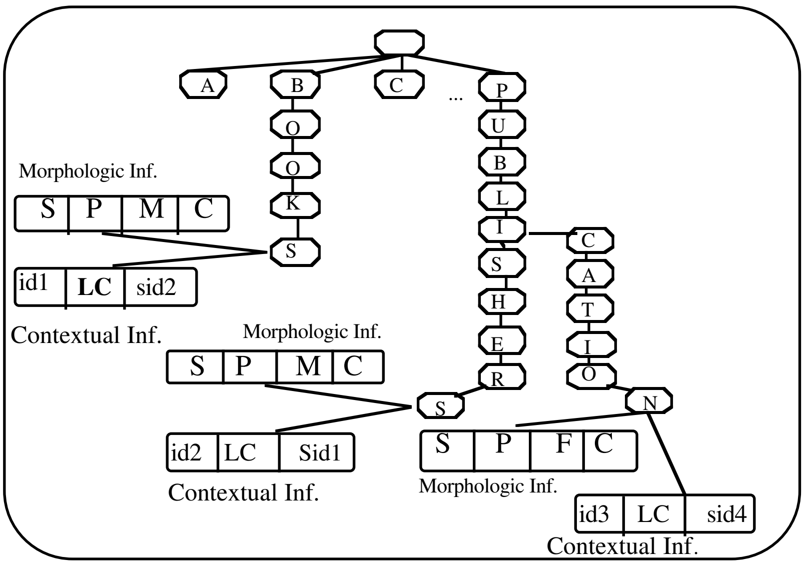

Figure 2 : R-Tree Structure

In this structure the last node has two pointers: - one to the **morphological information**, where *S* stands for substantive, *P*: plural, *M*: masculine, *c*: common, and - another to **contextual information**, where *id* stands for identification of the word, *LC*: the contexts in which the word is, *sid2*: word id1 is a subset of *id2* (for example, *books* is a subset of *publications*).

3.3 Syntactical Parser

For implementing an interface that handles natural language text, we use a grammar that allows the organization of words in sentences. Among several grammatical formalisms, we chose the Definite Clause Grammar (DCG) (Pereira and Warren, 1980), because DCG is the most popular approach to parsing in Prolog and it extends context-free grammar in three ways : 1. DCG provides context dependency ; 2. DCG allows arbitrary tree structures to be built in the process of parsing and 3. DCG allows extra conditions to be added to the rules, permitting auxiliary computations (Chau and Parker, 1992).

The syntactical analysis can be led in three ways: *top-down*, *bottom-up* and *graphs analysis*. We implemented two strategies: *bottom-up* (beginning by an initial symbol and applying grammatical rules until the nonterminal symbols of Tree correspond to the composition of the sentence analyzed); and *top-down* (where all rules begin with the search from words allowing to abandon the analysis if the word does not exist in the dictionary). After the tests, we note that bottom-up analysis is more efficient than the top-down analysis, due to the Left Extreme techniques (Lopes, 1993). This method analyzes the more left symbol to control the hypothesis generation during the analysis.

3.4 Heuristic Rules

We use heuristic rules to map system's requirements expressed in natural language to an E-R model. These rules are provided by the designer and other related works. The following examples illustrate the rules form:

- Rule 1: $\exists x$; verb(x) $\Rightarrow$ relationship(x)

  If the word(x) is a verb ***then*** the word(x) can be a relationship

- Rule 2: $\exists x$; substantive(x) $\Rightarrow$ entity(x)

  If the word(x) is a substantive ***then*** the word(x) can be an entity

- Rule 3: $\exists x,y$; substantive(y) $\wedge$ ((verb(x)=have) $\vee$ (verb(x)=is_composed)) $\wedge$ precedent(y,x) $\Rightarrow$ attribute(y)

If the word(y) is a substantive ***and*** the last word(x) before this substantive is the verb *have* or *is composed* ***then*** the word(y) is an attribute

Bigolin (1992) defined a set of rules that was still incomplete.

