An Intelligent Question - Answering System for Natural Language

Ahmed A. S. Yamani

A thesis submitted in partial fulfilment of the requirements of the University of Greenwich for the award of the degree of Doctor of Philosophy

School of Computing and Mathematical Sciences
Faculty of Science and Engineering

March - 1998

University of Greenwich

London

Abstract

As applications of information storage and retrieval systems are becoming more widespread, there is an increased need to be able to communicate with these systems in a natural way. Natural Language applications in the 1990s, as well as in the foreseeable future, have more demanding requirements. Current Natural Language Processing approaches alone have proven to be insufficient as they lack to obtain linguistic understanding. A more suitable approach would be to adopt Computational Linguistics theories, such as the Lexical-Functional Grammar (LFG) theory complemented with Artificial Intelligence representation and processing techniques.

A prototype Question-Answering System has been developed. It takes Natural Language parsed interrogatives, produces the Functional and Semantic structures according to the LFG representation. It compares the functional behaviour of verbs and their linguistic associations in a given query with a general Object Model in that specific domain. It will then attempt to deduce more information from the given processed text and represent it for possible queries. The structural rules of the LFG and the deduced common-sense domain specific information resolve most of the common ambiguities found in Natural Languages and enhance the understanding ability of the proposed prototype.

The LFG theory has been adopted and extended: (i) to examine the constituents of the theoretical, syntactic and semantic of Arabic interrogatives, an area which has not been thoroughly investigated, (ii) to represent the Functional and Semantic Structures of the Arabic interrogatives, (iii) to overcome the word-order problem associated with some Natural Languages such as Arabic, (iv) to add understanding capabilities by capturing the common-sense domain specific knowledge within a specific domain.

Table of Contents

1. Introduction

2. Overview of Natural Language Systems

3. Description of Arabic Syntax

8. Prototype Implementation of the Question-Answering System

9. Conclusions, Future Developments, and Industrial Applicability

Appendices:

Dedication

Several people have contributed to this work in various ways and at different stages of its development. I am grateful to their support throughout these years and for keeping my spirits up in times of doubt.

I would sincerely like to dedicate this work to a very special group of people in my life, the whole of my family, my mother, my brothers, and my sisters. These people made the writing of this thesis possible, in particular, my father, who sadly died before seeing the final product.

Another special person to whom this work is dedicated is my wife who patiently and untiringly gave her time, even during our honeymoon!

Acknowledgements

Several years ago, this thesis was merely a seed in my imagination. It has taken root, grown and flourished thanks to a number of people who have been generous with their ideas, time and support throughout. I cannot begin to express my gratitude and although I take the opportunity afforded to me here to express my thanks, it is altogether insufficient to convey my indebtedness to them. My supervisor Dr. Ala Al-Zobaidie, has offered me invaluable insight during his supervision and helped to shape my ideas with great patience and understanding. Without his constructive criticism, this thesis would never have materialised.

My special thanks also to my other supervisors, namely, Professor Brian Knight and Dr. Ephraim Nissan for their support and for dealing so tactfully with administrative and academic issues on my behalf.

Needless to say, life at the University would not have been so pleasant without the cheerful and helpful staff at the School of Computing and Mathematical Sciences.

I have also greatly benefited from the comments and suggestions made by my colleges and by researchers at the School.

Last but not least, I would like to thank all my dear friends who have offered me invaluable friendly support throughout these years.

I would like to express my gratitude to the Royal Embassy of Saudi Arabia for their financial support to complete this research.
This research has been supported by grant from

The Saudi Arabian Cultural Bureau
29 Belgrade Square,
London SW1X
and
The Ministry of Higher Education,
Riyadh, Saudi Arabia

Grant Number 81/S

Declaration

I certify that this work has not been accepted in substance for any degree, and is not concurrently submitted for any degree other than that of *Doctor of Philosophy (PhD)* of the University of Greenwich. I also declare that this work is the result of my own investigations except where otherwise stated.

Ahmed Yamani

Acronyms

ADJ	Adjective
AGR	Agreement
CFG	Context-Free Grammar
CF-PSG	Context-Free Phrase Structure Grammar
C-Structure	Constituent Structure
DBMS	Data Base Management System
DCG	Definite Clause Grammar
Det	Determinative
F-Structure	Functional Structure
GB	Government-Binding Theory
GF	Grammatical Function
LDD	Long-Distance Dependencies
LFG	Lexical-Functional Grammar Theory
IntLang	Interrogative Language
KB	Knowledge Base
NLI	Natural Language Interface
NLP	Natural Language Processing
NLQAS	Natural Language Question Answering System
NP	Noun Phrase
OBJ	Object
OBL-ARG	Oblique Argument
OBL-LOC	Oblique Location
PL	Plural
PP	Prepositional Phrase
PRED	Predicate
QAS	Question-Answering System
QASA	Question-Answering System for Arabic
S	Sentence
SA	Standard Arabic
S-Structure	Semantic Structure
SN	Slot Name
SVO	Subject Verb Object
SV	Slot Value
SUBJ	Subject
URL	Uniform Resource Locations - World Wide Web Sites
TNS	Tense
VP	Verb Phrase
VSO	Verb Subject Object
WHQT	WH-Question Type
XP	Maximal Projection of any Major Category

List of Figures

List of Tables

Chapter 1
Introduction

1.1 Introduction to Natural Language Processing and Computational Linguistics

The idea behind this work is to develop a computer system using Natural Language (NL) as a front-end communication language. As a native speaker of Arabic, we felt that there was a need for a more in-depth study of the syntactic and semantic structures as well as common-sense domain knowledge of the language and hence, was motivated to provide a NL interface to a Arabic text. This idea has finally found expression in this project.

Research in NL front-ends to database dates back to the mid 1970's. Some of these projects, (e.g., Robot; see chapter two) emerged from the laboratory to become commercial products. Others lacked popularity or are still in the research stage of development.

NL has been described as any language that humans learn from their environment and use to communicate with each other as part of their intelligent behaviour. Kay, [Kay-80] defined NL as a system for encoding and transmitting ideas. It is this "system" which allows us to understand what to do, when to do it and how to do it. Whether language is spoken or written, every message has a structure and the elements of the language are related to each other in a recognisable way.

NLP refers to the analysis of human languages and involves the study of the structure of these languages, e.g., their grammar rules, [URL 02]. A typical language system may require something along the lines of: knowledge about the language itself such as word order(s), knowledge about the word arrangement, and the actual meaning of these words in terms of semantics and common-sense. It should be capable of accepting input in NL text, storing knowledge related to the application domain, drawing inferences from that knowledge, answering questions based on the knowledge and generating responses.

The term NLP can be divided into two parts, namely, the theoretical and the practical. The theoretical aspects deal with the construction of computational linguistic theories in order to mimic the human mind using psychological and philosophical way for better understanding. The practical aspects involve the actual engineering of NL system in order to implement an application in a domain such as traffic accident. Therefore, and within the field of Artificial Intelligence, NLP and Expert Systems have been influenced by human technology as the overall structure of AI tree in Figure (1.1) illustrates.

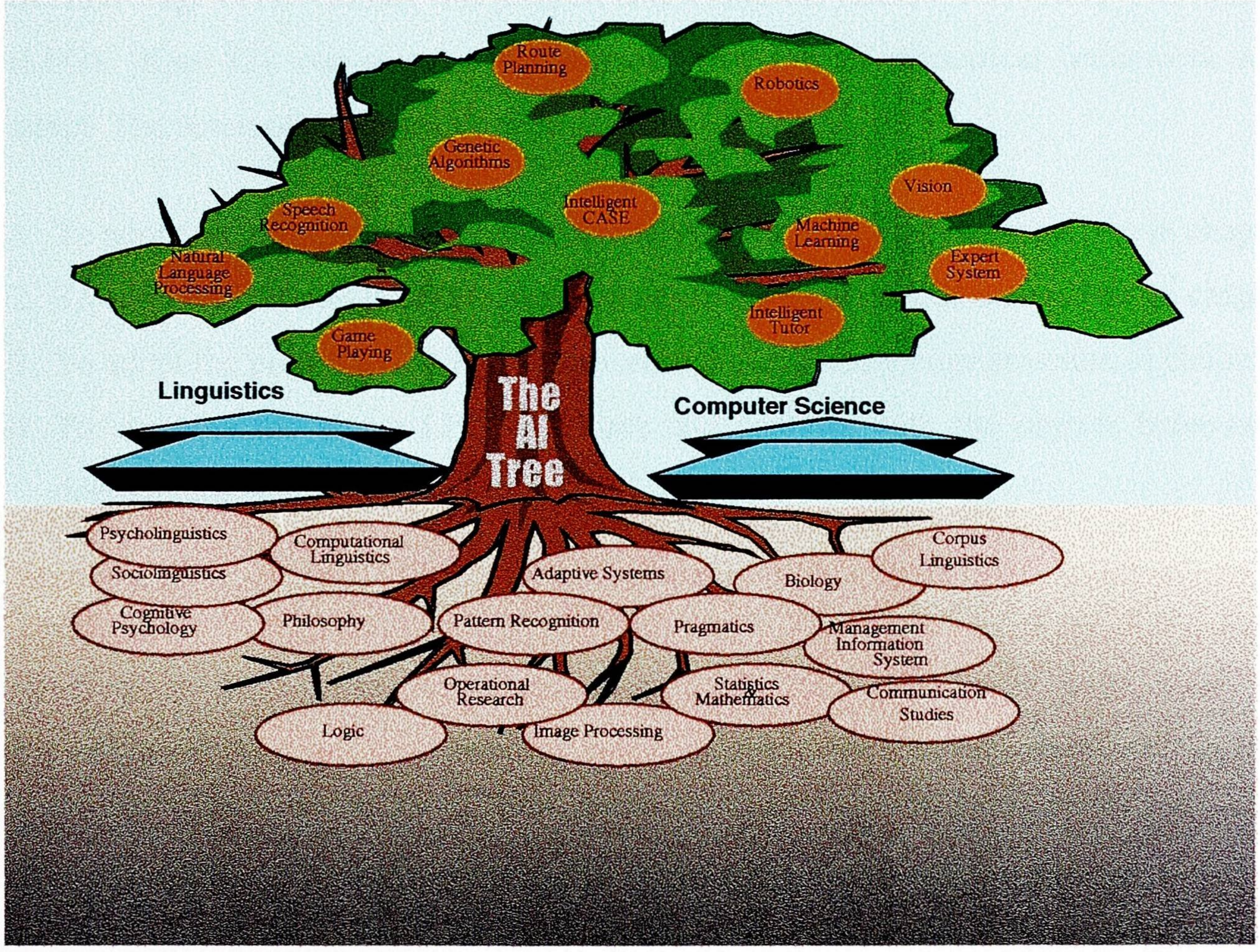


Figure A.1 The Origin of Computational Linguistics within the AI tree

Work undertaken under the umbrella of Linguistics and Computer Science disciplines combined has led to the construction of new working systems with various applications, such as:

- Spelling Systems such as Microsoft Word for Arabic 97, [URL 06];
- Speech Recognition Systems such as Natural Speck Software 97;
- Machine Translation such as Application Technology, [URL 04];

- Interfaces with databases such as CHAT [Patr-93], [URL 08].

1.2 The Problem of Natural Language Processing

Within the field of NLP, it is not possible to simply take an existing formalism or a theory, which works, for another language and apply it to say Arabic. Furthermore, to our knowledge, there are limited frameworks in Arabic linguistics, which cannot serve as a theory for the computational understanding of Arabic interrogatives. A number of aspects of Arabic syntax work have been dealt with, within the framework of the purely linguistic Government and Binding (GB) Theory[1], but with no computational linguistic approach. This forced researchers in this field to use their native knowledge and competence in Arabic and compare this with other languages such as English, which has many sources already available to it. In the first stage of developing this project, and within the Lexical-Functional Grammar Theory (LFG), [Kapl-82], our research focused on what sort of theoretical framework we should adopt. The result of this research was:

1. To adopt the proposed LFG syntactic analysis of the Arabic interrogatives in chapter four.
2. To adopt the proposed LFG semantic analysis of the Arabic interrogatives in chapter five.
3. To adopt the proposed LFG common-sense domain knowledge structure in chapter six.
4. To combine the above LFG structures and analysis with the objects behaviour of the domain model in chapter seven.

These four different approaches, combined with the Arabic language, form our unique integrated model; as it seems that they lend themselves to application to the Arabic language. For instance, Arabic tends towards frame structure by the very nature of its cases and agreement system. For example, within the Functional Structure of the LFG theory, verbs can be presented with their subjects/objects according to their gender, masculine verbs have to agree with masculine subject/object, whereas feminine verbs have to agree with their feminine subject/object. However, there are some exceptions that have to follow certain rules.

1.3 Computational Aspects of Natural Language Processing

[1] In 20th-century research into syntax has been contributed to Chomsky [Chom-81]. Chomsky is pivotal, whether his successive theories are adopted, or alternatives formulated in reaction to his emphasis on syntax in linguistics. Government and Binding theory has been proposed mainly for the syntactic analysis of languages. The theory provides an analysis of a given sentence and shows which category such as Noun Phrase governs other categories such as Adjective.

Computational Linguistic Approach

When undertaking development of a NL system, it is advisable that this system should be referred to a theoretical framework. If a particular system is set against a theoretical framework, it becomes possible to show explicitly how it is linked with other work in related fields. Of all Computational Linguistics theories known to us, the Lexical-Functional Grammar Theory (LFG) has been selected to fulfil our purposes. More justification details are in chapter four.

Unlike the GB theory, the LFG theory is designed to be computationally tractable, and in this respect, the theory is of interest to computer scientists. The original design of the theory has been developed with the intention of serving as a grammatical basis for a precise computationally and psychologically related model of human languages [Bres-85]. Another distinguishing feature of the LFG theory is that it applies different level structure to the constituent level as well as to the functional level. This can be applied to Arabic to describe the Arabic word orders in a satisfactory way. It is capable of expressing the properties of Arabic word orders and their phrasal structure, since it provides the Constituent Structure in which the order of the Noun-Phrase (NP) and Verb-Phrase (VP) constituents can be expressed (relatively) freely, and the Functional Structure, which can express the predicate-argument and agreement relations between the constituents in any word order of the Arabic language.

For Arabic Interrogatives, in our view, the best way of ensuring the grammar of an interrogative is to refer to a Computational Linguistic theory which can analyse and accommodate these grammatical rules within the lexicon. If such a theory were to be employed, the grammatical rules of a suggested system such as Chat-80 [Warr-82] would be given more linguistic solidity.

1.4 Motivations and Contribution of the Research

Natural Language Systems (NLS) are still in the early stages of their full development and are not yet widely known or fully accepted. Users of the interactive NLSs, very seldom type long complicated sentences of the kind that it's found in literary works. Given the Computational Linguistic approach and NLP, to our knowledge, there is nothing specific or oriented in Arabic Computational Linguistics, which may serve as a theory for the computational understanding of Arabic system [Yama-94b]. This has influenced our research decision to use a number of theoretical aspects within the AI field and Computational Linguistics and justify these theories and put them forward as requirements to achieve an ideal NLS which is suitable for Arabic. Furthermore, an approach or theory, which can handle English phenomena, will not necessarily handle the Arabic equivalent, as they differ in their structure and word orders.

Therefore, the main aim of this research is to help the user interact with the machine. The major challenge in designing a NLS, is to enable a computer to understand human languages. Conversation between humans is a powerful communication tool, and requires the full intelligence and world knowledge of the human participants. This implies that a computer able to interact, not only in NL but also in a natural way, offers unlimited scope for NLP.

Research Methodology

The problem domain was critically investigated with respect to related work. The theoretical work has been fully described in chapter four, five, six, and seven. In order to prove this theoretical work; it was necessary to construct a prototype, where several newspaper stories have been successfully queried. The building of a prototype helped to realise the specific goal and provided useful feedback into the research investigation.

The notation employed in this thesis for the representation of objects is that of Ignizio [Igni-91]. The Kappa Rule Based System with its presentation tools and the Object Management Workbench (OMW) have been used for implementation purposes.

1.5 The Main Theoretical Issues of the Thesis

Most Question-Answering Systems (QAS) lack linguistic analysis of the language they serve. Traditionally, QASs relied on NLP techniques instead of adopting a proper linguistic theory to analyse their data. In [Yama-94a], we have argued for the use of a linguistic theory as a main component of Natural Language Question-Answering Systems (NLQAS). Subsequently, this gives these QASs and our own project more linguistic substance, thus enhancing our understanding of the theoretical issues of NLs, whereby different levels of linguistic categories for possible answers can be analysed. In order to analyse these interrogatives, we have adopted the Lexical-Functional Grammar, [URL 10].

The LFG treatment of co-ordination deals with constituents such as NP subjects and objects, which can be distributed over predicates using linguistic rules. However, the distribution of a verb in a conjunction construction has not received formal treatment in LFG. In Arabic, as our traffic accident domain indicates, it is possible for co-ordinate interrogatives of both word orders to share a verb. The thesis focuses on three main theoretical issues:

1. Syntactic Analysis of the Interrogative;
2. Semantic Analysis of the Interrogative;
3. Common-sense Domain Knowledge of the Interrogative.

1.5.1 Syntactic Analysis of the Interrogative

The Arabic interrogative has been syntactically analysed by the adoption of a computational linguistic theory, namely, the LFG theory, to linguistically analyse the constituents of the interrogative with a view to gaining a broader understanding of the linguistic structure of its constituents. This theory has been partially extended in order to accommodate the interrogative verbal gapping phenomena of Arabic co-ordination in QASs, thus giving these interrogatives more linguistic substance. More details are given in chapter four.

The proposed prototype produces complete Functional-Structure (F-Structure) for the Interrogative including syntactic features such as Feminine/Masculine and their agreements in order to show the completeness of the F-Structure. This is important for the Arabic syntactic agreements. The constituents of the F-Structure contents come from the parsed interrogative words stored in the Lexicon. Each constituent appearing in the F-Structure has a category; each category has a category name and a list of feature values.

1.5.2 Semantic Analysis of the Interrogative

The second main theoretical issue deals with the semantics of the interrogative. It has become apparent during our research that the semantics for the declarative is not the same as the semantics for the interrogative (cf. [Groe-90], and [Engd-86]). Hence the proposal of a new approach of semantic for the interrogative. It appears that the declarative type theory of Montague's Semantics [Mont-74] as outlined and exemplified in [Dowt-81], can be extended to form a new Semantic Structure within the LFG theory specifically for the interrogative.

This, in our view, should form a suitable basis for a general semantics algorithm for the interrogative, and subsequently for a theoretical interrogative approach which can be implemented computationally in a QAS. A full explanation of these interrogative types and their rules is given in chapter five.

1.5.3 Common-sense Domain Knowledge of the Interrogative

The phrase 'Common-sense Domain Knowledge' is given to the process of understanding the behaviour of objects within the Object Model in a specific domain. Chapter six gives a formal presentation and description of the newly formed Common-sense Domain Knowledge (K-Structure) which complement the S-Structure. The K-Structure is needed to eliminate NL ambiguity that the S-Structure cannot deduce. The presentation is a NL Domain as an Object Model based on the LFG theory. In addition, the presentation proposes new theoretical methodology to model NL text as an application Domain, where the mapping of the LFG analysis to the Object database, modelling and the structure of the LFG promote a better understanding of the interrogative. The basic principle in structuring such knowledge is based on an extension of the S-Structure of the LFG to be expressed in a more formal presentation of knowledge structures that are organised into a hierarchy of concepts, categories, constraints, and the knowledge itself. During the inheritance of knowledge between concepts, sets of features have to be common between these categories, in other words, categories dealing with feminine verbs are grouped together so that they have common syntactic features. As for the agreement between these concepts, rules have to be enforced to determine the Arabic agreement system between these concepts.

1.6 The Structure of the Thesis

Chapter one has outlined in brief the main theoretical issues concerning the project and the thesis as a whole. The main observations, analysis, discussions and suggestions are set out in the remaining chapters.

Chapter two gives a general overview of the state of the art of Natural Language Systems, with particular emphasis on their understanding techniques and outlines current theoretical proposals which seek to improve the computational linguistic performance of some of these systems. The chapter compares understanding techniques used by these systems with the techniques embodied in our own project.

Chapter three describes the relevant features of Arabic syntax with the view of creating lexical entries for computational purposes. It aims to give the reader a basic understanding of the complexity of the Arabic language.

Chapter four presents a concise grammatical and syntactical description of Arabic interrogatives in both word orders within the framework of the LFG theory. It constitutes a sample grammar for Arabic interrogatives, which is computationally workable and capable of incorporating the semantic features in chapter five. The chapter also proposes a new notation to deal with verbal gapping within the constraints of an extended LFG framework and discusses the facts about Long-Distance Dependencies (LDD) and verbal gapping phenomena in Arabic.

Chapter five analyses the semantics of the constituents of Arabic interrogative types and argues for and establishes the basis for a concise Semantic for the Interrogative. It also builds on the analysis of Arabic co-ordinate interrogatives, and extends this semantic structure to incorporate semantic presentation for the prototype. The end result is an interrogative formula of expression for each of the interrogatives analysed in chapters three and four.

Chapter six builds on the previous analysis and extends the LFG theory by adding Common-sense Domain Knowledge Structure, namely K-Structure. This will work hand-in-hand with the semantic structure in order to answer any queries related to the domain, which can be deduced by applying the Common-sense Domain Knowledge Rules.

Chapter seven contains the design architecture of all the components of the overall project and intends to show what can be computationally designed according to the theories proposed in chapters four, five, and six.

Chapter eight covers the implementation of prototype QAS. It also addresses the various theoretical approaches, which can be adopted for implementation purposes of the project, and presents computational grammar rules for Arabic interrogatives. It then proceeds to outline the implementation of Arabic agreement system for both word orders. Chapter nine brings the thesis to its conclusion and outlines the potential for future extension and exploitation.

Chapter 2

Overview of Natural Language Systems

2.1 Introduction

For the layperson, the prime obstacle in using computers has been the need to either learn a special language for communicating with the computer or communicate via a machine-based assistant system. To help the user interact with the computer, the major challenge in designing a Natural Language System (NLS), [URL 02] and [URL 09], is to enable a computer to understand and use language as a human would. Conversation between humans is a powerful communication tool, and requires the full intelligence and world knowledge of the human participants. This required that a computer to interact, not only in Natural Language (NL) but also in a natural way.

NL is a primary means of thinking, learning and communicating. No other approach is as general and flexible (at least for the time being). A NL front-end to a data base or knowledge base system must have several crucial capabilities in order to be judged adequate by its end users. These capabilities must extend to the:

- linguistic coverage and understanding
- speed of response
- level of performance
- well-defined answer.

In the key area of successful communication, NLs have been compared unfavourably with artificial command languages e.g., the UNIX operating system. Furthermore, no standard functionalities have been defined for NL yet [Tenn-86].

This chapter is divided into four sections. Section two presents overview of existing NLS; section three puts forward the proposed approach of this work; with a final summary in section four.

2.2 Overview of Existing Natural Language Systems

This section reviews the understanding approaches used by NLSs. Naturally, a NLS takes a NL sentence or an interrogative as its input and produces some sort of understanding as a formal language presentation for its output. Naturally, a NLS is composed of a language processor and a translator in order to extract NL understanding. This section examines the language processors, and translators, centring on the issues of how they obtain NL understanding, and how they represent sentence meaning.

Taking NLSs as presented in the current literature, we have found different approaches and theories that have been adopted for NL understanding of these systems. This section classifies these systems into four categories. These can be broadly classified as follows:

1. Systems that possess no understanding.
2. Systems that possess basic or restricted understanding.
3. Systems that possess some skilful understanding for general domain.
4. Systems that possess comprehensive understanding in restricted domain.

Although the four classifications of these systems share the same common denominator, namely, the use of NL as front-end, they are differing in their ways of interpreting NL to their databases. The intention of the following overview is to provide the reader with knowledge about the reasoning power behind these systems in terms of understanding strategy rather than ranking these systems. Let us examine these systems with respect to our classifications.

2.2.1 Systems that Possess No Understanding

Many systems have fallen into this category. In the past two decades or so, many researchers have focused their attentions on the use of AI techniques. AI techniques can be of the use of Logic, pattern matching, Search and Key word Recognition, Frame presentation, word-by-word syntactic and semantic analysis. Other researchers paid attention to the syntactic and semantic analysis in the hope that some NL understanding may be drawn out of this analysis. While ambiguities still remain a problem to eliminate, other AI approaches have been put forward such as scanning the sentence or the query for key word recognition. The missing point here is that capturing the syntax and the semantic features of each word will not necessarily give the whole of the sentence meaning.

Due to the absence of linguistic understanding of NL such as knowing the subject from the object in a given sentence, AI techniques were the dominant approach used. Although the main aim of these systems is to eliminate or at least minimise NL ambiguities, no attempt has been made to boost the language process that can generate a proper and correct interpretation of NL. The main focus was on processing the NL rather than on understanding NL.

These systems were relaying on AI techniques like the syntactic analysis followed by semantic analysis. Many systems adopt this technique which performs the syntactic analysis first using a general syntax grammar. This results in a parse tree for linguistic categories, whereby domain semantics are applied to select these categories which have passed the parsing analysis. This technique has resulted in only word-by-word meaning that obviously resulted in no understanding.

Other techniques are used like semantic grammar. Semantic grammar means that words are not classified as nouns or verbs; they are grouped as to their relation of semantic meaning. The sentence can be interpreted as an intermediate structure by using a domain-specific that embodies syntax and semantics. This technique has given a domain word-by-word meaning with no understanding. Let us review some of these systems.

The LUNAR System

LUNAR, ([Wood-72], [Wood-73]) was originally developed with support from NASA for a geological application. The technique that the LUNAR system adopted was to perform the syntactic analysis using a general syntax grammar, thus producing a parse tree recognising individual words. It then applied semantics analysis to select the semantic meaning for the words that had passed the syntactic analysis, where the input sentence was parsed using the Augmented Transition Network (ATN) parser and the result was translated into Logic. The semantic interpretation component used pattern rewrite action strategy rules to transform the syntactic representation into semantic representation. Thus, these can retrieve information or store information from/to the database. The database was organised and presented in a rather conventional way, i.e. files that contained records, which themselves contained fields.

The ROBOT System

This approach of syntactic analysis and semantic analysis has opened the door to other systems using the same strategy; a typical one is ROBOT, [Harr-80]. However, ROBOT also consists of a language processor, which itself consists of three main modules - Parse, Weed and Decide. The Parse module is the syntactic analyser which generates the interpretation(s) of individual words in a given sentence, the Weed generates the semantic analysis, and Decide is the module used for selecting the actual interpretation from the set of syntactically valid interpretations. The technique used is the generation of a different interpretation for each meaning of the given word. Meanwhile, two separate modules within the system, that is Weed and Decide, perform the semantic analysis. The Weed module isolates semantically valid interpretations produced by the actual parser. This is achieved by interacting with the database to determine which interpretations are semantically correct with respect to the database. At the end, the Decide module selects one interpretation, if an ambiguity occurs the Decide module asks the user to select the correct interpretation. This is achieved by determining the most likely of a set of given interpretations. This module may query the user to confirm the intended meaning if it considers this necessary. Although words have been recognised, the semantic analysis did not manage to get the meaning of the sentence as a whole.

By using this technique, the system generates a very long set of interpretations. The interpretations have to be checked individually with the help of the user in order to single out the right one, which is obviously time-consuming.

The LADDER System.

The LADDER system [Rich-84] is the application of LIFER [Hend-81] to a database dealing with naval vessels. The main feature of this system is the use of semantic grammar containing the syntax and the semantic features. In a semantic grammar, semantic features are phrased in terms of domain concepts, rather than linguistically syntactic categories. For example, a sentence should linguistically consist of noun phrase and verb phrase, i.e., S → NP & VP. In the LADDER system, this linguistic rule is replaced by the domain naval vessels rule as:

Query Name → Ship Name & Ship Location. The understanding of the LADDER system is based on words of the domain. Linguistically, LADDER has no even basic linguistic categories let alone linguistic understanding.

The TEAM System

TEAM is an experiment in the design of transportable NL Interface (NLI) [Gros-87]. TEAM stands for 'Transportable English database Access Medium' and interacts with two types of users: database expert, and the end-user. It was constructed to test the feasibility of building a NL system that could be adopted to interface with new databases operated by users who were not experts in NLP.

TEAM uses logical forms, based on First Order Logic, which is extended by an intensional and higher-order operator. Word Recognition in TEAM is done by semantic translation. Each grammar rule has an associated function called translator. The translator in TEAM has no overall understanding of its sentence. It also lacks confidence in building logical form for its query presentation, and this has resulted in TEAM having to ask the user for confirmation in the building up of its logical translation of a query.

2.2.2 Systems that Possess Basic or Restricted Understanding

It was not until the early 1980s that researchers started looking for other alternatives in order to raise the standard in understanding NL. During the 1980s some new ideas emerged. These ideas came with the aim of conducting computational linguistic theories that can process and understand NL.

To boost these ideas, in 1982 Bresnan and Kaplan (a computer scientist and a linguist) developed a computational linguistic theory called the Lexical-Functional Grammar theory (LFG) [URL 10]. The theory is based on the idea that: in order to process and understand NL as a human would, a framework must be developed so that it can serve both mechanisms. (The reader can find the description of this theory in chapter four). During the 1980s, other computational linguistic theories have emerged such as Generalised Phrase Structure Grammar (GPSG), and Head-Driven Phrase Structure Grammar (HPSG). During their early stage of development, these theories helped defining the basic understanding of NLs. Theories like the LFG are still being updated with new ideas from other languages up to the time of writing this thesis.

Researchers find it fascinating that the more they subject NLs to such theories, the more understanding they find about them. For example, when researchers developed these theories, English was the first language to experiment with. Other difficult languages such as Arabic, have different word orders than English and have different syntactic agreements, (see chapter three). This requires that these theories be updated accordingly. This has been demonstrated during the development of the proposed prototype.

