

R-evolution

Newsletter for developer community
For private circulation only

RELIANCE
Communications
Anil Dhirubhai Ambani Group

Editorial

Dear Developers

We live in interesting times. The mobile has revolutionized how people communicate and share information. From an ordinary communication device it's metamorphosis into a more powerful gadget for accessing a variety of information and entertainment content on the move is truly amazing! Voice Applications have greatly facilitated access to information, marketing, entertainment content as well as transaction execution. Datamonitor estimates that global voice business will grow to more than \$ 2 Billion by 2007, a clear indication of the growth in commercial and consumer confidence related to this technology. The lead article " Building Enhanced Services through enhanced Standards " throws light on what constitutes enhanced services, changes in the last few years for development and deployment of applications in the enterprise and are now becoming accepted in the service provider environment. The Technical Article focuses on understanding and creating VoiceXML applications. In the Tech Tip Part 1 we highlight " Best practices to improve performance in servlets." In the interview of the month Mr Sandeep Ganguly – Onmobile's Sales and Marketing Head shares with us the product innovations that Onmobile has brought in the VAS (Value Added Services) domain in partnership with Reliance Communications.

We look forward to receiving your feedback and valuable suggestions.

Warm regards,

Chanda Mahajan
Reliance Developer Programme Team



Lead article

Building Enhanced Services
with Open Standards



Technical article

VoiceXML



Technical tip

Best practices to improve
performance in Servlets



Interview of the Month

Sandeep Ganguly



Monthly Quiz & Poser



Overview

With the widespread acceptance of open standards at both the network and application levels, the model for deploying service provider enhanced services has changed. This article touches on a few of the relevant standards, provides food for thought around the kinds of applications that can be deployed, and describes the benefits that can be realized through the use of standards based technologies.

What Are Enhanced Services?

In our daily use of the most ubiquitous communications device on the planet – the telephone – we interact with so-called Enhanced Services on a routine basis. Enhanced Services include network features that range from simple to complex. Some examples include:

- Voicemail
- Directory Assistance
- Prepaid calling card processing
- "Color" ringback
- Conferencing Services

These services have historically been provided by proprietary hardware platforms, often dedicated to a particular function. This meant that the service provider needed different platforms for each of voicemail, conferencing, and so on. These platforms were very reliable, and performed their functions well – but each required their own form of management, application updates could be challenging, and might even require the use of highly specialized professional services that were only available from the original vendor – if applications could be changed at all.

What Has Changed?

Over the past few years, a number of standards from the Internet Engineering Task Force (IETF – www.ietf.org) and

the World Wide Web Consortium (W3C – www.w3c.org) have become widely accepted for the development and deployment of applications in the enterprise and are now becoming accepted in the service provider environment.

On the 'plumbing' side, protocols like the Session Initiation Protocol (SIP – <http://www.ietf.org/rfc/rfc3261.txt?number=3261>) can provide the foundation for many different kinds of interactions, including telephone calls, instant messaging, and video interaction. SIP is important because it provides a standard way for different network and endpoint devices to interact, and the basis for building much more complex services. In particular, interoperability between different devices is more easily proven and understood than traditional telephony technologies.

SIP enables the infrastructure to work – but we still need to build applications. This is an area that is rapidly evolving, but some of the most interesting standards in this area include VoiceXML and CCXML. VoiceXML is a markup language that allows the management of dialogs with a caller. VoiceXML manages resources like speech recognition and speech synthesis, while allowing two-way interaction with the caller. There is a family of related W3C specifications that cover related technologies, and allow development of very rich, portable applications.

VoiceXML provides a way to interact with the caller – but it provides very little in the way of control over the call itself. This is where Call Control XML (CCXML) comes into play. CCXML provides a markup language that allows for granular processing of the signaling around a call. CCXML provides control over dialogs (implemented with, for example, VoiceXML) conferences and connections to callers and other endpoints (like

telephones, instant messaging clients, and so on).

The hardware advances tied to Moore's law have also been important. Rapid advances in commodity hardware have allowed similar advances in the ability of software solutions to meet advanced application needs. Server vendors also provide commodity hardware in form factors – including hardened platforms and blade-based configurations – that are very attractive to service providers for cost and maintenance reasons.

