European Journal of Scientific Research
ISSN 1450-216X Vol.23 No.3 (2008), pp.436-445

http://www.eurojournals.com/ejsr.htm

Implementation of a Multi-Channel Application for Customer Care Service Using Best-First Search Algorithm

Adeyeye O. M
Department of Electrical Engineering, University of Cape Town, South Africa
E-mail: micadeyeye@crg.ee.uct.ac.za

Atayero A. A
Department of Electrical & Information Engineering
Covenant University, Ota, Nigeria
E-mail: atayero@ieee.org
Tel: +234.805.256.7491

Abstract

It has become imperative to find a solution to the dissatisfaction in response by mobile service providers when interacting with their customer care centres. Problems faced with Human to Human Interaction (H2H) between customer care centres and their customers include delayed response time, inconsistent solutions to questions or enquires and lack of dedicated access channels for interaction with customer care centres in some cases. This paper reports the implementation of a framework and presents development techniques for a multi-channel application providing Human to System (H2S) interaction for customer care centre of a mobile telecommunication provider. The implemented solution is called Interactive Customer Service Agent (ICSA). Based on single authoring, it provides three media of interaction with the customer care centre of a mobile telecommunication operator: voice, phone and web browsing. A mathematical search technique called Best-First Search is employed to generate accurate results in the search environment.

Keywords: Artificial intelligence, Computer Languages, Internet, Multi-access communication.

1. Introduction

With the present state of development in telecommunication in terms of bandwidth and speed, people now demand for various Internet services via their mobile devices. Besides, they long for a more convenient way of using the web through voice interaction. Owing to this global state of technological advancement, industries have ceaselessly continued to satisfy their customers through astounding applications. Typically, banks and mobile operators now provide different means of rendering services from human to human interaction (H2H) to human to system (H2S) interaction.

This paper has been segmented into five sections. Section 2 gives brief and concise information on WEB, Wireless Application Protocol (WAP) and VOICE architectures with an integrated diagram of the architectures. In addition, it presents information on multimodal and multi-channel applications, single authoring and multiple authoring, and the Best-first search technique. Section 3 examines design

issues and practices. It describes how xml transformation is done and the N-tier models. In section 4, the framework is emphasized by indicating the features of each layer of the implemented N-tier model. Section 5 explains how the proposed framework will operate with the aid of a Use Case diagram and our conclusion is given in section 6.

2. Background Theory

2.1. WEB Architecture

The World Wide Web operates on Hypertext Transfer Protocol (HTTP). This is a client/server architecture whereby the server resides at one end and serves web pages to client at another end. A browser resides on the client and is used to interpret Hypertext Markup Language (HTML) codes passed by the server. Web pages or files are classified into static and dynamic. The static pages are fully interpreted at the client end while dynamic pages are executed at the server end and results passed to client. An application server carries out such processing, while a web server acts as a container or storage for the web documents or files with extensions like. html,. asp, and. jpg, database server stores data like images in binary format, figures, and texts.

2.2. WAP Architecture

Internet access over mobile devices requires a packet switching network which is obtainable today with the emergence of technologies like General Packet Radio Service (GPRS), Enhanced Data for GSM Environment (EDGE) and many more. Where this network is in place, a WAP gateway is required to interface with the existing Internet. The WAP gateway is also connected to a Base Transceiver Station (BTS) that provides wireless connection to the mobile devices referred to as clients. The lightweight protocol in use is called WAP [1]. While a browser such as Internet Explorer (IE) ™ or Netscape Navigator is required to interpret HTML codes passed to a Personal Computer (PC), microbrowser like Openwave™ or Nokia™ is required to interpret Wireless Markup Language (WML) codes passed to mobile phones.