3.5 Translator

The translator translates from a decision tree to a conceptual model (E-R Model) using heuristic rules and contextual rules. First, it uses the heuristic rules to translate general information and second contextual rules to translate the information related with the application context. These rules are described in the next section.

4 Contextual Rules

The contextual rules are more specific than the heuristic ones. They are at two different levels of knowledge. The contextual rules allow to delimit the possible solutions during the translation from natural language requirements to E-R Model. A contextual rule is defined as follows:

- $\exists$c,p; context(c) $\wedge$ proposition(p) $\wedge$ context_of(p) = c $\Rightarrow$ entity(c), entity(p), relationships(c,p) = composition

If Context Cx and Proposition Px ***then*** Cx and Px are entities and relationships between them is a composition.

This rule is based on the McCarthy's Theory (1993), where the main formulas are sentences of the form:

$$c' = ist(c, p)$$

where the proposition ***p*** is true in the context ***c***, itself asserted in an outer context ***c'***. According to McCarthy, propositions may be identified with sentences in English or in various logical languages (In our approach, the propositions are words in Portuguese). For example, in context of *university's organization*, the proposition *department* is true:

C' : ist (context_of(university's organization), department) = true

On the other land, in *post-graduation evaluation*, the proposition *department* is false, but the proposition *student* is true. However, *department* is an attribute of student in context *post-graduation evaluation.* Thus,

C' : ist (context_of(post-graduation evaluation), department) = false

C' : ist (context_of(university's organization), student) = true

This example is represented in Figure 3.

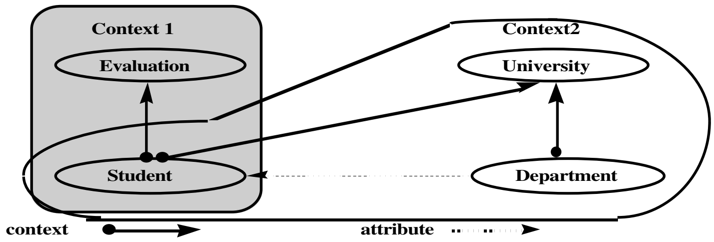

Figure 3 : Contexts

At the dictionary level, the word *department* is associated with *university* by a pointer. Thus, the system knows that *university* can be composed of *department, student*, etc. and with this information the translator generates a conceptual model (E-R Model) with two entities (*university's organization* and *department*) and a relationship of the type *composition*.

5 Experiments

To validate this system, we present an application, where the translation is performed along the following steps:

1. Morphological Analysis. The input is a description expressed in natural language; the output is the input words with their morphological characteristics. For example, the input is the sentence: *The publishers publish books* and the output is: **the**: article, definite, **publishers**: substantive, plural, masculine, **publish**: verb, **books:** substantives, plural, masculine. These words are stored in a data structure called R-Tree presented in Figure 2 (section 3.2).

2. Syntactical Parser. The input is a sentence with morphological characteristics and the output is the Derivation Tree. For example, Figure 4 shows the Derivation Tree, which is generated by the system, corresponding to the sentence *The publishers publish books*.

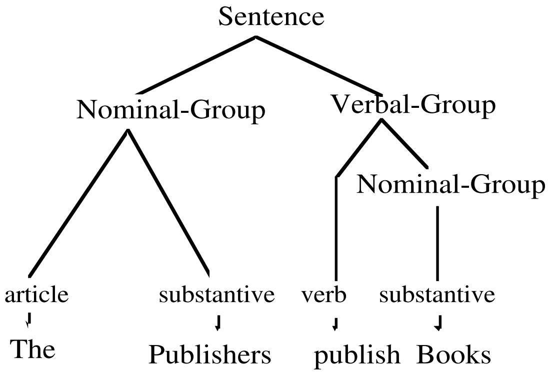

Figure 4 : Derivation Tree

3. Conceptual Translation. Using heuristic and contextual rules, the translation transforms the input (the derivation Tree) into the output (an E-R model), according to the model presented in Figure 5.

