Adapting the Interaction in a Call Centre

Adapting the Interaction in a Call Centre System

Federica Cena and Ilaria Torre

Department of Computer Sciences – University of Torino
Corso Svizzera 185 - 10149 Torino (Italy)
{cena, torre}@di.unito.it

Address correspondence to Ilaria Torre,
Dipartimento di Informatica, università di Torino,
Corso Svizzera 185, Torino, 10149, Italy.
Phone: 39 011 6706827 Fax: +39 011 751603

Abstract. This paper describes ARRS (*Adaptive Response and Routing System)*, a system which personalizes the management of the entire answering process in a Call Centre infrastructure. The system is characterized by combining two kinds of adaptation: to the caller and to the operator. The aim is to support the user incrementally, improving the global quality of the interaction. In particular, the system combines a first process of automatic and personalized response with a second one of adaptive routing of the call to the operator who best fits the caller's features, according to a user-centered work flow. The results of a preliminary evaluation confirm the validity of the approach.

Keywords. Adaptation, VUI (voice user interface), user modeling, automatic response, calls routing, Call Centre

1. Introduction

A Call Centre is one the most important instruments used by organizations to meet their customers. It may be an opportunity to know the customers and to build a relationship with them. Unfortunately, nowadays this happens only in a few cases. Very often, interacting with a Call Centre is a frustrating experience for the user because of the poor quality of the provided service. In fact, given the high cost of manpower, many companies are switching from traditional operator-based solutions to self-service ones in order to save money. However, a self-service cannot entirely substitute the role of human agents: it can automate some operations but it is hardly able to manage complex inquiries and special cases, to face unexpected situations and also to overcome the psychological resistance of users.

Selfservice solutions are based on the so-called IVRs (Interactive Voice Response systems). There are two main kinds of IVRs, depending on the modality for receiving input from users: *touch-tone* systems exploit the keypad selection, while ASR solutions use *Automatic Speech Recognition*. Both of them have the serious problem of low quality of interaction.

ASR is a novel approach and can be a valid help for people with disabilities, as they can speak instead of dialing on the phone’s keypad. Moreover, it provides a more human-like kind of interaction, but also the ASR-based systems show several problems that make the experience worse: frequent misunderstandings, standardized answers, difficulty to face unexpected situations, etc., as discussed in Suhm et al. (2002), Bocklund and Bengtson (2002). The consequence is the perception of a paradox: a system with the aim of reproducing a conversational style of interaction which often fails to meet the user’s expectations, such as avoiding the boring repetition of instructions.

In order to face these problems and increase the global quality of interaction, we moved from a commercial ASR-based Call Centre software and developed a system which applies personalization and user modeling techniques to the different phases of the call answering process. The system also manages complex inquires, reasoning on the opportunity to involve the support of an operator, and in this case exploiting adaptation techniques to route the call. This solution provides flexible and a more human-like dialog interface, shortens phone calls and increases the satisfaction of callers. The prototype we developed regards the Call Centre of a *bank*.

In Section 2 we describe the problems and the goals of the system, while in Section 3 we present the architecture of the system. In Section 4, we explain the model of the user (caller and operators) and of the domain, in Section 5 the workflow of the call, in Section 6 the strategies for the personalized automatic voice response, in Section 7 the strategies for the adaptive routing, in Section 8 the evaluation results and finally in Section 9 the conclusions.

2. Overview of the system

The goal of "increasing the quality of interaction" is clearly too generic. Thus, to better define the goals of the system, we shortly discuss the main problems of operator-based and automatic Call Centers.

2.1 Preliminary research

As a starting point for this analysis we made an preliminary research on the interaction of users with two types of Call Centers: a *traditional one* with human operator and an *automatic one*[1] based on Automatic Speech Recognition (ASR). We recorded the calls of 50 subjects (25 for each system: test 1; test 2) recruited among the customers of the bank. For the selection we used the quota sampling method, considering age (14-30, 31-45, 46-70, over 70) and domain knowledge (banking domain knowledge: low, medium, high) as quota controls (variables for stratification).
For recruitment and tests, we exploited inbound calls of customers to the Call Centre of the bank, which behaved as a traditional Call Centre (for test 1) and then switched in the automatic modality (for test 2). When a customer called, she was asked for her password. The caller was identified from the authentication system and her data were retrieved. If, given her age and domain knowledge, she fitted into a not yet filled quota, she was asked to take part to the test (the age was retrieved from the User Data repository and the domain knowledge was estimated by the operator on the basis of the technical level of her request and other features, such as age, school level and job, if available). Then, each subject had to accomplish the same task, consisting in three different requests to the Call Centre:

- request for balance on mortgage loan, and in particular for the total amount due
- request for the scheduled payments for the current month
- request for knowing the best account package for her.

This choice allowed us to analyze the interaction in case of questions that are:
i) easy to recognize and to respond, ii) potentially ambiguous but easy to respond, iii) easy to recognize but complex to respond. There was no limit of time and all subjects called from dial up phones, without noise on the line. Afterwards, we analyzed the calls according to a set of objective parameters[2] (time to complete a task, number and type of errors, time spent in errors and recovery, number of repetitions of failed commands, positive ending, number of times users expressed frustration or satisfaction), and then we interviewed the callers with a questionnaire to analyze other parameters regarding the user satisfaction the adequacy of the answers, style of talk, etc. all express as evaluation on a 5-points numeric scale. For details see Section 8 where we describe how we exploited the same metrics for the summative evaluation of the system.

2.2 Identification of problems and related goals of the system

The result of the research was the identification of a number of problems that we grouped in three main classes, in turn split into sub-classes. They were used to define the goals of our project.

1) A ***first class of problems*** concerns the *difficulty of the system in understanding user's input* (see Table 1): it can be caused by a failure of speech recognition or a wrong interpretation of user's

[1] The *automatic* Call Centre switches the call to an operator (using the FIFO technique) after four consecutive failed attempts of recognition.
[2] The parameters taken into account are a subset of the usability metrics defined by Whiteside et al. (1988).

request. This problem has the consequence of extending the time of the call. Thus, we define our ***goal 1*** as the objective of *reducing misunderstandings.*

2) A ***second class of problems*** concerns the *difficulty of the user in understanding system's output*, and can be split in two main sub-classes, related to the front end and to the back end of the system[3].

- *Front End:* the research pointed out serious problems of *the user in understanding system's prompt* (non intelligibility of vocal instructions, misunderstanding the meaning of the prompts, disorientation) together with the problem of *negative perception* of the interaction with the system (inadequacy of the style of interaction and repetition of prompts, which is boring for the user and a waste of time).

  So, we defined the ***goal 2a*** as the objective of improving the voice user interface, in particular:

  - *intelligibility*: improving vocal quality of system's output
  - *understandability*: increasing the understanding of system's prompts by the user
  - *orientation*: simplifying user navigation in the vocal tree
  - *compelling*: make the user experience pleasant.

- *Back End:* the user has often *problems in understanding the content of system answers (e.g.*, users don't understand system explanations because they are too technical or don't find them useful because they are too general). So, we identified ***goal 2b*** as the objective of *increasing the understanding of meaning of the answer.* This issue shares many aspects with the design of support systems from which we borrow contributions.

	1) System does not understand user input	**2) User does not understand system output**
Unintelligibility of the sound	Speech recognition problem (Possible reason: noise)	Text-To-Speech problem[4] (Possible reason: unnatural prosody)
Misunderstanding of the meaning	Problem of match with grammar (reason: words do not belong to the lexicon of the system)	Automatic voice response problem[5] (Possible reasons: non adequate detail level, non adequate style),

Table 1 – Problems (1) and (2) compared wrt two common parameters.

3) Finally, a ***third*** relevant ***class of problems*** involves aspects of *problem solving*, as it concerns the inability of the system to manage complex inquiries and special cases and to face unexpected situations. ***Goal 3*** regards the management of these aspects.

To conclude this analysis, we can observe that the main goal of the system (improving the quality of interaction) and the related sub-goals can be largely regarded as a ***usability*** issue. In fact, as stated by Larsen (2003), despite the differences among definitions of usability, almost all refer to the three key concepts defined in the ISO 9241 standard: effectiveness, efficiency, and satisfaction with which specified users achieve specified goals in particular environments.

2.3 Adaptation techniques to achieve the goals

Our hypothesis was that the application of user modeling and adaptation techniques could improve each class of problems in spite of their different underlying technology (speech recognition, telephonic routing, etc.).

The study of user modeling techniques (e.g., acquisition, representation, maintenance of user models) and adaptation strategies which exploit these models is attracting increasing interest over the last ten years [see the reviews in Brusilovsky (2001) and Kobsa et al. (2001)]. Several applications have been

[3] An Automatic Call Centre is composed of a *front-end* (which has the problems and requirements of a Voice User Interface) and a *back end* (which has the problem of Support Systems).

