Project 6b

Cloudlet-Based Multimedia Search and Retrieval

Name	Roll Number	E-mail
Anamika Roychowdhury	MT2011012	anamika.roychowdhury@iiitb.org
Joshi Shruti Sudesh	MT2011051	shruti.joshi@iiitb.org
Nikhil Kumar K	MT2011059	nikhilkumar.k@iiitb.org
Nitish Sharma	MT2011094	nitish.sharma@iiitb.org
Rijutha N	MT2011126	rijutha.n@iiitb.org

Team Leader: Nitish Sharma

Contents

1 Introduction

Cloud computing is a widespread method of making utilities available as a metered service. The end user does not have to know the physical location of the software or the data he/she is requesting. Cloud service providers rent datacenter hardware and software to deliver storage and computing services through the Internet. Cloudlets are an alternative model for building available, reliable and economically efficient cloudbased storage services [1]. The cloud paradigm can be divided into three different services:

- Software as a service
- Platform as a service
- Infrastructure as a service

There are two primary challenges which the cloud based services face:

- Cloud infrastructures are generally centralized for manageability and economy. Therefore, they are vulnerable to central point of failures caused by fires, power outages, natural disasters etc.
- The second challenge lies in the cost. For cloud storage providers, highly available storage means more storage arrays and power backups, escalating the cost of network bandwidth, data center cooling, and well trained IT management personnel [2].

The world has seen a massive growth in technology in the past few decades. Now we are able to connect our mobile devices to the internet and use them to access our email and web. The speed at which the retrieval of information is taking place has become an important aspect. Cloud computing helps us with large amount of storage without loading our mobile systems, but the speed in retrieval was a major flaw in the cloud computing architecture. Cloudlets were introduced to overcome this. The term **cloudlet** is most talked paradigm, primarily referring as the next generation technology for mobile computing [3]. It is not restricted only to the thick

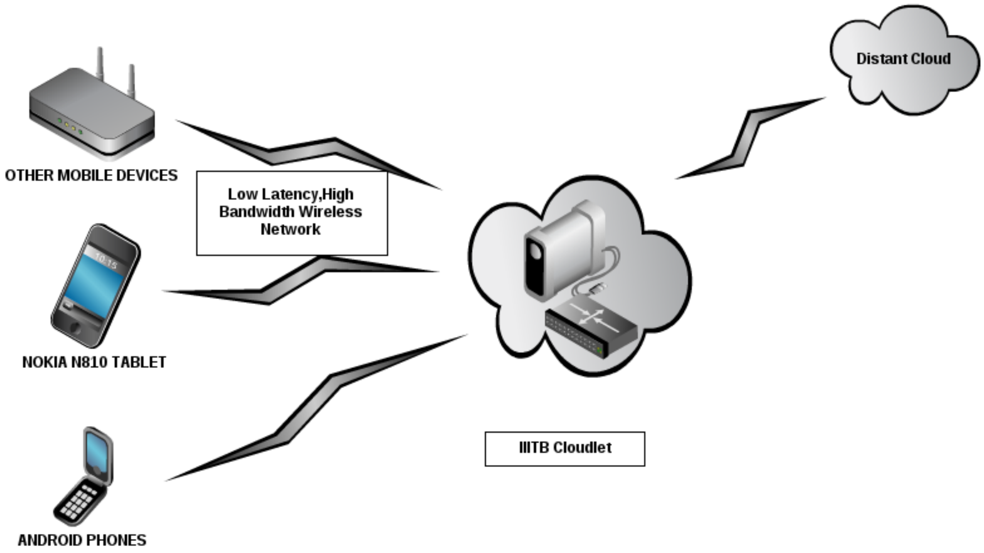

Figure 1: Concept of Cloudlet

clients but also recently there has been a lot of research in the context of the mobile computing. Mobile Computing, has some limitations in terms of the computational ability, which soon will perish because of extension of the cloudlet technology to the mobile platform. They act as a cache which gives faster access to the required data.

In Figure 1 it is shown that how the mobile devices can connect to a nearby cloudlet and can avail services so that they dont have to connect to a distant cloud everytime. This in turn will lead to saving of the system resources of the mobile devices and better utilisation of the system resources of the nearby cloudlet

- Mobile device connect to a cloudlet by a wireless link.
- They will start interacting with the cloudlet.
- The cloudlet can inturn interact with the cloud if needed.
- Need of directly connecting to the cloud is omitted.

