Performance based Analysis of Cloudlet Architectures in Mobile Cloud Computing

A.S. Jaiswal
Department of Computer Science and Engineering
S.G.B. Amravati University, Amravati (M.S) ,India

V.M. Thakare, Ph.D.
Department of Computer Science and Engineering
S.G.B. Amravati University, Amravati (M.S) ,India

S.S. Sherekar, Ph.D.
Department of Computer Science and Engineering
S.G.B. Amravati University, Amravati (M.S) ,India

ABSTRACT
With modern smart phones and powerful mobile devices, Mobile apps provide many advantages to the community but it has also grown the demand for online availability and accessibility. Cloud computing is provided to be widely adopted for several applications in mobile devices. However, there are many advantages and disadvantages of using mobile applications and cloud computing. The main disadvantage of cloud servers are the distance from the device, to overcome this disadvantage cloudlets are come in the picture. Cloudlets bring the cloud server closer to mobile devices which help to utilize many drawbacks of cloud server for the mobile industry. There are many different cloudlet architecture are developed by different researchers in the research work. In this paper the analytic and performance based review on different cloudlet architecture is done, which helps to understand the cloudlet and find out the best cloudlet architecture from them.

General Terms
Cloud computing, Mobile cloud computing architectures, cloudlet.

Keywords
Cloud, Cloudlet, virtualization, cloudlet architecture.

1. INTRODUCTION
With increasing number of mobile operating systems for Smartphone's, application market has grown bigger. Number different apps available for different mobile operating systems, but a user can't use them all on one Smartphone. Users are vendor-locked today. If a user wants to run a Windows app on an Android phone, it is not possible. Users are currently forced to live with older apps because they have older OS, which is no longer being supported, possibly due to limited hardware resources on their existing phones. With increasing usage of Smartphone's, they are becoming more of personal devices and this has made security of these devices as one of the prime concern, increasing the need for better protection against viruses and theft/data loss.

Mobile computing is at a fork in the road. After two decades of sustained effort by many researchers, researchers have developed the techniques, mechanisms and core concepts to provide a solid foundation for this still fast-growing area. The vision of "information at my fingertips at any time and place" was only a dream in the mid-1990s. Today, ubiquitous email and Web access is a reality that is experienced by millions of users worldwide through their BlackBerries, iPhones, Windows Mobile, and other mobile devices. Continuing on this road, mobile web-based services and location-aware advertising opportunities have begun to appear. Large investments are being made in anticipation of major profits. Yet, by staying on this path, mobile computing ignores its true potential. Awaiting discovery is an entirely new world in which mobile computing seamlessly augments the cognitive abilities of users using compute-intensive capabilities such as computer vision, graphics, speech recognition and natural language processing, machine learning, augmented reality, planning and decision-making. By thus empowering mobile users, we could transform many areas of human activity. The sidebar speculates on one example of such a transformation.

Cloudlet is a new architectural element that arises from the convergence of mobile computing and cloud computing. It represents the middle tier of a 3-tier hierarchy: i) mobile device ii) cloudlet iii) cloud. Cloudlet can be viewed as a "data center in a box" whose goal is to "bring the cloud closer". Cloudlet has four key attributes:

a) **Only soft state**: It means cloudlet has only the soft state it does not have any hard state. The main task of the cloudlet is to buffer the input data from the mobile device and bring it to cloud server and wise versa.

b) **Powerful, well-connected and safe**: Cloudlet are powerful systems because it has own powerful processors and RAM to work with. Cloudlets are connected with wired internet with the server hence the connectivity with the server is good and secure.

c) **Close at hand**: In logical term cloudlet are mediator between the mobile phones and the cloud server. It brings the cloud server more close to mobile phones, hence in logical terms it is close at hand.

d) **Builds on standard cloud technology**: It encapsulates offload code from mobile devices in virtual machines (VMs), and thus resembles classic cloud infrastructure such as Amazon EC2 and OpenStack. In addition, each cloudlet has functionality that is specific to its cloudlet role.

So in this paper the review is done on different architectures used for the cloudlet mobile cloud technology.

2. RELATED WORK
In [1], the authors present virtual Smartphone images running on the cloud and accessing them on smart phone over internet using VNC server running in each Android image and VNC client in the Smartphone. In [2] and [3], two instances of Android run in parallel on one device. In [4], authors create Clone Cloud system which is capable of offloading application execution from device to its clone on the cloud, and once the process is completed, new application state in returned to the phone.

