# DS-UCAT: A New Multimodal Dialogue System for An Academic Application

*Ramón López-Cózar*[1], *Zoraida Callejas*[1], *Germán Montoro*[2]

[1] Dept. of Languages and Computer Systems, University of Granada, Spain
[2] Dept. of Computer Science and Engineering, University Autónoma of Madrid, Spain
E-mail: {rlopezc, zoraida}@ugr.es, German.Montoro@uam.es

## Abstract

This paper presents our latest research efforts in the field of multimodal dialogue systems, concerned with setting up a system to assist professors and students in several of their daily activities within an academic centre (e.g. a University's Faculty). A distinct feature of this system is that in addition to interact with the user it does so with the environment where the user is at the moment, as an attempt to enhance its performance. The paper presents the system set-up and architecture, and comments on the usage of the current implementation.

## 1. Introduction

In the last years, new technologies have appeared to facilitate the user interaction with computers and the surrounding environment. Two of them, addressed in this paper, are called "Dialogue systems" and "Ambient intelligence". Dialogue systems are computer programs designed to emulate the behaviour and communication abilities of a human being interacting with another to carry out a particular task. These systems are nowadays employed to provide information about airplane or train schedules [1], call routing [2] or academic information [3], among others. In spoken dialogue systems the user-system interaction is carried out using speech as the only communication means [4]. These systems are typically used to provide automatic telephone-based services.

In multimodal dialogue systems the human-computer interaction relies on several kinds of communication channels, as for example speech, text, graphics, hand and body gestures, gazes or lip movements. The information generated by the user is captured by a variety of devices such as microphones, keyboards, mice, cameras for artificial vision or touch-sensitive screens. Several systems even include bio-sensors to capture biological signals from the user (e.g. heart beat, blood pressure or body temperature). Taking into account the diversity of available devices, there are systems that propose the user to interact using the more convenient modalities given the environmental conditions, e.g. in terms of acoustic noise or lightning. For example, users can be suggested to type in data via keyboard instead of uttering words in noisy environments. Moreover, this feature allows handicapped users to interact with the systems employing the interaction modalities that better adapt to their needs (e.g. blind users can use speech while mute people can use the keyboard). Therefore, multimodal dialogue systems can make the interaction be more human-like, reducing system errors and enhancing the user experience [5].

The aim of ambient intelligence-based systems is to provide a natural interaction between the environment and its inhabitants. This way, classrooms, offices, laboratories and homes can help people in their daily activities, offering non-intrusive ways of communication. The interaction in these settings is typically adaptive to the tasks, environment, users and available devices, which imply using different communication modalities depending on the user necessities [6].

Several projects have tried to combine the advantages of dialogue systems and ambient intelligence to facilitate the user interaction with the surrounding environment. For example, Aire [7] studied possibilities for combining sketching with speech in a multimodal design, while Homey [8] aimed at assisting patients in a tele-medicine application. The Smartkom project [9] uses as input automatic speech recognition and gestures, and generates text, graphics and speech as output, allowing that users can employ any of these modalities in three different settings: home/office, communication kiosk, and mobile.

## 2. The DS-UCAT system

Our work within the UCAT (Ubiquitous Collaborative Training) project[1] is concerned with setting up a multimodal dialogue system, termed DS-UCAT (Dialogue System for UCAT), to assist professors and students is several of their daily activities within an academic centre (e.g. a University's Faculty).

We assume the system will work in three different places of an academic centre: Library, Professors' Offices and Classrooms. In the current implementation, the system allows the user interaction via sound, speech, graphics and text. The multimodal input allows combining several modalities in one interaction. For example, a student can ask for information about available books on a particular subject by either speaking the subject, selecting it on the computer screen using the mouse, or writing the subject in a form field. Since the system output is also multimodal, a spoken message generated for this input indicates the requested information is available on the screen, where it appears as a list of books in text format.

We plan to set up a system's function to provide users with messages generated by the system's initiative as they move within the academic centre. For example, when a student passes by near the centre's library, the system will remind him of the borrowed books he must return soon to the library.

Fig. 1 shows the architecture of the system, which is comprised of an X+V document server connected with the users' mobile devices (Tablet PCs, laptop computers and PDAs) by means of wireless connections. In the current implementation we are only using laptop computers, which connect to the server employing the wireless network of our lab.

