

WIZARD-OF-OZ FOR ELDERLY EL-WOZ

1. INTRODUCTION

Elderly Wizard-of-Oz (EL-WOZ) is a tool for (spontaneous, read) speech collection in a real situated spoken dialogue task by simulated spoken dialog system.

1.1 OVERVIEW AND PURPOSE OF THE TOOL

The most important challenge in collecting elderly speech is to design a suitable interface for the specific target speakers. We developed EL-WOZ in a modular manner with a simple interface that makes the collecting easy and effortless. This system runs on two computers connected by a local network. The overall structure, outlined in Figure 1, consists of three modules: the Wizard, User, and Text To Speech modules.

2. TECHNICAL INFORMATION

SOFTWARE DEPENDENCIES AND SYSTEM REQUIREMENTS

The EL-WOZ is a client server based framework that communicates through HTTP connection. This software is implemented in Java. Thus, it requires the installation of Java SDK (version 1.7 or above recommended). It is recommended to run this software with more than 2GB RAM.

INSTALLATION

You can install this software by downloading the ZIP file and unzip it in a directory of your choice. You should notice that before running the software, a Text to Speech module (TTS) must be configured. The configuration steps are described in session 3.

EXECUTION INSTRUCTIONS

Assuming that Java SDK is installed and TTS is configured, this software can be run from any terminal in any Operative System (Windows, Mac OS, and Linux). In order to run the software you should follows these 4 steps:

- 1- Go to the root directory and Run the strat.bat file
- 2- Set the dialog system and create a new user
- 3- Open a browser and type the correct URL. By default the URL is :

```
HTTP://LOCALHOST:9696/WEBASSISTANT3/WEBASSISTANT-WOZDIALOG.HTML
```

- 4- Send the first sentences to the user and start the speech collection

3. CONFIGURATION

1- To configure the User interface, you should go to the root directory\config\WebAssistant.xml :

```
PROPS>  
<PROP KEY="CONTEXTNAME">/WEBASSISTANT3</PROP>  
<PROP KEY="SERVLETNAME">WEBASSISTANT.FLV</PROP>  
<PROP KEY="BASECACHEDIR">VIDEOCACHE2</PROP>  
</PROPS>
```

To set the port address you can modify the file at root directory \domains\WebAssistant\
WebAssistant.properties

```
SERVERPORT = 9696
```

The speaker character is set in the file at root directory \domains\WebAssistant\
WebAssistant.properties

```
# CHARACTER DEFINITION  
CHARACTER.DEFAULTCHARACTER = EDGAR
```

2. No TTS module is provided, you should use your own module

4. ARCHITECTURE

The system is composed of two main parts, the User Module (UM) and the wizard system. The UM runs on the client side and the wizard system runs on the server side. When both applications have started and the connection has been established, the experiment can be initiated. The wizard user on the server side may select one of the sentences among a set of predefined sentences. The corresponding text file will be then sent to the TTS module (not provided) and then to the UM (figure 1). In the next subsequent subsections, the main system modules are discussed in more details.

