International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)
ISSN: 0976-1353 Volume 22 Issue 1 –MAY 2016.

Data Processing While Charging: Constructing a Distributed Computing Infrastructure Using Smartphones

PRIYANKA RASAL[#1] and SWETA KALE[*2]

[#]*PG Scholar, Dept. Of Information Technology, RMD Sinhgad College , Pune, India*
[*] *Assistant Professor, Dept. Of Information Technology, RMD Sinhgad College , Pune, India*

***Abstract*— Smartphone is a cellular telephone designed with several integral components such as operating system, web browsing and the capability to run software applications. Nowadays, the massive amounts of idle smartphones are plugged into a power source for reviving the battery. The computing capability on smartphones is increasing, but holds a sizeable computing infrastructure. In an enterprise, the smartphones that were supplied with limited computing infrastructure to its employees. Henceforth, an idle smartphone needs to be charged, consumes high energy and cost- effective, overnight. To execute a multi-task, a modernized computing infrastructure should be developed. Several technical challenges like CPU clock speed, variability in network bandwidth and lower availability are restricted in smartphones. This paper addresses to develop a multi-task system with distributed computing infrastructure using smartphones. Our experimental results demonstrate that the performance on distributed infrastructure is effective and efficient.**

***Index Terms*— Smartphone, Energy consumption, Cost-effective, Distributed infrastructure.**

I. Introduction

Today, various associations supply their workers with cell phones for different reasons [1]; a study from 2011 [2] reports that 66% of associations do as such and huge numbers of these associations have 75–100% of their representatives utilizing smartphones. For an instance, Novartis [3] (with 100,000 representatives in 140 nations) passed out smartphones for its representatives to oversee the messages, timetables, and additionally data about health issues. Such an infrastructure can possibly lessen both the capital costs and energy costs received by associations. The embedded processor design in smartphone works effectively rather than the desktop processors [2], [3]. On the off chance that the associations could misuse the smartphones passed out to their workers to run some bit of their workload, it is possible that the expense of their computing infrastructure could have been lessened. All the more, the associations can save their energy (by closing down their servers) when they offload certain tasks to smartphones with lessened power than the server CPU [4] [5].

Other than its potential advantages, acknowledging such a smartphone registering infrastructure confronts various difficulties. We look to verbalize these difficulties and construct a structure to make such a stage suitable. Specifically, the greatest hindrances in utilizing smartphones for processing are the battery life and data transfer capacity. On the off chance that a smartphone is used for overwhelming processing when being used by its proprietor, the battery might deplete and render the phone into unusable. Further, sending extensive volumes of processing information to the phone utilizing cellular systems is not viable since information utilization is ordinarily topped via bearers. In this way, we also imagined utilizing smartphones when they are being charged during the night, when dynamic use by phone proprietors is not likely. Additionally, the telephones will be stationary and will probably interface with WiFi in proprietors' homes; this will decrease data transfer capacity changes and permit the exchange of registering information to/from the phone at no expense.

The rest of the paper is organized as: Section II describes the related work carried out by researchers. Section III describes the novel distributed computing infrastructure. Section IV describes the performance validation through experimental settings and at last concluded in Section V.

II. Related Works

To the best of our knowledge, no prior study shares our vision of tapping into the computing power of smartphones. However, some efforts resemble certain aspects of Data processing while charging. And then the study is organized as follows:

a) Testbeds of smartphones and platform based distributed computing:

Publicly- accessible smartphone testbeds have been proposed [11, 12] to empower smartphone OS and mobile applications research. CrowdLab [13] and Seattle [14] give assets on volunteer gadgets. There additionally exist frameworks where clients voluntarily contribute the idle time on their PCs to computational errands (e.g., [15]). Conversely, our vision is not for the smartphone infrastructure to be utilized for exploration and testing, yet to empower energy and cost preserving funds for genuine associations. Besides, the issues that we address have not been considered by these associations. Additionally, kinds of MapReduce for smartphones have been executed (e.g., [16]). In any case, such association doesn't address the issues of identifying the idle phone usage and scheduling the tasks crosswise over telephones with assorted abilities.

