Adding Telephone Interface to Web Service

Implication to the self-care support system for life-style diseases

Masatoshi Kawarasaki, Kyoji Konishi
Graduate School of Library, Information and Media Studies
University of Tsukuba
Tsukuba, Japan
mkawa@slis.tsukuba.ac.jp, s1121716@u.tsukuba.ac.jp

Makoto Ohara, Tetsuya Igarashi
Tsukuba University Hospital
Tsukuba, Japan
oharam@hosp.tsukuba.ac.jp, tediga@hosp.tsukuba.ac.jp

Abstract— **Various web services are offered over the internet and forms the social infrastructure indispensable to our daily life. Many of healthcare systems developed so far are also offered as web service. However, user interface of web services that require web access makes obstacle to elderly or inexperienced users. As a mean to mitigate such digital divide, we developed a generic method of adding telephone interface to web service and implemented it to the self-care support system for life-style diseases. In our method, by making software PBX cooperate with web service, input/output to web service can be achieved by ordinary telephone device and users can talk with web service interactively. Evaluation experiment revealed that the proposed method endures practical use.**

Keywords-component; Computer mediated communication, Health information management, Software architecture, User interface

I. INTRODUCTION

Although the number of patients and patients-to-be of lifestyle diseases such as diabetes is increasing, achievements of medical treatment are not effective enough. In this context, we developed the self-care support system for life-style diseases named iSMBG [1] and performed a demonstration experiment in a diabetes clinic to validate its effectiveness. The iSMBG is a web-based system to assist doctors and patients in interactively helping patient's self-care related activities by the use of cell phone. The server analyzes patient data automatically and sends an alert mail if self-care control is not sufficient. The validity of the system had been confirmed with an earlier experiment, but there was a problem existing that the elderly patients and/or patients inexperienced in handling information devices did not want to participate in the experiment. So, aiming at the increase of available users, we explored the method of adding telephone interface to the web-based iSMBG.

Regarding the addition of telephone interface, some methods are foreseen. The most likely method is remaking software PBX. There are some examples in e-commerce and call center that use IVR (Interactive Voice Response) and voice recognition [2]. But these are not general-purpose systems. More generic method would be the use of VoiceBrowser whose standards are being developed by W3C [3]. VoiceBrowser includes VoiceXML which allow XML to describe voice-applications just like HTML describes visual applications. It also includes supplementary XML such as CCXML[4] which describes the call control of the VoIP telephone service, and SSML[5] which describes the volume and pitch of voice. By using the function in this VoiceBrowser, it would be possible to add telephone interface to web service. But these standards are large in scale and open software is limited such as to a simple VoiceXML interpreter [6] or open source IVR[7]. It is difficult to cooperate with various web services by using VoiceBrowser in the current situation.

So we developed a new method of adding telephone interface to various web services that is easy to implement and has high scalability. In our method, by making software PBX cooperate with web service, input/output to web service can be achieved by ordinary telephone device and users can talk with web service interactively. Furthermore, fax output of user's web page and the use of voice message become available.

This paper proposes a generic method of adding telephone interface to web service and discusses its evaluation. Section II describes the overview and Section III compares existing and proposed methods. Section IV provides performance evaluation and Section V describes evaluation experiment of new iSMBG with telephone interface. Section VI concludes this paper.

II. ADDING TELEPHONE INTERFACE TO WEB SERVICE

Adding telephone interface to the web service is to connect PSTN (Public Switched Telephone Network) users to a web server via VoIP providers, by using software PBX (Private Branch eXchange), and to make the web service available to PSTN users through the telephone interface, as shown in Figure1. A patient can enter health data and receive response to and from the web service through telephone interface by operating a telephone dial with voice guide, thus allows the patient talk with his/her doctor interactively. Fax and voice message can also be utilized effectively. For example, a patient can receive a graphically-illustrated blood glucose data by fax, and can deliver his concern about self-care to his doctor by a voice message.

In order to add a telephone interface to a web service, it's necessary to share the database (in the following, DB) between web service and telephone service, and make database contents deliverable to the users through respective interfaces.

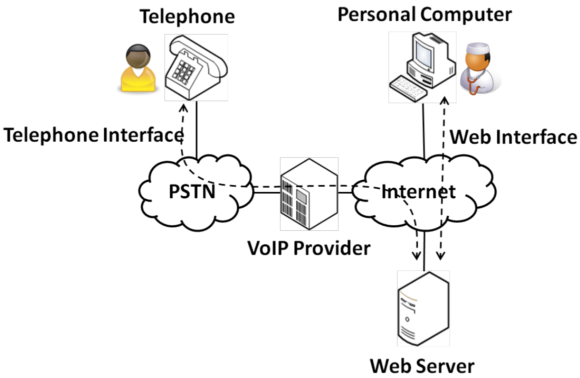


Fig.1. Addition of telephone interface to web service

Figure 2 shows the comparison of existing and proposed methods. In the existing method, access to web service and common database is achieved by an external program of the software PBX. In the proposed method, by adding HTTP input/output function to software PBX, input/output of database is performed only by the web service.

