Development of the Method for Integration of Mobile Applications and Corporate Information Systems

Pavel Sergeyevich Ptitsyn
Research Institute of Semiconductor Engineering, JSC
394033, Voronezh, Leninsky Prospekt, 160a. Russian Federation

Dmitry Vladimirovich Radko
Voronezh Innovation and Technology Center, LLC
394033, Voronezh, Leninsky Prospekt, 160a. Russian Federation

Alexey Vasilevich Skrypnikov
Voronezh State University of Engineering Technologies
394036, Voronezh, Revolyutsii Prospekt, 19. Russian Federation

ABSTRACT: In the design and development of mobile applications, there are significant technical and technological difficulties associated with the integration of these applications into a single information landscape of the enterprise, and supporting for various versions of mobile operating systems. The above-identified difficulties affect the cost of mobile application development as well as the cost of ownership of these systems. In addition, there is no rapid response to new market challenges and perspectives for business development. The aim of this work is to develop a flexible method for seamless integration of mobile applications and corporate information systems. The proposed method gives solution for integration using visual programming approach, which provides a unified model for design, development, and deployment of mobile applications for different mobile platforms as well as providing flexible data models and scenarios for integration with a wide range of data sources and corporate information systems.

Subject Categories
[C.1.4 Parallel Architectures]: Mobile processors; [C.2.1 Network Architecture and Design]: Wireless Communication; H.4 Information System Applications

General Terms
Mobile Applications; Information Systems; Corporate Communication; Enterprise Design;

Keywords: Mobile applications, Enterprise integration, Cross platform development

Received: 13 May 2016, Revised 20 June 2016, Accepted 4 July 2016

1. Introduction

Currently, mobile devices in terms of functionality are not inferior to personal computers, which opens up a wide field of application for mobile devices in various spheres of human activity. For efficient usage of mobile devices in specific spheres of life, it is necessary to develop the appropriate mobile applications, which implement the desired functionality. This circumstance gives rise to a demand for the development of various mobile applications.
Currently, mobile devices are manufactured on a variety of mobile operating systems, such as Android, iOS, Windows Phone. Each of the mobile operating systems in the mobile application development requires the use of their own development tools and programming languages. Development of mobile applications separately for each operating system is a fairly time-consuming and expensive solution to this problem [1, 2].

In addition, there are significant technical difficulties integrating mobile applications in the information landscape of the enterprise, because there are no flexible infrastructure for the integration of cross-platform mobile applications and corporate information systems (CIS) [3]. Using specialized software such as Enterprise Service Bus (ESB) or Enterprise Application Integration (EAI) does not solve the problem because these approaches have significant disadvantages, among them should be noted [4, 5]:

• There is no support for a variety of mobile platforms and features for application development for giving platforms.

• There is no support for design patterns of distributed enterprise applications, which simplify development of mobile applications.

• There is no support for configuration of data models and scenarios of business processes at the level of mobile applications.

• The high cost of ownership and implementation of integration solutions.

The aim of this work is to develop the method, which provides interaction between corporate information systems and mobile platforms, and fully ensures the achievement of the required specifications in terms of performance, reliability, security and scalability. In addition, the method should provide a cross-platform technology for creating mobile business applications in terms of business functionality, and provide high flexibility for different mobile platforms.

2. Methods

2.1 General schema of seamless integration

Seamless integration is the process where a new module or feature of an application is added or integrated without resulting in any discernable errors or complications. It simply means that whatever change is being applied to a system, it happens without any negative impact resulting from the integration. In the case of the integration of mobile application and CIS, seamless integration provides data modification and execution business logic between mobile applications and CIS [6, 7].

The basis of the method to implement the seamless integration of mobile applications and CIS are the following principles [8]:

• Using of visual programming approach. Visual programming provides integration by manipulating graphical objects instead of directly writing the code. This reduces development time, because the process of creating an integration between applications becomes automated, and it reduces the cost of development, as it requires the involvement of employees with lower skill levels.

• Using the universal interfaces of interaction. The mobile applications interact with a variety of CIS of universal methods and algorithms, which provides a mechanism for data transmission in different ways, depending on the integration technology approaches.

• Using of cross-platform mobile application development technologies. Cross-platform technology provides the creation of universal design model for different mobile platforms, which reduces the effort and cost of mobile development.

