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, methodology 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), Bouch (1986), Rumbaugh et al. (1991), and some of them include NLP technology, Saeki et al. (1989), Dunn and Orłowska (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 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 constraints, 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.
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. 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" is $\exists x \, \text{empleado}(x) \land \exists y \, \text{departamento}(y) \land \text{trabaja}_\text{en}(x,y)$.
- Mapping universal and existential quantifiers as well as other kinds of restriction 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. We have also considered some hypotheses about the linguistic elements that have a quantification sense. Tables 1 and 2, an extension of Moreno (1993), show the correspondences among some elements, adjectives, etc., and existential and universal quantifiers and cardinality restrictions. Some of them are:

1. Universal quantifier ($\forall$) is introduced by plural definite articles, a subset of definite adjectives and a subset of indefinite pronouns.
2. Existential quantifier ($\exists$) is introduced by indefinite articles, a subset of definite adjectives, personal pronouns and a subset of indefinite pronouns.

An example of previous hypothesis can be observed in the sentence "los profesores imparten varias asignaturas"$^{11}$: definite article "los" (the) inserts $\forall$ in the logic formula $\forall x \, \text{profesor}(x) \land \exists y \, \text{asignatura}(y) \land \text{imparte}(x,y)$ and it dominates subject, verb and complement; indefinite adjective "varios" (some) introduces $\exists$ 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 express 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 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>
<thead>
<tr>
<th>Table 1: Mapping among linguistic elements and logical quantifiers</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Definite Articles</strong></td>
</tr>
<tr>
<td><strong>Indefinite Articles</strong></td>
</tr>
<tr>
<td><strong>Indefinite Adjectives</strong></td>
</tr>
<tr>
<td><strong>Indefinite Pronouns</strong></td>
</tr>
<tr>
<td><strong>Personal Pronouns</strong></td>
</tr>
<tr>
<td><strong>Modal Verbs</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2: Mapping among linguistic elements and cardinality of a set</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>number</strong></td>
</tr>
<tr>
<td><strong>General Adjectives</strong></td>
</tr>
<tr>
<td><strong>Modal Adverbs</strong></td>
</tr>
<tr>
<td><strong>Quantity Adverbial phrases</strong></td>
</tr>
<tr>
<td><strong>Cardinal Adjectives &amp; Pronouns</strong></td>
</tr>
</tbody>
</table>

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.

**Step 1:** 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 cat-
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 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 is [I, [J, trabaja en(I,J)]] where the predicate is the name of the verb and the two arguments represent 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.

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” produces the result:

Syntax Tree = decl_s(ngart(cada), cn(empleado)), vg(y_bintr1(trabaja), prep(cen)), ng(adverbial.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

The semantic formula represents “for every employee X there is a number of departments Y greater or equal to 1 that match the subformula departmento(Y) Y 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.

**Figure 2:** Process for identifying and validating relationship cardinalities

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

The semantic formula represents “for every employee X there is a number of departments Y greater or equal to 1 that match the subformula departmento(Y) Y 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 \( \neg F \) 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 not 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.

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,
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)\(^\text{18}\).

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, M), (1, M), (N, M), (M, M)\(^\text{19}\). 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”\(^\text{21}\) the determiner "un"\(^\text{22}\) 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”

Step 4: The system interprets the cardinality constraints obtained in step 3, and the generated formulas are translated into relational model. The translation into relational model is also being investigated, Cuadra et al. (1999) and Martinez 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 presdesig 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 disponente can denote to give instructions, to prepare and to possess);
- 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.);
- deverb nouns 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.

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) Æ works(x,y))
6 “x(employee(x) Æ ¬(department(y) Æ works(x,y)))
7 First row shows restrictions and first column shows linguistic elements
8 “the professors teach several subjects”
9 “x(professor(x) £ y (subject(y) Æ 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 modeling.
12 AMZI! Prolog v 4.1
13 There are several classes of common nouns: relational and non-relational nouns, deverb 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

Table 3: Mapping rules

<table>
<thead>
<tr>
<th>Logic Formula Synopsys</th>
<th>Logical Formula</th>
<th>Graphical notation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{greater than} ) ( \text{less than} )</td>
<td>( \text{greater than} ) ( \text{less than} )</td>
<td></td>
</tr>
<tr>
<td>( \text{Sem_subject} ) ( \text{Sem_verb_and_compl} )</td>
<td>( \text{Sem_subject} ) ( \text{Sem_verb_and_compl} )</td>
<td></td>
</tr>
<tr>
<td>( \text{Sem_subject} ) ( \text{Sem_verb_and_compl} ) ( \text{Num} )</td>
<td>( \text{Sem_subject} ) ( \text{Sem_verb_and_compl} ) ( \text{Num} )</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 4:** Some representative examples
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


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


