ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)
Volume 5, Issue 4, April 2016

IVR-O-Fly : IVR Configuration On Fly

Suvarna Chavan, Dr.Y.V.Haribhakta

***Abstract*— In today's world Contact Centers plays very important role in business because they have a significant impact on customer retention rate. Today it's really challenging for contact center to serve the customer in a cost effective manner without disappointing the customer. Hence Contact center install Interactive Voice Response systems (IVRs) in order to reduce CSR (customer service representative) management cost and to provide high-quality service to customer. IVR promotes self-service and reduce agent handle time and helps contact center to drive profitable growth, without driving up costs. Perform better now and in the future. The problem with IVR system is the dynamic nature of services provided by the companies and the needs of various customers necessitate customizing the IVR as per the specific requirements, which require highly skilled developers to make changes to IVR like add new modules, flow scenarios etc. Customizing the modules of IVR for tailor-made applications is a very costly and time consuming task. This work aims at developing dynamic IVR configuration system which prevents the underlying code modification. This proposed system can be used in call centers to enable all kinds of IVR scenarios to be designed, changed, reported, inspected and managed by nonprogrammers. This proposed system supports ever changing requirements of wide variety of companies. And will enable on the fly customization and modification of the IVR modules, doing away with the need of time consuming programmatic modifications.**

***Index Terms*— ACD(Automatic Call Distributor); CC(contact center); Flow Scenario; IVR(Interactive Voice Response); Taskflow; TTS(text to speech); Voice XML.**

I. INTRODUCTION

Now days Contact Center [1] plays vital role in the business because they have a significant impact on customer satisfaction. According to survey about 70% of all business interaction is handled by the Contact Center [1]. In general, most of the management cost for call centers is labor cost for Customer Service Representatives (CSRs), So most of companies use Interactive Voice Response systems (IVRs) in order to not only reduce CSR management cost, but also to provide quality service to customer. IVRs are used to provide self-service to the customer resulting in reducing the burden from the customer service representatives and decrease the number of the agents in the contact center; it may lead to improved customer and agent satisfaction, reduced cost, and increased revenue. IVR is computer telephony integration (CTI) system that allows person to select an option from voice menu [2]. IVR allows providers to create complex menus which the caller can navigate by using touch-tone key presses or speech input. Use of IVR system results in high customer satisfaction and operational effectiveness as it offers a combination of touch-tone input, speech recognition and text-to-speech capabilities [3]. Companies with self-service programs enjoyed an 85% greater year-over-year increase in customer retention rates, compared to those without self-service [7]. But it led to the problems like interaction problem, complicated menu, homogeneous service, poor design of menu and the most important customer felt neglected [3]. The customer felt frustrated and angry due to wide adaptation of IVR system by the contact center. According to the survey the main reason of customer frustration and anger was the poor design of voice-based user interfaces that makes customers feel like the companies just doesn't care about them [5]. As a result of this customer started to avoid the IVR System. And the second major issue with IVR system is that; dynamic nature of services provided by the companies and the needs of various customers require experienced programmers to make changes to IVR system, like add new modules, scenarios and flows, change system parameters etc. Taking the drawbacks that are prevailing in the existing IVR system, we propose a system, which can interact effectively with the customer in a personalized way, In this paper we consider Two-stage call center which handles a call first by an IVR and then by a CSR if the call needs a further assistance after finishing service by an IVR.

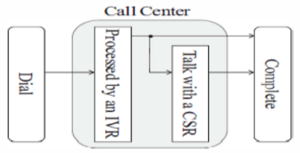

Fig. 1. Two stage IVR System

In order to fulfill changing needs of customers, it is desirable for the service provider to appropriately tailor each IVR service application. Although conventional IVR service enables small changes to the functionality such as recording and changing actual voice announcements, making announcement interruptible or non-interruptible, more significant changes, such as adding and deleting menu items and associated functionality require alteration of the underlying source code of IVR system [4]. Considering dynamic nature of services provided by the companies, experienced programmers needed when it is desired to make modifications to IVR, add new modules, flow scenarios etc. This process is very costly and time consuming. In order to overcome this, we have proposed a system that easily customizes IVR. Proposed system can be used in call centers and enable all kinds of IVR scenarios to be designed changed, reported, inspected and managed by nonprogrammers. System can be configured to utilize a variety of Voice Control features including the play back of recorded announcements and prompts, text to speech utilization and read and right access to ODBC data sources.