[1] http://orestes.ii.uam.es/ucat/

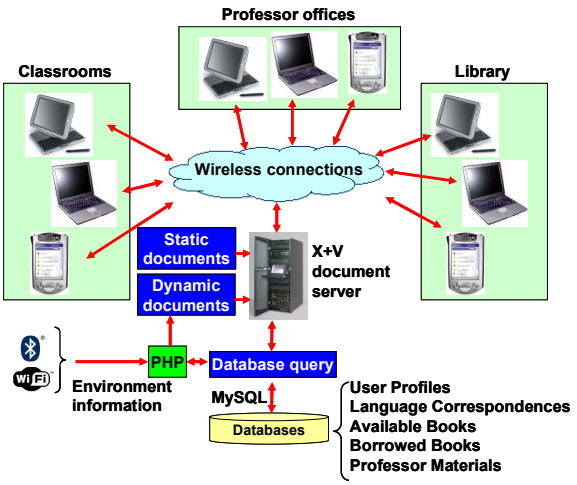


Figure 1: *The DS-UCAT system.*

### 2.1. XHTML+Voice documents

The system is being set-up using a toolkit[2] for building multimodal applications based on the W3C's XHTML+Voice language[3] (X+V). It is configured as a set of X+V documents, some of them are stored in the document server, while others are dynamically created using PHP programs that take into account the user features and preferences (e.g. gender and preferred interaction language), as well as the data extracted from the databases. X+V documents are comprised of forms with fields that are filled in with the user input via speech, text or mouse clicks. To visualise these documents, in our current setting users run in their communication devices the Opera browser[4], which supports multimodal interaction (voice, text and graphics). This browser is automatically installed and configured when the multimodal toolkit is installed. In addition to recognise spoken sentences concerned with our academic application, this browser allows the user to control the interface by means of spoken commands (e.g. "Opera reload", "Opera stop", "Opera close all", etc.). X+V handles the system-user interaction using speech-based and visual interfaces.

#### *2.1.1 Speech-based interface*

Automatic speech recognition is carried out by the Opera browser's built-in recogniser. In our setting the recognition is based on a tap-&-talk method, i.e. the user must click and hold a microphone icon or key while he speaks to the system (although it can be configured differently in the browser). To allow the spoken interaction the user must enable the voice feature of the browser, which automatically installs the necessary packages from the Internet, if necessary. Speech recognition and understanding is carried out using JSGF grammars (Java Speech Grammar Format) that are used either at form or field level. Some of these grammars are static, while others are dynamically created by means of PHP programs that query databases and include the obtained data as grammar vocabulary (e.g. book titles). For example, using the grammar to recognise book queries, if the user utters the sentence "I need books about Maths please" the system fills in the form field "subject" with the word "Maths". The recognition grammars used to handle book queries must be updated as the library catalogue changes, so that they are compiled dynamically using the contents of the Available Books database. To update these grammars we have implemented a PHP program that carries out two tasks. Firstly, it queries the databases using MySQL functions and obtains the data from the available books, such as the titles, authors or subjects. Secondly, it creates the grammars to recognise complete sentences as well as isolated data items (e.g. title, authors or subjects) using the information gathered in the first step.

In the system output, speech synthesis is carried out by means of the sentences included into the <prompt> … </prompt> labels typically used in VoiceXML[5]. These sentences are transformed into voice by a text-to-speech process using the Opera browser's built-in speech synthesiser. Some of these sentences are fixed, while others are created at run-time considering the user type (professor or student), the user gender (necessary to create several sentences appropriately) and the data extracted from the databases.

#### *2.1.2 Visual interface*

In the system input, the visual interaction is used to obtain data from the user via form fields and selection buttons typically used in XHTML (see Fig. 3). In the system output, the visual interaction is used to provide the data extracted from the databases (e.g. list of available books) and information about the current user's name and type (see Fig. 2).

#### *2.1.3. Connection of both interfaces*

The connection between both interfaces is carried out using event handlers, which are placed at the body section of the X+V documents. We use several types of event handlers available in X+V. For example, when the document used to enter book queries is loaded into the browser, the event onload is thrown and, in response, a VoiceXML form called initial_vform is executed to handle this event. XHTML+Voice allows that a user utterance to fill in several form fields in one interaction (mixed-initiative interaction strategy). To do so, we use a <vxml:initial name="initial_vform"> …</initial> section, typically employed in VoiceXML, which allows recognising the user utterance using a form level grammar. Thus, for the book query document the system generates the message "Please enter a book query" and the user can utter a variable number of data items (e.g. authors; authors and publication year; authors, publication year and subjects; etc.). We also use the ev:event="onclick" event, which is thrown when the user clicks on a form field. The handler for this event is VoiceXML code to obtain the value for that particular form field.

[2] http://www-306.ibm.com/software/pervasive/multimodal/
[3] http://www.w3.org/TR/xhtml+voice/
[4] http://www.opoera.com
[5] http://www.w3.org/TR/voicexml20/

### 2.2. Databases

To provide information to the users and interact with them properly, the system queries several databases. The User Profiles database contains personal data of the users such as name, gender, address and telephone number. It also stores four types of personal preferences: i) interaction language (English or Spanish at the moment), ii) oral interaction (enabled/disabled), iii) system voice type (male or female), and iv) acceptation of incoming messages from the work environment (enabled/disabled).

