STUDIES IN THE EVALUATION OF A DOMAIN-INDEPENDENT NATURAL LANGUAGE QUERY SYSTEM

Matthias Jarke
Jürgen Krause (*)
Yannis Vassiliou

April 1984

Center for Research on Information Systems
Computer Applications and Information Systems Area
Graduate School of Business Administration
New York University

Working Paper Series

CRIS #72

GBA #84-47(CR)

(*) Linguistische Informationswissenschaft, Universitat Regensburg, Universitatsstr. 31, 8400 Regensburg, West Germany.

To Appear in L. Bolc (ed.), Cooperative Interactive Information Systems, Springer-Verlag, 1984.

STUDIES IN THE EVALUATION OF A DOMAIN-INDEPENDENT NATURAL LANGUAGE QUERY SYSTEM

Abstract. There is growing consensus that some of the most crucial questions concerning the feasibility and desirability of natural language interfaces to databases can only be resolved by empirical research. This paper reports the results of several empirical studies which investigated the same domain-independent natural language query system, using various applications in two different natural languages -- English and German. Taken together, these experiments involved about 100 subjects and over 12,000 queries, constituting the bulk of empirical evaluations of natural query language systems reported to date. Some definitive results are derived from the combined experience, and plans are outlined to resolve several of the remaining issues.

1.0 INTRODUCTION

A large number of natural language interfaces (NLI) to information systems have been developed. The continued research interest is evidenced, for example, by no less than 45 papers in [Bundy 1983]. In industry, early skepticism seems to have given way to last-minute panic: there is hardly a major computer company or software house that is not working on product development.

On the other hand, many practical questions remain unresolved. Fierce battles are still being fought over the best overall architecture for implementing NLI, or -- more basically -- whether NLI are preferable at all to formal query languages designed with human factors in mind. The problem of how the alternative hypotheses can be verified or at least be made more plausible remains open. Besides this problem of evaluation methodology, three central questions concerning NLI themselves are still awaiting an answer.

(1) Can NLI be implemented at all? It seems clear that a full natural language system corresponding to interhuman communication is presently infeasible; any practice-oriented NLI must be application-specific. On the other hand, a NLI would be unacceptable if each user required support by language engineers for an excessive period of time, if the subset of natural language that can be implemented efficiently were not sufficient to support a practical application, or if users had insurmountable difficulties recognizing the boundaries of the implemented subset.

---

(2) If NLI can be implemented, do they support human problem solving more successfully than competing end user interfaces, such as formal query languages? A meaningful answer to this question requires measurements beyond the percentage of submitted queries that is accepted by a system.

(3) How difficult is it to transport a NLI to a new application? This question is important since it may not be economically feasible to develop a completely new NLI for each new application -- and maybe for each user of each application!

These questions must be further refined by user type and application area as well as by type of NLI. This paper focuses on NLI for database querying (NLQS) [1]. Within this group, two essentially different approaches can be distinguished: domain-specific NLQS in which a large portion of the system has to be redeveloped for each new application, and domain-independent systems in which most of the system is portable between applications and the parts to be changed are clearly isolated and relatively small.

Shwartz [1984] contrasts a knowledge-based domain-specific NLI called EXPLORER with domain-independent restricted subset systems, which draw on general language knowledge, application-specific vocabularies, and the database itself. He concludes:

> "Natural language systems lacking a knowledge base cannot understand anywhere near as wide a range of information retrieval requests as can knowledge-based systems." (p. 247)

The subset type of NLI, rejected by Shwartz and others [Malkovsky 1982; Morik 1982], is the focus of this paper. One reason is that only subset systems have reached a degree of maturity where they can be subjected to rigorous empirical testing. There is no indication that this will change in the near future. Indeed, the only commercially successful NLI so far, Intellect [Artificial Intelligence Corporation 1982], is of the subset type. Unfortunately, no formal performance studies of Intellect have been reported, although some global figures for its predecessor, ROBOT, appear in [Harris 1977].

The paper examines the three questions raised above in the context of a particular restricted subset NLQS, which represents this type of natural language system in a rather pure form. There seems to be no NLQS or other NLI that has been subjected to a comparable number of empirical studies. The first objective of this paper is to present — in a common framework — the experience gained from multiple evaluation methods applied to the same system. A second objective is to contribute to a better understanding of the overall feasibility and desirability of the domain-independent approach to NLI, based on the empirical assessment of one specific system.

In section 2, the NLQS under study is briefly described and a global framework for NLI evaluation methods is given. Sections 3 through 5 describe the design and results of several empirical studies of the NLQS. Section 6 presents a synopsis of the results concerning experimental methodology and NLI performance. Discrepancies and open questions requiring additional research are highlighted. Section 7 briefly summarizes the general conclusions.

[1] For a survey of other natural language applications, see [Waltz 1983].

2.Ø RESEARCH OVERVIEW

2.1 Natural Language Query System

The NLQS whose evaluation is reported here [Lehmann 1978; Ott 1979b; Ott and Zoeppritz 1979] provides a natural language interface (English, German, Spanish [Zoeppritz 1983b]) to relational databases. The system does not engage the user in clarification dialog, and to that extent the system is similar to any formal database query language. Structurally, the NLQS consists of a generalized parser, a semantic analyzer and executor, and a generalized DBMS (figure 2-1). In the sequel, system structure and main objectives of the system will be briefly reviewed.

System Structure. The parser [Bertrand et al. 1981], accepts general phrase structure grammars written in a modified Backus-Naur form. All parses are produced in parallel, bottom-up, and from right to left. Arbitrary routines can be invoked with any rule. The vocabulary is presented to the parser as part of the grammar. The semantic analyzer and executor [Lehmann 1978] consists of a set of interpretation routines which translate the syntactic structures to DBMS executable code. The formal query language SQL serves as the target database query language, supported by the third component - a relational DBMS [Astrahan et al 1976]. View definitions relate the vocabulary to the database fields. In another version -- which was used for the early empirical studies -- the experimental relational database system PRTV [Todd 1976] was used as the target DBMS and the target language for translation was relational algebra (ISBL).

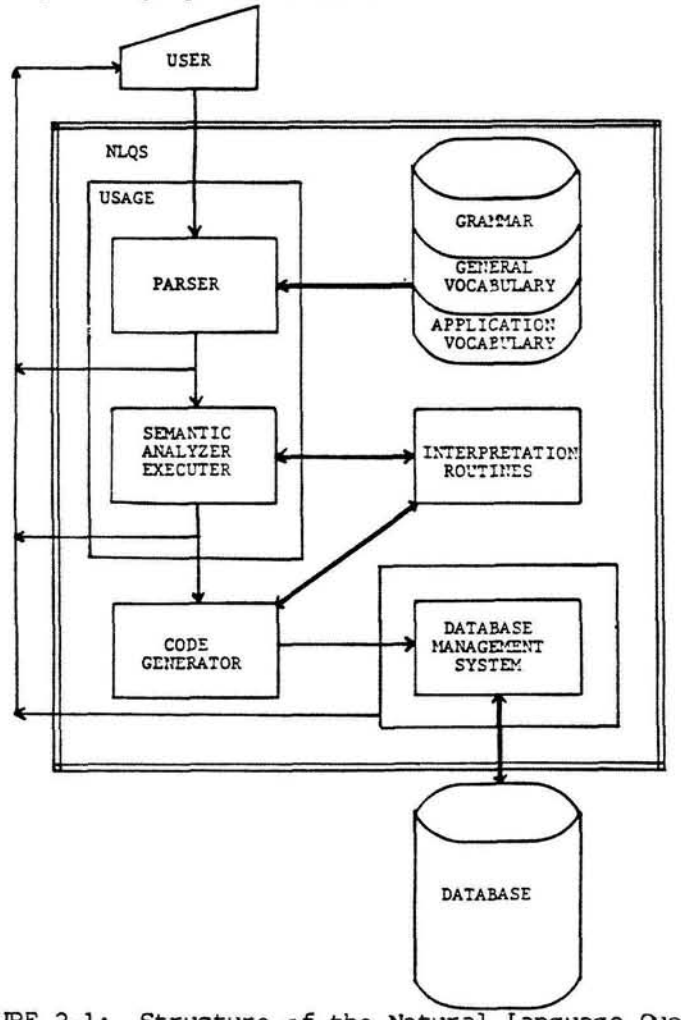

FIGURE 2-1: Structure of the Natural Language Query System

Transportability. The system emphasizes transportability across application domains, in the same way formal database query languages do. Moreover, it even achieves transportability across natural languages (from German to English, Spanish, and Dutch).

The goal of transportability has three major consequences that influence the design of the NLQS. First, only an application-independent kernel vocabulary is provided. It is the responsibility of the application developer(s) to build a special application vocabulary. Second, the linguistic component (information about language) is completely separated from the database component (information about domain and data retrieval). Finally, the system has few application-dependent deductive capabilities (the only exceptions being view definitions and the mathematical functions provided by the target language) which draw inferences from stored data and try to understand user intentions. Thus, the system provides limited feedback and seldom assumes control of the interaction.