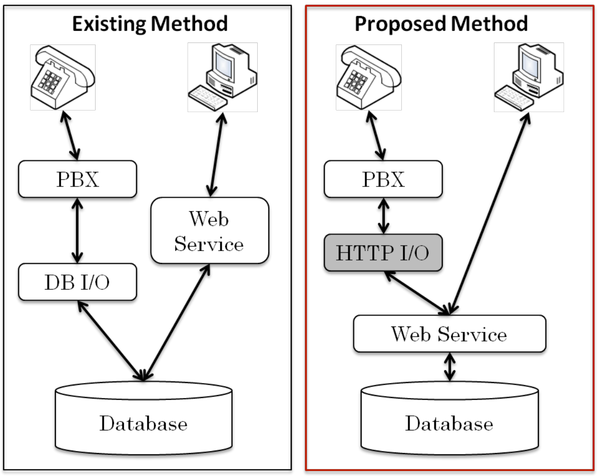


Fig.2. Comparison of existing and proposed method

III. Method of Adding Telephone Interface

A lot of software PBX is developed with the spread of VoIP, and most of the software PBX has the IVR function. The IVR function is used in a call center of an enterprise, where the voice file prepared beforehand is played on the user side of the system according to the user's dial control to perform an auto responding by voice. To achieve the telephone interface to web service using IVR functions offered by software PBX, it's necessary to develop software for respective web service. This makes it difficult to offer various web services interactively.

In this section, method of adding the telephone interface to web service is proposed. Proposed method is applicable to various web services in general, not limiting to iSMBG.

A. *Existing method and its problems*

To provide IVR service, conditions are usually described using the functions attached to software PBX. Figure 3 shows an example of Asterisk that is a typical example of software PBX. In Asterisk, it's possible to describe the response conditions of IVR in a procedural way by using AEL (Asterisk Extension Language). Furthermore, it's possible to write the user's entry data in the database because the external process can be performed by the special form as AGI (Asterisk Gateway Interface) from AEL. Similarly, it's possible to read data from the database and hand the result down to the telephone user.

By operating web service and common database in this way, it's possible to read out the contents offered by web service using the telephone interface. However, as IVR functions as attached to the software PBX are limited, it's difficult to provide complicated services by using this mechanism.

On the other hand, the necessity to prepare similar programs both in the web service side and the PBX side comes out because the intersection with the web service is only the database. As a result, it's necessary to mount a large-scale program both in the web side and the PBX side in order to provide PSTN users with interactive responses or with user specific recommendations from the web.

Furthermore, a developer has to describe the configurations of IVR in detail by different language specific to each software PBX, thus requires a great effort to a developer in adding the telephone interface to the web service that requests entry of various contents.

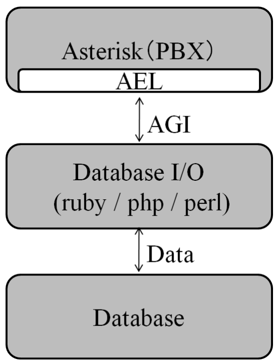


Fig.3. Interface structure of Asterisk

B. *Proposed Method*

1) Design Principals: Proposed method includes two design principles.

Firstly, the IVR function of software PBX is extended so that it can access to the established web server, play a voice file according to acquired XML, and transmit the user's entry data to web server. Extended function (HTTP I/O) behaves like a web browser. It reads XML data from the web server and transmits a voice file, user input format and input timeout period to the PBX. When PBX receives user's entry, it delivers the input data to the callback URL through HTTP I/O.

Secondly, a web service is assumed to have Model-View-Controller (MVC) architecture which is adopted in many of web framework. MVC architecture consists of *Model* which processes data, *View* which deals with display and data output, and *Controller* which controls *View*. We aimed at the architecture that makes it possible to provide the users with the result from web service through the telephone interface, only by just making alterations in *View* for HTTP I/O on the PBX side, keeping *Model* intact where various processing is performed.

2) System Behavior: The overall system behavior is shown in Figure 4.

