

Multi Platform Compatibility on Mobile Operating Systems

G.Aparna¹, D.R.S.Sai Krishna², Dr.K.Sreenivasa Ravi³

Electronics and computers engineering^{1,2,3}
IV/IV B.Tech^{1,2}, Professor¹, K.L.University^{1,2,3}, Guntur, Andhra Pradesh, India

Email: appy_happy91@yahoo.com¹ sai_krishnan123@yahoo.com² ravi.kavuluri@klce.ac.in³

Abstract: Mobile Computing and its vast applications is the present day's most widespread technology. Mobile operating systems demonstrate six different leading platforms, each an island onto itself with a set of applications developed for that platform alone using the designated Software Development Kit (SDK). In order to develop the same application on different platforms, end-users have to learn x-many development frame works and duplicate effort everywhere becomes an overwhelming prospect. This paper provides solution along with an application for operating on two different mobile operating systems 'Android' and 'Windows Mobile', i.e. development of an application in cross platform environment.

Key Words: mobile computing; software development kit; development frameworks; android; windows mobile.

I. Introduction

Android is a software stack for mobile devices that includes an operating system, middleware and key applications developed by Google. The Android SDK provides the tools and APIs necessary to begin developing applications on the Android platform using the Java programming language. Likewise, Windows Mobile is a mobile operating system developed by Microsoft that was used in smart phones and mobile devices. Visual C++ is its development language. Now when a standalone application is considered, both these two major operating systems have their own methodology for executing them individually ^[1]. The problem here arises when it is required to run one operating system application on another.

These problems are due to different kernel structures, even though each has its own importance. Their developmental languages are logically similar but theoretically and syntactically different. Hence to develop the same application on two different platforms, it's like go-to-eat-cake. Also a developer must be completely aware of the syntaxes and semantics of the two programming languages and methodologies even for a small application. Effort gets duplicated and even a small bug leads to erroneous result. The case is not only for these two operating systems, but with others too. ^[2]

This is where multi platform compatibility proves its essence.

II. Existing Resolutions

(i) Modification of Internals

Most of the answers can be generated from the question; as such this problem has a solution in the problem definition itself. The internals of these operating systems are to be modified and the library files are to be made

executable for these two operating systems. This is done by introducing special library class files into their respective software development kits (sdks) and kernels. The compiler structures are modified in such a way that any programming language code is applicable to an operating system.

These architecture changes inside as well as outside the kernel makes it compatible for any kind of programming language code. The special features of android kernel are ashmem, logger, binder etc. Maintaining all these features certainly increases the overall maintenance and integration costs. ^[3]

The authentication to modify these structures is not easily obtained.

(ii) Web Applications

Based on the drawbacks of the above method, it is obvious that modifications to internals are a hectic task. Android and Windows Mobile allow the applications written in HTML 5 and JavaScript. It includes detailed processing models to encourage more interoperable implementations; it extends, improves and rationalizes the markup available for documents, and introduces markup and application programming interfaces (API)s for complex web applications.

JavaScript is a prototype-based scripting language that is dynamic, weakly typed and has first-class functions. It is a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles. But this method also has a drawback that only web applications are supported.

(iii) NAND ROM

Additional ROM is installed into the operating system externally. This is connected as logic NAND to the original ROM of the operating system. This interchangeably

After Android is installed, applications are executed as like in an Android smart phone. But in order to switch between the operating systems, an application called DualBoot is designed. This is used to select the desired operating system as per the application to be executed.

The Dualboot application provides the flexibility to switch between the Windows and Android OS. Actually it is a program that calls all the functionalities of OS which includes root directories, ports connection and communication, auto-resolution, alignment etc. Even it has the capability of setting the priority to kernel on which a user is working with.

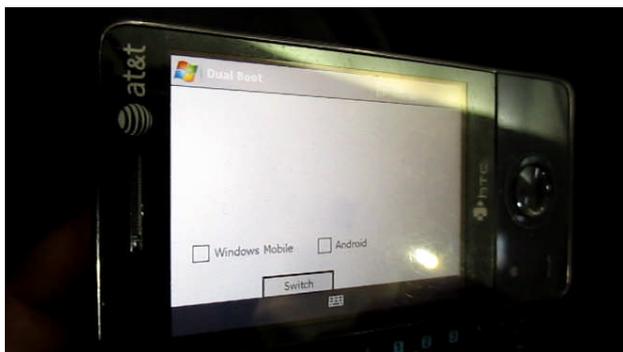


Fig VI : DualBOOT Application

The code is debugged and executed in emulators. It is run through Visual Studio for Windows Mobile and in Eclipse for Android emulators respectively.

Later this code is to be dumped into the mobile. In Android environment, the code is to be downloaded into the memory card partitioned area.

The operating system is switched from Windows Mobile to Android through the DualBoot application. On the selection of Android operating system and “Switch”, re-booting starts.



