Making Interoperation on Multiple Different Mobile Cross Platforms using LCIM Model

C. Shanthi[a], M.S. Josephine[b] and V. Jeyabala Raja[c]

[a]Ph.D., Research Scholar, Vels University, Chennai, India. Email: shan_c08@yahoo.co.in
[b]Asst Professor, Department of MCA., Dr. MGR Educational and Research Institute, Chennai, India. Email: josejbr@yahoo.com
[c]Professor, Deprtment of Computer Science and Engineering. Velammal Engineering College, Chennai, India. Email: jeyabalaraja@gmail.com

Abstract:

Objective: **Interoperability (**Best Interoperation between cross platform mobile apps using LCIM model). **ANALYSIS**: Mobile devices are gaining more importance and nowadays sharing across platforms is "unpredictable" at best. When we try to take a video on android phone and send it to a friend for them to view on their iphone? Android can interoperate with other android phone and iphone can interoperate with iphone pretty well that is about where it ends.

Finding: Multi-media interoperability is just the tip of the iceberg. Managing interoperability at the application level can be an astonishing task, and on the platform level is just the starting point of bridging the two worlds. It leads to further challenges down the road and several questions. Unfortunately this issue can be resolved by using levels of conceptual interoperability model. Now we can have a discussion on our regular analysis of standards and conventions for design of user interfaces for various mobile platforms, as well as scaling methods operational on different physical screen sizes. Interoperability of different systems, including HTML5, Java and .NET is also inside the center of this work.

Keyword: Mobile apps, interoperability, levels of conceptual interoperability model, Multi plate form, cross platform.

1.   INTRODUCTION

Many challenges for apps developers emerged for different platform application require to be educated rapidly in order to build up native apps for multiple platforms .The different platforms, mobile devices come with different screens in different sizes, resolutions, and aspect ratios. Many problems are faced during the development of mobile applications – especially in cross platform, the issues of user interface. In this paper we concentrate on the deal with questions? Can we go for best level of interoperation between cross plat form using LCIM model. The LCIM model is the process and form towards enhance interoperability among apps. The LCIM can be used in both descriptive and prescriptive form.

Applications developed can be used on all platforms is often extremely difficult. All platforms have different Software Development Kits with different libraries and different ways to design user interfaces. The programming languages also differ for these platforms. Android has many different devices available, including small smart phones and large smart phones. IOS, likewise, runs on the iPhone.Apps developers practice their craft in a digital universe that includes multiple versions of competing operating systems and thousands of devices in various screen sizes.

2. INTEROPERABILITY

The Challenge of Cross-Platform Development. Cross-platform can generate important effectiveness for mobile app development, but their receiving and usage among app developers is still low. The cross-platform development approach compared to a multi-platform approach using native development platforms come from the use of a single codebase, which in turn can result in improved interoperability with the cross platform development and maintenance. Interoperating different systems implemented on different platforms which run on a specific platform, Difficulty of entry into mobile application interoperability development can be lower in cross platform development environments with the help of LCIM Model.

In many iPhone and Android operating systems, panel layouts, functions are designing and maturing a cross-platform app which is giving a high-class client alertness on two or more platforms which will be certainly very tricky, When people consider the best option of successful consumer apps which is built on any Native iOS or Android. When we see Face book happening out with a Cross-Platform mobile app but it got changed directions to Native iOS and Android when people realized the user experience was not generally encouraging.

2.1. Interoperability and Integration Challenges

Integrating the app with preferences, limited settings, and notice apps can is quite a task such a task can be overcome with the help of LCIM model. To build a mobile application with best interoperability, Companies looking on a mobile platform which should ask themselves how to use programming languages to develop variations of the same application for multiple operating systems? In this Technical Perspectives, discover a relatively new cross-platform programming languages tools which helps us to overcome the difficulties of interoperability. Now we go for a model implementation which gives us best interoperability in between cross-platform LCIM-Levels of Conceptual Interoperability We start a process of interoperability between mobile apps cross-platform we can come across many levels of interoperability to succeed the level of data exchange between operating systems.

