2011 IEEE International Conference on Services Computing

Leveraging Cloud Platform for Custom Application Development

Nianjun Zhou[1], Da Peng An[2], Liang-Jie Zhang[1], Chih-Hong Wong[2]
[1] *IBM T.J. Watson Research Center, Hawthorne, NY 10532, USA*
[2] *IBM Global Business Service Solution, Beijing, 100085, China*
E-mail: {jzhou}@us.ibm.com

Abstract

Compared with packaged application, custom application developments (CAD) experience the frustration of higher project overhead and less certainty. The typical time spent on building the infrastructure for a CAD project is, on average, several weeks. Project uncertainty comes from unique customer requirements and lack of standardized methods and toolsets to follow. Therefore, a CAD project is more difficult to achieve cost reduction and asset reuse. In this paper, we present a cloud platform to alleviate this problem through an integration of a) standard methods; b) standardized toolsets aligned with those methods; c) project management environments with pre-defined work breakdown structure (WBS) aligned with those methods and toolsets; and d) infrastructure support from the cloud technology. We believe that such a cloud platform will become a fundamental approach for large enterprises to develop CAD or other solutions for their clients.

Keywords: *cloud computing, custom application development, solution workbench, project management, project assembly, cloud image, team collaboration, asset reuse*

I. Introduction

Custom application development (CAD) is a way to develop a business solution in a planned and structured process using IT methods and technologies. Unlike packaged applications, CAD requirements are mainly customer-specific. A CAD project becomes successful only if it meets all of the following three criteria:

- Time: Completed on schedule;
- Effort (or cost): Finished on or even under budget; and
- Goal: Satisfied all the customer functional and non-functional requirements.

Unfortunately, only a small portion of the completed CAD projects satisfy all three criteria. Quite a number of projects are cancelled for having failed financially or schedule-wise. Failure factors range from poor project estimation, unclear business requirements, to non-professional management and development. Today, almost all project managers have to face the conflicting goals of rapid project delivery and project quality assurance. Even worse, CAD projects encounter higher project overhead compared with packaged application projects. It is often the case that a project spends weeks to months to build an infrastructure to start a regular project execution. The time needed for preparing requirements is longer due to unique customer needs.

In this paper, we present an integration environment using the cloud technology [1] as a potential solution for CAD development to achieve cost and risk reduction. We can category this environment as *Platform as Service* (PaaS) using the definition of cloud computing. The service offered by our CAD platform allows a project manager to connect together a wide range of methods, development teams, tools and computing resources. Our goal of developing this platform is to identify a better approach for large enterprises to develop IT solutions for CAD projects. In detail, our CAD cloud platform provides an environment for rapid development involving a development model that a "skeleton" version of the application solution is available for use at the beginning of a project. The skeleton solution includes project execution plans, pre-defined work items and deliverables, and ready-to-use machines with configured software stacks. Furthermore, with the help of the cloud technology, we can make the development infrastructure ready in a few minutes.

The rest of this paper is organized as follows. Section II provides the business scenario of a typical CAD project and its challenges of achieving cost reduction and asset reuse. Section III presents a detailed architecture overview of the CAD cloud platform. Section IV introduces the method componentization and project assembly features in the CAD platform. Section V gives a case study. Section VI discusses related work. Finally, Section VII summarizes the paper and discusses future directions of technology enhancement.

II. CAD Development

CAD development typically is complex from the perspective of both team formation and project management. Team formation is complex because it consists of many different people with a variety of roles (such as business partners, business analysts, architects, project managers, consultants and system integration specialists and developers, testers, and administrators).

978-0-7695-4462-5/11 $26.00 © 2011 IEEE
DOI 10.1109/SCC.2011.110

Project management is complex because project execution involves many activities and tasks. Many of those tasks are project specific (such as unique interfaces, specific integration, possible extensive data flows, complex algorithms, and enterprise-specific process logics). To reduce complexity, a project manager typically breaks down the execution flow, following a specific development methodology, into processes and development phases. Each phase handles a certain number of activities and tasks.

A. CAD Development Overview

At the beginning of a CAD project, the customer team typically has only a very high-level abstract idea of what they want to develop. The development environment and technologies, as well as software stacks all depend on the detailed requirements and budget constraints.