2.3. VOICE Architecture

In voice architecture an application resides on a web server. Voice recognition is performed by a Automatic Speech Recognition (ASR) server and text to audio conversion is performed by Text-To-Speech (TTS) server. The VoiceXML interpreter executes the application according to the VoiceXML specification [2, 3, 16]. There are two common ways of implementing voice interaction namely directed dialogue and mixed initiative or Interactive Voice Response (IVR). The prevalent of the two is directed dialogue in which a user is replied by a system or application that gives a set of instructions in voice format. The application interacts with users based on Dual Tone Multiple Frequency (DTMF) pulses generated on pressing the phone key(s). In this case, speech synthesis and voice recognition are not involved unlike IVR, which requires that the application captures the voice of the user and intelligently interacts based on keywords or words that match available grammars. Grammars are set of words, which are matched with the spoken words supplied by the user. Where one or more words match, the application executes a set of predetermined instructions based on its algorithm. IVR applications are based on programming languages like Voice Extensible Mark-up Language (VXML), and Speech Application Language Tags (SALT).

Figure 1: WEB, WAP and Voice Integrated Architecture

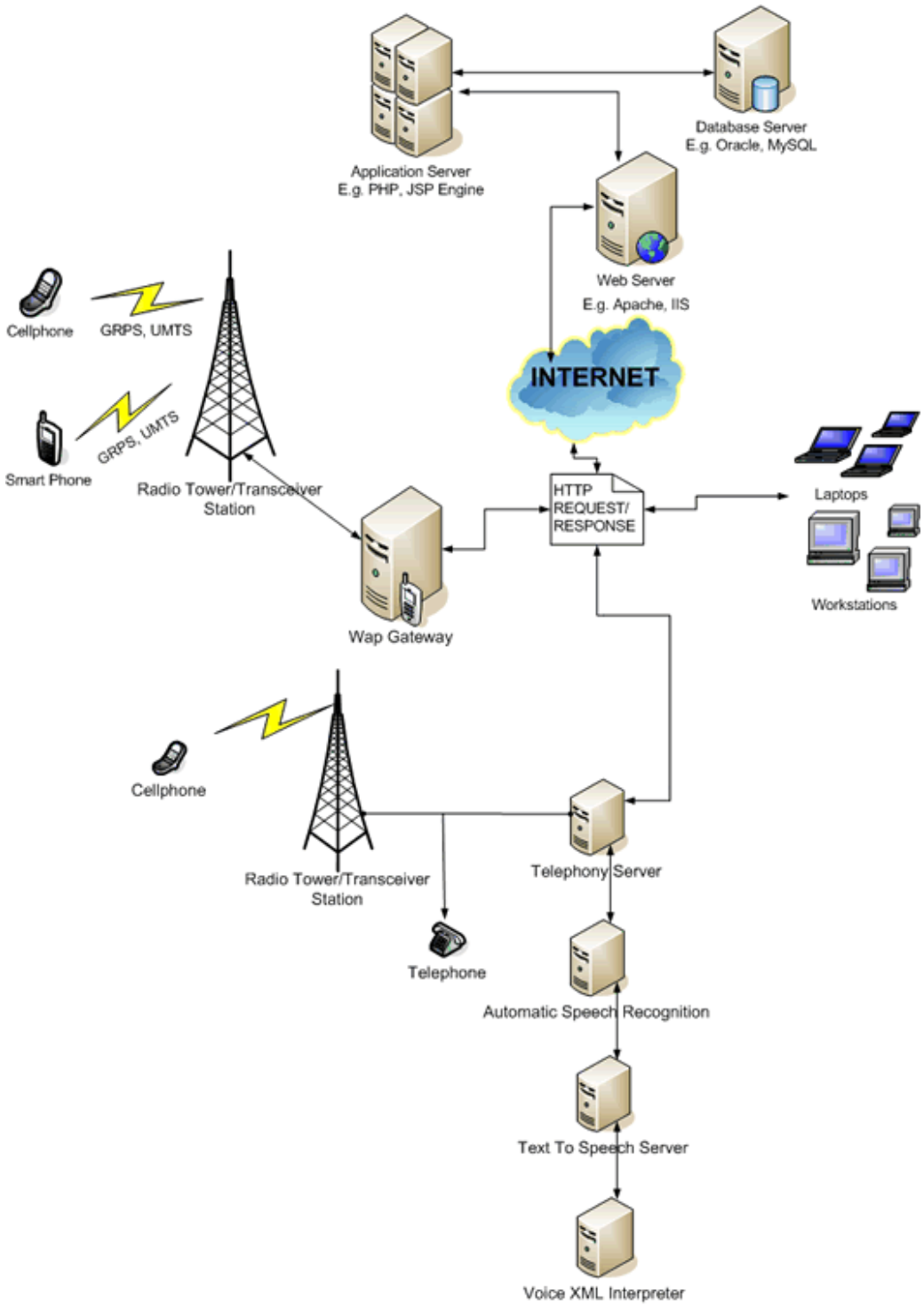


Fig.1 shows an integrated architecture that encompasses the WEB, WAP and VOICE networks. While PCs directly interpret HTTP request/response, mobile devices rely on the light-weight protocol known as WAP to present the information in compiled or binary format [1]. This is carried out through a gateway called WAP Gateway. For voice based interaction, the voice gateway may be made up of the telephony server, ASR server, TTS sever, and the VoiceXML Interpreter.

2.4. Multimodal Application vs. Multi-channel Application

Multi-channel access is the ability to access enterprise data and applications through multiple channels while multimodal access is the ability to combine multiple channels in the same interaction or session [3, 4]. X+V (XML+VXML) and SALT (Speech Application Language Tags) are the common tools for developing multimodal applications. While X+V focuses on combining web and telephony applications, SALT aims at enriching web applications with speech/telephony capabilities and turning them to multimodal. Also, SALT applications offer tightly-coupled multimodal browsing capability while X+V applications offer loosely coupled multimodal browsing in a more complete and structured manner.