Application-Specialist Computer-Novice Users. The system requires users to know their application well and to be able to compose questions in their native language. The intended users are neither EDP professionals (having, e.g., database skills), nor linguists. Users are also expected to define their own application-specific vocabulary. Consequently, the system is designed so that the generation of the application vocabulary should neither be a complex process nor require special database and linguistic expertise. This implies certain restrictions in the amount of application-specific linguistic information that can be provided to the system (e.g., no word semantic).

Syntax-Based System. To compensate for the lack of application-specific knowledge bases (which exist only in the form of SQL view defintions), the NLQS goes great lengths in exploiting the systematic connections between syntax and semantics of natural language. Syntactic structures carry meaning which is independent of the application domain. Consequently, the system's grammar is designed with emphasis on this kind of meaning.

2.2 Basic Evaluation Methodologies

The simplest and most widely used approach for the evaluation of NLI is the exchange of intuitive arguments about implementation techniques and language features. For example, the information about natural language systems found in the literature is typically highlighted with a list of supported features (e.g., coordination or ellipsis, see [Codd et al. 1978; Morik 1982]).

Such a list is only useful for the features not included. It can be very misleading since it rarely addresses the important question: "to what degree is the feature supported?" Therefore, it becomes almost impossible to effectively evaluate the usability of any system based on the information given by the system description. Furthermore, it has been shown [Lehmann and Blaser 1979; Krause 1980, 1982; Stohr et al. 1982] that opposing arguments of comparable plausibility are confronted without much prospect for a purely argumentative synthesis. There is growing consensus [Petrick 1976; Finnin et al. 1979] that only empirical evaluation research can lead out of this dilemma.

Answering the three questions, set forth in the introduction with respect to the domain-independent type of NLI, requires a carefully designed methodology for generating and verifying research questions. In this subsection, some of the basic design parameters for empirical investigations of NLQS will be analyzed. The leftmost two columns of table 2-1 provide an overview of such parameters (compare also [Krause 1982; Jarke 1983; Turner et al. 1984]).

DECISION VARIABLE	DESIGN ALTERNATIVES	STAGE A	STAGE B	STAGE C
evaluation team	designers	x		
	outside researchers		x	x
evaluation strategy	absolute	x	x field	
	comparative		(x) lab	x
evaluation criteria	quantitative:			
	success	x	x	x
	effort		(x)	x
	qualitative:			
	problems	(x)	x	(x)
	strategies		x	
	level:			
	work task			x
	query	x	x	x
evaluation object	simulated NLI			(x)
	real NLI	x	x	x
type of study	laboratory experiment		(x)	x
	field study	x	x	x
subject selection	students		x lab	
	paid subjects			x
	end users, novices	x	x KFG	
	end users, experts	x	x TA	
database and application	structure:			
	simple	x		
	medium	x	x	x
	complex			
	size:			
	small	x	x	x lab
	large			x field

TABLE 2-1: Design Parameters for Empirical NLQS Evaluation Studies and Characterization of the Studies Reported in this Paper

Evaluation Team. The first step in evaluating a natural language system empirically is an on-site test of the parser, often termed as an acceptance test. One or more 'toy' databases are created, and a series of queries are run against these databases by the system designers. Such studies attempt to test supposedly typical, as well as pathological queries. After an iterative process (each iteration corresponding to an improvement of the grammar and the interpretation routines) the system may reach a steady 'acceptable' state.

There is certainly a need for performing this kind of evaluation but there is also the danger of deriving optimistic conclusions about the usability of the system, after attaining a steady state, or of abandoning useful research efforts if a steady state is not reached. (This happened, for example, to the German natural language systems PLIDIS [Kolvenbach et al. 1979] and CONDOR [Fischer 1982], see [Krause 1983a].) The system designers cannot be termed 'objective' evaluators. Their invested interest and their detailed knowledge of system capabilities blur their ability to distinguish the needs of future actual users from what the designers want these users to do. In addition, the test applications are commonly several orders of magnitude smaller and conceptually less complex than 'real' applications. Most natural language systems have terminated the evaluation after this acceptance test (unless local test evaluation proved to be a non-ending process.)

Better control is provided by formal evaluation studies conducted by researchers outside the design team. Such an empirical evaluation can be seen as part of a cost-benefit analysis required before introducing a query language into an actual user environment [Jarke and Vassiliou 1982]. Several design deciions are of critical importance in this process.

Evaluation Strategy. The first issue is whether the NLI should be evaluated in the absolute or compared to a competing interface, such as a formal query language. Some useful analyses (e.g., of user problem solving strategies) can be performed in the first case. However, performance evaluations using this strategy are meaningful only if the system under study is either close to perfect, or the results are so disastrous that any alternative would be preferable. Otherwise, a comparative study is necessary.

Evaluation Criteria. This discussion leads to the second design question: how can one measure the costs and benefits of a natural language user interface? Of interest are: the success rate of users working with the system, the effort to achieve such success (or failure), the language and system related problems, the strategies users develop to work around the limitations, and finally the subjective perceptions and opinions of the users. Additional criteria may be required to control for confounding outside factors.

Orthogonal to these criteria are the amount of skills the user has acquired [Schneider 1984], and the level on which performance is evaluated. The former refers to the differentiation between learning and routine task performance [Moran 1981], which is closely related to the definition of user types [Jarke and Vassiliou 1982]. The latter addresses the distinction between the solution of a problem or work task, for which the database is a tool among others employed by the user, and the generation of an answer to a specific database query.

Evaluation Object. The organizational setting of the study must be decided. Some studies assume a simulated rather than a real NLI (e.g., [Chapanis 1973; Small and Weldon 1977; Shneiderman 1980; Miller 1981]). Studies of this type can give valuable hints concerning the desirability of NLI but are usually unsuited for establishing their feasibility.

Type of Study. A more important distinction is between laboratory experiments and field studies of real systems. Laboratory experiments allow or a controlled setting. Methodologies to run them have been extensively studied, and the experiments are economically affordable. Such studies, if performed correctly, are best suited for examining the short-term 'learnability' of a language, identifying language constructs likely to cause user difficulties, and for estimating the number and type of words used for a particular set of tasks, as well as the language features most likely to be employed.

On the other hand, drawing practical conclusions about the overall usability of a natural language system from laboratory experiments may be dangerous [Reisner 1981]. For example, it is not clear whether field performance will be superior or inferior to laboratory performance, or which factors influence the difference. Surveys of laboratory studies of query languages are given in [Reisner 1981; Jarke and Vassiliou 1982]. The most frequently studied language is SQL which has shown a consistent performance of 55 - 70% correct queries in paper and pencil tests after a few hours of training. Laboratory studies of NLQS, most not employing real systems, are surveyed in [Lehmann and Blaser 1979; Krause 1982; Vassiliou et al. 1983a].

Despite the critical remarks by Petrick [1976] and Tennant [1979], the lack of field studies has hardly changed. Aside from the studies described in this paper, the main exception is a year-long field study of TQA, yielding about 700 queries with an acceptance quote of approximately 65% [Damerau 1979]. However, the setting did not allow for the implementation of detailed controls, nor was this intended. Some even more informal studies [Woods 1977a; Harris 1977] report only about 20% language-related errors but disregard certain other kinds of failure of the man-machine communication. In general, field studies should be suitable for the evaluation of actual task performance over an extended time period if close observation or carefully designed controls permit the elimination of outside confounding factors. A research design which couples field studies with laboratory experiments, in a way that combines the strengths and reduces the weaknesses of both methods, seems most promising.

Subject Selection. The type and intrinsic motivation of users often has a strong impact on the results of laboratory and field studies. The preferred type of users, actual end users, can be quite demanding and may actually abandon system usage if an alternative way to solve their problems is available. On the other hand, student subjects may be less motivated to achieve good performance. The intermediate solution, using paid subjects, may yield good results if their compensation is related to their success with the system or a good motivation can be achieved in a different way.

Database and Application. Last but not least, the size and complexity of both the application domain and the underlying database may influence the outcome of the experiments, by response time effects [Barber and Lucas 1983] as well as by the impact of complexity on the user's ability to fully understand the application.

2.3 Overview Of Evaluation Studies

Experiments with the NLQS have been conducted by different research groups (IBM Scientific Center Heidelberg, University of Regensburg, New York University), using two different natural languages (German and English) and various experimental designs. Three stages of experimentation can be distinguished.

In the first phase (stage A), the development team tested the system informally to uncover errors and gaps in coverage. The second set of experiments (stage B, the KFG study at Heidelberg and at the University of Regensburg since 1978) was still performed in part at the development site and with technical support by the development team but by an external researcher. At the heart of these experiments was a long term (16 months) observation of a single user working on a practical application. Detailed qualitative analyses were performed, and the original field study was complemented by another field study and several minor laboratory experiments.

For the third series of experiments (stage C, the Advanced Language Project (ALP) at New York University from 1981-1983), the system was transfered to a different natural language (English), and to a site where little linguistic or technical support by the development team was available. A quantitatively oriented evaluation strategy was chosen for comparing the NLI to a formal database query language in a partially controlled field study and two controlled laboratory experiments.

The rightmost columns of Table 2-1 characterize each of the three stages by the design parameters presented in the previous section. The following sections provide more detailed information about each stage.