Real-World Applications

What kind of applications can be built using these technologies? The great thing about the combination of standards such as these, and commodity hardware, is that they act as true enablers, allowing the application visionary to realize opportunities that simply could not be achieved using less flexible technologies. Let's look at an example – conferencing.

We have all used teleconference 'bridges', where we call a free-phone/toll-free number, are prompted for a conference identifier, and perhaps a password or PIN. After listening to 'music on hold', the conference host starts the conference, and we are all joined to a conference, where we try to accomplish something.

Conferencing platforms are typically dedicated resources, with relatively fixed applications that control the conference. The host has some level of control over the conference.

Now, imagine having the ability to develop this application using web-based technologies like VoiceXML and CCXML. This will give you the ability to rapidly develop, test, and deploy changes to the conferencing application itself. If we were to make use of CCXML for overall control of the application, and VoiceXML for any

Lead article (Cont'd)

caller interaction, we are now able to develop an application that mirrors the traditional hardwired conference application.

So far, all we've gained is the ability to use commodity hardware for our conference processing. We will probably still want to use dedicated hardware for mixing of very large conferences, but we can mix smaller conferences in software quite effectively. However, there is more to think about here

Consider the authentication part of the conference – where you enter your conference identifier and PIN. Suppose we wanted to speech enable this – that is, use speech recognition to collect this information from the caller. How about using speaker identification and verification to allow the caller to identify themselves using a voice print? Technologies like VoiceXML allow this to be done very effectively.

How about extending the conference host interface such that the user could use speech to identify callers from their address book that they would like to add to the conference? The conferencing application can use VoiceXML to collect the information, while CCXML can be used to place the outbound calls to the people who are to be added to the conference.

Consider the music on hold that you listen to while waiting for the host. Making this customizable is also quite achievable using these technologies. Personalization is one of the most attractive benefits of web-based applications.

This brief description of how a 'traditional' conference application might be enhanced or extended will hopefully get you thinking about what might be possible. All of the enhanced service applications described earlier in this article, and more, have been deployed using VoiceXML – and are processing many millions of calls every day, around the world.

Tangible Benefits

There are a number of tangible benefits related to the use of open standards in deployment of enhanced services.

Time to Market

The use of open standards allows service providers and application developers to bring applications to market quickly. This has a number of significant implications, and allows service providers to become much more agile in their approach to the marketplace. Applications can be developed effectively at lower cost – and can then be modified and reinforced as dictated by the success of the application. This allows time-limited opportunities to be tackled effectively by the service provider.

Amortized Infrastructure Cost

The use of standards-based technologies allows the use of 'universal' infrastructure to support this broader range of applications. Instead of dedicating hardware to particular classes of enhanced services, it becomes reasonable to share infrastructure where it makes sense.

Reduced Development Cost

Particular phases of application development are much reduced in terms of cost. This means that it becomes more reasonable to trial applications in the marketplace. Those that are successful can be easily reinforced, while those that have not garnered market attention can be decommissioned without having paid the price (in time and cost) that a traditional proprietary deployment would entail.

Reduced Vendor Lock-In

The use of open standards for deployment of services and applications means that there is choice – for both the service provider, and the application developer – in terms of the platforms on which applications are deployed. Further, the ability to leverage commodity hardware provides both a selection of vendors, as well as lower cost, and improved performance.

Conclusion

Service Providers are increasingly leveraging open standards in their deployments of network services. This in turn allows rapid responses to changes in market conditions, the deployment of rich services, and the ability to take advantage of advances in hardware and applications.

Watch for future business and technical articles that will describe how you can take advantage of these technologies, and discuss real-world success stories.



About the author

Sunny Rao is the Managing Director for the South Asian operations of Genesys Telecommunication Labs which includes India, Sri-Lanka, Maldives & Bangladesh.

Prior to his current assignment, he used to run the Managed Services division of Genesys across Asia-Pacific and was instrumental in Genesys' relationships with Telecom Service Providers and ASPs such as Telstra, AAPT, ePLDT, LG Dacom Korea, China Telecom, StarHub, Global Speech Networks etc. He holds a Bachelor's Degree in Electronics and Communication Engineering and an MBA. He and his family have called Bangalore their home.



Voice-based products are becoming pervasive in the service industry today. Companies are realizing the benefits of using these products to save money and serve customers faster. Let us take a look at what it takes to make such an application.