Based on QSM software [2], an CAD development process has four phases. Phase I is *feasibility study* to complete both the technical and cost feasibility studies based on the initial requirements. Phase II is *functional design* to capture the business requirements and complete high-level design. In this phase, project business analysts and its chief architect analyze and determine the scope of the potential development. In this phase, the chief architect needs to work with business analysts to recognize any incomplete, ambiguous, or even contradictory requirements. Beyond that, they also need to identify those functional or non-functional requirements that are out of scope of the project under current budget constraints.

Phase III is the *main build* phase. It contains the activities of detailed design, code development, and testing. Typically, detailed design is needed to be documented both in the form of models and work documents for the purpose of future maintenance and enhancement. In this phase, programming environment is needed to be created for actual coding. Testing environment is required for project testers to recognize solution defects as early as possible for quality and cost control. In this phase, different application development methods and tools can be applied. For example, two well-known methods are *waterfall* and *agile*. In the agile method, requirements are completed in an iterative manner. The project team frequently demonstrates its milestones to the customer to help reduce the risk of ambiguity in the requirements.

Finally, Phase IV, or the *maintenance* phase, focuses on maintaining the application for customer daily usage. This phase needs to complete the tasks of installation and configuration of final application in the customer environment. Another important task is knowledge transfer from the project development team to the project maintenance team. In some cases, ignoring this knowledge transfer process can lead to unnecessary maintenance costs.

B. Challenges in Achieving Efficiency

Due to unavoidable ambiguity of CAD development, many CAD projects face challenges in project planning, estimation and execution. The challenges come from the following two aspects. The first challenge is the lack of standardization of both development methods and tools. Although there are many methods and tools developed, for a given project, it is common that there is either no proper applicable method or big customization effort required to meet the needs of the project management. Due to the complexity of CAD projects, sometimes it is difficulty for a project manager as well as the project team to be familiar with all aspects of the project. Therefore, the management and the execution of development tasks can be time-consuming and inefficient.

The second challenge is the extra project overhead. One common overhead is to create the project development infrastructure. Today, this needs to take an average of several weeks to complete. Another extra overhead is to track the project execution and deliverables.

C. Best Practices to Achieve Efficiency

To improve the chance of project success, best practices have been developed to govern project execution. First, it is always important to use a standardized, componentized architecture for CAD development. The initial cost and time saved for one application could be small due to the extra efforts required to create such standardization, but this standardized process could create reusable knowledge and assets, which can be applied to many other applications. Such a standardization process includes: 1) harvesting those methods into a standard format (say XML) and a proper tooling environment (such as IBM Rational Method Composer); and 2) developing a solution development environment to materialize such a method and consistently applying those to CAD project development. For example, we can convert the standard method into web accessible guidelines, standard design or work product/document templates and recommended *work breakdown structures*.

Second, it is important to reuse existing assets from other CAD projects, especially from those projects similar to the project under development. Those assets include business requirements, macro/macro design models/documents, testing scenarios, and service components [10]. To achieve the goal, we need to create an asset repository, and enforce a strict asset harvesting and reuse governance process. The success of reuse also relies on a high quality asset categorization approach. Besides, it is important to include existing software packages (such as SAP, Oracle, and Cognos) into the project development. Typically, those packages are very reliable and predictable, as they have been tested over many prior customer engagements.

Third, it is important to improve the knowledge and skill level of development team. Such improvement can be achieved through training and experience sharing through peer to peer coaching. It is also important to form stable teams to handle industry specific solutions. Such stable team could reduce overhead due to lack of domain knowledge of the development team for a CAD project.

III. CAD Cloud Platform

As a response to the complexity of CAD development, we develop a cloud platform to reduce project complexity through materializing some best practices into repeatable and automatic artifacts into the platform. Those artifacts are pre-defined methods and pre-packaged tools into the platform. Beyond this, we have made the platform to support distributed and dynamic project execution, including: a) participation from a global team with members located at different geographic regions; and b) integration with new middleware or end-user development software on demand.

