NAUDA
A Cooperative, Natural Language Interface to Relational Databases

D. Küpper[a], M. Strobel[ab], D. Rösner[a]

[a]Research Institute for Applied Knowledge Processing (FAW),
Postfach 2060, D-7900 Ulm, Germany
Tel.: ++49 731 501-530,
e-mail: <Lastname>@faw9370.faw.uni-ulm.de

[b]IBM Deutschland GmbH,
Institute for Knowledge Based Systems (IKBS),
Postfach 103068, D-6900 Heidelberg 1, Germany
Tel.: ++49 6221 404-288,
e-mail: STROBEL@DHDIBM1.bitnet

ABSTRACT

The NAUDA[1] System is a cooperative natural (German) language database interface for relational databases. The project is carried out at FAW[2] - funded by the state of Baden-Württemberg and IBM Germany. This paper describes the extension of a natural language interface to relational databases with respect to its cooperative behavior. We argue that cooperative support of users is especially important for a complex domain such as environmental protection. In order to enrich traditional database reports our system provides dialog-oriented features such as over-answering (providing more information than explicitly requested) and handling of presupposition failure, as well as presentation-oriented features such as natural language responses and geographical maps. Additional information by the system includes (meta-) information *about* the domain.

1. INTRODUCTION

Having the right information at the right time is an important topic in the field of environmental protection, e.g. in case of emergency. More and more information is being stored in databases which are becoming highly complex. Consequently, fast and easy access to required information for humans can only be achieved, either if they are database experts or if they are supported by a powerful access tool.

With this in mind, we are working on a system which allows a user to access a relational database with written German. For demonstration purpose restricted speech input is possible using a speech recognizer.

We are using the water and waste management database from the Baden-Württemberg Ministry of Environment as a realistic application. The relevant concepts in this domain are buildings, areas, bodies of water, and plants (together with their relationships). Currently anticipated users are employees at the local environmental departments and at the Ministry of Environment.

We assume that the users are familiar with the domain, but unfamiliar with the structure of the database. Therefore, assistance is needed in bridging this gap. Natural language is one means of closing the gap, since users may express queries in their own terms. The translation process from natural language to a formal database language requires a model of the database structure and the mapping of natural language concepts onto database expressions. This may help users to some extent, but they must know what concepts and relations have been modelled. Furthermore, they must be aware of the formal character of the translation process, i.e. their queries are answered literally. More cooperative systems try to give an answer which considers the assumed intentions of the user. NAUDA's cooperative behavior focuses on the correction of presupposition failures and over-answering. In addition, tables as traditional output forms are replaced and/or enriched by natural language and graphics.

2. MOTIVATION

The need for a cooperative natural language interface for complex databases can be best motivated by a short comparison with the normal methods for accessing databases. There are two main alternatives available, namely access via menus or via a formal database query language such as SQL. Menu-driven techniques are easy to use, but are not flexible enough for addressing more complex problem solving tasks. Unrestricted access to a database is currently possible with a formal language only. However, formulating a query with such a language is often time consuming and susceptible to errors. It requires proficient data processing skills and knowledge about the syntax and semantics of the language, as well as the names of the tables, columns, datatypes and possible join paths. To illustrate this, Figure 1 shows the

---

1 NAtürlichsprachlicher Zugang zu Umwelt DAten
Natural Language Access to Environmental Data

2 Forschungsinstitut für anwendungsorientierte Wissensverarbeitung
Research Institute for Applied Knowledge Processing

SIGMOD /5/93/Washington, DC,USA
© 1993 ACM 0-89791-592-5/93/0005/0529...$1.50

```
SELECT DISTINCT
WWAO.Obj_Art,WWAO.Obj_Nr,WWAO.Objektname,
AW04.Bezeichnung
FROM "ABW/ABF_Anlagen" A,
     Abwassreinanl AWR,
     "Abwrein-Bauwdat" AWR_BWD,
     Abwasserbauwdaten ABWBWD
     AW04,   WWAO,
WHERE
 WWAO.Obj_Art       = A.Obj_Art          AND
 WWAO.Obj_Nr        = A.Obj_Nr           AND
 A.Abfall_ID        = AWR.Abfall_ID      AND
 AWR.Abwreinanl_ID =
                AWR_BWD.Abwreinanl_ID AND
 AWR_BWD.Abwassbauwdat_ID =
              ABWBWD.Abwassbauwdat_ID AND
 ABWBWD."Bereich/Stufe"= AW04.SCHL_ID AND
 AW04.Bezeichnung = 'CHEMISCHE VERFAHREN'
```

Figure 1: SQL-statement corresponding to the sentence below

SQL-query to our database which corresponds to the natural language sentence:

Which sewage treatment plants have a chemical purifying stage?

