

Multimodal Applications for the Mobile Generation

Bogdan Blaszcak
Director of Enterprise
Product Management,
Intervoice, Inc.

i
n
t
e

Multimodal Applications for the Mobile Generation

Bogdan Blaszczak

Director of Enterprise Product Management,
Intervoice, Inc.

a
c
ti
ve

1. Don't Call Me a Phone

The number of mobile phone users continues to grow steadily. According to International Telecommunication Union (ITU), there were over 2 billion mobile subscribers in 2005, or approximately 30 percent of the world's population.

Beyond the sheer numbers, the other fascinating aspect is the evolution of the phone. It can still handle calls, of course, but even the low-end phones now come with many additional features. It all started with SMS (Short Message Service), which still dominates mobile messaging despite its extreme limitations. Now phones are advertised as multimedia mobile computers and provide various connectivity, productivity and entertainment functions.

The cellular networks also have been evolving. The 3G networks provide bitrates that are almost as fast as your home Internet connection. This makes

browsing and other data-intensive functions quite usable. IDC predicts that 1.3 billion people will connect to the Internet via mobile phones by 2008.

However, the key capability that inspired this paper is simultaneous voice and data connectivity. You can enjoy it on the new 3G GSM networks - if you have a 3G phone. This is not yet the case on the 3G CDMA networks, but their high data speeds with low latency make them a perfect candidate for "Voice over IP" that will enable the same result.

2. Let's Talk

There are essentially three modes of user interactions supported by the current mobile phones. Besides voice, of course, you can use the phone keypad and usually have some way of "pointing and clicking." Some recent announcements promise to add handwriting recognition, screen touch gestures, or even physical gestures with the phone.

A large segment of mobile users can easily create text messages through multi-key sequences on the phone keypad. A few of the new phones add a text keyboard that makes the task more straightforward. However, the keys remain tiny and are tricky to use... and then there also is "BlackBerry Thumb" to worry about.

The inefficiencies of text entry on the mobile devices become apparent when the text must conform to normal spelling and when it has to be complete. This is usually the case in web searches or financial transactions.

Wouldn't you rather say what you want instead of struggling with the keys?

The support for simultaneous voice and data connections makes multimodal interactions effective. The use of both modalities in an interaction is not new, of course. However, the previous level of mobile technology only lets one modality remain active at a time. For example, you could not have an active data connection while talking. This limitation resulted in a "choppy" user interface and a less than satisfactory experience.

The previous attempts to improve the multimodal experience depended either on SMS messaging or on voice recognition performed on the device.

The SMS delivery does not interfere with the voice channel. Hence, a device-based client can receive an SMS during a voice call and present a menu based on the message content. This solution has been successfully deployed but SMS limitations prevent it from supporting rich GUIs (graphical user interfaces) that modern devices can render. The SMS delivery also may introduce delays long enough to make the interactions tiresome.

Device-based voice recognition has been another approach to adding voice modality. Your voice is digitized and processed on the device. The results are then sent to the server over an established data connection. This solution is quite feasible for simple commands or search queries. However, an additional recognition client must be installed, so the device must support the required capabilities and provide the sufficient processing power. Also, the nature of this approach requires an explicit query/response flow of the conversation that falls short of state-of-the-art IVR (interactive voice response) capabilities.

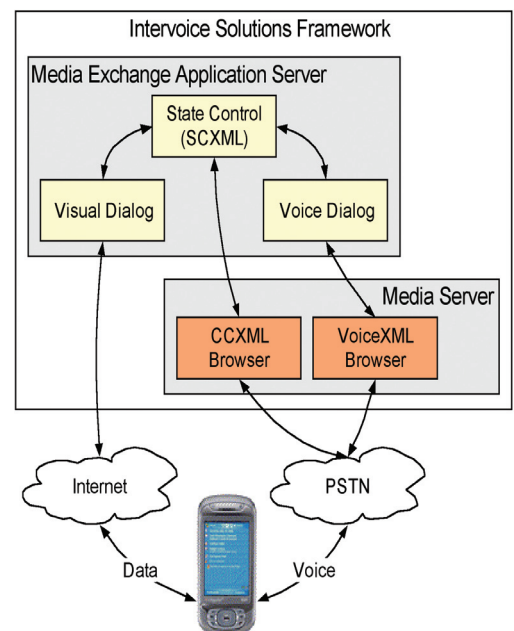
3. Multimodal Interactions

The new support for simultaneous voice and data connectivity on some mobile networks presents an opportunity to provide a robust, multimodal experience to mobile users.