3.0 STAGE A: SMALLER APPLICATION STUDIES

Since 1976, the development team tested the system in a series of small evaluation studies on real applications. However, with the exception of one application (SCHOOL), no actual field usage was reached since high error rates required continuous drastic changes of the prototype. The same problem prevented the success of an attempted comparison of the NLQS with another natural language interface [Kettler et al. 1981].

An overview of this first stage of system evaluation is given in [Lehmann et al. 1978] where 451 questions of these tests were analyzed. Krause and Lehmann [1980], and Zoeppritz [1983a] describe the application areas.

1. PLANNING (1976). Data on customers for planning purposes. Two users submitted 59 queries at an error rate of 46%.

2. SCHOOL (1977). Data on school attendance and background of pupils. One user submitted 356 queries at an error rate of about 13%.

3. RECEPTION (1977). The database contained information about departments. Receptionists used the NLQS to help visitors find appropriate people to answer questions. One user submitted 115 queries at an error rate of 47%.

4. ROOMS (1978). Allocation of rooms and office space. Three users submitted 781 queries at an error rate of about 40%.

The tests of stage A can be regarded as debugging tests, attempts to detect functional and grammatical gaps, and trials to obtain hints with respect to the size of the necessary subset and the transportability of the system. All tests of stage A used the German version; no comparison with formal query languages was attempted.

4.0 STAGE B: KFG STUDY AND RELATED TESTS

4.1 Project History And Study Description

The evaluation studies of stage B can be seen as parts of an extended evaluation scheme, outlined in Figure 4-1. The plan starts with a real application to be analyzed in a field study. Laboratory experiments are based on a typical session of this real application. Typical means, among other things, that the session contains a representative mixture of dialog types, the linguistic structures of the overall study are represented, and the error rate is near the average. The three formal query languages mentioned in Figure 4-1 could be changed. It is only important to select types of query languages which have been proposed (or used in practice) for efficient man-machine-interaction.

ISBL, as an example for algebraic query languages, and SQL were advantageous for comparison with the NLQS because the parser translated the natural language queries initially into ISBL, and since 1981 into SQL. The proposed user groups in Figure 4-1 are extreme points of a broad spectrum of possibilities. One may hypothesize that using different groups is best suited to uncover the common trends.

The field studies and laboratory experiments of stage B consisted of three subgroups:

1. A field study with teachers of the Karl-Friedrich-Gymnasium (KFG) at Mannheim in West Germany (the KFG field study).

2. An effort to transport the same system version to another application (the TA field study).

3. Several laboratory tests to compare error rates in the KFG field study with those achieved by using formal query languages.

4.1.1 KFG Field Study. - The KFG field study was carried out by three teachers, supported by the system development team. The teachers wanted to analyze data on student development. For instance, they wanted to know whether low grades in mathematics in earlier years have predictive power for grades at graduation. Typical questions were:

"Wieviele Schüler gehen in Untertertia?"
(How many students attend class 8?)

"Liste die Schüler, die nicht versetzt in Sexta sind."
(List students who are not promoted in class 5.)

The database contained 41,250 grades for 430 students and further information about social background and class repetition. Between August 1978 and September 1979, 7278 questions were asked in 46 sessions. The users worked 157 hours and 26 minutes with the system. Unfortunately, 6603 questions were submitted by a single teacher. Therefore, the KFG field study is in its substance a one-user study, extended by a smaller set of 675 questions by two more users.

4.1.2 TA Study. - A preliminary evaluation of the KFG field study [Krause 1980] showed that there was a real-world application, which could be queried in a natural language subset with an overall error rate of only 7%. Therefore, a second field study was prepared whose aim was to test whether the successful KFG version could easily be transported to a new application.

The application area of the study was a technical service department of IBM (Technischer Aussendienst = TA). The database included information about EDP systems for which the TA was responsible. The main information groups were: maintenance hours per month and EDP system, details of the systems, the customers, and the organization responsible for the customer. For example, a typical question was:

"Liste die Teams des Wartungsgebietes 424."
(List the teams of maintenance area 424).

The first user was an employee who had worked with the database for five years, using the formal query language IQRP [IBM 1976]. Initially, it was planned to bring in other users who had no knowledge of formal query languages. Since it was not possible to build up a version of the TA application with an error rate tolerated by the user (the error rate was about 53% [Krause 1982, chapter 5]), however, the only data for the TA study consisted of queries by the experienced user, submitted in pretests between April 1979 and September 1979.

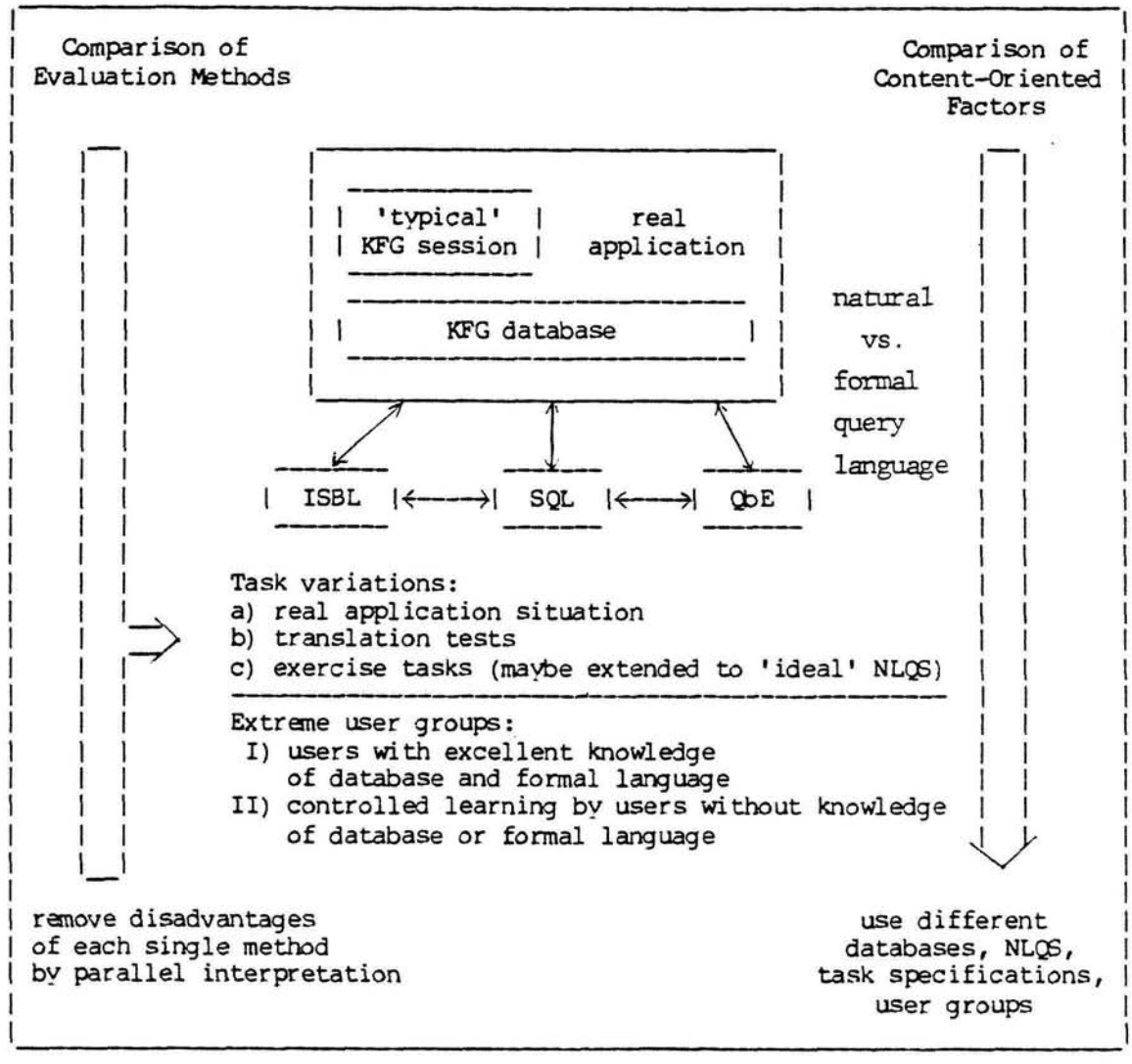

FIGURE 4-1: Evaluation Plan - KFG Studies

4.1.3 Laboratory Experiments. - The KFG field study was supplemented by laboratory experiments intended to determine whether users could have achieved comparable success with a formal query language, and to measure the time required to learn a formal query language.

4.2 Methodology

4.2.1 KFG Field Study: Research Design And Evaluation Parameters. - On the background of the overall evaluation plan, the stage B evaluation reached only some preliminary results. Nevertheless, in particular the KFG field study offers a large amount of experimental data. The analysis used the following primary data:

Computer protocols of the terminal sessions. Each user query, and the responses of the NLQS were stored automatically for later interpretation.

Observations of the users during the terminal sessions. The observer sat next to the user during the sessions. He introduced himself as a researcher with only superficial knowledge of the system, interested in knowing how the user worked with it and what improvements he might desire. The user was not led to expect the observer to help him with any difficulty [Krause 1980].

