Future Generation Computer Systems 29 (2013) 46–60

Contents lists available at SciVerse ScienceDirect

Future Generation Computer Systems

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

Towards an optimized abstracted topology design in cloud environment[☆]

Rosy Aoun[a,*], Chinwe E. Abosi[b], Elias A. Doumith[c], Reza Nejabati[b], Maurice Gagnaire[c], Dimitra Simeonidou[b]

[a] Department of Computer Science, Notre Dame University-Louaize, Zouk Mikael, Lebanon

[b] School of Computer Science and Electronic Engineering, University of Essex, Wivenhoe Park, Colchester, CO4 3SQ, UK

[c] Telecom ParisTech - LTCI - UMR 5141 CNRS, 46, rue Barrault F 75634 Paris Cedex 13, France

ARTICLE INFO

Article history:
Received 25 May 2011
Received in revised form
25 March 2012
Accepted 29 March 2012
Available online 23 April 2012

Keywords:
Cloud computing
Topology abstraction
Resource virtualization
Service provisioning
Scheduled traffic
Distributed storage
Integer linear programming
Heuristic

ABSTRACT

The rapid development and diversification of Cloud services occurs in a very competitive environment. The number of actors providing Infrastructure as a Service (IaaS) remains limited, while the number of PaaS (Platform as a Service) and SaaS (Software as a Service) providers is rapidly increasing. In this context, the ubiquity and the variety of Cloud services impose a form of collaboration between all these actors. For this reason, Cloud Service Providers (CSPs) rely on the availability of computing, storage, and network resources generally provided by various administrative entities. This multi-tenant environment raises multiple challenges such as confidentiality and scalability issues. To address these challenges, resource (network, computing, and storage) abstraction is introduced. In this paper, we focus on network resource abstraction algorithms used by a Network Service Provider (NSP) for sharing its network topology without exposing details of its physical resources. In this context, we propose two network resource abstraction techniques. First, we formulate the network topology abstraction problem as a Mixed-Integer Linear Program (MILP). Solving this formulation provides an optimal abstracted topology to the CSP in terms of availability of the underlying resources. Second, we propose an innovative scalable algorithm called SILK-ALT inspired from the SImple LinK (SILK) algorithm previously proposed by Abosi *et al*. We compare the MILP formulation, the SILK-ALT algorithm, and the SILK algorithm in terms of rejection ratio of users' requests at both the Cloud provider and the network provider levels. Using our proposed algorithms, the obtained numerical results show that resource abstraction in general and network topology abstraction in particular can effectively hide details of the underlying infrastructure. Moreover, these algorithms represent a scalable and sufficiently accurate way of advertising the resources in a multi-tenant environment.

1. Introduction

Cloud computing is a service paradigm that has emerged as a result of distributed Information Technology (IT) resources across the Internet. It provides access to *heterogeneous* IT resources, which can either be physical or virtual, as services over the Internet [1]. Examples of provided resources include storage resources such as those provided by Amazon S3 [2], computational resources such as Amazon EC2 [3] and applications such as the Google App Engine [4].

In the last few years, the number of providers offering IT Cloud services has increased very rapidly [5]. Furthermore, these Cloud Service Providers (CSPs) are supposed to provide the end-users a scalable distributed computing environment capable of achieving Quality of Service (QoS) targets in terms of availability and response time. In order to accommodate the rising demand for Cloud computing resources and the heterogeneous requirements of emerging applications, CSPs may have to connect to, share, or rent additional resources from other service providers [6]. For a given CSP, all other providers that offer access to additional resources and services are referred to as Third-Party Service Providers (TPSPs). For example, IBM [7] and Google are beginning to join their research in order achieve a higher profit from Clouds [8]. In this case, IBM is considered as a TPSP for Google. Through the aggregation and/or renting of resources, CSPs may have access to an almost unlimited pool of computing, storage and network facilities. The infrastructure owned separately by each of these providers may have to be transparently interconnected using high capacity and low latency connections. Wavelength

☆ This work has been supported by the Bone Network of Excellence.
* Corresponding author. Tel.: +961 76334464.
E-mail addresses: raoun@ndu.edu.lb, rosyaoun@gmail.com (R. Aoun), ceabos@essex.ac.uk (C.E. Abosi), elias.doumith@telecom-paristech.fr (E.A. Doumith), rnejab@essex.ac.uk (R. Nejabati), maurice.gagnaire@telecom-paristech.fr (M. Gagnaire), dsimeo@essex.ac.uk (D. Simeonidou).

0167-739X/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.future.2012.03.024

Division Multiplexing (WDM) transmission systems and the recent advancement in multi-granular switching technology [9,10] provide the bandwidth capacity and flexibility needed to achieve these connections. Web-based Cloud services are offered at a large scale, for instance nationwide or at the scale of a continent. By their very nature, optical networks are the best cost-efficient way to provide high bit rate and low latency interconnection links between the remote tenants cooperating towards the same Cloud service.

In this context, generic resources assignment rules have to be specified, for instance by means of Service Level Agreement (SLA) standardization. To meet required SLAs, a TPSP may have to share resource-state information. Meanwhile, as they are competitors on the same large scale market, TPSPs are in general reluctant to expose detailed information about their resources due to *confidentiality* and trust concerns. In addition, disseminating the full view of the underlying infrastructure of a single or several NSPs should generate a heavy volume of signaling traffic. To address these challenges, resource abstraction is proposed.