The proposed method of seamless integration contains the following basic stages [9]:

• Analysis of enterprise data sources, which provide information on the integration features and functions from CIS.

• Creating models for interacting mobile application and CIS. The generation of scripts and schemas which implement logic for communicating mobile application with CIS.

• Data conversion by Proxy-server. Proxy-server performs formatting of data transmitted and received between the mobile application and the services of CIS.

The method of seamless integration incorporated interaction between the following components [10]:

• The designer of mobile applications. The designer enables the creation of mobile applications by implementing the concept of visual programming. The process of creating mobile applications is fully controlled by the user.

• Proxy-server. The proxy provides an interface between the mobile application and CIS. Proxy-server exposes a unified interface to access mobile applications to the services of CIS, and it is universal for all mobile applications created by the designer. Proxy-server includes a number of tools that enable the server to communicate with the services of CIS through various common technologies.

• The generated mobile application. The result is the assembly of mobile application, which optimized for defining mobile platforms.

When communicating with CIS, Proxy-server analyzes CIS integration capabilities. The analysis results transferred to the designer of mobile applications for subsequent generation of the workflow interaction with Proxy-Server. The generated mobile application performs all requests to Proxy-server for a unified interface, passing the
parameters required to make the request to CIS. Proxy-server in turn queries CIS, using existing technology of CIS integration. Accordingly, given approach provided the integration of mobile applications with CIS [11].

The process of generation mobile application using this approach is represented in Figure 1.

During changing user data, the mobile application is making the request to change the data to the Proxy-server. Proxy-server by using information about the integration capabilities of CIS converts the data into the required format. After the conversion of data is processed, Proxy-server performs the connection to CIS. Then Proxy-server performs the request for data change, which is specific to a particular data source. All necessary metadata for the formation of the request are transmitted from the mobile application. This metadata includes a URL to access the service, the name of the performed procedure, and the parameters transmitted during the procedure call. The values of parameters are modifiable user data, other related information specified in the design time using the designer of mobile applications. Further, Proxy-server makes a request to CIS services and waits for a response from it. Proxy-server converts the response, if it is necessary. The converted data is transmitted as the response to the mobile application.

The flow diagram of the operations of CIS data is represented in the Figure 2.
2.2 Analysis of enterprise data sources
During analysis of enterprise data sources, it collects all the necessary information about the integration capabilities of CIS. The analysis is performed on Proxy-server side after establishing the connection with CIS. The result of the analysis is information about the available data types and their formats, as well as list of operations over them. Then the obtained data used to create communication models between mobile applications and CIS.

When adding a new CIS as Web service, Proxy-server performs a general request to the Web service to obtain information about it. The response contains the following data in WSDL-format [12, 13]:

- Definition of data types (types), which describe the types of messages.
- Data elements (message), which describe the data of web service.
- Abstract operations (portType) contain the list of operations that can be performed with messages.
- Binding services (binding), which describe the way of message delivery.

The service description is enough to form the basis of requests to perform operations provided by the CIS Web service. The structure of the web service description in the WSDL format is represented in Figure 3.
These structures contain the description of the business entities of web services. The field of the object is the definition of business entity, and the field values are the values of the business entity. Building a hierarchical structure performed for the analysis of each object and identifying it has references to other objects. After identifying, such links search facility to link and aggregate data objects.

The next step is the process of converting an object hierarchy in a JSON-formatted string. This format is most appropriate for the storage and transfer of structured information on the client-server architecture.

The resulting conversion data, if necessary, are available for the designer of mobile applications. The diagram of the implementation stages of the analysis of CIS Web services is represented in Figure 4.

Figure 4. The diagram of the implementation stages of the analysis of CIS Web services

### 2.3 Designing mobile applications

Creating mobile application is carried out by means of the designer of mobile application. The designer implements cross-platform web development environment that provides design and development of cross-platform mobile applications using JavaScript programming language and HTML5 technology [14]. The designer provides the following technical features:

- Designing a graphical user interface for the corresponding mobile device (smartphone, tablet, desktop) and operating system (iOS, Android, Windows RT, Windows Desktop, Linux) with the help of the visual designer.
- Configuring the data model of the application in graphical form.
- Compiling the application for a corresponding mobile platform.
- Debugging the mobile application.

