IDEA GROUP PUBLISHING

701 E. Chocolate Avenue, Suite 200, Hershey PA 17033, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com

ITP4515

Data Conceptual Modelling through Natural Language: Identification and Validation of Relationship Cardinalities[1]

Paloma Martínez, Adoración de Miguel, Dolores Cuadra, Carlos Nieto, Elena Castro
Departamento de Informática, Universidad Carlos III de Madrid, Avd. Universidad 30, 28911 Leganés, Madrid, Spain, Ph: +34 91 6 24 94 54, Fax: +34 91 6 24 94 30, {pmf,admiguel,dcuadra,cnieto,ecastro}@inf.uc3m.es}

ABSTRACT

The work presented in this paper is part of a research framework devoted to database conceptual modelling that integrates various knowledge sources and technologies for helping novice database analysts and students in different specific database development tasks. This paper shows an overview of our progress in a special topic: identification and validation of binary relationship cardinalities in conceptual modelling phase. Our approach profits from natural language processing (NLP) techniques, first-order logic and some modelling heuristics and it smoothes the way to face the identification and refinement of ternary relationships.

1. BACKGROUND

A conceptual schema, independently of data formalism used, plays two main roles in the conceptual analysis phase: a *semantic role*, in which user requirements are gathered together and the entities and relationships in a universe of discourse (UoD) are documented and a *representational role* that provides a framework that allows a mapping to the logical design of database development. Three topics are involved in the database conceptual modelling process: data formalism, methodological approach and CASE tool support.

Related to the data modelling formalisms, the Extended Entity Relationship (EER) model has proven to be a precise and comprehensive tool for representing data requirements in information systems development, mainly due to an adequate degree of abstraction of the constructs that it includes. Since the original ER model was proposed by Chen (1976), many extensions and variations as well as different diagrammatic styles have been defined, McAllister (1995).

Atkins (1996) describes two different approaches to conceptual modelling process: prescriptive and descriptive approaches. To achieve an engineering approach, we believe that, ideally, a database conceptual methodology should be a combination of both of them in a formalised framework, that is, including imaginative processes that can be guided by different heuristics and established methods that define the sequence of events that lead to a conceptual schema.

In practice, requirements elicitation and collection is mainly done using natural language. Thus, it is reasonable to search for methods for systematic treatment of specifications. Since the eighties many efforts have tackled methodological approaches for deriving conceptual schemata from natural language specifications, Chen (1983), Booch (1986), Rumbaugh et al. (1991), and some of them include NLP technology, Saeki et al. (1989), Dunn and Orlowska (1990). Recent state-of-the-art includes very interesting approaches such as COLOR-X, Burg and Van de Riet (1996), which transforms a textual document into conceptual models through a word semantic driven process using a lexical database; NL-OOPS, Mich (1996), that follows a syntax driven approach to extract objects and their associations to be used in building object schemata; Gómez et al. (1999) proposes a natural language (NL) based ER schema generator, whose semantic analysis produces a logical form for each sentence that is used to extract ER constructors including ternary relationships.

Concerning commercial CASE tools for database developments, they do not cover database analysis phase with real EER models, that is, they only provide graphical diagrammatic facilities without refinement and validation tools that are independent of the other development phases. CASE environments usually manage hybrid models (merging aspects from EER and Relational models) sometimes too close to physical aspects and they use a subset of ER graphical notation for representing relational schemata.

In this context, one of the most difficult concepts to be modelled in database conceptual analysis is relationship, specially higher order relationships[2] as well as its associated cardinalities. Some textbooks, Ullman and Widom (1997), Boman et al. (1997), assume that any conceptual design can be addressed by considering only binary relationships since models they manage are computer oriented. We understand the advantages of this approach although we believe that it may produce certain loss of semantics (some biases are introduced in user requirements) and it forces to represent information in rather artificial and sometimes unnatural ways. Concerning the literature on relationships, significant works are the studies on modelling performance for ternary relationships, Bock and Ryan (1996), on cardinality constrains, McAllister (1995), on rules and heuristics that should be followed in determining the relationships among entities, Batra and Zanakis (1994).

