International Journal of Scientific & Engineering Research, Volume 5, Issue 7, July-2014 865
ISSN 2229-5518

Cloud Computing for Mobile Applications

M.Swapna ,M.Tech, Assistant Professor
Computer Science and Engineering Department
Stanley College of Engineering Technology for Women-Hyderabad

Abstract—Cloud computing in mobile platforms has invoked a new wave of evolution in the rapidly developing mobile world. Although several striking research work has been conducted in the high computing counterparts of mobile technology, the field of cloud computing for mobile world is vastly unexplored. In this paper, i introduce the concept of Mobile Cloud Computing(MCC), it's inner workings and the various implementable architectures related to MCC.

1 INTRODUCTION

Cloud computing for mobile world or, rather, Mobile Cloud Computing (MCC) is a well accepted concept that aims at using cloud computing techniques for storage and processing of data on mobile devices, thereby reducing their limitations. According to ABI Research [7], "By 2015, more than 240 million business customers will be leveraging cloud computing services through mobile devices, driving revenues of $5.2 billion". While it must be noted that there were only 42.8 million Mobile Cloud Computing subscribers in 2008 [7]. This underlines the importance of cloud computing for mobile. The end mobile device user will eventually be the benefactor of the Mobile Cloud Computing. Company users can share resources and applications without a high level of capital expenditure on hardware and software resources. Nature of cloud applications also is advantageous for users since they do not need to have very technical hardware to run applications as these computing operations are run within the cloud.

This reduces the price of mobile computing to the end users. They could see a huge number of new features enhancing their phones due to Mobile Cloud Computing. At the same time the developers also have real advantages from Mobile Cloud Computing. The largest benefit of cloud computing for developers is access to a broader audience of a wide range of mobile subscribers. Since cloud computing applications go through a browser, the end user's mobile operating system does not have any impact on the application. Along with the plethora of benefits, there are a large number of issues to be addressed and unsolved problems to be solved. Several challenges such as the dependency on continuous network connections, data sharing applications and collaboration, and security Another key challenge for Mobile Cloud Computing is network availability and intermittency. Also Mobile Cloud Computing concepts rely on an always-on connectivity and will need to provide a scalable and highquality mobile access.

II. HOW MOBILE CLOUD COMPUTING WORKS

A. Architecture for Mobile Applications in Cloud Environment

We will look at a open source project for mobile cloud platform called openmobster [5]. Its architecture is as given in the Figure 1.

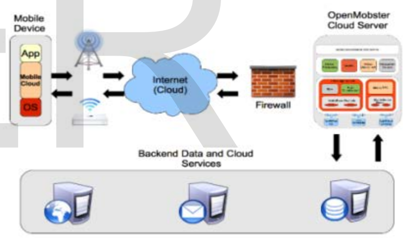


Fig. **1. [5] The Open mobster architecture for MCC**

B. Typical services needed by a mobile cloud client The most essential services include

Sync : This service synchronizes all state changes made to the mobile or its applications back with the Cloud Server.

Push : It manages any state updates being sent as a notifications from the cloud server. This improves the user's experience as it does not require the user to proactively check for new information.

OfflineApp : It is a service which carries the management capabilities to create smart coordination between lowlevel services like Sync and Push. It frees the programmer from the burden of writing code to actually perform synchronization as it

ISSN 2229-5518
is this service which decides synchronization management and mechanism which is best for the current state. The moment the data channel for any mobile application is established, all synchronizations and push notifications are automatically handled by OfflineApp service.

Network : It manages the communication channel needed to receive Push notifications from the server. It carries the ability to establish proper connections automatically. It is a very low-level service and it shields any lowlevel connection establishment, security protocol details by providing a high level interfacing framework.

Database : It manages the local data storage for the mobile applications. Depending on the platform it uses the corresponding storage facilities. It must support storage among the various mobile applications and must ensure thread safe concurrent access. Just like Network service it is also a low-level service.

