

A study of mobile cloud computing methods and possible use case scenarios (December 2017)

Tugcan OLGUN

Abstract—Smartphones are not the phones which have been used a decade ago anymore. They provide powerful and useful usage of technology in every aspect from financial management to entertainment applications. This advancement on the technology brings a problem which is the administering the computational ever increasing requirements of these mobile applications. In designing process of a system which mainly will be used in mobile platform, unlimited resources of a smart phone such as computational power, memory, storage, has to be considered along with battery drain of the software. Considering these attributes change from smartphone to smartphone, a global way to remedy this situation is necessary. To overcome this issue, cloud computing technology offers unlimited storage and computation.

Index Terms—Cloud computing, mobile cloud computing, smart phones, resource offloading, offloading decisions, mobile constraints, cloud computing methods

I. INTRODUCTION

Cloud computing receives an attention because it works in several situations and it can provide scalability both vertically and horizontally as well as providing accessibility. Mobile devices became very popular for multiple application areas. Because the smart phones can now provide several solutions, became the main focus lately. As they do have limited resources in terms of computational power, storage and limited battery capacity, mobile cloud computing methods comes to rescue. They can provide solutions to remedy these problems as problem based or general all-in-one solution.

Mobile devices can connect to internet via two ways, either via an access point network or through a mobile. When it comes to mobile network connection, a device can connect either through cellular or satellite, which is Base Station (BS) or satellite communication with direct link. As the former is not as popularly used because most smartphones are not equipped with satellite communication module, the designing may be considered only towards to the devices using BS as the main communication. The other way of communication is through access points, where users connect to the internet via Wi-Fi and reach cloud computing functionality without utilizing telecom services which may charge them credits or

money so the decision of choosing cloud computing must involve the idea of cost using the data as well.

Cloud computing decision is usually made based on the steps shown in Figure I.

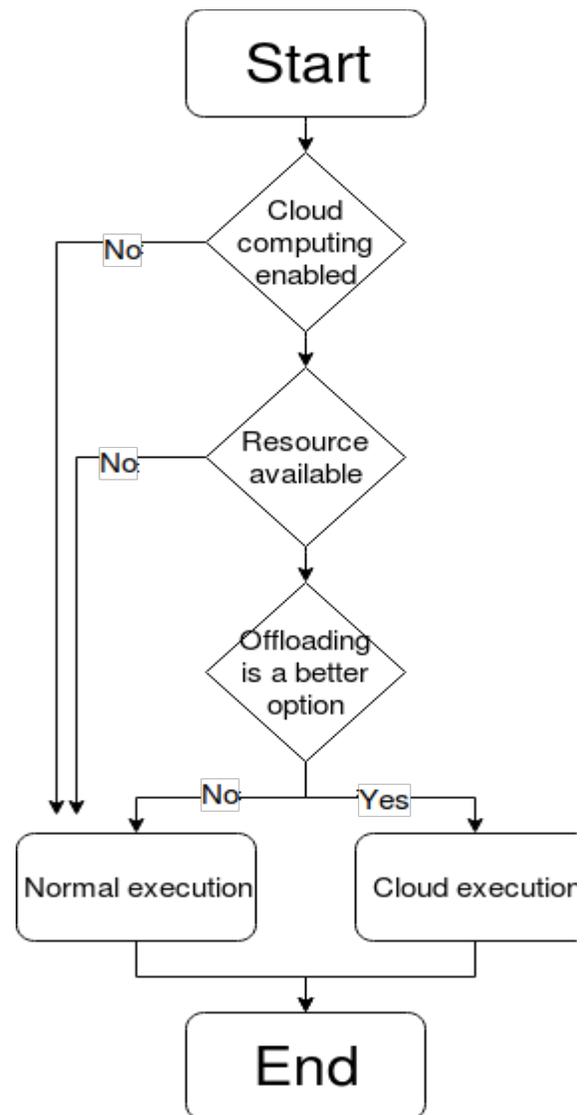


Figure I: Cloud computing decision

†This paper, based on cloud computing techniques on mobile devices for the class SWE 577, Bogazici University is sent to Fatih Alagoz (fatih.alagoz@boun.edu.tr) on 18th of December 2017.

