

Customizable Online Application Development Environment and Online Marketplace System

Byeong-Thaek OH*, Sung-Ik JUN*, Sung-Jin HUR*

* ETRI(Electronics and Telecommunications Research Institute), Korea

{btoh, sijun, sjheo}@etri.re.kr

Abstract— In this paper, we present a customizable online application development environment and an online marketplace system. We applied development resource virtualization technology in order to support multiple developers who create application with an each customized development environment.

Keywords—Online Development Environment, Multi-tenant, SaaS, Software-as-a service, Online Configuration

I. INTRODUCTION

In the past, software had to be installed in an infrastructure close to end users. Since early 90's there has been a rapid advance in the internet in terms of speed, connectivity and reliability. With the internet becoming ubiquitous, coupled with the vendor's interest in capturing the market of small customers who could not afford the expensive enterprise software, either on-premise or through Application Service Provider (ASP), fuelled the advent of Software as a service (SaaS). SaaS is a software delivery paradigm where the software is hosted off-premise and delivered via web and the mode of payment follows a subscription model. With SaaS, the customers' IT people had access to a number of different features or capabilities within the software set and could configure them via a web interface to fit the needs of their company. A true SaaS model adhered to the multitenant architecture, in which all users and applications share a single, common infrastructure and code-base that is centrally maintained. The fine configuration allowed this single-instance multiple-tenant model. Thus, even though there was single instance of the software running, the customers got a unique application depending on the configuration done by them. Because SaaS vendor clients are all on the same infrastructure and code-base, vendors can innovate or update the application more quickly and save the valuable development time previously spent on maintaining numerous versions of outdated code[1]. In addition, In a conventional way, software developers have made many efforts to establish a development environment by selecting and combining proper tools among a variety of tools such as a software designing tool, an implementation tool, a testing tool, and a distribution tool, and in this case, in actuality, each developer needs to personally have a personal PC or a server and establish a development environment by purchasing an operating system, a WAS (Web Application Server), and the like, incurring a great deal of cost for establishing the

development environment. In this paper, we present a customizable online application development supporting environment and marketplace system which is applied metadata model technology in order to serve multiple tenants with a web based online system.

II. RELATED WORK

Software developers have made many efforts to establish a development environment by selecting and combining proper tools among a variety of tools such as a software (SW) designing tool, an implementation tool, a testing tool, and a distribution tool, and in this case, in actuality, each developer needs to personally have a personal PC or a server and establish a development environment by purchasing an operating system, a WAS (Web Application Server), and the like, incurring a great deal of cost for establishing the development environment. Recently, online development environments have been provided by Salesforce.com, Google, Microsoft, and the like, but these online development environments are very limited and do not support the reuse of developed codes. In detail, in the case of Salesforce.com, Development as a service (DaaS) supports a Web-based Force.com Builder and an Eclipse-based Force.com IDE as an online development environment for supporting a development of an application service operated on a DaaS platform called Force.com, and provides a testing server along with a development server. However, DaaS provides a very limited development environment, so the user cannot select and configure a development environment as a customized development environment and can neither download nor reuse a developed code. In the case of Microsoft, the Azure Services Platform provides a development environment by using an SDK accessing Windows Azure called Azure Runtime to support a cloud computing and general application SW development [2]. However, the user of Azure is not allowed to select and configure a customized development environment, is not allowed to reuse a developed application code, and is not provided with a marketplace for operating applications developed in various environments. Also, CodeRun Studio is a cross-platform Integrated Development Environment (IDE), designed for the cloud. It enables you to develop, debug and deploy web applications using web browser. CodeRun features native support for C#/.NET (3.5),

PHP (5.1), JavaScript, HTML and CSS. Database support includes SQL Server 2005 and Amazon SimpleDB [3]. However, the user of CodeRul also cannot reuse a developed application code, and is not provided with a marketplace for operating applications developed in various environments.

III. SYSTEM DESIGN

We designed a multi-tenant supporting online development environment supporting system based on metadata model, shown as Figure 1, includes an Online Development Environment Server and the devices connected with it. The server follows the multi-tenancy platform model in order to provide the advantages of SaaS platform, the opportunity to be economically efficient along many dimensions through multi-tenancy, in which all users and applications share a single, common infrastructure, with platform provider and tenants.