So far, these theories have helped in distinguishing between Verbs, their Subjects, and their Objects. This has given more information about the structure of sentences and the correct position of their words. As a consequence, this has added more meaning to words. The structure has also helped in defining and correcting agreements (Masculine or Feminine agreements) between verbs and their subjects and their objects in languages like Arabic.

Therefore, this linguistic analysis has given us some basic understanding that is restricted to linguistic categories in NL. For instance, in a Question-Answering System like Mehdi's system [Mehd-86], he based his assumption on the following. If both the verb and the subject exist in a query and the object is missing, then there is a high probability that the answer could be the object stored somewhere in the database. However, this assumption is not correct as the missing object may be stored in a different linguistic category in the database such as in the form of a sentence that contains a verb as well as a subject. Consider also the following systems.

The IRENA System

The IRENA system, stands for Information Retrieval Engine based on Natural language Analysis [Aram-96]. The purpose of this system was developed to study the improvement of precision and recall in document retrieval. The queries presented to the system consist of Noun Phrases (NPs) only, and the syntactic analysis recognises keywords from those NPs. Once a keyword is extracted from the NP, this in turn, lists all related documents to this NP. As for structural ambiguity, it has been claimed to be solved by continued searching for all NPs until an answer is found. Certainly, the approach of this system did not eliminate ambiguity for the reason that if an answer is found, the user then decides on which of the retrieved documents are related to his request.

It is not clear how the understanding mechanism is exactly done, the retrieval strategy used is based on measuring the 'closeness' of one NP in the query to another in the document stored in the database. In addition to this, the system strategy also uses a hypothesis to locate the right document and subsequently the right answer.

Although the system is still in the experimental stages, it seems that it is limited in its vocabulary and also limited in its understanding. Furthermore, the system uses AI techniques rather than computational linguistic theories for understanding. Therefore the understanding of this system is based on unification of grammar rules rather than on defining the subject, object, or verb in a given query instead of using just NPs. However, the understanding of this system is too restricted for the reason that if we are to apply a language such as Arabic where the verb, not the NP, is the main source of semantic, the system will certainly fail. The reason for that is simply that Arabic verbs are important for deciding the overall meaning of a sentence. Furthermore, applying NLP techniques alone fail to resolve ambiguities in NL.

The CLARIT System

The CLARIT system is similar to the IRENA system in its approach. The CLARIT system Evans, [Evan-96], has contributed more reasoning power to the use of NP by creating indexing phrases for information retrieval. This has been noted when they based this technique on corpus statistical and NP linguistic heuristics. Basically, this method of understanding used indexing to match the user query keywords with the stored documents. The idea here is to increase the precision of retrieving the correct stored document. It starts by calculating what can be match between stored documents and the query, using (a) frequency patterns, (b) associations, and (c) individual term weights of each NP.

The Parser used employs bottom-up associated-based parsing [Gazd-89]. The strategy of these techniques is to group words together, based on association in order to gain more understanding. Word-pairs are given an association score according to the lexical rules: the score provide evidence for groupings in the parsing process. The results of these experiments show an improvement in recall and precision by up to 81.6% than the previous similar studies.

To summarise, the two systems, namely, IRENA and CLARIT are put forward mainly for improved precision in recalling stored documents. They seem to be drifting from the original problem of this approach as a whole. Therefore, CLARIT system has inherited the same problem as the IRENA system. The understanding of the two systems is restricted as they use NP linguistic category only.

The ALFRESCO System

ALFRESCO was developed by a team led by Olivier Stock, in Trento, Italy [Zanc-97] [Stoc-95], and was embodied in prototypes for *Multimodel* interaction for information access. In ALFRESCO, application integrates NLP and imagery: it is a tutor Fourteenth Century Italian frescoes, each painting being accompanied by explanation text.

Certainly, this approach, and subsequently the prototype have opened a new way for retrieving information in a specific domain and in understanding. The method is called *multimodel* interaction that is when a language and image-based interaction is integrated with hyper-textual capabilities. The aim of this system is not only providing information about paintings, but also of promoting other masterpieces that may attract the user. The user can interact with the system by a NL query combined with the use of a touch screen. In other words, the input is a combination of linguistic deictic references with pointing to image displayed on a touch screen.

The output is in the form of images and generated text, and the final decision is left to the user to choose between these. An example of this, the user asks a question like:

'Who is this ↗ person'

The arrow ↗ is a touch screen image. The system then answers the question by displaying the name of that person. The method used is based on speech acts, which have four conditions of user performance:

1. Propositional content condition,
2. Preparatory preconditions,
3. Conditions on sincerity,
4. Essential conditions.

In addition, the multimodel NL dialogue has two acts:

1. Non-linguistic acts; which have no interpretation, and it is the user's intention,
2. Linguistic acts; which have an interpretation, and cohesion i.e. user confirmation.

To summarise this approach and the prototype, although the latter is not a fully NL interface, it is a powerful concept in the field of Human-Computer Interaction. If ambiguity is found, this can partially be solved by the use of graphical representation, i.e., the image-screen touch. Any other ambiguities found in the query can be solved by the user's co-operation. The system obtains the understanding by referring any ambiguity to the user and that can be considered by our classification as a restricted understanding.

The CHAT System

The CHAT system, [URL 08], Conversational Hypertext Access Technology, is a computer programme developed by Communication Canada, [Patr-93]. The claim is that CHAT provides a NL interface that allows users to ask English questions about AIDS disease in Canada and receive answers.

CHAT uses a template approach. During the dialogue, each new question is compared with a series of templates formed from previous questions that have been recorded. The core algorithm is more like pattern recognition rather than linguistic understanding.

The system analyses the user's questions by comparing them to a list of templates stored in the information base. These templates are words, phrases, or question fragments prepared by the author of the information base. The templates are associated with hypertext 'links', and if a match is found, a link is followed to a new paragraph of information. The system, then, evaluates all the templates that match an input sentence and computes the best path through the information base that will account for the maximum number of characters in the question.

Although the system has a restricted domain, by using templates matching, it seems that this system has inherited the problem of pattern-matching technique, which has no understanding. It also not clear from its description how the system behaves when ambiguities occur. It seems that the trick used to eliminate ambiguities is by displaying a description of other related topics within the same template so that the user can find what ever they are looking for in that template.

The START System

The aim of this project, [Katz-98], is to construct a query knowledge based using English language. The understanding of START is based on *Annotations*, [URL 07]. Annotations are tokenised collections of NL sentences and phrases that describe the content of various information segments. START analyses these annotations in the same fashion as any other sentence, i.e. syntax and semantic, but in addition to creating the required representational structures, the system also produces special pointers from these representational structures to the information segments summarised by the annotation.

The basic understanding of this system is just another way of repeating template-matching mechanisms. It has restricted overall understanding of interrogatives that represent the whole meaning. As it has been indicated by Katz, [Katz-98], to solve problems like ambiguities of full-text and queries within START, is a future aim of START, which requires more research and developments.

The Yes/No Question System

The Yes/No question, [Schu-92], is the only system differing from the above, which performs a rather comprehensive computational linguistics analysis. It uses the computational linguistic theory, namely, the LFG theory to analyse its queries. Unfortunately, the system is limited to Yes/No queries types.

The restricted understanding of this system is based on the use of *situation logic* for natural language proposed by Fenstad, [Fens-87]. However, logic alone has proven inefficient at eliminating ambiguities in NLs [Lena-95] and [Saba-93]. The common-sense domain knowledge, which proved to be the right way in understanding NLs, can be better employed here. Within the LFG theory, the emphasis is on the semantic and pragmatic presentation, but again this presentation is limited to Yes/No question types and their negation. However, because it is not clear whether this approach has been implemented or not, the system as a whole has linguistic understanding since the LFG theory is used to analyse data, but, it seems that this understanding is limited.

Arabic Natural Language Systems

In this section, we have classified the following Arabic systems under the restricted understanding class. The reason for that is most of these systems have inherited the above weaknesses of the AI techniques such as Logic, pattern matching, Search and Key word Recognition, Frame presentation, and word-by-word syntactic and semantic analysis.

Most of NLS for Arabic consist in fact in a computational analysis being focused on a morphological analysis or machine translation. Up to the time of writing this thesis, there is no known product for Arabic NLS. Most of recent efforts were concentrating on importing and Arabizing packages, which had mainly been build for English. However, there have been a number of attempts to build NLS for Arabic.

The first one was developed by Mehdi, [Mehd-86], with no Arabic computational linguistic understanding such as the LFG theory. The system uses Definite Clause Grammars (DCGs) proposed by Warren, [Warr-82]. This type of grammar uses a logic approach in solving NL, which lacks the ability of solving NL ambiguities [Lena-95].

In comparison to the proposed approach of this thesis, Mehdi, [Mehd-86], suggested that: *if we were to rebuild this system in order to get a better understanding, we would consider selecting computational linguistic theories such as of the LFG theory to analyse our data.* Chapter four and seven of this thesis have taken this suggestion on board by adopting the LFG theory in order to gain linguistic understanding.

Another system was proposed by Al-Muhtaseb, [AlMu-88]. The system creates knowledge based by dividing the acquired text into two categories, namely, subject and action. The subjects are related to classes represented in a semantic network fashion. These relations are presented in predicate and production rules. The actions are presented according to their semantic features. The system has a limited linguistic understanding of Arabic subject/object, it cannot distinguish between the subject action or an object action, which obviously leads to ambiguities and subsequently, can lead to the wrong answer.

More attempts were put forward by Al-Safran, [Safr-93]. The system captures more semantic features in order to eliminate ambiguities. This has been developed by having an object oriented hierarchy of frames that classifies Arabic words into their categories. This approach has successfully created meaningful representation for each category. Other slots have been devoted to capture syntactic associations between words within the sentence, synonyms, antonyms, etc. But the approach is too restricted to capture the overall understanding of a sentence or a query, for the reason that there is no mechanism that can be used to capture or deduce knowledge related to words in a sentence. Moreover, unlike the approach of this thesis, Al-Safran's system has no mechanisms such as linguistic rules or domain rules that can describe the functions of the words in the nominated domain.

More detail descriptions of other Arabic systems such as morphological understanding or syntactic analysis can be found in [AlAa-94]. A more comprehensive overview of other previous systems and their descriptions can also be found in [Gros-86].

2.2.3 Systems that Possess Some Skilful Understanding for General Domain

Although the above computational linguistic theories, in section 2.2.2 have added some basic understanding, during the mid 1980s researchers were looking for ways to even deepen this understanding. What they had in mind was Common-sense Knowledge. Lenat [Lena-95] has been working for over a decade on a large-scale project, intended to capture Encyclopaedia (CYC) Common-sense [URL 01]. The approach is controversial (see [Smit-91] and [Lena-91]).

As an alternative to using AI techniques alone, and with the aid of the above linguistic theories, a radical approach called common-sense knowledge was applied to machine or computer in order to understand NLs and subsequently to communicate with humans. The common-sense approach is based on capturing world knowledge as a human would. The way this knowledge can be captured is based on the approach of Frame Representation, *thematic-role frames,* Winston, [Wins-92]. This technique can describe actions conveyed by the verbs and nouns appearing in typical sentences. This approach will not only apply semantic analysis to text, but also add common-sense knowledge in terms of rules of all possible associations between objects concerned (e.g. Verbs, Nouns, etc.).

The approach has opened the way for some challenging projects and has discovered a new way towards understanding NL. In addition to the above basic understanding techniques, the new approach has helped forming new research ideas to build new applications. These applications, as the CYC project embarked on, can be using common-sense on Machine Learning, Machine Translating, and serving the WWW. This has also boost the understanding mechanism of NL, which eliminates some ambiguities by applying common-sense knowledge rules. The following CYC project is best at describing this approach.

The CYC Project

The Encyclopaedia (CYC) project Lenat, [Lena-95], is perhaps the first project that adopted almost skilful understanding of NL, [URL 01]. Since 1984, a huge effort has gone into building the CYC project, which is a universal schema of concepts spanning human reality. Approximately 10^6 of common-sense axioms have been handcrafted for and entered into CYC knowledge base and 10^5 more have been inferred and cached by CYC. One can think of this project as a multi-domain project that spans all everyday objects and actions. The overall aim of this project is to understand NLs by building disembodied meaning to every domain human have.

With the help of AI techniques, the aim of CYC is to apply the common-sense knowledge to machine in order to understand NLs. The CYC project approach is based on capturing knowledge, this knowledge can be used not only in answering questions, but also in Machine Learning. Another application of CYC is to serve the WWW in different ways such as knowledge search for queries in the Internet. The CYC technology can examine retrieved data, recognised inconsistencies, contradictions with specific data from other sources, and violation of common-sense within the commercial databases. Another major application of CYC is to help in Speech Recognition application.

The use of techniques such as Frame Representation with thematic-role frames has made CYC capable of serving these applications although with some difficulties. As it has been described, the project is too ambitious as it requires a large number of researchers and much time. The project has some progress in some application areas, but large areas of research are still under development.

However, the limitations of the CYC project are caused by profound difficulty of modelling human common-sense - something which is by no means fully understood. More criticisms of the CYC project can be found in [Smit-91] and [Lena-91].

2.2.4. Systems that Possess Comprehensive Understanding in Restricted Domain

This class is where we make our contribution towards the ideal NLS. Skilful understanding of NL can be viewed in three dimensions:

- The Use of Linguistic knowledge from a well-founded theory
- The Use of Linguistic and Domain-Specific Rules
- The Use of Common-sense domain knowledge.

These three dimensions combined can add substance and subsequently understanding that make NLS more intelligence in three ways: (i) how, when, and where linguistic rules can be applied; (ii) to which domain these linguistic rules can be applied; and (iii) what sort of common-sense domain knowledge rules should be applied. These three dimensions have been described in section 2.3.

The approach that we have put forward in our proposal is to combine the above linguistic and domain rules so that ambiguities can be controlled in a given domain. Furthermore, this combination of rules has been amalgamated with the use of Object Modelling so that common-sense domain knowledge can be applied. This can be considered a skilful understanding of NL which resolved most, if not all, common ambiguities found in NL.

The proposed prototype of this thesis focuses on issues related to the interrogative common-sense domain knowledge structure within the framework of the LFG theory. The interrogative domain knowledge presentation has enhanced the understanding and subsequently the intelligence of the proposed prototype QAS.

The thesis presents a prototype model that has been conducted by developing a general Object Model for the traffic accident domain. This model has been used to capture the functional behaviour of objects, the related verbal interactions and their linguistic associations in a given query. The key novelty of this prototype is the combination of objects behaviour in the Domain Model with the LFG Structures rules, which resolves most, if not all, of the common ambiguities found in NL. The following section describes the prototype in more details focusing the attention on the uniqueness of this prototype.

2.3 The Proposed Approach of this Work

A review of the current literature on NLSs and their approaches prompted us to propose moving towards a model NLS. We believe that our proposed approach not only provides the strongest infrastructure of the suggested ideal NLS by Lenat, [Lena-95], but also has discovered new dimension for the understanding of NLSs. In comparison to the above approaches and techniques, our new dimension came as a result of the following innovations:

2.3.1 The Common-sense Knowledge

Background: During the 1950s, the aim of Artificial Intelligence (AI) researchers was to create a computer that could think. This has sparked the ideas of applying AI techniques to solve problems such as NL ambiguities.

To date, AI techniques alone did not help much in eliminating NLs ambiguities simply because they use no knowledge to understand NLs [Lena-95]. As we have seen from the above systems, AI techniques are good in analysing individual words whether syntactically or semantically, but they have failed in obtaining the overall meaning i.e. applying world knowledge. For example, AI pattern-matching techniques have failed to solve NL understanding. Ever since the 1960s when Weizenbaum's well known question-answering project ELIZA, [Weiz-66], many projects which followed ELIZA during the 1970s have inherited many of its problems [Gros-86]. As from the 1980s, researchers started working on the Pragmatics of interaction and common-sense knowledge in order to boost the understanding of NL Lenat [Lena-95].

As it has been described, Lenat, [Lena-95] project, namely, the CYC is too ambitious as it requires huge resources (given that the project is in its 14th year of research, and some say it just cannot work [Smit-91]).

In our view that what the CYC project promised initially has not been achieved. Moreover, it acquired for more research than what initially planned. One of the main reasons is dealing with multi-domain. Hence, the learned lesson is to focus our research in a small domain, namely, traffic accident domain. In this respect, there is promise for successful integration within a future architecture in the context of telematics for health care and telematics for traffic management and emergencies, provided more investment is put into capturing common-sense knowledge about social situations and social behaviour. For our small domain, we used common-sense domain knowledge with an Object Model and applied this knowledge in order to understand NL. Therefore our claim is that common-sense domain knowledge is the key to solving NL ambiguities.

2.3.2 The Use of the Lexical-Functional Grammar Theory

In our view, and that of [Schu-92], the best way of ensuring linguistic coherence of interrogatives is to refer them to a computational linguistic theory, such as the LFG theory. Such a theory can analyse and accommodate syntactic and semantic rules. If such a theory were to be employed, most of the above systems would be given more linguistic solidity and understanding.

The LFG theory is designed to be computationally tractable, and in this respect, is of interest to computational linguists and computer scientists alike. The description of the LFG theory will be given in the introductory section of chapter four. Grammatically speaking, it is important that the interrogatives and their answers should be well formed according to the lexical rules of the language used to describe the query.

Within the LFG theory, and for the interrogatives in particular, there is a sub-theory dealing with the phenomena of the interrogative, this being the Long-Distance Dependencies theory [Schu-92]. It uses FOCUS (for wh-words) and Grammatical Functions for interrogative sentences in order to determine the category of the displaced element within the interrogative.

Chapter four discusses this theory in more details while chapter seven creates syntactic and semantic rules for understanding the interrogative before searching for an answer. Within these rules, the type: *FOCUS* treats this interrogative through a functional equation that maps the syntactic rules into semantic. There are different *FOCUS-to-Grammatical Function* relationship type categories. These Grammatical Function relationships are dependent on the languages used and their respective grammatical functions. An example of this is the FOCUS-to-Object-type category relationship, FOCUS-to-Subject-type category relationship and FOCUS-to-Complement subject-type category relationship as in the examples below:
↑ FOCUS = ↑ < target>: While target could be an: OBJECT as in *who did John see*, a SUBJECT as in *who saw Mary*, or a COMP SUBJECT as in *who does John think saw Mary*.

Chapter five extends the role of the LFG Semantic Structure (S-Structure) from an S-Structure for the declarative, into an S-Structure for the interrogative and also the Co-ordinated interrogative. This S-Structure can be used to form an interrogative formula, thus extracting answers from the database. Furthermore, this interrogative S-Structure will not only present the semantics of interrogatives in the S-Structure, but will also use this structure in order to formulate the shape of the expected answer from the database. The aim of using the LFG theory is to boost the linguistic understanding so that we can minimise ambiguities.

Most of the above systems lack the ability to handle NL's ambiguities. Most of these ambiguities came as a result of analysing individual words rather than sentences as a whole. This will require not only applying a theory like the LFG theory, but also applying common-sense as it has been suggested by [Lena-95].

2.3.3 Linguistic and Domain-Specific Rules

The other uniqueness of our approach is the combination of object behaviour in the Domain Model with the LFG syntactic structure rules and semantic structure rules. Since no other system used this approach before, the combination has opened the way for resolving most of the common ambiguities found in conventional NL systems. It also enhances the understanding ability of the proposed prototype.

In addition, and since most of the above systems use English language for their interface which requires one word order, namely the SVO, in Arabic language, the problem is complicated, mainly because of the complexity of the language and the word orders (as introduced in section 3.3). For instance, although the CLARIT system [Evan-96] is portable to accommodate other domains, it has some difficulties in accommodating two word orders languages. For example, if we take two word orders languages such as Arabic, the mechanisms and the strategy of CLARIT system will fail to cope with such phenomena, as well as the system proposed by [Mehd-86]. The parsing phase of this system has been set up using logical grammar based on Prolog. This technique has its drawbacks, [Gazd-89] since it's relying heavily on Prolog.

Finally, the correspondence between syntax and semantics does not generally give the overall true meaning of a given interrogative [Lena-95]. We believe that we still need an alternative to this, such as common-sense knowledge understanding between the constituents of an interrogative to capture the true meaning and subsequently the correct answer for the interrogative.

2.4 Summary

In this chapter, we have given an overview of NLSs and classified these systems into four classes according to their understanding of NL. Furthermore, we have argued for the use of a computational linguistic theory, namely, the LFG theory as a key component for the improvement of NL systems and their understanding. In addition, the chapter gives arguments for the use of a common-sense domain knowledge approach as an alternative to just artificial intelligence techniques such as pattern matching which has limited understanding. These enhancements can obviously be judged according to their theoretical merit and linguistic comprehension, but ultimately, the refinements can only be practically evaluated in a real situation by communicating with the end user. The enhanced features of the system's functionality of linguistic understanding common-sense domain knowledge should determine the system's overall understanding.

F-Word:

is the Feminine of the Predicate e.g. happen - feminine in Arabic[1] (وَقَعَت)

M-Word:

is the Masculine of the Predicate e.g. happen - masculine in Arabic (وَقَعَ.).

G-Functions:

are the Grammatical Functions of the Predicate e.g. Subject, Obl-Argument within the F-Structure.

Thematic-Object:

The Thematic-Object represents the name of the object from the object model. This is important, as the name of the object will determine which attribute(s) could provide the answer.

Syntax Features:

are the syntactic presentation of the predicates e.g. past tense.

Semantic Features:

are the semantic presentation of the predicate e.g. human, animal, things or relationship to the predicate

3. The Nouns Frame

The nouns are linguistically divided into various types. For the interrogatives, the nouns are subcategorised within the F-Structures as Subjects, and/or Objects, and Obl-Arguments. The organisation of the nouns in the lexicon behaves just like that of the verbs except there is no G-Function. Nouns are part of the Grammatical Functions of the predicates whether they are in the subject position or object position. However, if an interrogative occurs without the verb (this is possible in Arabic, e.g.: Who is in the garden? (من في الحديقة)), the F-Structure rules will detect this so that the subject becomes the main predicate of the interrogative. This will be demonstrated in the chosen application in Appendix C.

[1] Since this research (and subsequently the prototype) focuses on issues related to Question-Answering System, no attempt was made to deal with the morphological analysis of the interrogative, as it is outside the scope of this research.

Representation and Strategy

Each interrogative must have a particle and particle features from the lexicon, either Interrogative-WH, Interrogative-H, or Interrogative-Y/N, as they are illustrated in the Lexicon structure in Figure 7.3. In addition, each interrogative must have either a Predicate or Nouns, or both and their presentation from the lexicon in order to satisfy the interrogative arguments and subsequently the domain.

7.5 Linguistic and Domain Specific Rules

Section 6.5 has outlined the Object diagram, which contains all possible object names in a traffic domain. Section four of this chapter also outlined the lexicon and its linguistic associates with respect to the traffic domain. This section shows the relationships between these two in terms of linguistic and domain-specific rules. The type of particle and the domain-specific rule determine the required slot of the frame.

In order to produce F-Structure, S-Structure, and K-Structure (if needed) and subsequently the correct answer for each interrogative, certain specific rules must be drawn. These rules are grouped into sets, each set contains one type of interrogative (e.g. where (أين) and when (متى) sets). Questions, which have more than one interrogative type (e.g. when and where), will of course require two sets. Within these sets, we have imposed rule ordering through rule priority.

The Search Strategy

Although the interrogatives are different in their structure they will, during processing, follow the same behaviour pattern in order to get the answers. The search strategy implied in the following rules is forward chaining. That is, the process begins by taking the interrogative particle, and verb or noun or both, with their lexicon presentation as indicated in Figure 7.4 and 7.5. These presentations, along with their constraints, serve to filter out the majority of the potential alternatives e.g. Location as a possible answer, and thus ultimately arrive at the right object name(s) and subsequently at the right attribute name and value. In this respect, there are two types of rules: Independent linguistics rules, and Domain specific rules.

1. The linguistics rules are domain independent rules, in other words, if the domain changed from traffic accident to, say weather forecast, this should not requires the linguistics rules to be changed. These rules are purely dependent on the grammar of the language chosen.
2. Domain specific rules are dependent on the actual domain, which in our case, is a traffic domain model.

Let us look at these two types in more details.

7.5.1 Linguistic Rules

In order to produce and present a complete F-Structure and S-Structure, certain independent interrogative linguistics rules must be drawn. The following sets illustrate the design of such rules.

The *Where* (أين) Set

1. where is the accident? (أين الحادث)
2. where was the accident? (أين كان الحادث)
3. where did the accident happen? (أين وقع الحادث)

F-Structure Presentation

The above third interrogative has been presented in the F-Structure Figure 7.6. It shows the missing Obl-Arg i.e. the object of the accident (مفعول به - ظرف مكان). The Interrogative F-Structure has been built by using the LFG sub-categorisation of the predicate happen (وقع). This predicate (PRED) can be sub-categorised into Focus i.e. the interrogative tool, SUBJ for subject, and OBL-ARG for any missing argument such as object, and the rule for this interrogative is as follows:

If interrogative tool = where (أين)

followed by nominative past tense verb (فعل ماضي- مبني على الفتح)

followed by nominative Agent (فاعل مرفوع)

Then PRED sub-categorisation are:

PRED = ↑ SUBJ, ↑ OBL-ARG - Object (مفعول به) **AND**

Focus = where (أينَ :اسم استفهام مبني على الفتح في محل نصب - مفعول به ظرف مكان) **Where**

↑ SUBJ is:

{SUBJ = happen (فاعل مرفوع بالضمة “الحادث”)}

↑ OBL-ARG - Object is:

{OBL-Arg - Object = object circumstantial of place (مفعول به - ظرف مكان).}

The above rule has successfully created F-Structure presentation as Figure 7.6 shows. This structure is concerning the syntactic analysis. But this analysis is not enough to obtain an answer, what we need here is a semantic analysis in order to obtain the answer.

Focus: where

[اسم استفهام مبني على الفتح في محل نصب - مفعول بة ظرف مكان : اين]

PRED “وقع” < ↑ SUBJ, ↑ OBL_ARG (مفعول به) >

SUBJ

[accident فاعل مرفوع بالضمه “الحادث”]

OBL-ARG

[مفعول به - ظرف مكان]

Figure 7.6 F-Structure

S-Structure Presentation

From the F-Structure we know the predicate, the subject, and a hint of what the attribute of the object should look like e.g. location. However, we still have no knowledge of any predicate Thematic-object[2] to which our location attribute may belong. By applying the semantic rules we have the following:

[2] Thematic-object is a term given to the name of the object in the Object model. For more discussion see [Wins-92].

If Focus = where (أين) **AND**

the PRED is nominative past tense verb (فعل ماضي - مبني على الفتح) **AND**

the SUBJ is nominative Agent (فاعل مرفوع)

Then PRED REL (Relation) sub-categorisation is:

PRED REL = ↑ ARG-1 for argument-1, ↑ ARG-2 for argument-2 **where**
ARG-1 is:
{ARG-1= where (أينَ : مفعول به ظرف مكان) and the predicate's Thematic-object}
ARG-2 is:
{ARG-2 = the Slot Value of the predicate's Thematic-object.}

Given the semantic presentation, we have created S-Structure as Figure 7.7 shows. The S-Structure has, therefore, located exactly where the answer can be found. In our domain, the answer will be the location of the predicate's Thematic-object *accident* as the S-Structure and linguistics rules illustrated.

PRED REL

"وقع" : Intransitive, human, animal, things

ARG-1

ظرف مكان : "اين"
"الحادث" : Occurance of accident

ARG-2

Slot Value of *accident* Thematic-object.
Answer = Location:SV مفعول به - ظرف مكان

Figure 7.7 S-Structure

The *Who* (مَن) Set

We would like to demonstrate another set of particles, the *Who* set. Consider the following interrogatives:

1. Who reported the accident? (مَن ابلغ عن الحادث)
2. Who caused the accident? (من مسبب الحادث)

3. Who is the one who caused the accident? (من هو مسبب الحادث)
4. Who is the one who caused the accident? (من المسبب)
5. Who are the injured? (من هم المصابين)
6. Who are all the injured? (من هم كل المصابين)

F-Structure Presentation

Although they are in the same set, the above interrogatives are divided into three types of answers. Type one is about how the accident has been known as in interrogative one, type two is about what caused the accident to happen as in interrogative two, three and four. While type three is querying about *who has been injured*. The F-Structure in Figure 7.8 shows the presentation of the interrogative in two. Unlike the above set of interrogatives of *where* (أين), the F-Structure in Figure 7.8 shows the missing OBL-ARG i.e. the subject (الفاعل) of the accident (فاعل – اسم عاقل / غير عاقل). The F-Structure has been built by using the LFG sub-categorisation of the predicate caused (من مسبب). This predicate (PRED) can be sub-categorised into Focus i.e. the interrogative tool *who*, the OBJ for object, and Obl-Arg for any missing argument in this case the missing subject, and the rule for this interrogative is as follows:

If interrogative tool = who (من)

followed by nominative past tense verb (فعل ماضي– مبني على الفتح)

followed by nominative Agent (فاعل مرفوع)

Then PRED sub-categorisation are:

PRED = ↑ OBJ, ↑ OBL_ARG - SUBJ (فاعل) **AND**

Focus = who (اسم استفهام مبني على السكون في محل رفع – مبتدأ : من) **Where**

↑ **OBJ is:**

{OBJ = accident (مفعول به لمسبب منصوب)}

↑ **OBL-ARG - SUBJ is:**

{OBL-Arg - SUBJ = Subject person/ animal/ things (فاعل – اسم عاقل / غير عاقل).}

Focus: who اسم استفهام مبني على السكون في محل رفع – مبتدأ : من

PRED "مسبب" < ↑ OBJ, ↑ OBL_ARG (فاعل) >

خبر مرفوع بالضمه

OBJ

[accident (مفعول به لمسبب منصوب الحادث)]

OBL-ARG

[Name (فاعل – اسم عاقل / غير عاقل)]

Figure 7.8 F-Structure

The above rule has successfully created F-Structure presentation as Figure 7.8 shows. This structure is concerning the syntactic analysis. But this analysis is not enough to obtain an answer, what we need here is a semantic analysis in order to obtain the answer.