InterApp Bus : This service provides low-level coordination/ communication between the suite of applications installed on the device.

Fig. 1. [5] The openmobster architecture for MCC

C TYPICAL SERVICES NEEDED BY A MOBILE CLOUD SERVER THESE ARE THE ESSENTIAL SERVICES THAT MUST BE PROVIDED TO THE MOBILE APPS BY THE SERVER.

Sync : Server Sync service synchronizes device side App state changes with the backend services where the data actually originates. It also must provides a plugin framework to mobilize the backend data.

Push : Server Push service monitors data channels (from backend) for updates. The moment updates are detected, corresponding notifications are sent back to the device. If the device is out of coverage or disconnected for some reason, it waits in a queue, and delivers the push the moment the device connects back to the network.

Secure Socket-Based Data Service : Depending on the security requirements of the Apps this server side service must provide plain socket server or a SSL-based socket server or both.

Security : Security component provides authentication and authorization services to make sure mobile devices connecting to the Cloud Server are in fact allowed to access the system. Every device must be first securely provisioned with the system before it can be used. After the device is registered, it is challenged for proper credentials when the device itself needs to be activated. Once the device is activated, all Cloud requests are properly authenticated/authorized going.

Management Console : Every instance of a Cloud Server must have a Command Line application such as the Fig. 2. [5] Mobile server cloud stack Management Console as it provides user and device provisioning functionalities. In the future, this same component will have more device management features like remote data wipe, remote locking, remote tracking, etc.

Figure 3 shows the mobile server cloud stack.

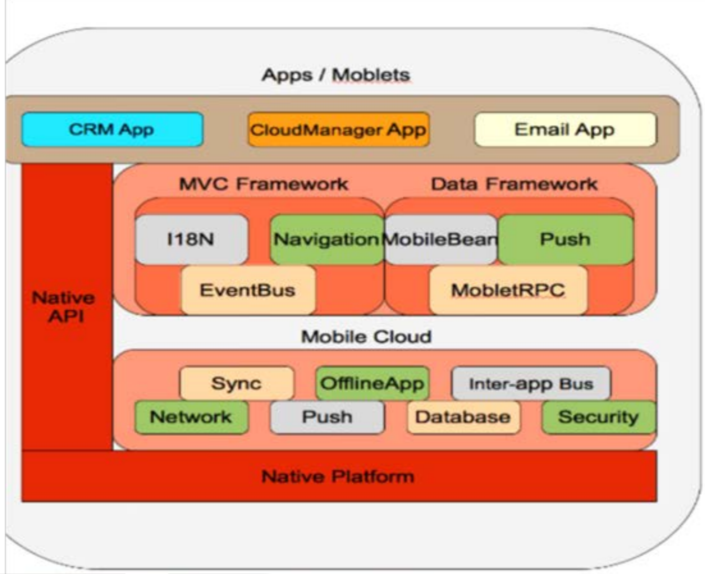


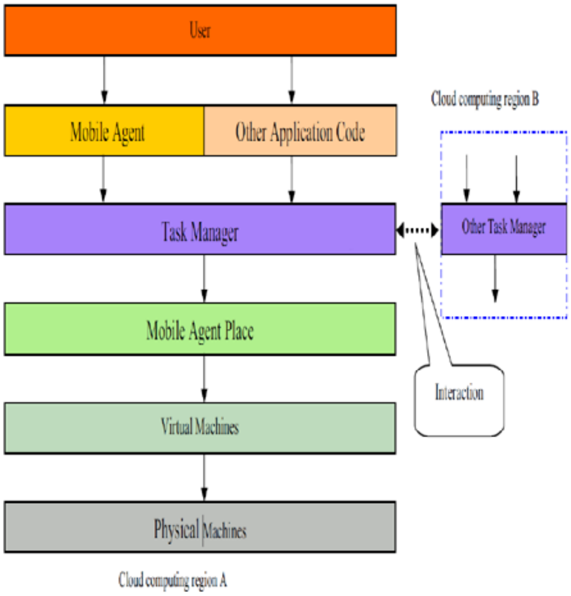
Fig. 2. [5] Mobile server cloud stack

III.CHALLENGES AND POSSIBLE SOLUTIONS OF MOBILE CLOUD COMPUTING

The following factors are essential to delivering a "good" cloud service:

Partitioning of application functions across cloud and device
Low network latency for faster responses
High network bandwidth for faster data transfer between cloud and devices
Adaptive monitoring of network conditions to optimize network and device costs

Fig. 3. [10] Architecture of Mobile Agent Based Open Cloud Computing Federation



There are several other issues related to implementation of Mobile Cloud Computing, but we shall be exploring only the following issues in this term paper.

A. Absence of Standards

Though cloud computing is considered to have many advantages including infinite scalability, lowered total cost of ownership (TCO), reduced investment and risk for the user and system automation, there is no open accepted standard available for Cloud computing. Portability and interoperability is also impossible between different Cloud Computing Service Providers, which handicaps the widely deploy and quick development of cloud computing. Customers are reluctant to transform their current datacenters and IT resources to the cloud computing platforms, because a number of unsolved technical problems still exist for these cloud platforms.
A possible solution proposed by many researchers from industry and academe [8], [6] called Open Cloud Computing Federation (abbreviated as OCCF later) may be an answer to many of these problems. The conception of Open Cloud Computing Federation is it incorporates multiple CCSP's (Cloud Computing Service Provider) service to provide a uniform resource interface for the user.

With reference to some viewpoints on cloud computing in [1], from University of California, Berkeley, following are the problems existing due to lack of open standards.

Limited scalability Most of the CCSP claim that they provide infinite scalability for the customer, actually, with the widely use of cloud computing and the rapid growth of the users, none of the CCSPs can meet all the requirements of all the users.

Unreliable availability of a service Actually, shutdown events happened recently with many of the of the CCSP's cloud computing service, including Amazon, Google and Microsoft. Dependence on a single CCSP's service can be bottleneck on the event of a breakdown since the application can't be migrate to another CCSP and hence the service will vanish from the network. On the contrary, in OCCF, user's application can migrate to another CCSP in case of breakdown, and multiple CCSP's service can be used simultaneously, offer a good availability of a service.

Service provider lock-in Absence of portability makes it impossible for data and application transfer among CCSPs; consequently, the customer is locked to a certain CCSP. An OCCF will democratize the whole cloud computing market where the small scale competitors can enter and thus promote innovation and vitality.

Unable to deployment service over multiple CCSPs Currently, application can't scale over multiple CCSPs since there is no interoperability between CCSPs. OCCF can make it possible for an organization to integrate different CCSP's service to provide more valuable service for the end user. Inspite of all these advantages there is no move for a common cloud standard basically because most of the cloud computing firms have their own private APIs and for setting them up lots of funds were spend. So to change to a new standard is resisted by most CCSPs. The OCCF, at present, lacks a practical realization mechanism as there is no cloud computing standard and portability, interoperability is impossible among CCSPs.

A possible approach is to have a Mobile Agent Based Open Cloud Computing Federation (MABOCCF) mechanism.

1) Architecture of MABOCCF: Figure 4 shows the architecture. Since compatibility is of high concern when one moves from one CCSP to another, this method makes use of Mobile agents for performing computation. First we shall define a Mobile Agent. A mobile agent is a software-data composition that can migrate its state from one environment to another, with its data intact, and still be capable of performing computations appropriately in the new environment. We shall see how this architecture can solve most of the the existing problems.

Portability The very definition of Mobile agent ensures portability. Each mobile agent runs on a place on the virtual machines (provided by the CCSPs) called Mobile Agent Place (MAP). Mobile agents carrying the application code or user's tasks can move from one MAP to another MAP independent of the CCSP thereby realizing portability among heterogeneous CCSPs

Interoperability Interoperability problem is now reduced to the negotiation and collaboration among agents which can be effected using agent interoperability standards.

