Porting Mobile Applications - Overcoming the hurdles

The market for wireless games, applications and content is booming, and is contributing to major new revenue stream for the carriers. Mobile games are becoming increasingly popular and new applications are being announced for mobile applications daily. This all suggests that this is just the beginning of the trend and next generation phones will allow even more complex applications.

At Rapidsoft Systems, we have been working for a number of years on developing mobile applications and have worked with number of third party developers of mobile applications. It is an exciting field with many opportunities for games and application developers, carriers and for aggregators, but faces a large number of problems that limit deployment of mobile applications to a wider audience. Most of these are human made challenges and aren't going to go away in the near future. We in this paper describe these challenges and how to overcome them.

One of the biggest challenges facing the mobile games industry, however, is the sheer number of different devices, operating systems, screen sizes and local market requirements. With more than 4000 different J2ME enabled devices and several hundred Brew Devices in the global market in 2008, along with multi-language and other customization requirements from mobile operators, a mobile application developer faces a challenge in tapping into this broad market opportunity. Porting applications and games from one mobile device to another has become a critical issue for any mobile software company.

Critical Components of Mobile Porting

Device Types

J2ME Devices – a large number of vendors such as Nokia, Motorola, SamSung, LG, Philips make Java enabled handsets. Java was supposed to be the unifying force for mobile applications where you write applications once and then they will run everywhere. Unfortunately, variation in device implementations have made that promise of device portability become just a dream. Our experience shows that there a big variation on standard J2ME functionality between manufacturers and sometimes

even between the phones of the same manufacturer. In summary, unless you run a game or application on the phone, you won't really know what to expect.

Brew Devices: Most, if not all, CDMA phones are based on Brew operating system from Qualcomm. Brew provided a closed sand box that not only includes operating system but its own development environment. It means that even if you made a game or application run on J2ME devices – you must now do the same development on Brew environment.

Smart Phones: This category includes Windows Mobile Devices, iPhone and ever popular BlackBerry phones. Each phone presents its own challenges when it comes to porting an application from one environment to another.

Operating System Variations

A number of applications require that application be developed using native functionality of the operating system. The current generation of phones use several operating systems such as Symbian, Brew, Windows Mobile, Palm O/S and Blackberry O/S. Add to this Linux based phones that keep being announced by many manufacturers from time to time. The fundamental differences in operating systems make it hard for anyone to write once and use everywhere almost impossible.

Device Feature Variations

 In addition to software differences, mobile phones differ in a number of other areas such as screen size, internal memory, keyboard type, Touch screen, Camera types, Screen pixel dimensions and so on so forth. These features must be reconciled and one has to create an application that can deal with these variations.

Porting Choices

In the past, many game studios and developers considered porting skills – whether it be from console to PC or across consoles – to be a key part of the value add they would bring as either a developer or publisher. However, with over 4000 device platforms plus 80 different mobile operators, just collecting the information and guidelines on these devices and markets can prove to be overwhelming and very costly for small software developers.

One major component is economics of mobile applications porting. If, for example, a mid-size game publisher has 30 games in its portfolio and wants these available globally in multiple languages it would have to create close to 15,000 different builds (30 games x 5 languages x 100 top devices). At an internal cost of approximately $3,000 per build it would need a budget of $ 45 million for mobile porting alone – something most mobile game budgets won't support. In addition to basic porting a developer must also take care of specific requirements from mobile operators for billing API's, game community API's or operator branding etc. To add these operator specific variations, one must spend even more money to accommodate these requests.

Given the above scenario, what are the choices for mobile application and game developers? Today there exists basically two ways to attack this problem as a mobile game developer / publisher: develop in-house mobile porting capabilities and become an expert on mobile porting internally; or outsource this task to external specialized "porting houses" like Rapidsoft Systems, or work with service providers, some of whom may who have developed internal expertise in the area. Either way, the porting process can be done manually or by using automated porting tools to drive down costs and speed time to market.