We focus this paper on the issue of relationships and cardinality constraints associated to them. Since one of the major purposes of a conceptual schema is to assist in communication of information structures, our concern in this research is an interactive prescriptive method that takes advantage of the knowledge acquired from NL for obtaining conceptual schemata. This approach is an extension of previous works, Martínez (1998), Martínez et al. (1998), that proposes a general structured knowledge model for text interpretation that provides a non deterministic control of the linguistic processes involved in text analysis by using different NLP techniques. Its main contribution is the definition of several strategies that combine the linguistic knowledge sources (morphology, syntax, semantics) to guide the analysis process. These analysis strategies are so-called linguistic perspectives and are based on syntactic and semantic cues embedded in the text, for instance, style specific patterns, keywords, verbs with semantic preferences, etc.

Cardinality constraints, especially in higher order relationships, are difficult to understand and model by students, and certain validation methods are required. What we propose in this paper is an approach that combines syntax (grammatical categories, word collocations, etc.), semantics (meanings of words, phrases and sentences) as well as first order logic to extract cardinality constraints and validate them with the user. The aim of this work is to enhance conceptual database learning in identification and validation of cardinality constraints, but it takes part of a wider project, PANDORA whose objective is to define methods and techniques for database development implemented in a CASE tool, useful for students and practitioners.

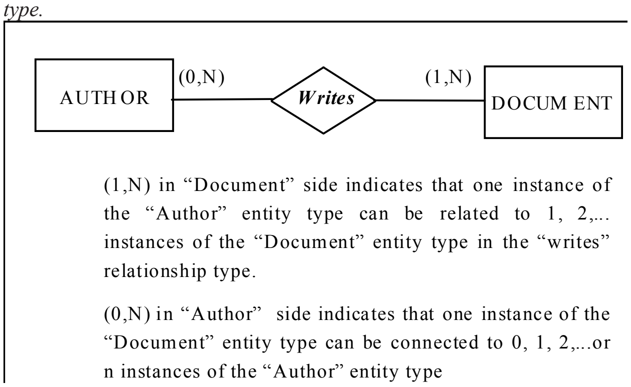
Next section is focused on the concept of cardinality constraint because there are several interpretations of this kind of constraints, Chen (1976), Tardieu (1989), McAllister (1995). The identification of cardinality constraints from linguistic elements is explained in Section 3. Section 4 describes the interpretation and validation process of cardinality constraints and, finally, some conclusions are given in Section 5.

2. DEFINING CARDINALITY CONSTRAINTS TERMINOLOGY

Chen (1976) characterises the maximum cardinality (also specified as connectivity) as the mapping of the associated entity occurrences in the relationship (values for connectivity are either 1 or N). Tardieu (1989) establishes the maximum and minimum cardinalities constraint as the maximum and minimum number of times that an occurrence of an object participates in the occurrences of a relationship. Following Chen's defini-

This paper appears in *Challenges of Information Technology Management in the 21st Century,* the proceedings of the Information Resources Management Association International Conference.

Figure 1: Example of minimum and maximum cardinalities of the "Author" and "Document" entity types in the "writes" relationship type.



tion, we define the maximum and minimum cardinalities of entity types that participate in a relationship type as the maximum and minimum number of instances in one entity type that can be associated in a relationship to a single instance in the other entity type(s), De Miguel et al. (1999). Graphically, cardinality constraints are represented by a tag, such as (0,1), (1,1), (0,N) or (1,N), located over the line connecting the entity type to the diamond that represents the relationship type (Figure 1).

Notice that as is proposed in Tardieu (1989), cardinality tags of binary relationships are inverted with respect to those of Figure 1.