(1) A user makes a call to PBX and sends input data. (2) PBX sends a caller ID and the input contents to HTTP I/O. (3) HTTP I/O sends a HTTP request with a caller ID and the input contents to *Controller* on the web service side. (4) *Controller* requests *Model* for data acquisition from DB. (5) *Model* performs read and write from DB according to the request from *Controller*. (6) *Model* receives a result from DB and returns the object to which O/R mapping was made to *Controller*. (7) The object which was returned from *Model* is processed at *Controller* and handed to *View*. (8) Processing result from *Controller* and XML generated from an output template are returned to HTTP I/O. (9) HTTP I/O issues a directive of voice file replaying to PBX according to XML description. (10) Voice is played by telephone and data entry is suggested to the user.

By repeating this procedure, all the services currently provided by web service can be provided through telephone interface.

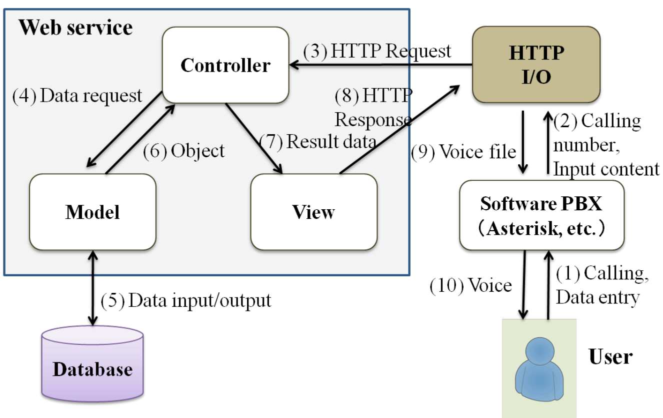


Fig.4. System behavior

IV. Performance Evaluation

The proposed method, when compared with existing method, voice file downloading on the web server side and real-time analysis of XML by HTTP I/O become overhead and may influence the performance in response time. Performance degradation caused by this overhead of the proposed method is evaluated in this section.

A. *Evaluation model and conditions*

To measure the overhead time in voice file downloading, we established an evaluation system where a web server and a PBX server are connected by 100Mbps line. Athlon64 2.2GHz was used as CPU for both servers, and CentOS5 was used for OS. RubyOnRails was used for web application framework and Asterisk was used for software PBX.

We measured the time length required to play voice after the user enters a key by telephone. Voice was compressed by 12.2kbps GSM-EFR voice coding system developed for cellular phones, and has the duration of 20 seconds.

We examined whether the overhead time of the proposed method is tolerable in actual use under these conditions. Following three kinds of overhead time as shown in Figure 5 were measured.

(1) Time length between a HTTP request and its response.
(2) Time length required for XML analysis (XML Parse)
(3) Download time of the voice file

It should be noted that required time to process input contents and get a result is not included in above three. This is because a similar overhead is needed as well in the existing method.

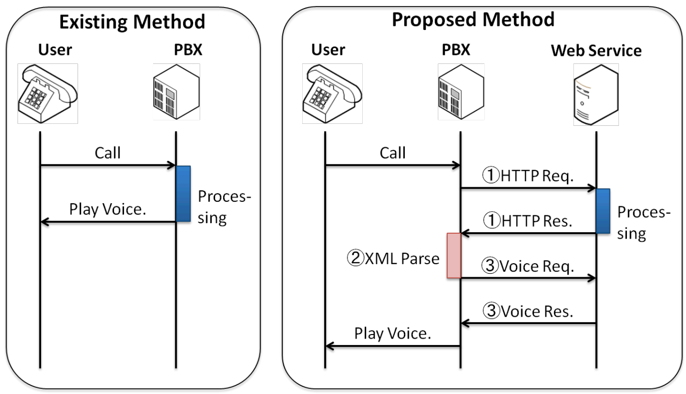


Fig.5. Signal sequence for performance evaluation

B. *Evaluation Results*

The measurement results of overhead time are shown in Table 1. We measured these metrics 20 times for each overhead and took an average. The result revealed that 80% of overhead time is download time of the voice file. Although HTTP request/response and XML analysis time don't fluctuate a lot, download time of voice guide is influenced a lot by the voice file size (the length of the voice guide). There is almost no overhead time on the PBX side because the processing time of HTTP I/O is simple and the voice file is compressed in low bit-rate.