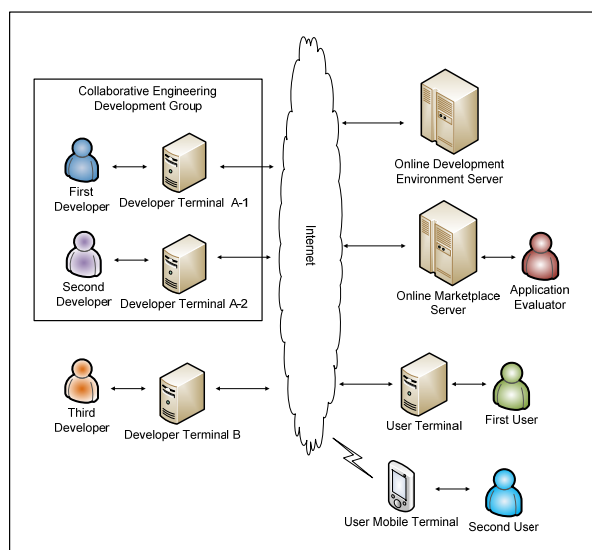


Figure 1. Online Development Environment Server and users

Figure 1. is a view illustrating an overall online development environment system. An online development environment system may be configured to include an online development environment server, an online marketplace server, one or more developer terminals, and one or more user terminals, and these elements are connected through the Internet, or the like. The online development environment server serves to provide a customized software development environment to a software developer online. A detailed configuration and function of the online development environment server will be described in detail with reference to Figure 2.

The online marketplace server serves to operate an application developed in a development environment provided by the online development environment server. A detailed configuration and function of the online marketplace server will be described in detail with reference to Figure 3. One or more developer terminals are connected to the online development environment server through the Internet to

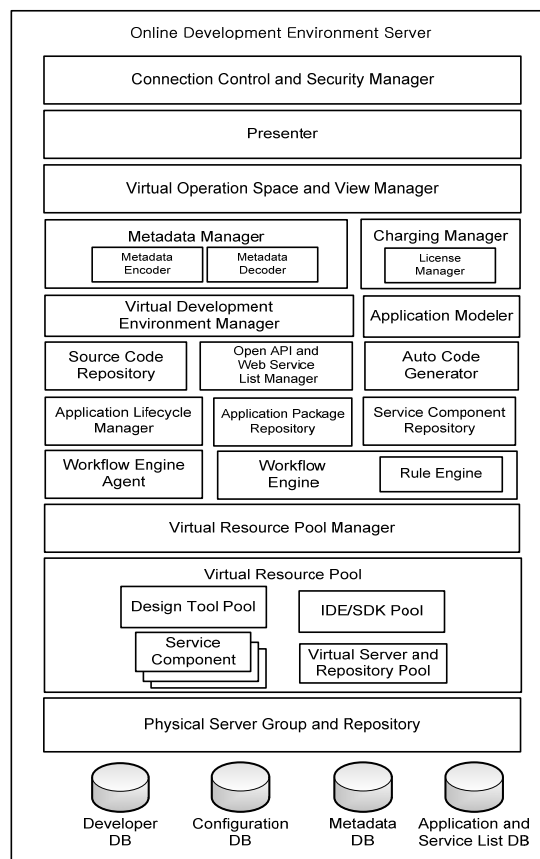


Figure 2. Architecture of Online Development Environment Server

provide a customized software development environment to developers. In other words, the first to third developers may be connected to the online development environment server through the developer terminals respectively, to configure a customized development environment fitting their requirements and then develop software. In particular, the first and second developers may be connected to the online development environment server through the developer terminals and collaboratively perform a development task. Also, in developing an application, the first to third developers may develop a new application by using already developed components in the development environment provided by the online development environment server. The one or more user terminals are connected to the online marketplace server through the Internet to provide a developed application to the user. In other words, the first and second users are connected to the online marketplace server through the user terminals respectively, to selectively use an already developed application or a desired application of a Web service, or a Web service, and if necessary, the first and second user may download an application package through their user terminals dispose the downloaded application package in a personal server, and operate it.