We have tried to give you a broad overview of what the VoiceXML markup language is, and more importantly, how it works and the type of applications you can develop using this technology.

VoiceXML (VXML) is a W3C endorsed markup language that allows developers to write advanced telephony applications using the same development models that are used in software development. It involves a combination of standardized languages and off-the-shelf hardware components.

There were a number of enterprises using voice applications before VoiceXML came along. But these products needed substantial investment in terms of expensive hardware and proprietary software. Scalability and inter-operability were also an issue. As the scope of the application increased, there would be a constant need for hardware and software upgrades. This forced most enterprises to stick with the bare-body system that they had developed right in the beginning.

As in the software industry, a strong need was expressed for a standardized voice model too.

VoiceXML allows the average web developer to write telephony applications with the ease and simplicity of writing an HTML web page. As VXML is a tag-based markup language, its structure is very similar to HTML in many ways, but instead of being a primarily visual medium, VoiceXML is an auditory medium that allows the end user to navigate through his/her 'telephony menu' by using voice commands, rather than by clicking a button on a web page.

How does VoiceXML work?

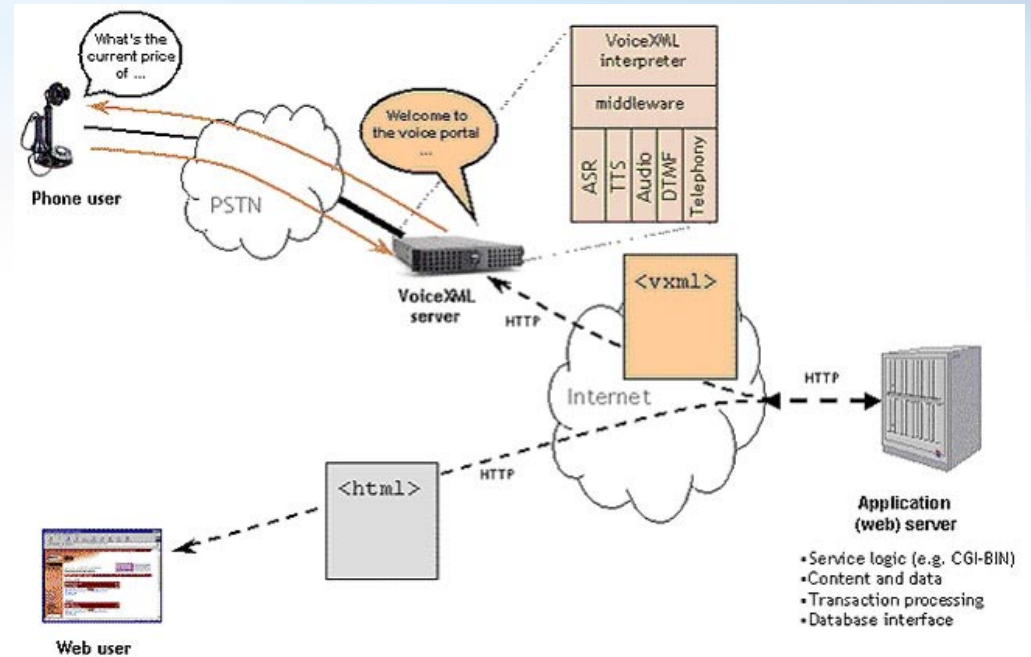
As mentioned above, VXML deployments involve VXML code working with a number of off-the-shelf components. 2 key components are the Automatic Speech Recognition (ASR) and the Text-to-Speech (TTS) engines.

Putting it simply, the ASR is the input component and the TTS is the output component. The ASR engine is used to grab the user audio input and process it as per the rules defined in the VXML code. The TTS engine is used to translate the text output of the system back to voice to be played back to the user.

Most advanced ASR and TTS engines have a level of self-learning built in that allows them to train itself as it receives more data. The ASR and TTS engines usually implement

VXML. But different vendors have slightly different implementations of VXML based on their product features.

We will discuss a generic VXML structure here that is usually the same across all platforms.



Source: Kenneth G. Rehor - www.voicexmlreview.org

Voice Interface v/s Web Interface

Here we have tried to draw analogies between actions involved in using a Web interface and using a Voice interface.

Users use

Web page: A screen + keyboard/mouse as a visual interface with software that interprets HTML and can interact with web servers. Which is a standard web browser.