[4] In this table, it corresponds to the first problem of class (2a).

[5] In this table, it corresponds to problems belonging to classes (2a) and (2b).

designed, in different areas and for different tasks, such as dialog and conversational systems [e.g. Horvitz and Peak (2001), Cavalluzzi et al. (2003)] intelligent tutoring systems [e.g., De Bra et al. (2004); Brusilovsky and Peylo (2003)], information systems [e.g., Ardissono et al. (2001), Cheverest et al. (2000), Billsus et Pazzani (2000)], etc.

The contributions of user modeling and personalization techniques is relevant each time the so called "*one-size-fits-all*" approach is not adequate. So the hypothesis was that that they could bring a relevant contribution in an automatic Call Centre, characterized by different users, predefined answers, problems of recognition and of understanding requests, time constraints, habit of users to interact with an operator instead of with an automatic system, etc. Different adaptation techniques are suited for the different classes of problems and goals discussed above. The table that follows summarizes the choice we made in ARRS. The rest of the paper will elaborate on it.

PROBLEMS	**SUB-PROBLEMS**	**GOALS**	**SOLUTIONS**	**ADAPTATION TECHNIQUES**
GLOBAL PROBLEM		**GLOBAL GOAL**		
User satisfaction		Improve global quality of interaction		
SPECIFIC PROBLEMS		**SPECIFIC GOALS**		
1. Problems of the system in undersanding user's input. → **Section 6.2**	- Failure of Speech recognition - Wrong interpretation of a request	- Improve speech recognition	- Manage habitability: bias user's answers	- Variation of the amount of information (adding hints and example of answer) - Variation of formulation of sentences
2. Problems of the user in understanding system's output (automatic voice response issue). → **Section 6.2**	**2.a** FRONT END: problems of the user in understanding system's prompts			
	- Intelligibility problems	- Improve vocal quality of system output	- Limit the use of TTS - Act on sound/type of words	
	- Understanding meaning problems	- Improve system understandability	- Consider the user familiarity in the use of the system	- Variation of the amount of information (hints, suggestions)
	- Problem of orientation	- Simplify navigation in the vocal tree	- Consider the user familiarity in the use of the system	- Variation of the amount of information (instruction on the use of system)
	- Negative perception of interaction	- Compelling: create a positive experience	- Change the application over the time	- Proactiveness: try to anticipate user needs
			- Errors handling	- Avoid recursive repetitions of user requests
			- Adapt the style of interaction	- Change the level of formality according to user age and cost
	2.b BACK END: problem of the user in understanding system's answers			
	- Too technical or too generic answers	- Make answers more understandable and useful for the user	- Consider the user knowledge of domain	- Additional explanations - Vary the technicality of language

3. Problem of adequacy and correctness of the system answer (problem solving issue). → **Section 7**	- Problem to face complex inquires, special issue, unexpected situations	- Adaptive context switch	- Consider caller and operators features	- Priority of calls - Selection of the best operator for the caller

Table 2 – A summary of system's problems, goals, solutions and techniques.

3. Architecture and components

Figure 1 shows the architecture of the system. The Call Centre infrastructure is the one of a typical CTI (Computer and Telephony Integration) connected to the Public Switched Telephone Network (PSTN) through a Voice Gateway. A *Communication Server* provides the basic functionalities for IP Telephony. The core of the system is represented by the *Response Manager (RM)* toward which the Communication server routes the traffic. The other agents are the *Speech Recognition engine*, the *Response Generation agent* (RGA), the *Routing agent* (RA) and the *User Modeling module*.

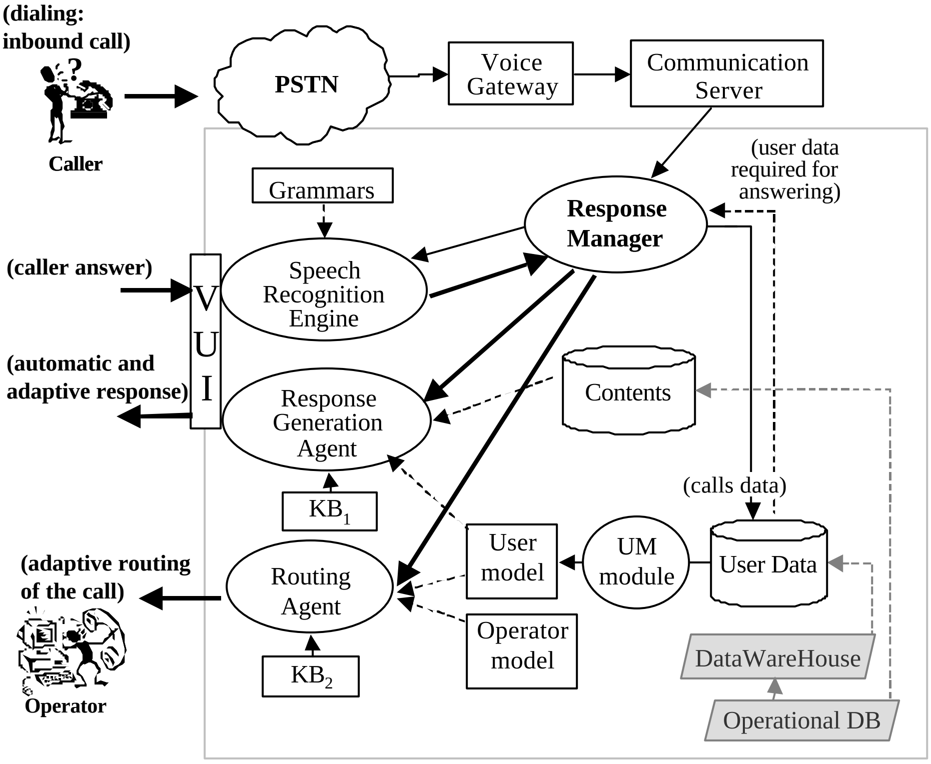


Figure 1 – Architecture of the adaptive work-flow

In the figure, light arrows indicate the processes managed by the legacy application while the bold arrows represent the interactions between agents to exploit adaptation. Finally, the dashed arrows point out the flow of data between agents and knowledge bases.

- The *Response Manager* controls the flow of the call, and the dialog among the modules
- The *Speech Recognition engine* is in charge of understanding natural language user input (Section 6.2)
- The *Response Generation agent (RGA)* composes prompts and answers, accessing the *contents DB*. For accomplishing the personalization tasks, it exploits a Knowledge Base (box KB_1), allowing it to determinate appropriate content given a user model (Section 6.2);
- The *Routing agent (RA)* manages the adaptive context switch. It routes the call to the operator with the highest probability of satisfying the caller needs, exploiting some heuristics (box KB_2) (Section 7).

- The *User Modeling Module* (*UM module* in the figure) is in charge of initializing and updating the models of the callers, starting from data stored in the *User Data repository* (Section 4).

Finally, the architecture includes a *Content repository and a User Data repository* .

- *Content repository* is fed by operational Data Bases of the bank: for the purposes of the prototype, we only take into account the portion regarding account operations and asset allocation. In order to classify the different requests of callers, we provide additional descriptions for the different types of operations.
- *User Data repository*: it contains data regarding the user (exploited by the User Modeling Module) and the calls. This database is fed by the DataWarehouse of the bank (see data in Section 4.1) and by the flow of the interactions (requests, percentage of calls switched to an operator and number of times a user hangs up while waiting for an operator).

Thus the system is built as a set of modules and knowledge bases which are integrated in an open and multi-purpose architecture. We developed a prototype of a *bank Call Centre*, based on a Cisco platform - *Customer Response Application* v3[6]-, which is supported by IP networks and runs in a Java environment. This platform allows the execution of the flow of the calls, providing a development tool (*Cisco Cra editor*) for building the tree of the call and the corresponding script. Other components of the platform are the ICM (*Cisco Intelligent Contact Manager*), which routes the call; the *TextToSpeech Nuance server***,** which translates text into voice; the *ASR Nuance server*, which contains the voice recognition engine (based on GSL -Grammar Specification Language).

On this commercia l platform, we integrated the specific modules, implemented with Jess (Java Expert System Shell) for carrying on the adaptation.

4. User and Domain Model

In order to perform adaptation, the Response Generator Agent and the Routing Agent exploit a model of the users (caller and operators) and a model of the domain.

4.1 User models

The Response Generator Agent personalizes the interaction with the user taking into account the model of the caller while the Routing Agent performs an adaptive switch taking into account the model of the *caller* and the models of all the *operators*, in order to select the most fitting one.