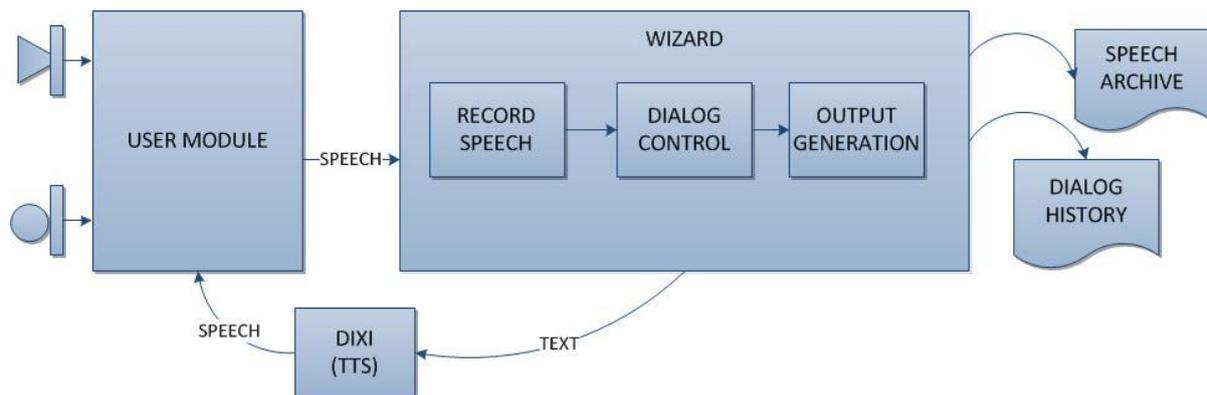


Fig.1 Architecture of the EL-WOZ system

4.1 USER MODULE

The user module runs on a separate computer (client), and is connected to the wizard's one (server) through the network (HTTP connection). Users interact with the system just by verbal communication. They have been informed at the beginning of the test that by clicking on the "Speak button" they can answer to the question. Finally, the UM takes the wizard outputs (speech) and play them to the user. Figure 1 shows an example of a UM interface. The character is not provided with this software.



Fig.2 Example of a User Module Interface

4.2 WIZARD MODULE

The Wizard includes three objects: Record Speech (RS), Dialog Control (DC), and Output Generation (OG). After receiving the speech in the server side, the RS module converts it to the WAV format and archives it in a speech archive folder. All the recorded speech is archived by user name and date. The Dialog Control object plays the managing and controlling role of the system. It is responsible for replying to the user by giving a suitable answer; this object is controlled by the wizard user. In fact, the wizard user is responsible for asking the proper questions.

Dialog control

Figure 3 shows the Dialog Control interface. Multiple solutions have been proposed to model and control the interactions between a dialog system and its users. The box number 1 located on the left side of figure 3 is the predefined scenario. Button number 2 is for escaping the current question to the next one. The box number 3 is the real time response where the wizard user can type manually any question. The box number 4 is the set of thanks sentence for finishing the conversation. The box number 5 is the set of repeat request sentences, the box number 6 is set of disagree sentences, the box number 7 contains agree sentences and the box number 8 on the right is the dialog history. This box is designed to better control the conversation and finally the button number 9 is responsible to send selected sentence to the user interface module.

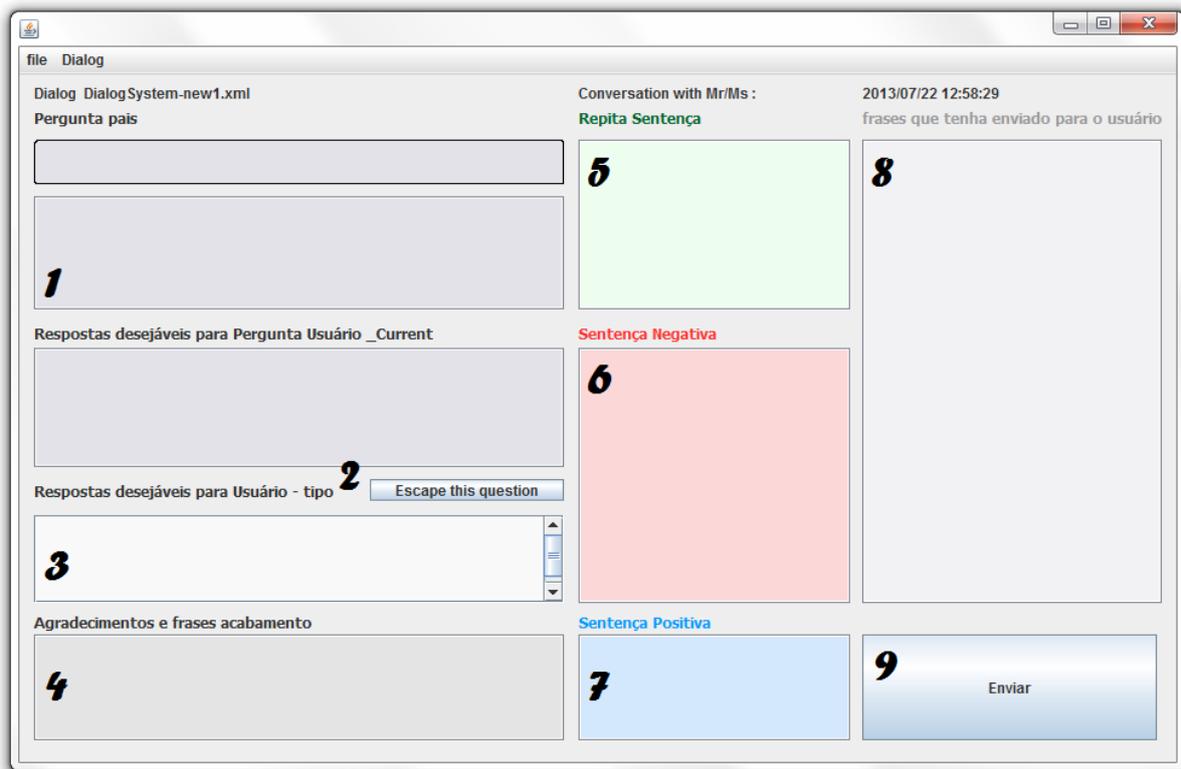


Fig.3 Dialog Control interface

Dialog control interface has two inner pages which one is responsible for create dialog (figure 4) and the other for setting it (figure 5).

