

March 2006

R-EVOLUTION

Newsletter for the Developer Community

For Private Circulation only



Editorial

Dear Developers:

As I write this column, our external jury panel for the Reliance Mobile Developer Awards 2005-06 would have picked up the 8 winners of the Contest. Not only would this impart fame and appreciation to these developers but also provide them with an opportunity to monetize their applications after their commercial launch on R World. I won't divulge the names of the winners here and would urge our readers to visit www.dadp.com to view the full list of winners and the winning applications. Do send me your feedback.

Our country has abundance of talent and creative ideas especially among the student community. The contest best exemplifies this statement both in spirit and action. There is no limit to the push this contest will give to the cause of evolution of the mobile developer ecosystem in India. It will provide developers an ideal platform to put ideas to actions, create software products for the mobile environment, and benefit monetarily from them. Read more about the contest roadmap in our 'Update' section in future issues.

Ashwin Rayaoprolu, from the Reliance Developer Programme team is the author of the guest article in this issue. Do let us know how you liked it. With more developers contributing articles regularly to 'R-evolution', it is definitely a big encouragement for the Reliance Developer Programme team to hear from our Developer community. For when developers participate in the process of creating a strong and vibrant community by sharing knowledge, the whole mechanism becomes more collaborative and gathers a self-sustaining momentum. This definitely augurs well for our Programme in future.

Thank you all for making the Reliance Mobile Developer awards 2005-06 a great success!

Saurabh Chakrabarti

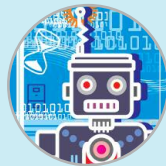
Reliance Developer Programme team



LEAD STORY
Innovation
Page 2



TECHTIP
Text Messaging
Page 3



GUEST COLUMN
Creating canvas
Page 4



FEEDBACK
Letters to Editor
Page 5

THE INNOVATION LAUNCH PAD

Page 2



THE INNOVATION LAUNCH PAD



The recently concluded Mobile Application Contest hosted by Reliance Developer Programme (RDP) received an overwhelming response from the developer community.

The four month-long contest that began on November 01, 2005 and ended February 28, 2006 saw enthusiastic participation from developers across the country. And it provided the winning developers an important launch pad for showcasing their innovations.

Innovations galore

The contest was a big success in terms of the number of entries and saw high levels of innovation in programmes which ranged from topics related to devotion to fun & entertainment.

A noteworthy feature was the participation of a large

number of student developers from leading technical institutes in India in the specially-introduced 'Student Category' in the contest. This separate category gave them a fair chance of competing with peer groups.

Through March 2006, the RDP technical support team checked all applications received in the contest for their completeness and final selection.

Developers whose applications needed modifications were asked to revert after the problems were rectified as communicated by the Reliance technical team within a stipulated time.

An external jury

To ensure fairness and transparency, all working applications were presented and demonstrated to a panel of external jury drawn from leading technology companies and experts from the IT and telecom domain.

Applications were judged on the basis of essential parameters like user interface, uniqueness/innovation, functionality, and business potential, with different weightage assigned to each parameter.

More details on the judges, judging process and selection of winners will be given in the next issue of R-evolution.

Sharing revenue

Any contest application, irrespective of whether it

wins a prize or not, would receive revenue share as per the Reliance Standard Revenue Share Agreement, if commercially launched on R World.

For students developing applications with Reliance Developer Programme as part of its University Relations Programme, a tripartite Revenue Share Agreement is being developed under which Reliance will share a certain percentage of application revenues with students and their institutes for any application developed by students, which is commercially launched on R World.

This initiative will not only enable students hone their skills by developing a carrier grade application but also provide an opportunity to cash in on their ideas.

Community recognition

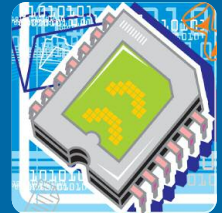
With initiatives like Mobile Application Contest and University Relationship Program, Reliance Developer Programme aims to provide continuous support and impart recognition to the developer community.

Developers may be established companies, individual entrepreneurs, budding university students and business professionals. The idea is to share knowledge and establish commercial relations between Reliance and Developers to grow the market and introduce innovative mobile products and services.

For more details, developers can visit and write to us at dadp.query@relianceinfo.com

SMS USING

WIRELESS MESSAGING APIS (MIDP 2.0) – PART 2



TECH TIP

In the previous issue we discussed WMA classes. WMA classes are an optional package in MIDP 2.0. In the concluding part, we'll discuss the implementation of the Java General Connection Framework.