Figure 5 : E-R Model

In the library application, we have user's descriptions expressed in natural language (in Portuguese) as shows Figure 6:

System's Requirements

As publicações são editadas por editoras diferentes que possuem nomes, endereços. As publicações possuem um código, um titulo e refere-se a um tópico. Cada tópico possui um código, um nome, uma área de pesquisa. Nas publicações interessam os livros, os quais possuem uma tiragem. Os livros são adotados por diversas universidades. Cada universidade possui um nome, uma cidade. As editoras editam livros.

Figure 6 : Requirements expressed in natural language

The text means in English: *Publications are edited by different publishers who have a name and an address. Publications have a code, a title and are associated to a subject. Each subject has a code, a name and an area of search. Among the publication items, our interest concerns the books which have an edition. The books are used by several universities. Universities have a name and a country. Publishers publish Books.*

First, the morphological parser classifies the system's requirements expressed in natural language using the morphological dictionary. Second, the syntactical parser builds a derivation tree from each sentence, using a syntactical dictionary. This dictionary was built only from words of the application. Third, the translator transforms the derivation tree into a conceptual model (E-R model) using heuristic and contextual rules. To explain the relevance of using context-based techniques, we first present a result (see E-R Model in Figure 7) obtained first without considering the context then by considering it.

This schema shows the relationship *interessam* of the type ***association*** between *Livros (Books)* and *Publicações* (*Publications)*. However, that association relationship is wrong because *Livros* is a type of *Publicações*. Therefore, the type of relationship between *Livros* and *Publicações* must be a ***composition***, not ***association***. To solve this problem, we use the contextual information stored in a dictionary and the contextual rules. In this application, *Livros* is stored in an dictionary as a subset (or proposition in the case of MacCarthy's Theory) of *Publicações*, and according to the rule :

***If** Context Cx and Proposition Px **then** Cx and Px are entities and relationships between them is a composition*,

we identify the relationship **composition**. Thus, in our example, ***Cx*** is *Publicações* and ***Px*** is *Livros*, then the relationship must be a **composition**. The result obtained when context is considered explicitly is showed in Figure 8.