CALLER MODEL. The model of the caller is composed of a set of dimensions and is stored in the customer DB (see **Figure 1**). It is composed of factual and estimated data. The former come from the DataWareHouse of the bank and from the flow of interaction with the Call Centre system. The latter are inferred using some rules provided by the domain expert (the marketing CEO, in our prototype) and are processed by the User Modeling module. The dimensions depending on calls are updated at each interaction, while the others are updated periodically:

- *age* (date of birth), gender and region of birth are non changing factual data;
- *familiarity with the application* is estimated on the basis of the number of calls
- *domain knowledge* is estimated on i) frequency of technical questions ii) the user's job iii) education level, if present in the DB;
- *satisfaction* is inferred from the number of unresolved problems and complaints collected in the past interactions;
- *cost* of the caller for the company is related to the time subtracted to other calls and also to the monetary cost of the call if the number is a toll free one. It is estimated from the number of calls to the Call Centre in a given period;

[6] http://www.cisco.com/en/US/products/sw/custcosw/ps1846/ps5368/index.html.

- *customer value*, namely the customer's profitability for the bank, is estimated as a weighted combination of commissions amount, assets, insurances and held services, each represented as a distribution of probability on a scale from 1 to 4;
- *risk of churn* (the probability that the customer closes her account/s) is estimated on the basis of parameters such as complaints, duration of relationship, etc.

OPERATOR MODEL. The models of the operators are initialized by the Call Centre supervisor and updated periodically through evaluation tests.
The features taken into account are: age, gender, region of birth, skill, domain knowledge, communicative ability, expertise, speed of answer.

4.2 Domain model

With this term we mean the model of the *user requests* and of the *system answers.* As said, it regards the bank domain, restricted, for the purposes of the prototype, to the part regarding account operations and asset allocation. The *request of the user* is classified according to the following parameters in a numeric scale from 0 to 3:

- category of required service (account balance, bonds expiration, mortgage rate, etc.)
- complexity, related to the number and the difficulty of operations the system has to perform to manage
- level of the technicality of language (used to infer the user's domain knowledge, as seen above)

These parameters are defined in the Grammar (see Figure 1) used for user requests recognition.

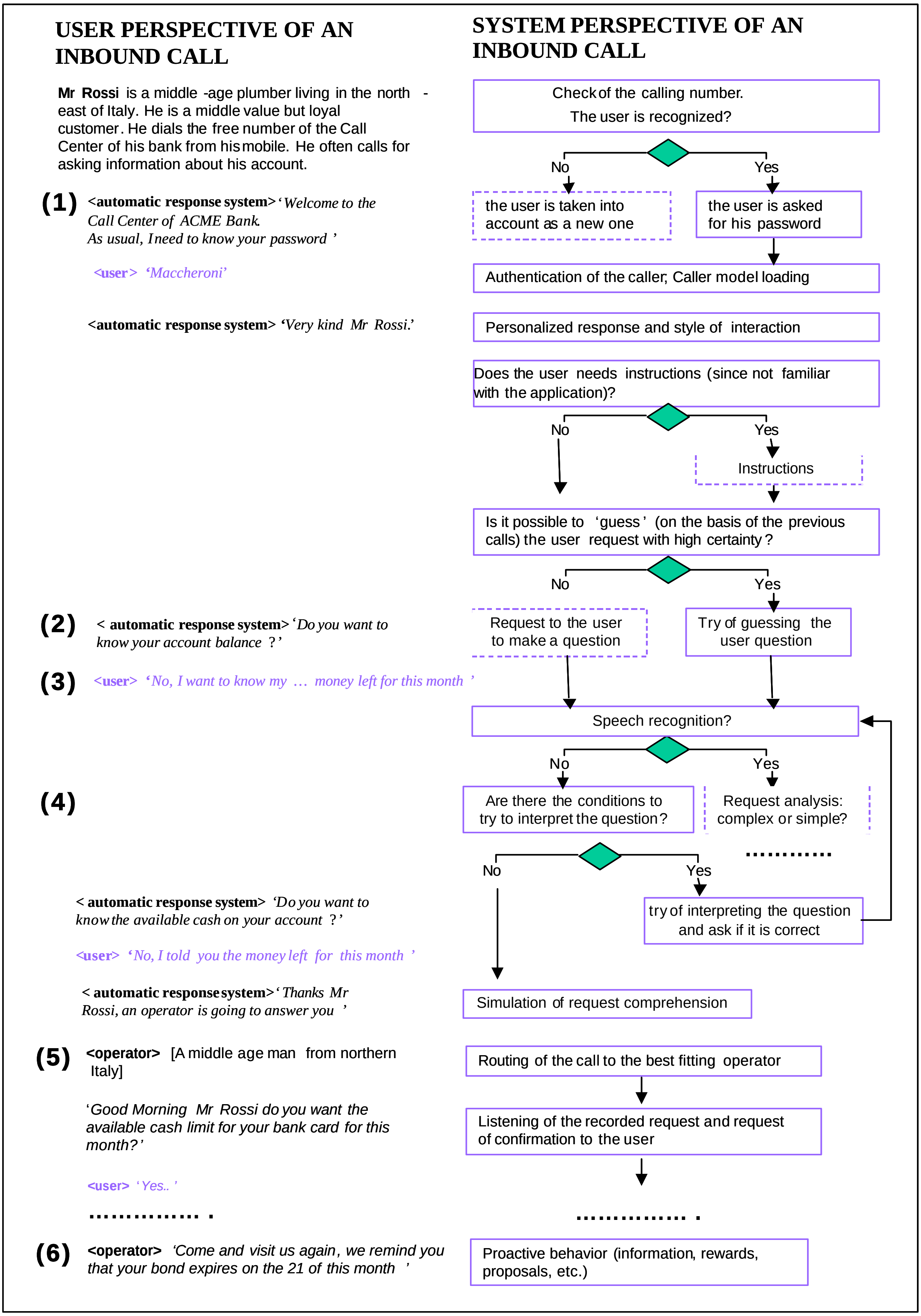
The *answer of the system* is modeled considering the following parameters (in a scale 0-3):

- type (answers or instructions),
- phase in the dialog tree
- category of required service,
- formality,
- complexity, in term of language used and detail level
- concision
- additional explanations regarding instructions and contents (explanations by examples are scored 3, the highest value in the scale)
- courtesy, like pleasantries, rewards, etc

As explained in 6.2, we record the invariant parts of the sentences, store them in the KB1, classifying them on the basis of the above parameters.

5. Work-flow of the answering process

In Figure 2, we provide an example of call-flow, analyzed from the user point of view and from the system point of view. Dashed boxes represent actions whose conditions that are not satisfied in the example.



USER PERSPECTIVE OF AN INBOUND CALL

Mr Rossi is a middle -age plumber living in the north - east of Italy. He is a middle value but loyal customer. He dials the free number of the Call Center of his bank from his mobile. He often calls for asking information about his account.

(1) <**automatic response system**> ‘*Welcome to the Call Center of ACME Bank. As usual, I need to know your password* ’

<**user**> ‘*Maccheroni*’

<**automatic response system**> ‘*Very kind Mr Rossi.*’

(2) < **automatic response system**> ‘*Do you want to know your account balance* ? ’

(3) <**user**> ‘*No, I want to know my … money left for this month* ’

(4)

< **automatic response system**> ‘*Do you want to know the available cash on your account* ? ’

<**user**> ‘*No, I told you the money left for this month* ’

< **automatic response system**> ‘ *Thanks Mr Rossi, an operator is going to answer you* ’

(5) **<operator>** [A middle age man from northern Italy]

‘*Good Morning Mr Rossi do you want the available cash limit for your bank card for this month?* ’

<user> ‘ *Yes..* ’

…………… .

(6) **<operator>** ‘*Come and visit us again, we remind you that your bond expires on the 21 of this month* ’

SYSTEM PERSPECTIVE OF AN INBOUND CALL

Check of the calling number.
The user is recognized?
No
Yes
the user is taken into account as a new one
the user is asked for his password
Authentication of the caller; Caller model loading
Personalized response and style of interaction
Does the user needs instructions (since not familiar with the application)?
No
Yes
Instructions
Is it possible to ‘guess’ (on the basis of the previous calls) the user request with high certainty ?
No
Yes
Request to the user to make a question
Try of guessing the user question
Speech recognition?
No
Yes
Are there the conditions to try to interpret the question?
Request analysis: complex or simple?
…………
No
Yes
try of interpreting the question and ask if it is correct
Simulation of request comprehension
Routing of the call to the best fitting operator
Listening of the recorded request and request of confirmation to the user
…………… .
Proactive behavior (information, rewards, proposals, etc.)