2 Project Description

2.1 Objective

The project is aimed at making a cloudlet based search system that can retrieve multimedia data and text files, so that the user can connect a mobile device to the nearby cloudlet and can perform retrieval at a much greater speed and efficiency as compared to connecting to a distant cloud which is inefficient in terms of speed of retrieval.

2.2 Description

Mobiles are generally considered to be resource poor systems since their hardware does not support a number of softwares and also the amount of data that can be stored is also restricted.

The mobile industry has concentrated more towards the weight, size, battery life etc because of which we had to trade processor speed, memory size and disk capacity. Even though mobile hardware continues to evolve, it will take a while before they will be competent with the static hardware in terms of the storage and speed.

Cloudlets are one way to compensate for this incapability. For example lets consider a student of IIITB who wants to retrieve an important multimedia file and all he is got is his mobile. As such a mobile device may not be fully equiped to search the net and play it on his mobile at a faster pace. But if a cloudlet was present somewhere near by, then he can use the cloudlet to search for the file and give it back to him at a appreciable speed. A cloudlet has advantages over a cloud in such circumstances since a cloud experiences a lot of drawbacks like WAN latencies and bandwidth induced delays and jitter. Users can accept delays when they are attempting to do some light browsing work but for resource intensive applications even small delays can distract users.

3 Gap Analysis

Mobile users should utilize nearby computers to obtain the resource benefits of cloud computing without having WAN delays and jitter. We are presenting here a solution to the above fundamental constraints. Rather than relying on a distant cloud. In this we will make the system such that the mobile user can intantiate the cloudlet only when he has a need and accesses it via a wireless LAN. We can deploy this system in:

1. Colleges
2. Coffee shops
3. Airports
4. Railway waiting rooms

Presently this technology has not been deployed in public places in India and in abroad too, even if you want this to be deployed, the present solutions are quite expensive. So we will try to make this technology in such a way that it takes mimimum resources and also easy for the public places to deploy. Some Counter technologies:

1. The Hyrax platform is derived from Hadoop and supports cloud computing on Android smartphones. It exhibits the possibility that a cloud composed of mobile devices can provide much needed services of cloud computing. Since we know long WAN latencies are a fundamental obstacle when a mobile device executes a resource-intensive application like

   - Speech recognition.
   - Donwloading of the multimedia data.
   - Natural language processing.
   - Computer vision and graphics.
   - Machine learning.
   - Planning and Decision-making on a distant high-performance server or cluster.

   We are making an effort in reducing this direct dependency on the distant cloud using **cloudlet based mobile computing**.

2. We can think of a bluetooth technology which can be used to exchange the multimedia files and data files but again the availability, slow speed, limited range and limited data constraint is a problem.

3. If you take the apex dc++ as in IIIT, it will help to solve the slow speed, and limited data constraint to some extent but again availability is an issue and the more important thing is that it allows only laptops to exchange the data not mobile devices. So in this system we will try to create a system that will be available 24x7 and any user under a prescribed phyical region can access the services provided by the cloudlet and can retreive the data and multimedia files with a very low cost without wasting their system resources.