The framework that is nearest to CWC is Condor [17]. Condor can be utilized to queue and schedule the assignments over a conveyed set of desktop machines. These machines are either committed to run jobs on them or worked by general clients for scheduled exercises. In the last case, Condor screens whether client machines are idle and saddles such idle out of gear CPU power to perform the calculations required by jobs. With the services of Condor with CWC, the key differences listed as:

- CWC tries to protect the charging profile of smartphones by means of its CPU throttling procedure. This is a test not tended by Condor, since desktop machines don't show such an issue.
- Desktop machines for the most part vary from their CPU clock speed, memory (RAM) and disk space. In a group, these machines are associated by means of Ethernet switches and this commonly results in uniform data transmission over machines. Accordingly, frameworks, for example, Condor don't commonly consider machine transfer speed in their scheduling tasks.

b) Participatory sensing:

Recent surveys, for example, [18], recommend the aggregate utilization of the detecting, storing, and manipulating the functions of smartphones. With participatory sensing [19], clients gather and examine different sorts of sensor readings from smartphones. Dissimilar to these associations, a recognizing part of CWC is that the information to be prepared does not start from the telephones. Also, CWC permits the executions of an assortment of assignments not at all like above, where normally a modified assignment (detecting) is supported [20].

c) Measurements of smartphones:

There have been estimation studies [21, 22] to portray regular network traffic and energy utilization on smartphones. Conversely, our attention is on creating an adaptable stage for social occasion estimations from the telephones in CWC. A few former studies [23, 24] have watched that telephones are idle still and are being charged for critical times in a day. We are however the first to perceive that these idle periods can be outfit to fabricate a distributed computing infrastructure.

d) Provisioning tasks on cloud services:

Former associations have too attempted to recognize when the utilization of cloud administrations is suitable (e.g., [25]), represents the difficulties included in utilizing them (e.g. [26]), or present solutions for provisioning applications (e.g., [27]). However, any organizations concentrate on conventional server-based cloud administrations.

III. DATA PROCESSING WHILE CHARGING (DPC)-A DISTRIBUTED COMPUTING INFRASTRUCTURE

In this section, we describe our task (job) execution model, and seek answers to the following:

- How can we predict task execution times?
- How can we implement automated task execution on smartphones without user input?
- How can we preserve user experience while the tasks are being executed on the phones?

The proposed DPC system is divided into four modules:

a) Task Model
b) Predicting task execution times
c) Automating task execution
d) Preserving user expectations

a) Task Model:

In CWC, a task is a system that performs a manipulation on a data record, for example, checking the quantity of events of a word in a content document. Like Map-Reduce, a central server segments an extensive input document into smaller pieces of information, transmits the segments to the cell phones. In the wake of getting the executable and the relating information, the phones execute the task in parallel and give back their outcomes to the central server upon the completion. The server coherently aggregates the returned results, contingent upon the tasks. For the word count examples, the server can essentially aggregate the quantity of events reported by every phone to figure the quantity of events in the original data.

b) Predicting task execution times:

When a task is booked on a phone, there are two key components that influence the finishing time. To begin with, it requires some investment to duplicate the executable (i.e., binary) and the partitioned data file to a phone. This relies on upon the transfer speed of the connection between the telephone and the focal server. Second, the same assignment takes diverse times to finish on various phones (contingent upon the computational abilities of the telephone). While "computational capacities" is expansive and can incorporate the pace of perusing a record from the disk (e.g., the SD card on a telephone) or the size and speed of the data cache, where we just concentrate on the CPU clock speed of a phone; a phone with a quick CPU (in GHz) ought to execute a given task in less time when contrasted with a phone with a moderate CPU.

c) Automating task execution:

One of the key prerequisites of CWC is the task to be executed without client data. The process of executing the task on smartphones is running an application (i.e., "application"). At the point, when a client needs to execute another assignment on her phone, she needs to download and introduce the application. This procedure requires human information for different reasons (e.g., Android clients need to physically accept the application authorizations). Such an instrument is unmistakably not adept for CWC, since the tasks are to be powerfully booked on smartphones and in this way can't require client's input. To run task on the phone, we influence a cross-stage system that uses the Java Reflection API for Android. With reflection, a Java executable (i.e., a .class document) can progressively stack different executables, instantiate their objects and execute their techniques, at runtime. This permits CWC to dispatch distinctive task executables and input documents to a specific phone in a mechanized manner. In addition, reflection can be implemented as an Android service (running in the background), thus bypassing the need for human input. With reflection implemented on smartphones, CWC does not require any additional infrastructure at the central server.