S-Structure Presentation

From the F-Structure we know the predicate, the object, and a hint of what the attribute of the subject should look like i.e. name of a person/ animal/ or things. However, we still have no knowledge of what is the predicate's Thematic-object. By applying the semantic rules we have the following:

If Focus = who (من) **AND**

the PRED is To Inform (خبر مرفوع بالضمة) **AND**

the OBJ is noun in the direct case accusative ((مفعول به لمسبب منصوب (الحادث))

Then PRED REL (Relation) sub-categorisation is:

PRED REL = ↑ ARG-1 for argument-1, ↑ ARG-2 for argument-2 **where**
ARG-1 is:
{ARG-1= who (اسم استفهام: من) and the predicate's Thematic-object}
ARG-2 is:
{ARG-2 = the Slot Value of the predicate's Thematic-object.}

Given the semantic presentation, we have created S-Structure as Figure 7.9 shows. The S-Structure has, therefore, located exactly where the answer can be found. Therefore, in our domain, the answer will be the name of the person who caused the accident. This can be of the predicate's Thematic-object *accident* as the S-Structure and linguistics rules illustrate this point.

PRED REL

"مسبب" : transitive, human, animal, things

ARG-1

اسم المسبب : "من"

"الحادث" : Occurance of accident

ARG-2

Slot Value of the predicate Thematic-object *accident*. Answer = Caused-Accident: SV(فاعل)

Figure 7.9 S-Structure

The *How many, How long, How much What is the (Date, Name, Number)* (كم) Set

1. How long did the rescue operation take? (كم استغرق الانقاذ)
2. How many people injured? (كم عدد المصابين)
3. How many people got killed? (كم عدد القتلى)
4. How many policemen were there? (كم عدد أفراد الشرطة)
5. How many children were in Joly's car? (كم عدد الأطفال الذين كانوا في سيارة جولي)

This set of interrogatives concerns / entails the following attributes name:

- Number of Passengers (as in interrogative five)
- Number of injuries (as in interrogative two)
- Employees (as in interrogatives four)
- Date/Time (as in interrogative one)

There are many other examples of this sort. These examples require not only linguistics rules, but also the use of the relation presentation of the object model. For instance, question number four above requires calculating the number of policemen. This can be achieved by using the object model relationship *Employed-by* between policemen and police station objects and then calculates the number of policemen. As for question number one, the time length of the rescue operation taken, this also needs to be calculated. This can be done by subtracting the starting date/time attribute from the rescue finishing date/time attribute in the object model. The linguistic syntax and semantic rules with their F-Structure and S-Structure presentation can be seen as follows:

F-Structure Presentation

The above first interrogative has been presented in the F-Structure Figure 7.10. It shows the missing Obl-Arg i.e. the object of the accident (مفعول به – يسأل به عن العدد). The Interrogative F-Structure has been built by using the LFG sub-categorisation of the predicate *taken* (استغرق). This predicate (PRED) can be sub-categorised into Focus i.e. any interrogative tool, SUBJ for subject, and OBL-ARG for any missing argument such as object, and the rule for this interrogative is as follows:

If interrogative tool = how much (كم) **AND**

followed by nominative past tense verb (فعل ماضي- مبني على الفتح) **AND**

followed by nominative Agent (فاعل مرفوع)

Then PRED sub-categorisation are:

PRED = ↑ SUBJ, ↑ OBL_ARG - Object (مفعول به) **AND**
Focus = how much (كم :اسم استفهام مبني على السكون في محل رفع مبتدأ) **Where**
↑ SUBJ is:
{SUBJ = rescue ("الحادث" فاعل مرفوع بالضمة)}
↑ OBL-ARG - Object is:
{OBL-Arg - Object = object of number date/time (مفعول به – يسأل به عن العدد).}

The above rule has successfully created F-Structure presentation as Figure 7.10 shows.

Focus: how long

كم :اسم استفهام مبني على السكون في محل رفع مبتدأ – يسأل به عن العدد)

PRED "استغرق" < ↑ SUBJ, ↑ OBL_ARG (مفعول به) >

SUBJ

فاعل مرفوع بالضمه - الحادث rescue

OBL-ARG

مفعول به – يسأل به عن العدد Number

Figure 7.10 F-Structure

S-Structure Presentation

The F-Structure has given us the predicate, the subject, and a hint of the attribute i.e., number or calculated number. However, we still need to know the Thematic-object from the domain model. By applying the semantic rules we have the following:

If Focus = how much (كم) **AND**

the PRED is nominative past tense verb (فعل ماضي – مبني على الفتح) **AND**

the SUBJ is nominative Agent (فاعل مرفوع)

Then PRED REL (Relation) sub-categorisation is:

PRED REL = ↑ ARG-1 for argument-1, ↑ ARG-2 for argument-2 **Where**
ARG-1 is:
{ARG-1= how much (كم : يسأل به عن العدد)and the predicate's Thematic-object}
ARG-2 is:
{ARG-2 = the Slot Value of the predicate's Thematic-object.}

Given the semantic presentation, we have created S-Structure as Figure 7.11 shows. The S-Structure has, therefore, located exactly where the answer can be found. In our domain, the answer will be calculated by subtracting the starting date/time attribute from the finishing date/time attribute in the predicate's Thematic-object *accident* as the S-Structure, and linguistics rules illustrate.

PRED REL

"استغرق" : Intransitive, human, animal, things

ARG-1

يسأل به عن العدد : "كم"

"الحادث" : Occurance of accident

ARG-2

<u>S</u>lot <u>V</u>alue of the *accident* Thematic-object.
Therefore,Answer = subtracting the starting date/time attribute:SV from the finishing date/time attribute:SV (مفعول به – يسأل به عن العدد)

Figure 7.11 S-Structure

Appendix D contains the rest of the interrogative particles with their F-Structure and S-Structure presentations. It also contains the linguistics rules associated to these presentations.

7.5.2 Ambiguity

NLs contain lots of ambiguities, which can be misleading as to the meaning of a sentence. As for interrogatives, consider the following example:

Where did the driver get hurt? (أين جُرِحَ سائق السيارة)

This interrogative is rather ambiguous in the sense that it is not clear whether we are looking for the accident location or the injury location. The system has no alternative but to produce the two locations, and will also notify the user of two possible answers. A linguistics rule can be drawn for such ambiguity as follows:

If Particle = Where (أين) **AND**

Verb = Past tense verb (فعل تام معلوم / مجهول) **AND**

Pro-Agent = Annexing (نائب فاعل مرفوع – مضاف) **AND**

Noun = Annexed (مضاف إليه – مجرور)

Then the answer is <u>all</u> direct patient complement of object accusative - circumstantial of place (مفعول به – ظرف مكان).

Other ambiguous interrogatives not only required linguistic rules to solve them, but also the actual domain rules. An example of this sort can be seen if we extend the above interrogative as follows:

Where did the driver of the car registration number xyz hurt?

(أين جُرِحَ سائق السيارة رقم xyz)

This type of interrogative requires checking the existence of the car registration number, and that the car has been involved in an accident. The process of checking and getting the answer to such an ambiguous query requires the combination of linguistics and domain rules which are discussed in the following section.

7.5.3 Domain Specific Rules

The Need

The domain knowledge, K-Structure, for the interrogative is needed when the S-Structure alone cannot answer a query. Furthermore, the K-Structure complements the S-Structure by identifying the right path for the required answer. K-Structure, its components, and its presentation have been described in section 6.3.2. This section gives a working example in order to illustrate its functionality.

The K-Structure consists of two main components, these are the Interrogative Nucleus and the Domain Rules. The Nucleus components consists of two sub- components, these are Nucleus-1 and Nucleus-2. Nucleus-1 represents the actual Theme[3] of the interrogative, its relations, and its Thematic-object, while Nucleus-2 represents the rest of the interrogative words/roots/ stems, their semantic features, and their Thematic-object. Although each interrogative has one Thematic-object, other words like *working-for* (يعمل في) may have more than one Thematic-object such as a police station, a fire station, and a hospital. The right Thematic-object will be decided by the Theme's Thematic-object. For instance, if the Theme is a policeman, then the *working-for* (يعمل في) Thematic-object will be the same as the Theme i.e. police station.

[3] The term Theme is a linguistic term used to identify the subject matter of the interrogative.

The information provided by Nucleus-1 and Nucleus-2 will direct the overall K-Structure in finding the right path and subsequently the right answer. This answer may be found by applying the Domain Rules within the K-Structure. The Domain Rules test the type of interrogative e.g. *where*, then select the appropriate Domain Rule *where* in order to find possible answers.

However, if a relation cannot be found between Nucleus-1 Thematic-object and Nucleus-2 Thematic-object, (providing that the Thematic-object of Nucleus-1 and the Thematic-object of Nucleus-2 are present), the Domain Rules can answer the query and also can find the missing relation between the two Thematic-objects as the K-Structure in Figure 7.14 shows.

The relationships between objects, within the domain object, determine the overall understanding of the interrogative. Consider the following interrogative: where does the policeman work? (أيـن يعمـل رجـل الشـرطة). From the domain model, we know that a policeman *works-for* police station. As the K-Structure consists of interrogative words with their relation features, and their Thematic-object, these relations are taken from the object model diagram presented in Figure 6.03 which models the overall domain. For example, the Thematic-object of our Theme is policeman and from the model we identify policeman relationships with other objects e.g. *works-for* police station. The idea here is to complement the S-Structure with all these relationships before we consider the target of the query. The next step is to identify a possible match from the relation features with that of the Theme. If found, then we can obtain an answer through the Domain Rules. However, we have to be sure that the word is not an alias for an existing relationship e.g. employed (يشتغل في) and working-for (يعمل لصالح).

Other type of relations can be drawn through deduction. These relations can be drawn as a result of answering queries that cannot be answered from exiting relations. An example of this type of queries is *where is the police dog registered?* (أين سُجلَ كلب الشرطة). Such police dog, should be registered in a police station, but there are no relationship between the object animal and the police station object. By applying the appropriate rule, the system first answers the query, and then notify the system administrator about the new discovered relationship, in our example, between object animal and the police station object. This will dynamically update our model which demonstrates and presents the system ability to answer queries using common-sense domain knowledge captured by the model.

The interrogative, *where does the policeman work*? (أين يعمل رجل الشرطة) has a relationship between the policeman and the police station that is policeman *works-for* police station. Therefore, the relation here is *works-for* (يعمل في) .

The interrogative gives us the relation phrase *works-for* (يعـمل في), and the Theme is the policeman, and since this phrase has not been stored or mentioned in the original story of the accident, we have to follow the Theme's relationships to other objects such that *works-for* is a relation to any object location. First, we have to prove that the S-Structure alone is not enough to answer such query. Consider the F-Structure in Figure 7.12, and the S-Structure in Figure 7.13.

```
Focus = where

    [ اسم استفهام مبني على الفتح في محل نصب - مفعول بة ظرف مكان : اين ]

PRED "يعمل" < ↑ SUBJ, ↑ OBL_ARG ( مفعول به ) >
        فعل مضارع / ماضي
SUBJ
              [ Policeman "رجـل الشـرطة" فاعل مرفوع ]
OBL-ARG
            [ Object - type of place (مفعول به - ظرف مكان )  ]
```

Figure 7.12 F-Structure

PRED REL

“يعمل” : Intransitive, human, + Works-for

ARG-1

ظرف مكان : “اين”

“رجل الشرطة” : Occurance of policeman

ARG-2

Slot Value of *policeman* Thematic-object.

Answer = Location:SV مفعول به - ظرف مكان

Figure 7.13 S-Structure

The *location:SV* of the S-Structure of the policeman Thematic-object gives us the wrong answer that leads to the accident location. The following K-Structure clarifies the ambiguity of this interrogative.

PRED <↑ **Interrogative Nucleus**, ↑ **Domain Rules** >

Interrogative Nucleus

Nucleus-1

PRED < ↑ Theme, ↑ Relations, ↑ Thematic-object >
Theme: Policeman رجل الشرطة
Relations: + Works-for
Thematic-object: + Police Station

Nucleus-2

PRED < ↑ Words, ↑ Semantic Features, ↑ Thematic-object >
Words: Work يعمل / عَملَ (The morphological root)
Semantic Features: + Works-for
Thematic-object: + Police Station, + Fire Station

Domain Rules

Particle Types

If Particle type = أين **Then** Answer in Set A **Where** Set A is as follows:

If Particle type = متى **Then** Answer in Set B **Where** Set B is as follows:

If Particle type = X **Then** Answer in Set Y

Rule Sets

SET A: *A1* **If** Nucleus-1 Relation is in the set of Nucleus-2 Semantic Features
Then Answer = Nucleus-1 Thematic-object Location.
A2 **Else** Build new Relation between Nucleus-1Theme and
Nucleus-2 Thematic-object.
Answer = Nucleus-1 Thematic-object Location.

SET B:
B1 **If** Nucleus-1 Relation is in the set of Nucleus-2 Semantic Features
Then Answer = Nucleus-1 Thematic-object Date/Time.
B2 **Else** Build new Relation between Nucleus-1 Theme and
Nucleus-2 Thematic-object.
Answer = Nucleus-1 Thematic-object Date/Time.

Figure 7.14 K-Structure

As the K-Structure in Figure 7.14 shows, Nucleus-1 represent the Theme, it's relationship, and its Thematic-object. Nucleus-2 represents the actual Theme of the interrogative, its semantic relation, and its Thematic-object.

The information provided so far in Nucleus-1 and Nucleus-2 needs to be driven by certain sets of rules. For instance, and as Figure 7.14 shows, the interrogative where (أين) has been allocated set A, and within the rule set A there are two sub-sets, namely, A1, and A2. A1 set will be given higher priority than A2 so that it can be fired first. By applying the domain rules, it can answer a query which has an existing relation in the object model such as *where does the policeman work?* (أين يعمل رجـل الشـرطة). A2, on the other hand, can answer a query which has no relation in the object model, (providing that the Thematic-object of Nucleus-1 and the Thematic-object of Nucleus-2 are present), such as: W*here was the police dog registered?* (أين تم تسجيل كلب الشرطة). It can also find the missing relation between the two objects as the K-Structure in Figure 7.14 shows. Different particles in the Domain Rules of the K-Structure follow the same pattern behaviour of the Rule Sets of the K-Structure.

7.6 Summary

In an ideal language processing Question-Answering System, a number of fundamental assumptions must be borne in mind, these being: an understanding of the constituents of an interrogative; an appreciation of inferencing, searching strategy, retrieval techniques, and the generation of the required answer. The formation of the proposed Question-Answering System has been based on the understanding of the syntactic, semantic, and common-sense domain knowledge of the constituents of the interrogative, this has been achieved by analysing the interrogatives using the LFG theory.

This chapter has shown that it was possible to generate answers by amalgamating the Lexicon and the Object diagram. This has been achieved by establishing relationships through linguistics and domain-specific rules. These relationships show encouraging results in finding the correct answer, and also proved to us that our approach creates a valuable basis for future industrial Question-Answering Systems, using the LFG theory for Arabic.

Chapter 8

Prototype Implementation of the Question-Answering System

The previous chapter outlined the design architecture of prototype QAS, where a number of ideas have been presented as an approach to Computational Linguistics for Arabic. This chapter considers the implementation of prototype QAS based on the above theoretical ideas. Thus, this chapter contains all the components of the overall proposed design prototype.

8.1 Kappa Tools and the ProTalk Language

Kappa Expert System V.311 was the chosen tool for implementing the prototype. The implementation language was ProTalk language and the inferencing technique was Forward Chaining. The prototype has been tested on sets of queries from different newspaper stories and produced the expected results.

Kappa has been built by Intellicorp, [Inte-96] and [URL 05], it has been considered as a Rapid Application Development (RAD) environment for developing Object-Oriented applications. The Kappa system is designed for capturing processes in directly executable models, resulting in immediate feedback to end-users and developers. It is a rich visual tool set, which is built to make the transition to Object-Oriented technology easy. OMW, on the other hand, is the Object-Oriented case tool built on top of the Kappa environment. OMW implements the Martin/Odell Object-Oriented Information Engineering methodology [Mart-95]. The key features and benefits that influenced our choice to use Kappa as the development tool are:

Object-Oriented Model

Kappa provides a rich and expressive Object management system and thus brings with it the benefits of Object-Oriented development. Its capabilities include multiple-inheritance, named objects and multi-value attributes. The object manager's persistence and dynamics make development fast and productive by enabling graphical browsing, interactive execution, and easy access to meta-data even at run-time.

Executable Models

OMW diagrams are executable, operations can be viewed during the execution process as well as through the linked operation. These provide immediate feedback and support evolutionary prototyping and RAD.

Visual Development

Visual Development enhances communication between members of a development team. It also results in superior maintenance and re-use characteristics. Kappa provides a powerful window-painting tool for creating Graphical User Interfaces (GUIs). Using the direct manipulation capabilities of the Interface Workbench, it is possible to create interfaces in minutes that would take days to create using the Motif Toolkit.

To reduce the need to write call-back code for graphic components, the Interface Workbench provides a 'Data Linkage' system. This enables setting up 'live' connections between window components and objects in the rest of the application. Once the link is set up, the system handles the task of displaying and changing values dynamically.

Rich Modelling Capabilities

These are ideal for representing the process logic and result in cutting down the number of lines required to be coded. Logic can be written in the ProTalk language or in C. External C programs can also be invoked. ProTalk is the high-level language of Kappa. It is hybrid, incorporating some features from procedural, fourth-generation (4GL) and object-based languages. ProTalk contains an object-query and pattern-matching engine. ProTalk also includes a full-featured rule system for representing rules and policies within the application.

Separation of Domain Specific and Domain Independent Data

Kappa/OMW allows domain specific data to be stored separately (within a specific scenario in the Scenario Manager) from domain independent data (for example, a Lexicon). Both can merged at runtime giving the application a unified view. There are two advantages of this approach - domain independent data can be re-used across a number of applications and it improves search performance since only relevant data needs to be accessed.

Limitation of Kappa/OMW

During the implementation of this project, a number of limitations were encountered in the Kappa/OMW tool. Kappa/OMW is a sophisticated application development tool and includes a number of complex but powerful features. For instance, whilst a helpful step-by-step tutorial is available for Kappa, no such guide is available for OMW. Much has to be mastered in learning various features and facilities [Peer-96].

Kappa/OMW maintains its own internal indexes and determines its own search order. No facilities are provided to the application developer to be able to influence the search. It is therefore not possible to optimise and fine-tune the search path for a particular application based on prior knowledge of the data or of the search characteristics. For example, if a particular object has 30 slots, say Slot1-Slot30, it is expensive to build and maintain indexes on each of these. If it is known that the search criteria always includes a particular slot, say Slot15, then it would be advisable to build an index on this slot and use it as the primary access path. Kappa/OMW does not provide any such facilities. The actual search logic used by Kappa/OMW is not documented as it is deemed to be a ‘trade-secret’.

A search is also performed when a number of rules within a rule set need to be evaluated. Here too, the order of search in determined internally by Kappa/OMW and the application developer has no control over it. Run time performance is therefore likely to be a key issue in Kappa/OMW applications which involve a significant amount of searching.

In Kappa/OMW, execution of rules, functions and procedures is synchronous and single threaded. Thus it is not possible to invoke a set of rules to be executed in a separate thread asynchronously and continue with some other independent logic. With multi-threading Kappa/OMW could perform some lengthy computations in one thread, while other thread(s) interact with the user. For instance, with multi-threading, during the implementation and while a compilation process is running the designer can view other OMW diagrams.

This limitation adversely affects the execution performance especially on a multi-CPU machine such as SUN work station. Here, spare CPU cycles on the additional CPUs will not be exploited.

8.2 User Interface

As it has been mentioned earlier, the software resource given at the time of implementing the prototype was Kappa, and no Arabic version is available. After investigation with Intellicorp, [Inte-96] and [URL 05], Arabic interface on Kappa is due to be included in a future version of Kappa. Our goal in implementing the prototype is to demonstrate what has been designed in the previous chapter using Arabic characters. Given this limitation, we have decided to implement the prototype using Latin characters. This limits the presentation of Arabic words to Latin characters.

8.2.1 Running the Prototype

In order to interrogate the prototype, various views have been conducted based on the newspaper stories. As a consequence, a collection of interrogative paradigms in both word orders have been created. This collection can be found in Appendix B.

The user can choose and interrogate any stored story in the system, the stories are presented with a fully Kappa object-oriented Graphical User Interface (GUI). This aids the user and allows access to the required knowledge by interaction with the prototype in a natural way. Appendix E provides commands on how to run the prototype.

The sequence of events are shown in the event diagram in Figure 8.1. Once the user starts executing the events diagram, the system displays available stories for the user to choose from. This can be shown in the event 'GetStoryID' which has an embedded menu of the available stories[1]. By clicking at the required story, the system automatically load that story as Figure 8.2 shows. The 'GetArabicInterrogative' event starts by displaying an input screen for the user

[1] The traffic accident stories are defined by combination keys of: Location, Date, and Time.

to fill in the Arabic interrogative, the 'BreakIntoWords' event starts breaking the interrogative into individual words for analysis.

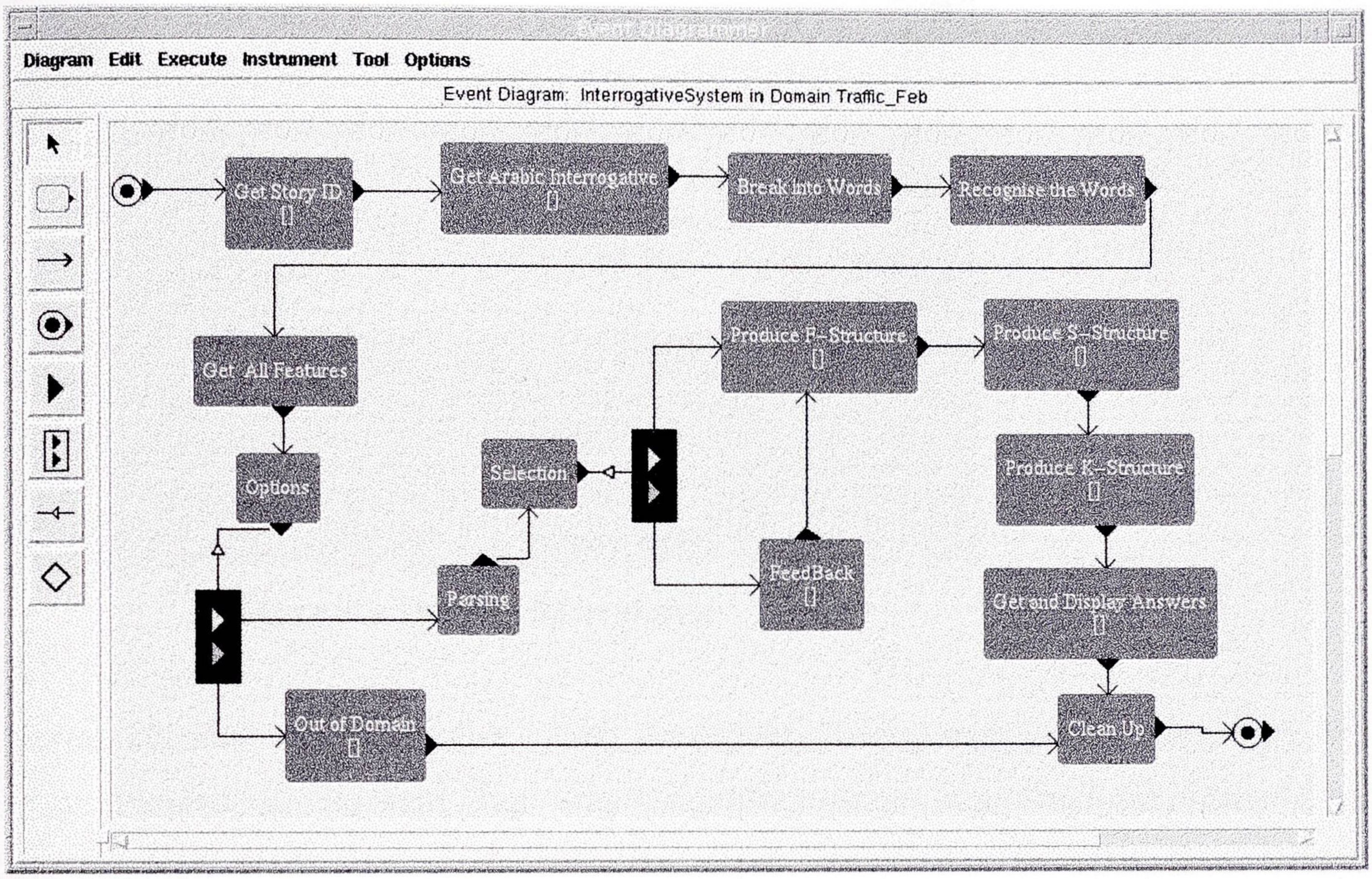


Figure 8.1 The Event diagram

The next stage is to recognise those words, the process of the event 'RecogniseTheWords' will take place and any offending words will be pointed out. Once the words have been recognised, the 'GetAllFeatures' event reads each word's Thematic-object/features from the lexicon. This process will also identify the required objects, their attributes, and their values for each word in the interrogative, this comes about as the result of combining the object model, and the lexicon.

At this stage, the system decides whether the overall words of the interrogative are within the domain or not. If it is not, the 'OutOfDomain' event highlights the words with some feedback, and then terminates the query. If the interrogative has sufficient words to qualify it to stay within the domain, but with some unrecognisable words, the process will continue and the offending words will be pointed out.

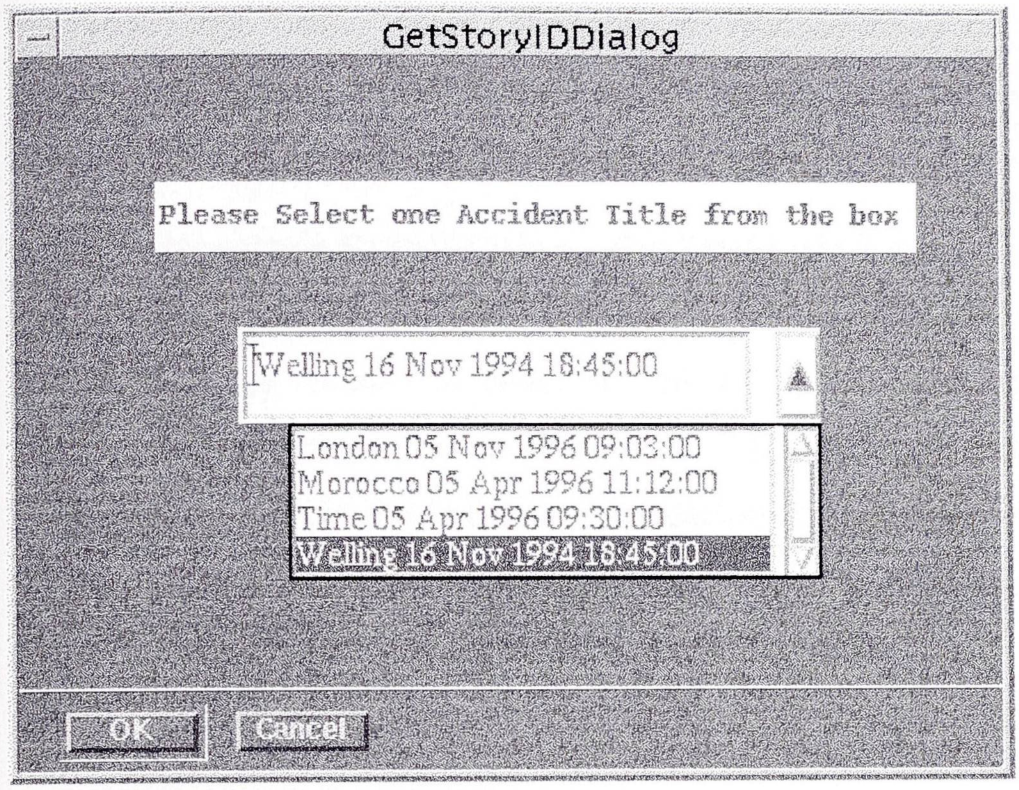

GetStoryIDDialog
Please Select one Accident Title from the box
Welling 16 Nov 1994 18:45:00
London 05 Nov 1996 09:03:00
Morocco 05 Apr 1996 11:12:00
Time 05 Apr 1996 09:30:00
Welling 16 Nov 1994 18:45:00
OK
Cancel

Figure 8.2 The Stories available

Having obtained an interrogative within the domain, the next stage is the parsing stage. The event 'Parsing' can currently deal with Arabic interrogatives using two word orders: ***verbal*** and ***nominal***. The particular properties the parser can recognise depend upon the lexicon. More syntactic/semantic features can be accommodated by augmenting the lexicon. The execution order of the general operations of the Parser, currently deals with the agreement system of Arabic as it has been illustrated in chapter three and four.