The Java General Connection Framework (GCF) connector class `javax.microedition.io.Connector` instantiates instances of `MessageConnection`. The URL that is passed to the `Connector.open()` method determines the connection that will be opened. The URL should be in the following pattern:

The URL `sms://+18005555555` specifies a connection to send SMS messages to the phone number 1-800-555-5555. (Note that the WMA has no phone number format requirements; you can use any series of digits that your phone and network will recognize.)

- The URL `sms://+18005555555:1234` specifies a connection to send SMS messages to port number 1234 at the phone number 1-800-555-5555.
- The URL `sms://:1234` specifies a server connection to receive messages on port 1234. A server connection can also send messages.

Code examples

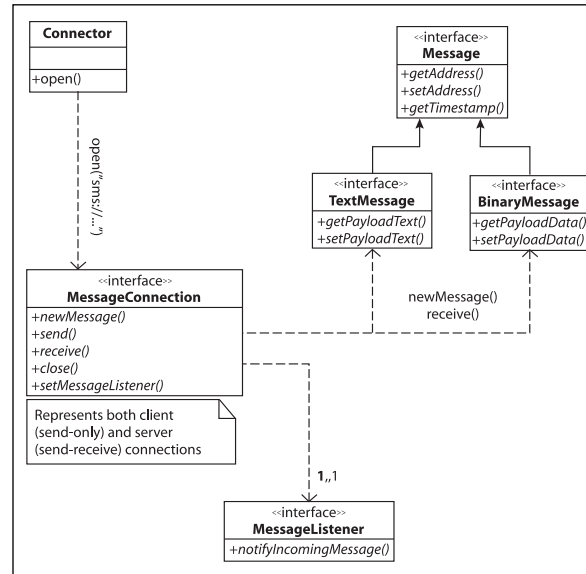
Let's take a look at some code examples.

Sending messages with the WMA is very simple. You can send a message to an arbitrary phone number and/or an SMS port through a `MessageConnection` constructed for that destination

Sending an SMS message

```
String urladdr = "sms://+123456789";  
// Or: String urladdr = "sms://+123456789:1234";  
MessageConnection conn = (MessageConnection)  
Connector.open(urladdr);  
TextMessage msg = (TextMessage)  
conn.newMessage(MessageConnection.TEXT_MESSAGE);  
msg.setPayloadText("Hello World");  
conn.send(msg);
```

A UML diagram for top-level WMA interfaces



For any more clarification write to dadp.newsletter@relianceinfo.com



CREATING UI WITH CANVAS

IN J2ME – PART I

In this article, written in two parts, I'll talk about the basic concepts needed by developers to build Lists using canvas. Here's the first part of the article:

Let me introduce you to a few low-level API classes, which will help build custom User Interface (UI) platform canvas, graphics, and font with great ease and confidence. All existing high-level APIs were initially built on a base of low-level APIs. So, by going a layer down, I can define the properties of UIs.

The Canvas class

The Canvas class is a base class for writing applications for low-level events and issuing graphics calls. Game applications are likely to make heavy use of the Canvas class. From an application development perspective, the Canvas class is interchangeable with standard Screen classes, so that an application may mix and match Canvas with high-level screens as needed.

For example, a List screen may be used to select the track for a racing game while a Canvas subclass would implement the actual game.

Canvas provides the developer with methods to handle game actions, key events, and pointer events (if supported by the device). Methods are also provided to identify the device's capabilities and keyboard mapping. The key events are reported with respect to key codes, which are directly bound to concrete keys on the device, use of which may hinder portability. Therefore, portable applications

should use game actions instead of key codes.

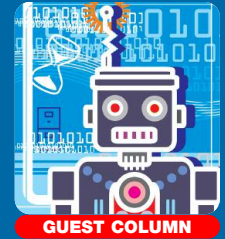
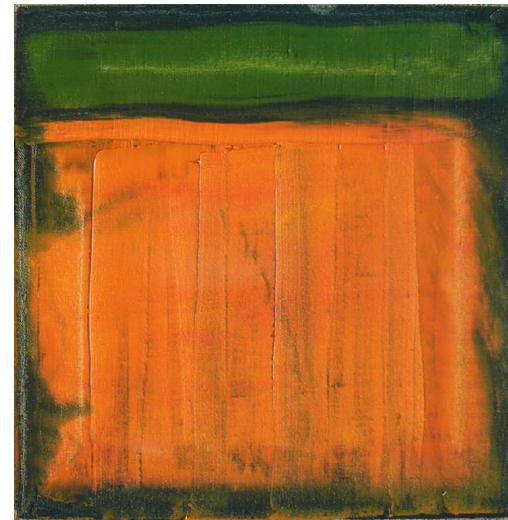
Like other subclasses of Displayable, the Canvas class allows the application to register a listener for commands. Unlike other Displayable, however, the Canvas class requires applications to subclass it for usage. The paint () method is declared abstract, and so the application must provide an implementation in its subclass. Other event-reporting methods are not declared abstract, and their default implementations are empty (that is, they do nothing). This allows the application to override only the methods that report events in which the application has interest.

