

## **ABSTRACT**

The aim of this study is to construct a natural language interface generator. This generator system generates natural language interfaces for any command-based computer programs (such as data base management systems, operating systems, etc.). The system offers a range of capabilities such as grammar editor and file server. This range of capabilities allows layman interface builder to rapidly define workable subsets of English and gives expert builders the tools needed to produce powerful and more efficient interfaces. The system provides many facilities with generated interfaces such as, **spelling corrector**, **ellipsis handler**, **previous commands editor** and **paraphrase handler** to extend the language recognized by a generated interface. This range of capabilities allows beginning interface builder to rapidly define workable subsets of English and gives expert builders the tools needed to produce powerful and more efficient interfaces.

The design of the system forced us to develop the semantic grammar technique to be able to describe the following:

- (a) A subset of natural language required to communicate with software,( which an interface is needed for ).
- (b) The commands of a system software.
- (c) The meaning of that subset of natural language in a system software world.

This developed notation of semantic grammar has been called **Modified Semantic Grammar**. To generate an interface, the builder must provide a grammar, written in modified semantic grammar notation, as input to the generator, then the system produces an interface as output coded in Turbo PROLOG language. The system runs at any IBM PC computer or compatible.