Natural language provides a means for unrestricted access, which does not require data processing skills or knowledge of the internal structure of the database.

The benefit of a natural language interface will increase, however, if the system provides cooperative behavior.

3. THE NAUDA SYSTEM

3.1 COOPERATIVE BEHAVIOR OF THE NAUDA SYSTEM

Cooperative behavior means supporting a user in the sense, that a system's response is as useful as possible with respect to the user's task(s). Although NAUDA does not deal with recognizing or estimating the user's task(s), several kinds of support for users may be provided in general within the scope of an information system. Therein we distinguish between two major branches. One is related to the use of natural language, i.e. to the cases in which an information system should provide more information than required by a literal interpretation of a query. The other deals with an adequate form of a system response, i.e. presenting the information in a way which is comprehendable and avoids syntactically ambiguous sentences when generating natural language output.

3.1.1 PRESENTATION OF RESULTS

In order to reach a high degree of user-friendliness, the form of the output should be appropriate for the type of information. Various phenomena may be fairly well expressed in natural language, whereas for others, graphics or combinations of language and graphics are more suitable. In addition to properties concerning the contents, aspects such as the amount of data play a crucial role in the mode of presentation.

In the NAUDA system different forms of presentation are chosen depending on the type and amount of information:

- The standard form of an answer to a w/h-question (i.e. a question containing "which", "where", "when", etc.) is a table. The result from the database manager remains unchanged for tables containing a large amount of data. But the presentation of tables consisting of a single column heading and a single data value seems to cause an information overload. Instead, the result should be provided in natural language – either as a sentence or a noun phrase – or simply as a value.
- Yes-no questions are not simply rejected or confirmed but are – if appropriate – answered as full sentences. This is salient and can, in the context of over-answering, be seen as a useful technique for directing the user's attention to the fact that s/he received more information than explicitly requested.
- The system has been enriched by a facility for the display of topographical maps which show sections of the area in question, an aspect which is especially relevant in our domain. Spatial relationships between objects should be visualized for the following reason: expressing spatial relationships with linguistic means or as a table is rather inefficient and inadequate and can often be done much better visually.

3.1.2 OVER-ANSWERING

Within information-seeking dialogs between humans, the informant often provides more information than explicitly required if it is relevant for the information seeker. This cooperative behavior is called *over-answering* [Wahlster et al. 83]. The problem of adapting this technique for an information system lies in adequately constraining the amount of additional information given to the user, i.e. avoiding the presentation of irrelevant facts. We have identified two cases in which we believe additional information is generally relevant. The first one is the presence of wrong presuppositions within the user's question; this will be discussed in the next section. The other deals with a user's yes-no question about a property value of a certain object when this question must be answered with "No". In this case, the system will assume that a) the user does not know the actual value of this attribute and b) the user is interested in this value. Consequently, the system should give the user the valid attribute values. Our system, for instance, will give the actual value of a numerical attribute if a corresponding yes-no question from the user must be answered with "No".

3.1.3 HANDLING PRESUPPOSITION FAILURE

Within the scope of natural language interfaces to databases, frequently used types of questions are "how-many" and "which/who" questions [Lehmann et al. 78]. These questions usually contain an object description where the user is interested in the number of such objects or s/he requests more information about them. Such queries presuppose that these objects are in the database and, consequently, that they are legal in the domain.

For instance, the question

Which simple sewage treatment plants have a chemical purifying stage?

contains the presupposition that the database stores at least one *simple sewage treatment plant with a chemical purifying stage.* This implies (among other things) that the database may store simple sewage plants, purifying stages, that a (possessive) relation between simple sewage plants and purifying stages exists, and that the range of this relation subsumes purifying stages of type 'chemical'.