4 System Architecture

4.1 Architecture

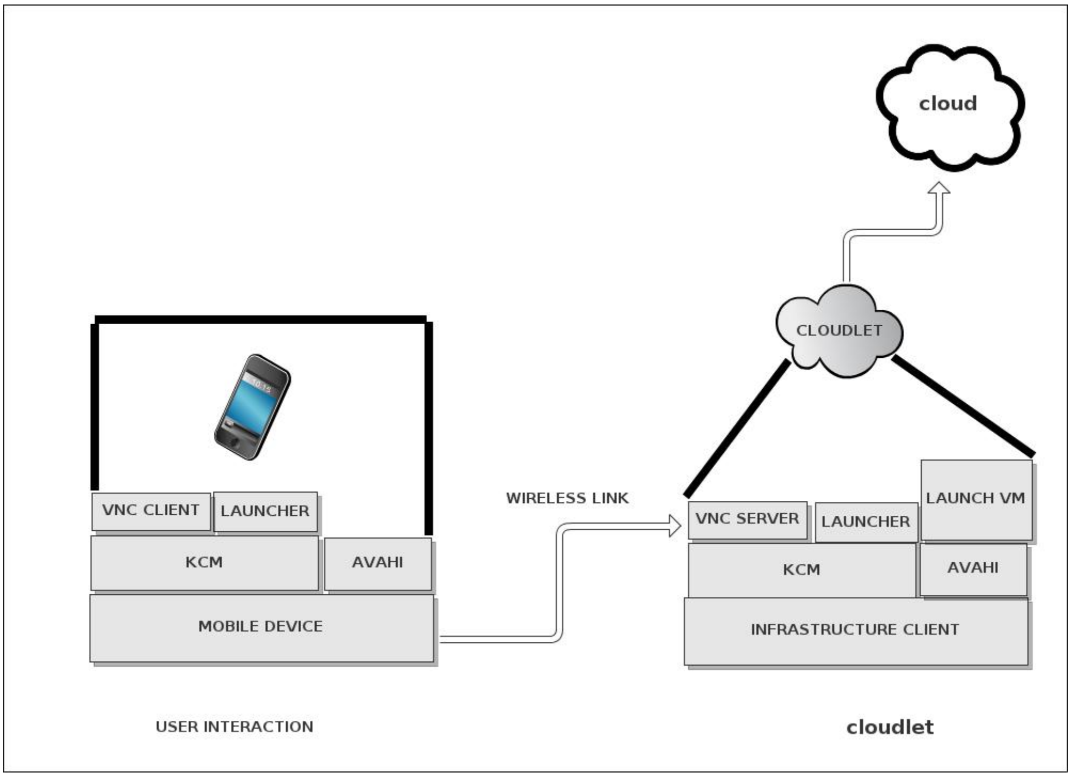

Figure 2: Architecture

VNC is Virtual Network Computing. Both the client and the cloudlet implementing the VNC have to install the necessary softwares to establish connection and transfer data. The softwares include Kimberly which will help in setting up the virtual connection and Avahi which is used to publish or discover other nodes. KCM is the Kimberly Control Manager which manages the connection process. The Launcher is the most important element which helps us in launching an application in the cloudlet. This system do the following things:

- VM overlay creation.
- Overlay binding and user can start interaction.
- VM residue sent back.

4.2 Proposed Solution

In our proposed solution we will focus on developing a cloudlet using hardware virtual machine (VM) technology which we can deploy easily. The Avahi software helps us to detect the IP address of the cloudlet. The Kimberly which is to be installed in both the server and the client helps in establishing a connection. There is a base VM which is installed in the system. The overlay VM holds the state of the application and this is sent to the base VM which is then integrated to launch the application in the cloudlet. This method ensures that the state of the application is carried over to the virtual machine [3].

Figure 2 illustrates the steps of this approach. In our project we are going to have the VM recieve our query and process it. The search result is transferred back to the client system (mobile devices). If we implement the cloudlet as a cluster the launch VM created would be cloned and provided to all the systems in the cluster. This helps in achieving parallelism. The mobile device which is installed with the base VM can connect to any cloudlet which has a public service to offer and also has a VM installed in it. The Avahi software is helpful in publishing or discovering the systems and the services they provide. This is termed as Dynamic VM synthesis [4].

The advantages of such a cloudlet architecture is that our system does not depend on WAN. Also the computing power of the mobile devices are saved since the infrastructure of the server is only used. If the cloudlet crashes, we do not lose data since this is only a cache and the original copies are available elsewhere on a distant cloud.

