# AN INTEGRATED SYSTEM FOR TEACHING SPOKEN DIALOGUE SYSTEMS TECHNOLOGY

*Kåre Sjölander and Joakim Gustafson*

Department of Speech, Music and Hearing, KTH
Box 70014, S-10044 Stockholm, Sweden
Tel.:+46 8 790 7879 Fax: +46 8 790 7854 E-mail: {kare | joakim_g}@speech.kth.se

## ABSTRACT

Spoken language systems are highly complex and teaching of students in this subject matter and in the underlying technologies could benefit greatly from instructional software. The aim of this work has been to give students hands-on experience via a fully functioning spoken dialogue system as a teaching aid. This dialogue system was built using our toolkit for spoken language technology as a dedicated laboratory environment for students. The system was used in a lab which was part of two courses on spoken language technology given by our department. Students were given some initial guidance on how to modify and extend the system but most of their work was unsupervised. Overall, the laboratory system was a success and we plan to improve and extend it for the coming academic year. Thanks to the rapid prototyping and development character of our toolkit we might even use it and the modules from the system as a software basis for student projects in spoken language technology.

## 1. INTRODUCTION

Traditionally, spoken dialogue systems have required high expertise to design and develop. The resulting applications have been large and complicated and not always easy to modify and extend or even comprehend. Thus, teaching in the subject of spoken dialogue systems and related technologies has mostly been done in lecture format with video taped demonstrations of actual systems. Live demonstrations are typically conducted by a well-behaved PhD student who is well acquainted with the given system and knows which questions to ask. Students have mostly been kept at a safe distance.

The aim of this work has been to put a fully functioning spoken dialogue system into the hands of our students as an instructional aid. They are given opportunity to test it themselves and to examine the system in detail. They are also given guidance on how to extend and develop it. In this way, we hope to increase their understanding of the problems and issues involved and to spur their interest for this technology and its possibilities.

## 2. THE TMH TOOLKIT

We have recently developed a toolkit to alleviate the arcane art of constructing spoken language systems. This toolkit is based on the software technology in our existing spoken dialogue system WAXHOLM [1]. We have extracted and redesigned different components such as speech recognition [2], speech synthesis [3], visual speech synthesis [4], and parser [5] and created Tcl language [6] modules from these. This has enabled us to take advantage of the rapid prototyping and development framework, which this language fosters and to create a toolkit for spoken language technology in the spirit of the ones created at OGI (CSLUsh) [7] and MIT (Sapphire) [8]. Our toolkit has empowered us to create new applications quickly and easily based on its modules using Tcl as a glue language and also to use the accompanying Tk-widget set for graphical user interfaces. Tcl is a rather simplistic language, with many shortcomings, but we use it only for system integration and user interfaces. The modules themselves are exclusively written in C. Module interfaces are string based which makes coding, testing, and debugging simple, but these interfaces could be limiting for future applications. We are actively investigating other alternatives, for example, CORBA.

## 3. THE LABORATION ENVIRONMENT

Currently, this toolkit has been used to create an integrated lab environment for the courses on spoken language technology given at the MSc level at the Royal Institute of Technology (KTH) and at Linköping University in Sweden. In this environment, students are presented with a simple spoken dialogue application for doing yellow pages search on selected topics using speech input (Swedish language) via microphone. The system has knowledge about streets, restaurants, hotels, museums and similar services. Results are presented