Truly multimodal applications should allow the user to respond through the interface most suitable for the step and the context of the interaction. Let's consider an application that has voice as well as screen and keyboard as interfaces. The user will usually favor the most effective interface for the step. For example, the user may make a voice request to avoid excessive typing. Then the user may "click" on a list or map on the screen instead of listening to long descriptions of this available options. However, the context of the interaction may dramatically influence the choice of interfaces. The user may be reluctant to make voice requests while in a crowded place and the typing may be an appreciated alternative. On the other hand, it is usually difficult to focus on a screen while walking. While driving, the use of a keypad may be unlawful and it should generally be avoided - a voice request is the better choice.

The key to the multimodal interactions is the effective synchronization of supported modalities. Within the Intervice Solutions Framework (ISF), the State Control element of Media Exchange provides an effective application environment for multimodal applications. Media Exchange is a server-based application execution environment based on industry standards. Besides the application engine, it also provides media management, administration and reporting services. The corresponding application development tools plug into the Eclipse framework.

The high level architecture of the multimodal application is shown in the following diagram.



The two main components required to run multimodal applications are the Media Server and the Media Exchange platform. The Media Server interfaces to the phone network and executes the call control and voice user interface (VUI) scripts provided by the Media Exchange. The scripts are encoded in CCXML and VoiceXML, respectively. Both scripting languages are standards of the W3C Voice Working Group.

The task of the Media Exchange is to support and synchronize the operations of its three clients. The first two are the call control and voice user interface browsers. The third one is a graphical client running on the mobile device. Most of the implementations in the past required this to be a custom client. However, some of the new phones provide support for dynamic web content. With these capabilities, the visual presentation can be rendered in direct response to server-side events and logic. This is the same technology that Google Maps uses to provide a responsive and efficient interface with the Internet browsers on the desktops.

The synchronization between the modalities is implemented using State Chart XML (SCXML) running in the State Control element of the Media Exchange platform. The SCXML language is a W3C standard for defining state machines in an efficient way. In our case, the state charts represent application core logic that is independent of any user interaction modes. The mode related details are taken care of by the additional dialog elements as shown in the architecture diagram. This architecture enhances maintainability and expandability of the overall application. For example, dialogs can be added or customized independently from the state control and other dialogs.

The visual and voice dialogs are responsible for providing effective user interfaces for their respective modalities. The visual dialog is based on the established technologies of dynamic web applications and AJAX (Asynchronous JavaScript and XML). The voice dialog can leverage the full power of VoiceXML, including the voice recognition and text-to-speech provided by the servers. This is the same technology that Intervoice has been successfully deploying for voice-only interactions. Its use in multimodal applications enables us to

continue providing the expected high quality of voice user interfaces. The server based voice recognition can effectively leverage new technologies without requiring updates to device software.

The multimodal applications built with Media Exchange also can benefit from all the other technologies that ISF provides. For example, application personalization can significantly improve the caller's effectiveness and satisfaction. This could be further enhanced by incorporating the presence and location factors in the personalization rules. The potential mobility of the caller is a new dimension with its own specific challenges and opportunities.

4. Application Design

Well-designed IVR applications successfully provide critical services and round the clock access for customers. The improvements in voice recognition and language understanding technologies enable a fluid conversation flow instead of the extensive cascading menus of older IVRs. However, there are still few callers who find current IVR applications exciting.

The multimodality, the graphical rendering, and the high speed data connectivity of the new mobile devices enable designers to add a "wow!" factor to their applications. The visual communication engages new senses and dramatically enhances the information flow. Given the rich possibilities and user experience delivered by these new technologies, we may even see more users who prefer to interact with self-service applications rather than agents. It is now quite feasible to provide mobile applications within domains that used to be constrained to the desktops. It also makes it possible to build social networks that are so popular on the web, like multiplayer online games, or virtual world interactions.