Figure 2. is a detailed block diagram of the online development environment server. The connection control and security manager manages controlling of a user connection, security on a communication channel, and security of the

overall online development environment server when a developer uses an online development environment through the Internet. Also, when the developer logs in, the connection control and security manager handles a developer authentication. The presenter serves to present a virtual view model configured by being bound by the virtual operation space and view manager to the developer. The virtual operation space and view manager binds a view of a virtual development environment of each user and each group through the virtual development environment manager and the virtual resource pool manager and provides the same. The metadata manager serves to store metadata constituted for virtualizing an operation space by using the metadata encoder and the metadata decoder and retrieve the stored metadata. In detail, virtual operation space environments and development environment configuration matters selected by the developer are stored and managed. Also, when a customized function for supporting a multi-tenant with a developed application is developed, a user interface, a database, a workflow, and the like, may be stored in the metadata DB by using the metadata encoder, and when an application is driven, a customized application may be configured by using the metadata retrieved from the metadata DB by using the metadata decoder. The charging manager performs a function of charging with reference to development environment and development tool license information and developer's contact information by using the license manager. The license manager manages the license of a development environment and development tool including WAS. The virtual development environment manager serves to establish a virtual development environment with reference to virtual resource information through the virtual resource pool manager within the online development environment server. The application modeler is a modeling tool provided by the online development environment server. The developer can model an application by using the application modeler.

The automatic code generator serves to generate a code with respect to an application modeled by the application modeler. The source code repository stores source codes, which have been generated by the developer by using the application modeler and the automatic code generator by version and manages the same. Also, the source code repository classifies the automatically generated code and source codes developed by the developer and stores the same to allow all the developers to use them in developing an application. The open API and Web service list manager manages an open API and Web service list provided by a Web-based server, and supports to allow the developer to apply the open API and Web service list in developing an application by using the virtual development environment or by using the application modeler. The application life cycle manager manages information regarding an evaluation, a disposition, and discarding in stages of developing each application. Detailed information of a corresponding application is stored in the application and service list DB and managed. Also, the same information as that managed by the application life cycle manager is also managed by the

application life cycle manager of the online marketplace server (to be described). The application package repository is a space for packaging source codes developed by the developer and storing and managing the same by utilizing the codes of the source code repository, and the service component repository is a space for packaging service components developed by the developer and storing and managing the same. After the packaging operation is performed, the developer transmits an application or a service component to the online market place server and requests its evaluation. After the application or the service components are evaluated by an evaluator in the online marketplace server, the service component approved for its operation may be disposed as a service component of the virtual resource pool of the online it in using a virtual development environment server so that the developer may apply it in developing a different application. In addition, the service component approved for its operation may be also disposed as a service component of the virtual resource pool of the online marketplace server so that applications developed by the developer by using the corresponding service components in the virtual development environment can be executed on a virtualized server. The workflow engine includes the rule engine. When the developer develops a process application employing a workflow and a rule, the workflow engine provides an API to the workflow agent to support the development. Like the foregoing service component, after the application, which has been developed by the developer by using the workflow engine agent in the online development environment server, is disposed in the online marketplace server, the developed application may be executed by using the workflow engine including the same rule engine installed in the online marketplace server and the workflow engine agent. The virtual resource pool manager serves to support to allow the developer to select a proper virtual resource in selecting and setting his development environment. The virtual resource pool includes the design tool pool, the IDE/SDK pool, the service component, and the virtual server and repository pool. In detail, the virtual resource pool provides the virtual server and repository pool by virtualizing the physical server group and repository and the operating system (OS), and constitutes the overall virtual resource pool by including the design tool pool and the IDE/SK pool on a server of each operating system along with the virtual server and repository pool. In addition, the service component supports an application development in a virtual development environment configured by the user along with the IDE/SDK. The DB serves to store various types of data required for the online development environment server. The DB may include the developer DB storing information regarding each developer, the environment configuration DB storing environment configuration information regarding a virtual development environment, the metadata DB storing metadata, and the application and service list DB storing detailed information regarding each stage of an application development.

Figure 3. is a detailed block diagram of an online marketplace server.