Questionnaires [Ott 1979a] and general statements of the users.

Comments of the users on the results obtained (worked out by the user, when reviewing the results of a session at home). For instance, the user reported which aspects of his problem had been solved.

Results the users achieved (in the case of the KFG field study a research paper written by a user [Schuetz 1979]).

One of the main difficulties in analyzing primary data is the detection, statistical description and detailed study of those phenomena which offer promise for plausible interpretations of general patterns. In the KFG field study, error information proved to be most instructive. Each situation in which the man-machine dialogue failed was defined as an error. Absolute and relative error rates were measured, extensive error classification was performed, and the distribution of the errors over sessions or with respect to different dialogue types was determined. Additionally, the error handling strategies of the user were analyzed, as the basis of an examination of all error chains (the starting error and all following errors).

4.2.2 Laboratory Experiments: Research Design And Evaluation Methodology - To compare user performance in the NLQS against a formal query language, five user groups were asked to translate 81 questions from a typical session of the KFG field study into the formal language, ISBL. Group 1 consisted of two users with several years of experience with ISBL who knew the KFG database well and therefore required no training. Groups 2, 3, and 4 consisted of students (altogether 20), who learned ISBL in a controlled procedure before the test. It has been argued that tests with students do not not yield results that are representative for real usage; typical user groups might have more difficulties in learning a formal language. Therefore, six secretaries (group 5) with experience in word processing were trained and tested in the same way as the student groups [Krause 1983b; Krause et al. 1983].

All groups were built up at random. The members of the three student groups attended courses in the department of Linguistic Information Science at the University of Regensburg, and the secretaries were employees of projects in the same department. Thus, the laboratory experiments of the KFG study have the status of pretests, rather than of statistically adequate experiments.

4.3 Experimental Results From Stage B

The presentation will be limited to a brief summary of more general results, and will omit those data which can be explained by the prototype state of the system (see [Krause 1982]). The experiments provided insights regarding the form of a natural language subset, as well as regarding the correctness of several arguments for and against natural language as a query language for databases.

4.3.1 Results With Respect To Some Individual Problems. - The literature contains various arguments concerning response time, conciseness of query formulation, change of query patterns over time, and learning time of natural language systems as compared to formal query languages. The stage B studies provided some partial answers to these questions.

Response time. It has been argued (see, for instance, Ghosh [1977]) that, in practice, natural language interfaces require an unacceptable amount of computer time for parsing the natural language query. The KFG field study typically showed additional CPU requirements of about 1.5 CPU-seconds for natural language queries over formal language queries (on an IBM 370/145). This means that it is possible to develop natural language translators which work acceptably fast. (This statement does not necessarily cover potential problems resulting from a very large database in combination with a possibly inefficient translation of natural language requests, see section 5.)

Conciseness. Woods [1977b] suggested as an advantage of formal language queries that formal expressions are more concise than natural language queries. The translation of typical KFG and TA questions into the formal query languages ISBL, IQRP and SQL showed that this conjecture does not hold in reality. Particularly in SQL, one has to expect longer input strings than with NLQS.

Changing requirements. Malhotra [1975], Woods [1977a], and Harris [1977] expected on the grounds of observations in shorter evaluation studies that the queries in long term studies could become more and more complicated. The KFG field study did not confirm this fear.

Learning time. One of the main arguments against the use of formal query languages is that inexperienced users need too much learning time. This hypothesis in favor of natural language could not be verified. Three groups of students could translate the questions of a typical session of the KFG field study into ISBL (in part with fewer errors than the user in the field study) after a maximum learning period of six hours and forty minutes (6-11% errors). These results indicate that the users of the KFG field study might have been perfectly capable of acquiring ISBL skills rather fast. The test with the six secretaries confirmed this result [Krause 1983b]. After 11 hours of training, they were able to use a formal query language with reasonable success. On the other hand, it seemed that learning a formal language can be a major psychological burden, especially for older users. Because of the nature of the selected tests, the effects of two factors cannot be predicted:

1. The possibility of forgetting the rules of the formal query language after a period of time was not taken into consideration.

2. The concentration of the user might be diverted from the problem to be solved by constructing the formal ISBL expression. This could lead to poorer problem solving than in the application of natural language.

4.3.2 Results With Respect To The Subset Definition. - There were mainly three results concerning the problem of how to define the subset of a practice-oriented NLQS.

The KFG and TA evaluation study confirmed that restricting the application area leads to a considerable simplification of language analysis. The vocabulary for all of the system's applications to date falls in the range between 100 and 300 words. These observations agree with those made in other practice oriented natural language interfaces (see for instance [Hendrix 1978], [Waltz 1978]).

Surface structural variations (for example: "Schueler, die nicht versetzt sind", students who are not promoted instead of "nicht versetzte Schueler", non-promoted students) are used extensively, even though there is no difference at all, as far as the expected answer is concerned. This result is in contrast with the assumption that users will not change successful input patterns and will generally prefer shorter formulations to reduce input time.

As a corollary, individual error categories cause interruptions of varying strength and nature in the man-machine interaction. Errors caused by surface phenomena (for example, word order) lead to serious difficulties [Krause 1982]. The user is often unable to develop effective error strategies, or to learn the restrictions of the language system in order to avoid 'dangerous' constructs. For example, the NLQS recognized only "Schueler, die nicht versetzt in Sexta sind" (Students who are not promoted in class 5), whereas the user wrote: "Schueler, die nicht versetzt sind in Sexta." On the other hand, users can develop sucessful strategies for errors based on application or user dependent word semantics (for example, a synonymous word is not defined).

4.3.3 Conclusions From Stage B. - It does not seem promising to work with heavy restrictions in the area of surface structures when defining a subset for natural language interfaces to databases. Possibilities of variation are used extensively and serious interruptions in man-machine interaction occur when the subset barriers concerning surface structure rules are exceeded. However, it appears that application and user dependent semantics remain within narrow confines. This means that the problem of ambiguity is reduced, the quantity of words to be defined is small, necessary relations can be sufficiently well established before the start of a study, and exceeding the subset boundaries causes only minor and easily manageable breaks in man-machine interactions.

From the results of the KFG field study, the realization of functionally capable natural language query components appears to be possible for the user group of application experts. Remaining problems include determining the nature and size of suitable application fields (transferability), and the relative performance in comparison with formal query languages. Natural language queries are not always superior to formal language queries, and vice versa.

5.0 STAGE C: THE ADVANCED LANGUAGE PROJECT AT NEW YORK UNIVERSITY

5.1 Project History And Study Description

The purpose of the Advanced Language Project (ALP) was to study the English language version of the system in a real application, and in a location remote from that of the development team. The application, a question-answering system about alumni of the Graduate School of Business Administration at New York University, maintains demographic data and donation histories of school alumni, foundations, other organizations, and individuals. The school has over 40,000 graduates as well as some 5,000 non-graduates who have given to the school over the past 20 years. The ALP database contained four base relations with approximately 100,000 tuples, substantially more than in previous applications. Data retrieved from this database usually serve as a basis for decision making in fund raising drives.

The research centered on the question of whether -- in this setting -- the system (as an example for a transportable NLQS) is superior to a formal query language, such as SQL, in terms of learnability, problem-solving success, or effort to use. A comparative study design and mostly quantitative evaluation criteria were chosen for all experiments.

