GOOAL: An Educational Object Oriented Analysis Laboratory

H. G. Pérez-González  
Facultad de Ingeniería  
Universidad Autónoma de  
San Luis Potosí, México  
Tel. 52(444) 826-2334  
hectorgerardo@acm.org

Jugal K. Kalita  
Dept. of Computer science  
University of Colorado at  
Colo. Springs CO, USA  
Tel. (719) 262-3325  
kalita@pikespeak.uccs.edu

Alberto Salvador Núñez Varela  
Facultad de Ingeniería  
Universidad Autónoma de  
San Luis Potosí  
Tel. 52(444) 826-2334  
Alberto_snv@hotmail.com

Richard S. Wiener  
Dept. of Computer science  
University of Colorado at  
Colo. Springs CO, USA  
Tel. (719) 262-3325  
rswiener@acm.org

ABSTRACT
Our goal is to enable rapid production of static and dynamic object models from natural language description of problems. Rapid modeling is achieved through automation of analysis tasks. This automation captures the cognitive schemes analysts use to build their models of the world through the use of a precise methodology. The methodology is based on the use of proposed technique called role posets, and a semi-natural language (called 4W). First versions of this tool were used as prototypes to produce early design artifacts for very small (toy) problems. Current version has been successfully used as an educational tool in object oriented software engineering courses. We present the tool with its new complete features and results of its application in learning process. Original problem statements are automatically translated to 4W language. The produced sentences then, are analyzed with role posets to produce static model views. Finally the 4W sentences are used to generate dynamic views of the problem. This set of methods maximizes analysis process agility, promotes reusability and constitutes a valuable tool in the learning process of object thinking. The prototype tool: GOOAL (Graphic Object Oriented Analysis Laboratory) receives a natural language (NL) description of a problem and produces the object models taking decisions sentence by sentence. The user realizes the consequences of the analysis of every sentence in real time.

Categories and Subject Descriptors
D.2.1 [Requirements / Specifications tools]: Object Oriented Requirements Engineering

General Terms
Design, Experimentation

Keywords
Requirements engineering, set theory, object oriented education, object oriented design.

1. PROBLEM DESCRIPTION
Although Natural Language (NL) processing is a very complex task [2][4][6], it is possible to extract sufficient meaning from NL sentences to produce reliable models [1]. This tool called GOOAL (Graphic Object Oriented Analysis Laboratory) supports a proposed methodology that helps in object-oriented design (OOD) from English or Spanish requirements sentences. Our goal is to take a NL specification such as the one given below and produce the UML [3] models for it. The following figure shows the sentences that describe a problem known as the towers of Hanoi as shown in Rumbaugh’s book [8].

![Figure 1 Towers of Hanoi description sentences](image)

Our challenge is to show which ones of the elements described here have enough merits to become a Class according with GOOAL tool. The Tool’s output is the probability each noun has to become a class, and a proposed problem domain class and object diagrams. In summary, this tool goes NL problem description to the UML diagrams using a systematic methodology that helps in the process of OO design learning.

2. ROLE POSETS
We use the notion of partially ordered set of roles (Role Posets) to emulate the reasoning analysts perform when they model a problem. Software analysis and design are the product of a sequence of decisions taken. Different modelers can produce different models for a given problem depending on the decisions they make. The decisions can be influenced by a number of different factors such as the design priorities and previous experience of the modeler, the nature of the problem itself and the priorities and expectations of the consumer. Role Posets are
based on the mathematical concept of poset (Partially ordered set) and the thematic (teta) roles due to Chomsky [5] and are used widely in many linguistics formalisms. We use Role Posets to construct the necessary structures to model a problem and a semi NL called 4W.[7].

3. CLASES AND SYNTACTIC ROLES

The decision that makes a noun becomes a class depends on the analysis of the complete requirements set, the potential future additions to the system and perception, experience and motivation of the analyst. Some automatic methodologies [4][7] make class identification using the frequencies of all the nouns in the requirements text. We propose a heuristic algorithm to identify classes considering also the roles that these nouns play in every sentence: The probability that a noun becomes a class varies proportionally to its importance or value in a text:

\[ \text{probToClass}(\text{NounX}) = \text{value}_i(\text{NounX}) \]

The importance of a noun in a text is computed as the sum of the importance of that noun in every sentence. We write it as:

\[ \text{value}_i(\text{NounX}) = \sum_{j=1}^{m} \text{value}_j(\text{NounX}) \]

where \( m \) is the number of sentences in the text and \( i \) is the index of the particular sentence. The importance of a noun in a sentence \( S \) depends on the role it plays in this sentence. We write it as:

\[ \text{value}_i(\text{NounX}) = \text{role}_i(\text{NounX}) \]

where \( i \) is the index of the particular sentence.

According to Chomsky[5]: “The role a noun plays in a sentence depends on the relative position it has in the sentence and on the semantic of the main verb of that sentence”. We express Chomsky’s statement as:

\[ \text{role}_i(\text{NounX}) = \text{position}(\text{NounX})_i + \text{semantics}(\text{V})_i \]

where \( i \) is the index of the particular sentence.

Details of calculating the significance of roles for a specific problem description text and the verbs that define these roles are discussed in our work [8] presented in OOPSLA 2002.

Although the criteria a noun should satisfy to be considered a class, depend on many factors including the past experience of the analysts, it can be said that one important consideration is the role the noun plays when interacting with other nouns in the text. The next step is the identification of the potential classes and their roles in the overall document context.

4. DEMONSTRATION OF GOOAL

The audience of this demonstration will see how a group of sentences describing a problem are received by GOOAL system, it takes illustrative decisions and produces static model views of the problem, then the system shows its interpretation in 4W language which is validated by the user. This process will be repeated with several examples. Figure 2 shows a produced general class diagram for the Towers of Hanoi example.

Figure 2. General Class diagram from Hanoi example.

5. CONCLUSIONS

The prototype tools (An early version that works with spanish sentences has been developed) have produced good results with problems described in no more than eight sentences and 100 words on an average. It is being improved with more complex problems. Observed advantages of the use of this tool are formalization, standard notation, validation, traceability, efficiency and early identification of misunderstood requirements. Individuals using it, will see how a group of sentences describing a problem are handled by GOOAL. The system takes decisions with minimal user participation, shows its interpretation in 4WL and produces model views of the problem. Unique features of this tool are the underlying methodology and the production of dynamic models.

With the comparison of results between both GOOAL tools (English and Spanish) and the tests we are running with the students, it can be concluded that the promising results shows the program as a good educative tool

6. ACKNOWLEDGMENTS

The authors gratefully acknowledge the sponsorship of PROMEP, CONACYT SEP and UASLP, Mexico

7.REFERENCES