3. DISCOVERING CARDINALITIES FROM LINGUISTIC ELEMENTS: QUANTIFIERS, ADVERBS AND OTHERS.

An analysis of a wide corpus of Spanish short descriptive texts describing several UoDs has allowed us to identify the linguistic elements (determiners, adverbs, adjectives and others) that help to obtain entity and relationship types as well as cardinality constraints. The idea is to translate the sentences into logical formulas, taking into account some rules of scope and precedence of quantifiers, in order to validate semantics with the user. First order predicate logic is a formal framework that allows us to represent the static part of conceptual schemata[3]. The aim of validation phase is to enhance the comprehension of conceptual modelling constructs. To accomplish the overall process, there are three topics to be investigated:

- Mapping natural language sentences into quantified logical formulas. For instance, the semantic interpretation of the sentence "*Todo empleado trabaja en un departamento*"[4] is úx(empleado(x) ‡ ∃ y (departamento(y) trabaja_en(x,y))[5]
- Mapping universal and existential quantifiers as well as other kinds of restrictors from set theory into possible relationship cardinalities (EER model). Some examples are shown in section 3.
- Logical formula validation through some examples given by the user. This phase is necessary to avoid literal translation errors, Batra and Antony (1994). The previous sentence can be translated to ~ (empleado(x) ~(departamento(y) trabaja_en(x,y)))[6] in order to be refuted and tested in PROLOG using instances of *empleado(x)*, *departamento(y)* and *trabaja_en(x,y)*.

We propose an interpretation of NL sentences based on Model-Theoretic semantics that relies on logic and set theory. For this purpose, the Definite Clause Grammar (DCG) formalism, Pereira and Warren (1980), has been used to develop a grammar for sentence analysis that integrates syntactic and semantic aspects. In order to automatically obtain a semantic representation of sentences, the grammar implements a set of hypotheses about the linguistic elements that have a quantification sense. Tables 1 and 2, an extension of Moreno (1993), show the correspondences among some articles, adjectives, etc., and existential and universal quantifiers and cardinality restrictors[7]. Some of them are:

1. Universal quantifier (ú) is introduced by plural definite articles, a subset of indefinite adjectives and a subset of indefinite pronouns.
2. Existential quantifier (∃) is introduced by indefinite articles, a subset of indefinite adjectives, personal pronouns and a subset of indefinite pronouns. It is also introduced by elliptical determiners.

Moreno (1993) and Abramson and Dahl (1989), describe several approaches for the scope and precedence of quantifiers in sentences that have been also considered in grammar definition. Some of these hypotheses are:

1. The quantification introduced by the subject of a sentence dominates the quantification introduced by the verb complement.
2. The quantification introduced by a noun complement dominates the quantification introduced by the noun.

An example of previous hypothesis can be observed in the sentence "*los profesores imparten varias asignaturas*"[8]: definite article "los" (the) inserts ú in the logic formula úx(profesor(x) ∅ ∃ y (asignatura(y) imparte(x,y))[9] and it dominates subject, verb and complement; indefinite adjective "varios" (some) introduces ∃ and it dominates the verb and its complement.

As is shown in Tables 1 and 2, cardinal adjectives and pronouns (*two*, *three*, ...) as well as some adverbial phrases that indicate quantity (*at least*, *at most*, ...) are required to extend logic quantifiers with the notion of cardinality of a set.

4. SUGGESTING AND VALIDATING BINARY RELATIONSHIP CARDINALITIES.

In this section, we explain the interpretation process of sentences[10] whose aim is to obtain the entity types that participate in the relationship type as well as minimum and maximum cardinalities associated to both entity types in the relationship type[11]. The process has been implemented in PROLOG[12] language and it is decomposed into three steps, last one decomposed again in two tasks (Figure 2).