8.3 Question-Answering System and Inference

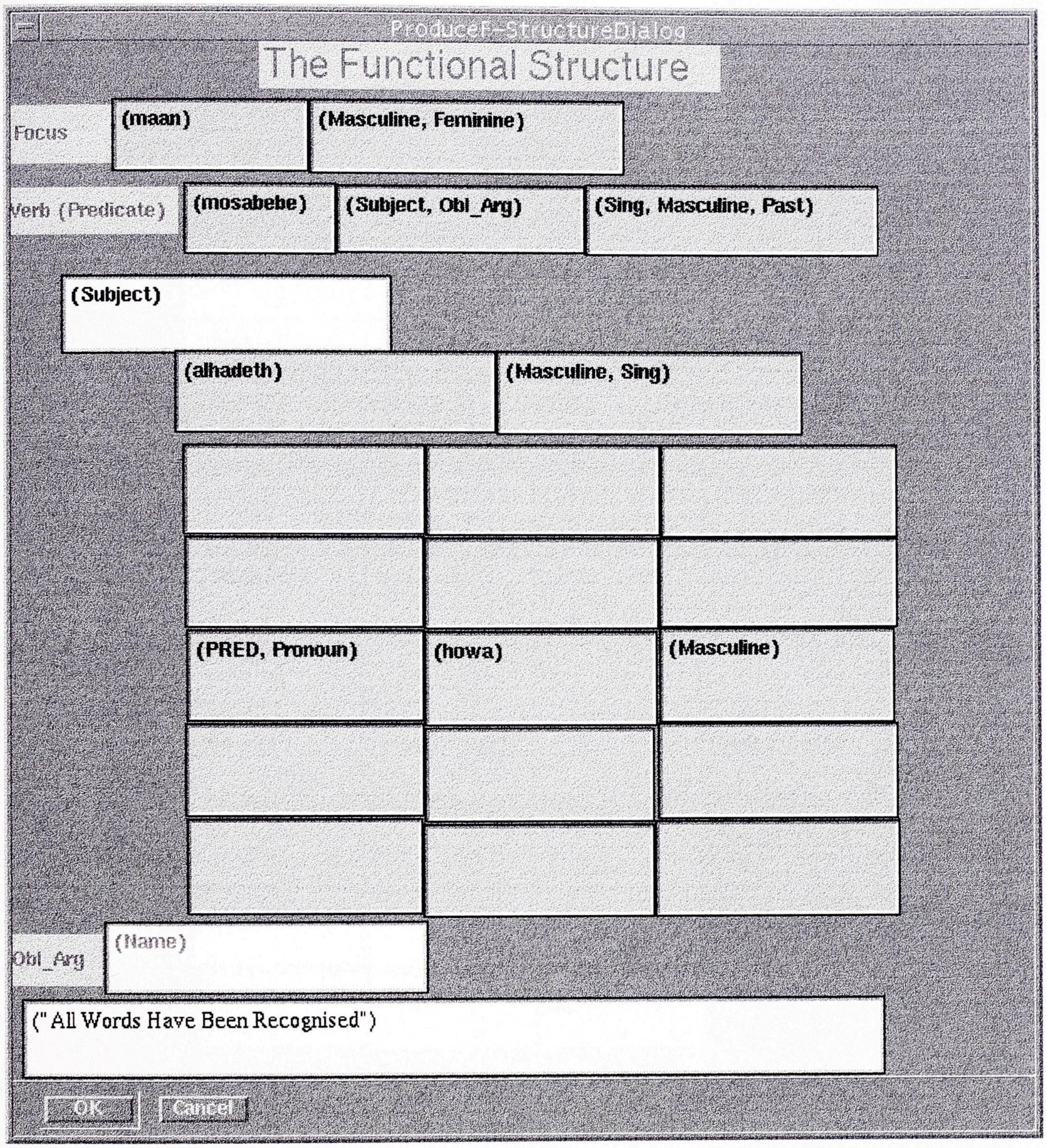
The first step towards finding the meaning(s) of a given interrogative is to assign to it some structures, such as LFG structures, that will be useful for further processing. Engineering a language is, in fact, the analysis of a given sentence, where the analysis not only recognises the words of the sentence but also assigns to it a construction in the proposed language. The output of this analysis, if successful, would be a structure outlining the words in the sentence and their syntactic features. This analysis can be stretched even further to the semantics of these words, so the output structure can include both the syntactic and the semantic features.

Furthermore, this analysis can be stretched to include the domain knowledge i.e. the common-sense domain knowledge deduced from what has been said. Thus giving the actual meaning of the sentence, and subsequently, helping in a major way in obtaining the correct answer from the knowledge base. Hence, the task in this section is not only to parse an Arabic interrogative, but to 'combine' the syntactic, semantic, and the domain knowledge structures in order to form a constructive formula to interrogate the knowledge base.

8.3.1 F-Structure Presentation - the Syntactic Level

The system produces complete F-Structure for the interrogative including the syntactic features such as, feminine or masculine, and their agreement system which has been shown in chapter three and four. The constituents of the F-Structure contents come from the parsed interrogative words stored in the Lexicon. Each constituent appearing in the F-Structure has a category. The category has a category-name and a list of feature values. Figure 8.3 shows a typical complete F-Structure for the following interrogative:

Who caused the accident?
(maan huwa mosabebe alhadth? من هو مسبب الحادث؟)



The Functional Structure

Focus (maan) (Masculine, Feminine)

Verb (Predicate) (mosabebe) (Subject, Obl_Arg) (Sing, Masculine, Past)

(Subject)

(alhadeth) (Masculine, Sing)

(PRED, Pronoun) (howa) (Masculine)

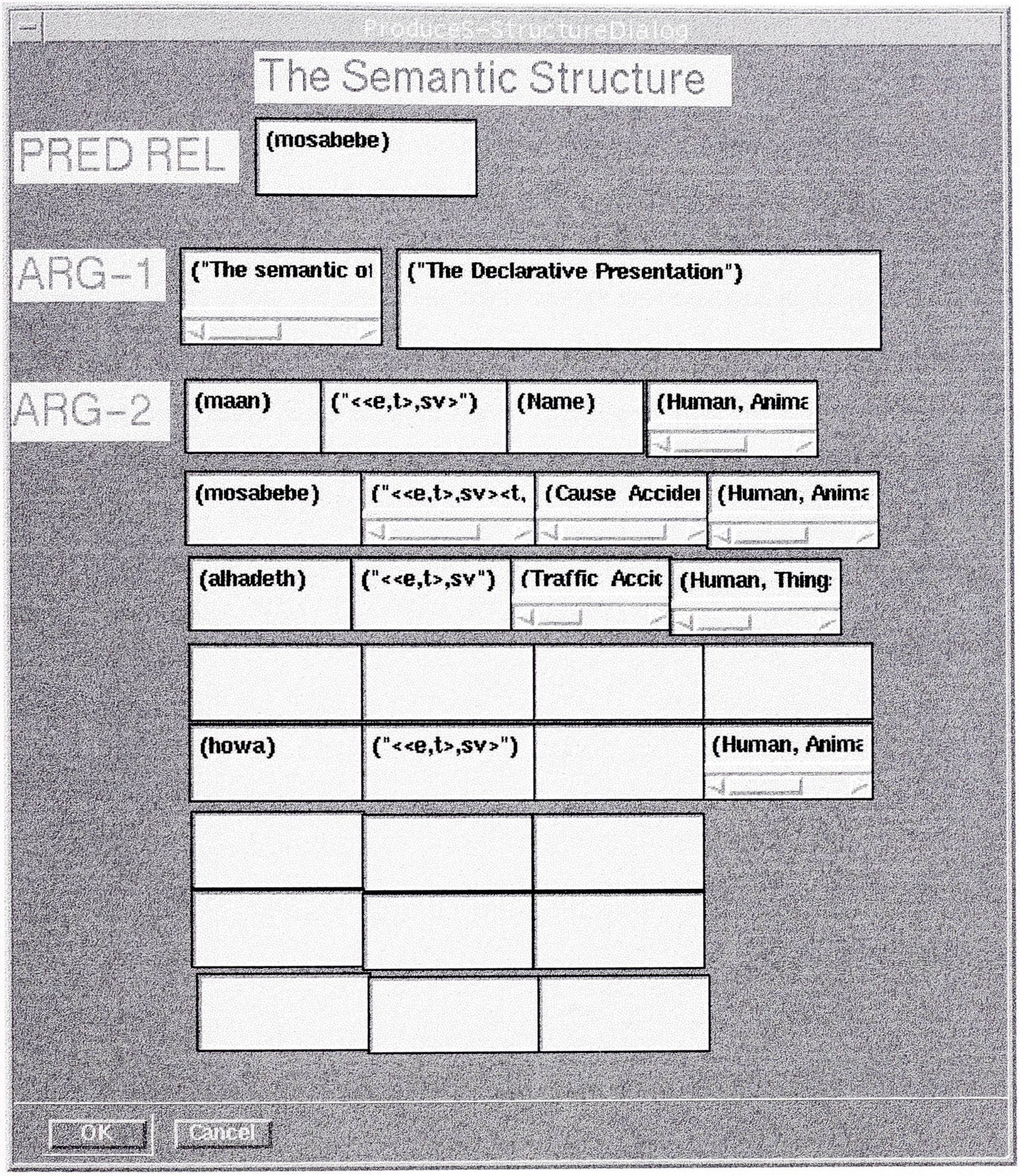
Obl_Arg (Name)

("All Words Have Been Recognised")

Figure 8.3 A Complete Functional-Structure Presentation

8.3.2 S-Structure Presentation - the Semantic Level

The system produces a complete S-Structure for the interrogative. Argment-2 of the S-Structure, as Figure 8.4 shows, will be filled by the semantic features provided by the lexicon. Knowing the semantic features of each word, the system will determine a list of features associated to this word. As it has been explained in chapter seven, once the semantic features have been met, the answer can be generated automatically. Figure 8.4 shows the semantic features of the above interrogative.



The Semantic Structure

PRED REL (mosabebe)

ARG-1 ("The semantic o1 ("The Declarative Presentation")

ARG-2

(maan)	("<<e,t>,sv>")	(Name)	(Human, Anim
(mosabebe)	("<<e,t>,sv><t,	(Cause Accide	(Human, Anim
(alhadeth)	("<<e,t>,sv")	(Traffic Acci	(Human, Thing
(howa)	("<<e,t>,sv>")		(Human, Anim

Figure 8.4 A Complete Semantic Structure Presentation

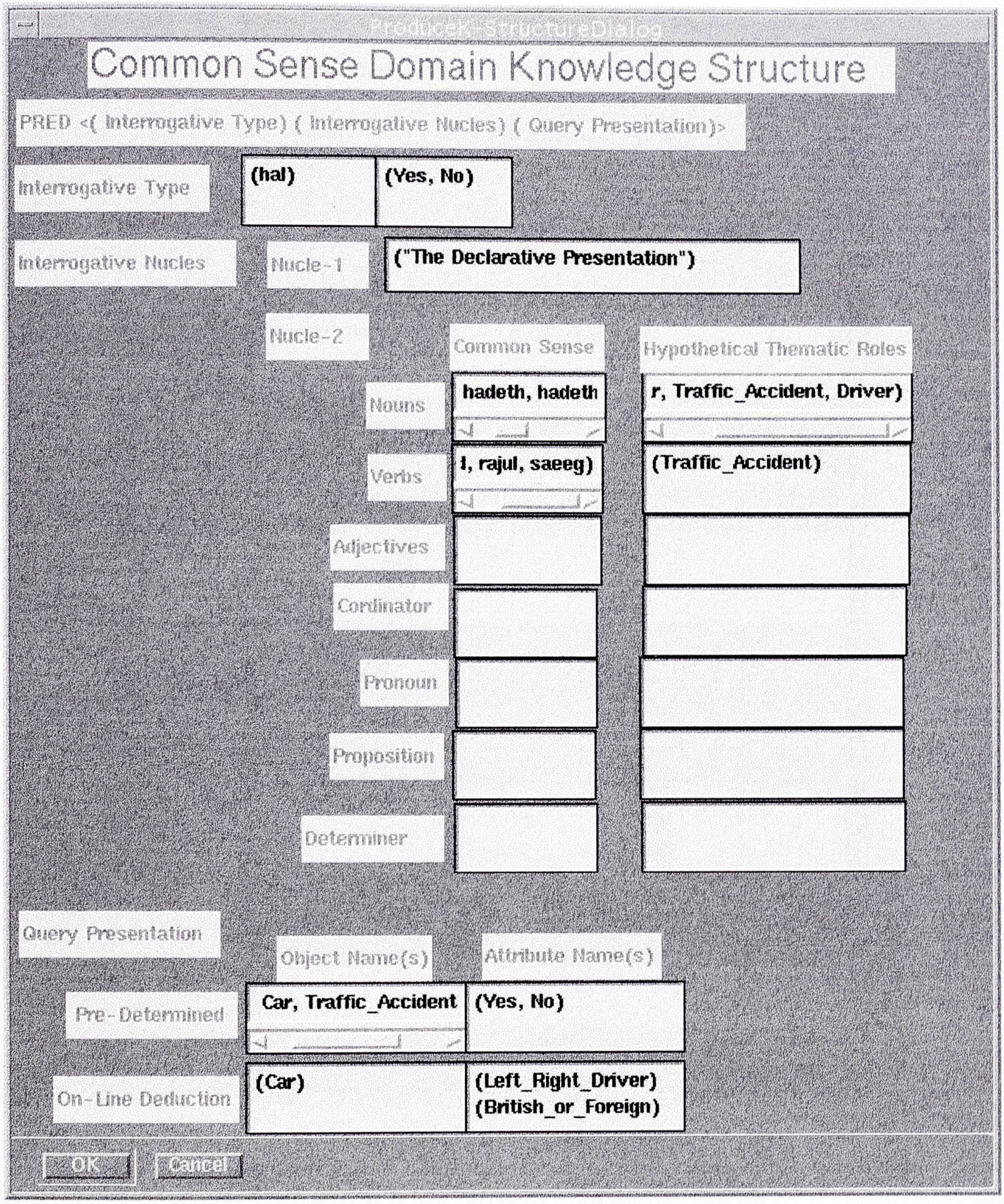
8.3.3 K-Structure Presentation - the Common-sense Knowledge Level

The prototype produces K-Structure only if the interrogative needs one. In other words, if the semantic is not enough to obtain the answer as we have seen in chapter seven, the prototype generates K-Structure based on the knowledge we have so far from the S-Structure and the object model with the appropriate Knowledge Domain Rules. This knowledge can be captured by applying the rules to the current analysed text and inflated with deduced common-sense features. Consider the following interrogative:

Did the driver of the registration number OAHAM-28 get killed?

(hal gutila saeeg alseara raggam OAHAM-28 هل قتل سائق السيارة رقم)

Figure 8.5 shows the K-Structure for the above interrogative. The Figure demonstrates how and where the correct answer can be obtained by applying the appropriate common-sense domain knowledge rule. For example, the declarative story in Appendix B (under heading "London لندن ") alone will not be enough to answer the interrogative above i.e. whether the driver has been killed or not. From the fist instance, the user might think that the driver has been killed as a result of the accident. But by applying the rule, and since this car is a foreign car, we find out that the driver location of this car is on the left, i.e. a left-hand-driver car, and the damage, as a result of the accident, was on the right-hand-side of the car. Therefore, the person who was killed was the passenger who was sitting on the right i.e. front passenger seat not the driver seat.


ProduceK-StructureDialog
Common Sense Domain Knowledge Structure
PRED <(Interrogative Type) (Interrogative Nucles) (Query Presentation)>
Common Sense
Hypothetical Thematic Roles
Object Name(s)
Attribute Name(s)
OK
Cancel

Interrogative Type: (hal) (Yes, No)

Interrogative Nucles

Nucle-1: ("The Declarative Presentation")

Nucle-2

	Common Sense	Hypothetical Thematic Roles
Nouns	hadeth, hadeth	r, Traffic_Accident, Driver)
Verbs	l, rajul, saeeg)	(Traffic_Accident)
Adjectives		
Cordinator		
Pronoun		
Proposition		
Determiner		

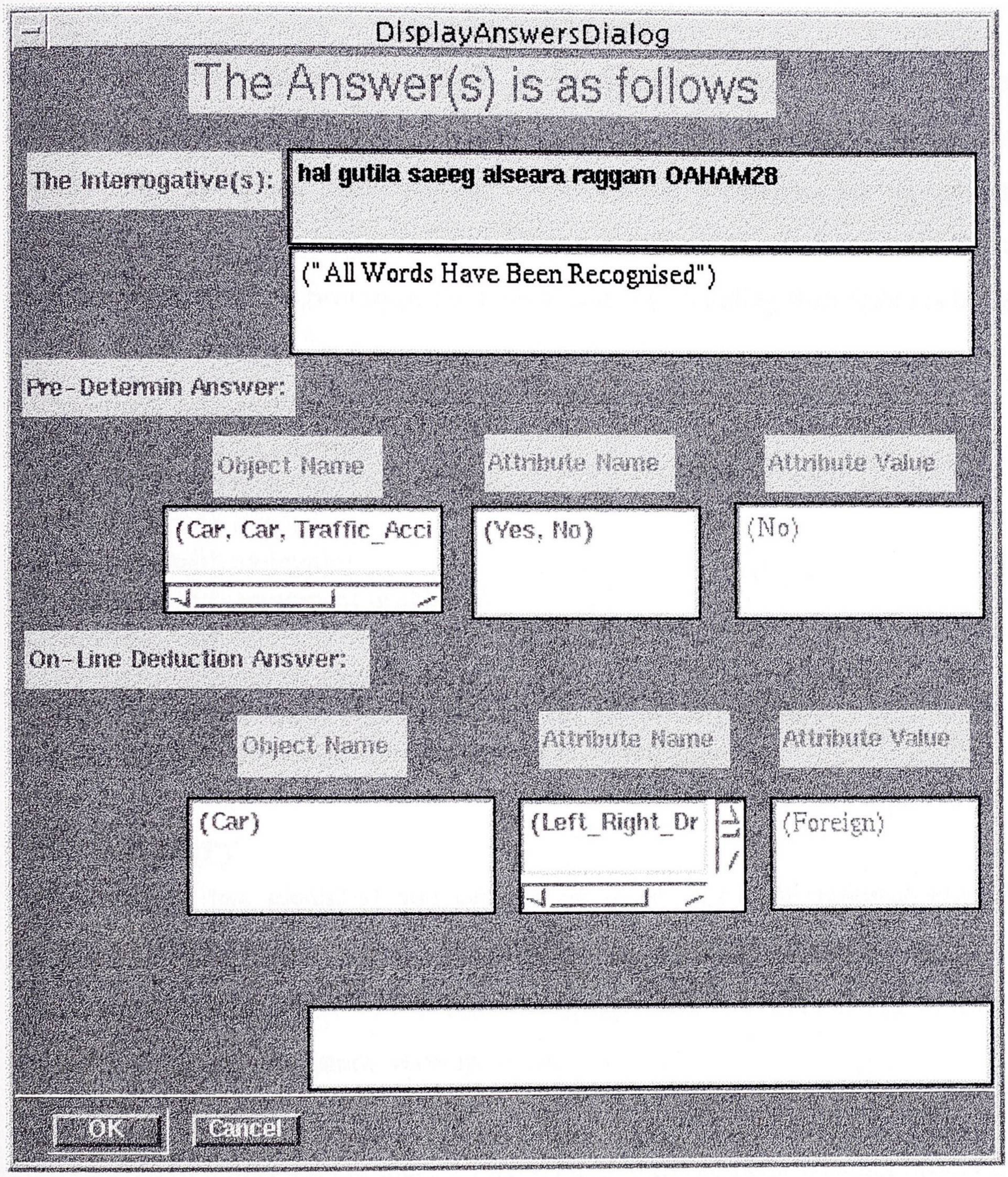
Query Presentation

	Object Name(s)	Attribute Name(s)
Pre-Determined	Car, Traffic_Accident	(Yes, No)
On-Line Deduction	(Car)	(Left_Right_Driver) (British_or_Foreign)

Figure 8.5 A Complete Knowledge Structure Presentation

Answer Presentation

As we have seen from Figure 8.5, there are two methods by which the prototype retrieves the answer, namely, a Pre-Determined, or On-line deduction. All answers are determined by Rules whether they are linguistic rules or domain specific rules. Rules are grouped into sets. The ideal way, as chapter seven shows, is to give one set of rules to each question type. In the case of interrogatives with two types of question e.g. when and where questions, the system triggers the control of these sets in order to answer these questions. Figure 8.6 shows how the system presents the answer to the user. The Figure also shows the name of the Thematic-object(s), Attribute(s), and their values for convenience.



The Interrogative(s): hal gutila saeeg alseara raggam OAHAM28

("All Words Have Been Recognised")

Pre-Determin Answer:

Object Name: (Car, Car, Traffic_Acci

Attribute Name: (Yes, No)

Attribute Value: (No)

On-Line Deduction Answer:

Object Name: (Car)

Attribute Name: (Left_Right_Dr

Attribute Value: (Foreign)

Figure 8.6 Answer Presentation

8.4 Different Newspaper Stories and Sample Runs

Most of the traffic accident stories experimented with in this prototype are published real traffic accidents. They have been taken from Arabic and English newspapers. These stories can be seen in Appendix B.

The sets of interrogatives used to interrogate the prototype, have shown the scope of the prototype. In constructing this prototype, our aim was not only to measure its efficiency, but to show that our theoretical computational linguistic analysis of the Arabic interrogative can be interpreted in a model QAS, and that it is indeed efficient enough to be used in practical applications.

The interrogatives have been utilised as specimens to prove the implementation of our theoretical work in order to retrieve answers. The following specimens have been experimented with, and show the real depth of the prototype, the example runs of these specimens can be seen in Appendix C. More interrogative types from other languages can be easily augmented to this prototype for experiments by including their features in the lexicon.

- Questions with answers
- Questions with no answers
- Questions with more than one answer
- Questions with quantifiers
- Questions with ambiguity
- Questions with agreement problems
- Questions with different word orders
- Questions with out of domain value
- Questions with multiple constituent answers and co-ordination
- Questions with co-ordination phenomena of verbal gapping

8.5 Summary

This prototype has identified and opened up areas for future research towards processing Arabic. This may involve the use of a morphological analyser as a front-end to the prototype, generating sentential answers i.e. Natural Language Generation instead of constituent answers, thus paving the way for future developments.

The process of implementing this prototype highlighted some interesting observations, such as the relationship between the Object Model and the Lexicon. Although NL provides vast ways of asking questions about the same story, the linguistic and the Domain specific Rules show the power in answering these questions in a rather surprising way. For example, as long as the words have the same meaning, to the prototype, they eventually lead to the same Thematic-object name and to the attribute name, this will consequently lead to the same answer. Therefore no matter how the question has been phrased, the user will eventually get the same answer.

Furthermore, although the user should observe Arabic word orders and the arrangement of interrogative words, if this order is not respected, the prototype, after alerting the user, can still provide the answer. Therefore, the combination of objects behaviour in the Domain Model with the LFG Structure rules, has resolved most, if not all, common ambiguities found in NL for that domain and enhanced the understanding ability of the proposed prototype.

Chapter 9

Conclusions, Future Developments, and Industrial Applicability

9.1 Conclusions

The overall objective of this research has been the development and implementation of a Question-Answering System in Arabic (QASA), (an area which has not, until now, been investigated in any real depth) using a computational linguistic approach. In order to achieve our objectives, we have adopted existing theoretical approaches and also put forward theoretical proposals of our own as a contribution to enhance and give substance to the understanding of both QASA and QASs in general.

The proposed model, in chapter seven and eight, and the project as a whole represent the first step in a very ambitious area of research. This project is far more significant and far-reaching than a first glimpse at its objectives might convey, since during the analysis of Arabic interrogatives, chapter three, four, five, and six, it has uncovered many problems which might prove significant in areas of further research related to QASs.

The project's implementation, chapter eight, rests on a series of fundamental theoretical ideas, widely accepted in the field of Computational Linguistics. The discussions in the body of this thesis have attempted to integrate these ideas with the overall objectives of this work, resulting in a QAS which integrates syntax, (chapter three and four); semantics, (chapter five); and common-sense domain knowledge, (chapter six). This has resulted in a prototype model for understanding Arabic interrogatives. It is this that distinguishes our model from other systems. The novelty of this project lies in the amalgamation of the following:

Firstly, the adoption of the LFG theory to linguistically analyse the constituents of the interrogative, with a view to gaining a broader understanding of the linguistic structure of its constituents. This theory has been partially extended in order to accommodate the interrogative verbal gapping phenomena of Arabic Co-ordination, (chapter four and five).

Traditionally, QASs have relied on subset of NLP techniques instead of adopting a proper Computational Linguistic theory to analyse their data. We have argued for the use of a Computational Linguistic theory, [Yama-98] and [Al-Kh-96], as a main component of NLQASs which has given our own project more linguistic substance, thus enhancing one's understanding of the theoretical issues of NLs. By adopting the LFG theory to analyse the Co-ordinated interrogatives of Arabic, it appeared that the expressive means at the disposal of the traditional LFG was not enough to accommodate the verbal gapping phenomena of Arabic and that a certain expansion of the LFG theory mechanism was necessary, [Yama-94a]. This expansion has accommodated the following interrogative co-ordination phenomena:

1. **verbal interrogative Co-ordination appearing with verbal gapping;**
2. **nominal interrogative Co-ordination appearing with verbal gapping;**
3. **multiple verbal interrogative Co-ordination appearing with verbal gapping;**
4. **multiple nominal interrogative Co-ordination appearing with verbal gapping.**

The conclusion then, is that the traditional LFG does not have the formal apparatus to encode the properties of the above phenomena. Thus, the ($\uparrow$ PRED = $\downarrow$ PRED) equation has been introduced to overcome this problem. Furthermore, to enforce the agreement in the two word orders, a further equation has also been introduced, namely, the ($\uparrow$ AGR = $\downarrow$ AGR). This has ensured the gender agreement in both word orders. The equation ($\uparrow$ TNS = $\downarrow$ TNS) has also enforced the tense attribute value of the second Co-ordination. In the same way, these equations have also accommodated multiple Co-ordination. As a result of this formal extension, a complete, coherent, and unique F-Structure has been formally devised to complete the linguistic analysis of the interrogative, more details are in chapter four and [Yama-94a].

Furthermore, although this thesis considers both the syntax and the semantics of interrogatives, the traditional LFG theory once again, has no formal account for the semantic structure of the interrogative. In order to formulate the semantics, we have proposed an S-Structure for the interrogative and the Co-ordinated interrogative based on [Halv-88a] approach for the declarative and have extended it to accommodate the constituents of the interrogative. This extension has also accommodated the semantic features of the new Interrogative Language IntLang. The proposed interrogative S-Structure has been used to form an interrogative formula, and not only presents the semantics of interrogatives, but also formulates the shape of the expected answer from the knowledge base.

Secondly, it became apparent during our research that the semantics for the declarative is not the same as the semantics for the interrogative [Groe-90] and [Engd-86]. This prompted us to gain a deeper understanding of the interrogative in general and the Arabic interrogative in particular - in both word orders - and led to the proposal of a new IntLang tailor-made for the interrogative, based on Montague's Semantics for the declarative, chapter five and [Yama-98]. It appeared that this declarative type theory could indeed be extended to accommodate the interrogative type theory of IntLang. We have investigated aspects of the semantics of the constituents of the interrogative, and have adopted Montague's <e,t> declarative type theory, extending this to accommodate the Intentional type <sn> theory and the Extentional type <sv> theory of an interrogative. The idea behind this was to incorporate these two type theories in an attempt to form an Interrogative-Type Theory as a basis for the proposed IntLang. The interrogative type <sn> theory serves as an intentional slot name, and the interrogative type <sv>theory serves as slot value, i.e. <<e,t>,sn> Intentionally and <<e,t>,sv> Extentionally. More details are in chapter five and [Yama-98]. Therefore, this research (theory and project combined) can be viewed as one of the first of its kind, in the sense that it deals with the Arabic interrogative using a computational linguistic approach, creating a semantic framework for the interrogative, a feature that no other system has used.

Thirdly the main extension of the LFG theory was the introduction of the K-Structure as a common-sense domain knowledge, and the addition of a formal representation of the Common-sense Domain Knowledge Structure (linguistic and domain specific rules), chapter six. The proposed K-Structure provides knowledge representation fundamental to the task of encoding Common-sense Knowledge about the Arabic language. Furthermore, it provides linguistic and conceptual knowledge organised into hierarchical associated knowledge structures that are metaphorically related or otherwise used in linguistic expression.

K-Structure is a natural extension of the S-Structure, it determines not only the semantic knowledge features of a sentence, but also how this knowledge can trigger another knowledge associated with it. The knowledge structure presentations are organised into hierarchies such as concepts, categories, which are constrained by using Grammar Rules in the Lexicon, and common-sense domain specific deduction Rules. Again no other system has used such power to overcome ambiguities found in Natural Languages.

Fourthly, chapters seven and eight present a prototype model which has been conducted by developing a general Object Model for the traffic accident domain. This model has been used to capture the functional behaviour of objects, the related verbal interactions and their linguistic associations in a given query. The combination of object's behaviour in the Domain Model with the LFG Structure rules, resolved most, if not all, of the common ambiguities found in Natural Languages (NL) and enhanced the understanding ability of the proposed system.

Since advanced publications in Arabic have only been materialised in recent years, in comparison to English in Computational Linguistics, and since Arabic differs in many of its linguistic aspects from the languages that dominate this research area, it requires its own Computational Linguistic analysis and subsequently its own processing model. Such a model must contain knowledge about the syntax and semantic of the constituents of interrogatives and also knowledge about these constituents. Therefore, it is necessary to combine this knowledge in order to design any QAS for Arabic.

The Lexicon for Arabic interrogatives with which this project deals involves Arabic interrogative verbal and nominal sentences. Since no previous Computational Linguistic work of this scale has been conducted for the purpose of QAS in this domain, it was necessary to restrict the study to these two types of interrogative categories whilst using a complex language like Arabic. We therefore narrowed the focus of the project down to one specific research area, namely the constituents of interrogatives in both Arabic word orders and examined this area in considerable depth. Nevertheless, although the scope has been narrowed, the configuration of this project as a whole, is actually quite broad in its specification. The project incorporates components such as interrogative syntactic analysis, interrogative semantic analysis, setting common-sense domain specific and linguistic rules in order to meet the computational demands made by this very specific area.

We believe that the above Computational Linguistic approaches and the combination of objects behaviour in the Domain Model with the LFG Structure rules, have opened a new dimension to the understanding of NLs in general.