The designer of mobile applications provides creating projects of mobile applications using the concept of visual programming. This concept implies a possibility to create applications by manipulating graphical objects instead of writing the application source code. This simplifies the process of developing mobile applications because this concept does not require the involvement of professional developers for creating the applications.

The user projects are stored in the MongoDB database. The server side of the designer module is implemented based on Node.js platform. The interaction between the designer module and the database carried out through the usage of the Mongoose data accessor, implementing the principles of object-relational mapping. The mechanism of interaction between client and server side of the designer module is built on the implementation of the CRUD-approach [15].

The architecture of developing mobile applications is built on MVC pattern. MVC pattern separates user data, user graphical interface, and processing of user actions to three separate levels [16]:

- Model, which represents the underlying, logical structure of data and the high-level class associated with it.
- View, which is a collection of classes representing the elements in the user interface.
Controller, which represents the classes connecting the model and the view, and is used to communicate between classes in the model and the view.

For building View component, the designer of mobile applications provides a wide set of basic visual components for data presentation. View component implemented by combining and building hierarchies of the basic components for data visualization. An example of the hierarchical structure of the components for data visualization is represented in Figure 5. The components for data visualization are divided into two types: simple component and containers. The simple components are responsible for data visualization and data exchange. The functions of containers are placement, positioning and organization of the links between the other components.

Model component is a set of fields of various types and proxies, which perform the queries during changing and initializing the fields. The building Model component provides without writing the code. Because the models have the same type of structure, their creation performs by filling the related form of user interface. In the process of creating the model, it performed the description of model fields, the definition of operations on the model, and the binding the operations of the model to the operations of CIS. For binding operation applies the information, which obtained from the analysis of the integration capabilities of CIS (for details, see section 2.2 Analysis of enterprise data sources). Model Designer provides creating the data models.

Controller component contains a set of actions for performing operations on the models. For setting actions can be applied set of templates, which bind the events with related data parameters of visual component. The visual components have parameters. The parameter assigns values to related data models, which designed using Model Designer. Such a way performs the setting of interconnection between the Model and the View.

2.4 Code generation of mobile applications
The technology of cross-platform development, applied for generating code of mobile applications. According to this technology, mobile application wraps a web-application that displays local content and can interact with the
functions of the mobile device. The generated content of web-application put in the local storage of mobile applications.

The developed applications are not just ordinary mobile web sites. They are able to interact with mobile services such as GPS, accelerometer, address book, which are not available for web applications. These mobile applications are built using the SDK of mobility platforms, and use all features of these platforms, including publications in Apple AppStore or Android Market [17, 18]. The structure of the cross-platform mobile application is represented in Figure 6.

Cross-platform mobile application has the following components:

- Web browser component.
- JS-Bridge.
- Native-Bridge.

Web browser component displays a graphical user interface, and perception of controlling by user’s gestures. JS-Bridge is a component, which connects the web browser and the mobile platform, and provides receiving notifications from the mobile platform. Native-Bridge is a component, which connects the web browser with the mobile platform, and provides data and notifications from the web browser to the mobile platform. JS-Bridge and Native-Bridge are implemented as web browser plug-ins, which provides interaction with the mobile platform. The sequence diagram of data exchange between the application components and the mobile platform is represented in Figure 7.

The code generation of mobile applications uses the configured MVC-components, which were designed using Model Designer. The configured MVC-components are stored in a structured JSON format. For more information, see Chapter 2.3 Designing mobile applications.

In the code generation process of mobile applications uses the following application libraries: SenchaTouch, Ext.JS, Apache Cordova, and SDK for the mobile platform for which to generate the application. The process of the code generation mobile applications consists of several stages [19]:

- Generation of the source code. Using configured objects of visual components and the data models, proceed automatic code generation process. The generated source code written in JavaScript programming language. The generated source code contains the objects and the components, which are linked within the specific application libraries and SDK.
- Formatting the file structure of the generated source code. At this stage, the intermediate file structures of the project mobile application are redefined. In addition, it provides attachment to the application project linked
libraries, styles, resources.

- Generation of the installation package. Generation of the installation package is done through the usage of Apache Cordova libraries. These libraries participate in the generation of applications using the SDK mobile platforms for a variety of mobile operating systems.

The Structural diagram of the generation of mobile applications is represented in Figure 8.

The result of the generation of the mobile application is installation packages for the defined mobile platforms.