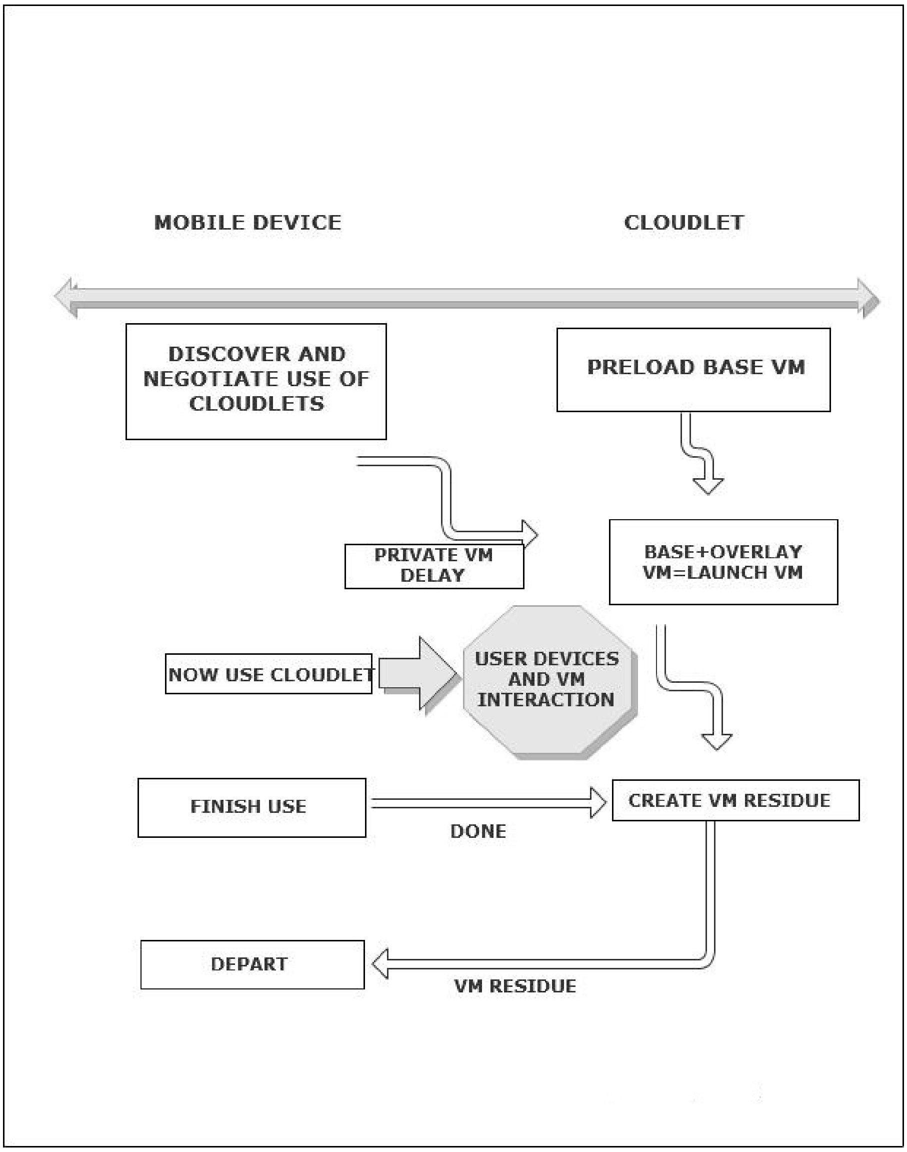

Figure 3: Dynamic Synthesis of VM

The following phases will take place:

1. **VM Overlay Creation**: `Kimberley` uses VirtualBox, a hosted VM for Linux. A tool called `kimberley` is used to create VM overlays, using baseVM. The tool first launches baseVM, and then executes install-script in the guest OS. The result is a VM that has been configured for use by the mobile device. Next, the tool executes resume-script in the guest OS. This launches the desired application, and brings it to a state that is ready for user interaction. This VM, called launchVM, will now be suspended. It can be resumed rapidly at runtime without the delays even in the application launch. After creating launchVM, kimberlize differentiates its memory and disk images with those of the baseVM to obtain the VM overlay. The final step is to compress and encrypt this overlay.

2. **Binding To The Overlay:** The controller of the binding between mobile device and cloudlet is a user-level process called `Kimberley` Control Manager (KCM). An instance of KCM runs on the mobile device and one on the cloudlet, and they together abstract service discovery and helps in binding. KCM supports the browsing and publishing of services using the `Avahi` mechanism in Linux. `Avahi` will be the service discovery mechanism in this project.

3. **Authentication:** After successful authentication, the cloudlet KCM executes a dekimberlize command. This fetches the VM overlay from the mobile device or a Web site, decrypts and decompresses it, and applies the overlay to the base VM (base+VM). The suspended VM is then launched, and is ready to provide services to the mobile device hence user can do the desired operations of multimedia and text retrieval [5].