Resource abstraction facilitates Cloud computing by dealing with resource heterogeneity, scalability, and confidentiality issues that impact, in general, the sharing and renting of distributed resources. Resource abstraction allows TPSPs to hide the details of their resources by summarizing physical resource information by means of reduced approximations. Resource abstraction provides a way for TPSPs to represent, advertise, and rent out their available resources in a uniform and scalable manner. The resulting abstract representation is chosen such that enough information is provided to client CSPs, enabling them to provide services over the abstracted infrastructure at a targeted QoS.

In a Cloud computing context, resource abstraction refers to network and IT resources abstraction. In this article, we focus on network abstraction. The Network Provider (NP) that acts as a TPSP computes an abstracted topology of its network domain. This abstraction transforms the physical topology into a reduced graph that can be offered as an Infrastructure as a Service (IaaS) to other CSPs. Thus, in the remainder of this article, we refer to the NP as a Network Service Provider (NSP) that serves as TPSP to the benefit of the CSP.

In this paper, we propose an original exact Mixed-Integer Linear Programming (MILP) formulation to enable us to determine the best abstracted network topology to be offered as a service to the CSP. Indeed, as it is explained below, an abstracted topology may be acceptable for the CSP in terms of physical connectivity but unsuited due to bandwidth expectations and restrictions. We compare for similar scenarios, the abstracted topology obtained by means of the proposed MILP formulation with those obtained by means of two approximate algorithms. The first of these two algorithms is proposed in [11], while the second one is an improved version of this same algorithm and is proposed in this paper. Three different types of requests generated by the end-users are considered: computing requests, storage requests and point-to-point data transfer requests. We assume that IT resources that are interconnected via the optical network are abstracted using the single-aggregate scheme [11]. Finally, we propose a provisioning algorithm based on the Simulated Annealing (SA) meta-heuristic to satisfy users requests over the adopted abstract infrastructure. A request may be accepted in terms of requested connectivity and bandwidth on the abstract topology but be subject to rejection by the NSP at its instant of activation, if the bandwidth capacity reserved to this request by the CSP over an abstracted link appears to be unavailable at the physical path serving this abstracted link. We call such a request service disruption "crank-back". In order to compare the performance of the three abstraction approaches, we evaluate the number of accepted requests and their crank-back ratio for different traffic loads. We define the crank-back ratio as the ratio of the number of rejected requests by the NSP (or the crank-back number) to the number of accepted requests by the CSP.

The remainder of this paper is structured as follows. In Section 2, we provide further background on the Cloud computing service delivery paradigm and discuss some of the work done in the literature on resource abstraction. In Section 3, we detail the interaction between the CSP and the NSP (the NSP acting as a TPSP). We also specify the outcomes expected from the proposed abstraction algorithms. Section 4 introduces the formal model of the Cloud environment, including the infrastructure model and the traffic model. Section 5 introduces the two approximate abstraction algorithms, namely SILK and SILK-ALT, in addition to the exact MILP formulation. For completeness, Section 6 presents the IT abstraction model considered in this paper. In Section 7, we explain the two resource provisioning algorithms used by the CSP and the NSP, respectively. The simulations conducted in order to evaluate these algorithms consider a 6-node bottleneck topology and are presented in Section 8. Finally, our conclusions are drawn in Section 9.

2. Background

2.1. Cloud computing

Cloud computing is a rapidly evolving Internet service delivery paradigm. It allows providers to offer Software as a Service (SaaS), databases and Virtual Machines (VM) for building and running custom applications known as Platform as a Service (PaaS), and network resources known as Infrastructure as a Service, (IaaS) [12]. All these resources are mutualized between multiple end-users.

SaaS provides access to remote computer applications via the Internet, rather than installing and running the application on users own computers. SaaS currently include online project management tools from Clarizen [13], as well as customer relationship management and human resource applications available from Salesforce [14]. A number of Cloud office applications are available as desktop tools including word processing, building databases, creating spreadsheets, and presentations, as it is the case with Google Docs [4].

PaaS delivers a computing platform and solution stack as a service, often consuming Cloud infrastructure and maintaining Cloud applications. It facilitates the deployment of applications without the cost and complexity of buying and managing the underlying hardware. At the PaaS level, not only is an execution environment provided, but also a set of infrastructure services (given by IaaS providers). A platform should be able to provide an environment comprising the End-to-End (E2E) life cycle of developing, testing, deploying, and hosting Web applications. Amazon was the first vendor to provide PaaS services by launching "Amazon Web Services" (AWS) [15]. The AWS services rely on Amazon infrastructure services. Another major PaaS vendor is Google that has launched a service called "App Engine" where developers can run Web applications on Google's infrastructure [4]. Microsoft also developed a platform called "Azure" as an online service offering flexible, familiar environment for developers to create Cloud applications and services [16].

In addition to SaaS and PaaS, Cloud computing also includes the development of **IaaS** where the underlying infrastructure, such as computer processing capacity, network bandwidth, and equipment, is offered to end-users as well as to other CSPs as a service. Rather than buying their own servers, data-center spaces, or network equipment, end-users access these resources as a fully outsourced service. IaaS can be achieved thanks to the abstraction and virtualization of resources, where a logical infrastructure, which may be a slice of the physical infrastructure,