2) Working Mechanism: A centralised approach based on Task manager is used in this architecture. Each administrative domain in CCSP has a virtual machine and a Mobile agent place installed. One of the virtual machines is chosen as the task manager which performs many services including resource

ISSN 2229-5518

indexing, authentication, security, billing, disaster recovery and fault tolerance.

Fig. 4. [10] Encapsulated datastructure sent to cloud

At the user end, task is encapsulated in the mobile agent, as a datastructure (Figure 5) and send to the cloud. Mobile agent place receives all the newly sent mobile agents. It informs the Task manager whenever a mobile agent is received and is responsible for the backup and monitor the mobile agent.The MAPs interact and interchange information with the Task Manager frequently.

B. Access Schemes

Mobile Cloud Computing will be deployed in a heterogeneous access scenario with a wide range of different radio access technologies such as GPRS, LTE, WLAN. Whichever be the access technology, Mobile Cloud Computing requires wireless connectivity with the following features.

MCC requires an 'always-on' connectivity for a low data rate cloud control signalling channel.
MCC requires an 'on-demand' available wireless connectivity with a scalable link bandwidth.
MCC requires a network selection and use that takes energy-efficiency and costs into account.

The most critical challenge of Mobile Cloud Computing is probably to guarantee a wireless connectivity that meets the requirements of Mobile Cloud Computing with respect to scalability, availability, energy- and cost-efficiency. Thus access management is a very critical aspect of Mobile Cloud Computing. A possible solution is to use context and location information [3] to optimize mobile access. Today, this is already used by a broad variety of applications, in particular contextaware services for mobile terminals. These services exploit data collected from terminal sensors (e.g., GPS, gyro, proximity detectors) or network sensors measuring network status and load. Not only consumer applications but also network services exploit this information. Deployment of Mobile Cloud Computing utilising the context information, such as device locations and capabilities and user profiles, can be used by the mobile cloud server to locally optimize the access management

C. Security

Most of the mobile devices (especially the smartphones) has almost all the functionalities of a standard desktop computer. This, unfortunately like the desktop machines, poses the same security threats to mobile devices. To combat the security threats, current mobile devices run the threat detection services on the mobile device itself. Such an exercise warrants intensive usage of resources both in terms of computation and power. A possible solution is to comes with a new model of security where detection services can be moved to cloud. It significantly saves the device CPU and memory requirements but at the cost of increasing bandwidth. Such an approach has several benefits:

Better detection of malicious software
Reduce on-device Resources consumption
Reduce on-device Software complexity

Such an approach has been proposed in [4]. They propose an architecture containing three components:

1) Host agent : It is a light weight process that runs on each device and inspect the file activity on the system. It has a cache where it stores the unique identifier (such as hash) for files received. Whenever a new file whose file identifier is not available in the cache, it will be send to the Network Service.

2) Network Service : This service analyses the files send to it by the host agent. There can be multiple instance of Network Service running on the cloud using virtualization; hence supports parallel detection of multiple files send by many Host agent.

3) Caching : There are two types of cache:

Local private cache is on the device where the host agent can put the identifier of inspected files.

Global shared cache resides on the Network Service which contains the identifiers of all inspected files received so far.

Apart from the anti-virus service provided, Mobile Cloud Computing platform must also address other mobile specific issues like:

SMS Spam filtering
Phishing Detection
Centralised Blacklisted

1) **Limitations** : Following are the limitations of this security design.