Given the above approaches, this research as a whole has achieved its initial objectives, in that it has designed and implemented a modest model which represents a first step on the way to a more industrial applications QASA in the future as the next section explains.

9.2 Future Developments

It is clear that the proposed prototype in its present form can be applied to more than one area. One area would be to apply it to the Machine Translation. For instance, users can submit their queries in different languages where the information is stored in one knowledge base. If the queries were in the English language, say, in the domain of tourism, the lexicon can be extended to contain the English equivalent words of Arabic interrogatives for this domain. The LFG syntactic module checks for syntactic features of both Arabic and English in order to produce a complete F-Structure. As for the semantic, the same strategy will be followed in order to produce the S-Structure for both languages. In this respect, the role of the common-sense module can help in resolving some ambiguities during the translation between English and Arabic. The result is a complete translation of interrogatives to their target language, in our case, Arabic. This language will be the only language used to obtain required answers from the knowledge base.

Recovery enabling to process queries in substandard Arabic shouldn't be difficult to develop. A bigger challenge is: incorporating elaborate pragmatic patterns for social contexts (specially to enable the processing of more difficult interrogatives); the integration with healthcare, logistic, police, legal, and traffic management ergonomic procedures, whether computerised or not, and ensuring portability for dissemination - not necessarily with Arabic as being the system's NL.

A particularly interesting possibility is the integration with some legal evidence management support system (e.g., MARSHALPLANE [Schu-97]), by customising it for the specific technical domain for road accidents. Rooted in forensic statistics, the application of AI to legal evidence is a new sector within AI and law [Mart-98]. In turn, the automotive sector is established within the forensic sciences. [Pete-94] is in forensic automotive engineering for the American Law. [Boha-91] is on the role, in courts of, computer-aided accident reconstruction. Further research should proceed along two trajectories within NLP: the pragmatic dimension of QAS and discourse analysis; and the modelling of interrogative understanding.

9.3 Industrial Applicability

Driven by our initial motivation to design this project, which was prompted by our interest in the contemporary aspects of the Arabic language and culture, and a desire to set the ball in motion for the development of Arabic Computational Linguistics QAS, we came to realise that the project would not only be useful as a foundation for further and more specific research in Arabic language, but could also be used as an industrial QAS in a traffic office in more than one of the Arab countries. To this respect, we have started approaching the traffic office and the Hajj ministry in Mecca, the Pilgrimage Ministry (وزارة الحج والأوقاف) in Saudi Arabia for a possible industrial implementation of this system.

In a management or even operation research perspective, application to the road system and hospitals during Pilgrimage seasons is an interesting challenge: seasonal load requires sophistication e.g., in the selection of target hospitals able to provide appropriate care. Such a capability would make the tool even more interesting, because of its robustness, for telematics for healthcare in general.

Perhaps there is even a potential for monitored vehicles: supposed cars venture into isolated areas, e.g., the empty quarter of Saudi Arabia (الربع الخالي). Electronics monitoring is an open area of research in Europe, e.g., in the development of automated driving system, or traffic load monitoring. Here, monitoring would be done for safety. It could be integrated with this tool for emergencies.

Finally, perhaps more importantly, future Computational Linguistic research in Arabic interrogatives will now have a guiding reference. It may well be the case that future Arabic Computational Linguistic systems will resemble this model in a few details. If that is the case, at the very least, referring to this research could help future researchers identify what the Arabic computational linguistic system should comprise. It is our hope that the theoretical proposals discussed in this thesis may have clarified some of the important theoretical issues involved in processing Arabic interrogatives and contributed to a greater understanding of the structure of the components of a QASA.

Bibliographic References

and

Uniform Resource Locators (URL) WWW Sites

Bibliographic References

[AlHa-90] AlHarbi A. 1990 Syntactic Approach to Arabic Verbal Morphology. Ph.D. thesis; Essex University.

[Alkh-94] Al-Khonaizi M., Al-Aali M., and Al-Zobaidie A. 1994 The Classification Approach. The Egyptian Computer Journal.

[Alkh-96] Al-Khonaizi M., **Yamani A.,** Al-Zobaidie A., and Al-Aali M. A New Declaratives & Interrogatives Approach to Natural Arabic Text: A Computational Linguistics Approach. In the 5th International Conference and Exhibition on Multi-lingual Computing, ICEMCO-96, Cambridge University, April 1996.

[AlAa-94] Al-Aali M., and Girgis M. Arabization: Actual and Objectives. In the 4th International Conference and Exhibition on Multi-lingual Computing, ICEMCO-94, Cambridge University, April 1994.

[AlMu-88] Al-Muhtaseb H. A Natural Arabic Language Understanding System, M.S. Thesis, Computer Science, King Fahd University of Petroleum & Minerals, Saudi Arabia, (KFUPM), 1988.

[Alne-96] Alneami Ahmed 1996 The Arabic Computational Lexicon. In the 5th International Conference and Exhibition on Multi-lingual Computing, ICEMCO-96, Cambridge University, April 1996.

[AlZo-95] Al-Zobaidie A., Al-Khonaizi M., **Yamani A.,** al-A'ali M, Common Sense Knowledge Representation for Natural Language Processing: An LFG approach. In Processing Arabic Journal, Report 8, Nijmegen 1995, Ditters E. (Ed.), Institute for the Languages & Cultures of the Middle East, Nijmegen University Holland, ISSN 0921-9145, Pp 47-78.

[Arab-97] ArabTrans. Arab Net technology, Ltd 184 High Holborn London Wc1 UK. http://www.arab.net/arabtrans/.

[Araf-95] Arafah A. G. 1995 A grammar for the Arabic language suitable for machine parsing and automatic text generation. Ph.D. Thesis, Illinois Institute of Technology, 1995.

[Aref-95] Aref M. & Al-Muhtaseb H., 1995 Khabeer: An Arabic Object Oriented Production System and Query Language, in Processing Arabic Journal, Report 8, Nijmegen 1995, Ditters E. (Ed.), Institute for the Languages & Cultures of the Middle East, -Nijmegen University Holland.

[Aram-96] Arampatzis A., Tsoris T., and Koster C. 1996. IRENA: Information Retrieval Engine based on Natural Language Analysis. ACM- SIGIR-96.

[Bate-86] Bates M., Moser M. G. and Stallard D. 1986 The IRUS Transportable Natural Language Database Interface. BBN Laboratories Cambridge, MA Expert Database Systems, Proceedings from the First International Workshop. Ed. Lavry Kerschbery, Benjamin Cummings 1986.

[Beln-81] Belnap N. 1981 Approaches to the Semantics of Questions in Natural Language Part I, Unpublished manuscript, 1981, University of Pittsburgh.

[Beln-82] Belnap N. 1982 Questions and Answers in Montague Grammar. In Peters S. and Saarinen E. (Eds.) Processes, Beliefs, and Questions, Reidel, Dordrecht.

[Benn-79] Bennett M. 1979 Questions in Montague Grammar. Indiana University Linguistics Club, Bloomington.

[Bogu-83] Boguraev B. and Sparck Jones K. 1983 How to drive a database front-end using general semantic information. Computer Laboratory, University of Cambridge, Corn Exchange St. Cambridge CB2 3QG.

[Boha-91] Bohan, T. L. 1991 Computer-aided accident reconstruction: its role in court. SAE Technical paper Series (12p.)

[Bres-85] Bresnan J. and Kaplan R. 1985 The mental representation of grammatical relation, Lexical Functional Grammar. The MIT Press, 1985, pp 173-281.

[Bres-90] Bresnan J., Zaenen A. 1990 Deep Unaccusativity in LFG. The 5th Biannual Conference in Grammatical Relations.

[Carb-87] Carbonell J. 1987 Requirements for Robust NL Interfaces: The Language Craft and Xcalibur. Experiences Carnegie-Mellon University and Carnegie-Group Inc. Pittsburgh, PA 15213, USA.

[Caws-98] Cawsey A.. 1998 The Essence of Artificial Intelligence. Prentice Hall Europe 98 ISBN 0-13-571779-5.

[Chom-89] Chomsky N. 1989 Some Notes in Economy of Derivation and Representation. MIT Working Papers on Linguistics 1989, Vol. 10, pp 43-74.

[Clay-84] Clayton B. 1984 ART Programming primer Report Inference Corporation. Los Angeles, 1984 (ART)

[Dahl-84] Dahl V. 1984 On grouping grammars. Proceedings of the Second International Conference on Logic Programming pp 77-88, Ord and Form, Uppsala, Sweden 1984.

[Dowt-81] Dowty D., Wall R. and Peters S. 1981 Introduction to Montague Semantics. D. Reidel Pub. 1981.

[ElDa-94] Antoine El-Dahdah, 1994 El-Dahdah Encyclopaedia of Arabic Grammar: A dictionary of Arabic Grammar in Chart and Tables. Revised by G.M. Abdul - Massih, Librairie du Liban Publishers.

[Engd-86] Engdahl E. 1986 Constituent Questions: The Syntax and Semantics of Questions with Reference to Swedish Studies. In Linguistics and Philosophy, Reidel, Vol. 27, 1986.

[Evan-96] Evans D. and Zhai C. 1996. Noun-Phrase Analysis in Unrestricted Text for Information Retrieval. The Association for Computational Linguistics ACL-96.

[Farg-86] Farghal M. 1986 The syntax of wh-questions and related matters in Arabic Ph.D. thesis; Indiana University 1986.

[Fehr-89] Fehri F. 1989 Generalised IP Structure, Case, and VS Word Order. In Itziar Laka and Mahajan, A. (eds) MIT Working Papers in Linguistics, Vol. 10, pp 75-111, MIT Cambridge 1989

[Fehr-84] Fehri F. 1984 Agreement in Arabic, Binding and Coherence. In Agreement in Natural Language Approaches, Theories and Description.

[Fens-87] Fenstad J., Halvorsen E., Langholm P., and Van Benthem J. 1987 Situations, Language and Logic. Dordrecht:Reidel.

[Gazd-89] Gazdar G. and Mellish C. Natural Language Processing in Prolog. An Introduction to computational Linguistics, 1989, Addison Wesley.

[Gins-83] Ginsparg J. 1983 A robust portable natural language data base interface. In Conference on Applied NLP, Santa Monica, 1983.

[Groe-89] Groenendijk J. and Stokhof M. 1989 Type-shifting Rules and the Semantic of Interrogatives. In Chierchia, Partee and Turner eds., Properties, Types and Meaning. Vol. II, pp. 21-68, 1988, Kluwer, Dordrecht.

[Groe-90] Groenendijk J. and Stokhof M. 1990 Partitioning Logical Space. Department of Philosophy, Department of Computational Linguistics, University of Amsterdam, the Second European Summerschool on Logic, Language and Information Leuven.

[Gros-87] Grosz B., Appelt, Douglas, Martin, Paul and Pereira 1987 TEAM: An experiment in the design of transportable natural language interface. AI 1987, 32, pp. 173-243

[Gros-86] Grosz B., Sparck J., Webber B. (ed) 1986 Reading in Natural Language Processing. Morgan Kaufmann Pub. ISBN 0-934613-11-7.

[Gull-97] Gully A. 1997 Arabic Linguistic Issues and Controversies of the Late Nineteenth and Early Twentieth Centuries. Oxford University Press, pp 75-120.

[Hafn-85] Hafner C. and Godden K. 1985 Portability of Syntax and Semantics in DATALOG. In ACM Transactions of Office Information Systems, Vol. 3, pp. 141-164, 1985.

[Halv-83] Halvorsen P. K. 1983 Semantics for Lexical-Functional Grammar. Linguistic Inquiry 14, pp. 567-615.

[Halv-88a] Halvorsen P. K. and Kaplan R. M. 1988 Projections and Semantic Description in Lexical-Functional Grammar. To appear in Proceedings of the International Conference on Fifth Generation Computer Systems Tokyo, Japan.

[Hamb-73] Hamblin C. 1973 Questions in Montague English. Foundations of Language 10, pp. 41-53, Reprinted 1976 in Partee ed., Montague Grammar, Academic Press, NY, 1973.

[Harr-80] Harris L. 1980 Robot: A high performance NL Interface for DataBase Query Natural Language Based Computer Systems. Ed. Leonard Bolc, Carl Hanser Verlag, Munchen Wien, 1980, pp. 285-318.

[Haus-83] Hasser R. 1983 The syntax and semantics of English mood. In Kiefer ed., Question and Answering, Reidel, Dordrecht 1983.

[Hend-78] Hendrix G, Sacerdoti E. D., Sagalowicz D., Slocum J. 1978 Developing a NL Interface to complex data. Reading in the NLP ed. Grose B, Jones K., Webber B. 1978.

[Hend-81] Hendrix G. and Lewis W. 1981 Transportable Natural Language Interface to DBs. 19th Annual Meeting of the Assoc. for Computational Linguistics 1981 pp. 159-165.

[Hend-87] Hendrix G., and Walter B., 1987 The Intelligent Assistant Technical considerations involved in designing Q and A's natural-language interface. Byte Dec. 1987.

[Hole-95a] Holes C. 1995a Modern Arabic: Structure, Function and Varietion (Longman Linguistics Library). Longman, London 1995.

[Hole-95b] Holes C. 1995b The Structure and Function of Parallelism and Repetition in Spoken Arabic: A Sociolinguistic Study. Journal of Semitic 40 (1) : pp 57-81.

[Igni-91] Ignizio James P. 1991 Introduction to Expert Systems, the development and implementation of Rule-based Expert Systems. McGraw-Hill, Inc.

[Inte-96] Intellicorp Inc. 1996b, Kappa/OMW Product Information. World Wide Web pages on the Internet, 'http://www.intellicorp.com/'.

[John-91] Johnson M. 1991 Features and Formulae. Computational Linguistics, ACL, 1991, pp. 131.

[Kapl-79] Kaplan S J. 1979 Cooperative Responses from a Portable Natural Language Data Base Query System. Ph.D., dissertation, Dept. of Computer and Information Science, University of Pennsylvania, 1979.

[Kapl-82] Kaplan R. M. & Bresnan J. 1982 Lexical-Functional Grammar: A Formal System for Grammatical Representation. In Bresnan (Ed.) 1982a, pp173-281.

[Kapl-83] Kaplan S J. 1983 Cooperative responses from a portable natural language database query system In Computational Models of Discourse. Michael Brady abd Robert C. Berwick 167-208, MIT Press Cambridge, 1983.

[Kapl-88a] Kaplan, R.M., Maxwell, J.T.1988 An Algorithm for Functional Uncertainty. Colling 88, pp. 297-301.

[Kapl-88b] Kaplan R.M., Maxwell J.T. 1988 Constituent Coordination in Lexical-Functional Grammar. Colling 1988, pp. 303-305.

[Kapl-89] Kaplan R., Netter K., Wedekind, J., Zaenen A. 1989 Translation by Structural Correspondences. In Fourth Conference of the European Chapter of the Association for Computational Linguistics Manchester, UMIST, pp. 272-281.

[Kapl-89] Kaplan R., Zaenen A. 1989 Long-Distance Dependencies, Constituent Structure, and Functional Uncertainty. In Baltin, M.R., Kroch, A.S. (eds) Alternative Conceptions of Phrase Structure The University of Chicago Press, pp. 17-42.

[Kart-77] Karttunen L. Syntax and Semantics of Questions. Linguistics and philosophy 1977.

[Katz-98] Katz Boris, 1998. From Sentence Processing to Multimedia Information Access. Artificial Intelligence Laboratory, MIT Cambridge, MASS 02139. http:www.ai.mit.edu.

[Kay-80] Kay M. 1980 Algorithmic Schemata and data structure in syntactic. Processing Nobel Symposium on Text Processing, Stockholm, 1980

[Kaye-85] Kayed 1985 The influence of Arabic grammar on edited and non-edited English used by Arabs. Ph.D. thesis, Univ. of Missouri 1985.

[Kehl-84] Kehler T. and Clemenson G. 1984 An application development system for expert systems. Systems and Software, Vol 3, 1984.

[Kono-82] Konolige K. 1982 A First-Order formalisation of knowledge and action for a multi-agent planning system. Artificial Intelligence Center SRI International, USA, In Machine Intelligence 10 ed. Hages, Michie, Pao Horwood Ellis 1982.

[Kunz-84] Kunz J., Kehler T. and Williams M. 1984 Applications development using a hybrid AI development system. The AI Magazine Vol. 5, No. 3.

[Lena-91] Lenat D. and Feigenbaum E.A. On the thresholds of Knowledge. Artificial Intelligence 47 (1/3) pp. 185-250.

[Lena-95] Lenat D. CYC: A Large-Scale Investment in Knowledge Infrastructure. Communication of the ACM Vol. 38. No. 11, Nov. 1995.

[Mart-98] Martino A. and Nissan E. (eds.) 1998 Formal Approaches to Legal Evidence. Special Issue of Artificial Intelligence and Law (to appear).

[Mart-95] Martin J. 1995, Object-Oriented Methods: A Foundation. Prentice Hall, Englewood Cliffs, New Jersey 07632.

[McCo-82] McCord M. C. 1982 Using Slots and Modifiers in logic grammars for Natural Language. Computer Science Dept. University of Kentucky, Lexington, Artificial Intelligence Vol. 18 pp. 327-367 1982.

[Mehd-86] Mehdi 1986, Computer Interpretation of Arabic. Ph.D. Thesis, Exeter University.

[Mehd-88] Mehdi S. and Narayanan 1988 A linguistic model for the computer interpretation of Arabic. Department of Computer Science, old library, University of Exeter, Devon, England. 1988.

[Mehd-88] Mehdi S. and Narayanan 1988 Logic programming and inflectional language: an experiment with Arabic. Department of Computer Science, old library University of Exeter, Devon, England 1988.

[Peer-96] Peerbhai, M. 1996. Answer it: An Intelligent Query Engine for a Database. Final year project, CMS School, Greenwich University.

[Mins-75] Minsky M. 1975 A framework for representing knowledge The psychology of computer vision Ed. Patrick Heney Winston New York; McCraw-Hill, 1975, pp. 211-277.

[Mont-70c] Montague R. 1970c The proper treatment of quantification in ordinary English. R. Montague Formal Philosophy in Thomason (ed.) Yale University Press, New Haven 1974.

[Mont-74] Montague R. 1974 Selected papers of R. Montague. Formal Philosophy in Thomason (ed.) Yale University Press, New Haven 1974.

[Moor-79] Moore R. 1979 Natural Language Access to Database. Theoretical Technical Issues Artificial Intelligence Center, SRI International, 1979, Menlo Park, CA 94025.

[Nasi-96] Nasir M., Roochnik P., Shihadah M., and Yaghi M. Technical and Linguistic Issues in the Construction of a Test Corpus for an Arabic-to-English Machine Translation System. The 5th International Conference and Exhibition on Multi-lingual Computing, ICEMCO-96. Cambridge University, April 1996.

[Niss-98] Nissan E. 1998 (in press) Word Formation (in language and Computation), ~ 70 p. In both, the Encyclopaedia of Computer Science and Technology (A. Kent & J.G. Williams, eds.), and the Encyclopaedia of Library and Information Science (A. Kent, ed.) Marcel Dekker Pub., New York.

[Ouha-88b] Ouhalla J. 1988b The Syntax of Head Movement a Study of Berber. Doctoral Dissertation, UCL, London 1988.

[Patr-93] Patrick A., Locmelis W., and Whalen T. 1993. The Role of Previous Questions and Answers in Natural Language Dialogues with Computers. International Journal of Human-Computer Interaction.

[Pete-94] Peters G.A. and Peters B.J. 1994 Automotive Engineering and Litigation. (Wiley Law Publication). John Wiley and Sons, New York.

[Rich-84] Rich E. 1984 Natural Language Interfaces. IEEE Sep. 1984, pp. 39-47.

[Safr-93] Al-Safran S., and Aref M. Semantic for Arabic Sentences. The second International Conference on Arabic and Advanced Computer Technology. Casablanca, Morocco 1993.

[Scha-75] Schank R. 1975 The primitive ACTs of Conceptual dependency. In Advance Papers of Theoretical Issues in NLP Workshop MIT, Cambridge 1975.

[Scha-85] Scha Remko J.H. 1985 English words and databases: How to bridge the gap. Philips Research Laboratories, Eindhoven. The Netherlands.

[Schu-97] Schum D. 1997 Evidence Marshalling for Imaginative Fact Investigation. To appear in Martino A. and Nissan E. (eds.) Formal Approaches to Legal Evidence. Special Issue of Artificial Intelligence and Law (to appear).

[Schu-92] Schulz Marion and Schmidt Daniela 1992. Yes/No Questions with Negation: Towards Integrating Semantic and Pragmatics. GWAI-92 Advanced in Artificial Intelligence.

[Sell-85] Sells P. 1985. Lectures on Contemporary Syntactic Theories CSLI.

[Smit-91] Smith B. C. 1991 The owl and the Electric Encyclopaedia. Artificial Intelligence 47 (1/3) pp 251-288.

[Sriv-95] Srivastava A. and Rajaraman V. 1995 A Vector Measure for the Intelligence of a Question-Answering (Q-A) System. The IEEE 1995 pp 814.

[Stoc-95] Stock O., Strapparava C., and Zancanaro M. 1995 Explorations in a Natural Language Multimodal Information Access Environment. In the Proceedings of the 15th IJCAI, 1995.

[Tenn-86] Tennant H. 1986 The Commercial Application of NL Interfaces. Computer Science Center, Texas Instruments, 1986, Dallas, Texas.

[Thom-75] Thompson F. B and Thompson B. H. 1975 Practical Natural Language Processing: The REL system as prototype. In Advances in Computers IS, M. Rubinoff and M.C. Yovits, Eds., Academic Press, New York.

[Thom-83] Thompson B. and Thompson F. 1983 Introducing ASK, a simple knowledgeable system. In The Conference of Applied Natural Language Processing, CA, 1983.

[Tich-78] Tichy P. 1978 Questions, answers, and logic. American Philosophical Quarterly, 15, 1978.

[Trav-79] Travis D. 1979 Inflectional Affixation in Transformational Grammar Evidence from the Arabic Paradigm. Reproduced by the Indiana University Linguistics Club Lindley Hall 310 Bloomington, Indiana 47405, 1979.

[Tult-96] Tulti A. and Yahia Synthesis of Arabic Speech through Linear and Overlapped Concatenation. The 5th International Conference and Exhibition on Multi-lingual Computing, ICEMCO-96. Cambridge University, April 1996.

[Turn-88] Turner R. 1988 Properties, propositions and semantic theory. In Proc. of Formal Semantics and Computational Linguistics, Swiss, 1988.

[Walt-78] Waltz D. 1978 An English Language Question Answering system for a Large Relational Database. Artificial Intelligence Language Processing, University of Illinois at Urbana-Champaign, Communication of the ACM, July 1978 Vol., 21, No. 7. p. 526.

[Warr-82] Warren D. and Pereira F. 1982 An efficient easily adaptable system for interpreting natural language queries. AM. J. Computational Linguistics.

[Wile-81] Wilensky R. 1981 A knowledge-based Approach to Language Processing. Progress Report 7th-IJCAI, 1981 pp. 25-30.

[Will-84] Williams C. 1984 ART: The advanced reasoning tool Inference. Corp. Report, Inference Corp., 5300 W. Century Blvd. Los Angeles, 1984 .

[Wins-92] Winston H. Artificial Intelligence. 3ed ed. Addison-Wesley, 1992.

[Wood-72] Woods W, Kaplan R. and Nash-Webber B. 1972 The Lunar Sciences Natural Language Information System. Final Report, Report 3438, Bolt Beranek and Newman Inc. 1972.

[Wood-73] Woods W. 1973 An experimental Parsing system for transition network grammars. In ed. Randall Rustin, Natural Language Processing, 1973.

[Yama-94a] Yamani A. and Al-Zobaidie A. Long-Distance Dependencies and Coordination Phenomena in Arabic Interrogatives: an LFG Treatment. The Second International Conference on Artificial Intelligence, Cairo,1994.

[Yama-94b] Yamani A. and Al-Zobaidie A. Semantic Structures for Arabic Interrogatives and Coordinated Arabic Interrogatives: an LFG Treatment. The Second International Conference on Artificial Intelligence, Cairo, 1994

[Yama-98] Yamani A. & Al-Zobaidie A., 1998 Towards an Interrogative Language (IntLang) for the Constituents of Arabic Text Retrieval. (to appear) In Processing Arabic Journal, Report 9, Nijmegen 1998, Ditters E. (Ed.), Institute for the Languages & Cultures of the Middle East, - Nijmegen University Holland.

[Zanc-97] Zancanaro M., Stock O., Strapparava C. 1997 Multimodal Interaction for Information Access: Exploiting Cohesion. To appear on the Computational Intlligence.

World Wide Web Sites and Uniform Resource Locators (URL)

World Wide Web (WWW) Sites
and
Uniform Resource Locators (URL)

[URL 01] The CYC project WWW site:
http://www.mcc.com/projects/cyc/cyc.html

[URL 02] The Natural Language Processing and Communication Group WWW site:
http://ecate.itc.it:1024/

[URL 03] The Arabic language in the Arab Countries WWW sites:
http://www.liii.com/~hajeri/arab.html
http://www.sakhr.com/
http://arabia.com/

[URL 04] The Application Technology WWW site:
http://www.apptek.com/

[URL 05] The Intellicorp WWW site:
http://www.intellicorp.com/default.htm

[URL 06] The SAKHR Arabic Language WWW site:
http://www.sakhr.com/

[URL 07] The START Natural Language System - the MIT WWW site:
http://www.ai.mit.edu/publications/pubsDB/pubsDB/search

[URL 08] The Chat project WWW site:
http://debra.dgbt.doc.ca/chat/chat.html

[URL 09] The Natural Language Engineering WWW site:
http://www.cup.cam.ac.uk/Journals/JNLSCAT95/nle/nle.html

[URL 10] The LFG WWW site:
http://clwww.essex.ac.uk/LFG/

Appendix A

List of Publications

1. **Yamani A.** and Al-Zobaidie A., Long-Distance Dependencies and Coordination Phenomena in Arabic Interrogatives: An LFG Treatment, The Second International Conference on Artificial Intelligence, Cairo, **1994**, Pp 166-188.

2. **Yamani A.** and Al-Zobaidie A., Semantic Structures for Arabic Interrogatives and Coordinated Arabic Interrogatives: An LFG Treatment, The Second International Conference on Artificial Intelligence, Cairo, **1994**, Pp 189-205.

3. Al-Zobaidie A., Al-Khonaizi M., **Yamani A.,** al-A'ali M, Common Sense Knowledge Representation for Natural Language Processing: An LFG approach, in Processing Arabic Journal, Report 8, Nijmegen **1995**, Ditters E. (Ed.), Institute for the Languages & Cultures of the Middle East, Nijmegen University Holland, ISSN 0921-9145, Pp 47-78.

4. **Yamani A**. and Al-Zobaidie A., Computational Linguistics Approach to Question-Answering System for Arabic. In the 5th International Conference and Exhibition on Multi-lingual Computing, ICEMCO-96, Cambridge University, April **1996**.

5. Al-Khonaizi M., **Yamani A.,** Al-Zobaidie A. and al-A'ali M. A New Declaratives & Interrogatives Approach to Natural Arabic Text: A Computational Linguistics Approach, In the 5th International Conference and Exhibition on Multi-lingual Computing, ICEMCO-96, Cambridge University, April **1996**.

6. **Yamani A**. & Al-Zobaidie A., Towards an Interrogative Language for the Constituents of Arabic Text Retrieval, (To appear) in Processing Arabic Journal, Report 9, Nijmegen **1998**, Ditters E. (Ed.), Institute for the Languages & Cultures of the Middle East, -Nijmegen University Holland.

7. **Yamani A**. and Al-Zobaidie A., Natural Language Understanding for Question Answering System. In the Fourth IEEE International Conference Electronics, Circuits, and Systems, ICECS **1997**.

8. **Yamani A**. and Al-Zobaidie A., Interrogative Common-sense Domain Knowledge for Enhancing the Intelligence of Question - Answering System. (To appear) in the 6th International Conference and Exhibition on Multi-lingual Computing, ICEMCO-98, Cambridge University, April **1998**.

Appendix B

List of Newspaper Stories and Set of Questions

من جريدة الشرق الاوسط بتاريخ ٤\٥ \١٩٩٦

Taken from Alsharq Al-wsat Newspaper 5, April 1996

مقتل ٩ بحادث سير في المغرب **Morocco**

لقي تسعة ا شخاص حتفهم وجرح ١٦ اخرون من بينهم ثمانيه في حالـه خطـيرة في حـادث سـير وقـع علـى الطريق بين سطات و مراكش. واوضح بلاغ للجنة المغربية للوقايـة مـن حـوادث االسـير ان الضحـايـا االذيـن كانوا عائدين من احد اسواق المنطقة كانوا على متن الشاحنة التي انزلقت وانحرفت عن الطريق نتيجة انفجـار احد العجلات .

واضاف المصدر نفسه انه تم نقل الجرحى الى مستشفى السلامة بقلعة السرغنة.

تقديم التوقيت يزيد معدل الحوادث **Time**

جاء في دراسة عن حوادث الطرق ان تقديم التوقيت في محاولة لتوفير الطاقة االذي بدأ العمل به يوم الأحد في الولايات المتحدة وكندا تسبب في زيادة عدد الحوادث المرورية في اليوم التالي للعمل بالتوقيت الجديد.

واضهرت دراسة عن الحوادث المرورية في كندا عام ١٩٩١ وعام ١٩٩٢ حـدوث زيـادة فـي عـدد الحـوادث المرورية في اليوم التالي لتقديم التوقيت بنسبة ثمانية في المئـة. ويعـزو سـتانلي كوريـن مـن جامعـة كولومبيـا البريطانية هذه الزيادة الى الساعة التي خصمت من ساعات اليوم.