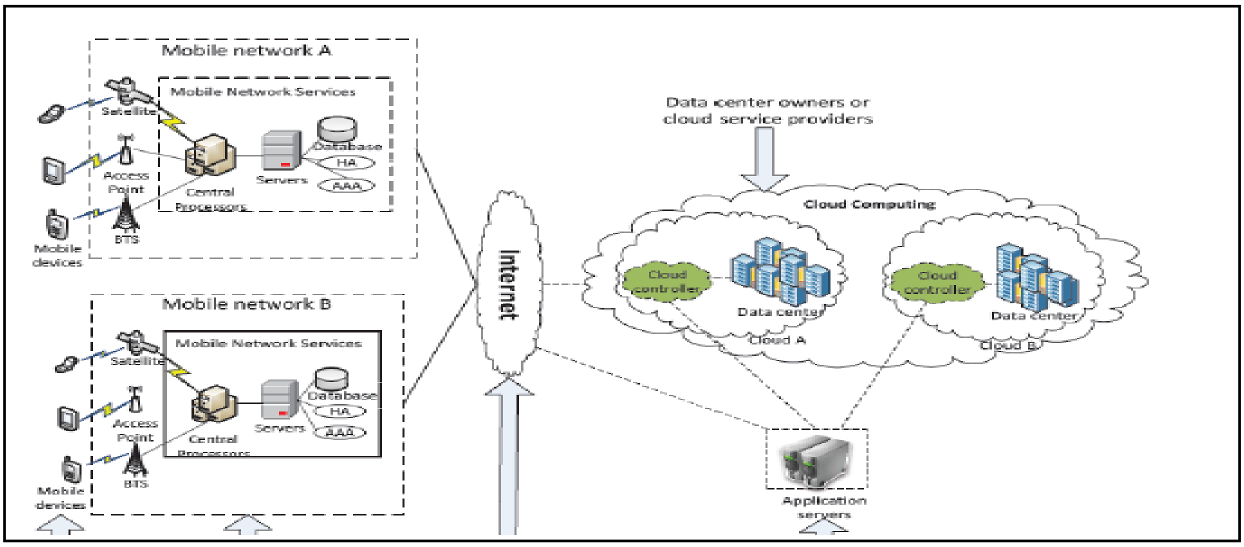


Figure 1 Architecture of Mobile Cloud Computing

In the above architecture, all mobile devices are connected to mobile networks via Access point, satellite and BTS (Base Transceiver Station) that establish and control the connector. After that mobile users request for information, then their ID and location are transmitted to the central processors which are connected to various servers and they provide mobile network services. Then after that mobile network operators provide several services such as AAA i.e. Authentication, Authorization, and Accounting to mobile users. They are based on subscribers data stored and HA (Home Agent) in databases. After that, the subscribers request is delivered to a cloud via internet. And then in the cloud, cloud controller is there to process the request of mobile user of the corresponding cloud services. The details of cloud architecture could be different in different contexts. The above description is according to W.Tsai et al. [5].

3. CLOUDLET ARCHITECTURES

3.1 Mobile Cloud Hybrid Architecture

Recently, a mobile-cloudlet-cloud architecture called MOCHA (Mobile Cloud Hybrid Architecture) has been developed at the University of Rochester as a framework for running computationally intensive mobile applications with low response time requirements [6], [7]. The introduction of a cloudlet, which serves as an edge-server, provides the potential to address the hurdles for mobile-cloud computing mentioned above.

Instead of transmitting data directly from a mobile device to the cloud servers, using MOCHA and send the data to cloudlet/cloudlets first. Cloudlet is capable of storing and updating a profile of the network latencies and their variation to reach different cloud servers. Using this approach, it can perform smart task division to select the best server(s) such that the overall communication latency is minimized.