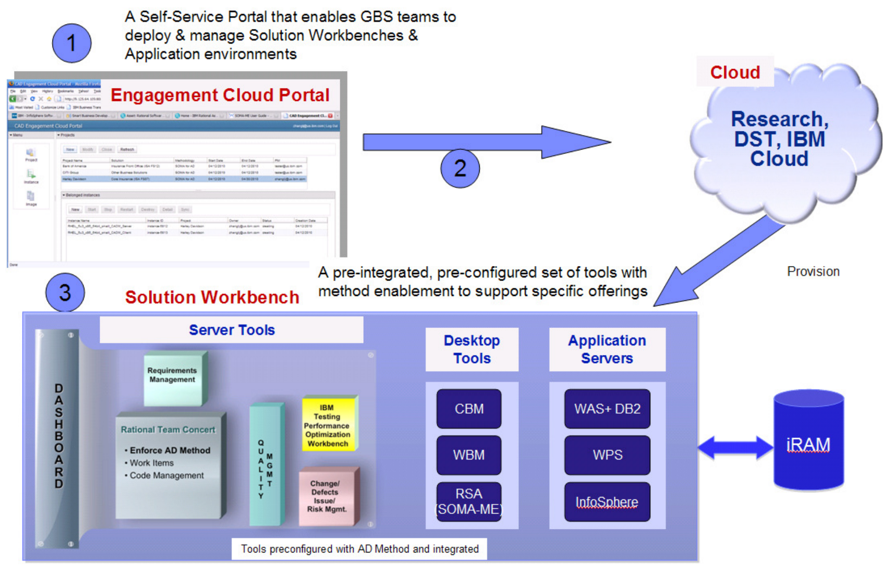

Figure 1. Architectural View of CAD Cloud Platform

A. Overview of CAD Cloud Platform

Figure 1 is the architectural view of the platform. The platform has three major building blocks. The first building block is a self-service web portal containing two web functionalities. The first functionality will be used by the potential CAD project managers. Project managers use this function to define a CAD project, to form a project team, and to create development environment through computing resources provisioning from cloud. The second web functionality is used by the CAD platform administrators. The administrators use this function to add additional cloud images with pre-installed tooling software, and to integrate with standard development methods and reusable assets.

The second building block is a set of cloud infrastructures from different vendors. We use a common set of APIs to access those cloud infrastructures from our CAD portal. With this set of APIs, we remove our dependence on a specific cloud infrastructure and allow our potential users to have various choices of the vendors based on the prices, supporting quality and available products. Each cloud infrastructure can host pre-generated images provided from the CAD platform. When there is a request of creation of a computer instance from a project manager, the portal will route this request to the designated cloud infrastructure selected by the project manage. The cloud infrastructure will dynamically create the instance.

The third building block is a set of cloud images owned by the CAD platform (Figure 2). Physically those images are hosted at different cloud infrastructures; contain preconfigured software stacks with corresponding operating systems. Considering cost constraints of project development, we create images with the same software stacks but different configurations (such as capacity and operating system). Hormonally, those images are categorized into three groups: a) server-side images, b) desktop-side images and 3) application development images. Vertically, they are categorized based on the specific business development types, named as workbench. We will have more detailed discussion of workbench in Section IV.

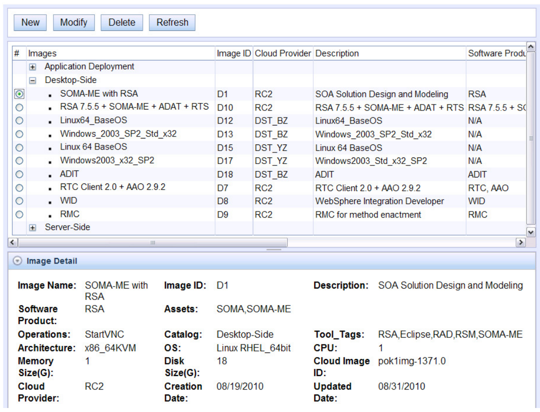

Figure 2. List of Pre-Build Cloud Images in CAD Platform

B. Services for Project Manager

The self-service portal provides all the basic functions for a project manager and the develop team to use the cloud CAD platform. Those functions are similar to the functions of an online store, where a buyer (project manager) selects required items from the product lists (images) and proceeds to the checkout (provisioning instances). The whole process is completed in three steps: 1) First, a project manager creates a project with basic project information; 2) then identifies project members and assigns roles to them; 3) finally creates a set of cloud instances for this project (Figure 3). Each instance contains a set of software required for completion of certain project tasks. Each instance is associated with one or more team member(s). Only a legitimated member can access the instance(s) assigned to him/her.

