An Architecture for Cloud-based e-Book Software-as-a-Service

1 Jyhjong Lin and 2 Chaoyu Lin
Department of Information Management, Ming Chuan University, Taiwan
1 jlin@mail.mcu.edu.tw; 2 lendy.lin@gmail.com

Abstract—For the rapid development of internet technologies, using internet-accessed devices for on-line reading has become a must part of everyone’s day life. Among all attractive readings, e-Book is receiving more attentions in recent years for benefiting on-line readers; two typical e-Book services can be found from Amazon and Google e-Book stores. However, some desirable features for more satisfying reader needs are still not supported well in these existing services such as recognition & comparison of desired e-Books and collaboration & communication in e-Book communities. To address this issue, we present in this paper a cloud-based architecture for such desirable e-Book services (i.e., cloud-based e-Book SaaS). The architecture is identified from the consideration of e-Book and cloud-based SaaS characteristics, through the recognition of the architectural components that support the employment of SaaS into e-Book, and finally ends with the deployment of these architectural components on the elements in realistic clouds for the realization of the desirable e-Book SaaS.

Keywords—e-Book, cloud-based SaaS, architecture

I. INTRODUCTION

For the rapid development of internet technologies, using internet-accessed devices for on-line reading has become a must part of everyone’s day life. Among all attractive readings, e-Book is receiving more attentions in recent years for benefiting on-line readers. The idea of cloud-based e-Book services is not new. Safari Books [1] was released to attempt improving the interactivity and collaboration among on-line readers under a basic concept “everything is always in sync because your library is in the cloud.” Other successful e-Book services can be found from Amazon [2] and Google [3] e-Book stores. Although e-Book has been commonly recognized as a trend for the next generation of on-line reading, many people remain reluctant to read digital documents from screens [4]. For this need, some desirable features such as recognition & comparison of desired e-Books and collaboration & communication in e-Book communities have been proposed for more satisfying reader needs to convince these book readers to accept e-Book. Nonetheless, existing e-Book services such as the two typical ones from Amazon and Google e-Book stores still do not support well these features.

To address this issue, we present in this paper a cloud-based architecture for such desirable e-Book services (i.e., cloud-based e-Book SaaS) that not only takes advantage of the cloud-based SaaS [5] but also possesses the desirable features by employing basic concepts of social tools [6,7] and consumer support [8]. Therefore, the architecture is identified from the consideration of e-Book and cloud-based SaaS characteristics under the concepts of social tools and consumer support, through the recognition of the architectural components that support the employment of SaaS into e-Book, and finally ends with the deployment of these architectural components on the elements in realistic clouds for the realization of the desirable e-Book SaaS.

II. THE ARCHITECTURE

The architecture is developed with the following three steps:

1. Requirement Identification - clarify SaaS features and e-Book characteristics under the concepts of social tools and consumer support, and then identify the possible employment of SaaS features into e-Book under the frameworks of social tools and consumer support (i.e., the desired requirements for e-Book SaaS).

2. Component Identification - determine the architectural components to support the above desired requirements.

3. Deployment Specification - identify the configuration elements in realistic clouds where the architectural components are deployed to realize the requirements.
2.1 Requirement Identification

Initially, consider the characteristics of e-Book with the features of social tools and consumer support. Fig. 1 shows its possible architecture where Reader\textsubscript{1...N} interact with Publisher\textsubscript{1...M} via Community, Knowledge Agent, and Service Agent.

1. It emphasizes on the community to help on-line readers share information about their desired books.
2. It emphasizes on valuable information by collecting knowledge from readers (i.e., residing in themselves).
4. It emphasizes on delivering the book information based on book requests from readers.

Then, based on [9], SaaS has the following features:

1. The architecture of a cloud may have a wide variety of configuration elements, including for example virtual machines, data storages, a/synchronous message queues, and user action portals/gadgets. Cloud applications may integrate the use of these elements to provide services.
2. Most of the elements in this architecture are dynamic and leverage a SOA. It is possible for clouds to interoperable among each other (i.e., interoperability among clouds).
3. Service elements such as virtual machines, data storages, and message queues may be used together to enable the customizations of services by encapsulating the desired services from prospective clouds (either local or interoperable clouds).
4. Interface elements such as user action portals and gadgets may be used to provide rich user interface controls that enable the customizations of user interfaces by encapsulating the desired portals or gadgets in different user panels.