II. Architecture

The system consists of many functional units such as; IVR unit, Voice XML Generator, Engine Unit, Recording Unit, Database and Configuration Unit, ACD as shown in Fig. 2. Customer calls come from web portals or voice gateways. Incoming call first hits the IVR unit then IVR unit sends call information to the voice XML generator unit. Voice XML Generator is the unit that gives the response to incoming calls and it creates an output depending on the Flow scenario that is currently activated. IVR unit plays the outputs of Voice XML Generator. Engine unit captures incoming query from Voice XML Generator and detects the called number, selects the corresponding flow scenario and the steps in the flow that will be run. The Database in the system is used to keep scenario information and save scenario usage information. Recording unit is used in the system to enable saving of the IVR scenario steps, which Engine Unit sends to Voice XML Generator, to the Database. The most important unit of the system is the Configuration Unit. This unit is used by contact center administrator or authorized personnel to carry out following operations: i) design the scenario flow that is task flows and IVR flows and working rules; ii) configure system parameters, iii) generate the reports; iv) test the codes to check the validity of flows v) transfer flow scenarios to live operation. After completion of IVR, if customer needs further assistance by agents then ACD (automatic call distributor) comes into picture and it routes the call to the skilled agent.

A. IVR Unit

Incoming call first hits the IVR unit and then becomes operational automatically and assigns a port for the call. Then IVR unit generates a section ID for this call and this generated section ID is sent to Voice XML generator together with other call specific data such as ANI (Automatic number identification) and VDN (Vector Directory Number). Voice XML Generator generates a section for the section ID received from IVR unit. Voice XML Generator asks the required steps for this specific section to Engine Unit.

B. Voice XML Generator

Voice XML Generator generates a dynamic scenario composed of flows that are run by Engine Unit for all types of calls that occurs in the call center. Any call started in IVR Unit sent to Voice XML generator and then to Engine Unit and matched with a saved taskflow scenario. Then, Voice XML Generator unit communicates with Engine Unit until the call ends at IVR unit. Voice XML generator sends call information, caller number (ANI-Automatic Number Identification) information, incoming source point (VDN-Vector Directory Number) to Engine Unit together with call data, if there is any.

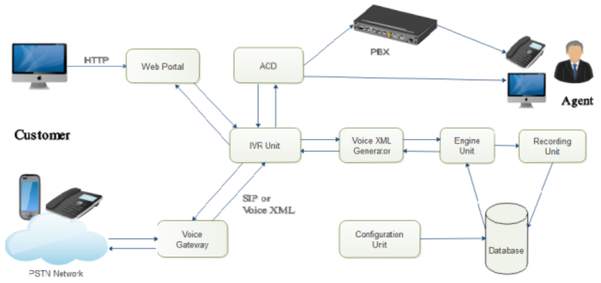

Fig. 2.Architecture of Proposed IVR System

C. Engine Unit

Engine unit observes the call data coming from Voice XML generator and decides which number is entered and which step needs to be followed. Engine unit acquires the information related to the flow scenario from database. It takes the required flow information from database with a query and transfers it to the Voice XML Generator. Engine unit makes a query to the IVR Unit with required action parameters. IVR Unit runs the step and sends back the result to the Voice XML Generator. Engine Unit sends the information of each step to the Recording Unit periodically. Voice XML Generator generates text files (Voice XML outputs) using the data from the Engine Unit. Generated text files are sent to IVR Unit by Voice XML Generator and then IVR Unit plays these Voice XML outputs.

D. Recording Unit

It saves IVR scenario steps, which are sent by the Engine Unit, to the database in sequential order.