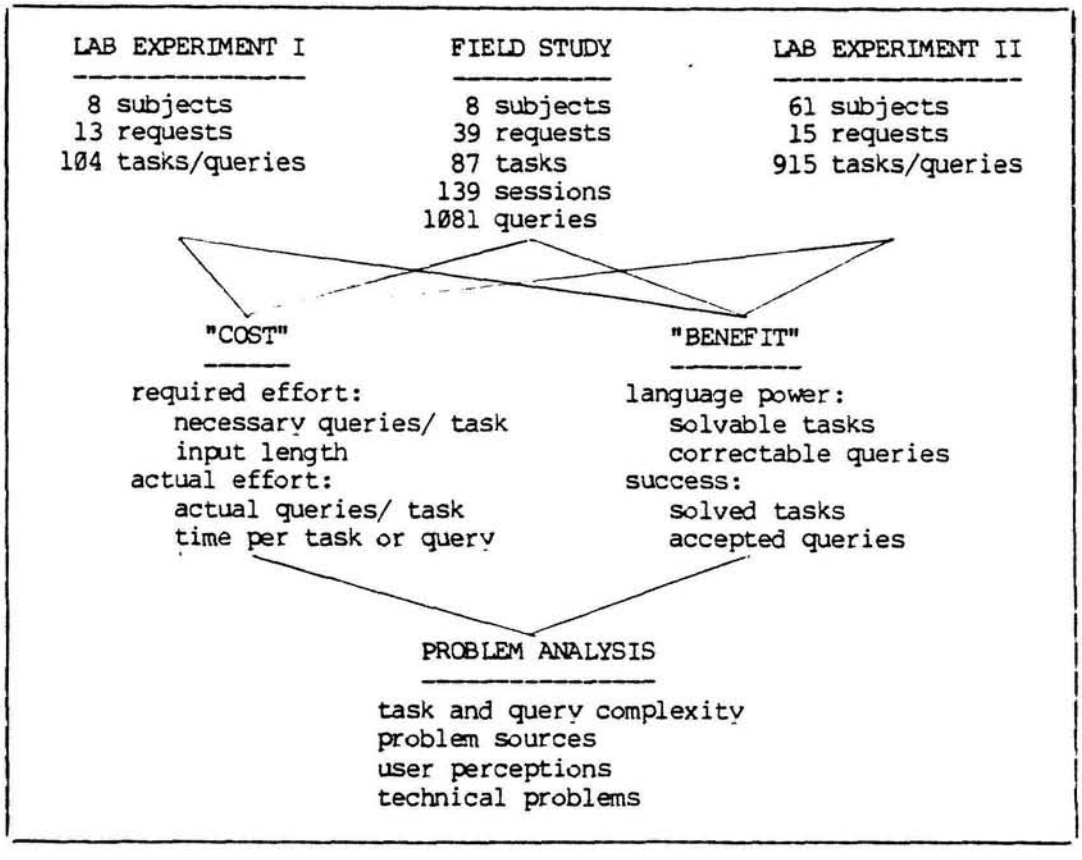

FIGURE 5-1: Evaluation Plan and Descriptive Statistics - ALP Studies

The project design coupled a field study with two controlled laboratory experiments. The experiments began in 1981 with the design and generation of the database and of the application-specific vocabulary, followed by the application and language training and testing of 8 experimental subjects. This skill acquisition phase was organized as a controlled laboratory experiment. After subjects had reached sufficient proficiency in application and language, they performed real work tasks in the actual setting for more than six months (the field study). The field study raised several additional research questions, and the results of the first laboratory test had to be confirmed with a larger number of subjects. Therefore, a second laboratory experiment with 61 subjects was conducted as a paper-and-pencil test in late 1982.

5.2 Methodology

Global design and descriptive statistics of the ALP project are summarized in Figure 5-1. In the following two subsections, the designs of the field study (together with the first laboratory experiment) and the major laboratory experiment are described.

5.2.1 ALP Field Study: Research Design And Evaluation Parameters -

Control of Outside Factors. The decision for a comparative and quantitatively oriented approach inspired a need to control for differences in outside factors, which could blur the results. This problem strongly influenced the research design for the field study.

The first control measure was to use paid intermediaries serving the information users or clients. This not only reduced the danger of losing users due to possibly poor performance of the prototype, but it also increased the number of users for statistical purposes.

Furthermore, the use of intermediaries enabled a counterbalanced and matched design. The field study was divided into two phases so that each subject used both languages but in different sequence (controlling both for inter-subject differences and for order effects). In addition, each work task (request by a client) was assigned to two subjects using different languages, thus controlling for differences in task complexity.

As a final control measure, changes of the application-specific system portion were avoided as far as possible during the field study — only a few problems in the English version were corrected — and a number of complexity measures were developed to ensure comparability in those cases, where perfect matching was prevented by scheduling constraints.

Skill Acquisition. The first laboratory experiment [Turner et al. 1984] served several purposes. The most important one was to make sure that subjects had acquired a level of skill, where acceptable field performance could be expected. Another goal was to determine the amount of training necessary for using a restricted NLI as compared to a formal query language. Finally, the experiment was needed to understand better the relationship between performance in laboratory and field settings; the use of the same subjects, application, and languages seemed to carry some promise in this respect. Previous studies of SQL were used to partially validate the results.

Evaluation Criteria. Data of the user sessions in the field study [Jarke et al. 1984] were captured from session logs and questionnaires, and coded using a multi-level coding scheme. The following kinds of measures were applied:

1. Success of subjects in solving decision-support oriented problems, and in phrasing queries acceptable to the system.

2. Effort required for solving the problem (in terms of input length and of time spent).

3. Factors inside and outside the languages that influence success and effort.

4. User reactions to the languages.

These measures apply at four different levels of measurement (figure 5-2). The main goal is to answer an information request, i.e., a problem description given by a client. Each request was given to one or more subjects as tasks to be solved in their assigned language. The subject could work on a task during one or more continuous sessions. During a session, the subject submitted one or more queries to the system. A system evaluation at the query level alone has been common in laboratory experiments with query languages which frequently use translation tasks for testing. However, in the field setting, this approach was deemed insufficient since it does not capture the contribution of each query to overall task performance.

At all levels, coding of session logs and questionnaires was performed independently by at least two persons, namely one of the researchers and one or more graduate research assistants. In addition, redundancy was designed into the criteria definitions that permitted computerized consistency controls to be implemented.

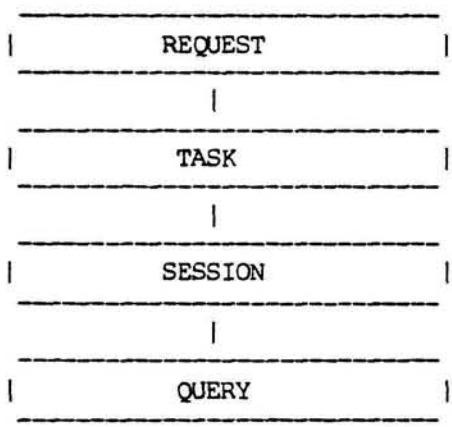

FIGURE 5-2: Evaluation Hierarchy for the ALP Field Study

5.2.2 Laboratory Experiment: Research Design And Evaluation Methodology - The paid subjects of the second laboratory experiment [Vassiliou et al. 1983a] were 61 business graduate and undergraduate students. This type of subjects has been termed as 'novice-casual' [Jarke and Vassiliou 1982]; they have little knowledge of either programming concepts or of the application domain.

Teaching. All subjects were first given a two hour application description. They were then assigned to three groups. Two groups were taught a language (NLQS or SQL) for three and a half hours and the third group was given no language training. Teaching in SQL followed the pattern established in [Reisner 1977; Welty and Stemple 1981]; teaching in the natural language subset concentrated on the language system philosophy and on examples of how to

get around language restrictions. The application was a scaled-down version of the NYU alumni database, which was used in the field study.

Testing. After training, all subjects were given the same pencil-and paper test consisting of fifteen questions. Exam questions were designed with no bias toward the NLQS or SQL. They described problem situations and subjects were asked to express a query (or a series of queries) to answer them (Figure 5-3). The group with no language training was asked to employ English queries.

```
AN EXAM QUESTION

Q6.- A list of alumni in the state of California
     has been requested. The request applies to those
     alumni whose last name starts with an "S".
     Obtain such a list containing last names and
     first names.

NLQS SOLUTION

Q6.- (NLQS). What are the last names and first
             names of all California
             Alumni whose last name is like S% ?

SQL SOLUTION

Q6.- (SQL).  Select lastname, firstname
             From donors
             Where srccode = 'al' and state = 'ca'
                   and lastname like 's%';
```

FIGURE 5-3: Example Exam Question - Second ALP Laboratory Study

Evaluation Method. Exams were graded by two examiners. A series of measures was used with the goal to facilitate comparisons with other laboratory experiments and with the field study. There were three main objectives in the laboratory experiment: a comparative study between the NLQS and SQL for ease of use (performance), a lexicographic analysis for the number and type of tokens used in the two languages, and an examination of the grammatical constructions employed by natural language users.

For the analysis of word usage, an equal number of natural language and SQL subjects were selected. Among the topics investigated were: tokens used per question and per subject, categories for the individual tokens used (e.g. pronouns, verbs), and commonality of word usage using a similar procedure as the one employed by Miller [1981].

The answers of trained NLQS subjects were further considered for the investigation of the general solution strategies followed, and grammatical correctness and naturalness of the constructs used in answering a question. The latter was graded by a student majoring in English.

5.3 Experimental Results

5.3.1 Results Of The ALP Field Study. - The results of the field experiment [Jarke et al. 1984], coupled with those of the first laboratory test [Turner et al. 1984] concern issues of training, language power, user effort and success, system problems and user perceptions and strategies.

Training. The 8 subjects achieved an acceptable level of skill (comparable to that of previous experiments with SQL [Reisner 1977; Greenblatt and Waxman 1978; Welty and Stemple 1981]) after three hours of classroom training and several practice terminal sessions followed by a refresher classroom hour. No difference was found between NLQS and SQL performance.

Language Power. Even though no perfect matching was achieved, the assigned tasks in each language were of roughly equal complexity. However, task complexity decreased somewhat over time in the languages. Both languages showed a reasonably good functional coverage of the application but SQL was somewhat more powerful, in the sense that more tasks were solvable and slightly less queries were required in principle to resolve a task than in the NLQS. (A task was said to be completely solvable (with x necessary queries) if all the required data were available in the database, and if a specialist could find a way to solve the underlying request with x queries in the given language.)