If one of these conditions fails the database answer will be "No (empty set)" or something similar. In contrast, within normal cooperative dialogs the failure will be noticed and explained if possible. On the other hand, this means that presuppositions which are not rejected explicitly might be regarded as confirmed. Because users of a natural language interface expect that this kind of interaction is comparable to normal situations where natural language is used (see [Joshi 83]), a presupposition failure must be noticed to prevent the user from assuming that the system agrees with his presupposition. The above example contains a presupposition failure with respect to the structure of the domain, which implicitly reveals that the user is not familiar with this topic. Because this knowledge is relevant for exploiting the database, the system should provide additional (meta-) information. Within our domain, *simple* sewage treatment plants cannot have *chemical* purifying stages, but they may have at least a *biological* one. NAUDA provides this information with the answer:

Purifying stages of simple sewage treatment plants are by definition biological.

3.1.4 DIALOG ASPECTS

Up to now we have discussed phenomena which are related to a single question. Usually the users' questions are related. Two important kinds of relation are reference (e.g. with pronouns) and ellipsis, for instance, a follow-up question which is shortened to a phrase which only expresses the difference from the previous question. These phenomena, which may significantly shorten a dialog sequence, are handled by most of today's natural language interfaces (cf. [Küpper et al. 90]).

Furthermore, considering a question within the dialog's context may lead to a disambiguation or to an interpretation which cannot be justified without regarding the context. For instance, the question

Which sewage treatment plants have 3 purifying stages?

can be interpreted in two different ways: the user may want to know which *types* of sewage treatment plants have 3 purifying stages, or he may want to see the corresponding individual database objects. If a presupposition failure with respect to similar objects is detected in the previous question, NAUDA presents the *types* of sewage treatment plants. Without a context like that the question is answered by querying the database.

