Exploiting an Elastic 2-Tiered Cloud Architecture for Rich Mobile Applications

M. Reza Rahimi[1],
Bren School of Information and Computer Science, University of California, Irvine, USA.
email:{mrrahimi}@ics.uci.edu

***Abstract*—This dissertation exploits 2-Tier cloud architecture for rich mobile applications. It is based on the observation that using local resources in close proximity to the user, i.e. local clouds or cloudlets (in the second tier), can increase the quality and performance of mobile applications. In contrast, public cloud offerings (in the first tier) offer *scalability* at the cost of higher delays, higher power consumption and higher price on the mobile device. We introduce new concept of *Space-Time* workflow as the modeling framework for mobile applications. Based on this framework we try to define the optimal mobile applications (considering power, price and delay as the Qos). This problem in general is NP-Hard. By considering a class of heuristics called CRAM we try to achieve near optimal solution. We will show the efficiency of our developed techniques through simulation and prototyping of different classes of rich mobile applications such as signal processing and streaming applications.**

I. Introduction

The explosive growth of wireless networking , mobile computing and web technologies in the last decade has profoundly influenced society at large. Almost anyone with access to a mobile device has access to services on the Internet and has the benefits of instant accessibility to Internet-enabled technologies such as mapping applications, media streaming applications, and email.

Cloud Computing environment enables a new framework that transfers the physical location of computation and storage into the network to reduce operational and maintenance costs. While mobile computing provides ubiquitous access of users to services, cloud computing harnesses the vast storage, computing, and software infrastructure resources into a single virtualized infrastructure.

One of the most important bottlenecks in ensuring mobile QoS is the *level of wireless connectivity offered by last hop access networks* such as 3G and Wi-Fi. These networks exhibit varying characteristics. For example, 3G networks offer wide area ubiquitous connectivity; however, 3G connections are known to suffer from *long delay* and *slow data transfers* [5] resulting in increased power consumption and cost at the user side. In contrast, Wi-Fi, exhibit low communication delays, connected to or collocated with Wi-Fi access points can be used to form a nearby local cloud [3], [5]. Using a local only solutions with Wi-Fi networks creates *scalability* issues; as the number of users increases the *latency* and *packet losses* increase causing a decrease in application performance [1].

[1]Advised By Prof. Nalini Venkatasubramanian, nalini@ics.uci.edu

978-1-4673-1239-4/12/$31.00 ©2012 IEEE

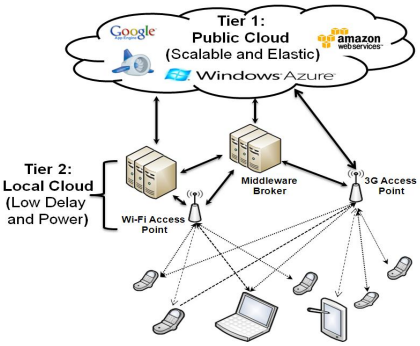

Fig. 1. 2-Tier Mobile Cloud Architecture.

In this dissertation, we will consider a *2-Tier Elastic Cloud Architecture* for the mobile applications that synergistically combines the capabilities of **local clouds and public cloud offerings** to increase the **performance** and **scalability** of mobile applications. Specifically, we will develop efficient techniques for discovering and allocating resources in such a tiered cloud architecture to meet the multidimensional QoS (***price, power, delay***) for mobile applications. In the next sections the general system architecture, the challenging problem formulation and modeling, achieved solutions and future directions and extensions will be reviewed.

II. General System Architecture

Fig. 1 shows the 2-tier cloud architecture for mobile applications. Tier 1 nodes in the system architecture represents public cloud resources such as Amazon Web Services and Google Application Engine. Services that are provided by these vendors provide high *scalability* and *availability on demand*, but do not have *fine grain location granularity* that are required for high performance mobile applications (in the best case they have *city level* granularity)[6]. The second tier, i.e. local cloud consists of nodes that are connected to access points - location information of these resources are available at finer levels of granularity (campus and street level). Mobile users are typically connected to local clouds through Wi-Fi (via access points) or cellular (via 3G cell towers) connectivity - this enables us to offload device tasks onto cloud nodes.

To manage and run mobile applications in this architecture we need a middleware broker which exploits this architecture efficiently. It performs task/resource mapping at a *middleware broker* node [2]. The broker maintains a registry of resources and services and resource allocation in both tiers of the cloud.

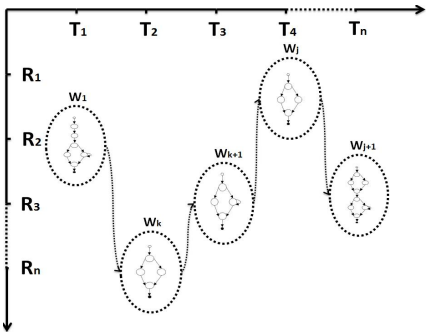

Fig. 2. Space-Time Workflow

A. Mobile Application Modeling

To formally model services and applications in the 2-tier cloud architecture, we use terminologies and techniques from Service Oriented Architecture (SOA). We have extend the Workflow concept to the novel concept called **Space-Time Workflow**. It models the mobile applications in pervasive environment completely. As it has been shown in Fig. 2 it consists of sequence of workflows which are indexed by mobile user's *region* (in space) and *time* or in other words it could be presented mathematically as:

$$W(u_k)_T^S = (w(u_k)_{t_1}^{r_1}, w(u_k)_{t_2}^{r_2}, w(u_k)_{t_1}^{r_2}, ..., w(u_k)_{t_n}^{r_n})$$

where u_k is the k^{th} mobile user and $w(u_k)_{t_n}^{r_n}$ is the user request workflow in region r_n and time t_n. We defined the optimal mobile cloud computing for u_k as the optimization problem for space-time workflow. Our objective is to *maximize the savings on price, power or delay* subject to constraints on available power, delay and price resources. It can be formally stated as:

$$\begin{aligned} &\max \min\{[\hat{W}(u_k)_T^S]_{power}, [\hat{W}(u_k)_T^S]_{delay}, [\hat{W}(u_k)_T^S]_{price}\} \\ &\quad subject\ to: \\ &\quad [W(u_k)_T^S]_{price} \leq C_{price} \\ &\quad [W(u_k)_T^S]_{power} \leq C_{power} \\ &\quad [W(u_k)_T^S]_{delay} \leq C_{delay} \end{aligned} \tag{1}$$

$[\hat{W}(u_k)_T^S]_{power}, [\hat{W}(u_k)_T^S]_{delay}, [\hat{W}(u_k)_T^S]_{price}$ are the **Normalized** power, delay and price for mobile application [1]. This problem in general is NP-Hard. The goal of this dissertation is to extend a class of heuristics which is called **CRAM** for solving it.

B. Cloud Resource Allocation for Mobile Applications (CRAM)

The CRAM algorithm is a **greedy heuristic** that generates a near-optimal solution to the tiered cloud resource allocation problem using a **simulated annealing** based approach. There are two intuitions behind such a greedy selection [1]:

- It is known that services in close proximity to the user usually provide better Qos performance in terms of delay and power consumption.
- Using services with high QoS will increase system utility. In our context, improved QoS can be realized using one of four metrics – normalized delay, power, price and total normalized QoS [1].

We have implemented and tested CRAM for single users (S-CRAM)[1] and it has been shown through simulation that it could reach about 84% of optimal solution. As the *future work* and current research direction we try to extend CRAM into 3 other categories:

- **Multiple Users CRAM**: Try to find optimal solution for group of users requesting an application.
- **Mobility Aware Single User CRAM**: Try to find optimal solution of single user knowing it mobility pattern.
- **Mobility Aware Multiple Users CRAM**: Try to find optimal solution of group of users requesting an application knowing their mobility path.

C. Different Mobile Applications Classes

We try to use the solutions an techniques from previous sections to use on real mobile applications. Two different classes of mobile applications will be considered in this dissertation; **intensive computing** and **streaming** applications.

1) ***Intensive Computing Applications****:* This class of mobile application contains intensive signal processing tasks. OCR+Speech (OCRS) and video content augmented reality (VCAR) applications have been developed as the rich mobile applications. In the first application the user takes a picture of the text page and the application will return a file which contains the spoken text. In the second application the user captures the video from a mobile device and uploads the video to the server. On the server, Augmented Reality will be run to infuse the 2D object in video. The resulted video will be send back to the user. 9 different RESTFul web services have been developed, and each service delay, power consumption and price has been measured. The results form prototyping will be used to tune the CloudSim [4] simulator to test the efficiency of the algorithms.

2) ***Streaming Applications****:* Simple video album Streaming application will be considered in this class. The application will contains data caching, transcoding and streaming services. Again each services will be implemented as the RESTful web service and extensive profiling and simulation will be used to measure the performance of different CRAM-based algorithms for such streaming applications.

REFERENCES

[1] M. Reza. Rahimi, N. Venkatasubramania "*MAPCloud: Mobile Applications on an Elastice 2-Tier Cloud Architecture*", submitted to *IEEE GLOBECOM 2012*.

[2] M. Reza. Rahimi, N. Venkatasubramania "*Cloud Based Framework for Rich Content Mobile Applications*", poster in the *IEEE/ACM CCGrid 2011*.

[3] Byung-Gon Chun, Sunghwan Ihm, Petros Maniatis, Mayur Naik, Ashwin Patti " *CloneCloud: Elastic Execution between Mobile Device and Cloud*", In *EuroSys 2011*.

[4] Rodrigo N. Calheiros, Rajiv Ranjan, Anton Beloglazov, Cesar A. F. De Rose, and Rajkumar Buyya, " *CloudSim: A Toolkit for Modeling and Simulation of Cloud Computing Environments and Evaluation of Resource Provisioning Algorithms*", Wiley Press, 2011.

[5] Eduardo Cuervo, Aruna Balasubramanian, Dae-ki Cho, Alec Wolman, Stefan Saroiu, Ranveer Chandra, and Paramvir Bahl " *MAUI: Making Smartphones Last Longer with Code Offload*", In *MobiSys 2010*.

[6] M. Satyanarayanan, P. Bahl, R. Cceres, N. Davies " *The Case for VM-Based Cloudlets in Mobile Computing*", In *PerCom 2009*.