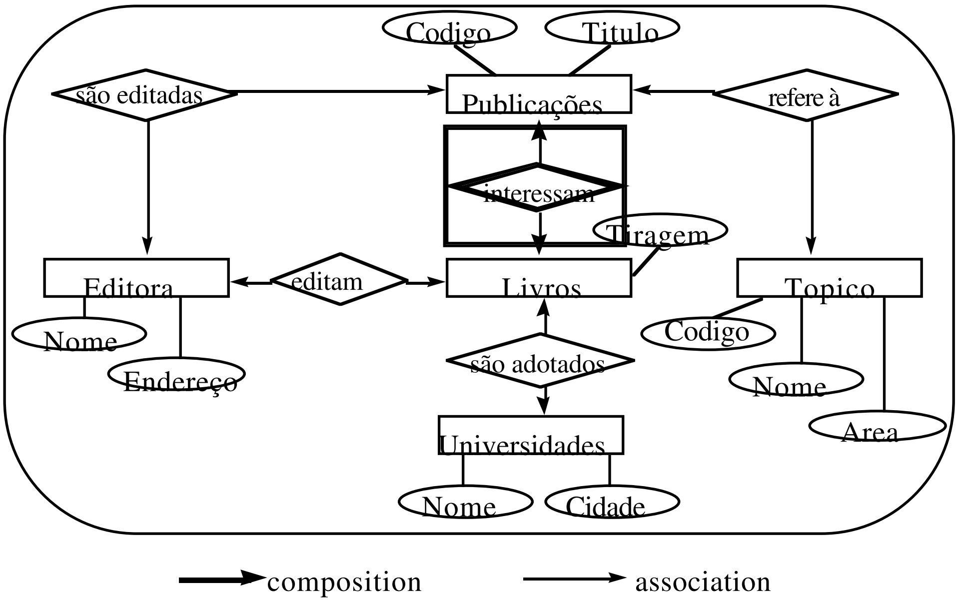

Figure 7 : Entity-Relationship Model without considering the context

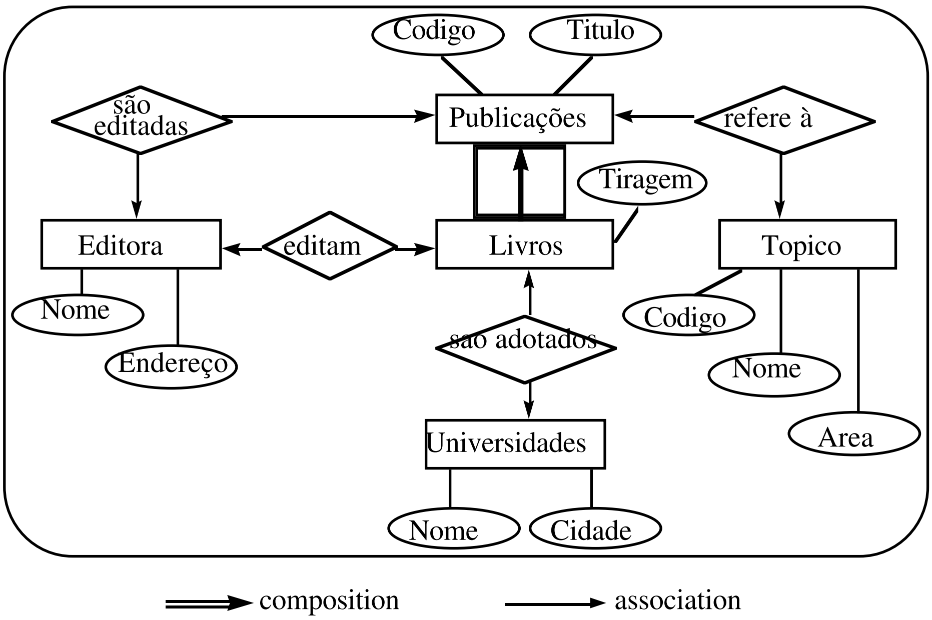

Figure 8 : Entity-Relationship Model considering the context

The input and output interfaces of the system were developed using a graphic tool. The input interface is a text editor and the output interface is a graphic editor. The Morphological Parser was implemented in Language C. The syntactical parser and the translator were implemented in

Prolog. The system inputs are the requirements expressed in natural language (Portuguese) and the output is an E-R Model. However, it is possible to adapt easily these techniques to another conceptual model.

The dictionary is composed approximately of 15.000 words and may be used in several domains. The tools have been tested by designers and have been considered robust.

6 Conclusion

This paper described a translation tool from system's requirements expressed in natural language to conceptual model (E-R) using contextual techniques. This system is composed of a morphological parser, a syntactical parser, heuristic and contextual rules, and (morphological, syntactical and contextual) dictionaries to simplify this process.

The main results of our investigation are:

- Automating the process of conceptualization of requirements (expressed in Portuguese Language) in database system design.
- Use of contextual approaches during this process to improve the formalization of natural language.

We believe that some of the ideas we showed have already been thought about by researchers. But our hope is to achieve to show the reader a valid experience on the use of context in translation of natural language. This study encourages us, in future works, to develop other heuristic and contextual rules to make the system more robust.

References

IJCAI-95 (1995), Workshop on Context in Natural Language Processing, Montreal, Canada.

Bigolin N.M et al. (1991), Dictionaries manager system, Technical Report, Institute de Informatica, PUC/RS, RS, Brazil. 96p.

Bigolin, N.M. (1992), *Tool for aiding the translation of specification languages in the development of database systems*, Technical Report, CPGCC-UFRGS, RS, Brazil, 58p.