EVALUATION CRITERION	NLQS	SQL
LANGUAGE POWER		
% completely solvable tasks	73.8%	84.4%
no. necessary queries per task	4.4	3.2
EFFORT SPENT (TASK LEVEL)		
no. queries submitted per task	15.6	10.0
time spent per task (minutes)	120	108
EFFORT SPENT (QUERY LEVEL)		
no. tokens per query (input)	10.6	34.2
time spent per query (minutes)	7.7	10.6

TABLE 5-1: Language Power and Effort to Use - ALP Field Study

Effort To Use. There was little difference between the two languages in terms of the total time subjects needed to complete a task. The average number of NLQS queries submitted per task was significantly higher than the number of SQL queries. However, SQL queries were three times longer than natural language queries and required 40% more total time per query, demonstrating the potential savings of using NLQS.

Success in Problem-Solving and Querying. The actual task level performance in both languages was much lower than one would have expected from laboratory results and language power. SQL achieved 44% and the NLQS 17% essentially correct solutions over all tasks. Natural language performance improved considerably (from 4.8% to 30%), after some initial system errors had been removed. In a direct request-by-request comparison, the NLQS was superior in 18% of the paired requests (21% equal, 61% SQL better). Query correctness was equally low, in terms of percentage of accepted queries as well as in terms of percentage of queries judged essentially correct except for trivial errors.

Experiment	Evaluation Criterion	NLQS	SQL
ACTUAL TASK SOLUTION PERFORMANCE			
Field	% essentially correct tasks	17.1%	44.2%
- phase 1	% essentially correct tasks	4.8%	39.1%
- phase 2	% essentially correct tasks	30.0%	50.0%
ACTUAL QUERY ANSWERING PERFORMANCE			
Field	% accepted queries	15.2%	26.5%
Field	% essentially correct queries	22.3%	45.6%

TABLE 5-2: Task and Query Level Performance - ALP Field Study

Problem Analysis. Interface and system unavailability problems -- caused by heavy system load, and by the use of hardcopy terminals and noisy dial-up lines -- were common to both languages but had a stronger impact on NLQS task performance, since natural language users had more difficulties in recognizing the source of a problem. The large size of the database (as compared to previous studies) also had a negative impact on system performance, not only because of long search times but also because inefficiencies of the NLQS—SQL translation coincided with certain weaknesses in the SQL query optimizer, leading to inefficient query processing. Response times of more than 10-20 minutes were not infrequent.

Otherwise, NLQS failures were mostly attributed to lack of language functionality or omissions in the application design, whereas user errors were the cause of most SQL failures. This can be interpreted in the way that more habitability (tolerance of surface structure variations) must be expected from an NLI than from a formal query language.

Interestingly, the number of typographical errors was quite small. The error rate was almost exactly the same in both language (0.97% respectively 0.94% of all input tokens contained errors), which is at the lower end of the spectrum to be expected from inexperienced typists [Embley and Nagy 1981]. One might interpret this as an indication that users are very careful in their computer interaction.

	TASK LEVEL		SESSION LEVEL	
main problem	NLQS	SQL	NLQS	SQL
lack of data/function	28.5%	14.8%	34.5%	13.0%
user problem	11.5%	55.7%	12.0%	55.6%
interface/system problem	34.3%	11.1%	38.0%	31.4%
combination of problems	25.7%	18.5%	15.5%	0.0%

TABLE 5-3: Reasons for Failure - ALP Field Study

User Perceptions and Strategies. Error handling strategies were different for the two languages. Natural language users had the tendency to rephrase a query in a different way, whereas SQL users usually retried the same query with only minor modifications. This gives a hint on the difficulties NLI users had in locating and correcting errors, given the poor error messages of the prototype. Not surprisingly, users rated the suitability of SQL for task solution higher.

5.3.2 Results Of The Laboratory Experiment. - There were three major objectives in the larger of the two laboratory studies: comparative performance of NLQS and SQL subjects and the impact of training, examination of the effect of subset boundaries for the natural language system, and determination of the grammatical correctness and naturalness of subjects' queries.

Performance. No significant differences in test scores were found between and SQL subjects - see Table 5-4 (t-test, n=51, p=.110). Users of the NLQS required less time and tokens per query - see Table 5-5 (t-test, n=765, p=.000). It was observed that training in the natural language subset is necessary, as evidenced by the poor performance of the untrained subject group (Table 5-4).

	NO. OF SUBJECTS	CORRECTNESS Mean	S.D.
Trained SQL	17	71.4	22.7
Trained NLQS	34	68.9	23.1
Untrained NLQS	10	28.3	18.9

TABLE 5-4: Performance of Subjects - ALP Laboratory Study

	TOKENS PER QUERY	TIME Mean	S.D.
Trained NLQS	21.2	3.06	1.92
Trained SQL	33.8	4.75	3.33

TABLE 5-5: Time and Tokens per Query - ALP Laboratory Study

Subset Boundaries. Natural language was less verbose than SQL, but had a larger vocabulary to draw upon (i.e., the number of unique words used in natural language was higher). Still, it was found that the size of this vocabulary was manageable; approximately 150 words would have to be defined for the application (nouns, adjectives, non-imperative verbs). Furthermore, NLQS subjects shared many such words, while infrequently used words (accounting for forty five percent of all unique words in the vocabulary) could probably have been dropped without serious performance problems (Table 5-6).

	NUMBER OF WORDS	APPLICATION DEPENDENT	APPLICATION INDEPENDENT	CONSTANT VALUES
UNIQUE WORDS				
NLQS	259	56%	24%	20%
SQL	180	50%	13%	37%
TOTAL WORDS				
NLQS	4478	44%	45%	11%
SQL	6081	28%	61%	11%

TABLE 5-6: Summary of Word Usage - ALP Laboratory Study

Grammatical Correctness and Naturalness. Even after training, NLQS subjects had a strong tendency to write non-grammatical queries. In addition, the subjects used fairly awkward expressions in attempting to meet the artificial restrictions of the NLQS subset.

5.3.3 Conclusion Of Stage C. - In both languages, performance in the field study appeared to be substantially lower than in the laboratory experiments. Since the evaluation criteria differed, the definitions of essentially correct and correctable queries from [Welty and Stemple 1981] were applied to the queries in all experiments (table 5-7). There is still a gap between the field and laboratory studies but the results on 'correctable' queries also emphasize the potential of a better adapted system. The SQL results are comparable to those found by Welty and Stemple [1981] who report 67.0% respectively 59.5% 'essentially correct' queries for two groups of subjects. Thus, the ALP laboratory results appear to be consistent with previous research, especially if the extremely short training period in the second experiment (less than three hours) is taken into account.

Experiment	essentially correct		at least correctable	
	NLQS	SQL	NLQS	SQL
Lab I	71.1%	67.3%	78.8%	76.9%
Lab II	44.6%	53.3%	59.2%	68.8%
Field	22.3%	45.6%	75.5%	57.0%

TABLE 5-7: Query Level Performance Overview - ALP Experiments (Welty Scale)

6.0 SYNOPSIS OF EMPIRICAL STUDIES

In this section, we investigate the relationships between the data gained by the evaluation studies of stage B (KFG) and stage C (ALP). Having a common empirical base, we point out the major results and attempt to explain the differences.

6.1 Assessment Of Methods And Research Designs

General Evaluation Plan. In contrast to previous evaluations of NLQS, neither the ALP nor the KFG study were carried out by the development team. Both studies combine field studies and laboratory experiments. Considering the plan underlying the KFG study (see Figure 4-1), ALP worked in a new application field and a new natural language (English). From this viewpoint, the most important progress is that extensive experiments to compare formal with natural language were conducted in ALP. From the viewpoint of ALP, the most important new element is the orientation on tasks and the analysis of the hierarchical levels request-task-session-query. The KFG laboratory experiments used translation tests and the KFG field study can be seen as one big task. Schuetz [1979] describes this task and shows its successful solution with the NLQS. But there is no analysis of subtasks in KFG, nor of the relationships between the hierarchical levels.

Field Studies. The KFG field study extended over a longer period of time (16:6 months), and the amount of queries was larger (7278:1081). On the other hand, there were eight users in ALP. The ALP users were not real users, in the sense of the KFG and TA field study, but ALP was very close to a real usage. Therefore, there is some common ground for a relating results from the KFG and ALP field studies. Results of the KFG field study which can be confirmed by the ALP material will gain more plausibility with respect to their user and application independence. However, there is still a major barrier to bringing the two studies together. Section 4.3 shows that the most important results of KFG came from a qualitative analysis of the queries, mainly concerning the internal grammatical structures and error situations, especially error chains and strategies to handle them. In the terminology of ALP, an additional layer is missing between the hierarchical levels session and query, namely 'error chains/query sequences.' Error chains are not determined by the task but by a user's effort to get an answer to one initially erroneous query. This level is only marginally covered by the statistical approach of ALP.

Laboratory Experiments. Laboratory experiments were methodologically stronger in ALP. One question remains open: why are the KFG results so much better? The conjecture that people learn ISBL easier than SQL does not appear to be plausible. Another conjecture is that the result differences can be attributed to differences in the design of the tests (tasks vs. translation test of a typical session).

6.2 Comparison Of The Results

6.2.1 Common Results. - Based on results of both studies, five statements seem to have a fairly strong empirical backing.

