

Mobile-Enabling Enterprise Business Applications using Model-Driven Engineering Techniques

Suman Roychoudhury and Vinay Kulkarni
{suman.roychoudhury, vinay.vkulkarni}@tcs.com
Tata Research Development and Design Center,
Tata Consultancy Services,
Pune – 411013, India

Abstract

According to a recent Gartner report it is predicted that the number of the combined installed base of smartphones will exceed the installed base of PCs by 2013. This indicates that there would be a huge market of mobile-enablement of existing enterprise systems. Current state-of-the-art within the industry is to develop a fresh channel application for all mobile platforms of interest. Instead we propose to use model-based generative approach to deliver the same specification onto multiple mobile platforms, i.e., an ability to model the service being invoked, the interfacing screen, the data along with the interaction protocol and the capability to generate code from its specification. We have initial proof-of-concept results that verify the validity of our proposed technique.

1 Introduction

The advent of rich mobile computing platforms [1] along with continuous improvement in the underlying hardware capabilities and network infrastructure [2] have resulted in a steady demand for mobile computing applications. Although early adoptions were focused towards entertainment centric applications (e.g., gaming, music, movies, videos), recent trends suggest a deeper interest in mobile enablement of enterprise business applications (EBA). For example, financial applications such as mobile banking, portfolio management, mobile brokerage etc are slowly gaining acceptance among mobile users [3]. Financial institutions and business houses who were earlier cautious on providing customer specific services on mobile devices are now following a more liberal path. Moreover, a recent Gartner report [4] predicts a steep incline in the number of smartphone users over PC users and this indicates a paradigm shift from the traditional desktop-based computing to mobile computing. However, this sudden rise in demand for mobile enablement of business applications may come at a high price. This is because of the diversity in hardware, platform, environment, user interaction etc., such applications have to be rewritten for every specific target context although their core functionally remains exactly the same. This increases the development cost of creating “multiple same” application thereby imposing a heavy financial burden on business institutions and also slows down the application’s time-to-market. Therefore in this position paper, we explore novel software engineering techniques that can transform EBAs to be mobile compliant.

2 Enterprise Business Applications

Typically, any n-tier enterprise business application can be modeled using 3 layers, namely, the business logic, the data and the user interface (UI) layer. The business logic is generally encapsulated within a set of services that interacts with the data layer. The UI layer typically constitutes the front-end of the system that involves interaction with the user interface. Fig 1a depicts the above scenario that shows how the interaction among these layers takes place via simple request and response protocol.

Fig 1b presents the same EBA model from a mobility perspective. Some of the concerns that are associated in modeling the data part of any EBA are synchronization, consistency, persistent storage etc. On similar lines, the concerns that are associated with modeling the UI part are the interaction objects (i.e., widgets), their mode of operation (modality) and their interaction pattern. For example, there may be multiple ways a user can browse a product catalog, but it is also important to find the most effective or optimal way the user can browse through the same information depending on the device and the platform (mobile) he/she chooses to use. Therefore interaction pattern plays an important role in the design of UI part of any application. Finally, the services that are generally distributed in nature are invoked via different communication protocols, for example SOAP, SMS, IVR etc.

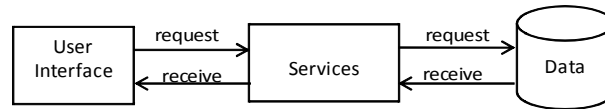


Fig 1a: 3-Tier Enterprise Application Model

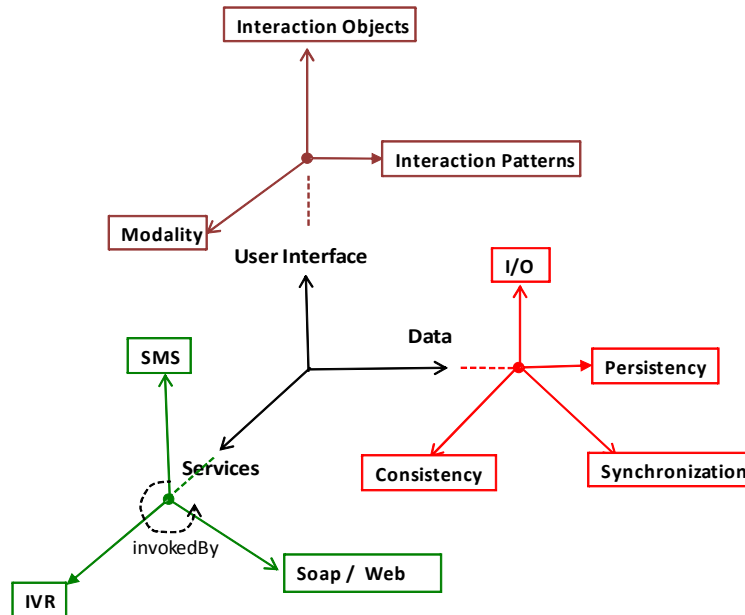


Fig 1b: Multi-dimensional view of a Business Application

3 Mobile-Enabling EBAs

For any enterprise that offers a set of services on the desktop platform, the very first question that comes into mind is - *How do I transform my existing enterprise application to be mobile enabled?* Following this question, the set of related questions are:

- How can the transformed application be deployed on any mobile platform or device?
- How to provide the best user experience on the target platform or handset?
- How to deliver mobile applications that are simpler to build, quicker time-to-market and relatively easy to maintain?