VoiceXML: A phone + engines that recognize spoken input and reads out text as a audio interface with software that interprets VoiceXML and can interact with web servers (Audio Web Browser).

1. Start

Web page: Connect to the Internet and open the browser software on your screen.

Technical article (Cont'd)

VoiceXML: Use a phone to call the VoiceXML platform – VoiceXML interpreter, ASR engine & TTS engine.

2. Request

Webpage: type the URL into the Address input box.

VoiceXML: The application lists out a menu from which a user can select a particular option by saying it out or selecting a certain key on the keypad.

3. Fetch

With both the cases, pages are fetched using the technologies like Perl, PHP, ColdFusion, ASP, and JSP can be used to write code that generates VoiceXML/HTML dynamically.

This is where the convergence of different platforms happens. The business logic could be written in one location with inputs from multiple locations and can output the data to different streams.

4. Display

Webpage: The page is interpreted by the browser and output is presented as text and graphics, and input is accepted from the keyboard or mouse.

VoiceXML: The page is interpreted by the VoiceXML interpreter and output is presented as audio (recorded or TTS converted)

Creating a basic VoiceXML Application

A VoiceXML application would be stored in a number of files with .xml as the file extension. Every file should ideally start with the regular XML tag

```
<?xml version="1.0"?>
```

The vxml tags should then enclose the remainder of the document's instructions. Within this tag, the version attribute should be set equal to the version of VoiceXML being used ("2.0" in the present case) as follows:

```
<vxml version="2.0">
```

Inside of the <vxml> tag, a document is broken up into discrete dialog elements called forms.

Each form has a name and is responsible for executing some portion of the dialog. For example, you may have a form called "Welcome" that prompts the caller to make a selection from a list of options and then recognizes the response.

A form is denoted by the use of the <form> tag and can

be specified by the inclusion of the id attribute to specify the form's name. This is useful if the form is to be referenced at some other point in the application or by another application. For example,

```
<form id="Welcome">
```

would indicate in a VoiceXML document the beginning of the "welcome" form.

<block> – encloses a sequence of statements for prompting and computation.

```
<?xml version="1.0"?>
<vxml version="2.0"
xmlns="http://www.w3.org/2001/vxml">
  <form id="WelCome">
    <block>
      Welcome
    </block>
  </form>
</vxml>
```

Grammars

A grammar is what the system expects the user to say. The platform compares what the user has said against the grammar and acts on it. The grammar consists of single words right up to complex phrases. Every field in the form has some grammar associated with it. Grammar can be present at different levels: for the entire platform, for a particular form or for a single input field. The scope of the grammar is defined the way we define the scope of variables in a software application.

A grammar consists of various rules that are used to match the input. Most applications have a root rule specified that can be used throughout the system. If the grammar has a root rule, then you can use the grammar in your VoiceXML application without naming which rule to start from. A larger or more complex grammar, however, may have several rules that can be used as a starting point. To use the subrules, the rule has to be requested for specifically. In addition to the subrules present, the root rule would still be considered valid. Any VoiceXML application specifies a grammar to use with the VoiceXML <grammar> tag. It can use built-in

grammars and application-defined grammars. A built-in grammar is one that is built directly into the VoiceXML interpreter. These grammars can be used right out of the box. An application grammar, on the other hand, is one that a developer defines from scratch. An application grammar may be a grammar that is defined specifically for a particular application or it may be a part of a general library of grammars to be reused across applications.

Application grammars can either be inline or external. The entire definition of an inline grammar appears directly in the <grammar> element of the VoiceXML document; the definition of an external grammar appears in a separate file.

If the <grammar> element contains content, that content is the definition of an inline grammar. If the element does not contain content, it must have a value for either the src attribute or the expr attribute. In this case, depending on the value of that attribute, the reference is to either a built-in grammar or an external grammar file. The example below shows one of the way to use grammar element in an VoiceXML application.

```
<?xml version="1.0"?>
<vxml version="2.0"
xmlns="http://www.w3.org/2001/vxml">
  <form id="WelCome">
    <block>
      Welcome
    </block>
    <field name="sandwich">
      <prompt> Please say the name of a sandwich. </prompt>
      <grammar type="application/x-jsgf"> hamburger |
burger {hamburger} | (chicken [sandwich]) {chicken}
      </grammar>
    </field>
  </form>
</vxml>
```

VoiceXML Services

Some key services that are best suited to the VXML platform are listed here. Most of these applications have been around for a while as standalone products. With the

Technical article (Cont'd)

emergence of VXML, these can be merged with data applications and be processed in common locations. Over a period of time, the number of services being offered using this platform has increased. Services that can be – and to some extent are – offered include information retrieval, directory assistance, e-commerce and unified messaging.