Figure 2 – A usage scenario of the call-flow

Whenever a call is received, the Response Manager checks the calling number; for an example of interaction see Cena and Torre (2003, 2004)[7]. If it is not recognized, the user is classified as a new one; otherwise, there are two options: the caller is a customer or she is using the phone of another customer. To avoid this possibility, the system asks for the password (**step 1** in **Figure 2**) to authenticate the caller. Consequently the model corresponding to the caller is loaded.

All decisions about the next action are carried out by the Response Manager, which requires the Response Generation Agent to welcome the caller and provide, if necessary, instructions about the use of the system. As it will be explained, for the composition of each sentence, the Response Generator Agent selects that one whose parameter values match with those of the caller. According to *goal 2a* (*personalizing the style of interaction*), the Response Generator Agent takes into account some user model features (age, level of scholarship, when available, and number of previous accesses) to produce a first form of personalization. Then it uses the age for personalizing the style (formal or informal) of the welcome formula *(goal 2a)* and the number of accesses for estimating the caller's familiarity with the application. For example, in case of new user, it provides instructions about the use of the application *(goal 1)*.

At the second stage of the workflow (**step 2** in **the figure**) the user is asked to make her question. In case the user is not a new one, in order to save time and change the interaction across time, the system tries to anticipate the user question (*goal 2a*). To evaluate this possibility, the Response Generator Agent analyzes the last caller requests, as it will be explained in 6.2.1 (in the example of call-flow, it asks Mr Rossi: "Do you want to know your account balance?", but unluckily he does not).

Whenever the caller makes a question (**step 3** in **the figure**), the sentence is processed by the Speech Recognition Engine (using a keyword spotting technique - Section 6.2) and evaluated by the Response Manager, in order to manage three possible situations:

- *the request is understood and identified as a simple one*; in this case, the Response Generator Agent is charged for supplying the requested service: it accesses the *contents DB*, it composes the answer, taking into account the caller features in order to adapt the content to her domain knowledge *(goal 2b)* and finally generates the final output;
- *the request is classified as a complex one*; in this case the Response Manager switches the call to the Routing Agent, which forwards the call to an operator, using the technique of adaptive context switch *(goal 3)*;
- *the Speech Recognition Engine does not recognize the request* (**step 4 in the figure);** the Response Generator Agent tries to interpret the question. The decision whether routing the call to an operator or asking the user to repeat the question depends on a combination of factors, which will be explained in Section 6.2.1. Anyway, when it decides to route the call to an operator, the Response Generator Agent answers to the user as if it had understood the question (in the example above "Thanks Mr. Rossi, an operator is going to answer you"). Subsequently, the Routing Agent is activated to perform the *adaptive context switch (goal 3)*.

At this stage of the workflow, the aim of the system is to create a context of interaction homogeneous with the first one, avoiding the perception of a system failure and satisfying the user request in the shortest possible time. To achieve that (**step 5 in the figure**),

- the call is routed to the operator which best fits the user's features, in term of characteristics such as competences, communicative ability, etc. (*adaptive routing*, see Section 7);
- then, the selected operator listens to the registration of the request that was not recognized by the automatic speech recognition: a human should have no problems to understand it and can thus provide the answer without asking for a new repetition;
- finally, at the end of the call (**step 6 in the figure**), a set of rules is fired with different purposes (alerts, requests of information, loyalty rewards, etc).

[7] The current version of the system has been slightly modified.

6. Personalized Automatic Voice Response

6.1 Voice User Interface features

Personalizing the Automatic Voice Response means acting on the Voice Interface, a relative novel way to interact with computer. So, first of all, we analyzed the differences between the interfaces (command line-based, graphical interface and voice interface) in order to choose the best adaptation techniques. An interesting comparison of command, menu-selection and natural-language interfaces is provided by Hauptman and Green (1983). In Table 3 we report the main issues.

	Command Line interface	**GUI (Graphical User Interface)**	**VUI (Voice User Interface)**	
			Touch-tone based	**Speech Recognition based**[8]
Interaction modalities in INPUT	Text	Text/ pointing	Number	Voice
Interaction modalities in OUTPUT	Text	Text/image	Voice	Voice
Input device	Keyboard	Keyboard/mouse	Keyboard	Voice
User control on system	High (continuous feedback)	High (continuous feedback)	Low (non continuous feedback)	Low (non continuous feedback)
System control on user	High (limited sets of inputs)	High (limited sets of inputs)	High (limited sets of inputs)	Low (unlimited sets of inputs)
User's training	Necessary (very difficult)	Necessary (simpler)	Not much necessary (enough intuitive)	Not necessary (very intuitive)
Presentation of information	Serial	Parallel	Serial	Serial

Table 3 – A comparison between different modalities of human-computer interaction

As we can see, Voice Interface has some peculiar features that make the interaction very different with respect to the other types of interfaces. First of all, of course, it allows human-computer interaction through voice instead of sight. This has important implications: on the one hand, this kind of communication is more natural, intuitive and simpler to use for humans, so they do not need any training to learn the use of vocal applications; furthermore, a VUI can be accessible for people with disabilities (especially mobility impairments of harms or blindness) and not very familiar with computers. On the other hand, a VUI has a lot of problems if compared to a GUI. First of all, [Dybkjær and Bernsen (2000)] "speech is perceptually transient rather than static" and imposes a serial presentation of information, so the interaction takes more time than in GUI, where the parallel presentation of data and the presence of icons generally make the communication faster. There is also a difference in the control level of the system on user input (speech-based VUI allow high freedom of formulation of questions that causes difficulties in recognition and understanding) and control of the user on system output (problem of latency of system's feedback that causes disorientation in the user). VUIs have a lot of similarities with Command Line Interface, mainly due to the serial presentation of information. On the contrary, concerning the difficulty of the interaction, we can place them on a scale from "very difficult" (command line interface that needs a lot of training) to "simple" (speech recognition based voice interface, since the voice is the most natural and intuitive modality of interaction between humans).

So, taking these issues into account, the design of efficient VUIs needs an approach that shares features with the other modalities but which also differ from them. Relevant contributions for

[8] This modality (used by ARRS) is not very frequent in commercial systems yet.

addressing usability issues in the design of the voice interface can be found in Dybkjær and Bernsen (2000), Sidhu (1996) and Stentiford and Popay (1999). The main objectives are to make the interaction faster and to increase the control of the system on user input.

6.2 The process of Voice User Interface adaptation

The VUI includes the processes of *user's request recognition* and *system's output generation* (see Figure 1).

The request of the user is processed by the Speech Recognition engine (developed with the Nuance ASR engine) through a speaker-independent *keyword spotting*[9] technique in unconstrained speech. The user composes her request with large freedom and then the system recognises only the keywords of the sentence (the service the user is interested in) included in the grammar (written in GSL language). Then, once recognised, the request is classified by the Response Manager according to the set of parameters seen in Section 4.2: (i) complexity, (ii) technicality and (iii) type of requested service.

The first parameter is used by the Response Manager to decide whether to invoke the Response Generation Agent (for an automatic answering) or the Routing Agent (for routing the call to an operator). The second parameter is exploited, together with her education level and job, to infer the user domain knowledge (Section 4.1); the third parameter is converted in an SQL instruction for retrieving the information on the Content DB.

The response of the system (Section 6.2.1) is generated by the *Response Generation Agent* (RGA), that is indeed made up of:

- a module that retrieves the answer to the user request from the Contents DB;
- a module that, given the type and category of answer, chooses the pre-recorded sentence (invariant parts of the answer) which is most adequate (in term of complexity, etc.) for that caller;
- a module which uses TextToSpeech for the dynamic generation of the words that are specific and variable (e.g. user name, user features, data from DB, etc.) and adds them to previously recorded message chunks (components of phrases to be composed).

6.2.1 Criteria of adaptation

The strategies for adapting the Voice Interface were elicited from two sources: i) the corpus of methodologies for Adaptive Hypermedia, as reviewed for example by Kobsa et al. (2001); ii) the study of the behavior of human Call Centre operators, driven by the fact that both *goal 2a* and *2b* (Section 2.2) are related with the objective of making the IVR more human-like. The main objective is to reproduce human operators behavior and their techniques of interacting with callers [see Bagnara (2000); Carlaw et al. (2003); Reynolds (2003)].

First of all, regarding ***goal 1*** (*improving speech recognition,* in Section 2.2**)**, we exploit an unusual methodology, which works on the sentences composed by the system rather than on the process of recognition. The basic idea is that personalizing the structure of sentences may bias the user answer and thus increase the probability of recognition (improving the "*habitability*"[10] of the system, i.e. the

[9] Keyword spotting is a speech recognition technique that associate a word, a group of words or a sentence in an utterance with a keyword. For example, the request "I want to know the money left for this month" is mapped by the ASR engine to the keyword "account balance".