Multimodal web applications are associated with many research challenges, mostly relating to multimodal interaction. One general usability requirement is to provide a highly improved experience to the end-user through robust, user-friendly applications [5, 6]. To this end, architectures capable of synchronizing and supporting voice and data simultaneously and seamlessly are required unlike Multi-channel where supporting data and voice are based on user's preference.

Network and computational requirements need to be taken into account in Multi-channel and Multimodal applications. An issue affecting the client devices is the presentation of different content formats enabling the various modalities, for both fixed and mobile devices. There are also issues relating to authoring and reusability of server-side components, application design, as well as the need for dealing with access and transformation of content at the system's back-end [5].

2.5. Multiple Authoring and Single Authoring

Multiple authoring provides simplicity because it relies on multiple representations of the target data for the supported modalities. Multiple authoring means that each modality has its own presentation tier. From an application developer's point of view, in a multiple-authoring approach each modality is served from a separate server or web directory [5].

A single-authoring approach reuses common blocks of the different presentation tiers, toward a common presentation tier for all modalities. This is based on generating and handling multimodal content through appropriate processing of user requests, content acquisition and content transformation according to the various modalities.

Key benefits of multiple authoring are that it facilitates content acquisition, minimizes content transformations and ease of extending existing content (e.g. HTML) to multimodal content. Scalability problems arise mainly because of redundant authoring, which is alleviated by single authoring. Single-authoring approach keeps content management tasks simpler and naturally tailors the application to multi-architecture [5].

2.6. The Search Technique

Artificial Intelligence (AI) is the study of how to make computers do things, which, at the moment, people can do better [7]. Common AI techniques include question answering, generate and test, breadth-first search, deep-first search, best-first search and many more. We now consider the Best-First Search technique in detail.

2.6.1. Best-First Search

Best-first search is a way of combining the advantages of both depth-first and breadth-first search into a single method. Depth-first search is good because it allows a solution to be found without the need for expanding all competing branches, while Breadth-first search is good because it does not get trapped on dead-end paths [7].