Information Retrieval includes address news, stock, traffic information based on the location. Directory assistance is another service where cost-savings and rising efficiency have been noticeably high. E-Commerce includes banking sites where users can get their account information on the phone without human intervention. This increases security as well as improves response time for such requests.

Unified messaging is one area where the voice platform will have substantial usage. The user can configure email accounts on a particular number, and when the user calls into the specified number, his/her emails can be read out to the user. This can also be extended to calendar and address book information.

References:

1. <http://www.vxml.org>
2. <http://www.voicexmlreview.org>
3. <http://www.voicexml.org>
4. <http://developer.voicegenie.com>
5. <http://cafe.bevocal.com>
6. VoiceXML: Strategies and Techniques for Effective Voice Application Development with VoiceXML 2.0 – Chetan Sharma and Jeff Kunins

About the authors :



Ritesh Shah is currently working as a Technical Lead with the Applications Solutions & Content Group, Reliance

Communications. He holds an MS in Computer Science from the University of Massachusetts. He enjoys reading and going on long drives. He can be reached at Ritesh.b.Shah@relianceada.com.



Vishnu Gouda

He is currently working as a Senior Developer with the Applications Solutions & Content Group, Reliance Communications. He enjoys listening to music, playing cricket and is passionate about blogging. He can be reached at Vishnu.Gouda@relianceada.com



Best practices to improve performance in Servlets

Technical tip

Part -1

Overview of Servlets

Servlets represent an extension to the HTTP server. It takes http request as an input and sends the http response to the client as an output. Servlet API provides two packages they are `javax.servlet` and `javax.http.servlet`. These packages contain interfaces and classes to deal with generic and http functionality that means you can write a Servlet in java to get http request and send a http response to the client. Client is typically a browser. These interfaces are implemented by Servlet Engines. There are numerous vendors who provide Servlet Engines to work with Servlets, for example

Tomcat, weblogic, webshpere etc.

Servlet is loaded into the memory by Servlet Engine and it calls `init()` method on first request and then onwards only `service()` method is called for every other request by creating a separate thread for each request and finally `destroy()` method is called when the Servlet is removed by the Servlet Engine. `Service()` method can be replaced by `doGet()` or `doPost()` method.

Note that this architecture is a multi threaded model which is generally followed in most of the applications. You can even work with single threaded model by implementing `SingleThreadModel` interface where the Servlet Engine creates a separate Servlet instance for each request.

Use `init()` method as cache

The default mechanism of a Servlet Engine is to load a Servlet in multithreaded environment. In this environment, a Servlet `init()` method is called only once in its life time. You can improve performance using `init()` method. You can use this method to cache static data. Generally a Servlet generates dynamic data and static data. Programmers often make a mistake by creating both dynamic and static data from `service()` method. Obviously there is a reason to create dynamic data because of its nature but there is no need to create static data every time for every request in `service()` method.

Technical tip (Cont'd)

Optimization techniques in service() method

When you write a service() method for your Servlet, you can improve performance by using following techniques.

1. Use StringBuffer rather than using + operator when you concatenate multiple strings.
2. Use print() method instead of println() method.
3. Use ServletOutputStream instead of PrintWriter.
4. Initialize the PrintWriter with proper size.
5. Flush the data partly.
6. Minimize the amount of code in the synchronized block.
7. Set the content length.

Cache the static and dynamic data

The use of caching in different areas of your application gives very good performance. Generally every application's database schema will have at least some read only tables. There is no need of accessing these tables every time. You can cache that data in memory and reuse it instead of accessing database every time. It reduces network traffic, consumes less CPU cycles and gives good performance.

Caching can be done in three flavors namely static data caching, semi dynamic data caching and dynamic caching. Static data means that the data doesn't change in its life time, it always constant. Semi dynamic data means that the data changes but not often. For example the data that changes after every one hour can be called as semi dynamic data, the data does not change for every request. Dynamic data means that it changes the data for every request.