- No interoperability, there No. exist collaboration of data between any applications;

- Partial interoperability, only some data collaborate between applications;

- Total interoperability, all data exchanged can be used in other exchanged application, is a level of interoperability better but difficult to reach.

Now we go for an analysis how the difficulty can be reduced using (LCIM) levels of conceptual Interoperability in mobile apps cross-platform.

2.2. Levels of Interoperability

Level 0: Individual systems which have No. Interoperability.

Level 1: There exist a lot for exchanging information among the participating systems. In this level where communication infrastructure is very good reputation allowing exchanging bits and bytes where essential networks and protocols completely clear.

Level 2: The Syntactic Interoperability level introduces a regular structure to the data which is used to swap over information.

Level 3: When we exchange common information, position model is used; now we reach the level of Semantic Interoperability.

Level 4: Pragmatic Interoperability can be achieved when the inter operation systems are alert for the methods and procedures that each and everyone are in work.

Level 5: Systems can be operated on data more than one time, state of that system will get changed, now this combines the assumptions and constraints which are concern with data interchange. The systems have attained Dynamic Interoperability and have ability to recognize the state changes which takes place in the assumptions and constraints that both are making overtime, and can have improvement for those changes.

Level 6: The final point of conceptual model has assumptions and constraints of the carrying great abstraction of reality are associated, the maximum level of interoperability is reached:

2.3. Conceptual Interoperability

Now we need a fully specified but implementation independent model as requested in Davis and Anderson [19] and it's not a text relating the conceptual idea.

The level of conceptual interoperability model (LCIM) was introduced by Tolk and Muguira (2003) which is not the end where two or more interoperation is possible. The Figure 1 shows the layered development of LCIM model which is shown below.
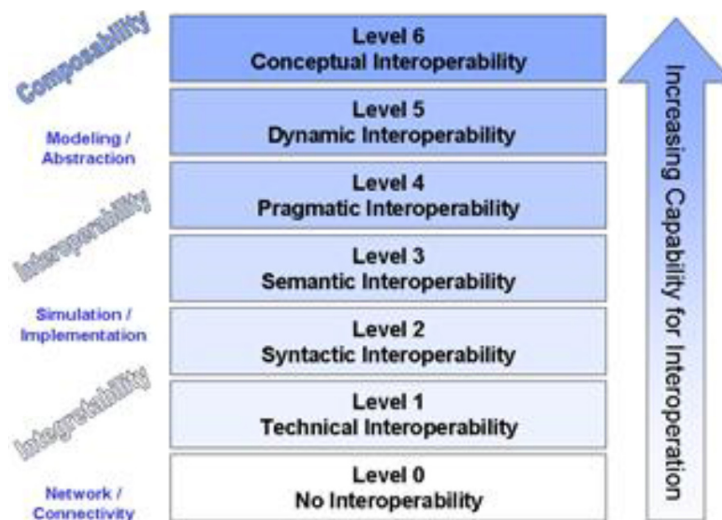

Figure 1: The Levels of Conceptual Interoperability Model

3. FINDINGS

When we develop cross platform mobile apps we find many tools, but with this tools we can find many pros and cons when we go for interoperability in between mobile apps. Now we can go for a view about the best of some tools which are there in the field and the pros and cons of those best tools. When we refer tools for developing interoperability among mobile apps there are many extra-ordinary tools but for best interoperation we process with LCIM model.

3.1. Apache Cordova/Phone Gap

Cordova/Phone Gap developers develop their own mobile applications using HTML, JavaScript and CSS. "Phone Gap" is possibly one of the most recognizable names in this space. Originally created by Nitobi, the name was changed to "Apache Cordova" when it was donated to the Apache Software Foundation.

Pros

- Cordova apps they are the set up, like a native application, and they have the capability to control app store.

- Cordova undergo a plug-in architecture, which is accessible in native device APIs and also used completely in a modular way.

Cons

- The command of being opens source and leveraging the great talent of a bulky collection of contributors is together a blessing and curse. When there is a need to expand app with a custom Cordova/Phone Gapplug-in, you will find one odds which may be out of date and that it won't support the target platforms you need.