Disconnected operation Due to network situations a mobile agent may not be able to effectively utilize network

ISSN 2229-5518

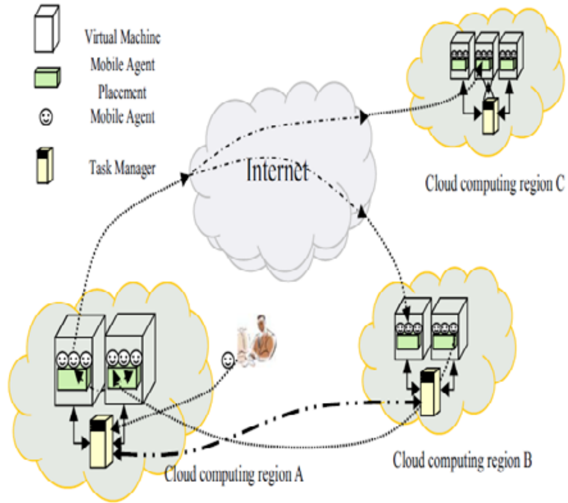


Fig. 5. [10] Mobile Agent Based Open Cloud Computing Federation mechanism based security services. This causes the mobile devices to be disconnected.

Privacy Sensitive user files are collected by organizations which host such services.

D NEED FOR ELASTIC MOBILE APPLICATIONS

As far as the end user is concerned, it does not matter how the service is provided. What the mobile user needs is a cloud mobile application store. But, unlike the applications that are downloaded onto the end user's phone, these applications can be launched on the device or cloud, and can be migrated between them according to dynamic changes of the computing environment or user preferences. Users can access them using mobile browser. Above all, there are heavy limitations on the applications due to limited resources such as low CPU frequency, small memory, and a battery-powered computing environment. As proposed in [9] this kind of applications are called elastic applications.

Elastic Framework Architecture: This framework contains the following components.

1) Elastic Application : are the user chosen applications to be run on the elastic framework under the various device constraints. It typically includes a user interface (UI).

2) Elasticity Manager : runs on the device and monitors and manages the resource requirements of the weblets in an application. It also makes decisions whether to run on the device itself or on the cloud. It also runs an optimizer to decide the cost of power consumption for running weblets and chooses the optimal solution.

3) Weblets : An application consists of a set of functionally independent and communicating set of units called weblets.

4) Router : is the intermediate layer which receives the requests from user interface and passes it to the weblets. This layer is necessary since migration of weblets must be invisible to user interface.

5) Cloud elasticity service : This service allocates resources to the weblets. It consists of

Cloud Manager maintains usage information for various parts of the applications running on cloud.
Cloud Manager provides facility to install and maintain application on device.
Sensing Information Collection performs the collection of operational data on cloud platform.
Node Manager runs on each node on the cloud and communicates directly with the cloud manager and application manager.
The Figure 7 shows the main components.

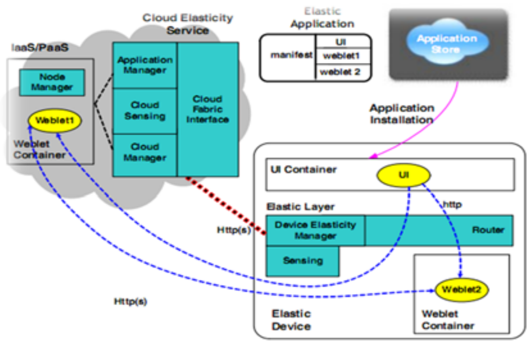
Elastic applications are not constrained by the facilities in mobile device. If more compute (or storage) is needed then this can be obtained from the cloud. Migration of functionalities gives more flexibility to the device.

IV. BENEFITS OF MOBILE CLOUD COMPUTING

Mobile cloud applications move the computing power and data storage away from mobile phones and into the cloud, bringing apps and mobile computing to not just smartphone users but a much broader range of mobile subscribers. In this section, we enlist the possible benefits of Mobile Cloud Computing.

Mobile Cloud Computing will help to overcome limitations of mobile devices in particular of the processing power and data storage.
It also might help to extend the battery life by moving the execution of commutation-intensive application 'to the cloud'.
Mobile Cloud Computing is also seen as a potential solution for the fragmented market of mobile operating systems with currently eight major operating systems.
Fig. 6. [9] Components of Elastic Framework Architecture

ISSN 2229-5518



Elastic framework architecture.