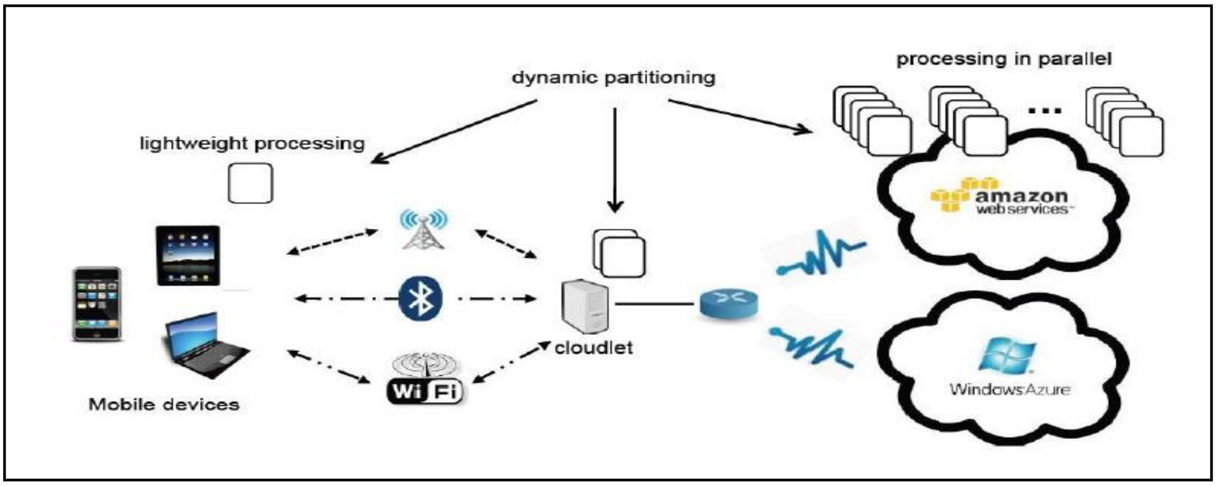


Figure 2: The MOCHA architecture: mobile devices interact with the cloudlet.

MOCHA (Mobile Cloud Hybrid Architecture) was created as a solution to support massively-parallelizable mobile-cloud applications. MOCHA, which is shown in Figure 2, consists of three parts: Mobile, Cloudlet and Cloud. It formulates a solution to allow mobile-cloud computing applications, such as object recognition in a battlefield, by introducing an edge-server, which is called a cloudlet [8].

Cloudlet is used as an intermediary between the mobile devices and the cloud servers and determines how to partition the computation, in terms of tasks, among itself and multiple cloud servers to optimize the overall quality of service (QoS), based on continuously updated statistics of the QoS metrics (e.g., latency).

Running many resource-intensive applications far exceeds the capabilities of today's mobile devices, such as conducting real-time face recognition of known criminals. There are many limitations of the mobile device, such as low processor speed, small memory size, and limited storage capacity. Facing these challenges, cloud servers with tremendous computing power and storage space as well as Figure 2: The MOCHA architecture: mobile devices interact with the cloudlet and the cloud via multiple connections and use dynamic partitioning to achieve their QoS goals (e.g., latency, cost) (reprinted from (Soyata, T., Muraleedharan, R., Funai, C., Kwon, M., & Heinzelman, W., 2012) with permission of the au- thors).

Access to particular databases is suitable choices to accelerate applications run on mobile devices. Nevertheless, several major hurdles limit these benefits, such as long network latency, which hurts the user experience in mobile-cloud computing. As a result, powerful, well-connected and safe cloudlets are necessary to intercept the data sent from the mobile and perform smart task division algorithms to minimize the overall communication latency to and from the cloud.

There are many applications that can benefit from utilizing an edge server (cloudlet in MOCHA), such as battlefield support applications, natural language processing, airport security, an enhanced Amber Alert system, among others.

3.2 VM-based Cloudlets in Mobile

In this architecture, a mobile user exploits virtual machine (VM) technology to rapidly instantiate customized service software on a nearby cloudlet, and then uses that service over a wireless LAN. The mobile device typically functions as a thin client with respect to the service. A cloudlet is a resource-rich, trusted computer or cluster of computers that is well-connected to the Internet and is available for use by nearby mobile devices.

Strategy of leveraging transiently-customized proximate infrastructure as a mobile device moves with its user through the physical world is called cloudlet-based resource-rich mobile computing. Crisp interactive response, which is essential for seamless augmentation of human cognition, is easily achieved in this architecture because of the physical proximity and one-hop network latency of the cloudlet. Using a cloudlet also simplifies meeting the peak bandwidth demand of multiple users's interactively generating and receiving media such as HD video and high-resolution images. Rapid customization of infrastructure for diverse applications emerges as a critical requirement of this architecture.

Imagine a future in which cloudlet infrastructure is deployed much like Wi-Fi access points today. Indeed, it would be relatively straightforward to integrate cloudlet and Wi-Fi access point hardware into a single easily deployable entity. A key challenge is to simplify cloudlet management. Widespread deployment of cloudlet infrastructure will not happen unless software management of that infrastructure is trivial — ideally, it should be totally self-managing. Tightly restricting software on cloudlets to simplify management is unattractive because it constrains application innovation and evolution. Instead, an ideal cloudlet would support the widest possible range of mobile users, with minimal constraints on their software .The solution is transient customization of cloudlet infrastructure using hardware virtual machine (VM) technology. The emphasis on "transient" is important: pre-use customization and post-use cleanup ensures that cloudlet infrastructure is restored to its pristine software state after each use, without manual intervention. A VM cleanly encapsulates and separates the transient guest software environment from the permanent host software environment of the cloudlet infrastructure. The interface between the host and guest environments is narrow, stable, and ubiquitous. This ensures the longevity of cloudlet investments and greatly increases the chances of a mobile user finding compatible cloudlets anywhere in the world. The malleable software interfaces of resource-rich mobile applications are encapsulated within the guest environment and are hence precisely re-created during pre-use customization of cloudlets. As a result, a VM-based approach is less brittle than alternatives such as process migration or software virtualization [9].

