



# INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

## DEVELOPMENT OF AN EFFICIENT PLATFORM TO USE ANDROID AS A SERVER

MUKTA DUTONDE<sup>1</sup>, SACHIN JADHAV<sup>2</sup>

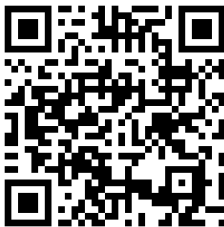
1. Computer Engineering Department, Student (M.E.), VBKCOE Malkapur, India.
2. Computer Engineering Department, Faculty of Engineering, VBKCOE Malkapur, Maharashtra, India.

Accepted Date: 05/03/2015; Published Date: 01/05/2015

**Abstract:** Nowadays the number of Smartphone users and mobile applications are growing rapidly. Smartphones want to adopt PC like functionality, but the hardware resources such as CPUs, memory and batteries are limited. For providing solution to this problem, many researches use server resources in the cloud for mobile devices. The system proposed development of an Efficient Platform to Use Android as a Server using network which enables multiple user Android applications on cloud server. Android is used to designed for physical Smartphone and also useful to construct a server platform. Android is open-source product and runs on an x86 CPU. The reason to use Android as a server platform is that it is able to run for smart phones and also for the x86 platform including servers. We show three types of multitenant architecture for an Android server platform and finds the way to take it to reality.

**Keywords:** Android, Cloud servers, Cloud computing, Multi-tenancy

Corresponding Author: MS. MUKTA DUTONDE



PAPER-QR CODE

Access Online On:

[www.ijpret.com](http://www.ijpret.com)

How to Cite This Article:

Mukta Dutonde, IJPRET, 2015; Volume 3 (9): 700-705

## INTRODUCTION

Nowadays the number of Smartphone users and mobile application are evolving rapidly. According to a recent report, 45 million people in the U.S. own Smartphone's and 234 million people subscribe to the mobile phone application stores [1]. Symbian, iOS, Android, and Windows Mobile are the types of mobile

Operating Systems (OSs). Users can easily enjoy the individual Smartphone life style, because thousands of application developers construct many kinds of applications for these platforms. Though Smart phones are expected to PC-like functionality, hardware resources such as CPUs, memory, and batteries are limited. Therefore, many application developers have to consider these limitations. Some researchers have proposed using server resources in the cloud for Smartphone's, to solve the resource problem. From this background, we proposed Development of an efficient platform to use Android as a Server that enables many users to use resources on remote cloud servers. The proposed system discuss analysis of process to adopt Android as a server OS as follow. Using a mobile OS enables the use of many mobile applications that is designed to be used on Smartphone interfaces, such as software key boards, touch panels and many sensors. It is better to use a remote application via a network than a desktop OS, because a resolution of mobile OS is small. Android is an open-source mobile OS initiated by Google. The main reason to use Android as a server platform is that it is able to run not only for Smart phone's but also for the x86 platform including servers. The system proposes a multi-tenant architecture of Android as a Server Platform.

## II. LITERATURE REVIEW

Integration of Mobile device and Cloud: Researches have proposed integration between mobile devices and cloud computing. Satyanarayanan et al. [2] mentioned their vision of allowing mobile users to seamlessly use nearby computers to obtain cloud-computing resources by instantiating a "cloudlet" that rapidly synthesizes virtual machines on a nearby infrastructure that can be accessed through a Wireless LAN. Canepa et al. [3] presented a frame work named "Ad Hoc cloud providers". In this structure, other device resources are used around them, mobile devices can execute their jobs as if it is executed on one cloud server. Our method is closely related to that of Chun and Maniatis [4]. As if they were running on mobile devices they proposed the creation of clone VMs to run mobile applications.

Multi-tenancy: Royon et al. Proposed multi-user, multiservice execution environment named "virtual service gateway" [5]. The existing multi-application environment approaches are classified by modifying Java runtime, and an overlay approach to run virtually original

application. Bezemer discussed the direction of multi-tenancy [6]. The concept of providing software service is generally known as SaaS (i.e., Software as a Service). However, the adaptation of such a model that requires the applications which are necessitates to be provided as a service should be generalized for individual users or groups of users.

Virtual Smartphone over IP :Beyond constructing a mobile application platform, the system has previously proposed a proof of concept prototype implementation named“ Virtual Smartphone over IP” [7].An overview of the Implementation is shown in Figure 1.In this prototype, Android-X86 is implemented on a mobile server OS running on a hypervisor. The client program installed on a physical Smartphone can remotely interact and control Android-x86images. The client program transmits various events from the physical device not only the key board but also the touch screen and various sensors such as GPSs, accelerometer, and thermometers, to the mobile server OS and receives graphical screen updates from it via Virtual Network Computing(VNC). These programs enable to use Server side virtual mobile OS applications as if it is running on a physical Smartphone.

The performance estimation using a common Smartphone and A server shows that our virtual Smartphone on a server is at least 10 times faster than on a physical Smartphone.