Table 1: Mapping among linguistic elements and logical quantifiers

	∃	ú	∃	ú
Definite Articles	el, la	los, las		
Indefinite Articles	un, una, unos, unas			
Indefinite Adjectives	Algún, alguna, algunos, algunas, varios, varias, determinados, diversos, distintos, ciertos	cada, cualquier, todo, toda,	ningún, ninguna, ningunos, ningunas	
Indefinite Pronouns	Alguien, alguno, alguna, algunos, algunas, varios, varias	cualquiera, todo, toda, todos, todas	nadie, ninguno, ninguna	
Personal Pronouns	el, ella, ellos, ellas			
Modal Verbs		deber, tener que,		poder

Table 2: Mapping among linguistic elements and cardinality of a set

number	equal to number único, solo,	less_than number	greater_than
General Adjectives	determinado, ...		
Modal Adverbs	sólo, solamente,...		
Quantity Adverbial phrases	exáctamente	a lo sumo, como máximo, menos de, como mucho,...	como mínimo, al menos, más de, ...
Cardinal Adjectives & Pronouns	uno, una, un, dos, tres, cuatro,...		

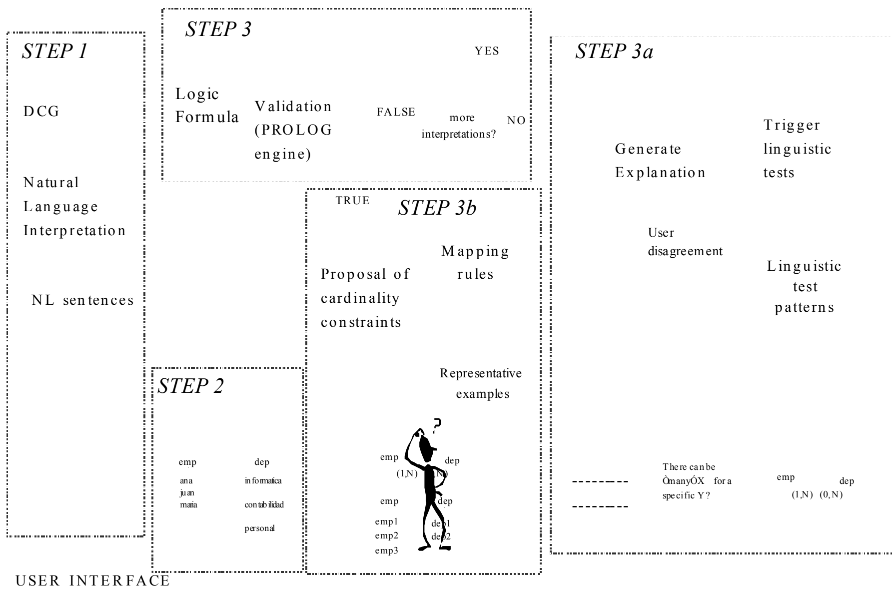
Briefly, first step is in charge of processing the NL sentence to obtain a logic formula, second step gives the user the possibility on entering data in order to validate the semantics; this validation is carried out in the third step with two different behaviours. Below, how these steps are carried out is further explained.

Step1: Analyse the sentence using a DCG grammar, whose rules have syntactic, semantic and pragmatic features.

The natural language analysis follows the compositional approach, that is, syntactic, semantic and pragmatic analysis are performed at the same time. Syntactic features of each rule denote the morphosyntactic category of sentence constituents (words or phrases) such as noun, verb, noun phrase, preposition, etc. Their objective is to get a parse tree that reflects the structure of the sentence and whose constituents are used to obtain at the same time the semantic constituents that compose the logic formula.

Semantic features have to do with lexical, phrase and sentence meanings, and they are used to obtain a predicate logic formula that represents the meaning of the sentence. In this approximation, the semantic feature associated to a non relational common noun[13] is a subformula composed of a unary logic predicate, for instance, the semantic feature of *department* is *[X, department(X)]*; n-ary logic predicates are defined by verbs (the number of arguments depends on the essential and modal syntactic and semantic roles required for verbs), for instance, the semantic representation for the verb *trabaja_en*[14] is *[I, [J, trabaja_en(I,J)]]* where the predicate is the name of the verb and the two arguments represents the agent (*I*) and the place (*J*) of the action denoted by the verb. The grammar incorporates the results of a syntactic and semantic verbal classification,