There are two different approaches to delivering VM state to infrastructure. One is a VM migration approach in which an already-executing VM is first suspended; its processor, disk and memory state are then transferred; finally, VM execution is resumed at the destination from the exact point of suspension. The basic feasibility of this approach has been confirmed by work on the Internet Suspend/Resume (ISR) system [10] [11] and SoulPad [12], and by other work such as the Collective [13] and Xen live migration [14].

Figure 3 shows the architecture of VM-based cloudlets. A small VM overlay is delivered by a mobile device to cloudlet infrastructure that already possesses the base VM from which this overlay was derived. The infrastructure applies the overlay to the base to derive the launch VM, which starts execution in the precise state from which the overlay was derived. In a language translation application, for example, the software in the launch VM could be a server that receives captured speech from a mobile device, performs speech recognition and language translation, and returns the output for speech synthesis. If the cloudlet is a cluster, the launch VM could be rapidly cloned to exploit parallelism, as described by Lagar-Cavilla et al [15].

Cloud and Ubiquitous are emergent buzzwords for Computing and Information Systems, respectively. The increasing interest and tentative to adopt them, make them relevant factors in nowadays continuous business models changes. Literature announces (virtually) unlimited capacity with Cloud Computing (CC) (Charlton, 2008), and Ubiquitous Information System (UIS) grants permanent services availability. This architecture defend that both properties (together) represent the perfect setting for future Manufacturing support and sustainability (Cloud and Ubiquitous Manufacturing Systems – CUMS).

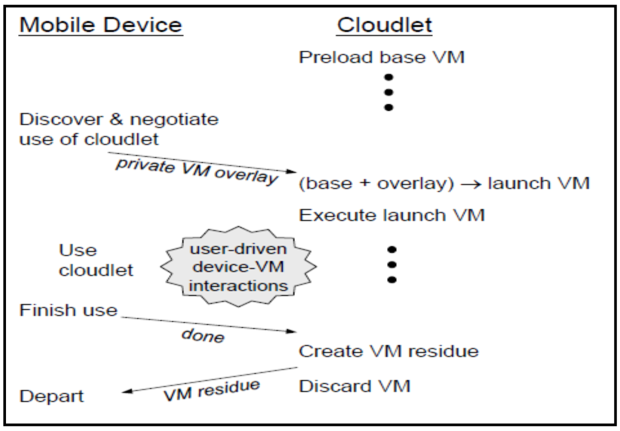


Figure 3 Architecture of VM-based Cloudlets

3.3 Cloudlet Architecture for Dashboard in Cloud

Dashboard cloudlet architecture is an integrated architecture which sustains the management and coordination of cloud-based services (resources) to grant technological integration requirements, as well as, communicational instruments as pragmatics tools to support human-to-human interaction, granting effective user participation. This base architecture follows Model- View-Control (MVC) pattern and the interface follows the RIA Presentation Design Pattern (Cunningham, 2003), having resources and their governance services (M) hosted in cloud, cloud-based Representational State Transfer (REST) services to support business rules and actions (C) and multimodal Rich Internet Application (RIA) Presentation Layer (V) to allow multimodal device interaction with each other. The global Market of Resources will be supported by cloud-based mechanisms (brokering) inherent to SOA Governance. The services, as instances of manufacturing resources (machines, persons, enterprises, etc.), are autonomously maintained following asynchronous subscribing pattern, classified using SLA (Services Layer Agreement) and geo-referenced with spatial data.

As a workflow overview, a Process Plan determines operations and resource specification (like a resource stereotype or meta-resource) to handle them; broker finds candidates resources able to support it, mapping them to resources on the ground. The mapping process is not necessarily automatic, but assisted with user participation, if needed, that is, using Pragmatics [16]. Each resource represents a service (or many) that is hosted on cloud. It has an interface description language (IDL) that allows its discovery, an interoperability specification to follow and an (REST) API that allows its integration with (or, use by) other resources. So, each resource has its own "information system" to handle its work. "Residing" in cloud they are named cloudlets.