1. Users do not communicate with a NLI in the way they do with a human, as suggested in [Chapanis 1973] (see also [Krause 1980; Zoltan et al. 1982; Zoeppritz 1983a]). In particular, they are very careful in typing input, as evidenced by a low percentage of typographical errors. It is open, how this would change with widespread availability of automatic spelling correction for NLI.

2. Small vocabulary subsets are sufficient for restricted application areas. This result may not extend to some of the knowledge-based systems which require the definition of all words used (including, in particular, values appearing in the database, see, e.g., [Bates and Bobrow 1983]).

3. Natural language is more concise than formal query languages. In particular, SQL requires substantially longer input even for rather simple queries.

4. Formal query languages cannot be rejected on the grounds that a substantial effort is needed to learn them.

5. Neither study confirmed the fear that natural language queries grow more and more complex over time. Rather, there seems to be evidence that users adapt to what they perceive as the system's limitations. In the KFG field study, query complexity remained about stable over time, whereas in ALP it actually decreased.

USER GROUP	NO. USERS	SESSIONS	QUERIES	ERROR RATE
STAGE A				
Planning	2		59	46.0%
School	1		356	47.0%
Reception	1		115	12.9%
Rooms	3		781	39.9%
STAGE B				
KFG main 1	1	39	6603	6.9%
KFG user 2	1	5	582	16.9%
KFG user 3	1	1	93	31.1%
TA study	1	1	67	52.7%
STAGE C (*)				
ALP phase 1	4	34	256	77.0%
w/o line noise				69.1%
ALP phase 2	4	31	271	82.3%
w/o line noise				74.9%

(*) ALP figures do not contain incomplete query typing attempts.

TABLE 6-1: Performance Overview NLQS Field Studies

6.2.2 Open Questions And Discrepancies. - On first sight, the main discrepancies between the results of ALP and KFG concern the error rates (Table 6-1). The most plausible explanation regarding the differences in the laboratory experiments seem to be deviations in the test designs. A second startling discrepancy is visible in the number of queries per session, resulting from the differences in time per submitted query. Possible explanations of the poor showing of the NLQS in the ALP field study in contrast to the good results in the KFG field study could be:

Language Dependence. The English syntax of the NLQS has been written on the model of the syntax for German. For example, morphological rules and the user-independent vocabulary were replaced, and the rules for dependent clause word order were deleted. The interpretation routines are the same as in the German version with some minor modifications [Zoeppritz 1983b].

ALP was the first application of the English system version. Therefore the simplest explanation of the high error rates would be that there was still a need for debugging tests. The high error rates reported for the stage A experiments (which were achieved in a far better technical environment, using smaller databases and screens instead of hardcopy terminals) give some support to this conjecture. Another more far-reaching conclusion could be that the differences between an efficient English and German subset are more extensive than expected. For instance, so-called ungrammatical queries were used more often in ALP than in KFG.

Database Dependence. While the database schemata of KFG and ALP as well as those of the stage A studies were of comparable complexity (two to six base relations), the size of the ALP database turned out to cause serious response time problems through inefficient translation of natural language into SQL. This does not affect the general concept of the system but stresses the necessity of query optimization in the natural language system.

User Dependence. Since the KFG study was mainly a one-user study, it could be suspected that the main KFG user was a happy coincidence and that the very long usage period and his involvement in the application design provided him with a deeper understanding of the system. On first sight, the fact that KFG was the only application reaching such a low error rate would seem to confirm this assumption. Even the other two KFG users had somewhat higher error rates. However, one has to be cautious: Krause [1982] shows clearly that the main KFG user had few changes in error rates over time, thus denying a learning effect after the initial phase.

Application Development Dependence. The system had to be adapted to the ALP application by defining the application vocabulary and the relational view definitions. Since the ALP team had difficulties in handling an SQL limitation in the number of views, and the geographical distance between the development team in West Germany and New York worked as an information barrier, it could be that the adaptation of the new application failed to be accurate. On the other hand, one of the system's claims is the ability to have non-linguists define their own application. ALP clearly demonstrated the limitations of this option.

Experimental Design Dependence. The application-specific part of the ALP grammar was hardly changed after initial testing, whereas the KFG application was adapted whenever problems became visible in a user session. On one hand, the KFG experience shows that the NLQS is powerful enough to cover the language subset required for a particular application in an impressive manner (93% success). Moreover, it is perfectly acceptable to expect a certain period of time, during which the system has to be adapted to a user. On the other hand, the question arises: when will this user adaptation terminate? The answer is clearly important for the commercial (rather than technical) feasibility of NLI.

Technical Environment Dependence. A final reason for the high NLQS failure rates in ALP is obvious when looking at the EDP protocols: the poor system performance at New York University (caused by slow and noisy communication lines, and system overload), and difficulties with the operating system.

7.0 CONCLUSIONS

The comparison of several experiments with the same domain-independent natural language query system has yielded methodological results and preliminary conclusions about this type of natural language interface, as well as gaps in the studies and opportunities for future research.

Research Methodology. There seems to be a natural sequence to be followed in the evaluation of a natural language query system in order to yield meaningful results. Starting with exploratory on-site system tests, the strategy proceeds towards a qualitative feature analysis, upon which structured quantitative evaluation models can be based. The ALP experience has demonstrated that such a schema can be exploited to its fullest only if the prototype under study has reached sufficient maturity; otherwise, quantitative analyses must be complemented by qualitative studies in order to separate generalizable results from those influenced by the prototype status of the system. It is also critical to provide an adequate technical environment.

Domain-Independent Natural Language Query Systems. Concerning the three introductory questions set forth about domain-independent natural language query systems, some conclusions can be drawn, whereas other issues require further study. Addressing first the desirability question, we know that natural language allows for more concise query input and requires less formulation time than a formal query language. However, nobody has been able so far to demonstrate advantages of natural language over formal query languages in terms of learnability, language power, task performance, or query acceptance rates.

Concerning NLQS feasibility, there is no evidence that any of the experiments exceeded the boundaries of what can be easily implemented within the domain-independent subset system approach. Thus, practice-oriented natural language query systems appear to be technically feasible and able to fulfill the purpose they were developed for. However, additional studies will be required to confirm this result.

The third question asked for the cost of adapting a NLQS to a new application. It is not clear how long the adaptation to an application or a new user will take, or to what degree end users will be able to take over this job from specialists in computational linguistics. The experience with ALP indicates that building and stabilizing a new application needs major linguistic information science (computational linguistic) support. That is, different personnel requirements from those for introducing an end user system based on formal query languages may arise [Vassiliou et al. 1983b].

Future Research. The intensive study of a natural language query system has revealed a number of empirical research questions that have to be answered to bring natural language closer to practical usability. As a first step, it is planned to further explore the reasons for the differences in performance between the ALP and KFG studies in order to make the results more comparable. For KFG, this means that the laboratory experiments will be repeated in a more controlled setting, using queries of the ALP application and SQL as the formal query language. This supplementary test promises interesting results for the comparison of the different design decisions in the laboratory experiments of ALP and KFG.

For ALP, a qualitative re-analysis of the protocols will be performed to make the results compatible with KFG. Another reason for qualitatively reanalyzing the ALP material with respect to error chains comes from the observation that the task orientation in ALP was partly impeded by the high error rates, which in turn led to the existence of almost only error chains in many sessions.

There are first hints that in addition to the performance problems some gaps and inadequacies in the application-dependent part of the NLQS are partially responsible for the high error rates in ALP. There are no hints so far that the general philosophy of domain-independent NLQS is insufficient. But these statements are subject to change pending further evidence.

Acknowledgments:

This work is based on several studies in cooperation with the IBM Corporation. The projects would not have been possible without the continued support by members of IBM Heidelberg Scientific Center, in particular, A. Blaser, H. Lehmann, N. Ott, and M. Zoeppritz. Besides the authors, principal investigators of the Advanced Language Project were Ted Stohr, Jon Turner, and Norm White. We would also like to express our gratitude to the subjects who participated the studies presented here, and to Margi Olson for many helpful suggestions concerning the presentation.

REFERENCES

1. Artificial Intelligence Corporation: Intellect Query System Reference Manual, 1982.

2. Astrahan, M.M. et al., "System R. A relational approach to data", ACM Transactions on Database Systems 1, 2 (1976), 97-137.

3. Barber, R.E., Lucas, H.C., "System response time, operator productivity, and job satisfaction", Communications of the ACM 26 (1983), 972-986.

4. Bates, M., Bobrow, R.J., "A transportable natural language interface for information retrieval", Proceedings 6th ACM-SIGIR Conference, Washington, D.C., June 1983.

5. Bertrand, O., Daudenarde, J.J., du Castel, B., "User Language Generator. Program description/operation manual", IBM France, Paris 1981.

6. Bundy, A. (ed.), Proceedings of the 8th IJCAI, Karlsruhe 1983.

7. Chapanis, A., "The communication of factual information through various channels", Information Storage and Retrieval 9 (1973), 215-231.

8. Codd, E.F., et al., "RENDEZVOUS version 1: an experimental English-language query formulation system for casual users of relational data bases", IBM Research Report RJ2144, San Jose, California, 1978.

9. Damerau, F.J.: "The Transformational Question Answering (TQA) System. Operational Statistics", American Journal of Computational Linguistics 7, 1 (1979), 30-42.