Figure 2: Process for identifying and validating relationship cardinalities

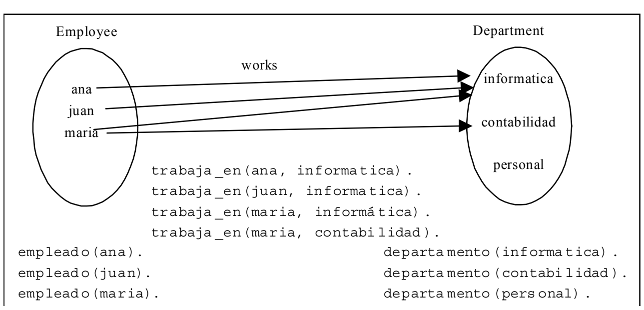


Martínez (1998), that distinguishes several types of verbs depending on syntactic arguments (subject, direct object, etc.) and semantic roles (agent, object, time, etc.).

Finally, pragmatic features are related to EER model constructs, De Miguel et al. (1999), in our case, entity and relationship types and cardinality constraints. For example, the analysis of the sentence "*Cada empleado trabaja en al menos un departamento*"[15] produces the result:

Syntax_Tree = decl_s(
  ng(art(cada),cn(empleado)),
  vg(v_biintr1(trabaja),
  prep(en),
  ng(adverb(al_menos),cardinal(un),cn(departamento))))
Semantic_Formula= [H180, forall(H180,
  implies(empleado(H180),
    [H504,num(H504,greater,1,
    conj(departamento(H504),trabaja_en(H180,H504)))]))]
Pragmatic_Result= [relationship_name,trabaja_en,
  [participant1,empleado,['0?','n?']],
  [participant2,departamento,[1,n]]]
Pseudo-graphical representation =
  empleado<—(0?,n?)—trabaja_en—(1,n)—>departamento

Figure 3: Venn diagrams and PROLOG facts for data supplied by the user



The semantic formula[16] represents that "*for every employee X there is a number of departments Y greater or equal to 1 that match the subformula departamento(Y)* Ÿ *trabaja_en(X,Y)*". The pragmatic result indicates that there is a relationship type called "trabaja_en" whose participants are "empleado" and "departamento" and a cardinality constraint associated to entity "departamento" [1,n]. The other cardinality appears as the least restrictive ['0?', 'n?'] because we do not know anything about it and in this case we do not introduce restrictions.

Step 2: The user is given the choice of introducing domain values corresponding to the semantic predicates that appear in the analysed sentence.

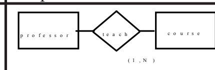
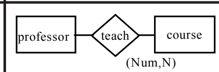
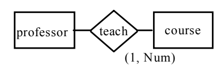
This means that some additional data are required to test if the semantic interpretation is correct. We assume that to achieve a data set representative enough is a complicated issue especially if there are many relationships, but it is not an obligatory step. Moreover, it can be solved in step 3 by means of explanations. So, this is the most crucial step and a dialog is started to obtain from the user instances of semantic predicates in the UoD that (s)he knows. Data sets are incorporated using Venn diagrams notation, as is shown in Figure 3, because we believe that this representation is intuitive and close to the user.

Step 3: Validate the semantic formula F with the domain values instantiated by the user in step 2.

Based on the data supplied, the Prolog interpreter validates the semantic form F by refutation, that is, proving if ÿF is unsatisfiable. We use this method because from a user viewpoint this validation could be more comprehensible, showing those values that make F false. In the previous example, F is true as "there is no employee working in no department". If the fact *trabaja_en(ana,informatica)* was no present in Figure 3, the system would inform the user that the employee *ana* does not work in any department and so, F would be false[17].