Mobile Cloud Computing can increase security level for mobile devices achieved by a centralized monitoring and maintenance of software,

It can also become a one-stop shopping option for users of mobile devices since Mobile Cloud Operators can simultaneously act as virtual network operators, provide e-payment services, and provide software, data storage, etc. as a service.

A number of new technical functionalities might be provided by mobile clouds. In particular, provisioning of context- and location-awareness enables personalization of services is an attractive functionality.

Mobile Cloud Computing might open the cloud computing business that is currently almost exclusively addressing businesses to consumers since they will significantly benefit from the above described options.

V. CONCLUSION

The concept of cloud computing provides a brand new opportunity for the development of mobile applications 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. A cloud application needs a constant connection that might prove to be an Achilles heel for the cloud computing movement. However as mobile internet capabilities continue to get better, it is likely that solutions to this particular problem will become apparent. New programming languages such as HTML 5 already provide a solution by enabling data caching through a mobile device, and this allows a cloud application to continue working if connection has been momentarily lost.

VI. FUTURE WORK

Considering the importance of Mobile Cloud Computing from this discussion, which is poised to be the fifth utility [2] in the future, we would like to explore further architectures that are plausible. Adequate security measures have to be incorporated to support the low processing ability at the client-side. Further, we would like to tes t the feasibility of extrapolating concepts from cloud computing in the domain of large-scale computers to the realm of mobile world. Also, the cost policy needs to be evaluated as it could prove to be a hindrance to the growth of Mobile Cloud Computing.

REFERENCES

[1] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy H. Katz, Andrew Konwinski, Gunho Lee, David A. Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia. Above the clouds: A berkeley view of cloud computing. Technical Report UCB/EECS-2009- 28, EECS Department, University of California, Berkeley, Feb 2009.
[2] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, and Ivona Brandic. Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Gener. Comput. Syst., 25(6):599–616, 2009.
[3] Andreas Klein, Christian Mannweiler, Joerg Schneider, and Hans D. Schotten. Access schemes for mobile cloud computing. In Eleventh International Conference on Mobile Data Management (MDM), 2010, pages 387–392, 2010.
[4] Jon Oberheide, Kaushik Veeraraghavan, Evan Cooke, Jason Flinn, and Farnam Jahanian. Virtualized in-cloud security services for mobile devices. In MobiVirt '08: Proceedings of the First Workshop on Virtualization in Mobile Computing, pages 31–35, New York, NY, USA,
2008. ACM.
[5]OpenMobster. http://code.google.com/p/openmobster/. 2010.
[6] Rajiv Ranjan and Rajkumar Buyya. Decentralized overlay for federation of enterprise clouds. CoRR, abs/0811.2563, 2008.
[7] ABI Research. http://www.abiresearch.com/. 2010.
[8] B. Rochwerger, D. Breitgand, E. Levy, A. Galis, K. Nagin, L. Llorente, R. Montero, Y. Wolfsthal, E. Elmroth, J. Caceres, M. Ben-Yehuda,W. Emmerich, and F. Galan. The RESERVOIR Model and Architecture for Open Federated Cloud Computing. IBM Journal of Research and Development, 53(4):Paper 4, 2009.
[9] Xinwen Zhang, Joshua Schiffman, Simon Gibbs, Anugeetha Kunjithapatham, and Sangoh Jeong. Securing elastic applications on mobile devices for cloud computing. In CCSW '09: Proceedings of the 2009 ACM workshop on Cloud computing security, pages 127–134, New York, NY, USA, 2009. ACM.
[10] Zehua Zhang and Xuejie Zhang. Realization of open cloud computing federation based on mobile agent. In ICIS '09: IEEE International Conference on Intelligent Computing and Intelligent Systems, 2009., volume 3, pages 642–646, 2009.