2.6.2. Capabilities of Best-First Search Technique

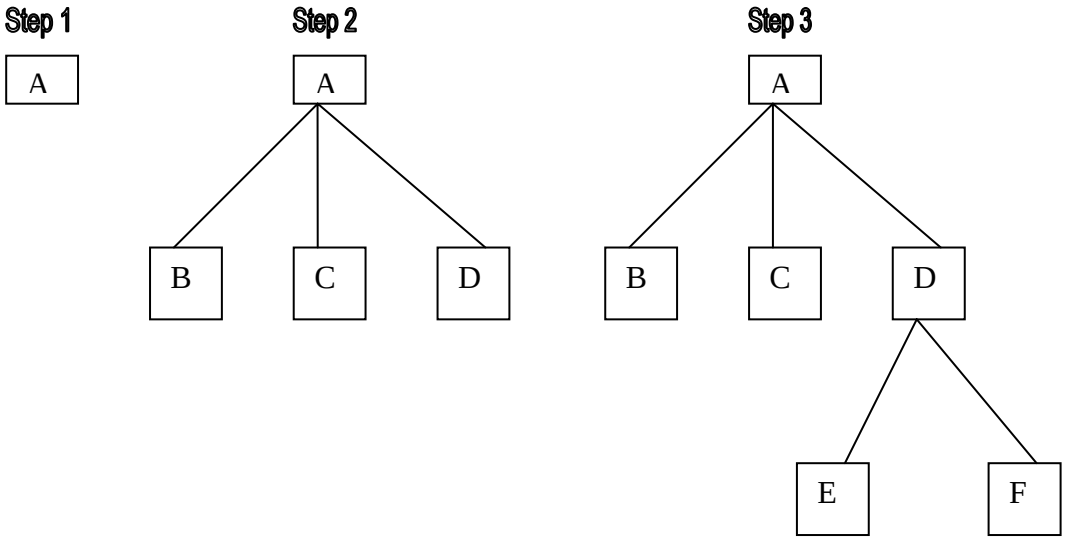
By combining the merits of Depth-First Search and Breadth-First Search techniques, it has less memory usage (an advantage of Deep-First Search). Elaborately, it examines a node (domain of likely answers) only when an answer has not been found as shown in Figure 2. Because of this, the time taken to perform a search will be reduced. Agreeably, the processing speed of a query will be high. In addition, it will provide a solution if it exists. That is, it would not get trapped or terminated abruptly (an advantage of Breadth-First Search).

To implement this search technique, it will be required that existing answers are grouped into domains. Also, the system must intelligently narrow question asked to very few domains of possible answers.

2.6.3. Best-First Search Tree

- Start with OPEN containing just the initial state.
- Until a goal is found or there are no nodes left on OPEN do:
    - Pick the best node on OPEN
    - Generate its successors
    - For each successor do:
        - If it has not been generated before, evaluate it, add it to OPEN, and record its parent.
        - If it has been generated before, change the parent if this new path is better than the previous one. In that case, update the cost of getting to this node and to any successors that this node may already have.

Figure 2: A Best-First Search Tree [7]



3. Design Issues and Practices

With a goal of developing a multi-channel application using single-authoring approach, basic design issues relating to n-tier application are considered. These design issues and practices include the XML/XSL Transformation and different layers in N-tier models. The attendant benefits include ease of content management, and scalability to mention a few.

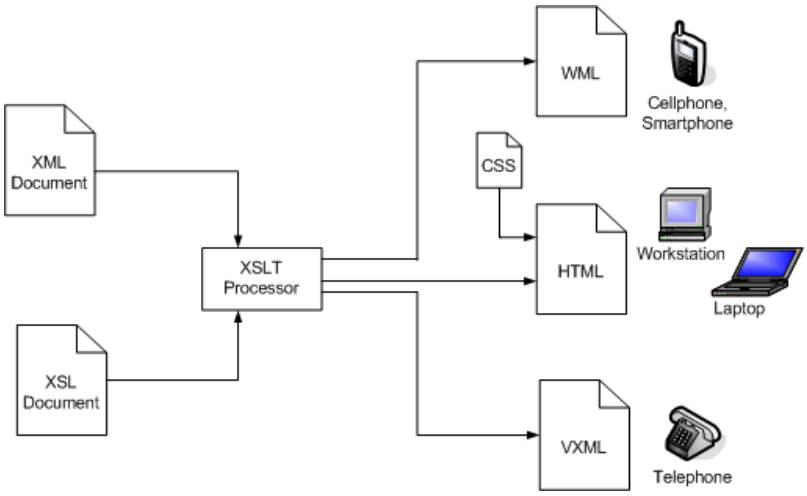
3.1. XML/XSL Transformation

Extensible Mark-up Language (XML) is a meta-markup language. It is a set of rules for creating semantic tags used to describe data. While HTML is used to specify the layout of a web page, XML is

used to describe data [8, 9]. Extensible Stylesheet Language (XSL) is an xml-based language that can be used to manipulate, sort, and filter XML data. XSL language has been further split into three parts, namely a) Transformation (XSLT) b) Rendition (XSLF) and c) XPATH