d) Preserving User Expectations:

While booking tasks on phones are critical, we should note that they are close to personal gadgets. Firstly, it must guarantees that when a client uses her phone, CWC stops the execution of the last doled out task to that phone so as not to unfavorably affect the end-user experiences. The tasks that are accordingly ceased are then relocated to different phones that are still connected to and not in dynamic use. Secondly, running the task on phones that are connected to minimize the affect of charging times using phones' batteries. Simultaneously, also we notice that an overwhelming use of a phone's CPU draws power and drags out the time taken to completely charge a telephone's battery.

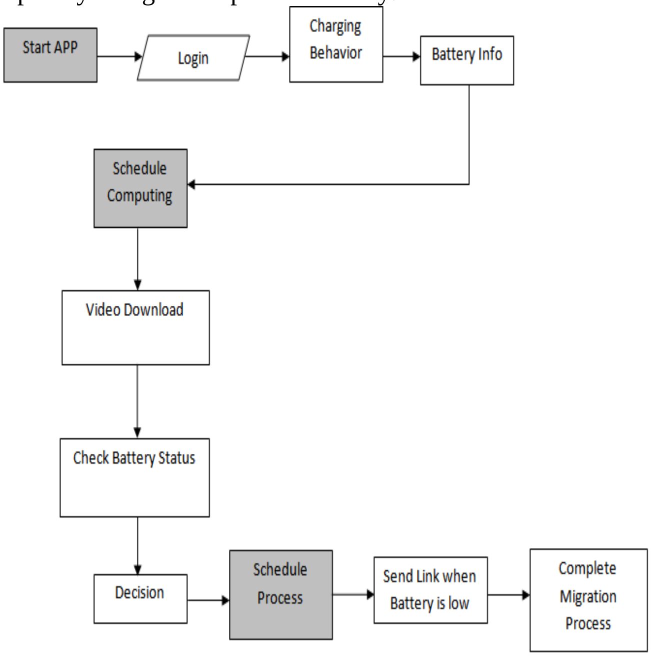


Fig.1. Proposed DPC –Workflow

IV. Experimental Designs and Results

Our testbed comprises of 18 Android telephones with changing network availability and CPU speeds. The network interfaces from WiFi (802.11a and 802.11g) to EDGE, 3G furthermore, 4G. The CPU clock speeds fluctuate from 806 MHz to 1:5 GHz. We measure bandwidth values with the iperf instrument. The phones holds the CWC programming, which keeps up a constant TCP organization with the server and licenses dynamic assignment execution as trained by the scheduler. The proposed Data processing while charging design is presented as follows:

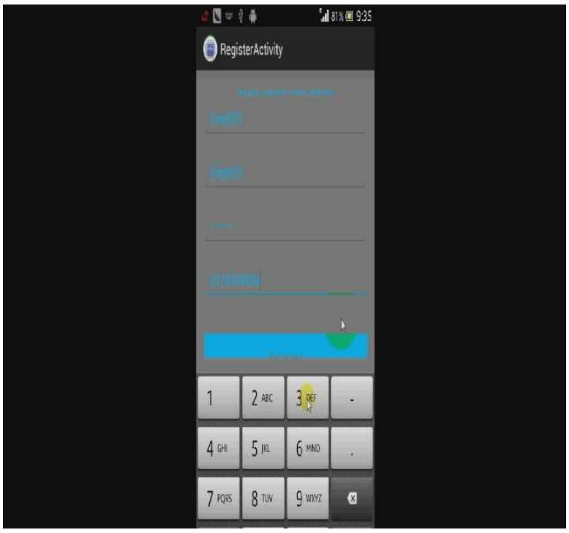


Fig.2. Employee's Registration activity

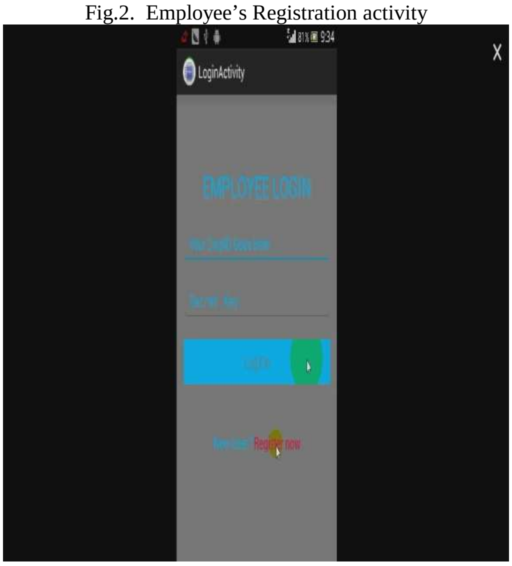


Fig.3. Employee login

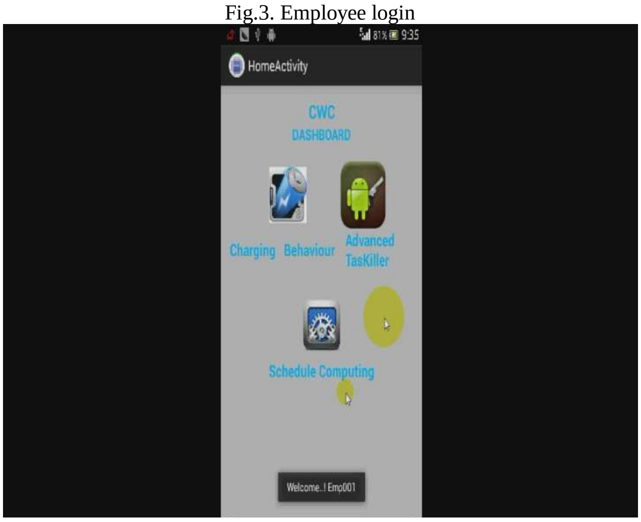


Fig.4. Gadget's in smartphones and clicking the "Schedule Computing"

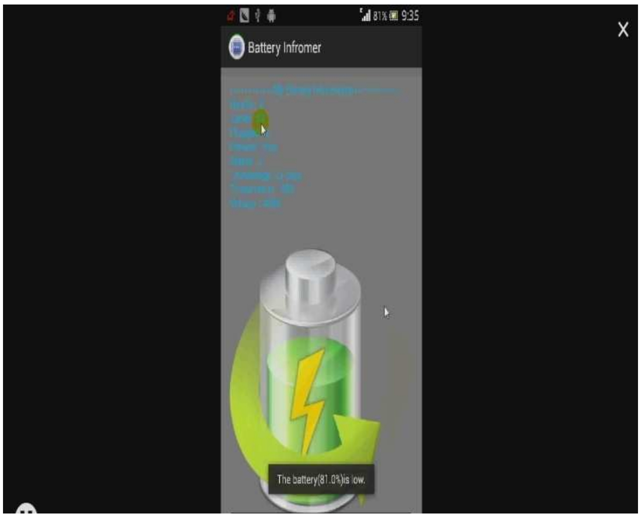


Fig.5. Knowing the current Battery status of the Smartphones

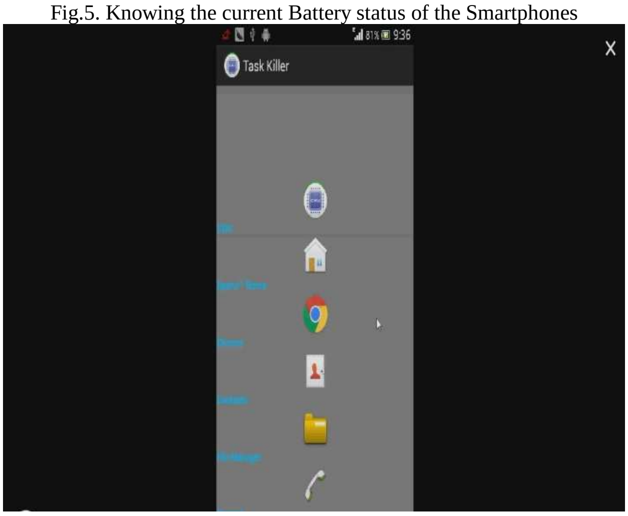
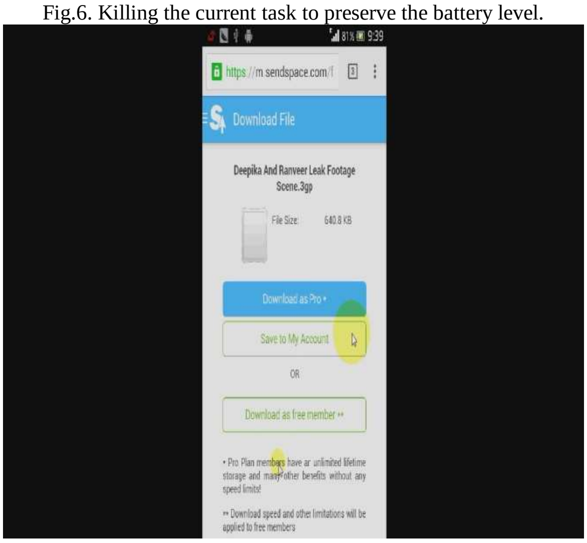