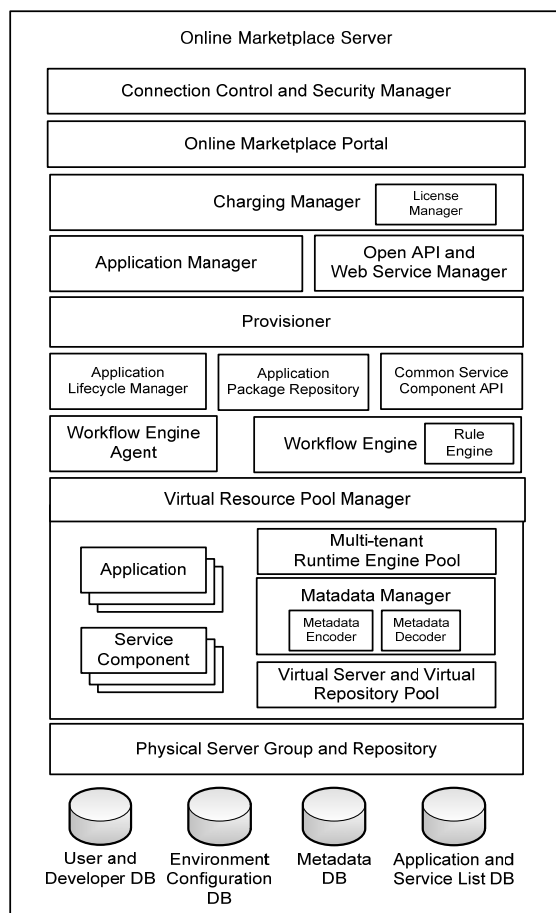


Figure 3. Architecture of Online Marketplace Server

The connection control and security manager manages controlling of a developer and user connection, security on a communication channel, and security of the overall online marketplace server when a developer and a user use an online marketplace through the Internet. Also, when the developer and the user log in, the connection control and security manager authenticates the developer and the user. The online marketplace portal serves to provide a function of allowing the user to check a list of application and Web service and make a contract to use a desired application and a desired Web service. The charging manager performs a function of charging with reference to license information of the application and Web service and user's contact information by using the license manager. The license manager manages the license of the application and the Web server. The application manager manages the list of applications approved to be operated after having been developed by the developer and then evaluated by an evaluator. The application manager supports to allow the user to check the application list through the online marketplace portal, and also supports to allow the user to download and use an application package if necessary, after making the contract. The open API and Web service manager manages a list of open API and Web services used by an application which has been developed by the developer and finally disposed, and a service status. Also, the open API and

Web service manager may perform a function of providing an open API and Web service to external servers by using service components disposed after being developed by the developer. The provisioner serves to dispose an application or service component, which has been approved to be operated by the online marketplace server, in the online marketplace server and allocate resources. After provisioning, the application is driven on the corresponding multi-tenant support runtime engine pool to support to allow multiple users and tenants to use the application. The multi-tenant support runtime engine pool may be configured to be different according to each virtual server, operating system, WAS (Web Application Server), or the like, so as to perform a function of providing a service by supporting multiple tenants with a single application instance. The application life cycle manager manages information regarding an evaluation, a disposition, and discarding in stages of developing each application. Detailed information of a corresponding application is stored in the application and service list DB and managed. Also, as discussed above, the same information as that managed by the application life cycle manager is also managed by the application life cycle manager of the online development environment server. The application package repository stores and manages an application and service package which has been developed by the developer and then transmitted to the online marketplace server, and also handles a function of supporting to download the application and service package through the application manager according to the user's contract. The common service component API is transmitted to and stored in the online marketplace server when the online marketplace server wants to support a development by adding a common service, among the service components developed by the developer in the online development environment server. The workflow engine agent and the workflow engine including the rule engine operate the same target (or object) as that operated by the online development environment server, and are used when an application developed in the online development environment server performs a process management function in the online marketplace server. The virtual resource pool manager handles establishing a virtual environment to allow the user to use an operating system and server according to a desired application. The virtual resource pool includes the application, the service component, the multi-tenant support runtime engine pool, the metadata manager including the metadata encoder and the metadata decoder, and the virtual server and repository pool. The metadata manager stores configuration information regarding a user interface, a database, a workflow, and the like, in the metadata DB through the metadata encoder when the user or the tenant performs customizing to use an application, and when the application is desired to be driven, the metadata manager supports to allow the application to be driven according to the configuration information through the metadata decoder. The DB serves to store various types of data required for the online marketplace server. The DB may include the user and developer DB storing information regarding each user and each developer, the environment configuration DB storing environment configuration