Step 3a: If the formula F is true then the system informs users about the cardinalities assigned to the relationship obtained using mapping rules as is shown in Figure 2. These mapping rules establish the correspondences between logic quantifiers and cardinality of a set and cardinality constraints of relationships, that is, they associate semantic interpretation of parts of sentences to minimum and maximum cardinalities. Some of them are shown if Table 3. We use the predicates *equal*,

Table 3: Mapping rules

Mapping Rules		Examples of Mapping Rules	
Logic Formula Synopsis	Max and min cardinalities	Logic Formula	Graphical notation
ú (Sem_subject ∅ ∃ (Sem_verb_and_compl))	(1,N) in entity type side represented by the complement[19]	úx(professor(x) ∅ ∃ y (course(y) teach(x,y))	
ú (Sem_subject ∅ (greater_than(Sem_verb_and_compl, Num)))	(Num,N) in entity type side represented by the complement	úx(professor(x) ∅ greater_than (y, (course(y) teach(x,y)), Num))	
ú (Sem_subject ∅ (less_than(Sem_verb_and_compl, Num)))	(1,Num) in entity type side represented by the complement	úx(professor(x) ∅ less_than (y, (course(y) teach(x,y)), Num))	

greater_than and *less_than* to control, when necessary, the number of occurrences of a variable that make the logic formula true. All of them use the predicate *card(X,F,N)*[18].

The system also generates a number of examples that are representative enough to further explain the relationships (Figure 2). There are eight combinations of minimum and maximum cardinality constraints for binary relationships: (0,1), (1,1), (0,N), (1,N), (0,*Num*), (1,*Num*), (*Num*,N), (*Num*,*Num*)[20]. The representative data sets corresponding to those combinations are automatically generated to explain the cardinalities using Venn diagrams notation. Figure 4 shows some of them for cardinalities (0,1), (1,1), (0,N) and (1,N).

Step 3b: If the formula F is false, there are two possibilities:

a) If there are more than one sentence interpretations (because of semantic ambiguity), a new one is selected and process goes back to step 3. In the sentence "*Cada empleado trabaja en un departamento*"[21] the determiner "*un*"[22] could mean "at least one" but also "exactly one", although the more usual sense is the first one.
b) if there are not more interpretations, an explanation is displayed and some linguistic tests are triggered to obtain from the user the information required to fill cardinality constraints. The linguistic tests, Guarino (1992), can be easily generated using syntactic patterns whose slots (X and Y) are filled with the names of the entities detected in the sentence analysis. Some examples are
"There can be "many" X for a specific Y"
"There can be "one or more" X for a specific Y"
"Every X is connected to some Y"
"One specific X can be connected to at most one Y"

5. CONCLUSIONS AND FUTURE WORK

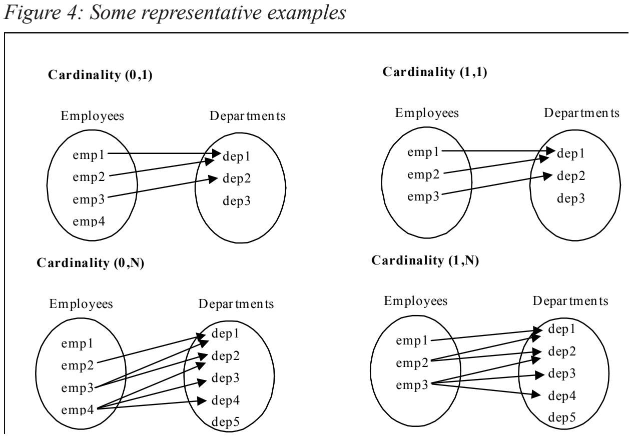
This paper presents our research progress on identifying and validating relationship cardinalities using NLP techniques. Currently, some experiments are being defined to test the suitability of the system with real cases with a test group of users. Moreover, in order to incorporate more knowledge in relationships validation, some experiments are being driven to investigate designer performance in conceptual modelling (problems in detecting higher order relationships, minimum cardinalities and others) as well as experiments on the influence of the way teachers explain the conceptual modelling. The control of cardinality constraints, its semantics and translation into relational model is also being investigated, Cuadra et al. (1999) and Martínez et al. (1999).