Extensible Stylesheet Language Transformations (XSLT) treats the document to be transformed as a set of nodes. An XSLT Stylesheet defines a set of rules or templates. When a template matches one of the nodes in the source document, it results in storing the output structure given by the template in a new document. XSLT uses the World Wide Web Consortium (W3C) XPATH specification language to query XML data. World Wide Web was created in October 1994 to develop common protocols that promote the evolution of the web and ensure its interoperability [10]. XPATH is strongly analogous to SQL and lets one specify complex rules to match nodes in a document [11]. A multi-channel application presents its content to the end users based on their connecting devices [5, 12]. XSL is an ideal tool for separating content and presentation. It can be used to overcome the inherent difficulties associated with the presentation of data to devices with limited capabilities as shown in Figure 3.

Figure 3: XML/XSL Transformation to HTML, WML and VXML

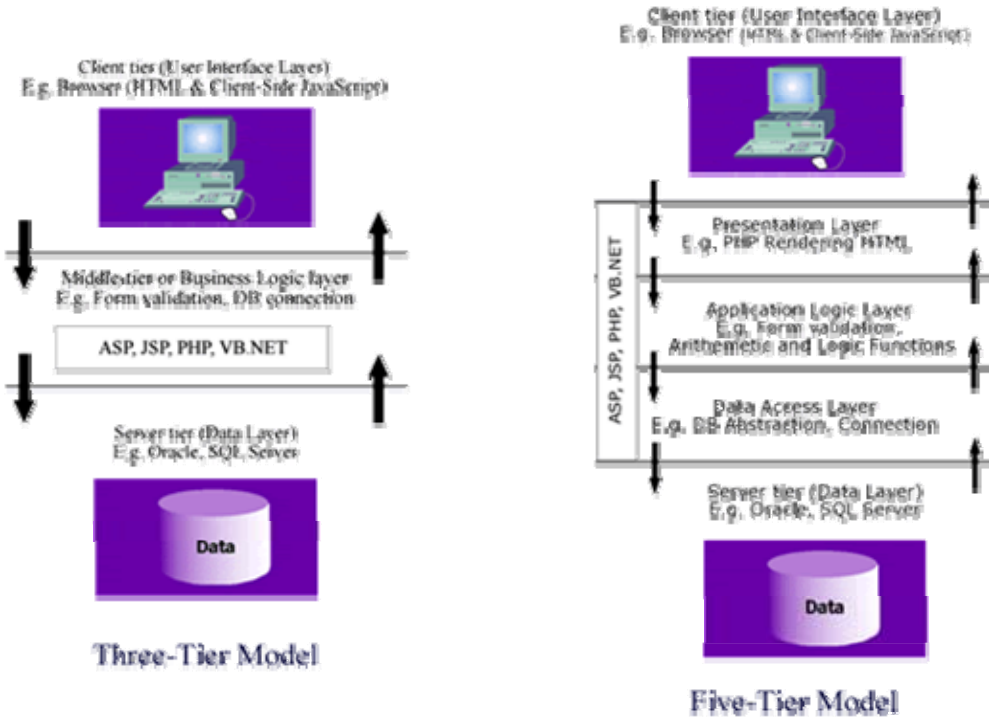


3.2. N-tier models

N-tier refers to the number of layers, into which a whole package or application can be classified. Better still, it can be referred to as Client-Server model where N is greater than or equal to 2 [13]. Web application would perform better if developed with a five-tier model in mine rather than three-tier model. Five-tier model is an expansion on three-tier model as shown in Figure 4

Key benefits of N-tier applications include scalability to cope with future traffic and performance demand; and design of well structured, flexible, and vendor-neutral applications that are very easy to maintain [13].