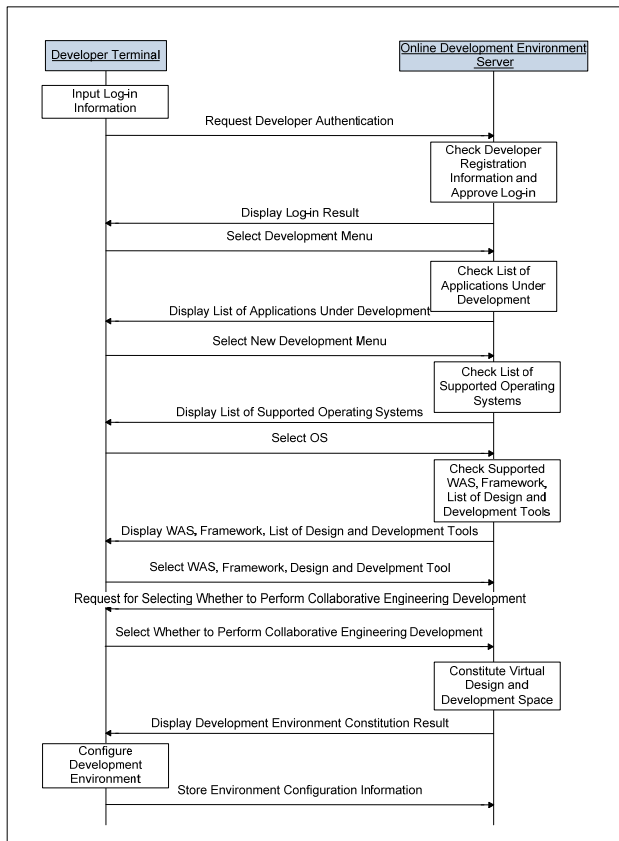


Figure 4. Process of configuring an online development environment

information, the metadata storing metadata, and the application and service list DB storing detailed information regarding each stage of an application development.

Figure 4. is a flow chart illustrating the process of configuring an online development environment using the online development environment server. With reference to Figure 4, first, when the developer inputs log-in information in the developer terminals request the online development environment sever to authenticate the developer. The online development environment server checks registration information of the developer with reference to the developer DB, approves the log-in, and displays the log-in results through the developer terminals. After log-in, the developer starts developing by selecting a development menu. Thereafter, the online development environment server checks the list of applications under development by the corresponding developer. If there is an application under development by the developer, the online development environment server displays the corresponding information through the developer terminals. When the developer selects the application whose development is desired to be proceeded with, a previously selected development environment is reconfigured. Meanwhile, when the developer selects a new development menu from the developer terminals and delivers the same to the online development environment server, the online development environment server checks the list of supported operating systems (OS) and displays the list of supported OS

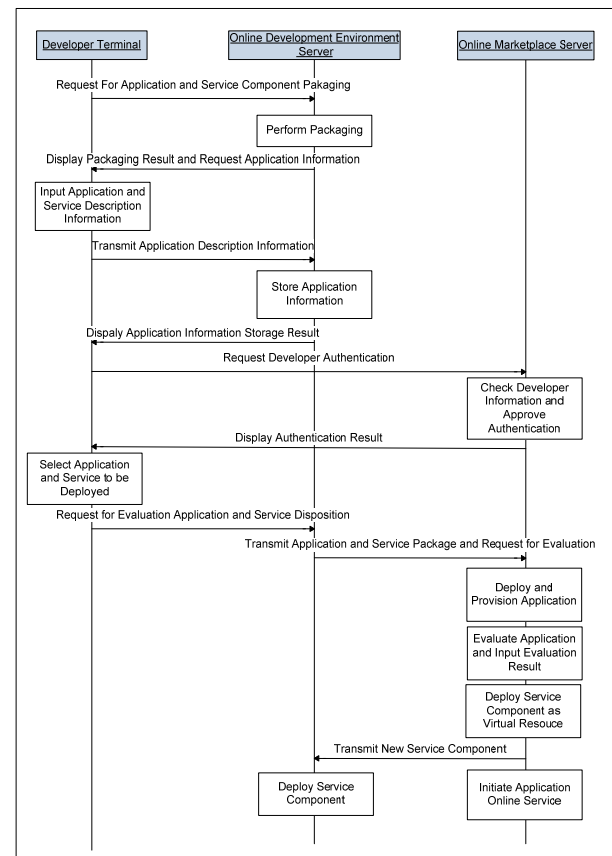


Figure 5. Process of providing a developed application

through the developer terminals. When the developer selects an OS from the developer terminals and delivers the same to the online development environment server, the online development environment server checks a supportable WAS, framework, and design and development tool list and displays the development environment through the developer terminals. Accordingly, the developer selects a desired development environment from the developer terminals and delivers the same to the online development environment server, selects whether to make a collaborative engineering development from the developer terminals according to a corresponding request from the online development environment server, and delivers the same to the online development environment server. Then, the online development environment server configures a virtual design and development space according to the developer's selected matters and displays the configuration results through the developer terminals. Accordingly, the developer terminals perform an environment configuration with respect to a development environment and store the environment configuration information in the online development environment server.