10. Embley, D.W., Nagy, G., "Behavioural aspects of text editors", ACM Computing Surveys 13, 1 (1981), 33-70.

11. Finnin, T., Goodman, B., Tennant, H., "JETS: Achieving completeness through coverage and closure", Working Paper, University of Illinois, Champaign/Urbana 1979.

12. Fischer, H.G. (ed.): Information Retrieval und natuerliche Sprache. Integrierte Verarbeitung von Daten und Texten im Modell CONDOR. Munich 1982.

13. Ghosh, S.P., Data Base Organization for Data Management, Academic Press, New York 1977.

14. Greenblatt, D., Waxman, J., "A study of three database query languages", in B.Shneiderman (ed.): Databases: Improving Usability and Responsiveness, Springer 1978, 77-97.

15. Harris, L.R., "User oriented data base query with the ROBOT natural language system", Proceedings 3rd VLDB Conference, Tokyo 1977, 303-311.

16. Hendrix, G.G., "A natural language interface facility and its application to a IIASA data base", in G.Rahmstorf, M.Ferguson (eds.): Proceedings of a Workshop on Natural Language for Interaction with Data Bases, Laxenburg 1978, 87-94.

17. IBM: MIS/370 Anwendung. IV Informations-Systeme, 1976.

18. Jarke, M., "Zur Beurteilung natuerlichsprachlicher Endbenutzerschnittstellen von Datenbanken", in J.W. Schmidt (ed.): Sprachen fuer Datenbanken, Springer 1983, 42-60.

19. Jarke, M., Turner, J.A., Stohr, E.A., Vassiliou, Y., White, N.H., Michielsen, K., "A field evaluation of natural language for data retrieval", IEEE Transactions on Software Engineering, forthcoming 1984.

20. Jarke, M., Vassiliou, Y., "Choosing a database query language", submitted for publication, November 1982.

21. Kettler, W., Schmidt, A., Zoeppritz, M., "Erfahrungen mit zwei natuerlichsprachlichen Abfragesystemen", IBM Heidelberg Scientific Center, TR 81.01.001, 1981.

22. Kolvenbach, M., Loetscher, A., Lutz, H.D. (eds.): Kuenstliche Intelligenz und natuerliche Sprache. Sprachverstehen und Problemloesungen mit dem Computer. Tuebingen 1979.

23. Krause, J., "Natural language access to information systems: an evaluation study of its acceptance by end users", Information Systems 4 (1980), 297-318.

24. Krause, J.: Mensch-Maschine-Kommunikation in natuerlicher Sprache, Niemeyer, Tuebingen 1982.

25. Krause, J., "Praxisorientierte natuerlichsprachliche Frage-Antwort-Systeme: Zur Entwicklung vor allem in der Bundesrepublik Deutschland", Nachrichten fuer Dokumentation 34, 4/5 (1983a), 188-194.

26. Krause, J., "Linguistic components in (office) information systems and a general evaluation strategy for automatic indexing", Journal of Information and Optimization Sciences, no. 87 (1983b).

27. Krause, J., Lehmann, H., "The User Specialty Language. A natural language based information system and its evaluation", in D.Krallmann (ed.): Dialogsysteme und Textverarbeitung, Essen 1980, 127-146.

28. Krause, J., Schneider, Ch., Spettel, G., Wormser-Hacker, Ch., "EVAL. Zur Evaluierung informationslinguistischer Komponenten von Informationssystemen", Regensburg 1983.

29. Lehmann, H., "Interpretation of natural language in an information system", IBM Journal of Research and Development 22, 5 (1978), 560-572.

30. Lehmann, H., Blaser, A., "Query languages in database systems", IBM Heidelberg Scientific Center TR 79.07.004, 1979.

31. Lehmann, H., Ott, N., Zoeppritz, M., "User experiments with natural language for database access", Proceedings 7th International Conference on Computational Linguistics, Bergen 1978.

32. Malhotra, A., "Design criteria for a knowledge-based English language system for management", MIT Project MAC, Cambridge, Mass. 1975.

33. Malkovsky, M., "TULIPS-2 natural language learning system", Proceedings Coling 82, Prague 1982.

34. Miller, L.A., "Natural language programming: styles, strategies, and contrasts", IBM Systems Journal, 20, 2 (1981), 184-215.

35. Moran, T., "An applied psychology of the user", ACM Computing Surveys 13, 1 (1981), 1-12.

36. Morik, K., "Differenzstudie zu frueheren sprachverarbeitenden Systemen in der Bundesrepublik Deutschland", HAM-ANS Report 6, Hamburg 1982.

37. Ott, N., "Bericht ueber die KFG-Studie", IBM Heidelberg Scientific Center TN 79.03, 1979a.

38. Ott, N., "Das experimentelle, auf natuerlicher Sprache basierende Informationssystem USL", Nachrichten fuer Dokumentation 30, 3 (1979b), 129-140.

39. Ott, N., Zoeppritz, M., "USL - an experimental information system based on natural language", in L.Bolc (ed.): Natural Language Based Computer Systems, Macmillan, London 1979, 3-32.

40. Petrick, S.R., "On natural language based computer systems", IBM Journal of Research and Development (1976), 314-325.

41. Reisner, P., "Use of psychological experimentsas an aid to the development of a query language", IEEE Transactions on Software Engineering SE-3 (1977), 218-229.

42. Reisner, P., "Human factors studies of database query languages: a survey and assessment", ACM Computing Surveys 13, 1 (1981), 13-32.

43. Schneider, M., "Ergonomic considerations in the design of control languages", in Y. Vassiliou (ed.): Human Factors and Interactive Computer Systems, Ablex, Norwood, NJ, 1984.

44. Schuetz, F., "Noten am KFG. Zufall oder Notwendigkeit", in Elternvereinigung des KFG Mannheim (eds.): Karl-Friedrich-Gymnasium Mannheim. Jahresbericht, Mannheim 1979, 39-87.

45. Shwartz, S.P., "Natural language processing in the commercial world", in W. Reitman (ed.): Artificial Intelligence Applications for Business, Ablex, Norwood, NJ, 1984, 235-248.

46. Shneiderman, B., Software Psychology, Winthrop 1980.

47. Small, D., Weldon, L.J., "The efficiency of retrieving information from computers using natural and structured query languages", Report SAI-78-655-WA, Science Applications, September 1977.

48. Stohr, E.A., Turner, J.A., Vassiliou, Y., White, N.H., "Research in natural language systems", 15th Annual Hawaii Conference on System Sciences, Honolulu, Hawaii, January 1982.

49. Tennant, H.R.: Evaluation of natural language processors, Ph.D. diss., University of Illinois, Urbana 1979.

50. Todd, S.J.P., "The Peterlee Relational Test Vehicle", IBM Systems Journal 15, 4 (1976), 285-308.

51. Turner, J.A., Jarke, M., Stohr, E.A., Vassiliou, Y., White N.H., "Using restricted natural language for data retrieval - a plan for field evaluation", in Y. Vassiliou (ed.): Human Factors and Interactive Computer Systems, Ablex, Norwood, NJ, 1984.

52. Vassiliou, Y., Jarke, M., "Query languages - a taxonomy", in Y. Vassiliou (ed.): Human Factors and Interactive Computer Systems, Ablex, Norwood, NJ, 1984.

53. Vassiliou, Y., Jarke, M., Stohr, E.A., Turner, J.A., White, N.H.: "Natural language for database queries: a laboratory study", MIS Quarterly 7, 4 (1983a), 47-61.

54. Vassiliou, Y., Jarke, M., Stohr, E.A., Turner, J.A., White, N.H., "Application development for natural language query systems", Proceedings IEEE Workshop on Languages for Automation, Chicago 1983b, 288-293.

55. Waltz, D., "An English language question answering system for a large relational database", Communications of the ACM 21 (1978), 526-539.

56. Waltz, D., "Artificial Intelligence: an assessment of the state-of-the-art and recommendations for future directions", AI Magazine 4, 3 (1983), 55-67.

57. Welty, C., Stemple, D.W., "Human factors comparison of a procedural and a non-procedural query language", ACM Transactions on Database Systems 6, 4 (1981), 626-649.

58. Woods, W., "Lunar rocks in natural English: explorations in natural language question answering", in Zampolli (ed.): Linguistic Structures Processing, North Holland (1977a).

59. Woods, W., "A personal view of natural language understanding", in D.Waltz (ed.): Natural Language Interfaces, SIGART Newsletter 61 (1977b), 17-20.

60. Zoeppritz, M., "Human factors of a 'natural language' enduser system", in A.Blaser, M.Zoeppritz (eds.): End User Systems and their Human Factors, Springer (1983a), 62-93.

61. Zoeppritz, M.: Syntax for German in the User Specialty Languages System. IBM Heidelberg Scientific Center Heidelberg (1983b).

62. Zoltan, E., Weeks, G., Ford, W.R., "Natural language communication with computers: a comparison of voice and keyboard input", in G.Johannsen, J.E.Rijsdorp (eds.): Analysis, Design, and Evaluation of Man-Machine Systems, Baden-Baden 1982, 27-28.