We will discuss a few caching techniques to improve Servlet performance. They are as follows

1. Utilizing Browser caching.
2. Caching dynamic data at the server.
3. Utilizing application server caching facilities.
4. Utilizing Servlet API's built in facility, HttpSession and ServletContext objects.

Caching at init() method is useful for caching static data and it reduces the creation time of static data for every request but any way finally we are passing data to the client on every request. This type of caching is useful

when you want to pass both static data and dynamic data to the client.

One more caching technique is utilizing the browser cache and also cache the content at the server, this can be done by avoiding a call to service() method if the output content is not changed. This technique is achieved by implementing getLastModified() method of HttpServlet class.

The third technique is that your application server may support caching facility for dynamic data. All you need to do is that configure the caching properties file which is supported by your server. You can give what Servlet you need to cache and session time out value to remove cache content.

The fourth technique is that you can use Servlet API's HttpSession and ServletContext objects for caching. HttpSession object is available for a user session across multiple requests and ServletContext object is available for all the users using the application. You can set cacheable objects into these objects and get those objects whenever you require within their scope. The methods that support caching are:

```
ServletContext.setAttribute(String name, Object cacheableObject);  
ServletContext.getAttribute(String name);  
HttpSession.setAttribute(String name, Object cacheableObject);  
HttpSession.getAttribute(String name);
```

....to be cont'd in the next issue



About the Author : Krishna Udipi
Krishna Udipi is currently working as a Java Mobile Application Developer with Applications Solutions and Content Group, RCOM. He enjoys blogging and listening to music. You can reach him at: Krishna.Udipi@relianceada.com

from page 10 ...

Monthly Quiz - Answer

1Q-ANS : B and C. The result range for random() is 0.0 to < 1.0; 1.0 is not in range.

2Q-ANS : B. Both substring() and charAt() methods are indexed with a zero-base, and substring() returns a String of length arg2 - arg1.

3Q-ANS : C and E are correct.

4Q-ANS : E. Hashtable is the only class listed that provides synchronized methods. If you need synchronization great; otherwise, use HashMap, it's faster.

5Q-ANS : A is correct. One of the main benefits of encapsulation is that encapsulated code is much easier to reuse than unencapsulated code. B, C, D, and E are incorrect. B is incorrect because inheritance is a concept that is independent of encapsulation. C and D are incorrect because encapsulation does not restrict the use of overloading or overriding. E is incorrect because HAS-A relationships are independent of encapsulation.

6Q-ANS : C. The code will not compile because a continue statement can only occur in a looping construct. If this syntax were legal, the combination of the continue and the if statements would create a kludgy kind of loop, but the compiler will force you to write cleaner code than this.

A, B, and D are incorrect based on the program logic described above.



Sandeep Ganguly leads the Sales and Marketing Team of Onmobile. He has over 14 years of sales and marketing experience in the Telecom Industry. In his current profile he is responsible for Business Development and Account Management for the Private Telecom Operators within India. Sandeep holds a Bachelor's Degree in Electronics and Communication (Pune University) and an MBA from IIM Kolkata.

VAS is also trying to convert the mobile phone into a multipurpose device providing entertainment like music, downloading full track songs, executing financial transactions like paying phone bills and using it as an extension of your computer.

Q1: Can you tell us about your company - OnMobile?

OnMobile is in telecom Value Added Services (VAS) and a pioneer in voice services and has developed and deployed a Indian speech recognition system across twelve languages and owns the IP for the same. In recent years it has ventured into new verticals of mobile content distribution, interactive media portal, mobile advertising, 1-to-1 direct mobile marketing and M-Commerce in India. OnMobile has operations across eight countries and its customers are telecom operators, media houses, Corporate houses & merchants by providing a turnkey solution to launch and operate VAS.