One major contribution is to smooth the way to face the identification of ternary relationships that are quite complex to comprehend and model, Batra and Antony (1994). Next objective will be to extend the NL grammar to include higher order relationships, taking into account that the aim is only to propose predesign schemata that are used to interactively validate the relationships, namely degree and cardinality constraints. In order to be included in the grammar some heuristics are under study:

- the type of complements that can appear with verbs (essential and optional arguments) considering that a specific verb could handle different syntactic and semantic frames (for example, the Spanish verb *disponer* can denote to give instructions, to prepare and to posses);
- several nominal phrase patterns that are used to set relationships among components of a noun phrase, for instance, relations between a noun and an adjective are of special interest as well as the use of preposition *de* (*of*) that represents different types of semantics (possession, place, description, etc.).
- deverbal nouns[23] which require the same arguments as the related verb. They are not always entity types in a conceptual schema,
- relative and conjunctive clauses for relationship identification.

Finally, from an user interface viewpoint, graphical representations for entering data and displaying enough representative sets are also being studied.

Figure 4: Some representative examples



ENDNOTES

1 This work takes part of the CICYT project PANDORA (CASE Platform for Database development and learning via Internet), TIC99-0215
2 n-ary relationships with n>2
3 The user mixes different logics in his discourse (temporal, deontic, etc.), Chomicki and Saake (1998), but predicate logic is enough for the purpose of this work
4 "Every employee works in a department"
5 "x(employee(x) ‡ $ y (department(y) Ÿ works(x,y))
6 ~ (employee(x) Ÿ ~(department(y) Ÿ works(x,y)))
7 first row shows restrictors and first column shows linguistic elements
8 "the professors teach several subjects"
9 "x(professor(x) ‡ $ y (subject(y) Ÿ teach(x,y))
10 The structure of sentences is noun phrase + verb phrase
11 Although this proposal takes part of a more ambitious project, in this paper only simple sentence for extracting binary relationships are treated. We are investigating to apply these ideas to ternary relationships, one of the most difficult constructs to be captured in ER modelling.
12 AMZI! Prolog v 4.1
13 There are several classes of common nouns: relational and non-relational nouns, deverbal nouns, etc. We have studied all of them in order to interpret NL sentences which contain them as well as to translate them to EER constructs.
14 to work in
15 "Each employee works in at least one department" (although it is not usual, we consider this assumption for illustrative purposes)
16 The PROLOG variables H180 and H540 correspond to X and Y respectively.
17 Notice that it always depends on data representativeness
18 *card(X,F, N)* returns in N the number of instances of X that make true the formula F
19 Notice that the universal quantifier leads to the minimum cardinality *1*

and the existential quantifier leads to the maximum cardinality *N*.
20 *Num* indicates exactly a number (2, 3, ...).
21 "Each employee works in a department"
22 a
23 a noun which derives from a verb (for example, *compra* –shipping- , *préstamo* – loan -)

REFERENCES

Abramson and Dahl (1989), Abramson, H. and Dahl, V. *Logic Grammars*. Springer-Verlag, 1989.

Atkins (1996), Atkins, C. Prescription or Description: Some observations on the conceptual modelling process. *Proceedings of International Conference on Software Engineering: Education and Practice*, 1996.

Batra and Antony (1994), Batra, D. and Antony S. R. Novice errors in conceptual database design. *European Journal of Information Systems*, Vol. 3, N. 1, pp. 57-69, 1994.

Batra and Zanakis (1994), Batra, D. and Zanakis, H. A conceptual database design approach based on rules and heuristics. *European Journal of Information Systems*, Vol. 3, N. 3, pp. 228-239, 1994.

Bock and Ryan (1996), Bock, D. B. and Ryan, T. Modeling Ternary Relationships. *Journal of Computer Information Systems*, pp 60- 65, Summer 1996.

Boman et al. (1997), Boman, M., Bubenko, J. A., Johannesson, P. y Wangler, B. *Conceptual Modelling*. Prentice Hall Series in computer Science, 1997.

Booch (1986), Booch, G. Object-oriented development. *IEEE Trans Software Engineering*, 12, 2, pp 211-221, 1986.