Olgun, T. studies Software Engineering at Bogazici University. He also works as a back-end developer for Ambeent Wireless A.S. at Istanbul Technical University Ari Teknokent building. (e-mail: tugcan.olgun@boun.edu.tr)

There are several models to provide model cloud computing regarding the concerns of performance, energy and storage.

II. MOBILE CLOUD COMPUTING APPLICATION MODELS

A. Performance Based Application Models

I. CloneCloud

CloneCloud is based on augmented execution technique which is giving parts of the execution to the cloud so it will be executed remotely. The main advantage of this method is the availability of recovering of data and the application in the case of losing the phone. On the contrary the main disadvantage of this approach are the security and privacy. In case of a security vulnerability, all the data is there for the steal. CloneCloud does not require programmer to convert the process. A clone of the application is run on the server and based on the pure request that is unmodified execution. When the application decides to send the execution to cloud, it goes into sleeping state while waiting the results from the cloud. CloneCloud system have five types of augmented executions:

Primary functionality outsourcing: It sends all intensive main executions to the cloud, leaving the application light and fast.

Background augmentation: It sends not frequently used non user interaction based executions, thus background servers to cloud.

Mainline augmentation leakage: It sends only debug related execution such as data leakage, fault tolerance and memory leakage.

Performance enhancement: It tweaks the Virtual Machine (VM) settings to enhance the performance.

Augmented throughput multiplicity: It schedules to gain performance by parallel executing the clones.

II. Zhang model

Zhang model is based on the approach which is called weblets. Which is a transparent functional unit that can compute, store and communicate. It has three types of elasticity patterns:

Replication pattern: Weblets have multiple copies executing only one task. It reduces the latency.

Splitter pattern: It includes several implementations on a single weblet in a shared resource. It has the advantage of containing multiple services thus increasing the user experience.

Aggregator pattern: When an activity occurs it pushes it to the user. It is very useful when it comes to certain functions such as email. When an email comes, it relays the information.

B. Energy based application models

The main idea behind energy based application models are to reduce the limited energy consumption on smartphones by offloading the energy heavy computations from mobile to cloud.

I. μ Cloud

In this method application components are differentiated from each other and the decision to compute either on the cloud or local is made locally. The application must be divided into several parts where each function is heterogeneously separated from each other. The main advantage of this approach is that every function is decoupled. The disadvantage of it that it requires skilled programmers to the development of given functions

C. Constraint based application models

It is design to execute on devices that have limited resources such as storage. In this method, lightweight are executions are carried locally.

I. Satyanarayanan

This method uses mobile devices as a client only tools while deploying all resource intensive workloads to the near infrastructure. The word infrastructure is used here with the purpose of explaining the main advantage of this method which is offloading to a nearby device instead of full fledged cloud solution on the hopes that it lowers the latency. Because the latency is one of the prime issues when it comes to any remote solution. It uses virtual machine method called cloudlets. While sending remoting the process, it compresses the execution thus it may burn more energy than it initially thought. Moreover the process of delegating the process between VMs in cloudlet system, it increases the latency thus unenabling this method to be used as a real time based applications.

II. Giurgiu

In this method the cloud solution thus the virtual machines are distributed on servers, allowing reduced latency on transfers and cost. It uses a layered functional structure distributed between smartphone and cloud thus allowing optimized computation. The advantage of this approach is heterogeneous client side environment. However the layering functions makes it more difficult to make a change because the user interface is coupled with service logic.

D. Multi-objective Application Models

Primary principle in multi-objective application model is to accomplish both performance gain and energy efficiency at the same time. There is a trade off for the trial of optimizing both aspects at the same time but in general, it is more general purpose than the rest based on the application that is.

I. MAUI

It provides offloading that requires minimum programmer intervention. It uses a profiler to ascertain whether an execution is resource heavy in both local and cloud computing solutions. By using the profiler if the remote execution is beneficial, it offloads the computation thus saving energy. The profiler gathers information for the energy consumption and data transfer requirements and with the Microsoft's .NET

Common Language Runtime based solver it makes the decision. In this method independent methods can be marked for remote execution allowing dynamic partitioning. Given that the data transfer is expensive, it chooses storage offloading based on the idea.