5 Development Plan

5.1 Development Stages

1. **Stage 1**

   - The advantages and disadvantages of such systems will be reported.
   - The challenges in the deployment of such a cloudlet based system will be analysed.

- We will make a cloudlet infrastructure.

Deliverables:
A better view of the whole concept of cloudlet, its use and scope.
A well structured cloudlet infrastructure.

2. **Stage 2**

    - Configuration of the Kimberley prototype on to the infrastructre.
    - We will deliver a small VM overlay of our client system to a cloudlet infrastructure that already posseses the base VM from which this overlay was discovered.

    Deliverables:
    System is configured and ready for deployment.

3. **Stage 3**

    - Proposal of an algorithm for searching of the multimedia and text files based on the query given by the user.
    - Deployment of the whole concept with the algorithm that we are proposing.

    Deliverables:
    Integration of the algorithm in the system for searching of the files based on their names entered by the user.

4. **Stage 4**

    - Development of the GUI for the client side.
    - We will show the retrievals based on the query given by the user.

    Deliverables:
    Final working of the whole concept.

5.2 System Requirements

5.2.1 Hardware Requirements

- Minimum 3 desktop machines with at least 2 GB RAM and 80 GB.
- Hard disk space each for deploying Ubuntu-one (or equivalent) cloudlet.
- Minimum 10Gig Ethernet connectivity between the machines used for cloudlet deployment.
- Laptop / PC for hosting client system.

5.2.2 Software Requirements

- Operating systems: Ubuntu 10.04 or higher for cloud deployment.
- JDK 6 with Eclipse IDE for development.
- MySQL 5.1 database system.
- `Kimberley` Prototype software for VM synthesis

6 Milestones

Milestone	Task	Due Date
MS1	- The advantages and disadvantages of such systems will be reported. - Presentation on the concepts involved and the architecture invloved. - Challenges in the deployment of such cloudlet based system will be reported. - Setting up cloudlet infrastructure. - This corresponds to stage 1.	7th Feb 2012
MS2	- Configuration of the Kimberley prototype on the infrastructre. - Review of the Deliverables: cloudlet setup.	1st Mar 2012
MS3	- We will deliver a small VM overlay to a mobile device to a cloudlet infrastructure that already posseses the base VM from which this overlay was discovered. - Proposal of an algorithm for searching of the multimedia and text files based on the query given by the user. -This corresponds to stage 2.	22nd Mar 2012
MS4	- Integrating the search algorithm with our present cloudlet infrastructure and testing the procedure. - This corresponds to stage 3.	3rd Apr 2012
MS5	- GUI for the client side. - Coding and configuration complete for Stage 3 and Stage 4. - Coding complete. - Documentation of first draft. - This corresponds to stage 4.	12th Apr 2012
MS6	- Incorporate review comments on project documentation.	15th Apr 2012
MS7	- Project completed. - Documentation submission	19rd Apr 2012

References

[1] W. Zeng, Y. Zhao, K. Ou, and W. Song, "*Research on cloud storage architecture and key technologies*," in *Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human*, ICIS '09, (New York, NY, USA), pp. 1044–1048, ACM, 2009. [Online].Available: `http://doi.acm.org/10.1145/1655925.1656114`.

[2] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "*A view of cloud computing*," *Commun. ACM*, vol. 53, pp. 50–58, Apr. 2010. [Online].Available: `http://doi.acm.org/10.1145/1721654.1721672`.

[3] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "*The Case for VM-Based Cloudlets in Mobile Computing*," *Pervasive Computing, IEEE*, vol. 8, pp. 14 –23, oct.-dec. 2009. [Online].Available: `http://research.microsoft.com/apps/pubs/default.aspx?id=102364`.

[4] A. Wolbach, J. Harkes, S. Chellappa, and M. Satyanarayanan, "*Transient customization of mobile computing infrastructure*," *Proceedings of the First Workshop on Virtualization in Mobile Computing MobiVirt 08*, no. April, pp. 1–5, 2008. [Online].Available: `http://portal.acm.org/citation.cfm?doid=1622103.1622108`.

[5] "*Kimberlize System*," [Online].Available: `http://kimberley.cs.cmu.edu/wiki/Kimberlize/`.