[10] *Habitability* is a term coined by Watt (1968) to indicate how easily, naturally, and effectively users can use language to express themselves within the constraints of a system language. A system is considered habitable if users can express everything that is needed for a task using language they would expect the system to understand.

ability of a user to stay within the system's vocabulary most of time, [see Capra (2002); Ogden and Bernik (1996)]), by:

- formulating the questions in order to bias the user to give an answer with high probability to be recognized (for example, questions that implicitly contain the right answer),
- composing phrases with a degree of complexity which varies according on her domain knowledge (we observed that the better the user understands what the system is asking her, the clearlier are answers; this would also restrict the range of possible answers).

Moving to the *automatic voice response*, we have to address the ***goal 2a:***

- *Intelligibility*. To improve the vocal quality of system's output, we recorded speech for all the invariant parts of sentences, selecting words with clear and non-ambiguous pronunciation. For the other parts, that change dynamically, we use Text-to-Speech. The reason to limit the use of TTS is that its output is often hard to understand because of its unnatural prosody and poor intelligibility; see the experience of British Telecom, in Stentiford and Popay (1999).
- *Understandability.* In order to increase the understanding of system prompts by the user, we take into account the user *familiarity* in the use of the application. So, we vary the *amount* of information provided (adding text), as well as human operators do. For example, if the caller is a novice, we add some *hints* to the system prompts (in the form of examples of a possible answer), e.g. "Tell me what service you are interested in, for example the account balance".
- *Orientation*. In order to simplify user's navigation in the vocal tree, we consider again her *familiarity* in the use of the application, changing the navigation tree and giving instructions for browsing the application.
- *Compelling.* There are several issues to consider in order to make the user experience pleasant. First of all, we exploit a *style* variation, generating natural-language sentences characterized, for example, by different levels of formality according to the user *age.* Moreover, to avoid the risk of boring the caller, we change the interaction over time, avoiding the repetition of the same prompts. In particular, we analyze the caller's *last requests* and try to guess her current need (if 40% of the past requests regards the same subject and every other subject was involved in less than 10% of the calls, the system estimates a high probability for this kind of request, asking, for example, "Do you want to know your account balance?" as seen in the workflow of the call – Section 5). Another issue regards speech recognition errors and in particular the decision to ask the user to repeat her question. It depends on: the number of times the user has already repeated the question (maximum twice), the percentage of switches to an operator and the availability of operators. Finally, another parameter taken into account is the cost of the caller. If low, answers are less concise and more compelling.

Regarding ***goal 2b** - reducing the problems of the user in understanding the content of system's answer***),** the system may produce different versions of the same answer, varying the level of explanation and the technicality of the language on the basis of the *user knowledge* of the domain (if the user is a domain expert, we do not provide explanations of basic concepts and use a technical language; if the user has very limited knowledge of the domain, we provide her additional explanations using a very simple language, without specific words that she could not understand. "You are 5,000 euro in the red" is as example of sentence that can be used when the knowledge of banking domain is low (and the style of interaction informal). With an expert user we might provide the same information, using the technical terms "cash balance" or "accounting balance".
Moreover, with respect to the user satisfaction, the system answer may contain incentives, rewards and information, as seen in the workflow in Figure 2. For example, if the user has low satisfaction, she receives one more free operation over her monthly limit.

6.2.2 The composition of sentences

The sentences composed by the Response Generation Agent may be of two *types*: (a) answers and (b) instructions

As explained in Section 5, the script containing the logic of the call-flow is managed by the Response Manager. It decides which *type* of sentence is needed at each *phase* of the interaction or if it is necessary to route the call to an operator. Just for the (a)-*type* of sentence the Response Generation Agent needs to access the Contents DB (e.g. to query the account balance of the customer, the commissions on the last operation, etc.) , while it needs to apply adaptation strategies to both *types* of sentences. To do that, it needs to know: the model of the caller and the description of sentences (Section 4), so to apply the rules for the match between user features and response features. Regarding the caller model, the RGA takes into account the following features: age, domain knowledge, familiarity with the application, cost of the caller (depending on frequency she calls) and satisfaction.

Regarding the description of recorded sentences, each phrase pattern is classified according to its type, phase in the dialog tree, content category and to a set of descriptive parameters (formality level, complexity, concision, additional explanations and courtesy) each one evaluated on a numeric scale (scores form 0 to 3). E.g., the sentence “your account balance, that is the money left on your account, is 300 €...” has score 0 as formality, 0 as complexity, 3 as explanations, etc.; while the sentence with the same meaning “Your net assets amounts at 300 € ” has score 1 as formality, 1 as complexity, 0 as explanations.

The objective of the adaptation rules is to estimate, for each descriptive parameter, the score that best fits the user features (given the type, dialog phase and content category), so to select the most appropriate sentence.

First of all, the values of the user’s features are converted into a numeric scale (values from 0 to 3), and a set of rules defines the correspondences between users features and descriptive parameters. These correspondences, defined in collaboration with the marketing staff of the bank, are expressed, for each user feature, by means of an associative table and a weight K:

(*userFeature*(v))→K(*sentenceParameter*(s))

with:
v = value of the user feature, in a scale from 0 to 3
s = score of the parameter, in a scale from 0 to 3, associated to the value of the user feature by means of an associative table
K = weight that expresses the strength of the association in a scale from 1 to 5

For example,

(*satisfaction*(v))→ 4 (*courtesy*(s))

(*cost*(v))→ 2 (*courtesy*(s))

(*cost*(v))→ 4 (*concision*(s))

(*age*(v))→ 5 (*formality* (s))

(*domainKnowledge*(v))→ 4 (*contentAdditionalExplanation*(s))

(*domainKnowledge*(v))→ 3 (*complexity* (s))

(*familiarityApplication*(v))→ 5 (*orientationAdditionalExplanation*(s))

(*familiarityApplication*(v))→ 3 (*inputFormatAdditionalExplanation*(s))

……..

Note that each user feature may be associated to one or more sentence parameters, which means that, given the value (v) of a user feature, the ideal sentence for the answer has specific scores (s) for one or more descriptive parameters.

So for example, the parameter *courtesy* is associated to *customer satisfaction* with weight k=4 and to *customer cost* with k=2.

As a consequence, the final score for each parameter is calculated as a weighted average of the scores of the parameters associated to different user features.

For example,

(*courtesy*(s)) = 4 * (*satisfaction*(v)) + 2 * (*cost*(v)) / (4+2)

Thus, given the associative tables

Satisfaction	*Courtesy*
0 (unsatisfied)	3 (rewards)
1 (low satisfied)	2 (pleasantries)
2 (satisfied)	1 (polite utterance)
3 (very satisfied)	1 (polite utterance)

Cost	*Courtesy*
0 (first call: no cost)	2 (pleasantries)
1 (low)	1 (polite utterance)
2 (medium)	1 (polite utterance)
3 (high)	0 (no courtesy)

If the user satisfaction is 1 and her cost is 2, the ideal score for the parameter *courtesy* should be:

(*courtesy*(s)) = 4 * (2) + 2 * (1) = 10/6 = 1,66

In this way, the optimal score for each parameter is obtained and the Response Generation Agent selects, at each phase of the dialog tree, the sentences whose parameters' scores have the *shortest distance from the optimal ones*. Since a sentence can be selected even if the scores of its parameter do not perfectly match the optimal ones, this solution allows to reduce the number of sentences to be recorded.

In the previous example, the system selects the sentence with the shortest score from 1,66 in the current phase of interaction. E.g.:

"*Very kind for the details you provided*" (score 2),

"*Thanks for having called*" (score 1),

"*I am very sorry but I could not understand your request*", (score 2)

etc.

Finally, in case two or more sentences gained the same score, the Response Generation Agent chooses that one whose responses have the higher level of recognition[11], addressing the goal 1 related to misunderstanding.

As a very last step, the TextToSpeech module inserts the words that are variable into the selected sentence (e.g. the name of the caller in welcome formulas, the values returned from queries on the Contents DB, etc.).

7. Adaptive Routing

As seen in Section 2.2, we identify a ***goal 3*** connected to the last class of problems (managing complex inquiries and unexpected situations).

[11] For each sentence composed by the system, a flag is set which indicates if the consequent user answer has been recognized by the system or not.