- Odds are that you chose Cordova in part, to avoid the requirement for specialized native platform skills.

- The performance of Cordova/Phone Gap apps which is often been criticized.

Appcelerator

This provides anintegrated (across devices) JavaScript API, attached with native-platform and some specific features.

Pros

- When we make use of native UI mechanism is a performance to win, and the Alloy framework attempts to normalize UI across platforms.

- The best use of JavaScript to standardize code across platforms enables us to control existing skills on multiple target platforms.

Cons

- Developers are much essential to deal with target platform SDKs locally. It's much recommended for the team to establish a prohibited build environment/Cross Interoperation process if you choose to manage SDKs locally, especially if you target multiple platforms.

- Standardized UI across platforms, while doubtfully a "pro", is also a "con" in which our team will require educating on a proprietary technology to increase skills that are not openly transferable exterior Titanium.

When we go for comparison of these tools which has many pro and equal cons when developing interoperation among the cross platform mobile apps. The level of interoperation which is followed in LCIM model makes the interoperability the best in cross-platform mobile apps.

The below mentioned Figure 2 shows some of the tools for cross platform and its pros and cons, how much it is used in the field has been mentioned .when we go through these tools we can find the drawbacks which we face in all the tools interoperation which can be rectified with the help of LCIM Model. Cross-platform

development approaches are one way of increasing asset reuse between platforms and reducing drawbacks of transmission of messages. The biggest drawback encountered in the approach is insufficient interoperability support on cross platform apps.
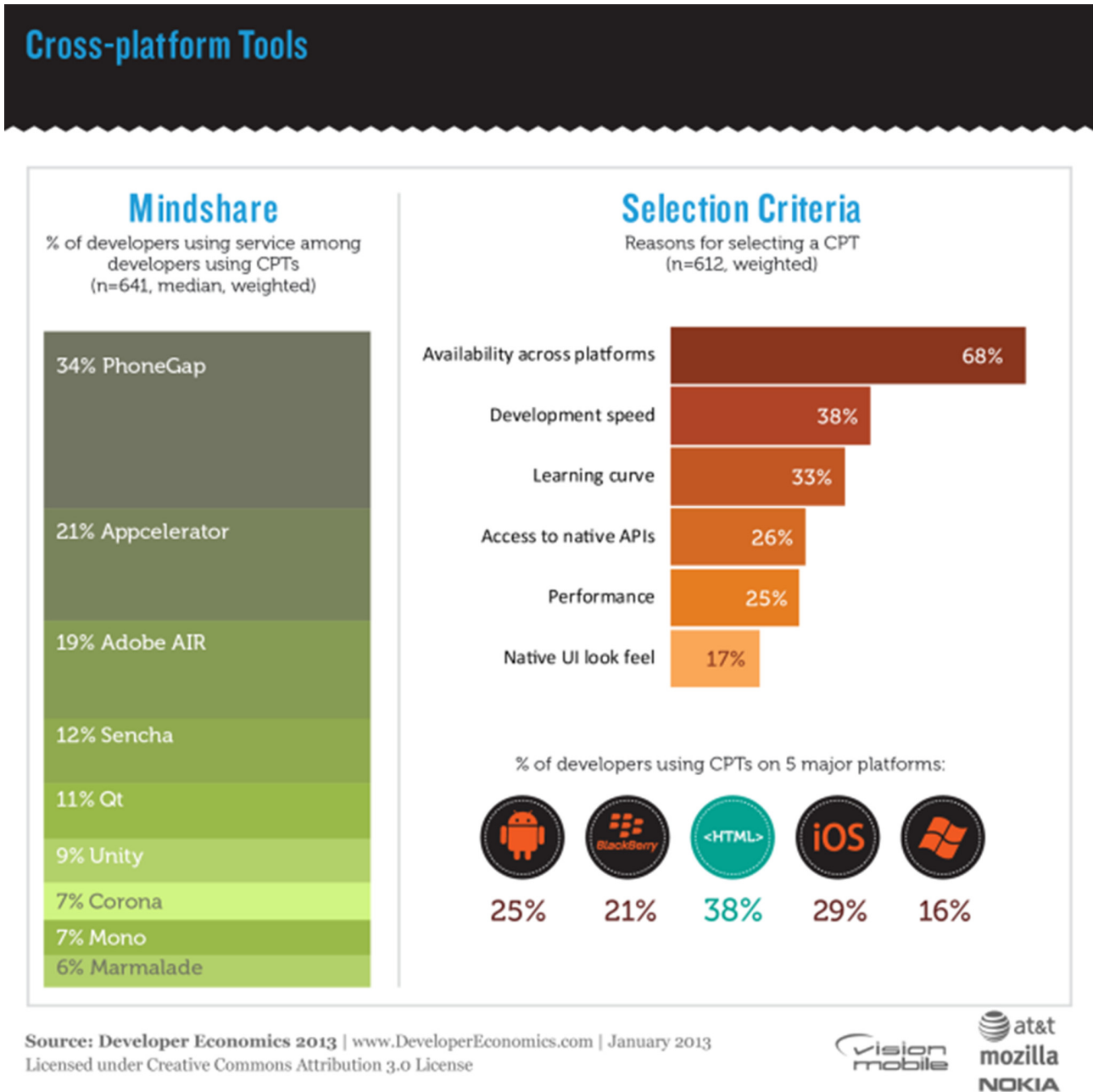