The application front-end has a RIA Presentation Layer behaving as a dashboard that, besides integrating common RIA web components, allows the management of each integrated cloudlet and global monitoring of associated resource (service).

RIA will be supported by emergent web 3.0 technologies (HTML5, JQuery, CSS3, etc.) and pragmatics instruments, communicational channels mainly, due to multimodality requirement, will be supported by open source communication technology as Web RTC and Web Media Capture. So, each cloudlet is enhanced with layers representing enhanced services. Considering the spatial data representation on google maps, for instance, the map services (Figure 4 (a)): zoom, 3D, etc. (supported by Google) will be enriched with advanced information windows (Figure 4 (b)) where (existent) communicational channel links will be enable (email, SMS, chat, RT video, audio recording, etc.) and thus, in dashboard, a direct synchronous conversation with resource's owner will be possible if required (Figure 4 (c)).

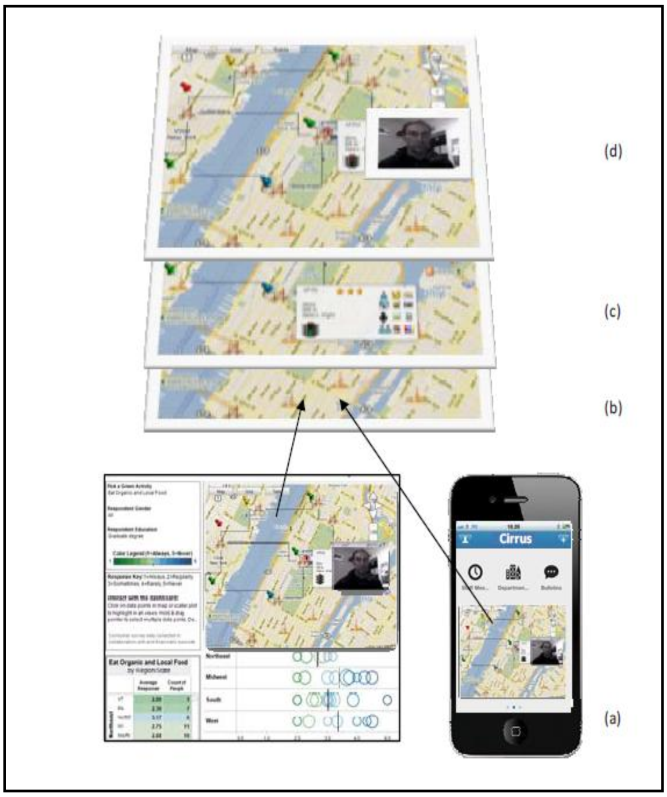
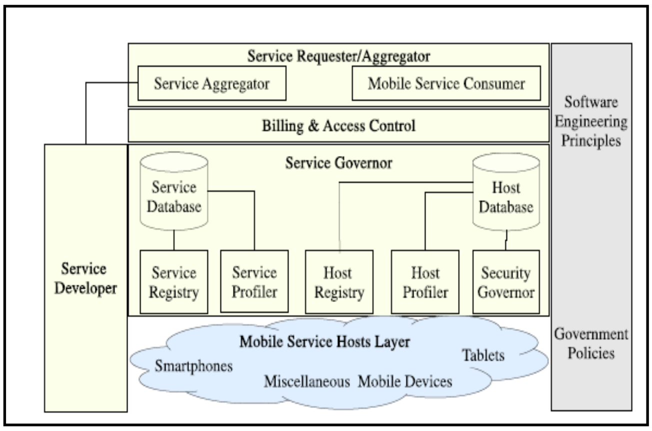


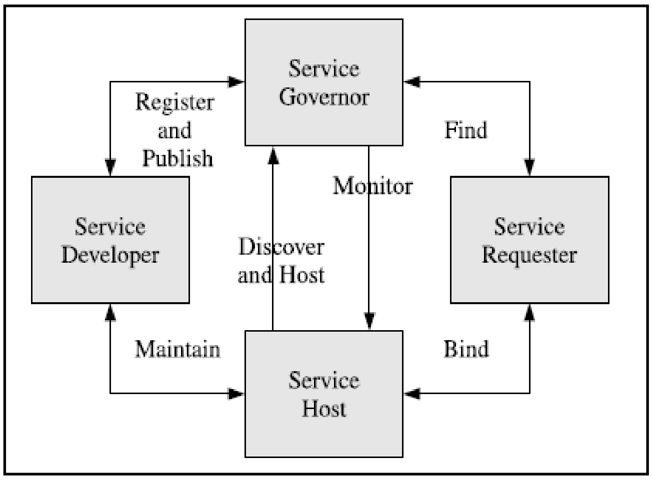
Figure 4 Cloudlet Architecture (a) Dashboards (b) Cloudlet (service) (c) Enhanced cloudlet (d) Cloudlet with pragmatics instruments