Burg and Van de Riet (1996), Burg, J.F.M and van de Riet, R.P. Analyzing Informal Requirements Specifications: A First Step towards Conceptual Modeling. In *Applications of Natural Language to Information Systems*. IOS Press, 1996.

Chen (1976), Chen, P. The entity-relationship model – Toward a unified view of data. *ACM Transactions on Database Systems*, Vol. 1, N.1, pp. 9-36, 1976.

Chen (1983), Chen, P. P. English Sentence Structure and Entity-Relationship Diagrams. *Information Sciences*, 29, pp. 127-149,1983.

Chomicki and Saake (1998), Chomicki J., Saake G. (Eds.): *Logics for Databases and Information Systems*. Kluwer Academic Publishers, 1998.

Cuadra et al. (1999), Cuadra, D., Nieto, C., Martínez, P. and De Miguel, A. *Control de restricciones de cardinalidad en una metodología de desarrollo de Bases de Datos relacionales*, Novática, nº 140, 1999.

De Miguel et al. (1999), De Miguel, A., Piattini, M. and Marcos, E. *Diseño de Bases de Datos Relacionales*. Rama Ed, 1999.

Dunn and Orlowska (1990), Dunn, L. and Orlowska, M. A Natural Language Interpreter for the Construction of Conceptual Schemas. *Proceedings of 2nd Nordic Conference on Advanced Information System Engineering (CAISE´90).* Stockholm, Sweden, 1990.

Gómez et al. (1999), Gómez, F., Segami, C. and Delaune, C. A system for the semiatomatic generation of E-R models from natural language specifications. *Data & Knowledge Engineering*, 29, pp. 56-81, 1999.

Guarino (1992), Guarino, N. Concepts, attributes and arbitrary relations. Some linguistic and ontological criteria for structuring knowledge bases. *Data & Knowledge Engineering*, 8, pp. 249-261, 1992.

Martínez (1998), Martínez, P. Propuesta de estructuración del conocimiento lingüístico para interpretación de textos: Aplicación al diseño de Bases de Datos. PhD. Dissertation. Universidad Politécnica de Madrid, 1998.

Martínez et al. (1999), Martínez, P., Nieto, C., Cuadra, D. and De Miguel, A. Profundizando en la semántica de las cardinalidades en el modelo E/R extendido. *IV Jornadas de Ingeniería del Software y Bases de Datos*, Cáceres, 1999.

Martínez et al. (1998), Martínez, P., De Miguel, A. and Marcos, E. A Knowledge-based Approach to Database Conceptual Modelling through Natural Language. *International Workshop on Issues and Applications of Database Technology, IADT'98*. Berlin, Germany, July 6-9, 1998.

McAllister (1995), McAllister, A. Modelling n-ary relationships in CASE environments. *Proceedings 7th International Workshop on Computer-Aieded Software Engineering*, Toronto, 1995.

Mich (1996), Mich, L., NL-OOPS: from natural language to object oriented requirements using the NLP system LOLITA. *Natural Language Engineering*, 2 (2), 1996.

Moreno (1993), Moreno Boronat, L. *Formalismos lógicos para el análisis e interpretación oracional del lenguaje natural*. PhD Dissertation, Universidad Politécnica de Valencia.

Pereira and Warren (1980), Pereira, F., and Warren, D. Definite clause grammars for language analysis - A survey of the formalism and a comparison with augmented transition networks. *Artificial Intelligence*, 13, pp. 231-278,1980.

Rumbaugh et al. (1991), Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F. and Lorensen, W. *Object-Oriented Modeling and Design*. Englewood Cliffs, NJ, Prentice Hall, 1991.

Saeki et al. (1989), Saeki, M., Horai, H. and Enomoto, H. Software development process fromm Natural language specification. *ICSI'89, International Conference on Software Engineering, ACM*, pp. 64-73, 1989.

Tardieu (1989), Tardieu, H. *La METHODE Merise*. Paris:Editions d'Organisation, cop. 1989.

Ullman and Widom (1997), Ullman J. D. and Widom J. *A First Course In Database System*, Prentice-Hall International, 1997.

Related Content