Fig VIII : Android OS Switch Screen

The Android welcome screen appears after Android gets booted inside Windows Mobile.

IV. Results



Fig VII : Windows Mobile Application

An application is executed on the Windows Mobile Operating System that prints the title and authors’ details.

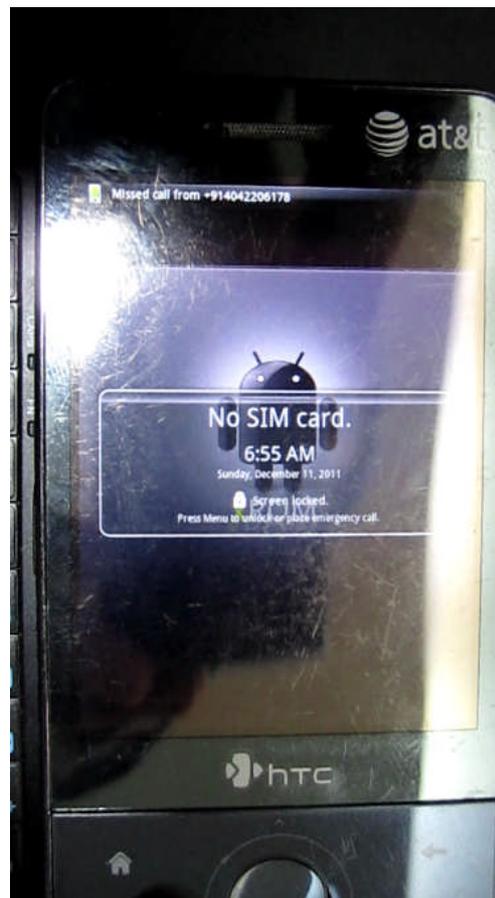


Fig VII : Android welcome screen

The same application that is previously executed on Windows Mobile OS is now executed in Android.



Fig XI : Android Application

These, therefore show a clear picture of how the same application can be executed on the two different operating systems.

V. Advantages

The proposed solution proves a lot than the existing solutions. New operating systems and/or software's can be tested without altering the current system in a safe method. Moreover different applications can be made to run simultaneously. It allows different services in single mobile which in turn saves money as no extra hardware is required. Any changes made in the settings and also the applications installed will all reside in the microSDcard itself. These changes do not affect Windows Mobile operating system. Even though, smart phones are provided with additional memory but much of it is not being used appropriately. Instead such platform virtualization or multi-booting enables to switch easily to the required operating system and operate on an application may it be web or stand alone.

VI. Conclusion

In this busy world, time efficiency and work efficiency are crucial challenges and are something that cannot be ignored. Then why should developers need to waste time in developing a code for an already existing one. So by creating this virtual or multi-boot environment, enormous amounts of human efforts and time are going to be reduced in developing the application codes for mobiles. In future it can be considered and implemented for all operating systems [5-10].

References

- [1]. Asoke K Talukder, Roopa Yavagal, Mobile Computing- Technology, Applications and Service Creation, Tata McGraw-Hill Education, Jan 1, 2005
- [2] Pei Zheng and Lionel Ni, Smart Phone and Next Generation Mobile Computing, Morgan Kaufmann, 2006
- [3].<http://developer.android.com/guide/basics/what-is-android.html>.
- [4]. D.R.S.Sai Krishna, et al., Cross Platform Compatibility on Mobile Operating Systems, International Journal on Computer Science and Technology, vol.2, no.4, pp.462-464
- [5] Y. Zhang, L. Wu, A Robust Hybrid Restarted Simulated Annealing Particle Swarm Optimization Technique, Advances in Computer Science and its Applications, vol. 1, no. 1, pp. 5-8, 2012
- [6] Chengwen Zhong, Min Zhong, Cunru Bai, A high-order discrete scheme of Lattice Boltzmann method for cavitation simulation, Advances in Computer Science and its Applications, vol. 1, no. 1, pp. 73-77, 2012
- [7] Vikas Deshmane, XML driven SCPI interpreter, Advances in Computer Science and its Applications, vol. 1, no. 1, pp. 59-62, 2012
- [8] Maziar Rezaei Rad, Mani Rezaei Rad, Shahabeddin Akbari, A Maximum Power Point Tracker for Photovoltaic Arrays Using Genetic Algorithm Plus Fuzzy Cognitive Networks, vol. 1, no. 1, pp. 35-40, 2012
- [9] Y. Zhang, et al., Artificial Bee Colony for Two Dimensional Protein Folding, Advances in Electrical Engineering Systems, vol. 1, no. 1, pp. 19-23, 2012
- [10] Mohammad Saad Alam, Mohammad Fazle Azeem, Dynamic Tuning of Scaling Factors of Fuzzy Knowledge Base Controller for Non-Linear Complex Systems, vol. 1, no. 1, pp. 60-64, 2012