Will your next banking application look like a game? I wouldn't bet on it. However, a well thought out combination of voice and visual interfaces will make the caller experience more efficient and satisfying.

Needless to say, the technology to deliver simultaneous multimodal interactions does not eliminate the need for good interaction design. The key to the successful design of a multimodal application for mobile users is in the flexibility of the flow and interactions. The following list outlines several of the design considerations.

- **Choice of Modality**
Where possible, the user should always be given a choice of interaction modes. The design should not force the use of one modality nor require the use of all modalities. The user choices will be driven by the information presented, the user's preferences, skills, and the environment. The environment and situational factors may exacerbate any user interface shortcomings and prevent the user from completing the interactions.
- **Best User Interface per Modality**
The design for each modality should fully leverage the corresponding technical capabilities and provide the best interface possible. For example, a voice interface that simply iterates through visual elements is inferior to one based on multi-slot grammars and mixed-initiative dialogs. Instead of a directed prompt and response sequence, the user will be able to speak more complex sentences with multiple data points. Though the visual presentation should help the user say the right things, the screen layout should not constrain the flow of voice conversation in any way. The designs of modalities must support each other without compromising their specific benefits and efficiencies.

- **Presentation Optimization**

The content provided through each modality should be complementary. For example, the screen may present the list of items while the voice may only say the number of items. This asymmetry minimizes information redundancy while still indicating the availability of multiple modalities. The user may choose to respond by selecting an item on the screen or by asking for more details over voice.

- **Presentation Synchronization**

Presentations through different modalities must be closely synchronized. If the changes do not happen in a timely fashion, the user will assume error on his part or system failure. Timely positive feedback throughout the application is critical to the smooth flow of the conversation and to the user's perception of communicating effectively with the system. However, the application has limited control over the timing of the data channel and the modalities may get out of sync. The application must be designed to support recovery from such situations.

- **Input Synchronization**

The user may respond through multiple channels. Depending on the context of the dialog, those individual inputs may be elements of a composite user input comprising of multiple modalities. For example, a user may point to a graphical element and speak a voice command. The semantics of individual inputs must be considered to discriminate between valid composite inputs and user errors.

- **Adaptive Assistance**

The level of assistance necessary for a given user is difficult to predict. The user's experience with the application, physical situation, and preferences are some of the deciding factors. A good design must consider all of these factors to determine the appropriate level of assistance. However, the user should be able to

adjust the level at will. The proper controls should be offered in both visual (as more/less options) and voice modalities (through universal commands like "explain", "keep quiet", "I am reading", etc.). This also will allow the user to focus on a single modality without a distracting nagging from the other ones, especially voice.

- **Situational Awareness**

The awareness of a user's situation and activities can help the application adjust the interface properly. The user should be able to make an explicit declaration ("I am driving" or "I am at a meeting"). The application also can obtain clues from the presence and location information as well as noise-level information from the recognizer. Future phones are expected to detect physical movement and orientation. Then the application will know to start talking when the user keeps the phone down or when he is walking. There has been some promising research conducted in this area. The success of the Nintendo Wii game controller proves that the gesture recognition is a viable interface.

- **Multi-User Interactions**

If the application allows multiple users to participate, the modalities they each choose may be different. The application should individualize the presentations accordingly. Furthermore, if the users can communicate among them, it would be highly desirable to provide on-the-fly media translations. If you consider that users may speak, text, click, or gesture, the task is quite difficult. The current technologies may not be able to handle all cases efficiently. However, in applications creating social networks of mobile users, the advantage of maintaining preferences may outweigh occasional quirks of the translation.

- **Graceful Degradation**

The reality of the mobile environment is that the radio signals may fade away and connections may be dropped. Such problems may not necessarily cause a total loss of communication but rather affect only some of the capabilities. For example, a phone may lose the 3G connection and start operating in a 2.5G mode. Voice and data connections would still be available but not simultaneously. Other failure modes may cause a loss of just the voice or just the data connection. The application should still maintain a level of functionality under limited connectivity capabilities. The user may decide to operate under reduced functionality or just wrap up what he was doing. In any case, the user will have a chance for a soft landing. The application also should try to recover the lost connections (by retrying the data connection or by placing a new voice call). Since the state of server-side application is maintained separately from individual modalities, a recovered modality will rejoin the conversation at the right point and with the relevant information.

