International Journal of Application or Innovation in Engineering & Management (IJAIEM)
Web Site: www.ijaiem.org Email: editor@ijaiem.org, editorijaiem@gmail.com
Volume 2, Issue 4, April 2013 ISSN 2319 - 4847

Opportunistic Job Sharing For Mobile Cloud Computing

[1]Paridhi Vijay, [2]Vandna Verma

[1]B.E, Computer Science and Engineering, Rajasthan College of Engineering for Women, Jaipur, Rajasthan

[2]Asst. Professor (CSE Dept.), Rajasthan College of Engineering for Women, Jaipur, Rajasthan

ABSTRACT

Cloud computing is an emerging concept of computing technology of the digital era. Mobile computing & its applications in smart phones enable a new, rich user experience. Explosive usage of limited resources in smart phones leads to problems such as battery life, memory, feasibility and CPU. To solve this problem, we propose a dynamic mobile cloud computing architecture framework to use global resources instead of local resources. In this proposed framework the usefulness of job sharing workload at runtime reduces the load at the local client and the dynamic throughput time of the job through Wi-Fi Connectivity.

Keywords: - Cloud Computing, Offloading, Smartphone, Wi-Fi.

1. INTRODUCTION

Cloud Computing technology maintain data and application using central remote server. It allows consumers to use applications without installation their personal files at any computer with internet access. Mobile computing is an interaction between human and computer by which computer is expected to be transported during usage. It includes mobile hardware, mobile communication n mobile software [4]. The greatest feature of the mobile cloud computing is that it allows user to connect its relevant data from anywhere in the world via network. Problems occur when trying to support mobility in computing devices: resource sparseness, hazardousness, finite energy source, and low connectivity [5].

Challenges in framework are job partition, job distribution and connectivity options in the cloud devices. In the job partition and distribution, offloading phenomena is based on the number of frames sent by the cloud client & how fast can server receive & process that data [10]. In this paper we refer job sharing based algorithm so that each connected devices gets their part of work and using offloading process each one can do their work properly & acknowledges to the central server. In the connectivity, previous work is done over the Bluetooth network due to which only local and limited resources can be utilized. In this paper we used Wi-Fi as connectivity option. Using Wi-Fi based architectural framework we can utilize all the global resources via network connectivity but not only limited to the local resources. Cloud is available for low end mobile device as well as high end mobile device in this framework. Most of the cloud resource would be mobile, computer, laptop etc. Dynamic mobile cloud framework can handle run time resources and connectivity. In the framework we explain vision towards the process large amount of job which requires huge hardware resources with smart phones by partitioning the task into the number of jobs which is cost-saving, battery-life saving. Using this architectural framework huge task can be done in just a matter of time using global resources.

2. MOTIVATION FOR THE WORK

Let's consider the scenario of Mr. John (Picture editor) travelling in a bus. He suddenly gets an email to edit a large size image. He starts editing. Since the large size image need to be edited only on laptop because it cannot be edited on the smart phones due to memory constraints, limited battery power & low CPU processing of smart phones. As the matter of the fact he cannot edit the image.

In this scenario if he has a dynamic mobile cloud computing framework through this he can create a cloud using network (Wi-Fi) then the result would be different. He uploads image to the central server (cloud) using Wi-Fi network & asks some of his colleagues to do it. All the cloud clients (colleagues) edit the particular part of the image and again send back the response to the server. Central server processes all the responds and again sends back to the John.

In this way John explores the dynamic architectural framework by using sharing/offloading process to complete his job and moved over four major challenges: reduce bulkiness, time-saving, limited memory and battery power. Now John is still available to do any urgent work which is the best part of using this framework.

International Journal of Application or Innovation in Engineering & Management (IJAIEM)
Web Site: www.ijaiem.org Email: editor@ijaiem.org, editorijaiem@gmail.com
Volume 2, Issue 4, April 2013 ISSN 2319 - 4847

3. PROPOSED ARCHITECTURE

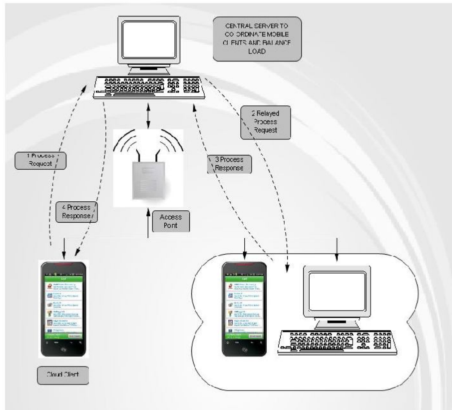

Figure 1 Main component of a cloud framework

The three main components of the architectural framework are cloud client, central server and ad-hoc network.