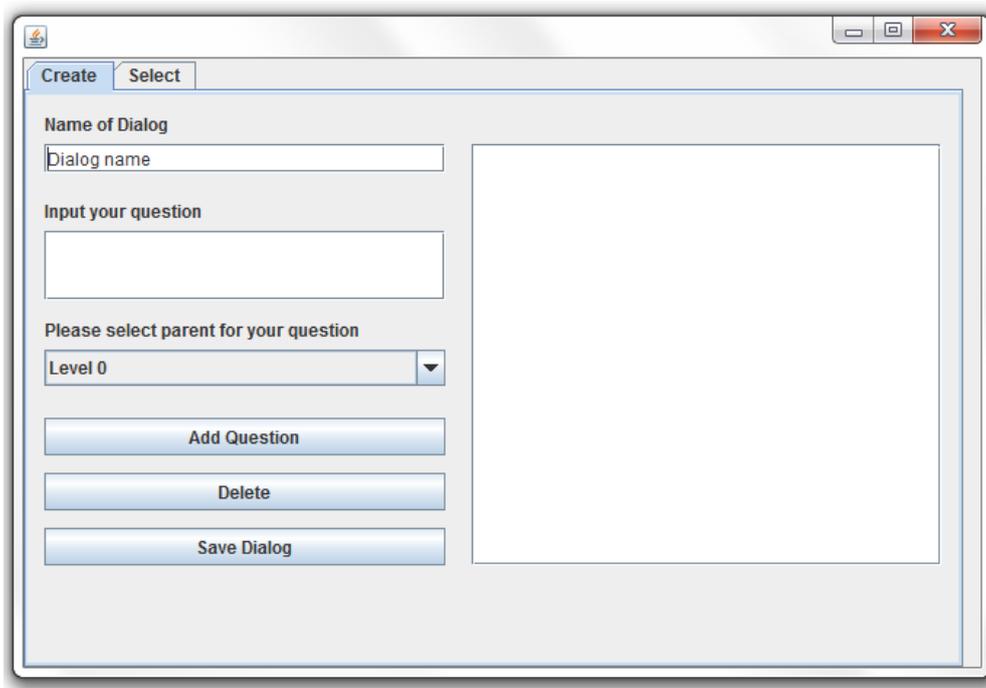


Fig.4 Create Dialog – create tab

After building and adding a new dialog you should find it from the drop box and set it as a default dialog.