The Screen class

This is in contrast with Screen classes, which allow the application to define listeners and register them with instances of the Screen classes. This style is not used for the Canvas class, because several new listener interfaces need to be created. An alternative way is to have fewer listener interfaces, but this would require listeners to filter out events in which they have no interest.

Developing AlphaScrollList

Now let's develop our First Application:



AlphaScrollList. The essential features needed are:

- 1) Four way Scrolling (Left, Right, Up, Down)
- 2) Show Title
- 3) Show Scrollbar (vertical and horizontal)
- 4) Index List so that we can get the Contents by Name, Position in list
- 5) Track Present position of list

I'll sign off for now. The second part of the article will explain the rest of the requirements for creating an AlphaScrollList.

In the meantime you can write to me with any queries at ashwin.rayaprolu@gmail.com

About the author



Ashwin Kumar is an active Reliance Developer Programme (RDP) member currently working in a major IT company at Hyderabad. He is passionate about Java and has developed a number of mobile applications based on J2ME. He has also participated in many RDP application development contests.



QUIZ

BRAINGYM

1. Which of the following network connections is supported by MIDP?

- a) UDP datagrams
- b) WAP
- c) HTTP
- d) All of the above

2. What is the maximum length for a record store in MIDP?

- a) 8 characters
- b) 32 characters
- c) 128 characters
- d) No limit on the length of a record store name

3. The CDC supports which of the following network connections?

- a) UDP datagrams
- b) TCP sockets
- c) HTTP connections
- d) None of the above

4. What is the Java API for XML Parsing (JAXP)?

- a) A new XML parser implementation
- b) A standard XML parsing technique
- c) A standard XML API that can be used by devel

opers with over an XML parser

5. What does SOAP define?

- a) The overall structure of the XML message
- b) The conventions representing the remote procedure call in the XML message
- c) A binding to HTTP
- d) The conventions to wrap and send an error back to the sender
- e) All of the above

Answers: 1C, 2B, 3A, 4C, 5E.

Monthly poser

What is the alternative developed by Hewlett-Packard to the Java technology?

- 1. Chaha
- 2. Chai
- 3. Chui
- 4. None of the above

Answer to last month's poser: Cyber Terrorism

The winner of last month's poser (through a lucky draw) is **Senthil Kumar K, Bangalore.** Winner will receive prize by courier.

Answers to the Monthly Poser should be sent to dadp.newsletter@relianceinfo.com mentioning Monthly Poser—March 2006 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.

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

Editorial Team : Saurabh Chakrabarti, Suresh Dabbara, Abhijeet Anand, Lakshman S Aiyar, Vijayam Raghunathan

Design: Ritu Nanda, Haresh Patel, Anand Sutar, Rohidas Adavkar

Published by the Reliance Developer Programme and Corporate Communications Teams, Reliance Infocomm, DAKC, Navi Mumbai



FEEDBACK

Letters to Editor

The Reliance Developer Programme Mobile application contest is good thing going for the developer community. Can we have a contest only for mobile games? We are students and are eager to develop games for Reliance.

Pradeep, Aakash, and Anirban

BE - Computer Science,
PCT Engineering College, Thane, Maharashtra

Editor: Dear Pradeep, Aakash, and Anirban, thank you for your feedback. We will surely keep your suggestion in mind when planning contests in future.

The 'Tech Tip' section on SMS using wireless messaging APIs in the February 2006 issue of R-evolution was very informative to developers like me. How do I contact Reliance for any specific query?

Lakshmanan Sridhar, Trivandrum

Editor: Dear Lakshmanan, it is good to know that you found the article in the 'Tech Tip' section of R-evolution, February 2006 informative. You may address any query to dadp.query@relianceinfo.com