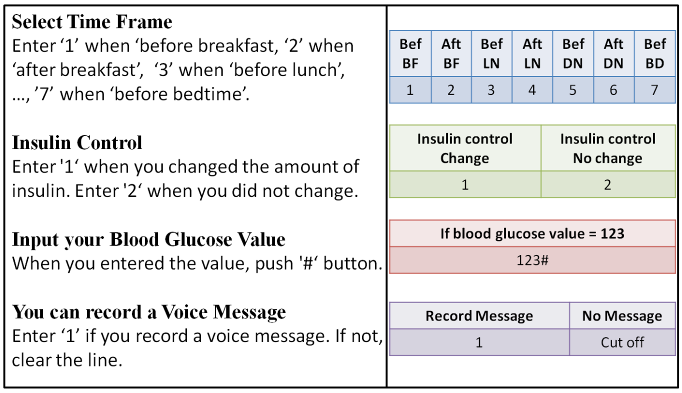
TABLE I
THE RESULT OF OVERHEAD TIME

HTTP request to its response	23ms
XML Parse	3ms
Download time	113ms
Total overhead time	139ms

From this evaluation result, extension in processing time caused by the overhead was approximately 0.14 seconds and it revealed to be hardly sensed by the user.

V. Evaluation Experiment Using iSMBG

We implemented above mentioned telephone interface to iSMBG and evaluated the usability of voice guide through a demonstration experiment. Ten students of Tsukuba University participated in this experiment as monitors. Each monitor student enters the virtual data to iSMBG through the telephone interface according to the voice guide manual as shown in Figure 6. The monitors enter five input items (four items of Figure 6 and one item for entry confirmation) twice a day during one week,



* You can enter your data even while voice guide is flowing.

Fig.6. Voice Guide Manual

A. *Evaluation Parameters and Exprimental Results*

The original purpose of the experiment was to investigate how the users acquire fluency in the telephone interface. However, on the way of investigation, it gave us useful parameters to evaluate the overhead time of the proposed telephone interface. These are three parameters whose meaning and experimental results are described below:

a) Appropriate response time from data entry: Response time is the time until the service returns a result by voice guide after user's data entry. When it's too early, there is a possibility that the user misses hearing the first part of voice guide, and when it's too late, the user has to wait for the result. As a result of questionnaire, 80% answered that one second was fine, and 20% preferred a slower response.

b) Upper limit in the number of entry items per one telephone communication: In this expriment, five entry items were requested during each telephone call. As a result of questionnaire, 90% answered that five input items were appropriate, and 10% answered that it could be a little bit more.

c) Upper limit of acceptable time length of a telephone call required to answer all the entry items: This is the acceptable time until the user disconnects the telephone after he/she makes a telephone call to the service. As a result of questionnaire, 30% answered 60 seconds and 70% answered 120 seconds, and the average was 102 seconds.

B. *Analysis of Experiment Results*

It revealed that an audio response from a system needs to be sent out at least more than one second from user's data entry, and that the total overhead time of 0.14 seconds as required in the proposed telephone interface is almost negligible level.

As for the length of voice guide which mainly determines the total overhead time, we calculated the upper-limit time needed to each entry item by the following formula;

(Total time length) / (Number of entry items)
= (Time length of each entry item).

The result was less than 21 seconds. This could be interpreted that 20 seconds of voice guide, as used in the experiment, is sufficiently long and appropriate, considering the time required for data entry and additional messages that flow besides the entry item.

VI. Conclusion

We proposed a method of adding telephone interface to web service as a measure to mitigate digital divide for elderly or inexperienced people who are not familiar to web interface. By expanding the IVR functions of software PBX and enabling coexistence with MVC architecture, the proposed method reduces workload in development and facilitates adding new functions. The result of evaluation experiment shows that increase in overhead is little and the response time fitted into a practical area.

As we use proprietary XML schema for the communication between web server and PBX server, it's a subject for further study to use standardized schema of VoiceXML or CCXML to make it a more general-purpose system.

We plan to carry out the demonstration experiment in a diabetes clinic again to evaluate new iSMBG added by telephone interface.

References

[1] M. Kawarasaki, T. Asano, M. Ohara, T. Igarashi, Development and validation of cell-phone based interactive self-care support system for life-style diseases, Japan Journal of Medical Informatics, Vol.29, No.5, pp.201-210, 2010 (in Japanese)

[2] WIPO Patent Application WO/2001/071543, System and method for using voice over a telephone to access process and carry out transactions over the internet, Quack, Com, Sep. 2001

[3] W3C "Voice Browser" Working Group - http://www.w3.org/Voice/, 2011

[4] D. Amyot and R. Simoes, Combining VoiceXML with CCXML, IEEE Proc. of Consumer Communications & Networking,(CCNC07), 2007

[5] O. Kiselyov, SXML Specification, ACM SIGPLAN Notices, 2002

[6] JVoiceXML (Open Source Voice XML Interperter), http://jvoicexml.sourceforge.net

[7] Zanzibar OpenIVR project, http://www.spokentech.org