Figure 4: 3-Tier and 5-tier models [13]



4. ICSA Implementation Framework

A framework for a multi-channel application called ICSA is introduced here. It employs a five-tier model as shown in Figure 5 and described below.

4.1. Data-Layer

It represents the database and web servers. Database server contains keywords to be looked for when question(s) are asked, appropriate or similar questions and possible answers to be presented.

4.2. Data Access Layer

This contains a server-side include file or script that connects to the database. It establishes a connection to the database and provides a connection identifier for query purposes. An Application Program Interface (API) that converts the entire database into xml files can also operate at this layer.

4.3. Application Layer

It is responsible for the arithmetic and logical operations that are performed with the application. It includes searching for keywords and storing new set of questions when answers can not be provided. An efficient search technique called Best-First Search is used to query the database.

Where XML files represent the database, XPATH (similar to SQL) can be used to locate answers within the XML files. Also, a third party API can be used to perform the search in XML files. A server-side script generates an xml file on the fly where the query results are stored.

4.4. Presentation Layer

This layer contains extensible Stylesheet files that can format the on-the-fly document into WML, HTML, and VXML files depending on the device or browser used to access the application. The

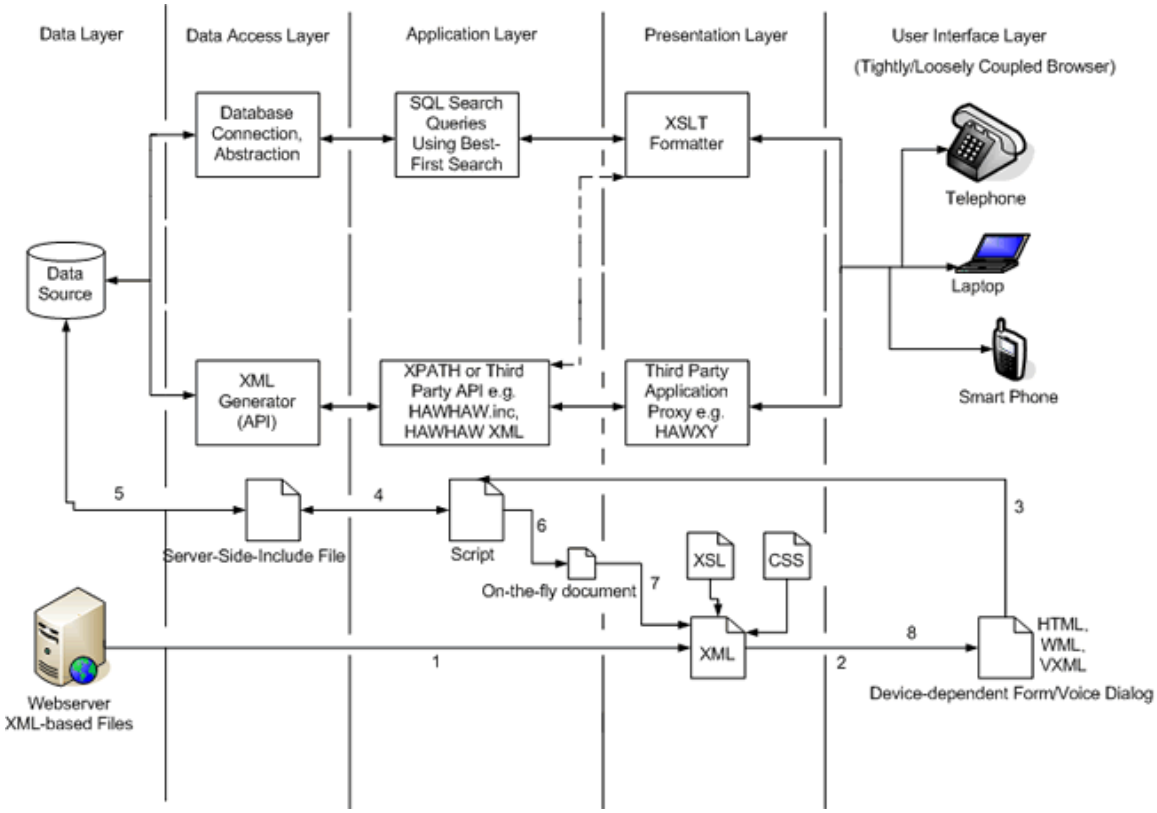
appropriate format is determined by examining the HTTP request made by the browser. Alternatively, a third party program like HAWXY can act as a proxy, which examines the user's connecting device or browser and presents appropriate message to it.