E. Configuration Unit

It is a Graphical User Interface (GUI) where users can transact depending on their access rules. Users can also configure all the system parameters such as agents, topics, team, agent profile, announcements, chat script, external destination. User can design scenario flows related to the system, testing of these flows for checking consistency and to take these operations live using this GUI. Configuration Unit also enables users to watch the instantaneous system situation and to take system statistical reports. Configuration Unit is developed to be interactive web based solution. Once configured, the IVR will query and determine which process has been requested from the Contact Centre Server. For example, it can determine which announcements are to be played to a caller. It can determine and provide customers with menu choices from which a selection can be made. It will respond to any customer input values that for example, have been entered from the keypad of a customer’s telephone. It can query an ODBC database to provide Contact Centre Agents with customer details stored within a database record. During the above processes, the call control remains within the configured Call Flow and the IVR script is referenced as an integral part of that call flow. If however the automatic agent feature is utilized the call is no longer under the control of the call flow.

F. ACD

When call does not end with IVR and needs further assistance then ACD (automatic call distributor) routes the call to the specific group or agent depending upon customer selection.

In depth implementation details are beyond the scope of this document hence not included here.

III. Operation

The flow starts with IVR Unit capturing the incoming call to the call center and opening a section for this call and requesting an output form Voice XML generator. Then Voice XML Generator starts a session and sends the session information to the Engine Unit. Engine Unit, depending on the input from Voice XML generator, finds the appropriate flow scenario and runs first step of the flow scenario. Then action items and results of it are written to the database by the Engine Unit. Following that the Engine Unit finds the next action step, repeating process of running a scenario step. Voice XML Generator generates a Voice XML output in the direction of the selected appropriate scenario step, and sends this output to IVR Unit. IVR unit plays this input and checks if the call is still active or not. If the call is active, it is checked, during the play, that the user will make an entry or not. If the user will makes an entry, IVR Unit sends the user entered values to the system. After that IVR Unit requests the following Voice XML output together with system values. If during the check, it is found that there is not any user entry, IVR unit directly requests Voice XML output with the system values. Due to the demand coming from IVR Unit Voice XML Generator request next step from the Engine Unit with system values, and action items. And results are written to the database by Engine Unit. If call needs further assistance after IVR then ACD routes the call to particular group or agent depending upon customer selection. If the call is closed during the Voice XML play, Voice XML generator closes the session and sends the session closed message to the Engine Unit. Engine Unit closes the session.

IV. Ivr Management Through Gui

The most important and distinctive point of the proposed work is that, Nonprogrammers can quickly and easily configure the IVR system using console menus displayed at a graphical user interface (GUI). This configuration unit is web based solution. The GUI enables users to design the flow scenarios and configure system parameters. In addition to that users can validate the flow scenario and take it to live operation means particular flow can be activated using GUI. The GUI platform also allows users to watch the system operation live and take statistical reports from the system. There are number of flow scenarios in Contact Center. These customer or service specific flows direct the underlying IVR application without altering the actual source code of the IVR application. Using GUI, the written flow scenarios can be seen in a window with specific information, definition, modification dates, status and user information and they are editable. User can create a specific flow for a specific scenario. User can configure system parameters as and when require. GUI enables user to create flow by using preset parameters and drag drop option for different widgets. GUI represents created flows scenarios in graphical format for easy reading. Creating a flow is very straightforward and any programming skill is not required. All of the scenarios, flows and system parameters are kept in database in the IVR configuration system so they can be used to create new scenarios or applications by different users. This easy framework allow user to easily configure different IVR scenarios without altering the source code, so the development is reliable and avoids costly and time consuming writing and debugging process. Also this enables the users without any programming skill to configure contact center objects, create flow scenarios without any extensive training. Moreover, the platform has the testing option that users can test their scenarios and flows before activating them. It checks the code and finds rule violations if there is any. The testing module helps users to correct their errors and decrease the time required for a scenario to be live on the system. Flow scenarios can be activated means taken for live operation directly from the GUI.