As you have realized by now, while multimodal applications make interactions more efficient and enrich the experience, the internal structure of multimodal applications is quite complex. The general architecture discussed earlier enables partitioning of the application logic into cooperating subsystems. This in turn offers opportunities for code reuse and improves the maintainability of the application.

The testing of applications for mobile users also requires new approaches. You will need to take your application out of the lab and test it in real world environments. The user valuation of modalities changes with the physical context in which they find themselves.

The crowd, noise, light, movement and other environmental factors can dramatically affect the usability and performance of the application. You will need to test it while driving, walking, or sitting at a street-side café.

5. Technologies

The simultaneous multimodal functionality described in this paper depends on simultaneous voice and data connections. This capability is currently available only on the 3G GSM networks. The 3G GSM networks use the UMTS (Universal Mobile Telecommunications System) technology that in turn uses the W-CDMA (Wideband Code Division Multiple Access) radio interface. Many of the 3G GSM networks also deploy HSDPA (High-Speed Downlink Packet Access) protocol to increase downlink speed, reduce latency, and increase the capacity through the better spectral efficiency. ITU and 3GPP (3rd Generation Partnership Project) are the standards organizations defining the above specifications. However, an easy way to learn a bit more is to browse Wikipedia at en.wikipedia.org/wiki/3G.

As of March 2007, there were 98 3G HSDPA GSM network operators in 52 countries. The two major GSM networks in the United States are AT&T (previously Cingular) and T-Mobile US. AT&T already has 3G/HSDPA in service. T-Mobile US is expected to start offering 3G/HSDPA later in 2007 and 2008.

The other two major U.S. mobile operators, Verizon and Sprint, use the CDMA technologies instead of GSM. Verizon and Sprint provide 3G capabilities based on the EV-DO (Evolution-Data Optimized) technology. They are currently deploying EV-DO Rev A that further increases the data speed.

The 3G EV-DO (CDMA) networks do not support simultaneous voice and data connections. On these networks, a multimodal application can only operate and deliver a “unimodal” flow of interactions.

Some phones support Wi-Fi and can maintain data connections over LAN (local area network) during voice calls. This enables simultaneous multimodal functionality around Wi-Fi hot spots. For example, T-Mobile MDA, while not a 3G phone, will support multimodal functionality under such conditions. The future adoption of WiMAX (also called 4G) will extend the hot spot size and make the wireless LAN approach less constraining.

Interestingly, EV-DO Rev A provides the data speed and short latencies that make it quite capable of carrying packet voice (VoIP) and thus theoretically enables simultaneous voice and data. However, operators have not yet announced any plans for such solutions. An additional push may come from future IMS (IP Multimedia Subsystem) deployments. IMS is a new telecom architecture that tries to leverage the IP-based protocols to create a more flexible telecom platform and to deliver “IP multimedia services” to the end user.

To take advantage of the 3G GSM capabilities, a compatible mobile device is required. As of March 2007, Cingular’s (AT&T) web page listed eight models of 3G phones. Alternatively, any unlocked 3G GSM phone can be used (after the SIM card is inserted), as long as it supports the GSM and UMTS frequencies provided on the intended network. A phone with “quad” GSM frequencies would provide voice connections on any GSM network (with the possible exception of Japan). However, this is not a sufficient specification for the 3G compatibility because UMTS may operate on different frequencies. The phone

specification should list the supported UMTS/HSPDA frequencies separately from the GSM frequencies.

Intervoice multimodal applications rely on a web browser for the visual presentation and interface. In other words, no special software or hardware is required on the mobile device for multimodal applications to function. For example, an off-the-shelf Cingular 8525 phone with Windows Mobile 5 PocketPC operating system can be used to simultaneously access the voice and visual modalities of a multimodal application. Cingular 8525 is a branded HTC Hermes phone with quad GSM and the tri-band UMTS/HSDPA. The critical requirement is the support for JavaScript and AJAX, which enable the dynamic HTML updates during the multimodal interactions. The PocketPC Internet Explorer (PIE) browser, which comes pre-installed on Windows Mobile 5 PocketPC devices, provides the required capabilities for visual interactions. It should be noted, however, that PIE has more limitations than the desktop IE, so not all AJAX fueled presentations will work. You can find additional explanations on the Microsoft MSDN web.