Figure 5. is a flow chart illustrating the process of providing a developed application by using the online development environment server and the online marketplace server. With reference to Figure 5, when the developer finishes a development of an application or a service component by

using the development environment provided from the online development environment server and request packaging of the application and service component from the online development environment server, the online development environment server performs packaging, displays the packaging results through the developer terminals and requests inputting of information regarding the application. When the developer inputs information regarding the application and the service component through the developer terminals and transmits the same to the online development environment server, the online development environment server stores the received information and displays the storage results through the developer terminals. Thereafter, when the developer wants to dispose the packaged application or the service component in the online marketplace server, the developer requests the online marketplace sever to authenticate the developer through the developer terminals. Then, the online marketplace server checks developer information, performs authentication, and displays the results through the developer terminals. When the developer, who has been authenticated, selects an application and a service desired to be disposed through the developer terminals, the developer requests the disposition and evaluation of the selected application and service from the online development environment server. Then, the online development environment server transmits the selected application and service package to the online marketplace server and requests its evaluation. The online marketplace server disposes the corresponding application and service package and provisions the same. Thereafter, the application is evaluated by an evaluator and the evaluation results are input. When the operation is approved according to the evaluation results, the new service component is disposed as a service component fitting the corresponding operating system, the WAS, and the like, in the virtual resource pool for operating the application in the online marketplace server. Also, the new service component is duplicated and transmitted also to the online development environment server and disposed as a service component of the virtual resource pool of the online development environment server, so as to be used when a different application is developed. Thereafter, the online marketplace server initiates an online service with respect to the approved application, so that the user can purchase and use it.

As set forth above, a customized development environment can be configured in real time according to various software development environment requirements of a developer in developing a new application, and an online development environment that can design, develop, utilize, store, and share software anytime and anywhere can be provided. In addition, re-usability of a developed software code can be improved. Thus, software productivity can be enhanced and a software production cost can be reduced.

IV. CONCLUSIONS

In this paper, we present a customizable online development environment that includes virtualized online development server that makes building proper development environment possible, online marketplace server supports scalability and flexibility. The online development server has several components. Among the components, Virtualized Workplace and View Manager support each development environment for an user or a group after binding virtualized view through the Virtualized Development Environment Manager and the Virtualized Resource Pool Manager. The Virtualized Development Environment Manager build virtualized development space after refer the information of resources with the Virtualized Resource Pool Manager in the online development server. The presented development environment can be configured in real time according to various software development environments. In addition, re-usability of a developed software code can be improved. Thus, software productivity can be enhanced and a software production cost can be reduced.

REFERENCES

- [1] Salesforce.com Corporation, "What is SaaS?", <http://www.salesforce.com/saas/>
- [2] <http://www.microsoft.com/windowsazure/>
- [3] <http://coderun.com/>
- [4] Y. V. Natis, "Reference Architecture for Multitenancy: Enterprise Computing in the Cloud," Gartner, Dec. 3, 2008, pp.4-8.
- [5] C. J. Guo, W. Sun, Y. Huang, Z. H. Wang and B. Gao, "Framework for Native Multi-Tenancy Application Development and Management", CEC-EEE 2007, 2007
- [6] P. K. Arya, V. Venkatesakumar and S. Palaniamami, "Configurability in SaaS for an Electronic Contract Management Application", Recent Advances in Networking, VLSI and Signal Processing, pp.210-216
- [7] Salesforce.com Corporation, <http://salesforce.com>, 2010
- [8] F. Chong and G. Carraro, "Architecture Strategies for Catching the Long Tail," Microsoft, Apr. 2006. <http://msdn.microsoft.com/en-us/library/aa479069.aspx>
- [9] "Software-plus-Services", Microsoft, 2009.
- [10] "Force.com: Create and Run any Application, On Demand," Salesforce.com
- [11] Software as a Service (SaaS) Resource Center, <http://www.deitel.com/ResourceCenters/Web20/SoftwareasaServiceSaaS/tabid/1659/Default.aspx>