V. Ivr Reporting

This IVR configuration system allows users to take the current system snapshot and watch the IVR system status and generate statistical reports from the system.Reporting module displays state of contact center objects such as Topic, Agent, Agent group, Team and IVR in real time and update it periodically. Topic reports shows incoming source, dialed number and calls are queued at topic. Topic Reports necessary for service level commitments to customers or provider. Agent Group, Agent and Team reports necessary for tracking agent productivity and internal view of resources and quality of employees. User can create new reports manually or automatically in a variety of layouts and outputs. This helps contact center to collect data about the customer habits or effectiveness of the menus through reporting. System generate certain reports about specific users, call times etc. These data can be used to create more effective or user specific IVRs. In addition to that we have designed a dashboard that gives real time information about the IVR system as shown in following figures. System dashboard gives information about the current call number on the IVR, peak call value, IVR current subscriber segment distribution, distribution of tasks (such as telephony, email and chat), TSF etc. Show key performance statistics e.g. average answer times, lost call rates, etc. This data is used by administrator such that, if he observes a sudden increase in the number of calls, it means that there is a service error. Then he locate the error and send information about the service error to subscribers, such as “the signal is weak due to the weak network or weather conditions” to decrease redundant calls. Moreover, we can get the information on the distribution of the call results such as if the call is abandoned, transferred, or self-serviced. This information give us the ability to act quickly if there is an increase in the abandoned calls, which can be translated into that the scenarios or the system is not effective. We can quickly locate the problematic part, dynamically change the scenario and then activate it. With the help of this feedback from the system, we can take actions and increase the quality of contact center and enhance the experience of our subscribers. Real-time call status and customizable reports help create efficient and productive call centers.

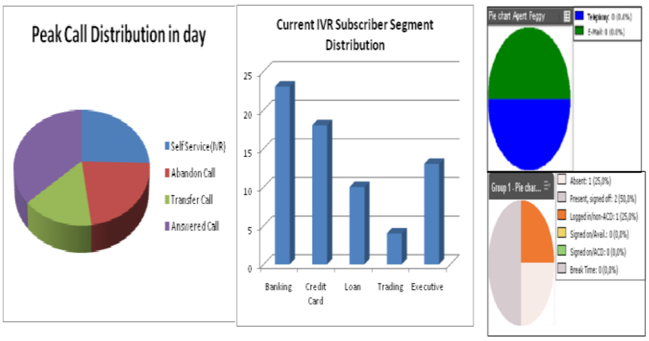

Fig 3: IVR reports

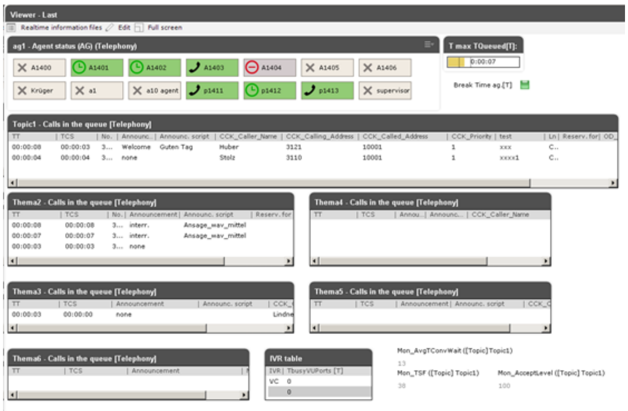
Viewer - Last
Realtime information files
Edit
Full screen
ag1 - Agent status (AG) (Telephony)
A1400
A1401
A1402
A1403
A1404
A1405
A1406
Krüger
a1
a10 agent
p1411
p1412
p1413
supervisor
T max TQueued[T]:
0:00:07
Break Time ag.[T]
Topic1 - Calls in the queue [Telephony]
Thema2 - Calls in the queue [Telephony]
Thema4 - Calls in the queue [Telephony]
Thema3 - Calls in the queue [Telephony]
Thema5 - Calls in the queue [Telephony]
Thema6 - Calls in the queue [Telephony]
IVR table
Mon_AvgTConvWait ([Topic] Topic1)
Mon_TSF ([Topic] Topic1)
Mon_AcceptLevel ([Topic] Topic1)