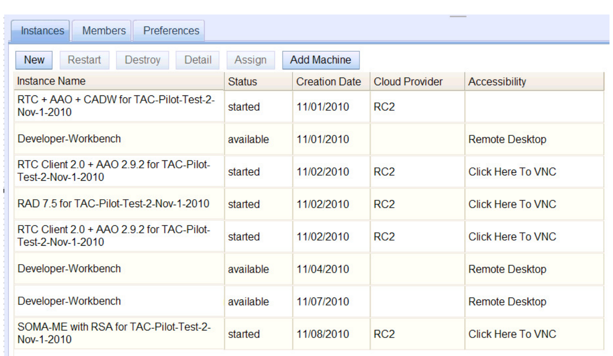

Figure 3. List of Instances of a Sample CAD Project

Similar to categorizing online products, there are multiple pre-defined workbenches to help CAD project managers to make their selection. Each workbench is composed of a development method and a set of tools installed into the cloud images (Figure 4).

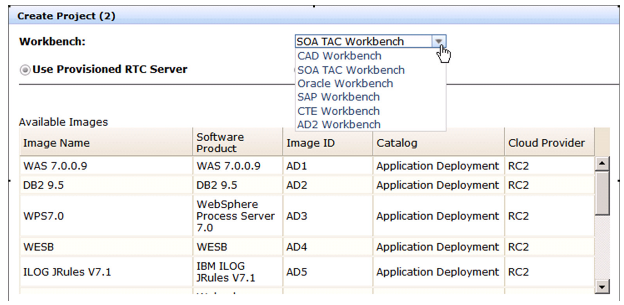

Figure 4. Available Images for a Specific Workbench

In cloud computing, provisioning means acquirement of virtualized computing resource (typically virtual machine) dynamically. To provide the flexibility of cloud platform, there is no explicit constraint from cloud portal to limit a project manager from provisioning the number of instances for his/her project.

Corresponding to three different types of images, there are three different kinds of instances owned by a project. They are:

- Server-side instance – Only one instance is needed with pre-installed and configured project management tool (such as Microsoft Project or IBM Rational Team Concert) for a given project. When a CAD project is created from the CAD portal, the initial *work breakdown structure* (*WBS*) will be created based on the method selected and industry solution type of the application. The project manager could customize the WBS if there is a need;
- Desktop-side instances – multiple instances could be created for a project. An instance is assigned to a specific team member or a group of members committed to the same set of tasks. Typically, most of the instances are created to complete the activities and tasks of Phase III of project life cycle (Section II.A);
- Application development instances – those instances are created for solution integration and testing purpose. Those instances mimic the customer IT environment for final preparing solution deployment.

C. Integration with Different Cloud Infrastructures

A set of APIs are created for the integration of the cloud portal and the underneath cloud infrastructures. From the CAD portal, a project member could remotely control project instances through *creation*, *starting* and *stopping* and *destroying* commands available at the portal site.

The cloud platform owns a pair of public and private keys shared with all the images and instances of all the projects. With the pair of keys, the portal is able to remotely configure any provisioned instance. When an instance is created, an initial remote configuration is carried out to handle to create initial *WBS*, and to set up connection between desktop instances and server-side instance.

IV. Workbench and Project Enablement

In the context of this paper, a workbench is defined as a combination of a development method and a toolset for a specific group of CAD development. Such a workbench is dedicated to simplify and optimize development activities. Typically, the development method is embedded into the context of a *work breakdown structure* (*WBS*). The CAD platform could support multiple workbenches.

The process of supporting a workbench is completed in two steps. The first step is to define a set of unique images as discussed in Section III.B. The second step is the method componentization. This step will separate the WBS into multiple components to provide maximum flexibility for project execution. Each component will be defined as a set of work items (activities and tasks). In a project execution, a *WBS* component is treated as an entity, either included or excluded from the project execution.