3.2 ARCHITECTURE OF THE NAUDA SYSTEM

The NAUDA System is a combination of components which were

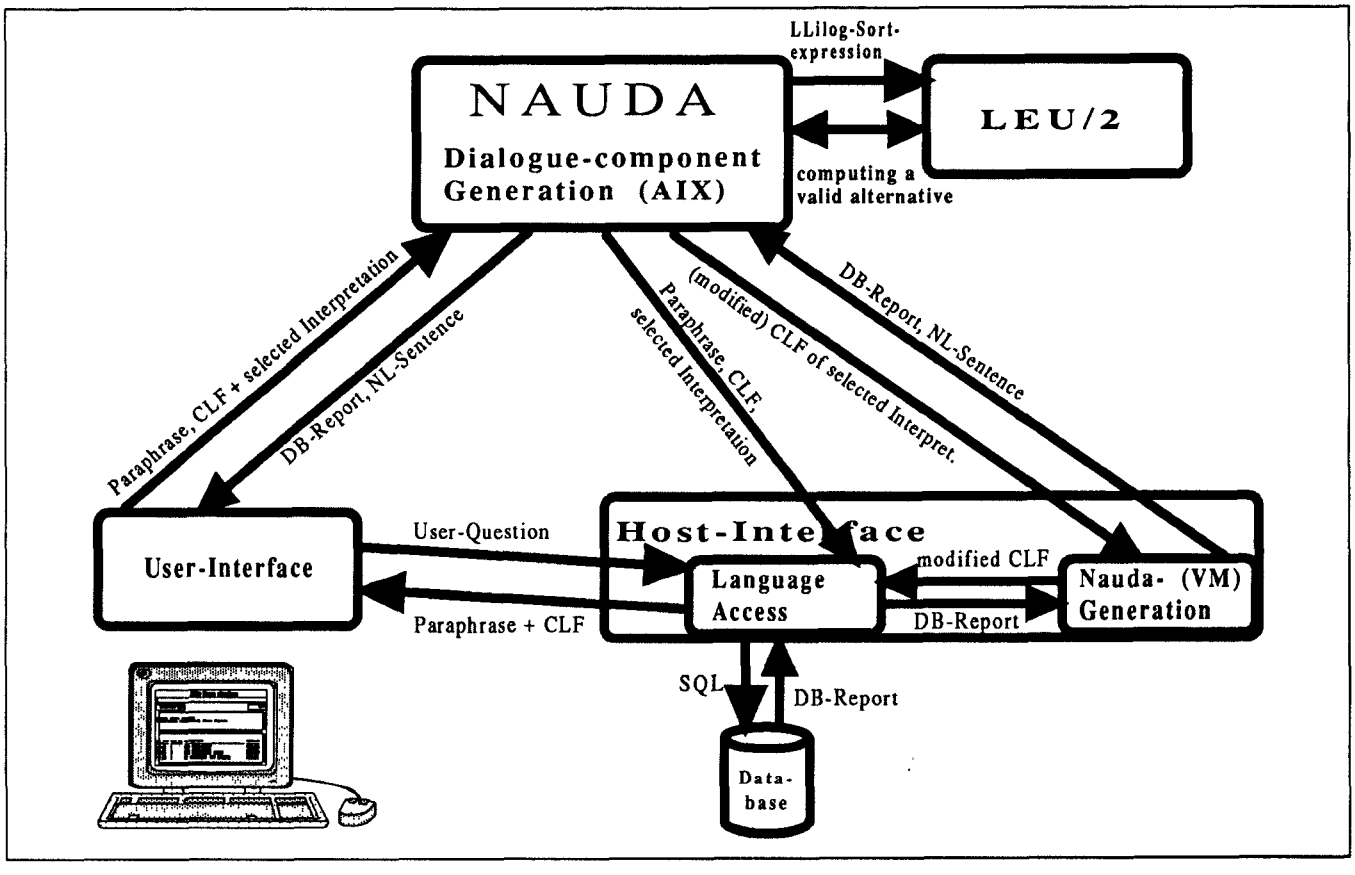

Figure 2: Architecture of the NAUDA System

developed by the NAUDA project team itself, and of two components from other research groups, namely LanguageAccess and LEU/2 (both IBM).

LanguageAccess is responsible for analyzing a user query. This includes solving references to the previous discourse. LanguageAccess makes use of a *conceptual schema* which contains information about the domain and the specific vocabulary. As a first result of the analysis, a paraphrase is presented to the user for verification (in the case of ambiguity, several paraphrases are presented for selection).

NAUDA's dialog component is responsible for handling false presuppositions on the conceptual level. It uses the inference engine of LEU/2 to check if the requested objects are legal within the domain. Due to different representations LEU/2 uses a modified version of LanguageAccess' conceptual schema as a model of the domain. If an inconsistent object description is detected, NAUDA computes a related description, which fits into the taxonomy of the domain. The generation component uses the difference between these two descriptions for constructing a correction in natural language which is presented to the user. It indicates the type of error, and contains meta-information which is related to the inconsistent object description (for details see [Becker et al. 1992]).

If no presupposition failure is detected but the question is over-answered (cf. section above), LanguageAccess is advised to generate an SQL-expression for a wh-question which corresponds to the original yes-no question of the user. Depending on the result of the database query, the generation component decides wheather to answer "Yes", or to provide a sentence with the actual value of the property in question.

If none of the above cases hold, LanguageAccess will query the database with an SQL expression which is constructed from the logical representation of the user's query.

Finally, the generation module decides on the form of the presentation of the database results. In addition, it must adapt the form of expression to the preferences of the user, who has the possibility of choosing between two different levels of granularity (that is, short or more precise).

Most of the components of our current prototype are implemented in PROLOG. LanguageAccess and the SQL-DS database itself are components which run on an IBM Mainframe (IBM9370) under VM while the other components run on a PS/2 under AIX. The communication is realized by three interface processes. The first one provides communication with LanguageAccess, the second one with the NAUDA components, and the third one with a user interface which has been developed on the basis of X-Windows and OSF-Motif.

4. FUTURE RESEARCH

4.1 INTEGRATION OF GRAPHICS

The visualization of spatial relationships by means of geographic maps can be seen as a starting point for further work. Currently tables, natural language, and graphics are used as media for the presentation of answers. Using an adequate output medium is a definite improvement in terms of human perception and information processing. Yet, the linking of the different channels remains to be worked out, both on the representational side and on the display side.

