Future Generation Computer Systems 37 (2014) 297–308

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Towards understanding the runtime configuration management of do-it-yourself content delivery network applications over public clouds

Zheng Li[a], Karan Mitra[c], Miranda Zhang[b], Rajiv Ranjan[b,*], Dimitrios Georgakopoulos[b], Albert Y. Zomaya[d], Liam O'Brien[e], Shengtao Sun[f]

[a] School of Computer Science, Australian National University and NICTA, Australia
[b] CSIRO Computational Informatics, Canberra, Australia
[c] Luleå University of Technology, Sweden
[d] University of Sydney, Sydney, Australia
[e] Geoscience Australia, Canberra, Australia
[f] Yanshan University, Qinhuangdao, China

HIGHLIGHTS

- We present the concepts and factors related to DIY content management applications.
- We show a prototype of the Cloud-hosted DIY content management applications.
- A typical methodology is demonstrated for evaluating Cloud-based application systems.
- Extensive experiments were conducted for understanding the runtime configurations.

ARTICLE INFO

Article history:
Received 29 August 2013
Received in revised form
12 October 2013
Accepted 3 December 2013
Available online 17 January 2014

Keywords:
Application runtime configuration
Cloud services evaluation
Content delivery network
Experimental design and analysis
Evaluation methodology
Mediawise cloud content orchestrator
Public clouds

ABSTRACT

Cloud computing is a new paradigm shift which enables applications and related content (audio, video, text, images, etc.) to be provisioned in an on-demand manner and being accessible to anyone anywhere in the world without the need for owning expensive computing and storage infrastructures. Interactive multimedia content-driven applications in the domains of healthcare, aged-care, and education have emerged as one of the new classes of big data applications. This new generation of applications need to support complex content operations including production, deployment, consumption, personalization, and distribution. However, to efficiently provision these applications on the Cloud data centres, there is a need to understand their runtime resource configurations. For example: (i) where to store and distribute the content to and from driven by end-user Service Level Agreements (SLAs)? (ii) How many content distribution servers to provision? And (iii) what Cloud VM configuration (number of instances, types, speed, etc.) to provision? In this paper, we present concepts and factors related to engineering such content-driven applications over public Clouds. Based on these concepts and factors, we propose a performance evaluation methodology for quantifying and understanding the runtime configuration of these classes of applications. Finally, we conduct several benchmark driven experiments for validating the feasibility of the proposed methodology.

1. Introduction

CISCO (a network technology giant) predicts that by 2016, 90% of internet traffic will be multimedia content (3D images, high resolution video, and audio). In addition to entertainment and advertising applications, the new multimedia content-driven applications in the domain of healthcare, aged-care and education will contribute significantly to this traffic. The new applications' contribution to the traffic will be due to their unprecedented processing (storage, distribution, and indexing) requirements for hundreds of petabytes of content. In the *healthcare domain*, live as well as archived videos will be used as a medium to educate patients about the aftercare treatment (follow-ups), once the patient is at home. This will include video instructions about how to change the dressing on a healing wound or how to brush their teeth after having braces installed. Other scenario from the healthcare domain

* Corresponding author.
E-mail address: rranjans@gmail.com (R. Ranjan).

0167-739X/$ – see front matter © 2014 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.future.2013.12.019

will arise from the problem of managing petabytes of multimedia content produced by advanced medical imaging devices. In conjunction with traditional X-rays, medical imaging can now delve deeper into the human body, discovering and analysing smaller and smaller details. In the *aged-care domain*, health professionals will rely on real-time or recorded video feeds from patient's home to monitor clinical signs and indicators such as skin colour, moods and activities to determine whether a patient is utilizing devices and medications appropriately. Finally, in the *education domain*, students will need to have the opportunity to access teachers from home; especially, the students in rural or remote areas need an opportunity to be able to receive interactive lessons or instructions via live video streaming from specialized teachers or trainers who are not available locally.

In the aforementioned application scenarios, hundreds of petabytes of multimedia content will be generated which will be required to be processed (stored, distributed, and indexed with a schema and semantics) efficiently in a way that does not compromise end-users' Quality of Service (QoS) in terms of content availability, content search delay, content distribution delay, etc. Many of the existing ICT systems (such as Content Delivery Networks (CDNs) provided by Akamai, Amazon CloudFront, Limelight Networks, etc.) that store, distribute, and index hundreds of petabytes of multimedia content either fall short of this challenge or do not exist. Hence, there is a need for powerful and sophisticated software tools and technologies that can support scalable storage and fast content distribution, while ensuring QoS on a case-by-case basis.

