

Chapter 12

Principles, Methodology and Tools for Engineering Cloud Computing Systems

Luis M. Vaquero

Telefónica Investigación y Desarrollo, Spain

Luis Rodero-Merino

INRIA, France

Juan Cáceres

Telefónica Investigación y Desarrollo, Spain

Clovis Chapman

University College London, UK

Maik Lindner

SAP Research, UK

Fermín Galán

Telefónica Investigación y Desarrollo, Spain

ABSTRACT

Cloud computing has emerged as a paradigm to provide every networked resource as a service. The Cloud has also introduced a new way to control cloud services (mainly due to the illusion of infinite resources and its on-demand and pay-per-use nature). Here, we present this lifecycle and highlight recent research initiatives that serve as a support for appropriately engineering Cloud systems during the different stages of its lifecycle.

1. INTRODUCTION: CLOUD SYSTEM PRINCIPLES AND IMPLICATIONS FOR THE MARKET

Following a quite comprehensive and often cited definition, Clouds can be described as follows: “Clouds are a large pool of easily usable and accessible virtualized resources (such as hardware, development platforms and/or services). These resources can be dynamically reconfigured to

adjust to a variable load (scale), allowing also for an optimum resource utilization. This pool of resources is typically exploited by a pay-per-use model in which guarantees are offered by the Infrastructure Provider by means of customized SLAs” (Vaquero et al. 2009).

Cloud Services can be divided into three major areas. Within these areas each service should fulfill the characteristics above. As many companies bundle their offers and put their own description behind them, the boundaries of provided services are not always sharp. Nevertheless, a general un-

DOI: 10.4018/978-1-60960-735-7.ch012

derstanding in the community has been established that these are the main areas of Cloud Computing services/product that are offered on the market (also see Figure 1):

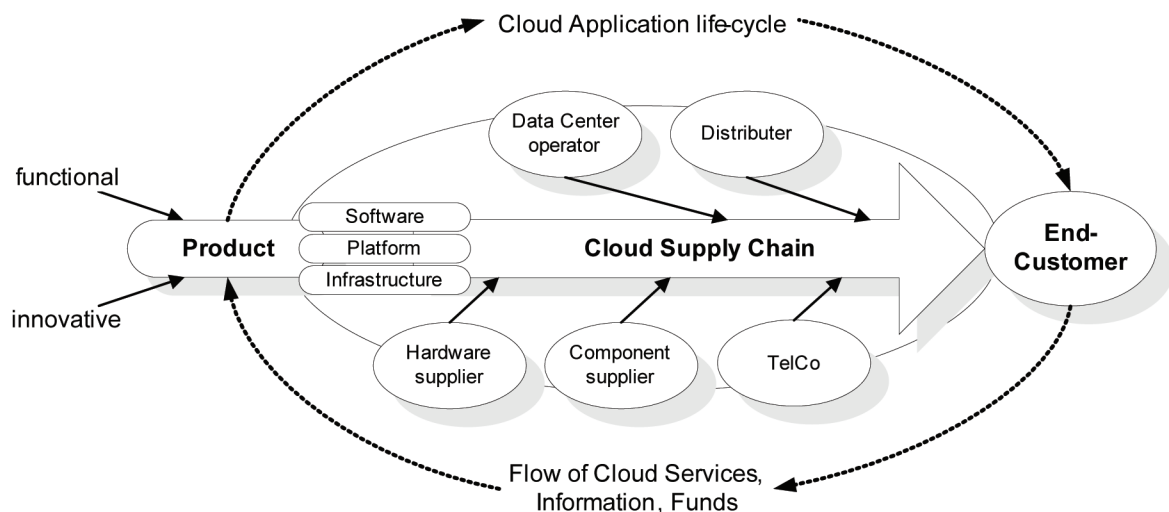
- At its most basic level Infrastructure-as-a-Service (IaaS) delivers resources like pre-packaged sets of e.g. CPU and RAM. Virtualized system images can be uploaded to a cloud provider who provides placement and execution of these images on physical hardware within their data centers or within a federated cloud infrastructure.
- Platform-as-a-Service (PaaS) delivers virtualization and scaling of abstracted software packages above the level of the operating system. Packaged applications are usually uploaded to a cloud platform, or directly developed on the cloud platform itself.
- Software-as-a-Service (SaaS) is perhaps the most common of the 'as-a-Service' terms, and describes fully managed applications delivered as a service. Customers do not need to upload server images or software packages. Instead, they rent ac-

cess to the software which has been created and is maintained by the cloud provider.

The approach described here represents a lifecycle-based methodology that is illustrated with relevant examples from significant and most recent literature and research. A coherent methodology, which could support companies to embrace the Cloud, is still lacking. That has held back progress both on provider and on consumer side. The presented innovative approach, once applied, has the potential to create transparency for the promising IT paradigm of Cloud Computing. To strengthen the proposed approach we apply principles, present a methodology and showcase tools for engineering Cloud Computing systems.

While understanding the basic technical and business features of this new computing paradigm is essential, one has to see the complete picture and understand the implications for the provision of complex Cloud services including components such as networks, machines, infrastructures and software. Therefore, management of the complexity of consuming Cloud services needs to be understood as a supply chain. As an explicit definition for the Cloud Supply Chain (C-SC),

Figure 1. Cloud Supply Chain



22 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/principles-methodology-tools-engineering-cloud/55444

Related Content

Software Engineering and HCI

Shawren Singhand Alan Dix (2009). *Software Applications: Concepts, Methodologies, Tools, and Applications* (pp. 122-127).

www.irma-international.org/chapter/software-engineering-hci/29385

Formal Specification and Implementation of Priority Queue using Stream Functions

Gongzhu Hu, Jin Zhangand Roger Lee (2013). *International Journal of Software Innovation* (pp. 66-75).

www.irma-international.org/article/formal-specification-and-implementation-of-priority-queue-using-stream-functions/103282

Usability Engineering in Agile Software Development Processes

Muhammad Aminu Umar, Sheidu Salami Tenuche, Sahabi Ali Yusuf, Aminu Onimisi Abdulsalamiand Aliyu Muhammad Kufena (2016). *Emerging Innovations in Agile Software Development* (pp. 208-221).

www.irma-international.org/chapter/usability-engineering-in-agile-software-development-processes/145041

Process Algebras for Locality

Rolando Blancoand Paulo Alencar (2012). *Handbook of Research on Mobile Software Engineering: Design, Implementation, and Emergent Applications* (pp. 544-556).

www.irma-international.org/chapter/process-algebras-locality/66486

Business Risk Analysis: Obsolescence Management in Requirements Engineering

Jasbir Virdi (2014). *Software Design and Development: Concepts, Methodologies, Tools, and Applications* (pp. 1736-1763).

www.irma-international.org/chapter/business-risk-analysis/77778