From HTML to VoiceXML: A First Approach

César González Ferreras, David Escudero Mancebo, and Valentín Cardeñoso Payo

Departamento de Informática, Universidad de Valladolid, 47011 Valladolid, Spain
E-mail: cesargf@infor.uva.es, descuder@infor.uva.es, valen@infor.uva.es

Abstract. In this work, we discuss the construction process of the voice portal counterpart of a departmental web site. VoiceXML has been used as the dialogue modelling language. A prototypical system has been built using our own VoiceXML interpreter, which easily integrates different implementation platforms. A general discussion of VoiceXML advantages and disadvantages is reported and a simple startup procedure is proposed as a means to build voice portals starting from legacy web sites.

1 Introduction

Internet is our biggest information repository nowadays and web browsing the natural access mechanism to it. However, network browsing is not the most used way to access information in everyday life. Mobile devices cope the market and voice is the prominent communication channel in this environment.

Nevertheless, not enough attention has been paid yet to methodological approaches to spoken dialogue systems design and scarcely any reference can be found to ways of exploiting all the legacy information systems that companies are developing and maintaining using generic web technology.

Since spoken dialogue based interaction is essentially affected by sequential mode of operation, complemented in best cases with some form of asynchronous reaction from the user side, the careful design of spoken dialogue systems starting from their counterparts in web environments is still an open question.

In this work, we present our first efforts towards the definition of a integration methodology of web contents navigation and spoken driven information accessing. To this end, we present a first case of study in which a simple web site has been analyzed and rebuilt as a voice portal, using a VoiceXML platform developed at our site.

2 Description of the Problem

The goal of our application is to access information through the telephone line: a voice portal version of the Computer Science Department web site at *Universidad de Valladolid*.

The system started from HTML material, from the web pages of the Department, where you can find general information, information of the teaching activities, a description of the research teams and members contact information.

All this information has to be provided through a voice interface, so we have to be aware of the characteristics of the voice compared to GUI: voice interaction must be sequential, while visual interaction can show all the information at once [3]. This implies that an explicit navigation model is needed, and a technique or language for modelling applications.

Petr Sojka, Ivan Kopeček and Karel Pala (Eds.): TSD 2002, LNAI 2448, pp. 441–444, 2002.

3 Solution

Finite State Diagrams (FSD) have been used to model the dialogue. Using this formalism, we can specify dialogue states and transitions between them [2]. In each state there is a system message, which will be send to the user, and a language model (LM) to describe the expected user input. Each transition has a condition which is used to select the next state.

This is the easiest approach, where the system has the initiative and guides the user through the contents of the portal. There are some limitations of flexibility, although this model is really suitable for navigation, because it resembles the web one.

To map web contents to a FSD, representing the way the system makes the navigation, we have proposed a set of rules to automatize that transformation:

1. The structure of the web site must be captured using a graph. This graph can be used as our first FSD, which has to be improved to achieve an user-friendly interaction.
2. Each state in the FSD with large amount of information to be send to the user has to be changed: navigation through its contents must be allowed. There are two ways of doing this:
   (a) Split that state into several ones, adding the proper transitions between them.
   (b) Enable a search mechanism through the contents of that state.

Eventually, LM must be associated to each state:

1. A grammar made up of all literal strings found in all associated conditions that go out from that state.
2. In search states, we have to build a grammar that can cope with all possible queries we can expect from the user.

The finite state diagram associated to our application is shown in Figure 2, which evolved from the web structure in Figure 1.

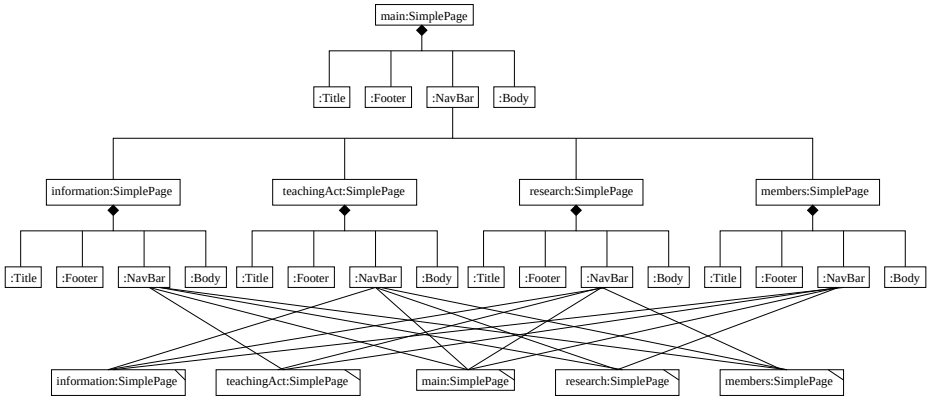

Fig. 1. Departmental Web Site Structure.