The Language Correspondences database stores expressions in several languages corresponding to particular sentence types which are used depending on the selected interaction language. An example of these expressions is the welcome message generated by the system as the user logs in: either "Welcome to the DS-UCAT system" for interaction in English, or "Bienvenido al sistema DS-UCAT" for interaction in Spanish.

Additionally, in our current configuration the system uses two other databases for experimental purposes, which in a real-world implementation of the system should be replaced by the real ones. On the one hand, the Available Books database stores information of books supposedly available in the academic centre's library, while the Borrowed Books database stores data about books borrowed by the system's users (professors and students). Using these databases the system can answer queries as that shown in Fig. 3 for an interaction in English. Using the form in this Figure, when the user clicks on a field, he receives a spoken message asking for the data to be entered (e.g. "Book title?"), which can be provided either orally or in text format. After the Available Books database is queried, the system informs visually (using a table) about the available books and generates the spoken message "The following books are available". Alternatively, it generates the spoken message "No books were found" if no records were retrieved from the database.

The Professor Materials is a database to be created in order to store information about the class materials made available by the professors. Our plan is that the X+V documents for the Classroom and Professors' Office work environments (also to be created) will show a view to query this database, similarly as the X+V document shown in Fig. 3 used to query the Available Books database.

## 3. System usage

To interact with the system the user must firstly log in. Using the login the system determines the user type (professor or student) by querying the User Profiles database. Finding out the user type allows adapting the interaction adequately in terms of interaction language, oral interaction, system voice and acceptation of incoming messages from the environment. For instance, Fig. 2 shows the welcome window shown on screen for a user who selected English for the interaction language. As this user selected to use oral interaction in his profile, the spoken message "Hello Ramon, welcome to the DS-UCAT system" is generated.

As can be seen in this Figure, after the user has logged in he must choose a work environment (either Library, Classroom or Professor Office). This initial selection allows him to interact in an environment which is not the one in which he is at the moment, thus enabling for instance to make book queries from a classroom. Note that the place where the user is at the moment (Library in the example) is selected by default as explained in Sect. 3.1.

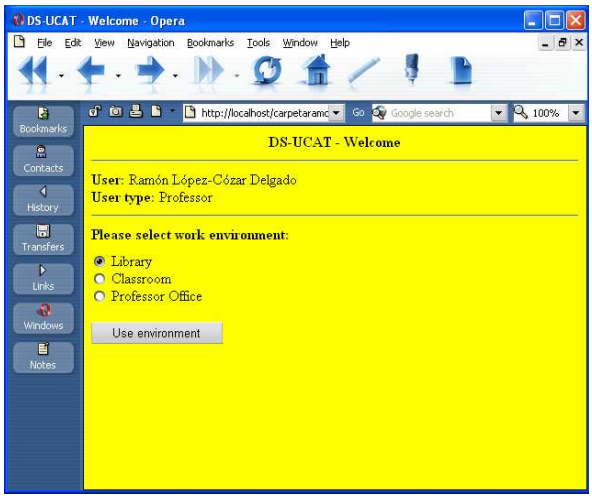


Figure 2: *Welcome window for interaction in English*.

Fig. 3 shows the X+V document shown on the screen when the user starts to use the Library work environment. Since this user enabled the oral interaction in his profile, the spoken message "Please enter a book query" is generated. To enter a spoken book query, he must click and hold the tap-&-talk key, or click on the browser's microphone icon while he speaks to the system. Alternatively, he can enter the required data using the keyboard and the mouse.

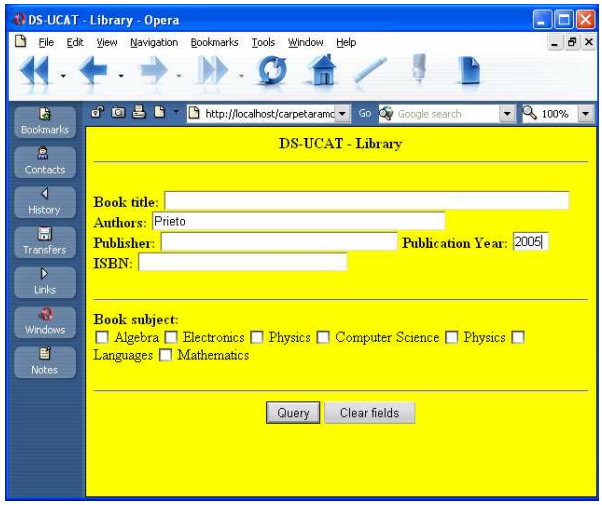


Figure 3: *Book query for interaction in English*.