وكتب كورين في رسالة نشرت في مجلة (نيوانغلآند) الطبية "تلك البيانات تظهر ان أي تغير بسيط في ساعات النوم يمكن ان يؤدي الى عواقب وخيمة في النشطة اليومية".

واظهرت الدراسة التي شملت ٢١٦٠٣ حادث في كندا عـامي ١٩٩١ و ١٩٩٢ انـة فـي الخريـف حيـن تؤخـر عقارب الساعة وتزيد فترة النوم ساعة اضافية ينخفض معدل الحوادث فـي اليـوم التـالي لبـدء العمـل بـالتوقيت الجديد بنسبة ثمانية في المئة.

Sunderland سندرلاند

قتلت طفلة بريطانية تدعى ناتالي تبلغ السادسة من العمر عندما انحرفت سيارة عن الطريق تعطل كابحها باتجاهها ودهستها امام مرأى من والدتها فيما كانت الطفله والام تنتظران وصول الباص في موقف للباصات في سندرلآند شمال شرق انجلترا.

وقالت الشرطة ان سائق السيارة وهي من نوع فورد فيستا فقد السيطرة على عربتة بعدما تعطل كابحها ؛ فدهست السيارة الفتاة ثم اصطدمت بمظلة موقف الباص الذي هبط على الطفلة جاعلا امكانية انقاذها مستحيلا.

وحاولت والدة الفتاة ماندي (٢٦عاما) ان تنقذها عندما رأت السيارة انحرفت عن الطريق العام وتتوجه نحو موقف الحافلات ولكنها لم تجد الوقت الكافي لالتقاط الطفلة التي كانت قد اختفت تحت عجلات السيارة خلال بضع ثوان.

وأضافت الشرطة ان سكان المنطقة الذين سمعوا الصراخ وارتطام المعدن والزجاج هرعوا الى مكان الحادث واتصلوا برجال الاسعاف ولكن ناتالي كانت قد فارقت الحياة قبل نقلها الى المستشفى.

واشارة الشرطة الى ان والدة ناتالي واخاها الذي شاهد الحادث ايضا نقلا الى المستشفى للعلاج من جراء الصدمة التي اصابتهما.

London لندن

في شارع اكسفورد بوسط لندن حصل حادث مروع لسيارة رقم OAHAM-28 وقد نجم عن الحادث وفاة شخص يعتقد بأنه السائق وذلك لتهشم السيارة من الجانب الايمن.

غير ان الشخص الذي كان في الجانب الأيسر خرج من السيارة من غير جروح تذكر.

Runaway Does Major Damage - NEWS Shopper Wed. Nov. 16 1994

عصف، وقوع حادثين كبيرين، بسكون شارع سكني هاديء، وذلك في غضون ساعتين من الاسبوع الماضي. ولا زال احد السائقين بالمستشفى في وضع حرج يعاني من اصابات في الرأس بينما تعرض عدد من السيارات وحافلة لنقل الركاب لأضرار كبيرة او انها تلفت تماما ,. كما تهدم الحائط الامامي ورواق احدى المنازل، وقد وقعت هذه المأساة في اوكيهامبتون كرسنت .بمنطقة وبلنج في الساعة ٦:٤٥ مساء من يوم الاربعاء وذلك عندما اصطدمت سيارتان وجها لوجه احداهما من نوع اوستي ايجرو والاخرى من نوع سيرا كوسورث .هذا وقد اخرج سائق سيارة الاوستي (انتوني هيل) والذي يبلغ من العمر ٢٦ عاما، من سكان ريكلمارش رود .بمنطقة بلاك هيث، من سيارته بواسطة قوة اطفاء منطقة بلمستيد وذلك بقطع اجزاء من سيارته ليتمكن من الخروج من بين حطامها ,,. ومن ثم اخذ الى المستشفى (كوين ميري) .بمنطقة سيدكب، ثم نقل بعد ذلك الى قسم "العناية المركزة " حيث ذكر يوم الاثنين بأنه خرج من غرفة العناية المركزة وان صحته في تحسن .... اما سائقة سيارة السيرا (جولي اندروود) وهي من سكان نور شميرلاند افنيو .بمنطقة ديلنج فتعاني من اصابات في الوجه، وقد أخرجت من مستشفى (كوين ميري) يوم الجمعة الماضي. كما ان اربعة اطفال كانوا داخل السيارة اصيبوا باصابات طفيفة. وبعد ذلك بساعتين فقط وفي اثناء وجود الشرطة في مكان الحادث لمعاينته تدحرجت حافلة لنقل الركاب الى الخلف على طريق ايكسهوث بعد تعطل فراملها واصطدمت بعدة سيارات واصطدمت .بمؤخرة سيارة شرطة سرية وتحركت الحافلة عبر الشارع حيث هشمت حائط امامي ثم اصطدمت بسيارة كانت تقف على مدخل جانبي ثم اصطدمت بالمنزل مما ادى الى تهدم الرواق وقد تعرض اثنان من رجال الشرطة الى اصابات خفيفة، اما سائق الحافلة وهو من سكان بير كشير فلم يصب بأذى، وتقول الشرطة انه من حسن الحظ ان ثلاثين طفلا من مدرسة .بمنطقة ميدلسيكس كان قد تم اخلائهم من الحافلة في وقت مبكر من اليوم بعد اكتشاف عطل ميكانيكي فيها وقد وجهت الشرطة نداء الى شهود عيان للحادث الاول كما قالت ان كل العربات التي كانت في الحادث قد تم فحصها وكل من لديه معلومات عليه الاتصال برقيب الشرطة (سوندرز) على هاتف رقم (١١٢١٢-٣-١٨٠).

Runaway Does Major Damage - NEWS Shopper Wed. Nov. 16 1994

A quiet residential street was shattered by two major road accidents in the space of two hours last week. One driver is still seriously ill in hospital with head injuries while a number of cars and a coach were either written off or badly damaged, and the front wall and porch of one house were demolished. The drama began in Okehampton Crescent, Welling, at 6.45pm last Wednesday when an Austin Allegro and a Sierra Cosworth crashed head-on. The driver of the Allegro, Anthony Hill, aged 26 of Wricklemarsh Road, Blackheath, had to be cut out of the wreckage of his car by Plumstead firefighters. He was taken to Queen Mary's hospital, Sidcup, and then transferred to the Brook, where he was said on Monday to be out of intensive care and improving. The driver of the Cosworth, Julie Underwood of Northumberland Avenue, Welling, suffered facial injuries, and was released from Queen Mary's last Friday. Four children in the car suffered minor injuries. Just two hours later, while police were still on the scene dealing with the accident, a runaway coach rolled backwards down Exmouth Road after the brakes failed. It hit a number of cars, smashed into the back of an unmarked police car, careered across the road, smashed through a front wall, hit a car parked in the driveway and then hit the house, demolishing the porch. Two officers in the police car suffered whiplash injuries, but the coach driver from Berkshire was unhurt. Police say that fortunately 30 children from a school in Middlesex had been taken off the coach earlier in the day after a mechanical fault had been discovered. Police are appealing for witnesses to the first accident and say all the vehicles involved are now being examined. Anyone who can help should telephone Police Sergeant Saunders on 081 301 1212.

The following is list of 51 interrogatives.

1. Question type Ayna اين

اين الحادث ayna alhadeth

اين هو الحادث ayna howa alhadeth.

اين هي الحادث * ayna heia alhadeth

اين وقع الحادث ayna wagga alhadeth

اين وقعت الحادث * ayna waggat alhadeth

*اين وقع الحادثة ayna wagga alhadetha

اين وقعت الحادثة ayna waggat alhadetha

اين يوجد الحادث ayna ujad alhadeth

اين يوجد الشارع السكني ayna ujad alshara alskanny

اين يوجد الشارع السكني و اين وقع الحادث ayna ujad alshara alskanny wa ayna wagga alhadeth

اين الشارع السكني ayna alshara alskanny

اين الشارع الهادىء ayna alshara alhadi

اين مرتكب الحادث ayna murtakeb alhadeth

اين جرح سائق السياره رقم ا ب ت ** ayna joriha saeeg alseara raggam ABC

اين الشرطة ayna alshorta

اين الاطفائين ayna aletfaeen

2. Question type Mata متى

متى ابلغ عن الحادث mata ablagga aan alhadeth

متى وقع الحادث mata wagga alhadeth

متى واين وقع الحادث mata wa ayna wagga alhadeth

متى وقع الحادث ومتى الاصتدام mata wagga alhadeth wa mata alestdam

متى وقع الحادث و اين وقع الحادث mata wagga alhadeth wa ayna wagga alhadeth

متى حضرت الشرطة mata hatharat alshorta

متى حضر الطفائين mata hathara aletfaeen

متى حضر الاسعاف mata hathara alessaf

3. Question type Who كيف

كيف وقع الحادث kefa wagga alhadeth

4. Question type Is هل

هل وقع الحادث hal wagga alhadeth

هل وقع حادث hal wagga hadeth

هل قتل سائق السيارة رقم hal gutila saeeg alseara raggam OAHAM-28

5. Question type Who من

من ابلغ عن الحادث maan ablagga aan alhadeth

من مسبب الحادث maan mosabebe alhadeth

من هو مسبب الحادث maan howa mosabebe alhadeth

*من هي مسبب الحادث maan heia mosabebe alhadeth

من المسبب maan almusabeen

من هم المصابين maan hom almusabeen

من هم كل المصابين maan hom kul almusabeen

6. Question type Why لماذا

لماذا وقع الحادث lematha wagga alhadeth

7. Question type ما

ما هو سبب وقوع الحادث ma howa sabab waggooa alhadeth

*ما هي سبب وقوع الحادث ma hea sabab waggooa alhadeth

ما هي نتائج وقوع الحادث ma hea nataeag waggooa alhadeth

*ما هو نتائج وقوع الحادث ma howa nataeag waggooa alhadeth

ما هي امكانات تجنب الحادث ma heia imkanat tajanob alhadeth

ماعدد الشرطة ma adad alshorta

8. Question type fe في

في أي شارع وقع الحادث fe ayy shara wagga alhadeth

في أي تاريخ وقع الحادث fe ayy tareeh wagga alhadeth

9. Question type Kam كم

كم استغرق الحادث kam istagraga alhadeth

كم المصابين kam almusabeen

كم عدد القتلى kam adad algatla

كم عدد المصابين kam adad almusabeen

كم الشرطة kam alshorta

كم عددالشرطة kam adad alshorta

كم عدد المصابين في الحادث kam adad almukteen fe alhadeth

كم عدد الاطفال الذين كانوا في سيارة جولي kam adad alatfal allatheena kano fe searat joly

كم كان عدد الاطفال الذين كانوا في سيارة جولي kam kana adad alatfa allatheena fe searat joly

kam adad alrukab allatheena kano fe alseara raggm PARXXX كم عدد الركاب الذين كانوا في
السيارة رقم PARXXX

kam kana adad alrukab allatheena kano fe alseara raggm PARXXX كم كان عدد الركاب
الذين كانوا في السيارة رقم PARXXX

Appendix C

Examples of Application Runs

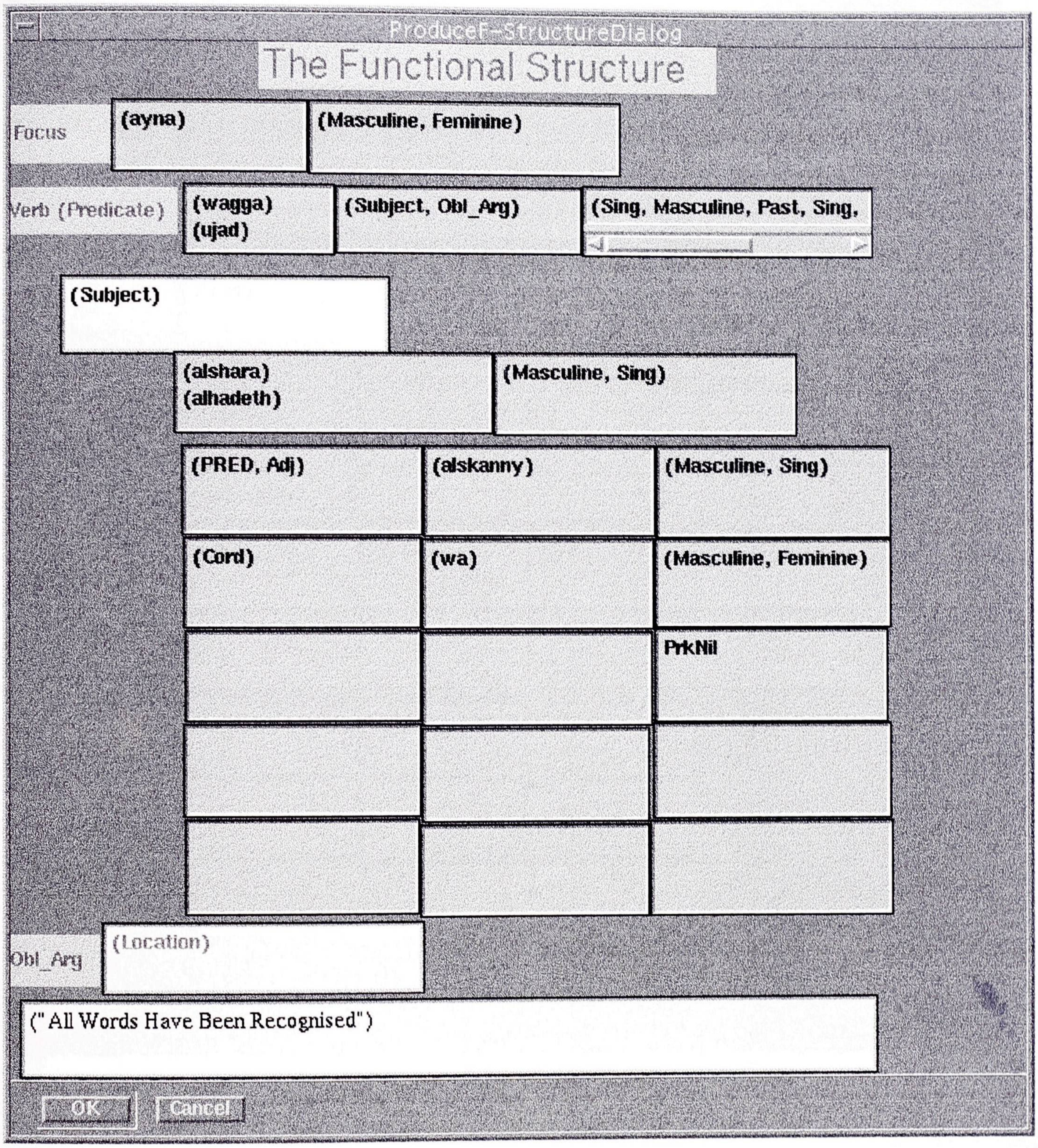
We have experimented with different types of questions. There are more than two hundred question set-ups in order to test the prototype.

This appendix provides some selected interrogatives. For each interrogative, an F-Structure, S-Structure, K-Structure, are given including the answer. The following are interrogatives taken from the list of questions in appendix B.

1. Where is the residential street and where did the accident happen?

(ayna ujad alshara alskanny wa ayna wagga alhadeth أين يوجد الشارع السكني و أين وقع الحادث)

1.1 The Functional Structure Presentation:



(ayna) | (Masculine, Feminine)

(wagga)
(ujad) | (Subject, Obl_Arg) | (Sing, Masculine, Past, Sing,

(Subject)

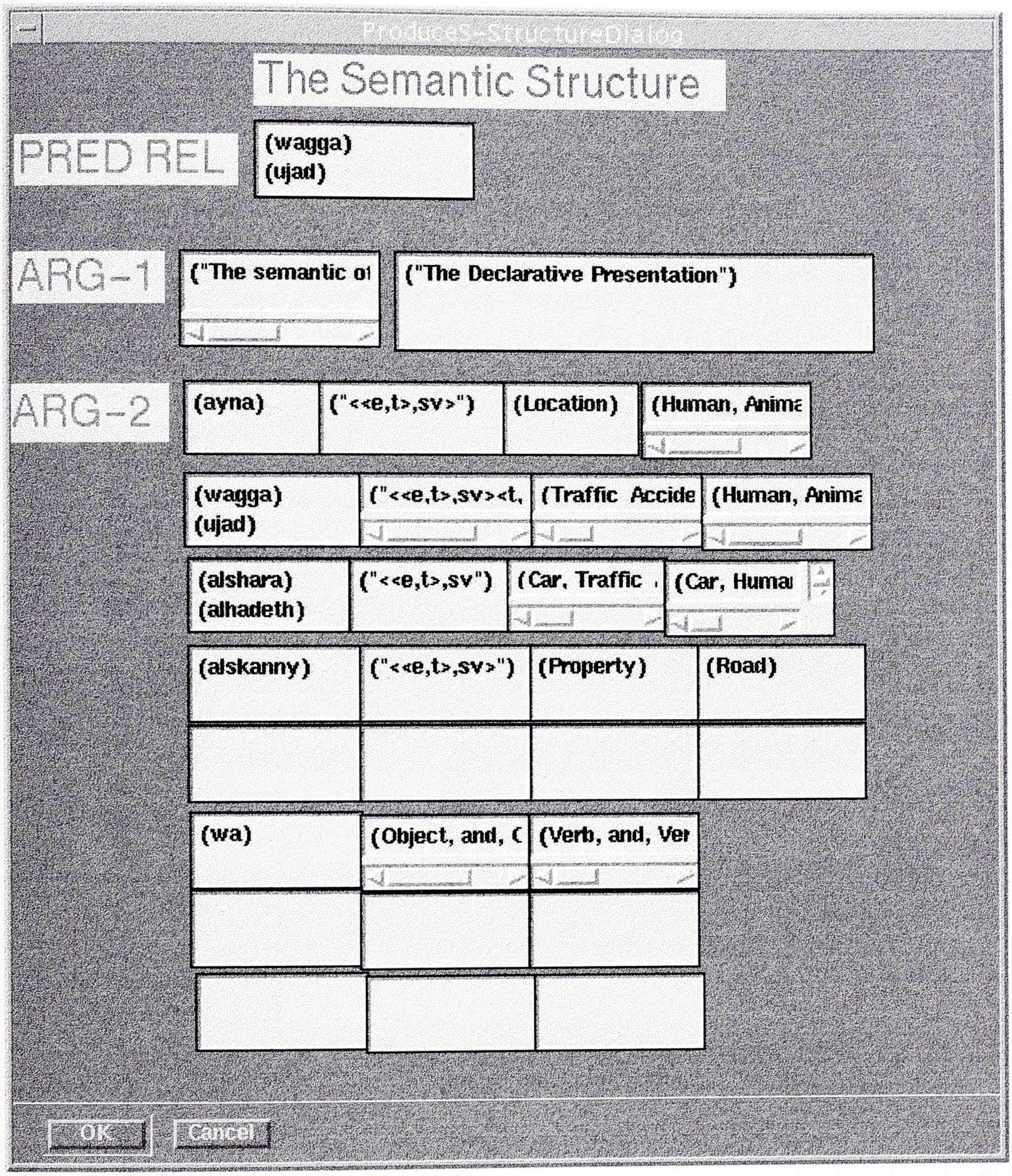
(alshara)
(alhadeth) | (Masculine, Sing)

(PRED, Adj)	(alskanny)	(Masculine, Sing)
(Cord)	(wa)	(Masculine, Feminine)
		PrkNil

(Location)

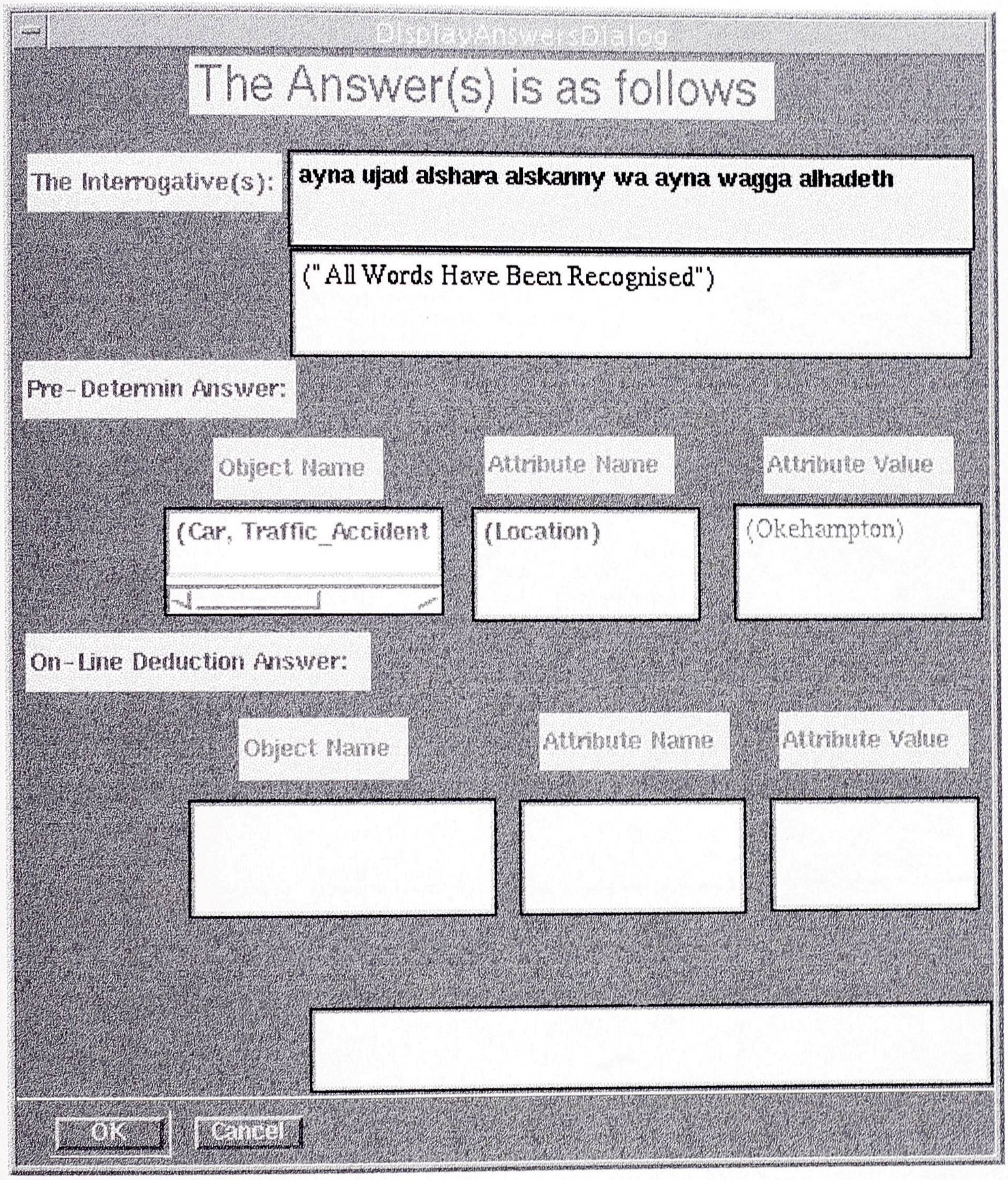
("All Words Have Been Recognised")

1.2 The Semantic Structure Presentation:



PRED REL	(wagga) (ujad)			
ARG-1	("The semantic o1	("The Declarative Presentation")		
ARG-2	(ayna)	("<<e,t>,sv>")	(Location)	(Human, Anima
	(wagga) (ujad)	("<<e,t>,sv><t,	(Traffic Accide	(Human, Anima
	(alshara) (alhadeth)	("<<e,t>,sv")	(Car, Traffic ,	(Car, Huma
	(alskanny)	("<<e,t>,sv>")	(Property)	(Road)
	(wa)	(Object, and, (	(Verb, and, Ver	

1.3 The Answer Presentation:



The Interrogative(s): ayna ujad alshara alskanny wa ayna wagga alhadeth

("All Words Have Been Recognised")

Pre-Determin Answer:

Object Name: (Car, Traffic_Accident

Attribute Name: (Location)

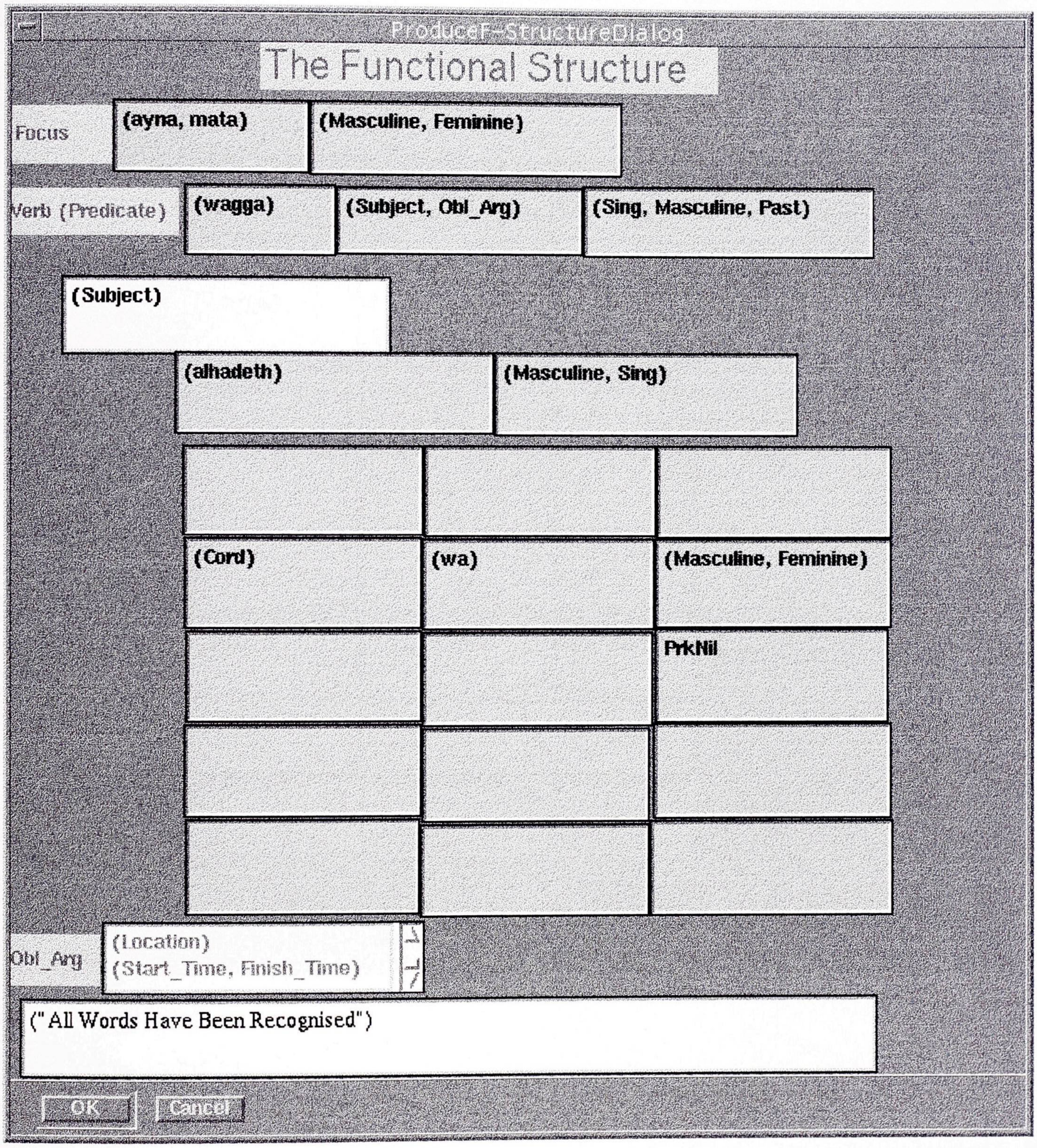
Attribute Value: (Okehampton)

2. When and where did the accident happen?

When did the accident happen and where did the accident happen?