The Value Added Services ecosystem has 3 main constituents:

- The Platform/Technology Provider
- Applications Developer
- Content Aggregator
- Content Generator

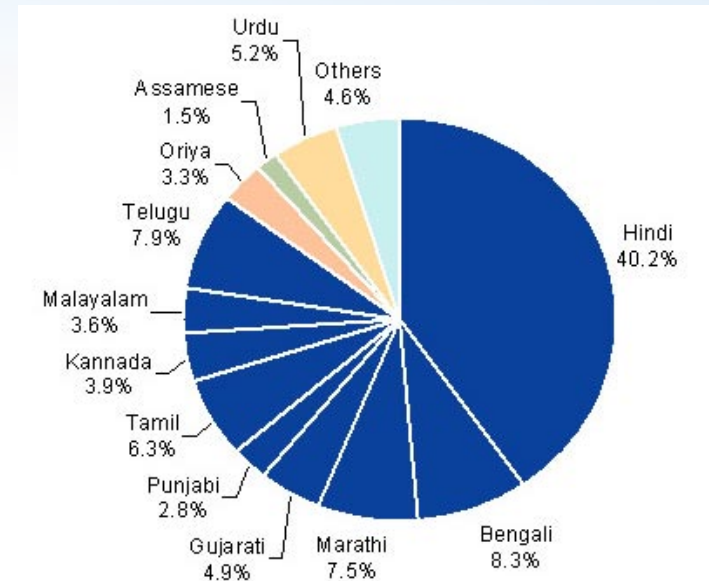
OnMobile provides a comprehensive managed solution to its customers that include the first 3 components. Additionally, we provide Technical, Operational and Marketing support services for all our customers, which enable them to focus on marketing the product, expanding the subscriber base for these services, increasing usage and maximizing revenues.

Q2: In what areas is OnMobile currently working with Reliance Communications?

Voice Portal

The primary service offered by OnMobile is the Multi-modal Multilingual Voice Portal services branded as 51234 by Reliance. Reliance subscribers can access a host of services on this voice portal in 10 regional

languages like English, Hindi, Punjabi, Marathi, Telugu, Tamil, Malayalam, Kananda, Gujrati and Bengali. The services are varied; some of the popular ones are Music, Mytunes, Contest, Voting and Dating. The language break-up on our platform are as follows:



MRadio - the latest and most innovative product in the music vertical.

'Music lovers rate OnMobile's MRadio better than regular Radio channel'
The consumer chooses the song he wants to hear on it by just dialing 51234777 (on monthly subscription plan) or 51234778 (non-subscriber) and enjoys an uninterrupted flow of songs of his choice - skipping songs he does not like & uninterrupted by advertisement.

Interview (Cont'd)

The service is a music lover's delight across the country, offering latest full length songs ranging from the latest Bollywood hits, regional hits, devotional etc. Music lovers in small cities & towns often deprived access to the latest songs are now enjoying their favorite music on MRadio, by accessing it from anywhere & anytime - even in the dead of the night. .

MRadio has many innovative features like personal album feature turns it into a 'Personalized Radio'. It empowers the subscriber to build his own collection of latest hit songs.

The approach is to provide a radio experience to the consumer where he/she controls the listening pattern. So from a mass product the radio becomes a personal entertainment tool.

Find-a-Friend

A user generated application, Find-a-Friend is a platform to get to know like minded people. OnMobile's in-house consumer research team conducts regular & periodic survey to capture the consumer behaviour across rural & urban markets - what are the end consumer's need & expectation from the various services being consumed by him. The study on Find-a-Friend service highlighted some exciting new opportunities which led to an introduction of a host of new features like online chat, dedication of songs and event based promotions.

Q3: How are OnMobile and Reliance Communications together creating value for the RCOM subscribers?

Product innovation has predominantly been the mantra - in the form of simplifying user experience, packaging and flexible pricing. Providing a simple, intuitive and elegant user experience is one of the key factors that have determined the success of our relationship.

Reliance Communication has always provided better value for money for its world class services to its subscriber's

base & we at OnMobile keep working on multiple fronts of User Interface and Packaging to reach out to more consumers with the same philosophy. While pay-per-use business models were relevant during the initial phases, we've introduced Subscription Services, which assures users of getting the content or service they desired at a defined price point.

Q4: Your views on the future of VAS in India / your future on the VAS scenario today and the days to come?

While traditionally the mobile operators have been the primary brand and provider of mobile content, Media companies have started to brand and offer their own products and content cutting across all Operators. Called 'Open-Garden' services, the mobile operator still plays a pivotal role in delivering content to the users, but the products are packaged branded and promoted by the media companies. This is leading towards a convergence in the Telecom and Media space with mobiles becoming an essential channel alongside print, radio and TV. This convergence is being driven by new technologies like Live Streaming, Personalized Radio and the like.