Fig 4: Agent Real-time Information

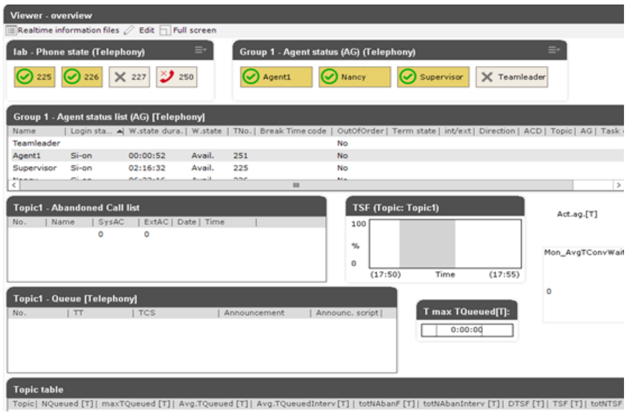
Viewer - overview
Realtime information files
Edit
Full screen
lab - Phone state (Telephony)
225
226
227
250
Group 1 - Agent status (AG) (Telephony)
Agent1
Nancy
Supervisor
Teamleader
Group 1 - Agent status list (AG) [Telephony]
Topic1 - Abandoned Call list
TSF (Topic: Topic1)
Act.ag.[T]
Mon_AvgTConvWait
Topic1 - Queue [Telephony]
T max TQueued[T]:
0:00:00
Topic table