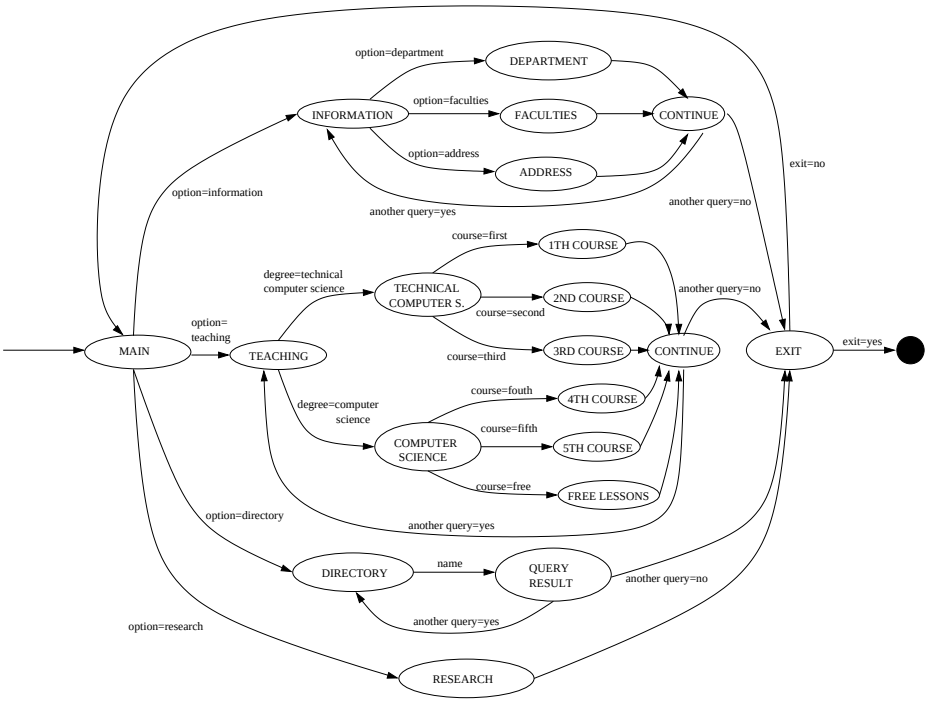

Fig. 2. Finite State Diagram of the Application.

By now, the construction of the FSD is made manually, following the proposed set of transformation rules, although we are working in the automatization. This will involve the parsing of the HTML code and the identification of the relevant elements of each page. Having in mind that each web site has its own style, an initial configuration would be required from the user, to enable the system parsing web pages properly. This avoids the need of a special design of the web pages and enables the reuse of legacy systems.

4 Underlying Platform

Once the application is modelled, the next step is designing the underlying platform to implement the whole system.

VoiceXML was chosen to modelling dialogues. It is one of the emerging voice-telephone technology, and is a standard that allows layered independence between the application and the technological platform at hand. As a result, developed applications can be executed in different platforms.

Taking the architectural model described in the standard as reference [4,5], the system has three main components: the *Document Server*, the *VoiceXML Interpreter* and the *Implementation Platform*. The system developed by our group implements the version 1.0 of the standard [4], but we are expecting to adapt to the newer version 2.0 [5], at the moment under development.

5 Conclusions and Future Work

Our main goal, the development of a Voice Application was achieved. VoiceXML is becoming a standard for browsing voice contents. It takes ideas from HTML, and borrows web information browsing model [3]. No doubt its key to success relies on application portability [1].

But that is not enough. An extension of the standard would be needed to overcome some of its limitations (e.g. [6]). There are two main points to be addressed: support of spontaneous user utterances and the ability to design more flexible dialogues which lets the user to take the initiative.

A major challenge would have to do with direct HTML contents reuse for spoken dialogue environments. This is not an easy task, since web pages lack the structure needed to model voice navigation. A reorganization of contents is also needed to achieve a natural and user-friendly interaction with the user.

A set of rules were presented to map web contents into voice ones, although further work has to be done for accessing complex sites.

References

1. Danielsen, P. J.: The Promise of a Voice-Enabled Web. Computer. Vol. 33(8), August 2000.
2. De Mori, R.: Spoken Dialogues with Computers. Academic Press. 1998.
3. Lucas, B.: VoiceXML for Web-Based Distributed Conversational Applications. Communications of the ACM. Vol. 43(9), September 2000.
4. VoiceXML Forum: Voice eXtensible Markup Language (VoiceXML) Version 1.0. March 2000. `http://www.voicexml.org/specs/VoiceXML-100.pdf`.
5. W3C: Voice eXtensible Markup Language (VoiceXML) Version 2.0. October 2001. `http://www.w3.org/TR/2001/WD-voicexml20-20011023/`.
6. Tsai, A., Pargellis, A.N., Lee, C., Olive, J.P.: Dialogue Session Management Using VoiceXML. Proceedings of the European Conference on Speech Technology, EuroSpeech. 2001.