CLOUD CLIENT: It is in charge of launching and intercepting an application at loading time. It's like a master component of the cloud. This client sends request to the central server. SOAP protocol is used as communicating medium among the connected devices. This is the master user as this controls the all the query. It offloads all it works to the central server.

SOAP sender: Cloud Client

CENTRAL SERVER/Resource Manager:-It is the heart of the architecture. It gets all the SOAP requests from the master that is cloud client and converts into XML language. This server uses basic job sharing algorithm for distribute the job & intimidate to the other cloud connected devices according to their resource and capabilities. It acts as a resource manager.

SOAP message path: Central Server

AD-HOC NETWORK/Job Handler: - It is the bunch of connected devices which is responsible for the load balance. These are kind of slave devices who acts on getting the SOAP request from the server. Whole devices share the same cloud and every device gets the SOAP request from the central server depending on the size of task from the master cloud client. Once the jobs have been distributed, the clients would proceed to execute their job/s. When the job handler (client) devices finish their job, result are sent back to the master and reassembled.

SOAP receiver: Ad-hoc network

4. PROPOSED WORK AND IMPLEMENTATION

4.1. Job Scheduling:-

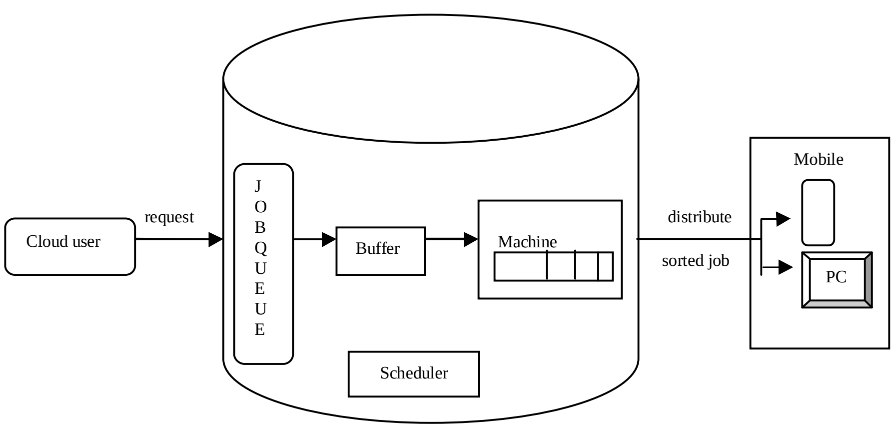

Figure 2 Job Scheduling System

Proposed Algorithm described as follows:

Step 1: Cloud user send job request to the server.

Step2: Job request will be store in the JOB QUEUE accrording to their occurence time.

Step 3: Select the ready job from the JOB QUEUE and put into BUFFER according to the job.

Step 4: Place this job into MACHINE and process the job according to the FCFS (First Come First Serve) algorithm method.

Step 5: Scheduler distribute the sorted list according to the mobile client and balance loader and send to the Resource pool.

Step 6: Repeat Step 3 to 5 for next set of job.

Advantage of the system:-

i. More relaible
ii. Low cost resource
iii. Less execution time

4.2. Image Processing:-

Image processing is a form of signal processing for which the input is an image, the output of image processing may be either an image or a set of characteristics or parameters related to the image [13]. Image processing are computer graphics and computer vision. Image processing is a process to convert an non edited image into more clear image through converting into digital signals in order to get more detailed image or to apply some more effects on it .The purpose of image processing are image sharpening, restoration , visualization ,image recognition etc.

4.2.3 Convolution Operation:-

In image processing, many operators are based on applying some function to the pixels within a local window to finding the value of an output pixel, a window is centre at that location, and only the pixels falling within this window are used when calculating the value of that output pixel. Applying the convolution operator, the function we apply is merely a weighted average of the within-window pixels [12].
If we let *f* be the image we want to filter, *g* the corresponding output image, and let *l* be the convolution kernel, we have [12]:

$$g(x,y) = \sum_{i=-p}^{p} \sum_{j=-p}^{p} l(i,j) f(x-i, y-j) \qquad (1)$$

Where the size of the kernel is (2p+1) × (2p+1).
The convolution operator is linear, that is, we get the same result if we perform the convolution on two separate images and sum their results as if we were to sum the two images before we apply the convolution. According to the convolution theorem, applying convolution is equivalent to a per-frequency multiplication in the frequency domain. That is, if we were to change the basis for both the convolution kernel and the image to one that consists of simple sine and cosine functions (applying a discrete Fourier transform), we can take each of these components and multiply them and get the same result [11].
Let f and g be two function with convolution $f * g$ (asterisk denotes convolution in this context not multiplication). Let $\mathcal{F}$ denote Fourier transform operator, so $\mathcal{F}\{f\}$ and $\mathcal{F}\{g\}$ are the Fourier transform of f and g, respectively [11].