3.4 Market-Oriented Mobile Cloud Architecture

Figure 5 (a) depicts MOMCC layered architectural model consists of four entities, namely Service developer, governor, host, and requester/aggregator. The functional relationships between these four building blocks are depicted in Figure 5 (b). Initially, the Service developer registers and publishes its Service(s) to the Service governor which plays the role of UDDI for Web Services. Service host communicates with the Service governor to browse available Services and request for hosting. In runtime, the requester or aggregator (program developer who aggregates services to quickly build new composite applications) will query the required Service against the Service governor to identify the nearby Service host. Once found, the requester can directly invoke and bind the Service. Service developer has also direct link with the Service host to maintain and update the Service if required. Service governor monitors the performance and reliability of the Service host for future decisions. In this architecture there is no direct communication between service requester and developer. Anonymity of Service requester will likely discourage attackers and likely protect privacy of hosts against potential malicious Service developer.



(a) Layered Architecture of MOMCC



(b) The Block Diagram of MOMCC

Figure 5. The Architecture of Market-Oriented Mobile Cloud Computing (MOMCC)

3.5 SAMI: Service-based Arbitrated Multi-Tier Infrastructure

The SAMI architecture is used to tackle aforementioned issues that are mainly created due to heterogeneity and trust which is depicted in Figure 6. SAMI architecture leverage SOA since it is a service driven approach to generate platform-neutral applications. In SAMI architecture consists of three major layers, namely SOA, arbitrator, and infrastructure. The main strength of this architecture is in its multi-tier infrastructure layer which leverages infrastructures from three main sources of Clouds, MNOs, and MNOs' authorized dealers. On top of the infrastructure layer, arbitrator layer is responsible to classify Services and allocate the suitable resources to them based on several metrics such as resource requirement, latency, and security.

3.5.1 SOA Layer

Service oriented layer is responsible to perform Service related tasks such as Service registry, discovery, and composition.

3.5.2 Arbitrator Layer

In this layer MNOs act as arbitrators between front-end (Cloud-mobile users) and back-end (Cloud service providers). It receives resource allocation requests from SOA layer and monitors infrastructure layer to determine where the Service should be allocated for more efficient execution. One decided, a stand-alone copy of the Service will be transferred to the infrastructure for future reference. In this way, the network overhead is noticeably reduced.

3.5.3 Infrastructure Layer

In order to enhance quality of computing, decrease communication latency, and improve energy efficiency we propose a multi-tier infrastructure layer including three major layers, namely clouds, MNOs, and MNO's authorized dealers.