Figure 2: Cross Platform Tools and Their Usability

4. CONCLUSION

The current Smartphone market has made it necessary to develop applications for several platforms. Cross-platform development approaches are one way of increasing asset reuse between platforms. Our study focused on the application interoperability approach using the popular existing tools which has many pros and cons when

interoperating between cross platform. Overall, the level by level approach was found using LCIM model and suitable for best interoperability among the cross platform apps. Comparison of the tools and models among application approach with other cross-platform approaches would be another interesting topic, perhaps by implementing the same demonstrator using different approaches.

REFERENCES

[1] Tolk A, Muguira JA. "The levels of conceptual interoperability model". *Fall Simulation Interoperability Workshop. Orlando, Florida: Simulation Interoperability Standards Organization,* 2003.

[2] International Journal of Emerging Technology and Advanced Engineering Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 3, Issue 5, May 2013).

[3] Tolk, A. (2010). Interoperability and Composability. Chapter 12 in J.A. Sokolowski and C.M. Banks (Eds): Modeling and Simulation Fundamentals - Theoretical Underpinnings and Practical Domains, John Wiley, 403-433.

[4] Applying the Levels of Conceptual Interoperability Model in Support of Integratability, Interoperability, and Composability for System-of-Systems Engineering. Andreas TOLK, Saikou Y. DIALLO, Charles D. TURNITSA Virginia Modeling Analyses & Simulation Center, Old Dominion University Norfolk, VA 23529, USA +1.757.686.6200 (Phone), +1.757.6214 (Fax), [atolk, cturnits, sdiallo]@odu.edu.http://www.iiisci.org/journal/CV$/sci/pdfs/P468106.pdf.

[5] [Levine 03] Levine, L.; Meyers, C.; Morris, E.; Place, P.; & Plakosh, D. Proceedings of the System of Systems Interoperability Workshop (February 2003) (CMU/SEI-2003-TN-016). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2003. <http://www.sei.cmu.edu/publications/documents/03.reports /03tn016.html>.

[6] [Tolk 03] Tolk, Andreas & Muguira, James A. ―The Levels of Conceptual Interoperability Model.‖ Proceedings of the 2003 Fall Simulation Interoperability Workshop. Orlando, Florida, Sept. 14-19, 2003. Orlando, FL: Simulation Interoperability Standards Organization, 2003.

[7] Current Perspectives on Interoperability CMU/SEI-2004-TR-009 ESC-TR-2004-009, Lisa L. Browns word David J. Carney David Fisher Grace Lewis Craig Meyers Edwin J. Morris Patrick R.H. Place James Smith Lutz International Journal of Applied Engineering Research, ISSN 0973-4562 Vol. 10 No. 60 (2015).