4.5. User Interface Layer

It features different browsers or connecting devices. Where telephone or mobile phone is used, the application converts its message to voice with the aid of TTS server. In case of phone browsing, the browser feeds on WML files while in web browsing, the PC feeds on HTML files additionally formatted by Cascading Stylesheet (CSS).

The flow process is also shown in Figure 5. When a user connects to the application either by dialing a number or typing in a web address, the index file on the webserver is passed. As an xml file, information in the header request determines what XSL file should be used to present the response as HTML form, WML form or voice dialogue. On submission, a script queries the database and generates an on-the-fly xml document. The xml document is styled by XSL, and if need be by CSS also before presenting the response to user.

Figure 5: The ICSA Framework

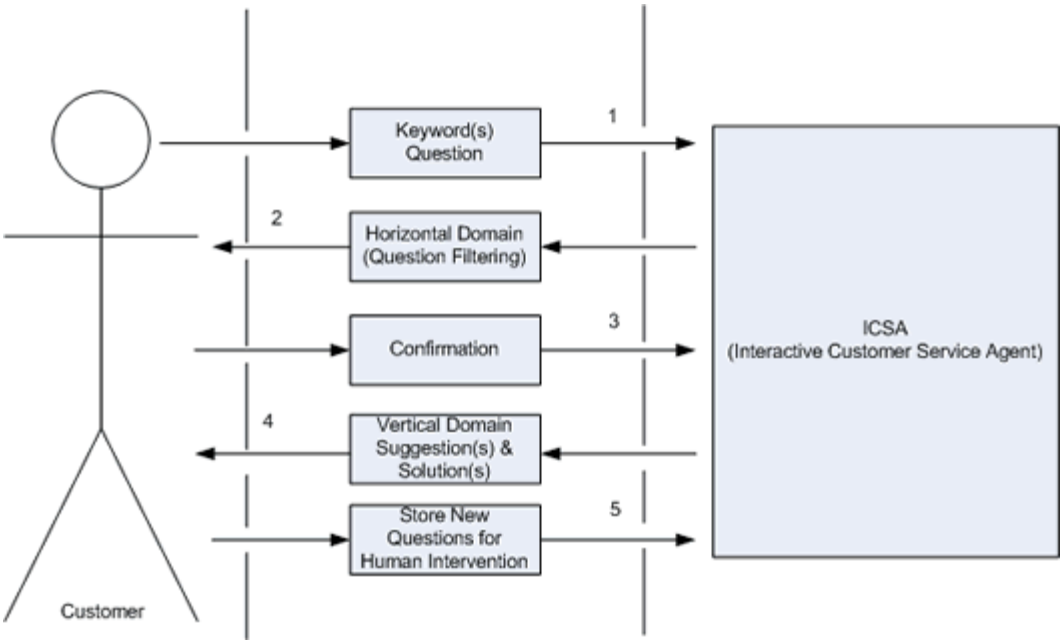


This framework will require few XML files; separate XSL files for WEB, WAP, and VOICE presentations and server-side scripting or application files to implement. It will provide a robust web application with ease of content management. In addition, it becomes easy to manipulate the xml files with the use of XSL. The framework gives room for scalability in order to add new features or extend it functionalities.

5. Discussion

The use case diagram shown in Figure 6 pinpoints five steps involved in answering queries by the ICSA. A customer types in a question or a short sentence indicating his problem. On submission, the agent filters out the keywords, queries the database, and presents similar questions for the customer to pick the most similar to his. This first level of interaction helps in isolating the problem by searching for keyword(s) over a wide area (that is matching keywords in the entire database). This is referred to as horizontal domain sorting [14]. A confirmatory message is expected from the customer either by clicking a link (that is, the most similar question) or a voice response [15]. Thereafter, a thorough search called vertical domain sorting is carried out. This is an extensive knowledge on a narrow subject having isolated the problem from others. The most appropriate answer is presented. Should no answer exist, the question is stored for human response and a call is forwarded to a human or a message popup on a system at the customer service centre [2].