II. ThinkAir

This method allows method-level offloading with deploying multiple copies of the smartphone on the cloud and managing the execution. It requires only minimal changes to the source code so its usage is easy and changing the code does not crave excessive coding. Profilers that the ThinkAir method place monitor the executions and saves it for the later use. An execution controller on the local side makes the decision based on the methods that are marked offload available.

III. Cuckoo

The method is designed for android platforms and based on partial offloading. The developer of the android app has to separate between remote available methods from the local apps via a definition language called AIDL. The fundamental benefit of this method is that it allows partial offloading of applications based on predefined rules. The downside is the method does not allow asynchronous communication and does not store states thus it is a stateless management.

III. CLOUD COMPUTING APPLICATION EXAMPLES

I. Mathematical tools

These tools are very computation heavy. For instance the operations on very large matrices require an intensive computation power. This type of applications can be developed using MAUI, ThinkAir, μ Cloud, eXCloud and Cuckoo model.

II. Files search

Due to the increasing amount of storage capacities on mobile devices, searching for a file may require a long time. Given for example a device that has 128 GB of storage capacity with thousands of documents, it can benefit from cloud computing options such as Satyanarayanan and CloneCloud method.

III. Imaging tools

As imagine processing operations are very resource heavy that can take up several minutes, these kind of applications can get cloud computation aid with methods such as MAUI, ThinkAir, μ Cloud, eXCloud, and Cuckoo model.

IV. Games

Games use excessive computation but because the user interaction is extremely important in games, latency is something to be considered. Using cloud computing solutions for games that require instant input and output such as real

time strategy games is not advisable. However when it comes to use the formula on games that does not require instantaneous response such in case of turn based games, the following methods can be used: MAUI, ThinkAir, μ Cloud, eXCloud, and Cuckoo model.

V. Download applications

Downloading speed changes depends on the distance and latency of the source material, for certain applications it is better to use cloud for downloading then download the files directly from cloud for consistency. For this purpose Zhang model can be selected.

VI. Antivirus applications

To keep the threats away from smartphones, antivirus applications are crucial but there is a trade off. Because these applications scans every file in the system, the computational power it requires is huge. To remedy this problem Satyanarayanan and CloneCloud models can be considered.

IV. CONCLUSION

Several methods are discussed to find a suitable mobile cloud computing method. To make the decision which method is to be used, several applications types are examined. To ease the developers' coding process in dealing with cloud computing, new programming tools are required to overcome the complexities. The model can be based on augmented models where the mobile and the cloud copy of the application work together or it can be based on per method based cloud solutions.

V. RESOURCES

- [1] R. K. Bedi, J. Singh, S. K. Gupta, "Current Trends in Cloud Storage for resource constrained Mobile Devices" IEEE WiSPNET, 2016 .
- [2] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "A Taxonomy of Offloading in Mobile Cloud Computing" IEEE Pervasive Computing, vol. 8, no. 4, pp. 14–23, 2009.
- [3] H. J. La, S. D. Kim, "A Taxonomy of Offloading in Mobile Cloud Computing" IEEE 7th, 2014 .
- [4] M. Altamimi, K. Naik, "A Practical Task Offloading Decision Engine for Mobile Devices to Use Energy-as-a-Service" IEEE 10th, 2014 .
- [5] X Lyu, H. Tian, C. Sengul, P. Zhang, "Multiuser Joint Task Offloading and Resource Optimization in Proximate Clouds" VOL. 66, NO. 4, APRIL 2017 .
- [6] J. Yue, D. Zhao, T. D. Todd, "Cloud Server Job Selection and Scheduling in Mobile Computation Offloading" Wireless Networking Symposium, Globecom 2014 .
- [7] S. K. Abd, S.A.R Al-Haddad, F. Hashim, "Energy-Aware Fault Tolerant Task offloading of Mobile Cloud Computing" IEEE 5th, 2017 .