Internal Porting

Developing the expertise to port across mobile devices internally is a risky approach. Firstly, strong relationships with the mobile operators and device manufacturers around the world are needed to ensure availability of the necessary information, as well as guidelines and devices to port the applications. Secondly, global testing facilities are needed to be able to load applications onto the actual devices and test them – the frequencies and network protocols of wireless networks in various parts of the world often differ from a local network. Thirdly, software and tools are required just to manage the immense numbers of source code builds, and staff trained on all of the devices and tools is critical.

As recently as a year ago, internal porting seemed doable for mobile game developers. There were only twenty or so devices and only a few mobile operators selling games. Now as the market grows and matures, the challenges to scaling this business are a significant hurdle. Automated tools may be the saviors. With automation, publishers can rely on the tool providers for device information and to provide the necessary workflow engines and data bases to effectively and efficiently perform the ports and to manage the growing number of builds.

The Outsourcing Alternative

As the pressure mounts to get more versions of games out faster, game publishers find themselves turning to outsourcing. Whether it is with small local shops or with larger, low cost, offshore software houses, the challenges remain substantial.

While a company that specializes in mobile game porting is more likely to have success in sourcing devices and mobile operator guidelines, they remain challenged with global testing facilities. But probably the biggest challenge, not unlike any outsourcing project, is maintaining quality. Often, to meet the strict deadlines imposed on them, outsourcing firms employ multiple employees to port the same application. The result is inconsistencies that are unacceptable to the publisher. No matter how clear the guidance, individuals will approach creating code differently, resulting in different end user experiences.

Again, porting partners that employ automation tools can provide time to market and consistency across a broader range of device builds. This is a key differentiator that must be looked for when selecting a porting partner.

Porting Strategies and Solutions

While the mobile market is quickly becoming a real opportunity for existing PC and console game titles and applications, and the revenues are starting to roll in, the porting challenge can easily derail the opportunity to reach this massive market.

Through standardization and reuse of code elements, a mobile developer has to streamline its internal process to create significant efficiencies in porting. Some large mobile software companies have developed teams on multiple continents to be able to handle local testing and relationships with mobile operators. Others have created long term porting arrangements with porting teams like at Rapidsoft Systems, situated in other locations.

Whatever approach is taken, it is clear that game developers and publishers have to consider their porting strategies carefully. The market for mobile applications games is here and growing rapidly. The demographics of the market for wireless services and for console and PC applications and games are very similar – young males – and thus there is a significant untapped opportunity to exploit existing titles on this new breed of mobile phones. It is also clear this market opportunity will only drive further advances in porting technologies tools from which the whole industry will benefit.

Rapidsoft Systems Porting Capabilities

Rapidsoft Systems provides both application development and porting capabilities valuable insight into the numerous device parameters required for planning the development and porting strategy of your application.

RapidSoft Systems, provides cost effecient porting solutions for all major platforms. We port for all platforms - be it J2ME or Symbian or BREW or Windows Mobile or BlackBerry or iPhone. Mobile Porting services offered by RapidSoft Systems can be divided into the following areas:

- **Cross platform or Intra platform porting:**
    - o   J2ME to BREW, Symbian to BlackBerry, J2ME to iPhone etc.
- **Operator or Carrier specific porting:**
    - o   Most US (AT&T (Formerly Cingular), T-Mobile, Sprint, Verizon), European (Vodaphone, Orange, Virgin Mobile etc.) and Canadian Carriers (Rogers, Telus etc.)
- **Inter platform porting:**
    - o   Up-gradation of OS versions, new device compatibility, new features addition or enhancement, etc.