In order to achieve coherence within the output, referent specification should be done in a multimodal way (e.g. connecting language and graphics by deictic expressions like *this* and *here* which are used to point to things in the nonlinguistic context like visually marked objects shown on the map).

In addition to these issues, the interplay between language and imagery for thinking about spatial relations remains to be investigated.

4.2 META-INFORMATION

In the current state of the system, only meta-knowledge resulting from false presuppositions is made explicit. But there are several other kinds of meta-information which could be helpful if presented to the user, e.g. information about the structure of the database and about definitions of concepts.

Especially when dealing with highly complex databases, users may have problems with the types and localizations of the information available. A highly cooperative system could support users in becoming familiar with the contents and structure of the database if it were able to answer questions such as *'Where can I find information about XY?'*.

For providing meta-information the system needs to have access to an explicit representation of the meta-knowledge, i.e. the knowledge about concepts and relationships in the domain and their database representations. Only a small part of this knowledge is contained in the database catalog and can be exploited automatically. It should be investigated how data dictionaries can be used to make more information *about* the database available; moreover what kind of knowledge is required exactly for supporting users which are unfamiliar with the structure of the database. This meta-knowledge would be useful in general and is not restricted to natural language access. The importance will increase with the possibility of accessing distributed databases via data connection.

4.3 SUPPORTING PROBLEM SOLVING

Information systems are usually only one of several sources of knowledge which is used to solve an overall task. For complex databases, users cannot be expected to know which contents of the database can support the solving of their task(s). Because data cannot be exploited unless its existence and relevance for a specific task is known, the benefit of a database may increase if the system helps the user to find relevant data. In this case we expect that a user will start with a (probably incomplete) description of her/his task, rather than asking for concrete data. The system should be capable of processing such descriptions.

Furthermore, it must deduce, verify and update its assumptions about the user's goals in order to provide data which are relevant for her/his specific needs. For this purpose a dynamic model of the user's goals is necessary. Additionally, the system's knowledge of the structure of the domain and database must be enriched especially with respect to the purpose of the data.

5. SUMMARY

We have outlined our approach to increasing the cooperative behavior of a natural language interface to a 'real-life' database (i.e. with a broad spectrum of modelled objects and relations). The system can handle some presupposition failures within the user's

query. Corrections, over-answering results and the contents of single values are given in natural language. Output modalities have been extended so that answers containing certain spatial aspects are enriched with topographical maps. To date, a first prototype has been implemented.

ACKNOWLEDGEMENTS

We would like to thank Harald Bayer for his work on the communication processes, Michael Horn for the implementation of the user interface, and Manfred Reichert for his help resulting in many system inprovements (e.g. integration of topographical maps, adaption of a speech recognizer).

REFERENCES

[Becker et al. 1992] R. Becker, D. Küpper, M. Strobel, D. Rösner: A cooperative, natural language environmental information system. In *Information Processing 92 - Proceedings of the IFIP 12th World Computer Congress* (Madrid, Spain, 7-11 September 1992), Vol. II, R.Aiken, ed. Elsevier, North-Holland, 1992, 655-665.

[Joshi 83] Aravind Joshi: Varieties of cooperative responses in question-answer-systems. In *Question and Answers*, F. Kiefer, ed. Reidel Publishing Company, Dordrecht, 1983 (=Linguistic Calculation 1) S. 229-240.

[Küpper et al. 92] D. Küpper, D. Rösner, A. Striegl-Scherer: Natürlichsprachliche Zugangssysteme mit Deutsch als Interaktionssprache - eine vergleichende Studie -. In *Sprache und Datenverarbeitung*, 1992.

[Lehmann et al. 78] H. Lehmann, N. Ott, M. Zoeppritz: User experiments with natural language for data base access. In *Proceedings 7th International Conference of Computational Linguistics*, Bergen, 1978.

[Wahlster et al. 83] W. Wahlster, H. Marburger, A. Jameson, S. Busemann: Over-answering YES-NO questions: extended responses in a NL interface to a vision system. In *Proceedings of the 8th IJCAI*, Karlsruhe, FRG, 1983, 643-646.