To answer the above questions, we reflect into our past software engineering practice and experience. For the past several years, we are using model-driven engineering (MDE) techniques to address the need of developing enterprise application from high-level models [5]. Models capture the design and specification of any application and raise the level of abstraction from code-centric way of application development to model-centric way of application development. Fig 2 presents our current MDE technique for constructing n-tier enterprise business applications (EBA). It shows three primary constituents (i.e., Data, UI and Services) of an EBA grouped across three principal technical spaces – namely the specification space, the generation space and the implementation space. Out of these three technical spaces, the specification space is used to capture the high-level specification of an application whereas the generation space is composed of model-based code generators that take the model specifications as input and produces the low-level implementation as output. Finally the target platform compiler(s) compiles all the generated artifacts which are then packaged, deployed and run on the runtime system.

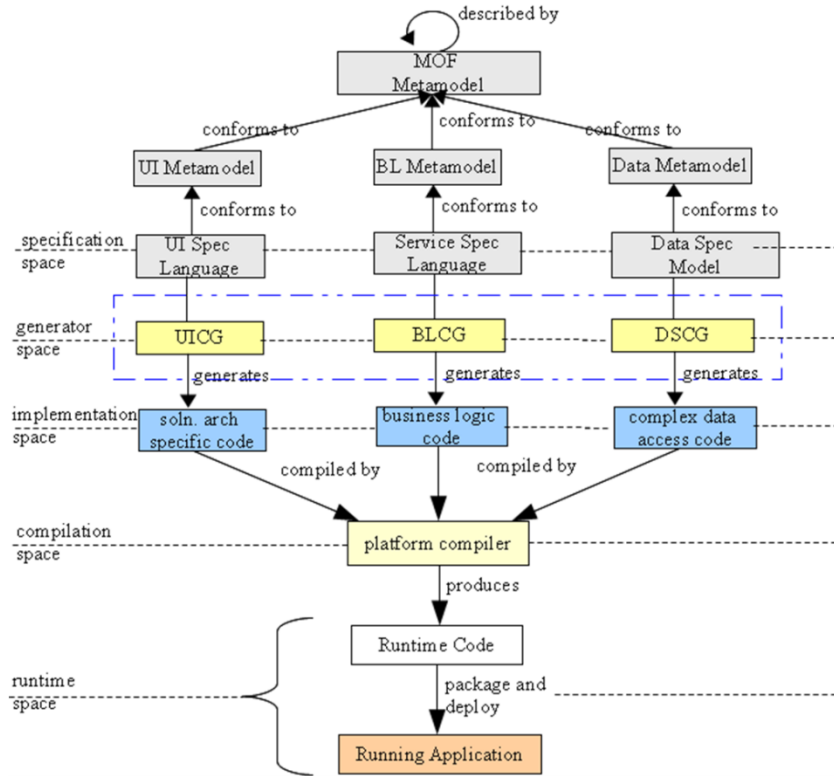


Fig 2: Building Enterprise Applications using MDE Techniques

We believe the principles, the techniques and the methodologies that we used in building EBAs can well be applied to mobile-enable business applications. From our initial experience in manually transforming EBAs to be mobile-enabled, we have observed that for the same application, the functionality across the data and the services layer generally do not change from one mobile platform to another. However, due to the variability in the UI implementation space across various mobile platforms, one needs to redevelop the entire UI part of the application. Especially, in case of native UIs (e.g., Java in Android, ObjectiveC in iOS), that generally offers a richer user experience than web-based UIs (i.e., using webkit and embedded browser), the problem is several times compounded. For example, Android supports several widgets that are not supported in iOS (e.g., ¹*checkbox*). Like this, there are several other variations in the native UI layer across mobile platforms. These delta variations in the UI specification for the same application are depicted in Fig 3 as *UI Spec α* and *UI Spec β* . Thus, in order to capture such Δ variations in the UI layer, the individual platform specific code generators are configured via a parametrization process that takes into consideration the abstract UI mapping models across multiple mobile platforms. The bottom of Fig 3 shows the corresponding platform specific applications that were generated from the same application specification.

4 Conclusion

There are several challenges in building mobile applications catering to different functional and non-functional requirements of a business application (e.g., sensor management, energy management etc). However, in this position paper, we tried to address a particular challenge in mobile-enabling existing EBAs by reusing most of existing artifacts (e.g., data and services) by leveraging and extending existing model-driven software engineering techniques. As a proof-of-concept, we validated our ideas on the Android and iOS platform. However, there are many open issues (e.g., native vs. web) when building mobile applications that requires further exploration of other best practices in software engineering, which can evolve as a new discipline of its own.