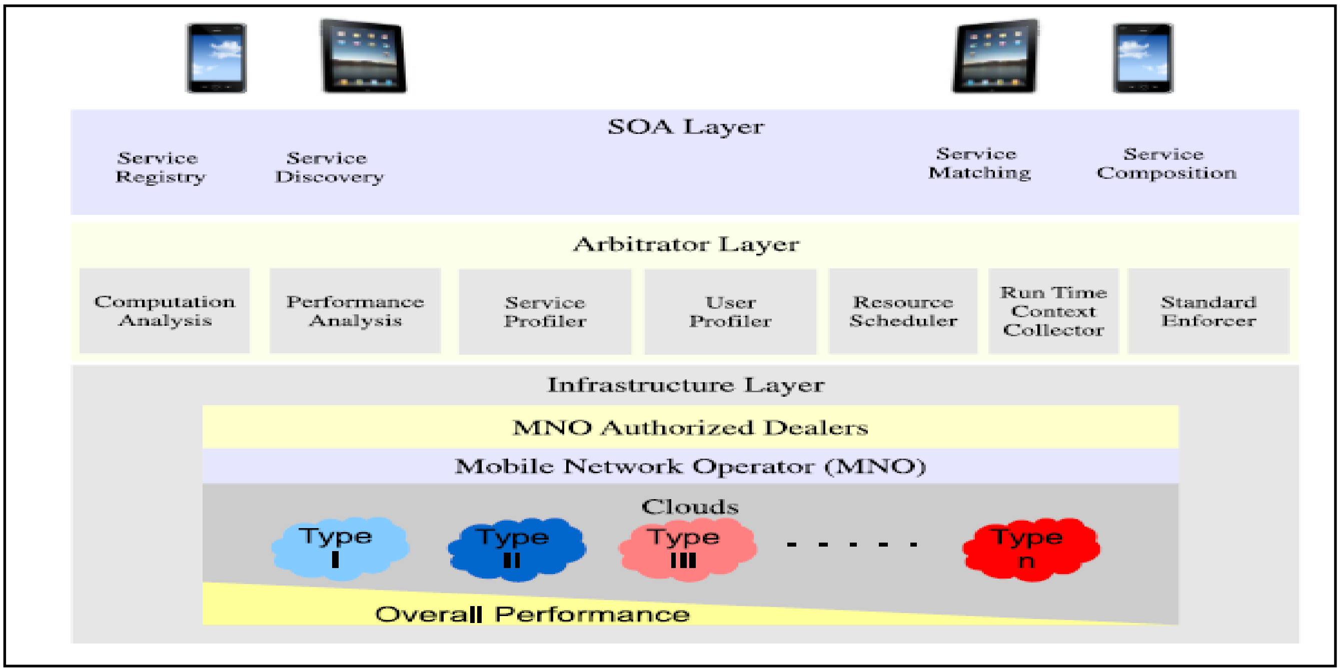


Figure 6 SAMI: The Service-Based Arbitrated Multi-Tier Infrastructure for Mobile Cloud Computing.

4. PERFORMANCE ANALYSIS AND DISCUSSIONS

The different cloudlet architectures are analyzed for the performance point of view. Mobile cloud hybrid architecture increases resource availability by leveraging large number of nearby mobile devices in public places like shopping mall, cinema, and airport the Service availability is increasing noticeably. It also Enhances Security because of the dynamic partitioning of the communication channel in the between mobile device and cloud server. Mobile cloud hybrid architecture also Reduce the Latency because of the parallel processing. But the main drawbacks of this architecture are speed, its complex structure and this will not be used in offline mode. VM-based Cloudlet architecture removes the most of the drawbacks of the MOCHA it has speed, simple structure, reduced latency and improved QoS for the end user. VM-based architecture also has its own drawbacks like Application virtualization, is not suitable for every type of environment, it uses more resources, and this also not work in offline mode. MOMCC architecture use the very few resources form the mobile device to use its services. MOMCC architecture is also very simple, enhanced security improves reliability and required no skills for hosting. One of the drawbacks of MOMCC architecture is that complexity of Services is highly dependent on computing capabilities of hosting devices which are not very high. Due to limitation of host in MOMCC, Services are often fine in granularity which causes extra execution overhead on mobile hosts. Such overhead prolongs execution time and increases communication traffic. MOMCC architecture is highly dependent on nearby computing devices and will be affected in the remote environment such as mountain or jungles where less mobile devices exist and this also not work in offline.

SAMI architecture utilizes the concept of SOA which is a widely-accepted solution to overcome application portability in MCC. In SAMI architecture mobile users are able to utilize the closest available resources to bring down the communication latency. Hence, the application execution time is reduced and energy efficiency is increased. The arbitrator layer in SAMI takes this headache away from mobile users and ensures them the optimal resource. The main drawback of SAMI is the MNO Overhead and very complex management system.

From the above discussion the concept of cloudlet is analyzed briefly. All the different architectures used for the mobile cloud i.e. cloudlet is studied and discussed. QoS (Quality of Service), load balancing, delay and throughput are the important parameter of the cloudlet which is taken in consideration for improving the performance of the cloudlet. QoS played a very important role in any cloud related application. So the QoS should be always high for the better performance of the cloudlet. Delay should always be low and throughput should always be high. From the above discussion and according to researchers study performance of VM based cloudlet is best from the other cloudlet architectures. All the parameter gives the better results than the other cloudlet architecture because of better load balancing techniques. VM based architecture mainly focused on the QoS which improve the performance of the cloudlet.

5. CONCLUSION

This paper has attempted to review a significant number of papers to cover the recent development in the field of cloudlets. Present study reveals that various architectures have been used for mobile cloud computing. This paper give the analysis of the five main commonly used cloudlets architecture and they are Mobile cloud Hybrid Architecture, VM-based cloudlets in mobile computing, Cloudlet architecture for Dashboard in cloud, Market-Oriented Mobile Cloud Architecture and Service-Based Arbitrated Multi-Tier Infrastructure. In this paper the performance based comparative study of different cloudlet architecture is done on the bases of the advantages and drawbacks in the analysis section

6. FUTURE WORK

In the paper the analytic review on the different cloudlet architecture is done. In this paper advantages and disadvantages of different cloudlet architecture is discussed. From the discussion it concludes that the performance of VM based cloudlet is comparatively on higher side from the other cloudlet architectures, but it has some disadvantages like energy consumption and application virtualization. So in the future work, more detail component level analysis is required to carry out for detail comparison.