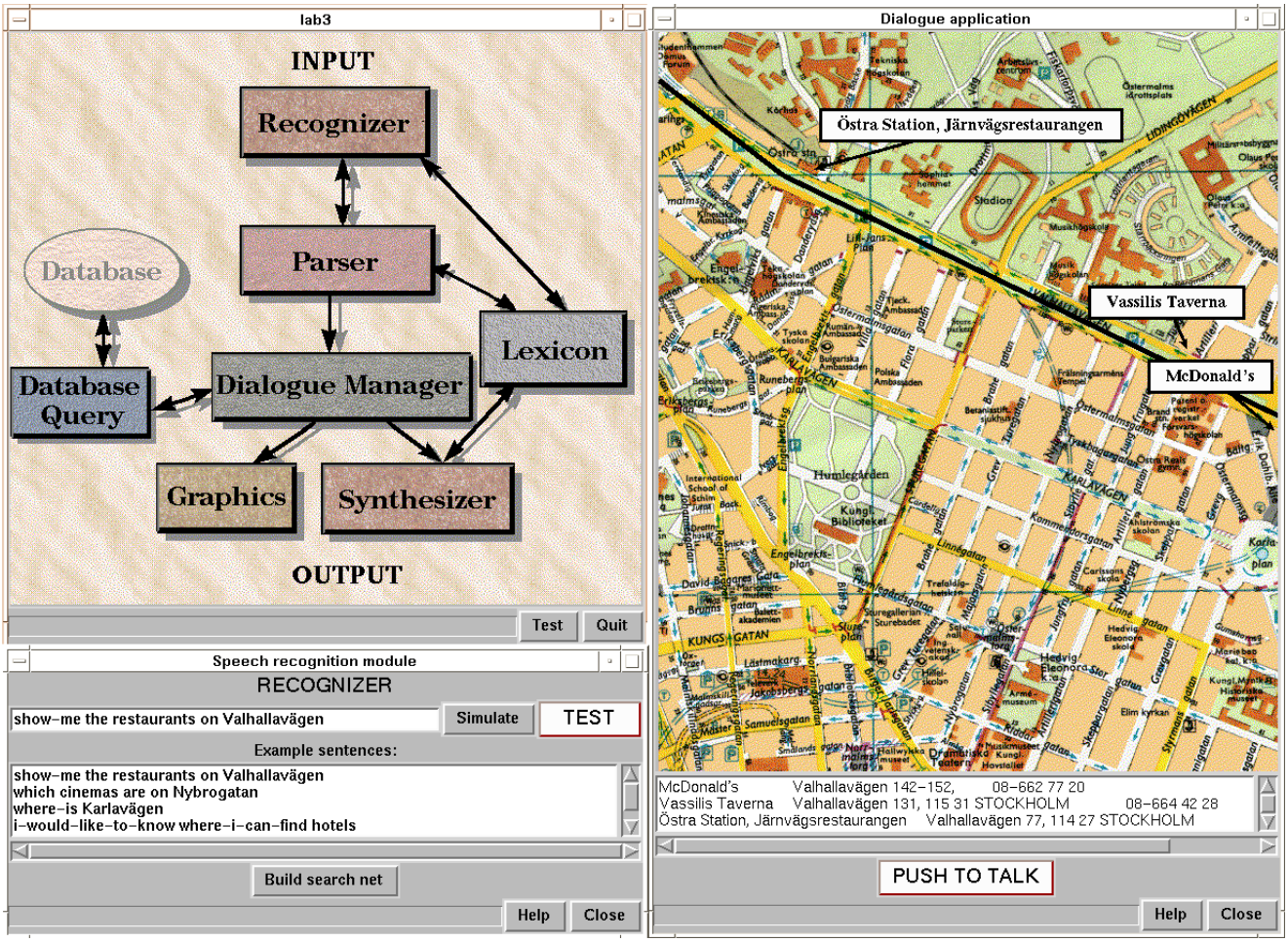


**Figure 1. A screenshot showing the control window (upper left), the dialogue application (right), and the speech recognition module (bottom left).**

using combinations of speech synthesis, an interactive map or Netscape Navigator. This application is accompanied by a development environment which enables the students to interactively study and modify the innards of the system even while it runs. Each module has its own control window, which dynamically updates to reflect the processing as it takes place. It is possible to add new words or phrases to the lexicon and then be able to use them a few seconds later in the speech recognizer or text generator and speech synthesizer. Students can modify the results of the recognizer and parser to test how the system is affected by different inputs and errors. Thus, it is possible to control the rest system from each module in the chain. We have chosen to keep the modules relatively simple in this initial version.

## 4. SYSTEM MODULES

### 4.1. Control Window

This window shows an outline of the components of the system and how they interact and depend. For each box the corresponding module window can be opened with a mouse click. Also, the complete dialogue application is launched from this window. There is no explicit building step involved, as all changes to the system are made incrementally. When the system runs, the boxes highlight as processing in the respective modules takes place.

### 4.2. Speech Recognition Module

The continuous speech recognizer uses phone models trained on spontaneous speech data collected for the WAXHOLM application. As that domain (boat traffic information) differs from the current one, we use a simple class pair grammar that is based on example sentences given to the lab system. This set can easily be extended by the students to incorporate new ways of formulating questions to the system. In this module, it is possible to modify and extend the grammar and also to test the recognizer stand alone, without running the complete system. Recognition output can also be edited and sent back into the system to simulate its operation.

### 4.3. Parser Module

Parsing is either done by a statistical parser or by a simple keyword spotter. The simple grammar used by the recognizer often produces results that are hard to

parse correctly, but performance using keyword spotting is still quite useful. Keywords are tagged semantically, which is used later for the database search. Results from the parser can be modified and re-sent into the system for subsequent processing.