7.1 Criteria of adaptation

The idea is that adaptation techniques can be relevant to manage the workflow when i) a question is identified as a *complex* one or ii) there are several *failures in understanding* the user request. In these cases the system performs an ***adaptive context switch,*** which routes the call to the operator that best fits the caller's features and thus integrates the two phases of the answering process. This is aimed at replacing the idea of inefficiency, disorientation and waste of time (that is typically associated by the user to the experience of switch from automatic interaction to human operator) with the perception of incremental support.

Common Call Routing systems route the calls following techniques like FIFO (as the system we used for the preliminary test) or considering some kinds of priority on groups of users. For example, a commercial advanced solution, CISCO ICM[12] Intelligent Contact Management, profiles each customer using contact-related data such as dialed number, caller-entered digits, data submitted on a Web form, or information obtained from a customer database. At the same time, the system knows which resources are available based on real-time conditions (agent skills and availability, IVR status, queue lengths, and so on) continuously gathered from various Call Centre components.

However, given our goals, this solution is still not satisfactory, since it does not provide the required flexibility. In particular, the requirements for our system are: i) determining the priorities of calls, ii) selecting the operator that is most adequate for answering a specific call; iii) performing the adaptive context switch, as explained in the work-flow of the call (Section 5). To implement that, we integrated the routing system with an *agent* (the *Routing Agent*) that takes into account the model of the caller and those of all the operators .

Both the *Response Generation Agent* and the *Routing Agent* access the same user model of the caller, but use different dimensions. The specific features taken into account by the *Routing Agent* are customer value, her risk of churn, domain knowledge, satisfaction, age, gender and region of birth. As regards the operators, the system stores a model for each one, considering: skill, domain knowledge, communicative ability, expertise, speed of answer, age, gender and region of birth.

7.2 Call routing

The routing of the call is managed in three steps: definition of the priority of the call, identification of the ideal operator wrt the caller features and final selection of the operator.

1) Definition of the priority of calls

Ordering the calls is a relevant task as the waiting time is one of the critical factors for the caller satisfaction. Thus, as described by Koole and Mandelbaum (2002), abandonment, waiting and/or retrials have to be taken into account carefully.

In ARRS, the order of calls is based on their priority, which is defined by means of a set of rules stored in a Knowledge Base (KB_2, in Figure 1). The antecedent contains a set of conditions on caller features and the consequent assigns the priority value. Note that all these rules are based on the knowledge elicited from the marketing staff.

The caller features taken into account are her customer value, risk or churn and satisfaction. Two examples of rule are:

if <riskOfChurn=2 AND customerValue:3> then <priority=3>
if <riskOfChurn=3 AND customerValue:1> then <priority=2>

In addition, we consider the number of times the user waited for an operator and then hanged up in the last week. If the number is higher than three, the call gets a high priority value, independently of the other features.

[12] http://www.cisco.com/en/US/products/sw/custcosw/ps1001/index.html

2) Identification of the best operator for the caller

The criteria for the match caller/operators are defined in a set of rules, stored in a Knowledge Base (KB_2, in figure 1). Some of them are simple correspondences between caller and operators features. E.g. "same area of birth" is due to the fact that people are pleased when the unknown voice that answers is somehow familiar to them. Other rules use a compositional approach similar to that one described for the match caller/sentences by means of associative tables. In this case, instead of selecting the sentence with the optimal scores for its parameters, the problem is selecting the operator with the optimal values for her features:

$$(\textit{userFeature}(v)) \rightarrow K(\textit{operatorFeature}(s))$$

Thus, first of all, we identified the requirements operators have to satisfy, given the caller features. For example, in our rules, the *customer value* is relevant for choosing an operator with an adequate *expertise* and *speed of answer*, respectively weighted as 4 and 2. For the customer *risk of churn* we consider the *communicative ability* of the operator and her *expertise*, weighted as 4 and 3, etc.:

Like for all rules, weights and associative tables were defined on the basis of the marketing staff requirements.

Similarly to the approach in 6.2.2, using these correspondences we compute the score **s** of each ideal operator feature and the final score is calculated as a weighted average of the scores of the parameters associated to different user features.

Following the above associations, we have for example:

$$(\textit{expertise}(s)) = 4 * (\textit{customer value}\ (v)) + 3 * (\textit{risk of churn}\ (v)) / (4+3)$$

3) Final selection of the operator

Finally the system produces an ordered list of operators with the less distance from this score.
The very last decision about the choice of the operator depends on: i) the ranked list of operators ii) the priority of the call.
In particular, if the first operator in the list is idle, the call is routed to this operator, otherwise the Routing Agent checks the status of the second one in the list. If both of them are not idle, the call is inserted into both of their queues in a position given b its priority score.
After three minutes of waiting (which means that the caller is probably not a high value customer), the call is queued to the third and fourth operators queues too and so on[13].

8. Evaluation

As observed at the end of Section 2.2, the main goal of the system (improving the quality of interaction) and the related sub-goals can be regarded as a *usability* issue.
Thus, for the evaluation, we exploited some usability metrics, and in particular those from Whiteside et al. (1988), taking into account the differences between vocal and visual interfaces, as seen in Section 6.1 and stated by Dybkjær and Bernsen (2000).

The evaluation of adaptive systems is often done by comparing a non-adaptive version of the system to an adaptive one. In our case the non-adaptive system corresponds to the automatic Call Centre analyzed in the explorative analysis used as a control group in order to make the comparison. Therefore, to evaluate our application we repeated the tests of the explorative research, using the same criteria for the recruitment of the subjects and for the choice of tasks (explained in Section 2.1), and we calculated the percent difference between the two systems.

[13] Consider, anyway, that the queuing problem is very complex, due to the random arrival of calls. See Koole and Mandelbaum (2002) for a survey on queuing models for Call Centers.

i) *Recruitment*. We selected 48 subjects, with a stratified sampling method considering again age and domain knowledge (Section 4.1) as variables for stratification, since they are parameters used by the system for relevant aspects of personalization. As in the previous test, we exploited inbound calls.
ii)*Tasks*. As already described in 2.1, each subject solved a set of three tasks, all accomplished during the same call: i) asking for the total amount due for mortgage loan, ii) asking for the scheduled payments for the current month, iii) asking for information about profitable account packages.
Then, we exploited two distinct measures, one regards observable user performance, the other user opinion and satisfaction. In literature they are often referred as objective and subjective measures. For the first analysis we recorded the calls and elaborated statistics of time to complete a task, errors made, etc. For the second one, we exploited a close questionnaire with a final open question.

OBJECTIVE MEASURES

Evaluation parameters:
The parameters taken into account measure the achievement of the goals indicated in brackets. Three further parameters are not related to a specific goal and instead measure side effects of the *global goal of increasing satisfaction*. They are highlighted with the mark (*).

- Failures in speech recognition (*goal 1*)
- Wrong interpretations of a question by the system (*goal 1*)
- Repetitions of failed instructions (*goal 2a*)
- Selections of wrong vocal commands (*goal 2a*)
- Misunderstandings of system's answer by the user (*goal 2b*)
- Switches to operator (*goal 3*) as a consequence of understanding errors
- Time to complete a task (*)
- Time spent in errors and recovery (*)
- Number of times user expresses frustration (*).

Non-mentioned goals of classes 2a and 2b will be described later in this Section, since they need subjective measures.

Results analysis
For the analysis of the recorded calls, we used a matrix user/task for each parameter. For example, for the parameter "Time to complete a task", each cell contains the time expressed in minutes. In the tests, it varies from 0.66 (recorded in task 1) to 6.33 (recorded in task 3).
To compute the final value of parameters (one for each matrix) we take the geometric mean[14] of each cell value. Nielsen provides two main reasons for using geometric mean rather than the more common arithmetic mean: "first, you don't want a single big number to skew the result; second, the geometric mean accounts fairly for cases in which some of the metrics are negative" (Nielsen, 2001).
For each task and for each parameter we also computed the standard deviations, for evaluating the reliability of that measure (in term of homogeneity of behaviors wrt the task).
Finally, we compared the test group results with the control group results. This computation produced the following results which show the *variability* of the *geometric mean* between the first and the second group. This value is the final average of the variability related to each quantitative parameter, grouped by goal (details in **Appendix A**). In order to compute the statistical significance of the results, we also provide the P-value[15] with a T-test.[16]

[14] The geometric mean of N numbers is the N'th root of the product of the numbers.
[15] A P-value is a measure of the size effects of the results, so how much evidence we have against the null hypotheses H0. The smaller the p-value, the more evidence we have against H0.
[16] T-test deals with the comparison of two averages in order to assess whether they are statistically different from each other.

Goal	Variability (%)	P-value
Goal(1)	+13	0.77
Goal(2a)	+ 29	0.20
Goal(2b)	+ 36	0.38
Goal(3)	+ 4	0.89

Table 4 – Overall results for (objective measures)

SUBJECTIVE MEASURES

Evaluation parameters

In order to evaluate the goals that could not be included in the previous test, we exploited a close questionnaire structured into four blocks, plus a final open question. Each close question was evaluated on a 5-points scale.