To support project management, our CAD platform also provides a project assembly and asset reuse features. The project assembly feature allows a project manager to customize the initial *WBS* created for a project. The asset reuse feature allows the project team to reuse the assets from previous projects into this project and submit back some harvested asset from the current project from an asset repository (as shown in Figure 1, iRAM is a asset repository).

A. Method Componentization

We separate a method into multiple *WBS* components. The componentization typically follows the milestones of a development process. Each component describes a variety of tasks or activities, potentially takes place during the development process. Components have dependencies. Such dependencies could be mathematically modeled as a directed tree. The characteristics of the components are:

- Each component describes a set of self-contained activities/tasks;
- An directed edge reflects the potential sequence of the project execution of two neighboring components;
- Each component contains entry tasks as prerequisite of the execution of component; and exit tasks for the completion of the set of tasks of this component;
- Some of the components are mandated and some are optional;
- Starting node of the component tree is the initiation stage of a project;
- Ending node of the component tree is the completion state of a project.

B. Project Plan Assembly

Project assembly is a web feature for a project manager to customize an initial project plan on demand. Project manager can leverage this initial to finalize his/her project plan.

The process of creating a detailed *WBS* for a project is demonstrated in Figure 5. In this process, initialing a *work breakdown structure* is modeled as a process of interactive steps of filling in the details of the project plan (modeled as a project plan tree). The root node represents an empty project plan. A project manager selects a folder or a *WBS* component to add into the tree. *WBS* components are the leaf nodes of the project plan. A folder node is used to align multiple WBS components to complete a phase or sub-phase of a project. The *WBS* component added has to follow chronically requirements of project execution. In the case of multiple parallel activities needed to complete, then parallel branches under a parent folder could be created.

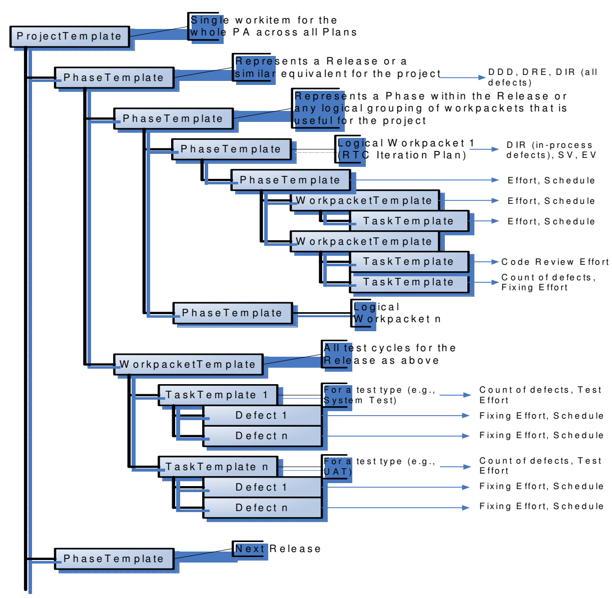

Figure 2. Project Plan Assembly Process

C. Integration Development Project Team Members

Each project team member can access the project management web site (i.e., an IBM *rational team concert* server – RTC) to find out the activities/tasks (work items) assigned to them. Those activities/tasks also could be automatically mailed to them by the project management tool (say IBM RTC server). From those work items, a project member could find out the web link to the tooling environment (cloud instances(s)) and method guidelines to complete this work item. Typically, the credentials of accessing those cloud instances are mailed to the team members through e-mail from the project manager. In this CAD platform, the supported the operating systems of cloud instances are Linux and Microsoft Windows.

To encourage the team collaboration, a portal collaboration area (Figure 6) is created to facilitate development process. Project team members can easily access their cloud instances from the web linkages specified in this collaboration area.