7. REFERENCES

[1] Chen, Eric Y., and Mistutaka Itoh. "Virtual smartphone over IP." World of Wireless Mobile and Multimedia Networks (WoWMoM), 2010 IEEE International Symposium on a. IEEE, 2010.

[2] Lange, Matthias, et al. "L4Android: a generic operating system framework for secure smartphones." Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices.2010.

[3] Lange, Matthias, et al. "Demo: L4Android Security Framework on the Samsung Galaxy S2."

[4] Chun, Byung-Gon, et al. "Clonecloud: elastic execution between mobile device and cloud." Proceedings of the sixth conference on Computer systems. ACM, 2011

[5] W. Tsai, X. Sun, and J. Balasooriya, "Service-Oriented Cloud Computing Architecture," in Proceedings of the 7th International Conference on Information Technology: New Generations (ITNG), pp. 684-689, July 2010.

[6] MOCHA, \Mocha research project." http://themochaproject.com/index.html.

[7] T. Soyata, R. Muraleedharan, C. Funai, M. Kwon, and W. Heinzelman,"Cloud-Vision: Real-time Face Recognition Using a Mobile-Cloudlet-Cloud Acceleration Architecture," Proc. of IEEE ISCC, Jul. 2012.

[8] T. Soyata, R. Muraleedharan, J. Langdon, C. Funai, S. Ames, M. Kwon, and W. Heinzelman, "COMBAT: mobile-Cloud-based cOmpute=communications infrastructure for BATtle_eld applications," Proc. of SPIE, vol. 8403, 84030K-1, 2012.

[9] Osman, S., Subhravati, D., Su G., Nieh, J. The Design and Implementation of Zap: A System for Migrating Computing Environments. In Proc. of the 5th Symposium on Operating Systems Design and Impl, Boston, MA, Dec. 2002.

[10] Kozuch, M., Satyanarayanan, M., "Internet Suspend Resume". In Proc. of the Fourth IEEE Workshop on Mobile Computing Systems and Applications, Calicoon, NY, June 2002.

[11] Satyanarayanan, M., Gilbert, B., Toups, M., Tolia, N., Surie, A., O'hallaron, D. R., Wolbach, A., Harkes, J., Perrig, A., Farber, D. J., Kozuch, M. A., Helfrich, C. J., Nath, P., And Lagarcavilla, H. A. Pervasive Personal Computing in an Internet Suspend/Resume System. IEEE Internet Computing 11, 2 (2007).

[12] Caceres, R., Carter, C., Narayanaswami, C., And Raghunath, M. "Reincarnating PCs with Portable Soul-Pads.", In Proc. of the 3rd Intl. Conference on Mobile Systems, Applications, and Services Seattle,WA, June 2005.

[13] Sapuntzakis, C., Chandra, R., Pfaff, B., Chow, J., Lam, M., Rosenblum, M. "Optimizing the Migration of Virtual Computers." In Proc. of the 5th Symposium on Operating Systems Design and Impl. Boston, MA, Dec. 2002.

[14] Clark, C., Fraser, K., Hand, S., Hansen, J. G., Jul, E., Limpach, C., Pratt, I., And Warfield A. "Live Migration of Virtual Machines." In Proc. of the 2nd USENIX Symposium on Networked Systems Design and Impl. (NSDI) Boston, MA, May 2005.

[15] Lagar-Cavilla, H. A., Whitney, J., Scannell, A., Patchin, P., Rumble, S., De Lara, E., Brudno, M., And Satyanarayanan, M. "SnowFlock: Rapid Virtual Machine Cloning for Cloud Computing." In Proceedings of EuroSys 2009 Nuremberg, Germany, March 2009.

[16] Putnik, G. D., Ferreira, L., Cunha, M., Putnik, Z., Castro, H., & Shah, V. "User Pragmatics for Effective Ontologies Interoperability for Cloud-based Applications: The case of Ubiquitous Manufacturing System UI. Electronic Markets", The International Journal on Networked Business, 2012.

[17] G. P. Perrucci, F. H. P. Fitzek, and J. Widmer, "Survey on energy consumption entities on the smartphone platform," in *Proc. VTC Spring*, 2011, pp. 1–6.

[18] Luís Ferreira, Goran Putnik, "Cloudlet architecture for dashboard in cloud and ubiquitous Manufacturing" at 8th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 2013 366 – 371.

[19] Zohreh Sanaei and Saeid Abolfazli, "SAMI: Service-Based Arbitrated Multi-Tier Infrastructure for Mobile Cloud Computing at 27 June 2012.