$$\mathcal{F}\{f * g\} = \mathcal{F}\{f\} \cdot \mathcal{F}\{g\} \qquad (2)$$

Where $\cdot$ denotes point-wise multiplication.

$$\mathcal{F}\{f \cdot g\} = \mathcal{F}\{f\} * \mathcal{F}\{g\} \qquad (3)$$

By applying the inverse Fourier transform: $\mathcal{F}^{-1}$

$$f * g = \mathcal{F}^{-1}\{\mathcal{F}\{f\} \cdot \mathcal{F}\{g\}\} \qquad (4)$$

5. RELATED WORK

In this section we provide a review of related research efforts, ranging from the earlier approaches that focus two methods relating to offloading, job scheduling work from mobile.
Marinelli [2] introduce Hyrax, a mobile cloud computing client that allows mobile devices to use cloud computing platforms. Based on Hadoop1, the main focus of this work is to port a client into a mobile device to enable the integration. The author introduces the concept of using mobile devices as resource providers, but further experimentation is not included.

Integration between mobile devices and cloud computing is presented in several previous works. Christensen [1] presents general requirements and key technologies to achieve the vision of mobile cloud computing. The author introduces an analysis on smart phones, context awareness, cloud and restful based web services, and explains how these components can interact to create a better experience for mobile phone users.

Fernando, W. Loke and Wenny Rahayu [10] introduce the feasibility of a mobile cloud computing framework to use local resources. The framework aims to determine a priori the usefulness of sharing workload at runtime. The results of experiments conducted in Bluetooth transmission.

6. CONCLUSION

The concept of cloud computing and job sharing over cloud provides a brand new opportunity for the development of mobile applications that can get heavy tasks done over cloud by offloading computation tasks on cloud, since it allows the mobile devices to maintain a very thin layer for user applications and shift the computation and processing overhead to the virtual environment. Using the proposed framework the usefulness of job sharing workload at runtime reduces the load at the local client and the dynamic throughput time of the job through Wi-Fi Connectivity instead of the Bluetooth.

REFERENCE

[1] J.H. Christensen, "Using Restful web-services and cloud computing to create next generation mobile applications," Proceeding of the 24th conference on Object oriented programming systems languages and applications - OOPSLA '09, New York, New York, USA: ACM Press, 2009.

[2] E. Marinelli, "Hyrax: Cloud Computing on Mobile Devices using Map Reduce,", Master Thesis Draft, Computer Science Dept., CMU, September 2009.

[3] K. Kumar and L. Yung-Hsiang, "Cloud Computing for Mobile Users: Can Offloading Computation Save Energy?," IEEE Computer , vol.43, no.4, pp.51-56, April 2010. doi: 10.1109/MC.2010.98

[4] Mobile computing - Wikipedia, the free encyclopaedia en.wikipedia.org/wiki/Mobile computing

[5] M. Satyanarayanan. Fundamental challenges in mobile computing. In Proceedings of the fifteenth annual ACM symposium on Principles of distributed computing, PODC ’96, pages 1–7, New York, NY, USA, 1996. ACM.

[6] S. Cherry. Update: Wi-Fi takes on Bluetooth. Spectrum, IEEE, 45(8):14, 2008.

[7] R. Kemp, N. Palmer, T. Kielmann, and H. Bal. Cuckoo: a computation offloading framework for smart phones. In Proceedings of the Second International Conference on Mobile Computing, Applications, and Services, MobiCASE ’10, 2010.

[8] M. Black and W. Edgar, "Exploring mobile devices as Grid resources: Using an x86 virtual machine to run BOINC on an iPhone," 2009 10th IEEE/ACM International Conference on Grid Computing, IEEE, 2009, pp. 9-16.

[9] Job Scheduling Strategies for Parallel Processing: 13th International Workshop JSSPP 2007 Seattle, WA, USA June 2007 edited by Eitan Frachtenberg, Uwe Schwiegelshohn.

[10] Niroshinie Fernando, Seng W. Loke and Wenny Rahayu "Dynamic mobile cloud computing: Ad Hoc and opportunistic Job Sharing" 2011 Fourth IEEE International Conference on Utility and Cloud Computing, ucc, pp.281-286, 2011.

[11] Weisstein, Eric W., "Convolution Theorem" from Math World.

[12] http://www.imageprocessingbasics.com Digital image processing tutorials and interactive applets.

[13] Wikipedia – Image processing http://en.wikipedia.org/wiki/Image_processing

AUTHOR

Paridhi Vijay received the B.E degree in Computer Science and Engineering from R.C.E.W, Jaipur in 2007 and pursuing M.Tech in Computer Science from R.C.E.W, Jaipur. She has one year of experience in teaching field in COMPUCOM College.