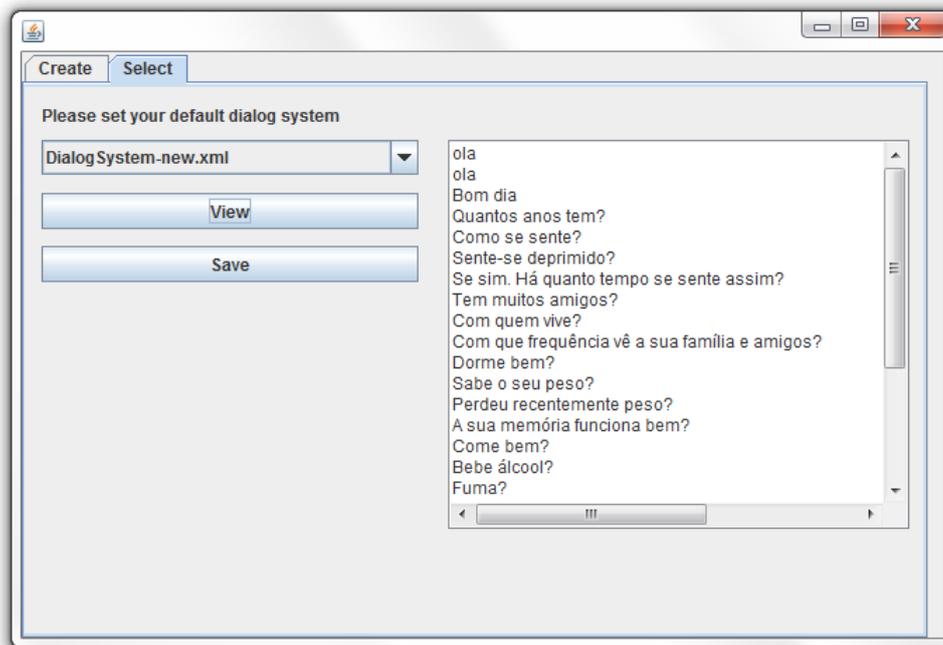


Fig.5 Create Dialog – Select tab

How to define the questions and answers

The questions and the answers that the Wizard can use in the dialog control interface are defined in several XML files:

Figure 4 and figure 5 shows the steps of creating and setting a new dialog. For example to creating a small dialog you need to follow the steps below.

Sample dialog:

BOA TARDE, CHAMO-ME EDGAR. GOSTAVA MUITO DE FALAR UM POUCO CONSIGO. EU SOU DE SINTRA.

OLA, CHAMO-ME EDGAR. GOSTAVA MUITO DE FALAR UM POUCO CONSIGO. EU SOU DE SINTRA.

BOM DIA, CHAMO-ME EDGAR. GOSTAVA MUITO DE FALAR UM POUCO CONSIGO. EU SOU DE SINTRA

O SENHOR É DE LISBOA?

A SENHORA É DE LISBOA?

- 1- Go to dialog In dialog interface and select the create tab, Set the name of dialog
- 2- Write the first sentence of dialog in (input your question) box.
- 3- For the first question we set the 0 level
- 4- Simply add question.

Notice that all question in a same level will be appear in a same category at dialog control interface, then you can select one of them to send. Setting same level for sentences means they have almost the same meaning. Levels are designed to avoid repetitious question.

- 5- For the other sentences you need to repeat all steps.

- 6- In case of typing mistake or any other mistake you can simply select the sentence and push delete button.
- 7- At the end you need to save dialog.

The new dialog can be find in this directory:

```
ROOT DIRECTORY \WEBASSISTANT-WIN32\DIALOG
```

System will generate a XML file similar to below which you can edit them with any text editor.

```
<?XML VERSION="1.0" ENCODING="UTF-8" STANDALONE="NO"?>
<DIALOGSYSTEM>
<QUESTIONS>
<QUESTION PARENT="0" >BOA TARDE, CHAMO-ME EDGAR. GOSTAVA MUITO DE FALAR UM POUCO CONSIGO. EU
SOU DE SINTRA. </QUESTION>
<QUESTION PARENT="0">OLA, CHAMO-ME EDGAR. GOSTAVA MUITO DE FALAR UM POUCO CONSIGO. EU SOU DE
SINTRA.</QUESTION>
<QUESTION PARENT="0" >BOM DIA, CHAMO-ME EDGAR. GOSTAVA MUITO DE FALAR UM POUCO CONSIGO. EU
SOU DE SINTRA.</QUESTION>
<QUESTION PARENT="1">O SENHOR É DE LISBOA?</QUESTION>
<QUESTION PARENT="1">A SENHORA É DE LISBOA?</QUESTION>
</QUESTIONS>
</DIALOGSYSTEM>
```

Output Generation

This module is responsible for sending the text of the questions to the external TTS software (not provided). This module is also responsible for sending a copy of all questions to the history dialog folder. The TTS system converts the output text to speech and sends them to the user module.

OUTPUT FILES IN ROOT DIRECTORY:

- 1- The user folder, include speech wav file and dialog xml file

```
ROOT DIRECTORY \USER\WAV&XML
```

- 2- Dialog xml files

```
ROOT DIRECTORY \DIALOG\
```

- 3- External modules files

```
ROOT DIRECTORY \ RESOURCES
```

- 4- Configuration files

```
ROOT DIRECTORY \CONFIG
ROOT DIRECTORY \DOMAINS\WEBASSISTANT
```

- 5- Web part files :

```
ROOT DIRECTORY \WEB
```

APPENDIX

THE MODULE THAT WE PROVIDE AT ZIP FILE

- | |
|-------------------------------------------------|
| 1- Web assistance interface |
| 2- Framework (WebAssistant-win32) |
| 3- Source code of Wizard of OZ(WozDialog-Maven) |

THE MODULE THAT YOU NEED TO PROVIDE

- | |
|------------------------------|
| 1- TTS – Dixi is recommended |
|------------------------------|