¹ Instead the *checkbox* functionality can be captured via a *switch* in iOS

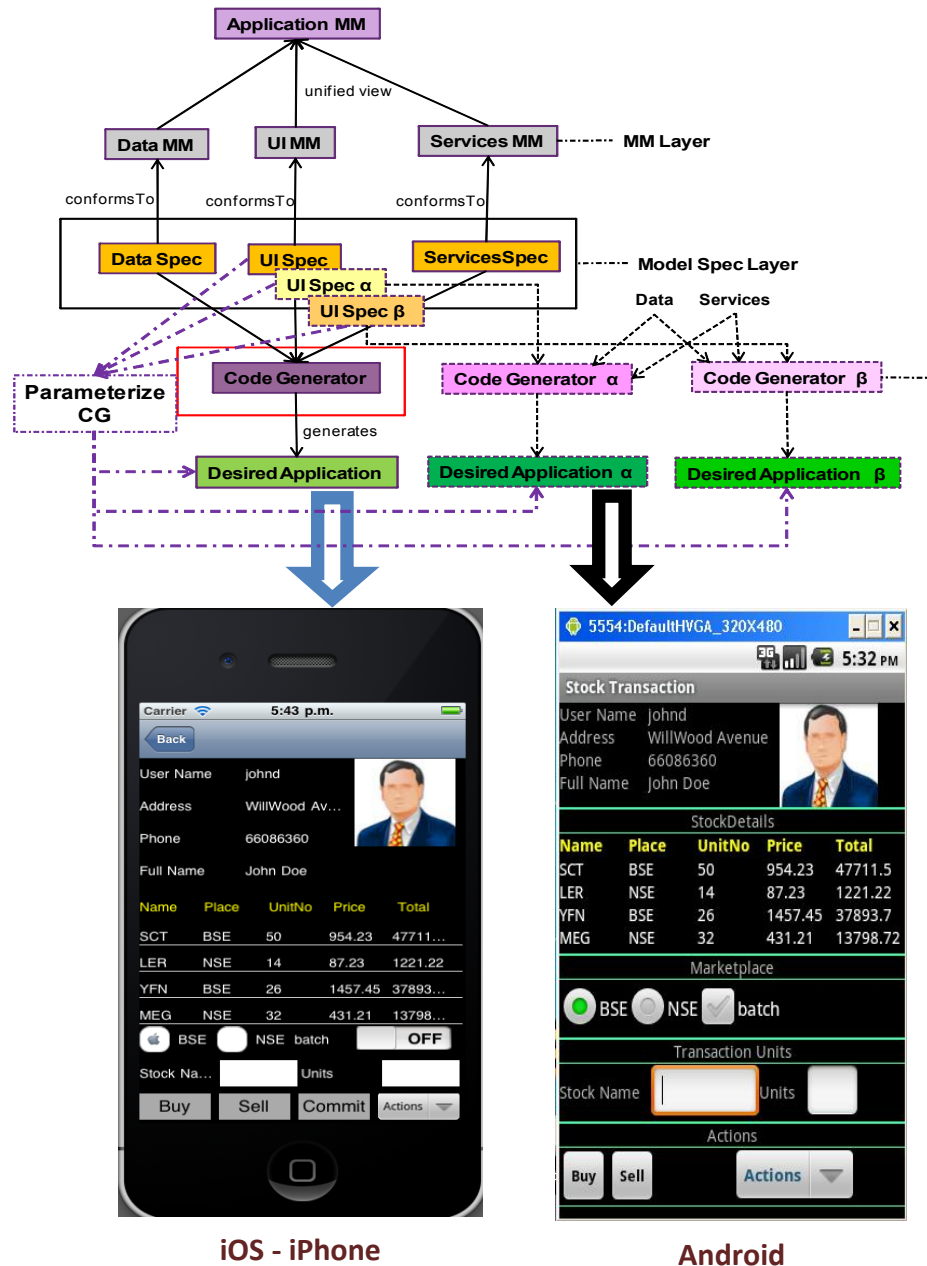


Fig 3: Mobile-Enabling Enterprise Business Apps

References

1. Tor-Morten Grønli, Jarle Hansen, Gheorghita Ghinea "Android vs Windows Mobile vs Java ME: a comparative study of mobile development environments", proceedings of the 3rd International Conference on Pervasive Technologies Related to Assistive Environments, June 2010.
2. David Szczesny, Sebastian Hessel, Felix Bruns, Attila Bilgic, "On-the-fly hardware acceleration for protocol stack processing in next generation mobile devices", proceedings of the 7th IEEE/ACM international conference on Hardware/software codesign and system synthesis, October 2009.
3. Michalis Zarifopoulos, Anastasios A. Economides, "Evaluating mobile banking portals", International Journal of Mobile Communications, January 2009, Volume 7 Issue 1.
4. Key Predictions for IT Organizations and Users in 2010 and Beyond, Gartner Report, January 2010, <http://www.gartner.com/it/page.jsp?id=1278413>.
5. Vinay Kulkarni, R. Venkatesh and Sreedhar Reddy. *Generating enterprise applications from models*. Object-Oriented Information Systems, 2002, LNCS 2426, pp. 270-279.