The system implements the two interaction strategies most commonly employed in dialogue systems: mixed and system-directed. The use of one or another depends on the X+V document the user is interacting with. For example, if he interacts with the one designed to carry out book queries, the interaction is mixed in order to allow filling in several form fields in just one interaction (e.g. uttering the sentence "I'd like to get information of books written by Allen concerned with

Computer Science"). If the sentence cannot be recognised, the system prompts for the necessary data to query the database field by field (system-directed strategy). The system takes into account the typical events that can occur in a spoken interaction, i.e. the user asks for help, there is no input from the user, and the input cannot be recognised. These event types, usually called help, noinput and nomatch, respectively, are handled by the VoiceXML's catch element.

### 3.1. System-environment interaction

Our goal is that the user-system interaction can be carried out in such a way that the location in which the user is interacting at every moment (e.g. Library) can be taken into account without the user being concerned. However, as the user localisation within the academic centre is not yet implemented, at the time of writing we simulate this information by fixing manually a variable that represents the current user location, using the values "Library", "Classroom" or "Professor Office". This variable's value is taken into account when several X+V documents are dynamically created. For example, in Fig. 2 the work environment selected by default is Library because the user is supposed to be located in this environment. To simulate this localisation, we simply set the variable's value to "Library" before the X+V document is dynamically generated.

In order to set the value for this variable automatically, we plan to implement a procedure to detect the location change as the user moves within the academic centre. The goal is that whenever the location changes (e.g. a student leaves the classroom and enters into the library), the browser shows a small window to suggest the use of the work environment that best fit the new location. For example, this window will suggest to use the Library work environment if the user enters into the library.

## 4. Conclusions and Future Work

This paper has presented our latest work within the UCAT project concerned with setting up a multimodal dialogue system to assist professors and students in several of their daily activities within the Library, Professors' Offices and Classrooms of an academic centre (e.g. a University's Faculty). The paper has focused on the system implementation based on XHTML+Voice documents, some of which are stored in a document server while others are dynamically generated using PHP programs. The server interacts with the users' mobile communication devices (laptops at the moment) by means of wireless connections. The paper has also described how the system can be used to enter book queries. Future work includes the following lines:

- Create the X+V documents and databases necessary to interact within the Classroom and Professor Office work environments.
- Set-up a procedure to automatically localise the user within the academic centre. To implement this feature we plan to experiment with Bluetooth emitters, WI-FI access points and RFID (Radio Frequency ID).
- Enable a system's function to provide the users with incoming messages generated from the environment.
- Implement the system ability to operate devices of the academic centre in order to change their status. This will allow that in a professor office the system can turn on/off lights or ambient music as the professor enters/leaves the office. To set up this feature we plan to adapt a middleware we previously created for the Interact project [10], which was used to set up a dialogue interface for a home environment.

## 5. Acknowledgements

This work has been funded by the Spanish Ministry of Science and Technology, under project TIN2004-03140 Ubiquitous Collaborative Adaptive Training.

## 6. References

[1] T. Torres, E. Sanchís, E. Segarra, E. Development of a stochastic dialog manager driven by semantics. Proc. Eurospeech, pp. 605-608, 2003

[2] C.-H. Lee, B. Carpenter, W. Chou, J. Chu-Carroll, W. Reichl, A. Saad, Q. Zhou. On Natural Language Call Routing. *Speech Communication*, 31, pp. 309-320, 2000

[3] Z. Callejas, R. López-Cózar. Implementing Modular Dialogue Systems: A Case Of Study. ISCA Tutorial and Research Workshop on Applied Language Interaction in Distributed Environments, 2005

[4] M. F. McTear. Spoken dialogue technology. Toward the conversational user interface. Springer, 2004

[5] R. López-Cózar, M. Araki. Spoken, Multilingual And Multimodal Dialogue Systems: Development And Assessment. John Wiley & Sons Publishers, 2005

[6] M. C. Mozer. Lessons From An Adaptive House. In D. Cook & R. Das (Eds.), Smart Environments: Technologies, Protocols, And Applications, Pp. 273-294. Hoboken, NJ. John Wiley & Sons, 2005

[7] A. Adler, R. Davis. Speech And Sketching For Multimodal Design. Proceedings Of The 9th International Conference On Intelligent User Interfaces, Pp.214-216, 2004

[8] D. Milward, M. A. Beveridge. Ontologies And The Structure Of Dialogue. Proceedings Of CATALOG, 8th Workshop On The Semantics And Pragmatics Of Dialogue, 19th – 21st July, Barcelona, Spain, 2004

[9] W. Wahlster (Ed.). SMARTKOM: Foundations Of Multimodal Dialogue Systems, Cognitive Technologies Series. Springer, 2006

[10] G. Montoro, P. A. Haya, X. Alamán. Context Adaptive Interaction With An Automatically Created Spoken Interface For Intelligent Environments. International Conference On Intelligence In Communication Systems, Bangkok, Thailand. November, 2004