Fig.6. Killing the current task to preserve the battery level.



Deepika And Ranveer Leak Footage Scene.3gp

File Size: 640.8 KB

• Pro Plan members have ar unlimited lifetime storage and many other benefits without any speed limits!

•• Download speed and other limitations will be applied to free members

Fig.7. File Downloading process

V. CONCLUSION

Nowadays, Smartphones creates a great demand among the users. In this paper, we visualized the distributed computing infrastructure using smartphones. This is mainly studied for lessening the energy consumption and cost-effective in business use. Several technical challenges are CPU clock speed, bandwidth etc., are studied and proposed a novel structure to effectively utilize the scheduling system in smartphones. We implemented DPC program on a testbed of 18 android phones. Using this implementation, we successfully achieved our vision of the study..

REFERENCES

[1] Iphone in Business. http://bit.ly/LE4dAp.
[2] Enterprise Smartphone Usage Trends. http://bit.ly/loIqE1.
[3] Novartis: Apps for good health. http://bit.ly/KEdJbh.
[4] Lowe’s : Building better customer service. http://bit.ly/MiT9bk.
[5] IBM gets US$2mn data center contract from Novartis. bit.ly/L51P8i.
[6] NVIDIA says Tegra 3 is a "PC-class CPU". http://engt.co/srvibU.
[7] S. Harizopoulos and S Papadimitriou. A Case for Micro-Cellstores: Energy-Efecient Data Management on Recycled Smartphones. In DaMoN, 2011.
[8] Variable SMP - A Multi-Core CPU Architecture for Low Power and High Performance. bit.ly/n65KzQ.
[9] Smart Phone Chips Calling for Data Centers. bit.ly/LdM9fS.
[10] WiFi Bandwidth Use in the U.S. Home Forecast to More Than Double in the Next Four Years. http://yhoo.it/L9Po9A.
[11] PhoneLab. http://bit.ly/NYwRhI
[12] Almudeua Díaz Zayas and Pedro Merino Gómez. A testbed for energy profile characterization of IP services in smartphones over live networks. Mob. Netw. Appl.
[13] E. Cuervo, P. Gilbert, Bi Wu, and L.P. Cox. CrowdLab: An architecture for volunteer mobile testbeds. In COMSNETS, 2011.
[14] J. Cappos, I. Beschastnikh, A. Krishnamurthy, and T. Anderson. Seattle: A Platform for Educational Cloud Computing. In SIGCSE, 2009.
[15] SETI@home. http://setiathome.berkeley.edu.
[16] P. R. Elespuru, S. Shakya, and S. Mishra. MapReduce System over Heterogeneous Mobile Devices. In SEUS, 2009.
[17] http://research.cs.wisc.edu/condor/.
[18] Tathagata Das, Prashanth Mohan, Venkata N. Padmanabhan, Ramachandran Ramjee, and Asankhaya Sharma. PRISM: platform for remote sensing using smartphones. In ACM MobiSys, 2010.
[19] D. Estrin. Participatory Sensing: Applications and Architecture. In ACM MobiSys, 2010.
[20] N. Eagle. txteagle: Mobile Crowdsourcing. In Internationalization, Design and Global Development, 2009.