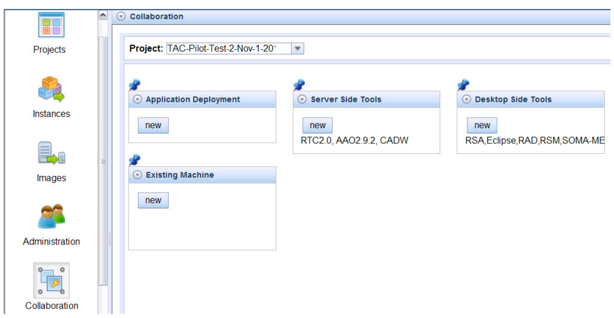

Figure 3. Collaboration Area of a CAD Project

V. Case Study

In this section, we discuss how to board a *SOA engagement workbench* into the CAD cloud platform. The advantage of this workbench is to leverage model-driven development technology to automatically generate production-quality business service components from high-level business process and service models following *Service Oriented Architecture* design principles.

This workbench supports two mutual complement development approaches. The first development approach is called *business entity life cycle analysis* (BELA) [3]. BELA shifts the process-modeling paradigm centered on business activities to a modeling paradigm centered on business entities. The second development approach is called SOMA [4]. SOMA is an end-to-end modeling with the latest SOA method guidance and artifacts to guide architects to evaluate, design, discover, and compose solution artifacts to create an solution based on IBM SOA Reference Architecture [5].

A. Defining *WBS* Components

BELA and SOMA development approaches define design and development activities/tasks in Phase III of a project plan. These activities/tasks are used to create the *WBS* components. The major activities/tasks for BELA are:

- Identify business entities;
- Specify business entities;
- Decompose business processes from specified business entities;
- Transform business entities into service identifications and specification tasks;
- Generate service models including skeleton of data structure and business process;
- Supporting synthesis tools then automatically generate the interacting business entity service components and their associated data stores and service interface definitions;
- Support Service-Oriented Architecture Solution

The major activities/tasks for SOMA are:

- Identify candidate service(s) from use case, business process, functional analysis or legacy asset analysis;
- Carry out service litmus tests – it is a process to identify the final services in a SOA solution;
- Specify final service(s) to identify the data object and interface;
- Realize service through service componentization and sub-system analysis;
- Validate created service model through completion check;
- Generate skeleton developing code;
- Map services into SOA Reference Architecture [5] to have architecture layered analysis.
- Complete the architecture design by gradually filled in architecture building blocks into IBM SOA Reference Architecture.

To have a complete list of *WBS* components to support a whole life cycle of SOA project execution, this workbench also contains other *WBS* components. Those other *WBS* components are shared with other workbenches.

B. Identification Platform Images

To support this workbench, we have created the following set of cloud images (Table 1) with preconfigured software stacks as workbench toolset. The toolset covers the needs from project management, requirement analysis, macro/micro architecture design, coding, testing, and solution integration.

Table 1. List of Images for SOA Workbench (IBM Software Stacks)

Phase	Image ID	Role	Software Stack	Description
Functional Design (Phase II)	Server-Side (I)	Project Manager	Rational team Concert	Managing project using WBS containing all the work item
	Desktop-Side (I)	Business Analyst(s)	Rational Requirement Composer	Capture business requirements including use case(s)
	Desktop-Side (II)	Solution Architect(s)	IBM Rational Software Architect	high level design based on business requirements SOMA-ME service identification
Main Build (Phase III)	Desktop-Side (II)	Solution Architect(s)	Rational Software Architect	Low Level design; SOMA-ME Specification; SOA Solution Stack Design
		Developer(s)		Code development
	Desktop-Side (III)	Solution Architect(s)	WebSphere Business Modeler	Low Level design; business object identification (BELA)
	Desktop-Side (IV)	Developer(s)	WebSphere Integration Developer	Development business process generate BPEL and database schema
	Desktop-Side (V)	Tester	Rational Quality Manager	Capture business requirements
	Application Development (I)	Tester(s)	WebSphere Server	Integration Test
	Application Development (II)	Tester(s)	WebSphere Process Server	Integration Test

VI. Related Work

Zhang and Zhou in [6] propose a *cloud computing open architecture* and introduce seven principles of cloud computing architecture to try to explain how cloud works. We can position our CAD cloud platform into this open architecture as *PaaS* application. The architecture design of this CAD cloud platform is influenced by the layered cloud open architecture.