### 4.4. Dialogue Manager Module

This is only a simple control loop which activates the different modules in turn and passes information between them. Currently there is no way for the students to influence the dialogue management module, but this is the main focus for work in progress, see section 6.

### 4.5. Database Module

The database has been extracted from a Stockholm Yellow Pages web service. This is provided 'as is' and there is no way for the students to modify this currently. The database has also been augmented with web links to information about most of the facilities.

### 4.6. Database Query Module

This module handles database queries. It translates semantic knowledge extracted by the parser module and translates this to appropriate query strings. A list with database hits is returned. It is possible to search the database by manually entering query strings.

### 4.7. Lexicon Module

The system has a lexicon module that stores transcriptions and syntactical and semantical information for the task specific vocabulary. It can also suggest rule based transcriptions or transcriptions from a larger general lexicon for new words entered into the lexicon. Speech synthesis searches this task specific lexicon in the first place and falls back to use the larger lexicon or even rules when needed. The recognizer uses only the task specific lexicon for performance reasons. It is possible to check transcriptions by listening to the speech synthesis reading them directly in this module.

### 4.8. Graphics Module

The graphics module displays a map with a graphical presentation of the results from the database query. Streets are highlighted and facilities marked in the map, as shown in Figure 1.

### 4.9. Speech Synthesis Module

Speech Synthesis has been combined with text generation into a single module. The result from the database query and parser analysis is used to select a response template to fill in. There are multiple templates for each possible response type. This allows the system to choose one at random resulting in more varied system responses.

### 4.10. Spoken Dialogue Application

This is the sample dialogue application built using the previously mentioned modules. Spoken queries are input using push-to-talk and results are presented with speech synthesis and graphics.

## 5. THE LABORATION ASSIGNMENT

The lab environment was used for two different courses. At KTH it was used in the course Advanced Graphics and Interaction, in the section on multi- modality. This was followed by about 45 last-year MSc computer science students. At Linköping University the lab was part of a course on Speech Technology taken by 12 students from the MSc computer science and computational linguistics programmes. The students worked in groups of two and were given a list of modifications to apply to the system.

- To start with, they had to use the dialogue application in order to determine its capabilities and limitations.
- They were told to test the speech recognition module stand alone, with the explicit purpose that they should gain some insight into the limitations of current HMM based speech recognition technology. For example, regarding noise, speaking style and out of vocabulary words.
- They were given a number of street names and facility terms to add to the lexicon. Transcriptions, as well as syntactic and semantic tags had to be included.
- Also, they had to extend the example based grammar with new constructs.
- In the text generation module, they had to insert additional response templates to handle the new facilities.
- Finally, the students had to demonstrate that the extended system worked according to specification.

Overall, the students were very satisfied with the system and they rated it at 4 points out of 5 in the course evaluation. The main criticism was that they wanted to be able to make greater changes to the system and to go deeper into some of its modules.

## 6. FUTURE WORK

Our main focus concerning the continued development of the dialogue environment is to integrate a real dialogue manager into the system. We are planning to launch a joint research project together with the Natural Language Processing Laboratory, NLPLAB, at the University of Linköping, which aims at integrating the highly flexible dialogue manager [9] developed at that site with our system. In this model, a dialogue grammar based on speech act information is used together with a dialogue tree which handles focus structure.

Some preliminary work has been done to make it possible for the students to penetrate deeper into the workings of the recognition and synthesis modules. We have, for example, experimented with visualization of word graph re-scoring and direct editing of low-level

synthesis parameters. It would be possible to use the environment for dedicated labs in these areas.

A more active World Wide Web integration might also be possible. Currently, some information is only presented by fetching web pages and displaying them. There are no possibilities for spoken interaction with the contents. Our ambition is that users should never have to resort to using the mouse in any situation.

In the current course, the emphasis was on giving the students an understanding of how the technology works and not on letting them build actual new systems themselves. The latter has proven successful in a dedicated course at OGI (Colton, Cole, Novick, Sutton, 1996). However, we decided that this was a too big step initially due to available time and resources. Student projects, with focus on system building, could be a natural and very interesting development for our courses.

Some work has been made to port the lab to the English language, mostly for demonstration purposes. The main problem for the current system and the yellow pages domain is the pronunciation of Swedish street names in English.

Another likely development will be the porting of the system from the current HP Unix platform to Linux.

## 7. CONCLUSION

This preliminary and rather limited lab environment was a success with the students. It was the result of about two man months of work, with programming, writing laboration instructions, and testing. For us it was a big step forward, putting spoken language technology into the hands