Fig 5: Historical Reporting

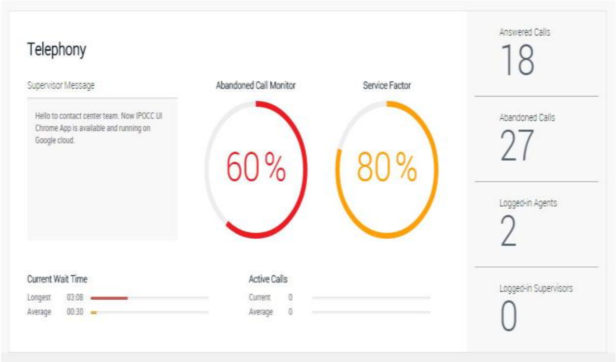

Fig 6: Agent Telephony Screen

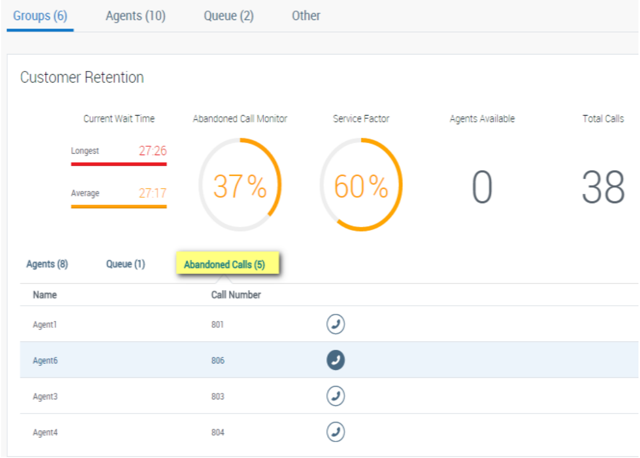

Fig 7: Supervisor – Group View Real Time Functionality

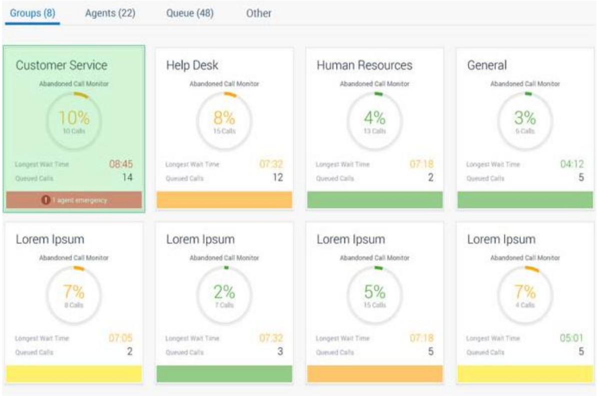

Fig 8: Supervisor – Group View Real Time Functionality

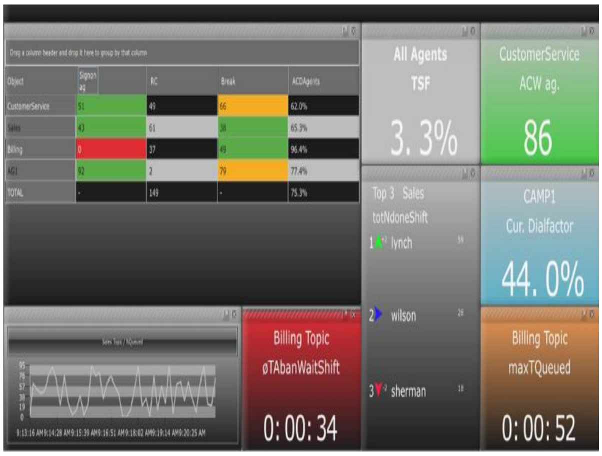

Fig 9: Contact Center Wallboard

VI. Conclusion

This IVR configuration system is developed to supports dynamic interactive voice response platform and used in call centers. Developed IVR system enables users to design new user specific application by using pre-defined widgets through a GUI without modifying underline source code. The platform can also be used to quickly validate the created flow scenario and to take it to the live operation. Considering the various needs of customers and dynamic nature of service, this system results in time and cost saving. Another aspect of the developed platform is that it facilitates gathering useful data about customer habits or usage statistics which can be used for development of user or application specific IVR scenarios and ultimately enhance the quality of contact center. The system can be deployed on cloud as future work. This cloud deployed IVR configuration platform provides enough APIs to develop application equivalent to web access.

ACKNOWLEDGMENT

Authors acknowledge the support of Avaya India Pvt. Ltd., Vinit Rewatkar and Yogesh Sonawane for providing valuable inputs and suggestions for execution of this project.

REFERENCES

[1] K. Dawson, "The Call Center Handbook: The Complete Guide to Starting, Running, and Improving Your Call Center." 5th ed. CMPBooks, San Francisco,CA. 2004.

[2] S. S-Sanz, M.Naldi, A. M.P-Bellido, J. A. P-Figueras and E. G. OGarcia, "Evolutionary Optimization of Service Times in Interactive Voice Response Systems" IEEE Tran. On Evolutionary Computation,vol. 14, 4, p. 602-616, 2010.

[3] P. Khudyakov, P. Feigin and A. Mandelbaum, "Designing a Call Center with an IVR" *Journal of Queueing Systems*. Vol.66, 3, pp 215-237, 2010.

[4] Ramazan Karademir and Emre Heves, "Dynamic Interactive Voice Response (IVR) Platform"EuroCon 1-4 July 2013, Zagreb, Croatia

[5] Mudili Soujanya and Sarun Kumar "Personalized IVR system in Contact Center" 2010 International Conference on Electronics and Information Engineering (ICEIE 2010)

[6] Mr. Ritesh Chauhan and Mr. Vivek Joshi "A Comprehensive Study of Design, Development andImplementation of an Automated IVR Systems"IRACST - International Journal of Computer Science and Information Technology & Security (IJCSITS), ISSN: 2249-9555Vol. 2, No.6, December 2012

[7] "Why your customers hate your ivr systems?" Aspect Communications.2003. Available at http://www.aspect.com/mm/pdf/products/interactive

First Author Suvarna Ramchandra Chavan
M.Tech (2nd year) Computer Engineering, College of Engineering Pune
Maharashtra, India.
Second Author Dr.Y.V.Haribhakta (*Author*)
Computer Engineering, College of Engineering Pune
Maharashtra, India.