Bigolin, N.M. (1994), *Application of natural language treatment techniques to support database design*, M.sc. these, CPGCC-UFRGS, RS, Brazil, 178p.

Brézillon, P. and Abu-Hakima, S. (1995), Principles for the applications of context in diagnostic problem solving*, in* Working notes on *the IJCAI-95 Workshop on "Modeling Context in Knowledge Representation*, Montreal, Canada.

Brézillon, P. and Abu-Hakima, (1993), S. Report on the IJCAI-93 Workshop on Using Knowledge in its Context*, in AI Magazine*, 16(1): p.87-91.

Burstein, J. and Kaplan, R. M. (1995), On the application of context to natural language processing applied to the analysis of test responses*, in IJCAI-95 Workshop on Context in Natural Language Processing*, Montreal, Canada.

Buvac, S. (chair), editor. (1995), *Formalizing Context*, AAAI-95 Fall Symposium, MIT, Cambridge, Massachusetts, November 10-12.

Chau, H.L. and Parker, D.S. (1992) Narrowing Grammar: Theory, Implementation, and Applications, *in Journal Logic Programming* 14, pages 253-286.

Dybkjaer, L. Ole Bernsen, N. Dybkaejr, H. and Papazachariou, D. (1995), *On the use of context in building spoken language dialogue systems for large tasks, in* IJCAI-95 Workshop on Context in Natural Language Processing, Montreal, Canada.

Edlund, C. Weise, C. and Lewis, M. (1995), Context engineering for human problem solving*, in IJCAI-95 Workshop on Modeling Context in Knowledge Representation and Reasoning*, Montreal, Canada, pages 79-88.

Farwell, D. and Helmreich, S. (1995) Contextualizing natural language processing*, in AAAI-95 Fall Symposium - Formalizing Context*, Cambridge, Massachusetts.

Giunchiglia, F. (1993) Contextual reasoning*, in Proceedings of the IJCAI-93 Workshop on Using Knowledge in its Context.*

Hian Goh, C. Madnick, S. and Siegel, M. (1993), Context Interchange: Overcoming the challenges of large-scale interoperable database systems in a dynamic environment*, in Proceedings of the Third International Conference on Information and Knowledge Management*, Gaithersburg, pages 337--346.

Lopes, J.G. (1993), *Natural language*. Internal Report, CPGCC-UFRGS, Porto Alegre, RS, Brazil.

Maier, E. (1995) A multi-dimensional representation of context in a speech translation system- a practical approach*, in IJCAI-95 Workshop on Context in Natural Language Processing*, Montreal, Canada.

McCarthy, J. (1993) Notes on formalizing context*, in Proceedings of IJCAI -1993*, pages 555-560, http://www-formal.stanford.edu/jmc/context.html.

McCarthy, J. and Buvac, S. (1994), *Formalizing context,* Technical Report, Computer Science Department - Stanford University, 26p., http://www-formal.stanford.edu/jmc/ index.html.

Paton, R. et al. (1993), Domain characterization in context*, in Proceedings of the IJCAI-93 Workshop on Using Knowledge in its Context*, pages 100-110.

Pereira, F.C.N. and Warren, D.H.D. (1980), Definite clause grammars for language analysis*, in* Artificial Intelligence, 13(3), p.231-278.

Storey, V.C. (1993), A selective survey of use of artificial intelligence for database design systems*, in Data & Knowledge Engineering*, Amsterdam, pages 61-102.

Strat, T.M. and Mundy, J.L. (1995), editors*, in Proceedings of the IEEE Workshop on Context-Based Vision (CBVIS '95).*

Teorey, T.J. and Fry, J.P. (1982), *Design of Database Structures*, Prentice-Hall.

Zarri, G.P. (1995), Internal and external knowledge context, and their use for the interpretation of natural language*, in IJCAI-95 Workshop on Context in Natural Language Processing*, Montreal, Canada.