Q5: Future plans of OnMobile

We believe in providing cutting edge easy to use technologies & some our other successful products areas are -

On Device Portal: A handset client that is resident on the user's mobile phone downloading content automatically at requested periodical intervals and allows users to easily browse through the content catalogue or get the latest information, via a simple and easy to use graphical user interface

Phone Backup: The product allows users to backup all the contents on their mobile phone including address book, messages, pictures and downloaded applications onto a central network repository

Ease of content discovery, i.e the long tail approach. User should be empowered to discover content on the platform, instead of being stuck with what is offered upfront.

Press * to Copy: When users listen to someone's RBT and likes the song, they can simply set it as their own RBT by just pressing a single key on their phone keypad. This is by far the simplest application seen across the world today, wherein a customer can get something he likes at literally a single click of a button. The magnificence of the service is sometimes missed because of its simplicity.

Media: 505-xxxx is a service that is used by media and print companies to reach out to all customers on a single access code.

Rural: This is a new initiative and focus. These are initial days and we see a huge potential from these areas going forward.

The objective is to understand the needs of the new consumers of RCOM and create Value Added Services that they need for their daily activities. It is a difficult objective but the consumer will reward the company addressing the consumer's needs.



Monthly Quiz

1. What two statements are true about the result obtained from calling `Math.random()`?

(Choose two.)

- A. The result is less than 0.0.
- B. The result is greater than or equal to 0.0..
- C. The result is less than 1.0.
- D. The result is greater than 1.0.
- E. The result is greater than or equal to 1.0.
- F. The result is less than or equal to 1.0.

2. Given the following,

14. `String a = "newspaper";` 15. `a = a.substring(5,7);`

16. `char b = a.charAt(1);` 17. `a = a + b;`

18. `System.out.println(a);`

what is the result?

- A. apa C. aepa E. papp
- B. app D. aep F. papa

3. What two statements are true about properly overridden `hashCode()` and `equals()` methods?

- A. `hashCode()` doesn't have to be overridden if `equals()` is.
- B. `equals()` doesn't have to be overridden if `hashCode()` is.
- C. `hashCode()` can always return the same value, regardless of the object that invoked it.
- D. If two different objects that are not meaningfully equivalent both invoke `hashCode()`, then `hashCode()` can't return the same value for both invocations.
- E. `equals()` can be true even if it's comparing different objects.

4. Which collection class allows you to access its elements by associating a key with an element's value, and provides synchronization?

- A. `java.util.SortedMap` D. `java.util.HashMap`
- B. `java.util.TreeMap` E. `java.util.Hashtable`
- C. `java.util.TreeSet`

5. Which is true?

- A. Tightly encapsulated classes are typically easier to reuse.
- B. Tightly encapsulated classes typically use inheritance more than unencapsulated classes.
- C. Methods in tightly encapsulated classes cannot be overridden.
- D. Methods in tightly encapsulated classes cannot be overloaded.
- E. Tightly encapsulated classes typically do not use HAS-A relationships.

6. Given the following,

1. `int I = 0;` 2. `label:` 3. `if (I < 2) {`

4. `System.out.print("I is " + I);` 5. `I++;`

6. `continue label;` 7. `}`

what is the result?

- A. I is 0 C. Compilation fails.
- B. I is 0 I is 1 D. None of the above

Answers :

1) B and C. 2) B 3) C and E 4) E 5) A 6) C
.....for detailed explanation go to page no. 7



Monthly Poser

Who is the President of China Telecommunications Corporation?

- 1) Wang Xiaochu
- 2) Zhang Weihua
- 3) Chang Xiaobing
- 4) Wu Andi

Answers to the monthly poser should be sent to dadp.newsletter@relianceada.com mentioning monthly poser - July 2007 as the subject with the sender's location stated. The winner will be decided on the basis of a lucky draw and walk away with a prize.

Congratulations

The winner of last month's (June 2007) Monthly Poser through a lucky draw is Mr. Sureshbabu N from Tamilnadu

You can contribute ideas and information to R-evolution at the following e-mail address: dadp.newsletter@relianceada.com. Please note that contributions may be edited for clarity, style or length.

Editorial Team: Suresh Dabbara, Krishna Udupi, Chanda Mahajan

Design Team: Jophi Jose

Published by the Reliance Developer Programme and the Brand Team, Reliance Communications, DAKC, Navi Mumbai