Finally, with SaaS features, their incorporations into e-Book as e-Book SaaS may then be identified:

1. With interoperability among clouds, respective clouds may be incorporated into the architecture of e-Book as in Fig. 2 where Community, Reader\_Knowledge\_Agent, Service\_Agent, Publisher\textsubscript{1...M}, and Store\textsubscript{1...M} become collaborative clouds (denoted as cloud@xxx).
2. *Cloud@Community* enables Reader$_{1...N}$ to share info. about their desired books where such shared info. is re-structured by *Cloud@Knowledge_Agent* into specific reader knowledge and then sent to *Cloud@Publisher$_{1...M}$* for Publisher$_{1...M}$ to provide the desired books into *Cloud@Store$_{1...M}$* for these Reader$_{1...N}$.

3. *Cloud@Community* interacts with *Cloud@Service_Agent* to accomplish the book requests issued by Reader$_{1...N}$.

4. Based on book requests, *Cloud@Service_Agent* collects and evaluates book info. from *Cloud@Store$_{1...M}$* and then presents the info. to desired Reader$_{1...N}$.

5. *Cloud@Community* provides Reader$_{1...N}$ with rich user interface controls for accessing the evaluated information from *Cloud@Service_Agent*.

2.2 Component Identification

With the above five requirements, e-Book SaaS should be designed to address them by the collaborative clouds between Reader$_{1...N}$ and Publisher$_{1...M}$. For example, *Cloud@Community* is organized for readers to share info. about their desired books. In addition, it is responsible for forwarding such shared info. to *Cloud@Knowledge_Agent* for re-structuring into knowledge. Finally, it also cooperates with *Cloud@Service_Agent* to receive the book info. relevant to those book requests from Reader$_{1...N}$.

In summary, these requirements for *Cloud@Community* can be described as follows: (1) helps to share info. among Reader$_{1...N}$; (2) forwards the shared info. to *Cloud@Knowledge_Agent* for re-structuring into reader knowledge; (3) cooperates with *Cloud@Service_Agent* to process book requests from Reader$_{1...N}$; and (4) presents book info. to Reader$_{1...N}$. To address these requirements, it is therefore designed with six role-specific components as shown in Fig. 3.
2.3 Deployment Specification

With architectural components, the next is to map them into various configuration elements in realistic clouds. Such a mapping is to ensure the feasibility of further development steps where each component follows all issues under the topologies of these clouds. Fig. 4 illustrates the mapping of Cloud@Community in Fig. 3 to configuration elements in the Amazon EC2 cloud environment [10]. More specifically, respective virtual machines (VMs) are imposed for hosting its various responsible works where an asynchronous queue is particularly used for managing the two different types of works for sharing info. among Reader$_{1...N}$ in a real time work and forwarding the shared info. to Cloud@Knowledge.Agent in a batch work.

Figure 4: The Amazon EC2-based deployment of Cloud@Community for cloud-based e-Book SaaS

III. CONCLUSIONS

In this paper, we present an architecture for the e-Book SaaS. The architecture is developed from the identification of e-Book characteristics and SaaS features. After then, the cloud-based architecture and deployment of system components are specified to support and realize the incorporation of SaaS features into e-Book. For e-Book SaaS, these cloud-based components focus on supporting both of the knowledge from on-line readers to book publishers and the reverse delivery of book information from book publishers to on-line readers. In particular, within the context of realistic clouds, prospective configuration elements are specifically identified where architectural components are deployed to realize the e-Book SaaS.

In general, for the advances of internet technologies in recent years, on-line reading has become a must part of everyone’s day life. It is therefore for e-Book to receive more attentions for benefiting on-line readers. Many successful e-Book services can be found from the market such as those by Amazon and Google e-Book stores. However, some issues can still be found in the context of on-line reading where the most critical one seems as that many people remain reluctant to read digital documents from screens. To address this issue, some desirable features such as recognition & comparison of desired e-Books and collaboration &
communication in e-Book communities have been proposed for more satisfying reader needs to convince these book readers to accept e-Book. As such, the architecture herein is presented not only taking advantage of the cloud-based SaaS but also possessing these desirable features by employing the frameworks of social tools and consumer support in its components.

As our future work, we will continue to explore the construction of the e-Book SaaS by applying well-known cloud computing techniques like Amazon EC2 [10] and Google GAE/GCE [11,12] cloud environments. Thereafter, we will look forward to the practical use of our work in various domains like e-Learning systems; its usability on such cloud-based customer support systems will also be carefully experienced.

ACKNOWLEDGMENT

The material presented in this paper is based on work supported by the National Science Council in Taiwan under Grant MOST 103-2410-H-130-015.

REFERENCES