Block1 contains brief questions about the user (age, gender, school level, experience in the banking domain, region of birth).

Block2 consists of overall user reactions (*global goal*):

- idea of efficiency of the system
- perception of control of the application
- perception of pleasant interaction
- perception of global satisfaction

(the first three questions are a sort of control questions for the fourth one, which explicitly regards the global goal of satisfaction)

Block3 contains judgments about the adaptive automatic response process
(*goal 2a*):

- difficulties in understanding commands and requests of the systems,
- learnability of menu commands,
- adequacy of command instructions,
- adequacy of the style of talking

(*goal 2b*):

- adequacy of explanations and detail level
- adequacy of system answers regarding pertinence

Block4 consists of questions for subject whose call was routed to an operator (*goal 3*):

- sense of frustration for call switch
- sense of disorientation for call switch
- sense of system failure for call switch
- adequacy of the operator

Results analysis

From the open question, we mainly had suggestions for improving the intelligibility of Text-to-Speech.

For the analysis of close questions, we used a matrix user/question. Each cell contains the value selected by the user for the specified question. To compute the final value of each question we use again the geometric mean, and a T-test to obtain the P-value as well. The results for each parameter are summarized below and reported in detail in **Appendix B.**

Goal	Variability (%)	P-Value
Goal(global): block 2	+ 25 (direct question on satisfaction), + 53 (control questions)	0.51 0.03

Goal(2a): block 3, questions 1, 2, 3, 4	+ 23	0.42
Goal(2b): block 3, questions 5,6	+ 36	0.37
Goal(3): block 4	+ 26	0.02

Table 5 – Overall results for (subjective measures)

Analyzing the all results we can observe that the *variability* values confirm our hypothesis that the application of user modeling and adaptation techniques can improve the different classes of problems (Section 2). In particular, regarding objective measures, the best result is obtained by Goal(2b) which means that adaptive techniques improve considerably the user's understanding of system's answers (indeed such techniques are well established in the community of Adaptive Hypermedia for this kind of problems). However, P-values higher than 0.1 would suggest no statistical significance of these results. Surely one of the reasons concerns the small size of the samples and in particular of the sample used for the first test. This hypothesis is confirmed by the fact that worst results have a big Confidence Interval (C.I.).

Subjective measures confirm in general the *variability* results of the objective ones, but gain higher P-values. An interesting result regards the comparison between parameters in Block 2: the first three questions in block2 have been conceived as a sort of *control questions* for the forth one regarding the global user satisfaction. However, the first ones obtained scores much higher than the direct question on satisfaction. This could lead to the interpretation that other parameters besides those considered may influence the user perception of satisfaction. Another interpretation which does not exclude the previous one is that the common negative feeling about Call Centers may influence the explicit opinion about satisfaction, as demonstrated by the fact that correlated variables (idea of efficiency of the system, perception of control of the application, perception of pleasant interaction) are scored with higher values and also gained very good *P-values*.

9. Conclusion

In this paper we presented ARSS, a prototype for the adaptive management of calls in a Call Centre system, which aims at supporting the user incrementally, starting from a personalized automatic support and moving towards a proficient human support. The prototype concerns a bank Call Centre but it is built using an open architecture and re-usable modules which allow us to port the same application to other domains. In particular the re-usable knowledge regards the model of the operator and the application independent dimensions of the caller model (e.g. number of calls, expertise in using the application, etc.). Most of the adaptation rules can also be exploited in CRM[17]-based applications.

The advantages coming from this approach are not domain specific and can be summarized as decreased misunderstanding, shortening of calls, higher control of the application, increased quality of interaction, and as a consequence of the global satisfaction.

Beside that, another effect is the decreased frustration of the operator due to the fact that repetitive operations are left to the IVR and the operator can manage situations that are adequate for her competences, due to the adaptive routing of the call. For an extensive discussion see Carlaw et al. (2003).

As future work, first of all we are going to repeat the evaluation with bigger samples to discover the causes of high P-values, moreover we would like to test the application using one of the frameworks developed for the evaluation of spoken dialogue applications. PARADISE [Walker et al. (1997)] for example, allows using multiple evaluation measurements, including quantitative metrics and

[17] The CRM (Customer Relationship Management) area is very wide as it covers all the fields and applications related to the management of the customer (contact systems, marketing automation systems, personalization agents, e-commerce systems., olap datawarehouse systems and ERP systems. Some of the vendors that manage are: Siebel, e.piphany, Pivotal, Avaya, Remedy, Genesys, etc. (see for example Dyché (2002) for an overview).

qualitative ones, such as the user satisfaction, then combined into a single measure and models performance as a weighted function, which takes into account task success and dialogue costs.

Regarding improvement of the system, we plan to apply adaptivity to the operator screen as well, in order to provide an adaptive support to manage calls, on the basis of the caller features. Moreover we plan to exploit Voice Xml Technology, to improve voice recognition. In particular, VoiceXML 2.0 (now accepted as a W3C recommendation) has also relevant features to improve accessibility for people with visual impairments, and, via text phones, for people with speaking and/or hearing impairments.

Another complex issue concerns errors handling. A dynamic adaptation of errors handling has not been studied yet, even if it could open interesting scenarios for the user support. The same thing can be said for the adaptation of interface (e.g. voice sound) to the device currently used by the caller (e.g. dial up phone, cellular phone, PDA, etc.) or to the context (e.g. level of noise).

10. Acknowledgments

Our thanks to Delos spa for providing the hardware platform, allowing to develop the application. We also thank Luca Console for his help and suggestions.

11. References

[1] L. Ardissono, L. Console, I. Torre, An adaptive system for the personalized access to news, *AI Communications* 14: 3 (2001) 129-147.

[2] S. Bagnara, Towards telework in Call Centres, *Euro Telework*, Bruxelles, (2000).

[3] L. Bocklund, D. Bengtson, Call Centre Technology Demystified, *CallCentre Press ICMI Inc.*, Maryland, (2002).

[4] D. Billsus, M. Pazzani, User modeling for adaptive news access, *User Modeling and User-Adapted Interaction* 10:2/3 (2000) 147-180.

[5] P. Brusilovsky, Adaptive hypermedia, *User Modeling and User Adapted Interaction* 11:1-2 (2001) 87-110.

[6] P. Brusilovsky, C. Peylo, Adaptive and intelligent web-based educational systems. *International Journal of Artificial Intelligence in Education*, Special Issue on Adaptive and Intelligent Web-based Educational Systems, P. Brusilovsky and C. Peylo (eds.) 13:2-4 (2003) 159-172.

[7] R Capra, Shared Context and Habitability in Voice User Interfaces. In the *Human Computer Interaction Consortium Winter Workshop (HCIC '02),* Snow Mountain Ranch, Fraser, CO, (2002).

[8] M. Carlaw, P.Carlaw, V. K. Deming, K. Friedmann, Managing and Motivating Contact Centre Employees : Tools and Techniques for Inspiring Outstanding Performance from Your Frontline Staff, The McGraw-Hill Companies, New York, (2003).

[9] A. Cavalluzzi, B. De Carolis, V. Carofiglio, G. Grassano, Emotional dialogs with an embodied agent, User modeling '03, P. Brusilovsky, A Corbett and F de Rosis (Eds), *Springer LNAI* 2702 (2003).

[10] F. Cena, I. Torre, Adaptive management of the answering process for a Call Centre system, *HCITALY Simposium*, Turin, (2003).

[11] F. Cena, I. Torre, Increasing performances and personalization in the interaction with a Call Centre system, *Proceedings of IUI* , Funchal, (2004) 226-228.

[12] K. Cheverest, N. Davies, K. Mitchell, P. Smith, Providing tailored context-aware information to city visitors, Adaptive Hypermedia and Adaptive Web-Based Systems (AH 2000*), Lecture Notes in Computer Science* 1892, Springer Verlag (2000) 73-85.

[13] P. De Bra, N. Stash, D. Smits, Creating adaptive applications with aha!, tutorial for aha! version 3.0, *Tutorial at the AH2004 Conference*, Eindhoven, Computing Science Report 04-20 (2004).

[14] L. Dybkjær, N. O. Bernsen, Usability issues in spoken dialogue systems, *Natural Language Engineering* 6: 3/4 (2000) 243-271.

[15] J. Dyché, The CRM Handbook: A Business Guide to Customer Relationship Management, Addison-Wesley, US, (2002).