In our previous work [1,2], we presented MediaWise Cloud Content Orchestrator (MCCO)—a novel system that facilitates Do-It-Yourself (DIY) CDN application orchestration for simplifying the management of multimedia content (e.g., audio and video) using public Cloud services. Unlike existing commercial CDN providers such as Limelight Networks and Akamai, MCCO eliminates the need to own and manage expensive infrastructure while facilitating content owner requirements pertaining to price, privacy and QoS. It offers enhanced flexibility and elasticity as it supports the pay-as-you-go model. MCCO content orchestration operations include: (i) production: create and edit; (ii) storage: uploading and scaling of storage space; (iii) indexing: keyword-based content tagging and searching; and (iv) distribution: streaming and downloading. Similar to any other Cloud application, MCCO suffers from performance unpredictability due to many unknown factors. The availability, load, and throughput of Cloud services, for example, CPU, storage, network and software appliances can vary in unpredictable ways. Thus, the assurance of QoS targets for the DIY CDN applications can be challenging.

Alhamazani et al. [3] discussed that QoS uncertainty is the chief technical obstacle to successful adoption of Cloud computing. The recent very high-profile crash [4] of Amazon EC2 cloud, which took down the enterprise applications of many SMEs, is a salient example of unpredictability in Cloud environments. Some applications were down for hours, others for days. Theoretically, the elasticity provided by Cloud computing can accommodate even unexpected changes in capacity, adding Cloud services when needed, and reducing them during the periods of low demand. However, the decisions to adjust capacity must be made frequently, automatically and accurately to be cost-effective. Hence, understanding the relationships between application workload, Cloud service configuration, and QoS delivered to end-users is mandatory. To understand such relationships, there is a need to conduct several experimental evaluation studies in a systematic way where a systematic approach may involve the following steps: (i) requirement recognition; (ii) service feature identification; (iii) QoS metrics and benchmarks listing; (iv) metrics and benchmarks selection; (v) experimental factors listing; (vi) experimental factors selection; (vii) experimental design; (viii) experimental implementation; (ix) experimental analysis; and (x) conclusion and reporting. However, these steps are specific to the type of applications (DIY CDN, web applications, and the like) being evaluated as well as the target computing infrastructure (web services, Clouds, grids, and the like).

In this paper, we: (i) present concepts and factors related to hosting DIY content management applications over public Clouds; (ii) present extensive performance evaluation techniques for quantifying and understanding the runtime configuration management of DIY content management applications; and (iii) conduct several benchmark driven experiments for understanding the runtime configuration of DIY content management applications and service configuration (virtual machines, BLOB storage, Cloud location, pricing, etc.) offered by public Clouds.

The remainder of this paper is organized as follows. Section 2 summarizes the existing work related to CDNs and Cloud service evaluation methodology. Section 3 briefly introduces the MediaWise Cloud. The comprehensive methodology of evaluating MediaWise Cloud is specified step by step in Section 4, with reporting the early-stage experimental results. Conclusions and some future work are discussed in Section 5.

2. Related work

Due to recent emergence of the Content Distribution Networks (CDN), increasing attention has been paid to their performance characteristics [5]. For example, Johnson et al. [6] conducted one of the earliest CDN performance evaluations to compare different commercial products. More specifically, performance evaluation has been employed to study different CDN architectures [7]; and some other studies used to study performance evaluation to identify the best algorithm that met hierarchical streaming requirements [8]. Given the emerging computing paradigm—Cloud computing, researchers and practitioners have started using network virtualization and Cloud techniques to satisfy CDN requirements. In particular, suitable Cloud resources are used to help provide viable and cost-effective solutions for realizing CDN services [9]. Since Cloud usage requires deep understanding of how the relevant Cloud services may (or may not) match particular demands, Cloud services evaluation would act as a crucial part of CDN performance evaluation.

Compared to the traditional CDN systems, the performance evaluation of Cloud-based CDN systems would be more challenging. There are two main reasons for this. First, the back-ends (e.g., configurations of physical infrastructure) of Cloud services are normally uncontrollable (often invisible) from the perspective of consumers. Unlike consumer-owned computing systems, Cloud users have little knowledge and control over the precise nature of Cloud services even in a "locked down" environment [10]. Evaluations in the context of Cloud computing are then inevitably more challenging than that for systems where the customer is in direct control of all aspects [11]. Second, the open indicators (e.g., compute units, memory size, service price, etc.) often lack the provision of comprehensive information about a service regarding specific tasks [12]. As a result, there could be uncertainty in the runtime of Cloud services due to the various Quality of Service (QoS) consumption requirements of different applications [13]. Consequently, service evaluation would be one of the prerequisites of employing Cloud-based systems.

When it comes to performance evaluation of Cloud-based CDN systems, it has been recognized that Cloud services evaluation [14–16] belongs to the field of experimental computer science [11] which requires a suitable evaluation methodology as a strategic role in directing experimental studies [17]. An evaluation methodology instructs a complete evaluation implementation that may cover various aspects, for instance, workload selection, experimental design, and result analysis [17]. Therefore, a concrete methodology adopted in Cloud service evaluation should distinguish