(mata wagga alhadeth wa ayna wagga alhadeth متى وقع الحادث و أين وقع الحادث)

2.1 The Functional Structure Presentation:



The Functional Structure

Focus (ayna, mata) (Masculine, Feminine)

Verb (Predicate) (wagga) (Subject, Obl_Arg) (Sing, Masculine, Past)

(Subject)

(alhadeth) (Masculine, Sing)

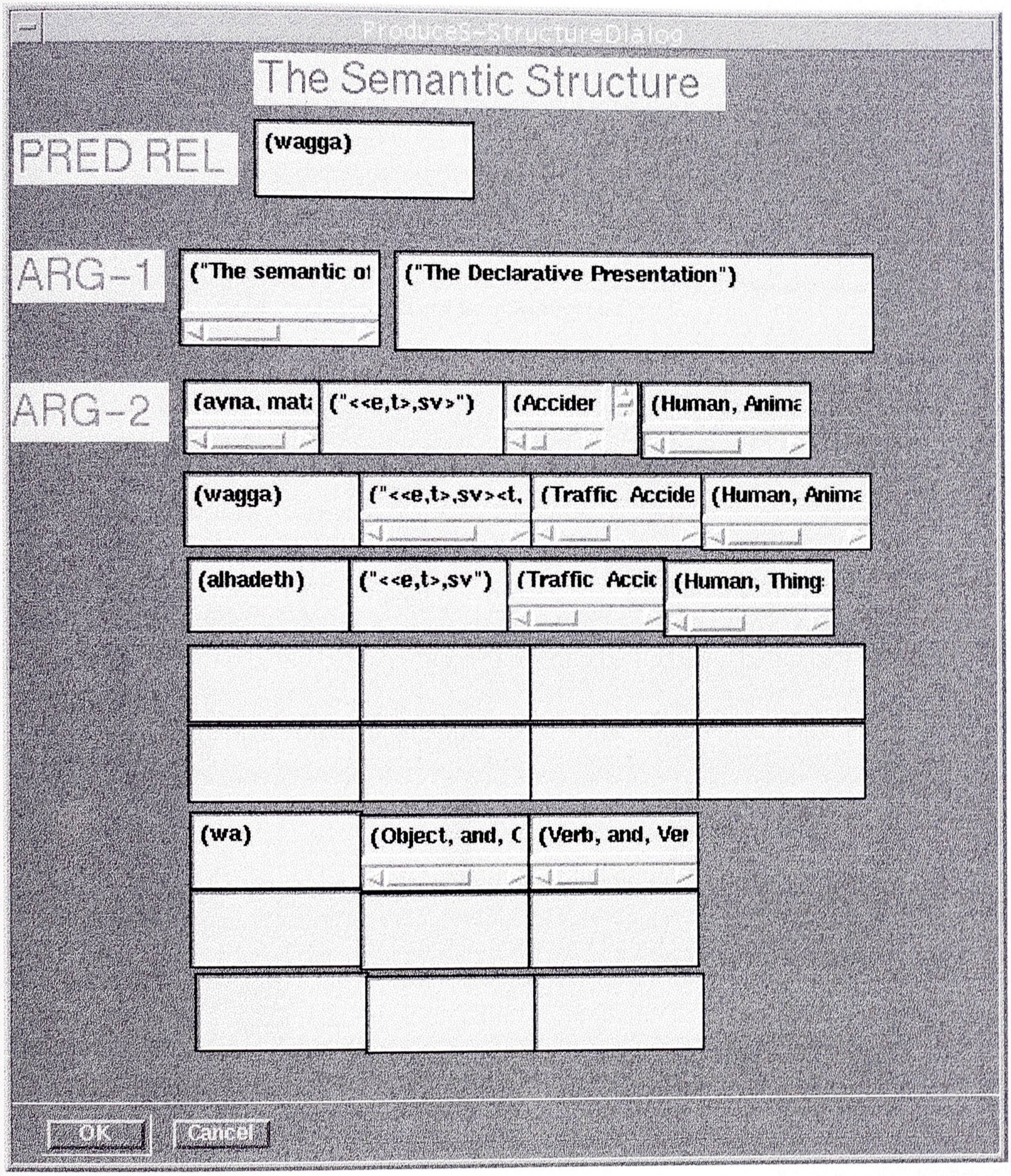
(Cord) (wa) (Masculine, Feminine)

PrkNil

Obl_Arg (Location)
(Start_Time, Finish_Time)

("All Words Have Been Recognised")

2.2 The Semantic Structure Presentation:



The Semantic Structure

PRED REL (wagga)

ARG-1 ("The semantic ot ("The Declarative Presentation")

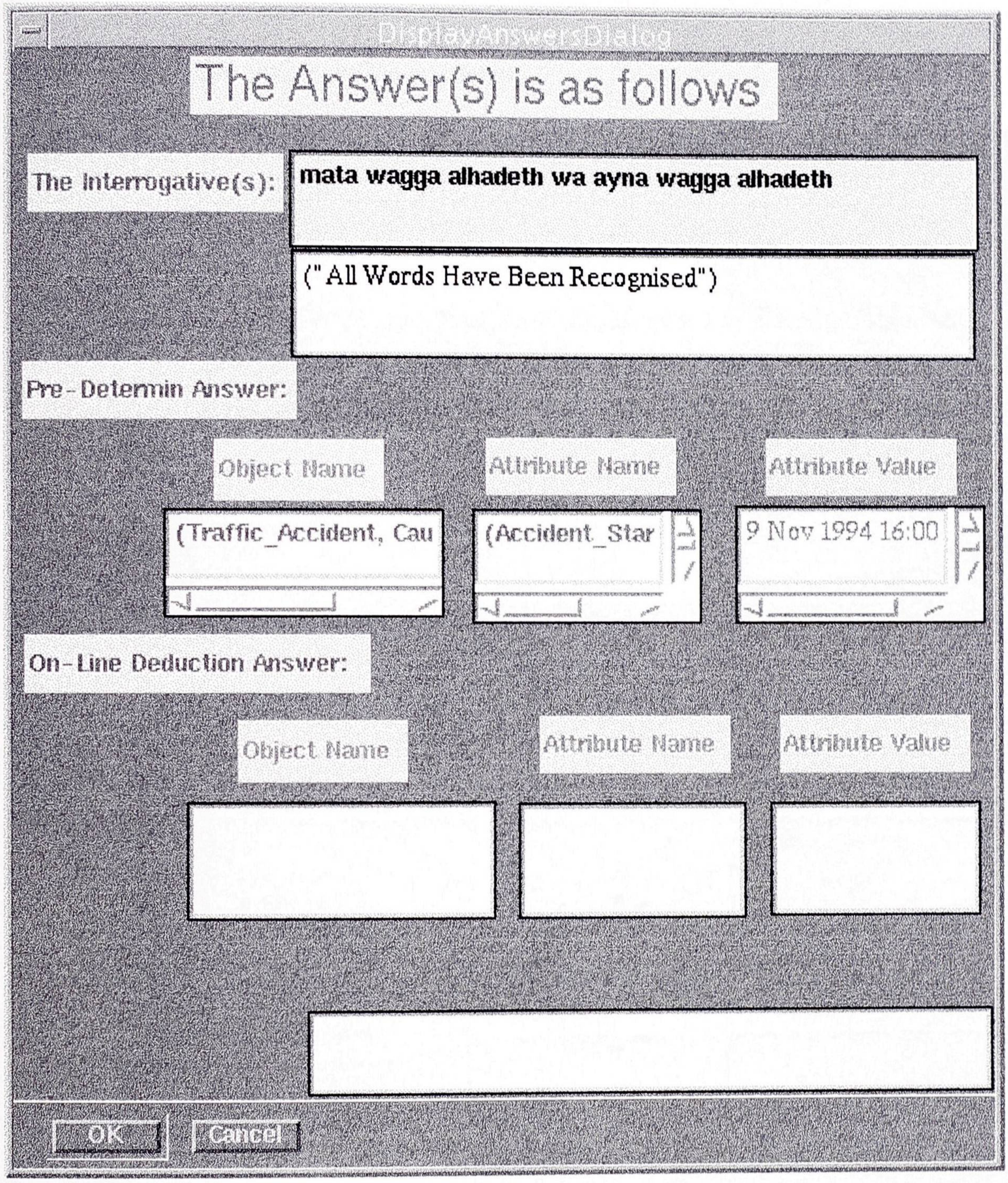
ARG-2 (avna, mat: ("<<e,t>,sv>") (Accider (Human, Anima

(wagga) ("<<e,t>,sv><t, (Traffic Accide (Human, Anima

(alhadeth) ("<<e,t>,sv") (Traffic Accic (Human, Thing:

(wa) (Object, and, ((Verb, and, Ver

2.3 The Answer Presentation:



DisplayAnswersDialog

The Answer(s) is as follows

The Interrogative(s): **mata wagga alhadeth wa ayna wagga alhadeth**

("All Words Have Been Recognised")

Pre-Determin Answer:

Object Name: (Traffic_Accident, Cau

Attribute Name: (Accident_Star

Attribute Value: 9 Nov 1994 16:00

On-Line Deduction Answer:

Object Name

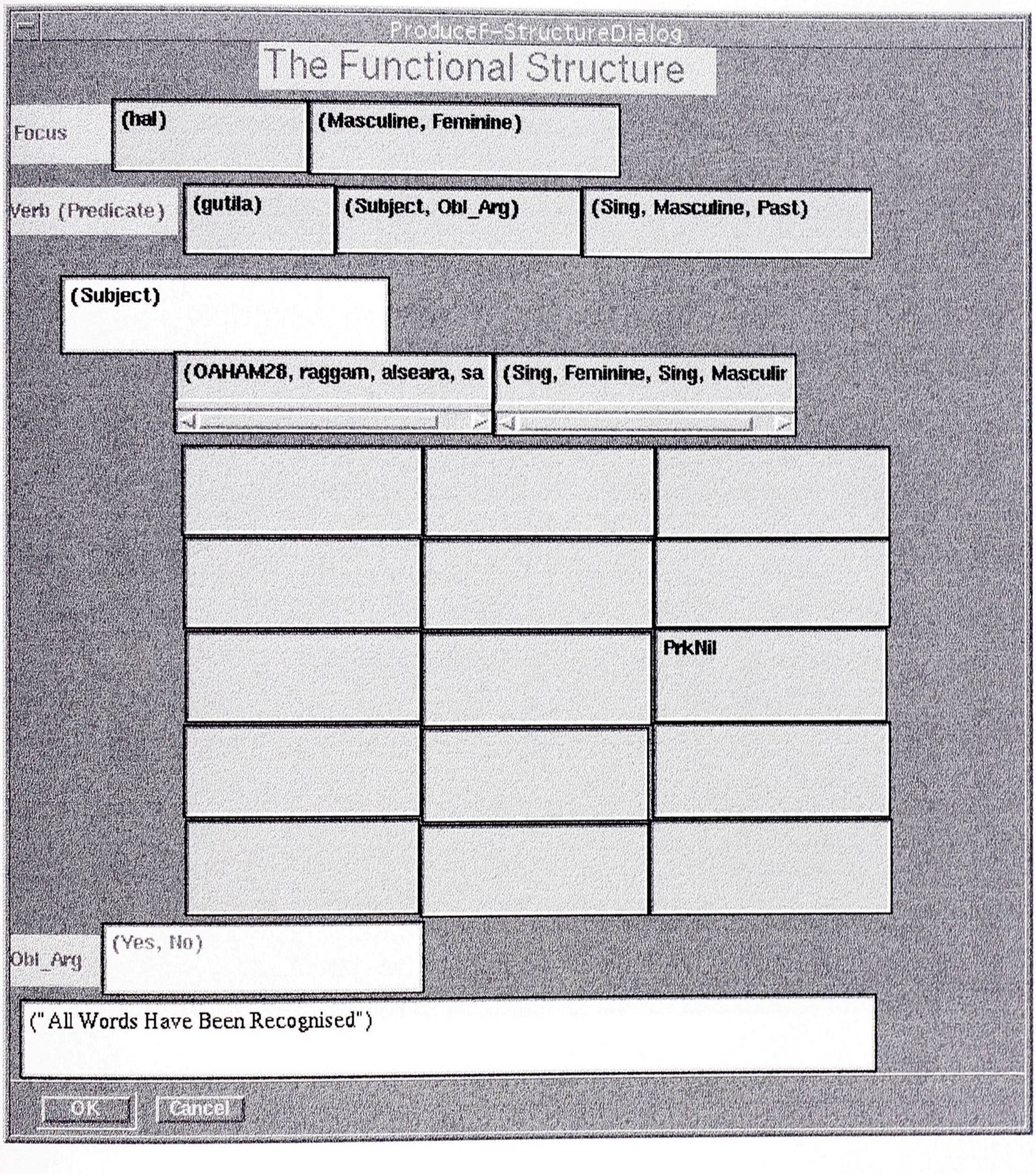
Attribute Name

Attribute Value

3. Did the driver of the registration number OAHAM-28 get killed?

(hal gutila saeeg alseara raggam OAHAM-28 هل قتل سائق السيارة رقم)

3.1 The Functional Structure Presentation:



The Functional Structure

Focus (hal) (Masculine, Feminine)

Verb (Predicate) (gutila) (Subject, Obl_Arg) (Sing, Masculine, Past)

(Subject)

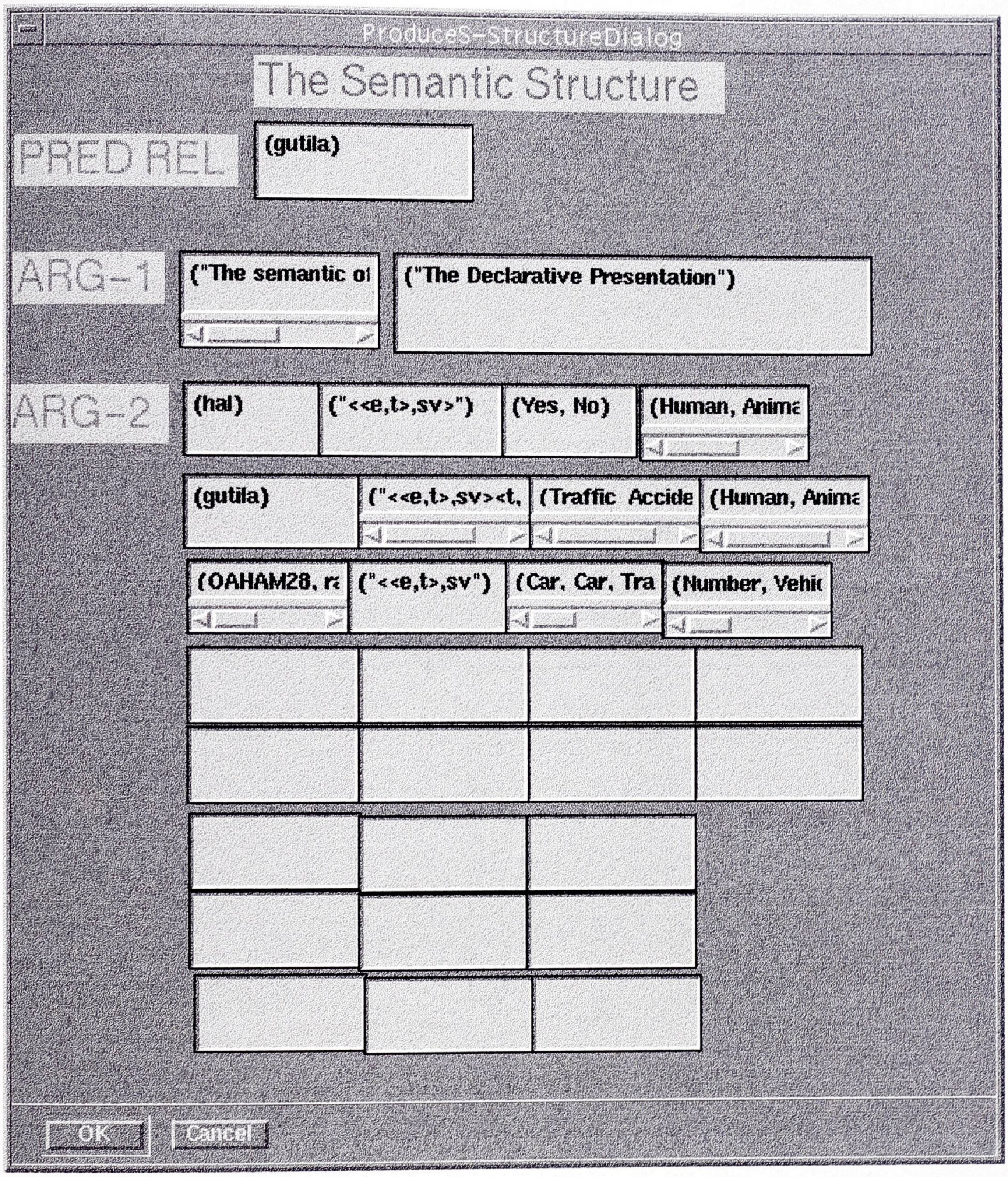
(OAHAM28, raggam, alseara, sa (Sing, Feminine, Sing, Masculir

PrkNil

Obl_Arg (Yes, No)

("All Words Have Been Recognised")

3.2 The Semantic Structure Presentation:



The Semantic Structure

PRED REL (gutila)

ARG-1 ("The semantic o ("The Declarative Presentation")

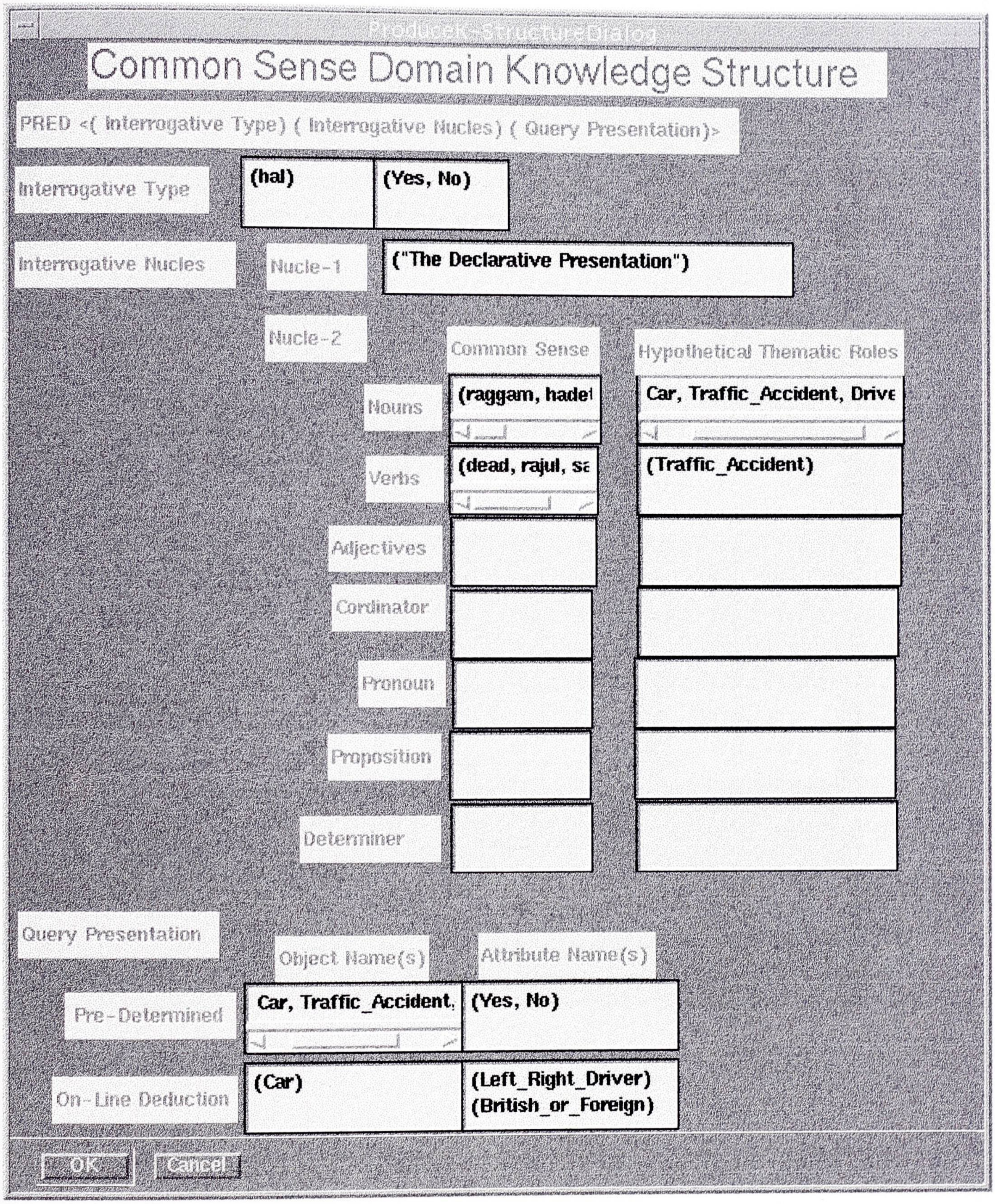
ARG-2

(hal) ("<<e,t>,sv>") (Yes, No) (Human, Anim

(gutila) ("<<e,t>,sv><t, (Traffic Accide (Human, Anim

(OAHAM28, r ("<<e,t>,sv") (Car, Car, Tra (Number, Vehi

3.3 The Knowledge Structure Presentation:



Common Sense Domain Knowledge Structure

PRED <(Interrogative Type) (Interrogative Nucles) (Query Presentation)>

Interrogative Type: (hal) (Yes, No)

Interrogative Nucles

Nucle-1: ("The Declarative Presentation")

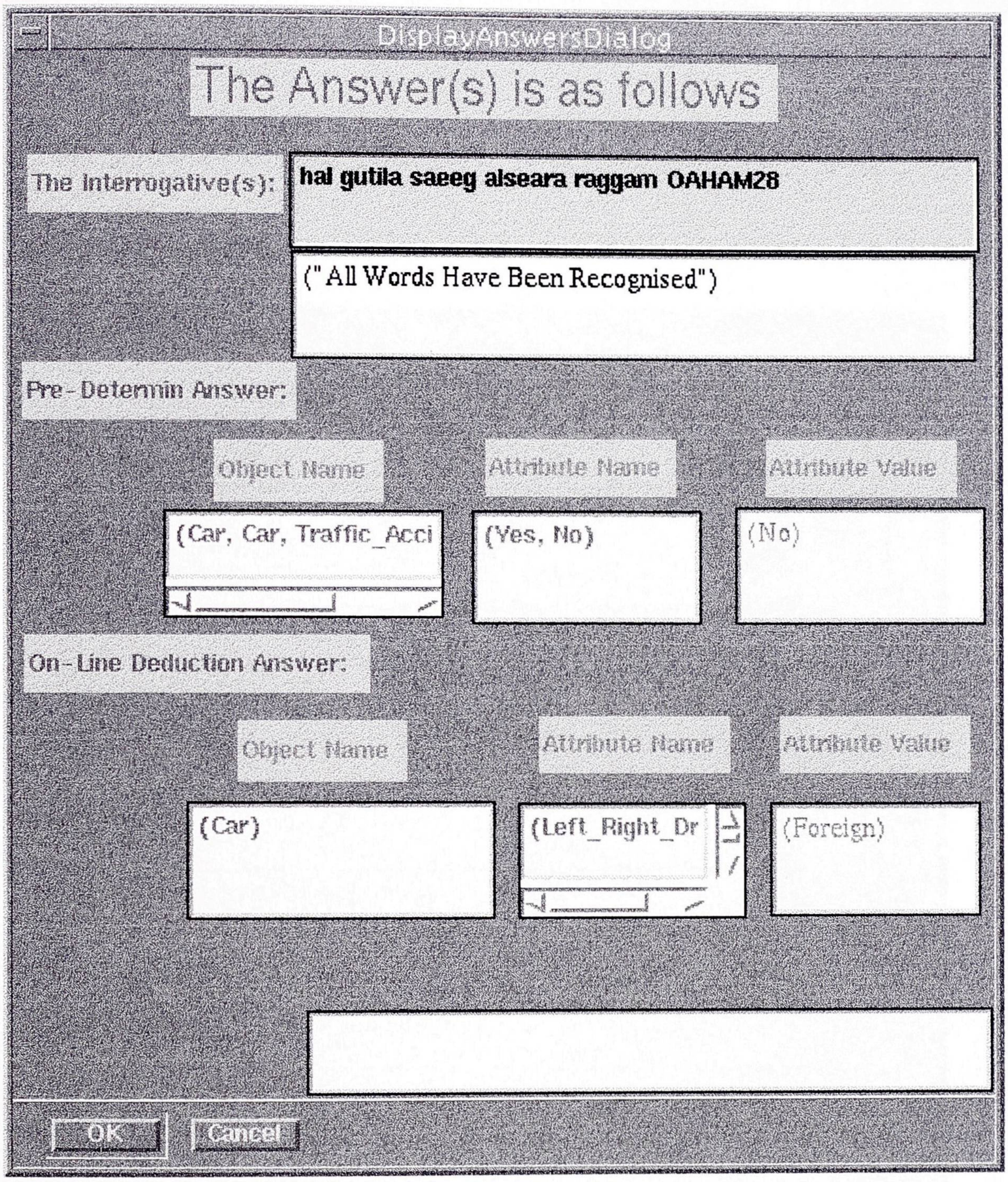
Nucle-2

	Common Sense	Hypothetical Thematic Roles
Nouns	(raggam, hadel	Car, Traffic_Accident, Drive
Verbs	(dead, rajul, sa	(Traffic_Accident)
Adjectives		
Cordinator		
Pronoun		
Proposition		
Determiner		

Query Presentation

	Object Name(s)	Attribute Name(s)
Pre-Determined	Car, Traffic_Accident,	(Yes, No)
On-Line Deduction	(Car)	(Left_Right_Driver) (British_or_Foreign)

3.4 The Answer Presentation:



hal gutila saeeg alseara raggam OAHAM28

("All Words Have Been Recognised")

(Car, Car, Traffic_Acci

(Yes, No)

(No)

(Car)

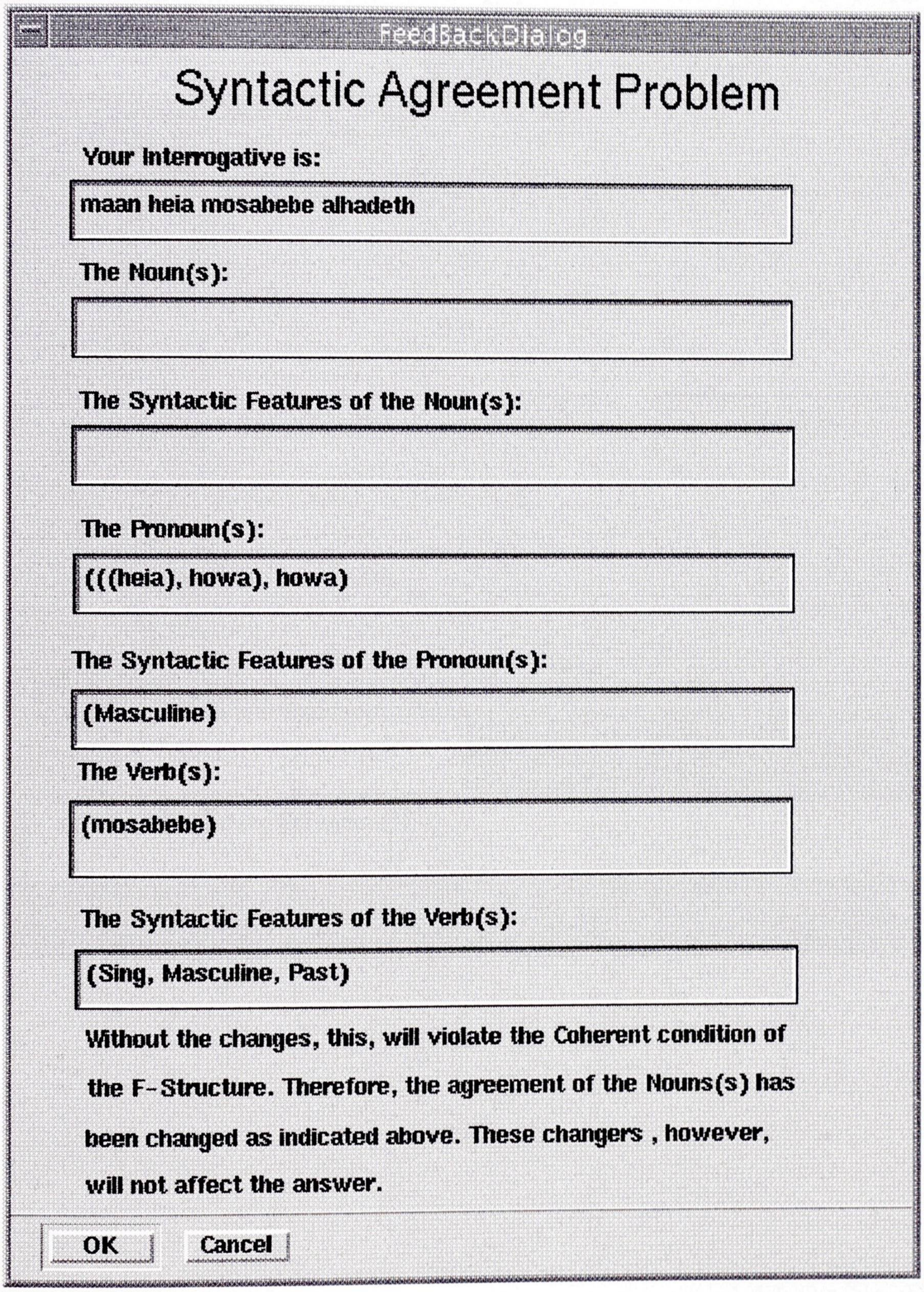
(Left_Right_Dr

(Foreign)

Question number four has an Arabic agreement problem between the pronoun and the verb i.e. between masculine and feminine. The pronoun is given for the feminine, where the verb is for masculine. The system solves this problem and gives feedback to the user on where about the problem has occur.[1]

4. Who caused the accident to happen? (maan heia mosabebe alhadeth *من هي مسبب الحادث)

The Parser feedback:



Syntactic Agreement Problem

Your Interrogative is:
maan heia mosabebe alhadeth

The Noun(s):

The Syntactic Features of the Noun(s):

The Pronoun(s):
(((heia), howa), howa)

The Syntactic Features of the Pronoun(s):
(Masculine)

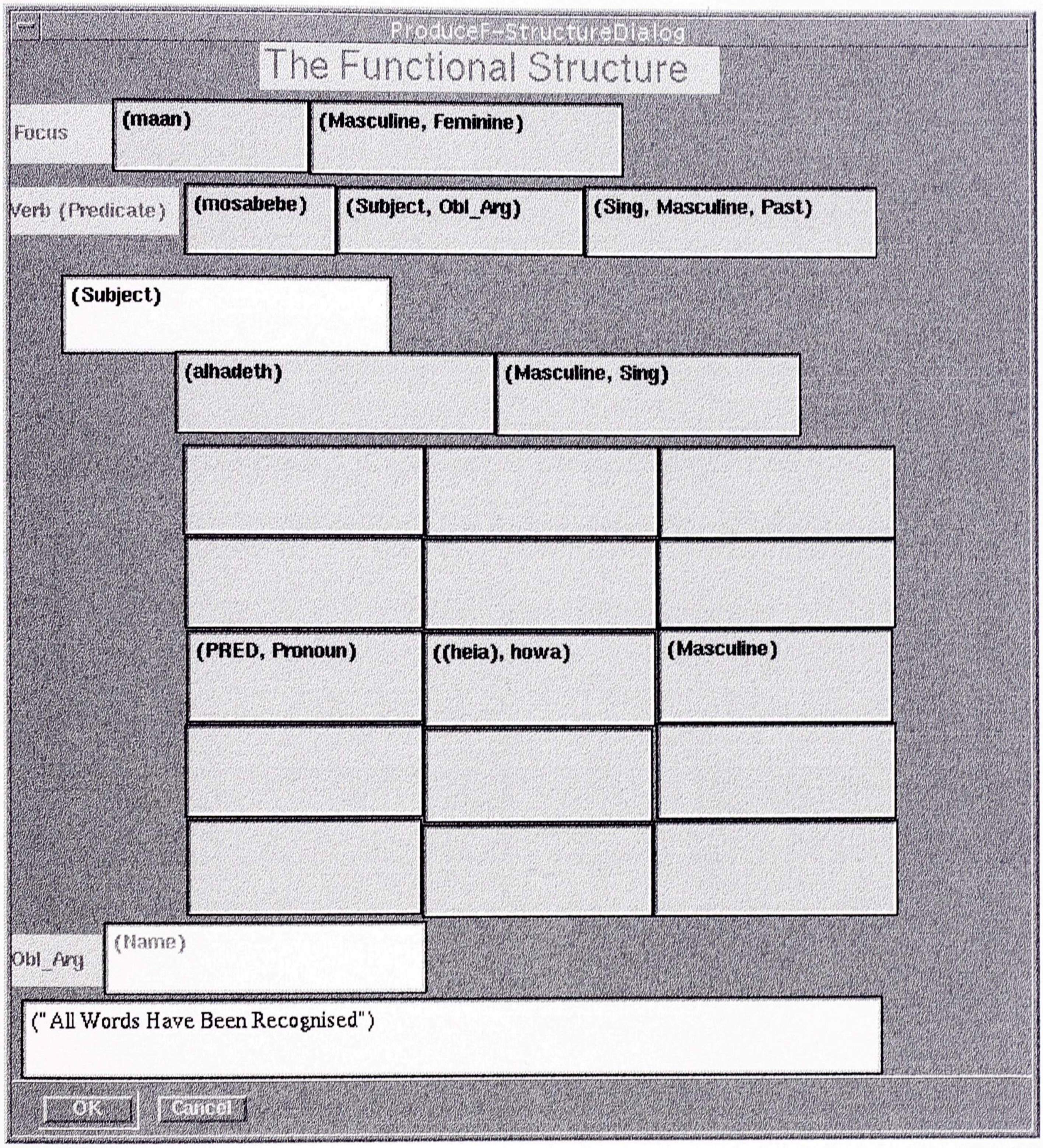
The Verb(s):
(mosabebe)

The Syntactic Features of the Verb(s):
(Sing, Masculine, Past)

Without the changes, this, will violate the Coherent condition of the F-Structure. Therefore, the agreement of the Nouns(s) has been changed as indicated above. These changers , however, will not affect the answer.

