Chapter 23
Multichannel Service Delivery Architecture: A Case Study

Randall E. Duran
Catena Technologies Pte Ltd, Singapore

Anh Duc Do
Catena Technologies Pte Ltd, Singapore

ABSTRACT

System architectures that deliver real-time services to customers must be flexible, scalable, and support a wide range of communication channels. This chapter presents an architecture that was designed to support multiple delivery channels and was successfully used to implement mobile banking services. The considerations behind the design and the approach used to deliver SMS-based mobile services using service-oriented architecture principles are reviewed and some of the practical challenges that were encountered with the implementation are explored. The ability for this solution architecture to support other real-time service channels is also examined.

INTRODUCTION

The proliferation of real-time communication channels has provided businesses with many new ways to reach customers; however, integrating and managing communications across these channels has also become a challenge. Over the past decade, Web sites, email, SMS messages, and more recently via mobile applications have become common means for businesses to communicate with their customers. This chapter presents the design of a multichannel service delivery architecture based on service oriented architecture (SOA) principles that was successfully used to implement delivering SMS-based services as part of a mobile banking implementation.

A critical consideration when providing real-time services to customers is selecting the channels that will be supported. Some of the options include Web browsers, email, short message service (SMS), and native mobile phone applications. As might be expected, each technology has its strengths and weaknesses. Web browsers are ubiquitous, provide a rich environment for
presenting information, and support user-driven interactive communication, but they do not readily support push notifications and the quality of presentation can vary significantly depending on the device on which they are run. Email is available on standalone computers and as well as smartphones, but may not be viewed by the recipient, if he or she is not actively monitoring the email account and may be accidentally treated as spam. SMS is pretty much guaranteed to reach the recipient’s mobile phone quickly and likely to be viewed; however, the length and presentation of the information delivered is quite limited. Native applications that run on mobile phones have presentation that is optimized for the device, provides two-way interactive communications, and can pop up short notifications that are visible when the application is not in the foreground. Nevertheless, native mobile applications are tightly coupled with the mobile phone hardware and operating system environments, making development and deployment more cumbersome. Likewise, the user must first download and run the application to enable the communications channel.

Another important consideration for providing real-time services is the architecture strategy used to implement the selected channels. Core systems, such as ERP or banking platforms, may provide their own proprietary gateways that allow customers to access services over multiple channels, including mobile devices. Vendors also offer channel gateway platforms that can provide common support for specific channels to multiple back-end applications. In general, both types of these off-the-shelf solutions are easier to implement, but are less flexible and are dependent on the capabilities that the vendor provides. Alternatively, channel implementations can be created on top of open, service-oriented development environments in conjunction with other business applications. This approach can provide solutions that are better catered for specific business needs, but will require substantial development capabilities.

The solution approach chosen will depend on the number of channels that must be supported and the complexity of the systems environment to which they will connect. On one hand, using off-the-shelf solutions may be suitable for companies that have a relatively simple systems environment, require less flexibility, or do not have the resources to support customized solutions. On the other hand, companies that have more complicated needs, have taken an open systems approach, and have more significant development capabilities may be better served by homegrown solutions.

The solution described in this chapter delivered banking services to retail customers using the SMS mobile channel, and was built upon an open SOA platform. SMS was selected as the initial channel to be implemented due to its broad appeal and accessibility. The channel implementation was built on top of a SOA development platform that was already available and had been successfully used to implement Web-based channels. This previous implementation provided the bank with existing business services that could be reused for the mobile channel, and development expertise with the SOA channel platform.

This chapter reviews the engineering methods used to design and deliver SMS-based mobile services using SOA and examines the challenges that were encountered with the implementation. It presents a practical and proven design that can be reused as the basis for other types of channel delivery implementations in the future. Specifically, it considers how the design can be extended to support native, thick-client mobile applications.

The structure of the chapter is as follows. The next section reviews prior research related to SOA-based mobile channel implementations and multichannel services architectures used in banking. The third section describes the solution’s business context and highlights key technical considerations. The fourth section provides an overview of the design of the SMS channel services. The fifth section reviews challenges and concerns
Related Content

**Integrating Semantic Web and Software Agents: Exchanging RIF and BDI Rules**
[www.igi-global.com/article/integrating-semantic-web-software-agents/55062?camid=4v1a](www.igi-global.com/article/integrating-semantic-web-software-agents/55062?camid=4v1a)

**Experiences with Requirements Model Reuse: The NFR Catalogue for Ubiquitous Systems**
[www.igi-global.com/chapter/experiences-requirements-model-reuse/66468?camid=4v1a](www.igi-global.com/chapter/experiences-requirements-model-reuse/66468?camid=4v1a)

**Formal Semantics of Dynamic Constraints and Derivation Rules in ORM**

**A Method for Detecting Bad Smells and its Application to Software Engineering Education**
[www.igi-global.com/article/a-method-for-detecting-bad-smells-and-its-application-to-software-engineering-education/122789?camid=4v1a](www.igi-global.com/article/a-method-for-detecting-bad-smells-and-its-application-to-software-engineering-education/122789?camid=4v1a)