

## Chapter 2

# A Cloud Portal Architecture for Large-Scale Application Services

**Jun-Jang Jeng**  
IBM, USA

**Ajay Mohindra**  
IBM, USA

**Jeaha Yang**  
IBM, USA

**Henry Chang**  
IBM, USA

### ABSTRACT

*Application services entail multi-billion dollars of market in IT industry. However, to construct an application service is a labor-intensive and error-prone process. Application services developed through traditional development methods expose the same pitfalls witnessed in most development processes of enterprise applications such as late delivery, over budget, unpredictable quality, lack of reuse and so forth. We have leveraged clouds in developing application services within the context of large corporate with the magnitude of thousands of application services being built, delivered and used. Instead of using cloud simply for a better runtime engine, it is being used as the development platform to accelerate and optimize the solution development process based on large scale application services. This paper will focus on the portal architecture of this framework—coined as Cogito-C that contains four spaces: (a) infrastructure space; (b) application space; (c) business space; and (d) presentation space. This paper illustrates Cogito-C by scrutinizing the models in the aforementioned spaces. This paper will focus on the descriptive models of this framework. Examples are used to explain how this framework is organized and exploited for large-scale application services.*

DOI: 10.4018/978-1-4666-0336-3.ch002

## 1 INTRODUCTION

Application services entail multi-billion dollars of market in IT industry. As such, application service providers and consumers continue investing considerable amount of time and efforts to develop application services-based solutions. Prepackaged business applications such as enterprise resource planning and customer relationship management offer significant benefits for businesses and are critical for business success. In this direction, leading package application service providers such as HP, IBM and SAP (IBM, 2009a), exploit domain-specific skills to help their clients excel competitors through cost reduction and risk mitigation. However, current approach is quickly encountering its very own limitation. First, the project planning and implementation for application services are still time-consuming and costly. It usually requires a variety of skills and expertise that many companies do not possess. There are also high cost associated with the ongoing management and maintenance of these applications. Second, because packaged applications are often tightly integrated with existing systems, the clients typically require a broad range of technical expertise to run them, which is usually hard or expensive to obtain from other parties. Third, with major concern about cost, both service providers and consumers fall into the vicious cycle—more cost reduction leads to more efficiencies which raise the expectation of more cost reduction. Consequently, no one is the winner because service quality is overlooked in this cycle. No room is left in the delivery process for considering quality—not to mention to improve it.

Application services pertain richer functionality and content than normal specialized services. They scale up the concepts of service-oriented computing to business-level so that service-oriented computing concepts became more receptive to business professionals. Technically, application services can be composed and configured to different but similar domains. An application

service hides the implementation detail from its users and, by itself, can be also a composite of other services. To construct an application service is a labor-intensive and error-prone process. Moreover, application services developed through traditional development methods expose the same pitfalls that are witnessed in most development processes of enterprise applications such as late delivery, over budget, unpredictable quality, lack of reuse and so on.

Clouds are the next generation of infrastructure, bestowing the mechanism of virtualization technologies such as virtual machines. Clouds are able to dynamically provision services on demand as a personalized resource collection to meet a specific service requirement, which is established through negotiation and accessible as a service via network (Herssens, Faulkner, & Jureta, 2008a, 2008b). We have leveraged clouds in developing application services within the context of large corporate with the magnitude of thousands of application services being built, delivered and used. Instead of using cloud simply for a better runtime engine, it is being used as the development platform to accelerate and optimize the solution development process based on large scale application services. This paper will focus on the portal architecture of this framework - coined as *Cogito-C* that contains four spaces: (a) infrastructure space; (b) application space; (c) business space; and (d) presentation space. This paper illustrates *Cogito-C* by scrutinizing the models in the aforementioned spaces. This paper will focus on the descriptive models of this framework. Examples are used to explain how this framework is organized and exploited for large-scale application services.

The rest of this paper is organized as follows. Section 2 presents the overall architecture of *Cogito-C* with emphasis on market viewpoint. Section 3 presents briefly the cloud platform being used for realizing *Cogito-C*. Section 4 shows how applications are provisioned and deployed to *Cogito-C* and made ready for use by users. Section 5 describes the space that provides business

13 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

[www.igi-global.com/chapter/cloud-portal-architecture-large-scale/63941](http://www.igi-global.com/chapter/cloud-portal-architecture-large-scale/63941)

## Related Content

---

### Using Web Portals to Model and Manage Enterprise Projects

Sergey Zykov and Alexey Kukushkin (2013). *International Journal of Web Portals* (pp. 1-19).

[www.irma-international.org/article/using-web-portals-to-model-and-manage-enterprise-projects/103979](http://www.irma-international.org/article/using-web-portals-to-model-and-manage-enterprise-projects/103979)

### Watermarking Integration into Portals

Patrick Wolf (2007). *Encyclopedia of Portal Technologies and Applications* (pp. 1104-1109).

[www.irma-international.org/chapter/watermarking-integration-into-portals/18015](http://www.irma-international.org/chapter/watermarking-integration-into-portals/18015)

### A Comprehensive Methodology for Campus Portal Development

Tharitpong Fuangvut and Helen Hasan (2007). *Encyclopedia of Portal Technologies and Applications* (pp. 166-171).

[www.irma-international.org/chapter/comprehensive-methodology-campus-portal-development/17864](http://www.irma-international.org/chapter/comprehensive-methodology-campus-portal-development/17864)

### Research Essay: Improving Our Approach to Internet and SOA Projects

Neil Richardson (2010). *International Journal of Web Portals* (pp. 52-56).

[www.irma-international.org/article/research-essay-improving-our-approach/49567](http://www.irma-international.org/article/research-essay-improving-our-approach/49567)

### Ontologies in Portal Design

G. Bhojaraju and Sarah Buck (2007). *Encyclopedia of Portal Technologies and Applications* (pp. 653-657).

[www.irma-international.org/chapter/ontologies-portal-design/17944](http://www.irma-international.org/chapter/ontologies-portal-design/17944)