On phones running Symbian S60 (Nokia and others), the Opera web browser provides AJAX support.

There are other technologies that could be used to provide a thin client for the mobile device. Adobe Flash is a very attractive candidate. This is the same technology that the ubiquitous Flash Player provides for desktop web browsers to enable dynamic multimedia content. Flash has not yet penetrated the mobile realm to the same extent, but Adobe seems to be working hard to improve the functionality and to support more phones. Further increases in mobile processing power and memory sizes will improve Flash performance.

A thin client also can be developed in Java for mobile devices (J2ME). The J2ME application can communicate over TCP/IP sockets, so the general IP client/server architecture can be retained. To keep the client thin, the complete screen content and layouts will have to be generated by the server and simply rendered by the client. This approach can produce an efficient solution that can be adapted to many phones. The negative aspect is that the client will have to be installed on the phone either by the user or by the carrier. The additional problem with Windows-based phones is that J2ME may not be pre-installed. J2ME availability on Windows Mobile devices depends on the phone vendor and model. For example, Cingular 8525 and T-Mobile MDA are both Windows Mobile 5 PocketPC phones built by HTC, but only the 8525 comes with J2ME.

The Media Exchange server architecture discussed in this paper is a general application server environment that can support a mix of various applications and callers. A desktop user with a softphone (a desktop application providing VoIP capabilities) can call and successfully interact with a multimodal application. A combination of a traditional phone and a desktop also will work. The call control part of the application can execute call transfers in the network or it can connect the caller to a contact center.

Intervoice Media Exchange and Media Server are based on standards defined by W3C Voice Working Group (www.w3.org/Voice/).

Here are the references to the standards mentioned earlier in this paper:

- SCXML: State Chart XML (www.w3.org/TR/scxml/) SCXML is a language based on Harel State Charts (see the original paper by David Harel at www.wisdom.weizmann.ac.il/~dharel/SCANNED.PAPERS/Statecharts.pdf).

SCXML provides an efficient state machine notation thanks to the support for hierarchical and parallel states. Media Exchange uses SCXML as a server-side application notation that is independent from the server environment and the client types.

- CCXML: Call Control XML (www.w3.org/TR/ccxml/)

CCXML is a "third party" call control model, including call joining and conferencing. Intervoice CCXML browser is provided on Media Server. CCXML documents can be dynamically produced by the SCXML-encoded logic running on Media Exchange. Third party, CCXML compliant documents should also work.

- VoiceXML: www.w3.org/TR/voicexml20/

VoiceXML is designed for creating audio dialogs that feature synthesized speech, digitized audio, recognition of spoken and DTMF key input, recording of spoken input, telephony, and mixed initiative conversations.

The standard compliance of the Intervoice VoiceXML browser provided on Media Server was certified by The VoiceXML Forum (www.voicexml.org).

6. Conclusions

Mobility has become a fact of life. New devices and network capabilities offer an opportunity to improve the user experience and to create completely new categories of mobile applications. Multimodal applications can provide the flexibility and adaptability that mobile users require. This is new and exciting territory.

Intervoice Solutions Framework and its Media Exchange provide a robust and efficient environment for the development and deployment of multimodal applications for mobile users. At the same time, all of the traditional capabilities expected from a telecom platform are still available to build upon.

7. Appendix

Author:

Bogdan Blaszcak,
Director of Enterprise Product
Management,
Intervoice, Inc.

Please send feedback to bogdan.blaszcak@intervoice.com

For additional information on Intervoice please visit www.intervoice.com

The products, features and specifications discussed in this document are subject to change without notice.

All trademarks are the property of their respective owners.

Copyright ©2007, Intervoice, Inc.

World Headquarters

Intervoice, Inc.

17811 Waterview Parkway
Dallas, TX 75252

(US) 800.700.0122

(Int) 1 972.454.8000

International Headquarters

Intervoice Limited

50 Park Road
Gatley, Cheshire UK
SK8 4HZ

+44 (0) 161 495 1000

Offices worldwide, including Santa Clara,
Orlando, Sao Paolo, Dubai, South Africa,
Singapore, Ireland, Germany, The Netherlands
and Switzerland.