Besides our effort, there have also been a number of similar or related approaches to accelerate and facilitate development by leveraging the cloud technology. Kondo et al. [7] compared and contrasted the performance and cost benefits of clouds for desktop grid applications. They concluded that monetary costs of clouds are lower than desktop grid applications. With recent advances in virtualization and cloud computing services, the cost of developing custom applications has dropped significantly. Virtualization allows IT organizations to quickly provision servers to develop and test applications without having to actually set up physical servers. Klems et al. [8] presented a basic framework for estimating value and determine benefits from cloud computing as an alternative to conventional IT infrastructure, such as privately owned and managed IT hardware.

IBM has done much research work in creating standard development methods and solution assets. Such research work becomes the base for us to create different solution workbenches. In the cloud side, IBM *Smart Business Development and Test on the IBM Cloud* [9] is built on an agile cloud infrastructure that is designed to provide an end-user with rapid access to security-rich, enterprise-class virtual server environments, well suited for development and test activities and other dynamic workloads. Our solution moves a step further to include the whole lifecycle project development into cloud.

VII. Discussions and Future Directions

In this paper, we present a CAD cloud platform as an integrated environment to enforce the consistence of development methods and tools applied for project development. By leveraging the cloud technology, the platform can dramatically reduce the time of setting up of the development infrastructure from weeks to several hours.

From engineering perspective, the immediate future work includes: 1) adding additional workbenches into this platform to support other CAD development projects; 2) hosting additional pilot project developments to validate this new development model, and 3) enhancing the maturity of the platform itself.

From research perspective, an interesting problem is to quantify and measure the efficiency enhancement using such platform. One possible quantifying approach could be defining the enhancement as complexity reduction in development. The complexity of a development could be measured as the steps and efforts required defining, managing and developing a solution. The maximum efficiency enhancement can be found through the trade-off between *WBS* componentization and customization required.

From business perspective, how to balance the standardization (both method and tooling environment) and flexibility is also very important. It is typically harder to enforce tools than methods due to some management and business reasons. Those reasons are 1) the habits of using same tooling of the development team and 2) the availability and the expense of some advanced tools.

Finally, what is the business model we should consider to adopt for this platform? We need to deal with service fees for providing the method supports, and also need to charge the computing resources usages (that may involve a third-party infrastructure provider).

References

[1] M. Armbrust, et. al. "Above the Clouds: A Berkeley View of Cloud Computing", U.C. Berkeley Technical Report No. UCB/EECS-2009-28- http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.html, February 10, 2009

[2] QSM Software, "SIZING AND ESTIMATION - Key Information for SLIM Forecasting", http://www.qsm.com/data2.pdf

[3] J. K. Strosnider; P. Nandi; Kumaran, S.; Ghosh, S.; Arsnajani, A.; "Model-driven synthesis of SOA solutions". In IBM Systems Journal, vol 47 no.3, pp. 415 – 432, 2008

[4] L.-J. Zhang, N. Zhou, Y.-M Chee, A. Jalaldeen, K. Ponnalagu, R. R. Sindhgatta, A. Arsanjani and F. Bernardini, "SOMA-ME: A Platform for the Model-Driven Design of SOA Solutions". In IBM System Journal, vol. 47, no. 3, pp. 397-413, 2008.

[5] A. Arsanjani , L.-J. Zhang , M. Ellis , A. Allam, K. Channabasavaiah, *S3: A Service-Oriented Reference Architecture*, IT Professional, May 2007

[6] L.-J Zhang and Q. Zhou, "CCOA: Cloud computing open architecture,"IEEE ICWS, July 06-09, 2009, pp. 607-616.

[7] D. Kondo; B. Javadi; P. Malecot; F. Cappello; D.P. Anderson; "Cost-benefit analysis of Cloud Computing versus desktop grids", IEEE International Symposium on Parallel & Distributed Processing, May 23-29 2009, pp: 1 – 12.

[8] M. Klems, J. Nimis and S. Tai, "Do Clouds Compute? A Framework for Estimating the Value of Cloud Computing", Lecture Notes in Business Information Processing, 2009, Volume 22, Part 4, 110-123.

[9] IBM Smart Business Development and Test, http://www-935.ibm.com/services/us/igs/cloud-development/

[10] N. Zhou, L.-J. Zhang, Y.-M. Chee, L. Chen: *Legacy Asset Analysis and Integration in Model-Driven SOA Solution*. IEEE SCC 2010: 554-561