[1] The (*) indicates that this interrogative is invalid.

4.1 The Functional Structure Presentation:



Focus (maan) (Masculine, Feminine)

Verb (Predicate) (mosabebe) (Subject, Obl_Arg) (Sing, Masculine, Past)

(Subject)

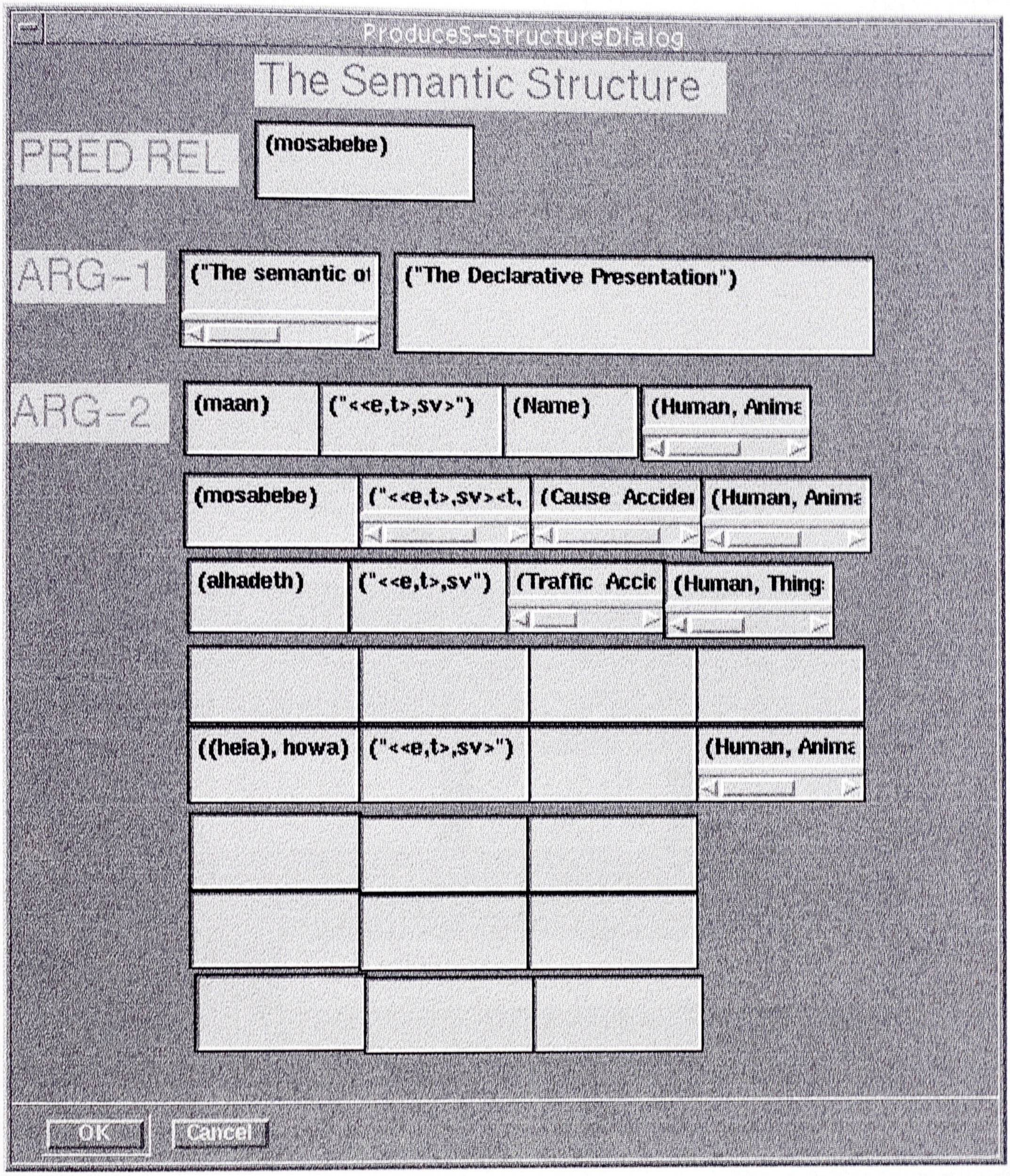
(alhadeth) (Masculine, Sing)

(PRED, Pronoun) ((heia), howa) (Masculine)

Obl_Arg (Name)

("All Words Have Been Recognised")

4.2 The Semantic Structure Presentation:



The Semantic Structure

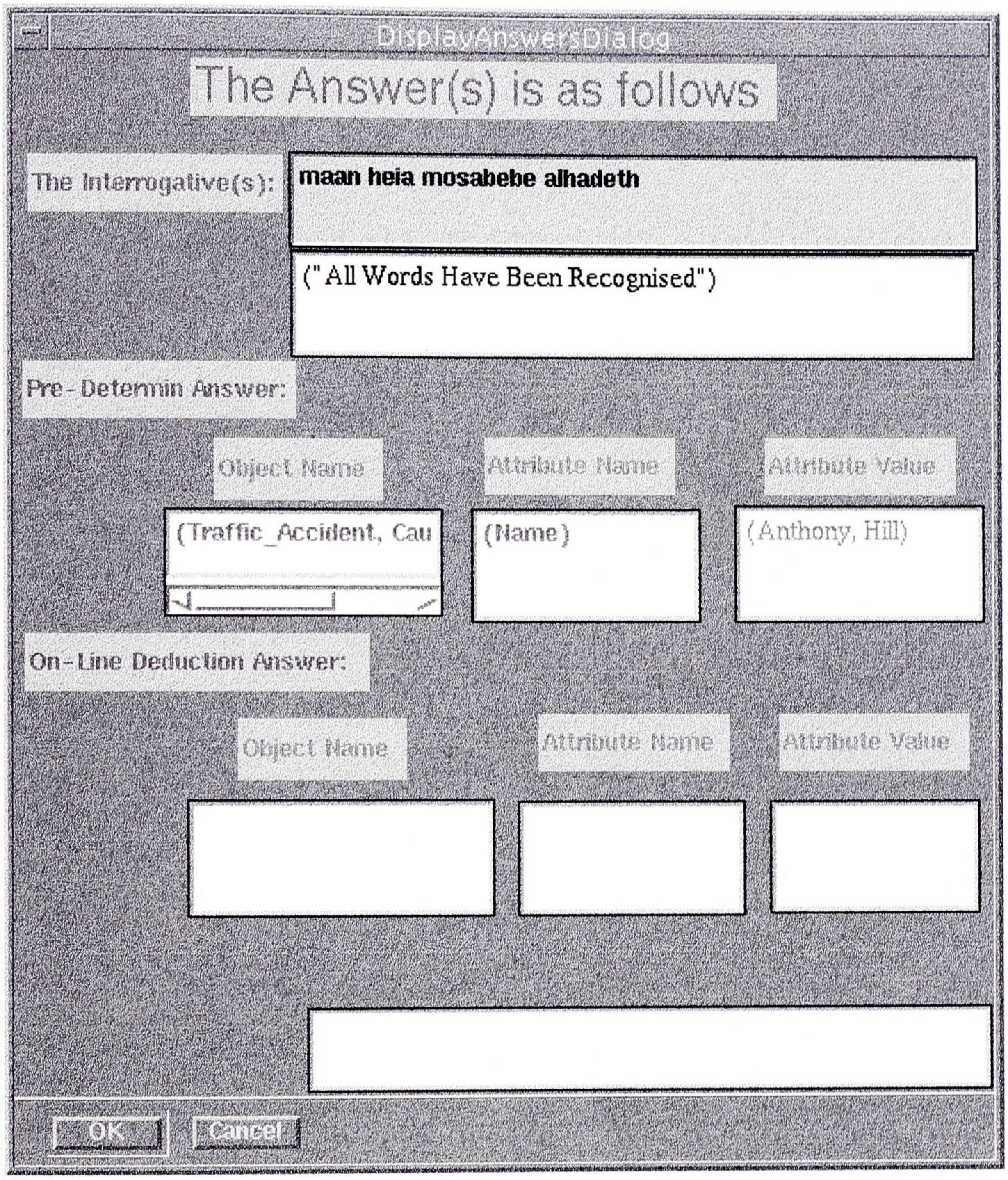
PRED REL (mosabebe)

ARG-1 ("The semantic ot ("The Declarative Presentation")

ARG-2

(maan)	("<<e,t>,sv>")	(Name)	(Human, Anima
(mosabebe)	("<<e,t>,sv><t,	(Cause Accidei	(Human, Anima
(alhadeth)	("<<e,t>,sv")	(Traffic Accic	(Human, Thing:
((heia), howa)	("<<e,t>,sv>")		(Human, Anima

4.3 The Answer Presentation:



The Answer(s) is as follows

The Interrogative(s): maan heia mosabebe alhadeth

("All Words Have Been Recognised")

Pre-Determin Answer:

Object Name	Attribute Name	Attribute Value
(Traffic_Accident, Cau	(Name)	(Anthony, Hill)

On-Line Deduction Answer:

Object Name	Attribute Name	Attribute Value

Appendix D

System Design Architecture - Linguistic Rules

Linguistic Rules

The following particles sets are a continuation from chapter seven. Each set contained its interrogative, its F-Structure rules, and its S-Structure rules.

The *When* (متى) Set

When did the accident happen? (متى وقع الحادث)

F-Structure Presentation

The above interrogative has been presented in the F-Structure Figure D.1. It shows the missing Obl-Arg i.e. the Thematic-object of the accident (مفعـول بـه - ظـرف زمـان). The interrogative F-Structure has been built by using the LFG sub-categorisation of the predicate *happen* (وقـع). The predicate is sub-categorised into Focus i.e. any interrogative tool, SUBJ for subject, and Obl-Arg for any missing argument such as object, and the rule for this interrogative is as follows:

If interrogative tool = when (متى)

followed by nominative past tense verb (فعل ماضي- مبني على الفتح)

followed by nominative Agent (فاعل مرفوع)

Then PRED sub-categorisation are:

PRED = ↑ SUBJ, ↑ OBL_ARG - Object (مفعول به) AND

Focus = when (متىَ :اسم استفهام مبني على الفتح في محل نصب - مفعول به ظرف زمان) **Where**

↑ **SUBJ is:**

{SUBJ = happen (''الحادث'' فاعل مرفوع بالضمة)}

↑ **OBL-ARG - Object is:**

{OBL-Arg - Object = object circumstantial of Date/Time (مفعول به - ظرف زمان).}

The rule has successfully created F-Structure presentation as Figure D.1 shows.

Focus: when

[اسم استفهام مبني على الفتح في محل نصب - مفعول بة ظرف زمان : متى]

PRED “وقع” < ↑ SUBJ, ↑ OBL_ARG (مفعول به) >

SUBJ

[فاعل مرفوع بالضمه ”الحادث“]

OBL_ARG

[مفعول به - ظرف زمان]

Figure D.1 F-Structure

S-Structure Presentation

The F-Structure presentation gives us the predicate, the subject, and a hint of the attribute i.e. Date/Time. We still have no knowledge of what is the Thematic-object is. By applying the semantic rules we have the following:

If Focus = when (متى) **AND**

the PRED is nominative past tense verb (فعل ماضي - مبني على الفتح) **AND**

the SUBJ is nominative Agent (فاعل مرفوع)

Then PRED REL (Relation) sub-categorisation is:

PRED REL = ↑ ARG-1 for argument-1, ↑ ARG-2 for argument-2 **Where**
ARG-1 is:
{ARG-1= when (متى : مفعول به ظرف زمان), and the predicate’s Thematic-object}
ARG-2 is:
{ARG-2 = the Slot Value of the predicate’s Thematic-object.}

Given the semantic presentation, we have created S-Structure as Figure D.2 shows. The S-Structure has, therefore, located exactly where about the answer can be found. Therefore, in our domain, the answer will be the Date/Time of the predicate’s Thematic-object *accident* as the S-Structure, and linguistic rules illustrated this point.

PRED REL

"وقع" : Intransitive, human, animal, things

ARG-1

"متى" : ظرف زمان

"الحادث" : Occurrence of accident

ARG-2

Slot Value of *accident* Thematic-object.
Answer = Date/Time: SV مفعول به - ظرف زمان

Figure D.2 S-Structure

The *How* (كَيف) Set

How did the accident happen? (كَيف وقع الحادث)

F-Structure Presentation

The interrogative has been presented in the F-Structure Figure D.3. It shows the missing Obl-Arg i.e. the object of the accident (مفعول به - حال). The interrogative F-Structure has been built by using the LFG sub-categorisation of the predicate *happen* (وقع). This predicate (PRED) can be sub-categorised into Focus, i.e., any interrogative tool, SUBJ for subject, and Obl-Arg for any missing argument such as object, and the rule for this interrogative is as follows:

If interrogative tool = how (كَيف)

followed by nominative past tense verb (فعل ماضي- مبني على الفتح)

followed by nominative Agent (فاعل مرفوع)

Then PRED sub-categorisation are:

PRED = ↑ SUBJ, ↑ OBL_ARG - Object (مفعول به) AND

Focus = how (متىَ :اسم استفهام مبني على الفتح في محل نصب - مفعول به حال) **Where**

↑ **SUBJ is:**

{SUBJ = happen ("الحادث" فاعل مرفوع بالضمة)}

↑ **OBL-ARG - Object is:**

{OBL-Arg - Object = object circumstantial of status (مفعول به - حال).}

The syntax rule has successfully created F-Structure presentation as Figure D.3 shows. This structure is concerning the syntactic analysis of the interrogative. But this analysis is not enough to obtain an answer, consider the following semantic analysis in order to obtain the answer.

Focus: how

اسم استفهام مبني على الفتح في محل نصب – مفعول بة حال : كيفَ

PRED "وقع" < ↑ SUBJ, ↑ OBL_ARG (مفعول به) >

SUBJ

accident "الحادث" فاعل مرفوع بالضمه

OBL_ARG

description مفعول به – حال

Figure D.3 F-Structure

S-Structure Presentation

The predicate of the F-Structure gives us, the subject, and a clue of what the attribute of the object should look like i.e. status. However, we still have no knowledge of what object name in the domain model is the status for. By applying the semantic rules we have the following:

If Focus = how (كيفَ) **AND**

the PRED is nominative past tense verb (فعل ماضي – مبني على الفتح) **AND**

the SUBJ is nominative Agent (فاعل مرفوع)

Then PRED REL (Relation) sub-categorisation is:

PRED REL = ↑ ARG-1 for argument-1, ↑ ARG-2 for argument-2 **Where**
ARG-1 is:
{ARG-1= how (كيفَ : مفعول به حال) and the predicate's Thematic-object}
ARG-2 is:
{ARG-2 = the Slot Value of the subject Thematic-object.}

With the semantic presentation, we have created S-Structure as Figure D.4 shows. The S-Structure has, therefore, located exactly where about the answer can be found. Therefore, in our domain, the answer will be the status attribute of the Thematic-object *accident* as the S-Structure, and linguistic rules illustrated this point.

PRED REL

“وقع” : Intransitive, human, animal, things

ARG-1

حال : “كيف”

“الحادث” : Occurrence of accident

ARG-2

Slot Value of the Thematic-object *accident*

Answer = Status: SV مفعول به - حال

Figure D.4 S-Structure

The Why (لماذا) Set

Why did the accident happen? (لماذا وقع الحادث)

F-Structure Presentation

The syntax of the above interrogative has been presented in the F-Structure Figure D.5. It shows the missing Obl-Arg i.e. the object of the accident (مفعول به - مبرر). The interrogative F-Structure has been built by using the LFG sub-categorisation of the predicate *happen* (وقع). This predicate (PRED) can be sub-categorised into Focus i.e. any interrogative tool, SUBJ for subject, and Obl-Arg for any missing argument such as object, and the rule for this interrogative is as follows:

If interrogative tool = why (لماذا)

followed by nominative past tense verb (فعل ماضي - مبني على الفتح)

followed by nominative Agent (فاعل مرفوع)

Then PRED sub-categorisation are:

PRED = ↑ SUBJ, ↑ OBL_ARG - Object (مفعول به) AND
Focus = why (لماذاَ :اسم استفهام مبني على الفتح في محل نصب - مفعول به - مبرر) **Where**
↑ **SUBJ is:**
{SUBJ = happen ("الحادث" فاعل مرفوع بالضمة)}
↑ **OBL-ARG - Object is:**
{OBL-Arg - Object = object circumstantial of status (مفعول به - مبرر).}

The syntax rule has successfully created F-Structure presentation as Figure D.5 shows. This structure is concerning the syntactic analysis of the interrogative. Consider the semantic analysis in order to obtain the answer.

Focus: why

[اسم استفهام مبني على الفتح في محل نصب - مفعول بة حال : لماذا]

PRED "وقع" < ↑ SUBJ, ↑ OBL_ARG (مفعول به) >

SUBJ

[فاعل مرفوع بالضمه "الحادث"]

OBL_ARG

[Reason مفعول به - مبرر]

Figure D.5 F-Structure

S-Structure Presentation

From the F-Structure we know the predicate, the subject, and an indication of what the attribute of the object i.e. reason. We still have no knowledge of the Thematic-object in the domain model. By applying the semantic rules we have the following:

If Focus = why (لماذا) **AND**

the PRED is nominative past tense verb (فعل ماضي - مبني على الفتح) **AND**

the SUBJ is nominative Agent (فاعل مرفوع)

Then PRED REL (Relation) sub-categorisation is:

PRED REL = ↑ ARG-1 for argument-1, ↑ ARG-2 for argument-2 **Where**

ARG-1 is:

{ARG-1= why (لماذا : مفعول به - مبرر) and the predicate's Thematic-object}

ARG-2 is:

{ARG-2 = the Slot Value of the predicate's Thematic-object.}

The semantic rule have created S-Structure as Figure D.6 shows. The S-Structure has, therefore, detected where about the answer can be found. Therefore, in our domain, the answer will be the reason attribute of the object name *accident* as the S-Structure, and linguistic rules illustrated this point.

PRED REL

"وقع" : Intransitive, human, animal, things

ARG-1

"لماذا" : مبرر

"الحادث" : Occurrence of accident

ARG-2

Slot Value of *accident* object name. Therefore,
Answer = Reason: SV مفعول به - مبرر

Figure D.6 S-Structure

The *Did, Is, Have, Has, Can* (هل) Set

Other set of particles we would like to demonstrate is the particle *Did*. Consider the following interrogatives:

1. Did an accident happen? (هل وقع الحادث)
2. Did the car driver killed? (هل قتل سائق السيارة)

F-Structure Presentation

The above interrogatives, querying the occurrence of their subject/object. For instance, interrogative two looking for the subject, and if the subject found we would like to query the states of that subject. The F-Structure in Figure D.7 shows the presentation of the interrogative in two. The F-Structure has been built by using the LFG sub-categorisation of the predicate killed (قتـل). This predicate (PRED) can be sub-categorised into Focus i.e. the interrogative tool *Did*, the OBJ for object, and Obl_Arg for any missing argument in this case the missing subject, and the rule for this interrogative is as follows:

If interrogative tool = Did (هل)

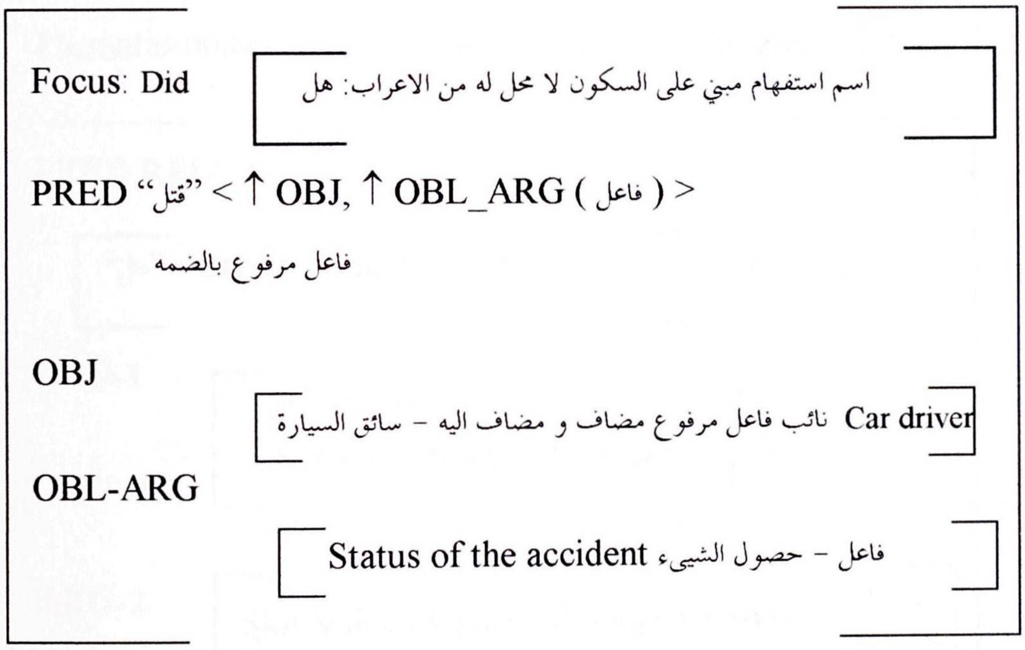
followed by nominative past tense verb (فعل ماضي– مبني على الفتح)

followed by nominative Pro Agent Annexing (نائب فاعل مرفوع – مضاف)

followed by accusative Annexed (مضاف أليه مجرور)

Then PRED sub-categorisation are:

PRED = ↑ OBJ, ↑ OBL_ARG - SUBJ (فاعل) AND
Focus = Did (اسم استفهام مبني على السكون لا محل له من إعراب : هل) **Where**
↑ **OBJ is:**
{OBJ = car driver (فعول به – مضاف و مضاف أليه سائق السيارة)}
↑ **OBL-ARG - SUBJ is:**
{OBL-Arg - SUBJ = Status of the accident (محل فاعل – حصول الشيء).}



Focus: Did اسم استفهام مبني على السكون لا محل له من الاعراب: هل

PRED "قتل" < ↑ OBJ, ↑ OBL_ARG (فاعل) >

فاعل مرفوع بالضمه

OBJ

Car driver نائب فاعل مرفوع مضاف و مضاف اليه – سائق السيارة

OBL-ARG

Status of the accident فاعل – حصول الشيىء

Figure D.7 F-Structure

The above syntax rule has successfully created F-Structure presentation in Figure D.7. This structure is concerning the syntactic analysis of the interrogative. Consider the following semantic analysis.

S-Structure Presentation

The predicate, the object, and a reference of what the attribute of the subject i.e. the status of the driver. What about the Thematic-object, consider the following semantic rules:

If Focus = Did (هل) **AND**

PRED = about status of killing (حالة موت) **AND**

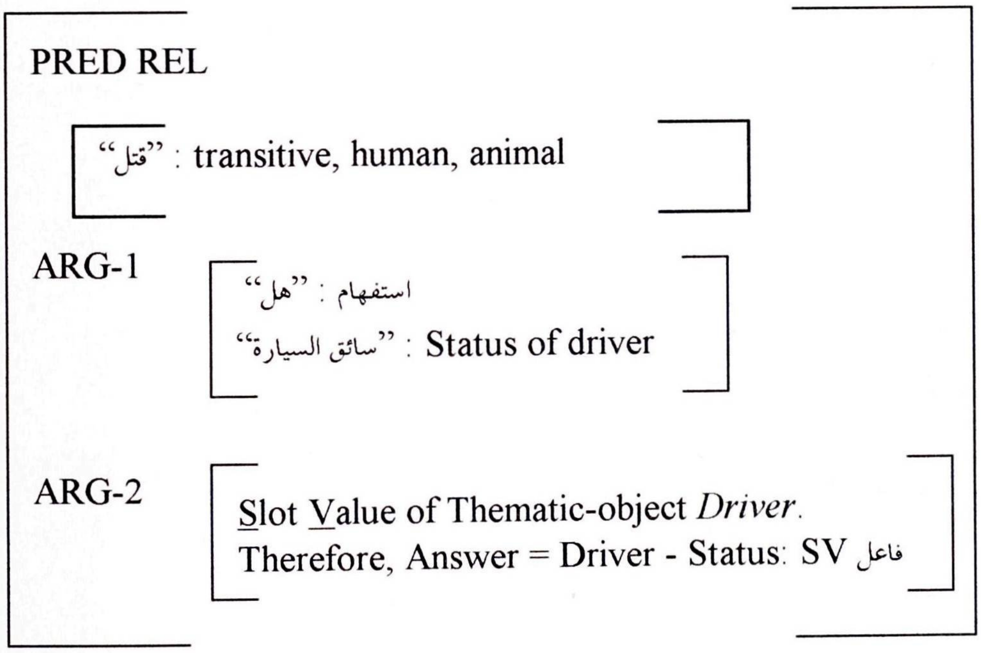
OBJ = noun in the case Annexing/ Annexed (مفعول به – مضاف و مضاف إليه)

Then PRED REL (Relation) sub-categorisation is:

PRED REL = ↑ ARG-1 for argument-1, ↑ ARG-2 for argument-2 **Where**
ARG-1 is:
{ARG-1= did (اسم استفهام: هل), and the predicate's Thematic-object}

ARG-2 is:
{ARG-2 = the Slot Value of the predicate's Thematic-object.}

Given the semantic presentation, we have created S-Structure as Figure D.8 shows. The S-Structure has, therefore, found precisely where about the answer can be. Therefore, the answer will be *the status of the driver killed/not killed* (حالة موت / غير موت السائق) of the Thematic-object *Driver* as the S-Structure, and linguistic rules illustrated this point.



PRED REL

"قتل" : transitive, human, animal

ARG-1

استفهام : "هل"
"سائق السيارة" : Status of driver

ARG-2

Slot Value of Thematic-object *Driver*.
Therefore, Answer = Driver - Status: SV فاعل

Figure D.8 S-Structure

Appendix E

Log-in Guide

Log-in Guide

Running the Prototype

Prototype users and prototype developers are advised to consult all Kappa manuals. These are available at the computer lab. Kappa and Object Management Workbench (OMW) package is running on UNIX Sun/Solaris platform. The name of the Sun Station is (amigo). In order to login into any Sun/Solaris machine, the user must have a login name that can access the machine (amigo) if the user is running Kappa from a different machine. This can be run from any machine that can run Common Desktop Environment (CDE) Software. Once this has been set-up, the user must follow the following commands.

1. From any UNIX Solaris station run CDE software. If the user is running Kappa from (amigo) machine, commands 2 to 5 below should be ignored.

2. From the window manager of CDE, open CDE window (terminal), and type the following command (xhost + amigo). This command identifies the machine (amigo) as a server that contained Kappa/OMW package.

3. Type (rlogin amigo).

4. Enter a user password. Password can be obtained from the system administrator.

5. Type (setenv DISPLAY 'the name of the machine e.g., (bond machine) ', followed by ':0.0' with no space between.

6. Type 'omw –kappa' in any window (terminal).

7. At this stage the omw and kappa package runs and it may take up to five minutes depending on the speed of the network.

8. From the omw tools select 'load domain' option, after that add the directory path which is (traffic/ourtraffic).

9. From the domain menu select 'traffic_Feb' domain.

10. The domain should take two to three minutes to load. Then the system is ready to be used. Follow instructions from the screen.