We have the experience of porting applications in more than 1500 handsets across the globe for diverse platforms and operators.
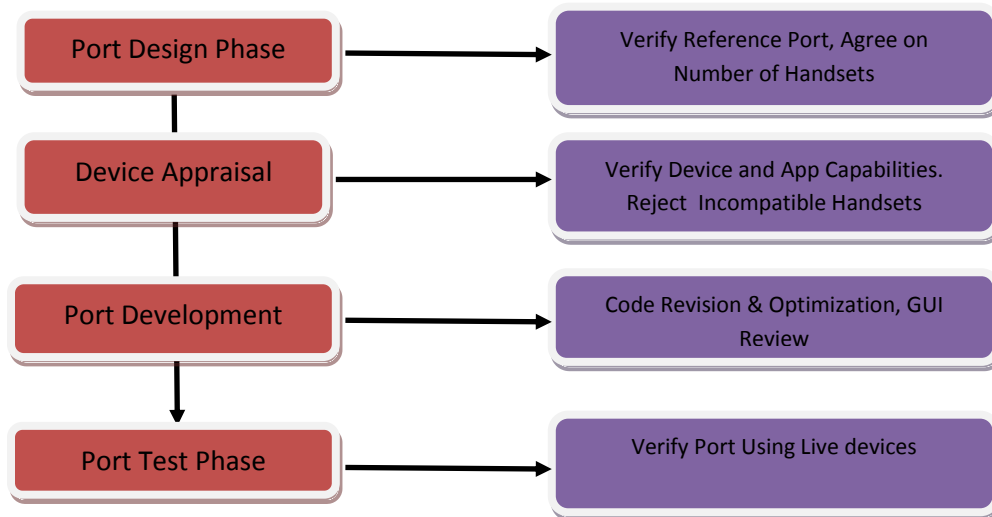
We support porting and testing, for the following mobile technologies:

- **J2ME (CLDC 1.0/1.1, MIDP 2.0/1.0, Mobile Media API, Wireless Messaging API, Games API etc.)**
- **Symbian**
- **Brew/C++**
- **Blackberry SDK**
- **Flashlite**

- **Windows Mobile 5.0, 6.0**
- **Embedded Visual Studio (embedded Visual Basic/VC++)**
- **Microsoft .NET Compact Framework**
- **Metrowerks CodeWarrior**
- **iPhone SDK**
- **Custom SDKs**

Our Porting Process Summary :

1. **Port Design Phase** – We create a reference port, decide target number of handsets and prepare a technical analysis of our porting strategy.

2. **Handsets Appraisals** – We make sure we have the phone and there no know limitations of the phone that prohibit porting of any specific application or specific features of that application. We will let the customer know any discovered limitations that will make exact port impossible (like no camera on the phone for a mobile application whose primary input comes from camera).

3. **Port Development Phase:** In this phase, we set customization strategy and goals and carry out application code revision and optimizations. GUI changes are also optimized and dealt in this phase.

4. **Port Test and Verification:** In this phase, we test each port on the live device for functionality test and carry out agreed tests. It is ensured that there no major anomalies from the reference port. The diagram below summarizes our four phase approach to porting.

Port Design Phase	→	Verify Reference Port, Agree on Number of Handsets
Device Appraisal	→	Verify Device and App Capabilities. Reject Incompatible Handsets
Port Development	→	Code Revision & Optimization, GUI Review
Port Test Phase	→	Verify Port Using Live devices

Conclusions

Use of external porting can cut down the cost and time to market applications. While it is possible to create an internal porting capabilities for most companies that option will be quite expensive since it requires significant man power efforts that are cyclical in nature. This means using a trusted partner like Rapidsoft Systems to port your applications can be the best solution for you.

For more information and specific questions, please contact us at:

Rapidsoft Systems, Inc,
Mailing Address: 7 Diamond Court, Princeton Junction,
New Jersey 08550, USA

www.rapidsoftsystems.com

Phones: **1-609-439 9060 (US East Coast, NJ Office)**
          1-408-829-6284 (US West Coast, San Jose Office)
          Toll Free (North America): 1-800-946-5490
          Fax: 1-831-855-9743

Email: info@rapidsoftsystems.com