The installation package is able to be downloaded from the Application Store or installed directly to mobile devices. The mobile applications have identical graphical user interfaces for a variety of mobile platforms.

The project of mobile application is stored in the database on the server side. This makes it easier the processes for changing, modifying, and rebuilding mobile applications.

Because of the process of mobile application development does not require qualified software developers, it significantly reduces the cost and the time required to integrate with CIS.

2.5 Execution of queries by Proxy-server
Developed mobile applications perform queries to the Proxy-server, which provides interaction with CIS systems. The interaction between the mobile application and Proxy-server is performed through a unified interface [20, 21]. The parameters of query passed the following information:

- Name of the source of corporate data.
- The URL to access the web service.
- Name of the execution procedure.
- Array of the data elements, which contain input parameters for the execution procedure.

This set of the data is required to identify the source of corporate data, and provide correct execution queries for it. The resulting application data are available in JSON format, and they do not comply with the request to CIS services. Therefore, it is necessary to convert data into the acceptable data format for the service.

The result of the execution procedure is converted, to the JSON format, and transmitted as a response to the request of mobile application. A mobile application from the
resulting response from Proxy-server defines how to interpret the data. The generated code is built on the interpretation of the data mapping of model’s fields and output parameters of the procedure.

The mobile application interacts with Proxy-server using HTTP-requests. These requests used CRUD principle, according to which data can be executed using four types of operations: Create, Read, Update, and Delete. Consequently, for each of these types of operations are performed specific HTTP-request.

The diagram of query execution between mobile application and Proxy-server is represented in Figure 9. This approach applies Proxy-server, which creates a mobile application using Model Designer, and provides independence from data formats of CIS systems.

3. Results and discussion

The developed method for integration of mobile applications and corporate information systems provides minimized efforts for the creation of mobile applications in comparison with standard development tools. It is achieved through applying the principle of the universality programming code in relation to the mobile platform, which makes easier software development process on a variety of mobile operating systems.

For designing and development of mobile applications is applied the special tool (Application Designer), which minimized the resources to write the source code to interact with CIS systems. Application Designer has an ergonomic graphical user interface that enables the development of mobile applications for end-users without relating expertise in development of mobile business applications. Thus, the use of the technology will reduce the cost of system integration of mobile applications into the single enterprise information landscape.

The developed technology of the proposed method for seamless integration of mobile applications and corporate information systems provides the following key features:

- Support for business users, includes the ability to design screen forms and business logic based on template and visual approaches without the knowledge of programming languages.
- Providing a single distributed development environment, which automated the processes for development, building, and publishing of mobile applications. It is a significant advantage for reducing the cost of software development and support process.
- Support for industry standards, such as SOA, Web Services, REST, SOAP, and JSON.
- Support for integration adapters for such corporate
• systems as Microsoft Dynamics, SAP Business One, 1C:Enterprise.

• Support for mobile platforms such as Android, iOS, Windows Phone, Windows RT.

The implementation of method for seamless integration, mobile applications and corporate information systems provides the following technical parameters:

• Reducing the time for developing mobile applications more than 3 times in comparison with the use of standard development tools.

• Reducing the cost of system integration of corporate systems and mobile applications more than 3 times in comparison with the use of integrated software, such ESB, EAI.

• Reducing the cost of ownership for mobile applications more than 2 times in comparison with the use of standard ways of mobile application support.

• Increasing reliability and availability of applications with proven enterprise application platforms.

4. Conclusions

The proposed method for seamless integration of mobile applications and corporate information systems provides mechanisms for integration and mobile applications and CIS on the following principles: usage of visual programming approach, usage of the universal interfaces of interaction, usage of cross-platform mobile application development technologies. The process of creating mobile applications and their integration with CIS contains the following basic stages: analysis of enterprise data sources, creating models of interaction mobile application with CIS and related user interface forms, data conversion by Proxy-server.

The implementation of proposed methods for integration of mobile applications and corporate information systems provides the following advantages:

• Reduce the cost of mobile application development.

• Reduce the cost of system integration of mobile applications in the enterprise information landscape.

• Reducing the cost of ownership for mobile applications.

• Ensuring a high level of security, reliability, and scalability of mobile solutions.

• Provide flexible management capabilities and support for mobile business applications.

Acknowledgments


References