Figure 6: Use Case Diagram of the application



6. Conclusion

The framework will produce a five-tier web application, which is vendor-neutral, platform independent, scalable enough for future expansion, and robust. More so, it is a single-authoring programming paradigm using XML and can be implemented by coding in any server-side web language that supports XSLT. It can operate as a DTMF-based application over an analogue telephone network or IVR-based application over an IP-based or digital telephone network needless to say it provides three dedicated channels for customer interaction: phone, web and voice browsing.

Inherently, the application becomes reusable in future technologies or generations of fixed and mobile telecommunication. It will improve Customer Relationship Management (CRM) in telecommunication industry and any other firm that require 24/7/365 support for their invaluable services. It is germane to conclude that this proposition can boost productivity and sales for any business to consumer (B2C) model industries that implement this framework.

References

[1] Chris Tull, WAP 2.0 Development, Que Publication, USA, 1st Edition, pp 11-42, 2002.

[2] Atayero, A. A., Olugbara, O. O., Ayo, C. K., Ikhu-Omoregbe, N. A., "Design, Development and Deployment of an Automated Speech-controlled Customer Care Service System," *Proc. Of the GSPx 2004*, Santa Clara, CA, September 27-30, 2004

[3] Vasiliki Demesticha, Jaroslav Gergic, Jan Kleindienst, Marion Mast, Lazaros Polymenakos, Henrik Schulz, L. Seredi: "Aspects of Design and Implementation of a Multi-channel and Multimodal Information System," *Proc. of the IEEE Conference on Software Maintenance*, Florence, Italy (ICSM 2001), pp. 312-319, November, 6-10, 2001.

[4] Scott A., "The rise of the Multimodal Web," *Proc. of the W3C/WAP Workshop: The Multimodal Web*, Hong Kong, 2000.

[5] Lazarous C.P. and John K. S.(2005), "Multimodal web applications: Design issues and an implementation framework", International Journal for Web Engineering and Technology, Vol. 2, No. 1. 2005.

[6] Ernest Chang, "The Design of a Secure and Pervasive Multimodal Web System," *Proc. Of the 19th IEEE Conference on Advanced Information Networking and Applications*, 2005.

[7] Elaine R. and Kevin K., Artificial Intelligence, McGrawHill, Singapore, International Edition, pp. 3-98, 1991.

[8] Ellen Pearlman and Eileen Mullin, Programming the web using XML, New York, McGraw-Hill, 1st Edition, pp 1-9, 2004.

[9] Richard Light, Presenting XML, USA, Sams.net, 1st Edition, pp. 2, 202-209, 1997.

[10] Max Froumentin, 'The World Wide Web Consortium,' Speech Technology, March/April 2005.

[11] Yannis Despotopoulos, John Soldatos, George Patikis, Lazarous Polymenakos, Jan Kleindienst and Jaroslav Gergie: "Accessing and Transforming Dynamic Content based on XML: Alternative Techniques and a Practical Implementation", *Proc. of the third international conference on Information Integration and Web based Applications and Services*, Linz Austria, 10-12 Sept. 2001.

[12] M. Satyanarayanan, "Integrated Pervasive Computing Environments," *IEEE Pervasive Computing*, 1(2), 2002, pp. 2-3.

[13] Adeyeye Michael, "Design and Implementation of a distributed Web-based application," Journal of Computer Science and its Applications, Vol. 12, No. 1. 2006.

[14] Boris Galitsky and Rajesh Pampapathi, "Can many agents answer questions better than one?," First Monday, Vol. 10, No. 1, 2005.

[15] OnOn Hong, VoiceXML Application Development Life Cycle: Building Usable Voice User Interface, http://devresource.hp.com/drc/resources/vxusability/vx_devlifecycle.pdf, Accessed: April, 2005.

[16] William A. Yarberry, Jr., Computer Telephony Integration, Averbach Publication, USA, 2nd Edition, pp 1-107, 2003.