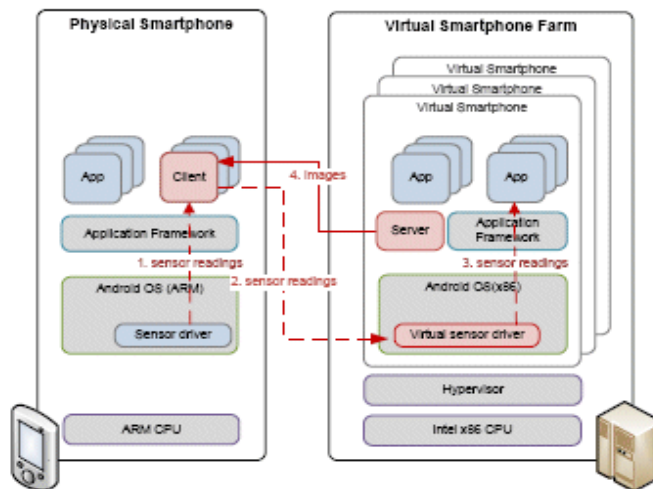
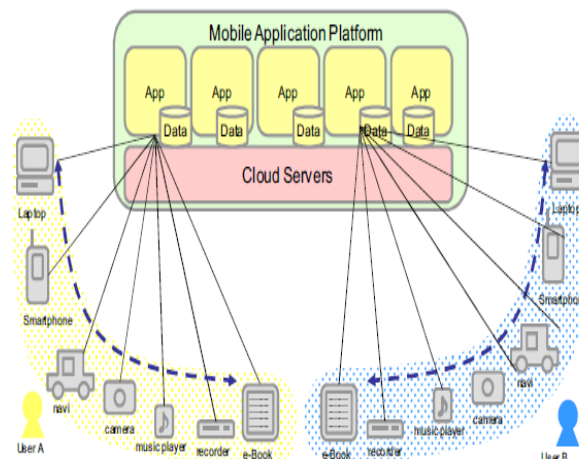


Figure1: Overview of Virtual Smartphone over IP system

### I. ANALYSIS OF PROBLEM

Mobile Application Platform on Cloud Server: The architecture for remotely using mobile application on server has many benefits for users, as a numbers of service providers such as

Dropbox[8] and ZumoDrive[9] provide online storage services. The approach, of Mobile Application Platform on Cloud Server, intends to handle user data and user applications in a cloud server. "Figure 2 illustrates an overview of the Mobile Application Platform. By executing a mobile application in the cloud server, users and developers free from device limitation such as CPU power, memory, and battery, and from device software environment such as OS or version. Moreover, once a user installs an application on the cloud server, they can use the application anywhere, on any device.



**Figure 2. Mobile Application Platform.**

Multitenant for Android: Multitenancy, Which means that software running on a server provides Services to many users, is one of important features for cloud computing. From the viewpoint of both economy and ecology, it is beneficial to share Hardware resources among users. Using a mobile OS would be more effective than using a desktop OS because the resource requirements of mobile OSs are smaller. The proposed system discusses the multitenant architecture for Android and how to construct it.

Multi-tenant architecture for Android: This section discusses the process to construct multitenant architecture for Android. The proposed System discusses the three types of approach, hypervisor layer, kernel-layer, and framework layer, for multitenant architecture.

1. Hypervisor: The hypervisor-layer uses the Virtual Smartphone over IP system. Each user own her/his Android OS image on a server and freely runs her/his application in a separate VM.

2. Kernel-layer: This approach changes android OS to run multiple user application in separate processes. This approach is similar to an ordinary thin client server running multiple user application in a server.

3. Framework-layer: Another approach is to create a multi-tenant function at framework-layer, similar to existing a Java-based multi-tenant framework. This approach remodels Android the framework and APIs to support multiple user application. The main challenge is how to run exiting Android application in modified framework.

The quantitative evaluation of these three types of architecture as, shown in Table 1.

## CONCLUSION

The proposed system Development of Android cloud for efficient implementation of platform as a service, system that enables the use of sharing server side Android OS among multiple users. The system also showed the technical difficulty and approach related to multi-tenant architecture for Android OS, which is originally designed to use single user. The proposed Android architecture is planning to develop a prototype system about multi-tenant. The system believes that proposed architecture shows high performance on virtual image-based virtualization for mobile application.

## REFERENCES

1. Android Phones Steal Market Share. [http://www.informationweek.com/news/mobility/smart\\_phones/showArticle.jhtml?articleID=224201881](http://www.informationweek.com/news/mobility/smart_phones/showArticle.jhtml?articleID=224201881).
2. M. Satyanarayanan, V. Bahl , R. Caceres, and N. Devise, "The case for VM-based Cloudlets in Mobile Computing" , IEEE persive Computing, Carnegie Mellon University, Microsoft Research, AT&T Research, Lancaster University , 2009.
3. G. H-Canepa and D.Lee, "A Virtual Cloud Computing Provider for Mobile Devices" . San Francisco: MCS'10, 2010.
4. B. G. Chun and P. Maniatias. "Augmented Smartphone Application through Clone Cloud Execution", ACM, Intel, Berkeley, Princeton, 2011.
5. Y. Royon, S. Frenot, and F. L. Mouel, "Virtualization of Service Gateways in Multi-provider Environments" , Heidelberg : CBSE 2006, 2006.

6. C. P. Bezrmer and A. Zaidman, "Multi-tenant SaaS Application: Maintenance Dream or Nightmare? Antwerp", Belgium: IWPSE-EVOL' 10, 2010.
7. E. Y. Chen and M. Ito. , "Virtual Smartphone over IP. Montreal", QC, Canada: IEEE WOWMOM, 2010.
8. Dropbox – Home – Online backup, file sync and sharing made easy.  
<http://www.dropbox.com/>.
9. Zumodrive- Enjoy your media and documents from every device.  
<http://www.zumodrive.com/>.