[16] D. Goodine, L, Hirschman, J., Polifroni, S. Seneff, V. Zue, Evaluating Interactive Spoken Language Systems. In Proceedings of *ICSLP-92* (Vol. 1, pp. 201-204). Banff, Canada, (1992).

[17] A.G Hauptman,. B.F Green A comparison of command, menu-selection and natural-language computer programs. *Behavior and Information Technology,* 2 (2) , (1983) 163-178.

[18] E. Horvitz, T. Peak, Harnessing models of users' goals to mediate clarification dialog in spoken language systems, in *LNCS 2109*, (2001) 3-13.

[19] L.B. Larsen, Assessment of spoken dialogue system usability – what are we really measuring, *EUROSPEECH 2003 Proc.*, Geneva (2003)1945-1948.

[20] A. Kobsa, J. Koenemann, W. Pohl, Personalized hypermedia presentation techniques for improving online customer relationship, *The Knowledge Engineering Review* (2001).

[21] G.Koole, A. Mandelbaum, Queueing models of Call Centres: an introduction, *Annals of Operations Research*, 113 (2002) 41-59.

[22] J. Nielsen, Usability Metrics, in *Bi-weekly column Alertbox*, http://www.useit.com/alertbox/, January 21, (2001).

[23] W. C. Ogden, P. Bernick, Using Natural Language Interfaces, Computing Research Laboratory, New Mexico State University, Las Cruces, New Mexico in *Handbook of Human-Computer Interaction*, M. Helander (ed.) Elsevier Science Publishers B.V. North-Holland (1996).

[24]P. Reynolds, Call Centre Staffing: The Complete, Practical Guide to Workforce Management, The call centre School Press, US (2003).

[25] C.K. Sidhu, Designing for usability in industry-speech based services, IEE Colloquium on Interfaces - *The Leading Edge*, *IEE*, 126 (1996) 4/1-2.

[26] F. W. M. Stentiford, P. A. Popay, The design and evaluation of dialogues for interactive voice response services, *BT Technology Journal*, 17 (1999) 142-148.

[27] B. Suhm, J. Bers, D. McCarthy, B. Freeman, D. Getty, K. Getty, P. Peterson, A Comparative Study of Speech in the call centre, *ACM Press*, New York, (2002).

[28] M. Walker, D. Litman, C. Kamm, A. Abella, PARADISE: A framework for evaluating spoken dialogue agents, *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, ACL (1997) 271-280.

[29] J. Whiteside, J. Bennett, K. Holtzblatt, Usability engineering: Our experience and evolution, M. Helander (Ed.), *The handbook of human computer interaction*, New York (1988) 791-817.

[30] S.J. Whittaker, A. Forcella, C.K. Sidhu, M.C. Hall, Improving customer service through speech technology, *BT Technology Journal*, 14:2 (1996).

[31] W.C. Watt, Habitability, *American Documentation*, July, (1968) 338-351.

APPENDIX A

Results of evaluation: objective measures

In Table 6 we report the detailed results for each parameter, where *geometric mean* and *standard deviation* are calculated as the average of the results for each task for all subjects.
The *variability* is calculated with the standard formula of percent variation:
(*y* – x) / (*x*)

- x = Group 1, which represents the control group (automatic Call Centre)
- y = Group 2, which represents the test group (adaptive Call Centre)

Total indicates the average of the variability of all the parameters related to each goal.

Note that in presenting the **Totals for each goal**, negative values are converted into positive since they represent the decrement of negative parameters.

Finally, *T-test* and *P-value* are standard metrics used to compute the statistical significance of these averages.

Goal	**Evaluation parameter**	**Group 1 (x)**		**Group 2 (y)**		**Variability (y-x)/x**	**T-Test**	**P-value**
		Geometric Mean	**Standard deviation**	**Geometric Mean**	**Standard deviation**	**%**		
Goal (1)	Failures in speech recognition	0.76	1.77	0.74	1.56	- 3	0.048	0.9621
Goal (1)	Wrong interpretation of a question by the system	0.70	1.03	0.57	0.89	- 22	0.535	0.595
Total (1)						**+ 13**		0.77
Goal (2a)	Repetitions of failed instructions	0.49	0.45	0.34	0.30	- 43	1.502	0.1388
Goal (2a)	Selection of a wrong vocal command	0.49	0.23	0.43	0.20	- 15	1.105	0.2709
Total (2a)						**+ 29**		0.20
Goal (2b)	Misunderstanding by the user of system's answer	0.19	0.12	0.14	0.36	- 36	0.874	0.3839
Total (2b)						**+ 36**		0.38
Goal (3)	Number of switches to operator	0.15	0.37	0.14	0.17	- 4	0.128	0.8977
Total (3)						**+ 4**		0.89
(*)	Time to complete a task	2.57'	0.89	2.41'	1.10	- 14	0.671	0.5068
(*)	Time spent in errors and recovery	0.56'	0.45	0.47'	0.95	- 20	0.549	0.5865
(*)	Number of times user expresses frustration	0.49	0.39	0.30	0.66	- 63	1.543	0.1266

Table 6 – Summative results of objective parameters

(*) Side effect of the Global goal

APPENDIX B

Results of evaluation: subjective measures

In Table 7 we report the detailed results for each parameter, where *geometric mean* and *standard deviation* are calculated as the average of the results for each task for all subjects.
The variability is calculated with the standard formula of percent variation:
$(y - \text{x}) / (x)$

- x = Group 1, which represents the control group (automatic Call Centre)
- y = Group 2, which represents the test group (adaptive Call Centre)

Total indicates the average of the variability of all the parameters related to each goal.

Note that the negative results which represent a decrement of a negative parameter (such as "inefficiency of the system") are converted into positive for the **Total**s.

Finally, *T-test* and *P-value* are standard metrics used to compute the statistical significance of these averages[18].

Goal	**Parameter**	**Group 1**		**Group 2**		**Variability (y-x)/x**	**T-Test**	**P-Value**
		Geometric mean	**Standard deviation**	**Geometric mean**	**Standard deviation**	**%**		
Block 2								
(*)Global Goal	Idea of efficiency of the system	1.32	2.11	2.35	1.12	+ 78	-2.279	0.0276
(*)Global Goal	Perception of control on the application	2.15	0.91	2.90	0.78	+35	-3.505	0.0012
(*)Global Goal	Perception of pleasant interaction	1.13	0.94	1.64	1.32	+45	-1.905	0.0614
Global goal	Perception of global satisfaction	1.37	2.21	1.71	1.79	+ 25	-0.664	0.5144
Total (global)						control questions **+53**		0.03
Total (global)						direct question on satisfaction **+25**		0.51
Block 3								
Goal(2a)	Difficulties in understanding commands and requests of the systems	2.56	1.98	1.97	1.06	-30	1.39	0.1699
Goal(2a)	Learnability of menu commands	1.16	1.32	1.23	1.09	+ 6	--0.228	0.8197
Goal(2a)	Adequacy of command instructions	1.31	2.38	1.55	1.03	+18	-0.481	0.6386
Goal(2a)	Adequacy of the style of talking	0.82	0.79	1.13	0.50	+38	-1.785	0.0845
Total (2a)						**+ 23**		0.42
Goal(2b)	Adequacy of explanations and detail level	0.56	2.43	0.80	1.25	+42	-0.463	0.6509
Goal(2b)	Adequacy of system answers regarding	2.18	1.74	2.83	1.03	+30	-1.718	0.0952

[18] For its computation we used the free tool at the URL http://home.clara.net/sisa/t-test.htm

	pertinence							
Total (2b)						**+ 36**		0.37
Block 4								
Goal(3)	Sense of frustration for call switch	3.49	0.89	3.14	0.58	-10	1.779	0.0835
Goal(3)	Sense of disorientation for call switch	3.24	0.56	2.79	0.45	-14	3.476	0.0013
Goal(3)	Sense of system failure for call switch	2.82	1.01	1.83	0.99	-35	4.001	0.0002
Goal(3)	Adequacy of the operator	2.65	1.21	3.82	0.96	+44	-4.196	0.0001
Total (3)						**+ 26**		0,021

Table 7– Summative results of subjective parameters

(*) Control question for the Global Goal regarding satisfaction.

Figures

Figure 1 – Architecture of the adaptive work-flow 5
Figure 2 – An usage scenario of the call-flow 8

Tables

Table 1 – Problems (1) and (2) compared wrt two common parameters. 3
Table 2 – A summary of system's problems, goals, solutions and techniques. 5
Table 3 – A comparison between different modality of human-computer interaction 10
Table 4 – Overall results for (objective measures) 18
Table 5 – Overall results for (subjective measures) 19
Table 6 – Summative results of objective parameters 22
Table 7– Summative results of subjective parameters 24