TASK BASED MIGRATION USING MOBILE CLOUD COMPUTING

1MINAL ZAA, 2J.P.GABHANE, 3A.V.DEHANKAR

1Student, 2,3Asst Professor, RTMNU, PCE, Nagpur

Abstract- With the increase in technology, today’s world is using the most important and effective feature as Cloud Computing. Cloud computing acts as services in which shared resources, information, hardware, software are made available to the users as well as other devices as an utility over the network. With increasing technology, handheld devices have become ruggedized for use in different fields. Handheld devices usually includes your mobile devices, tablets, Smartphone, PDAs. With such devices connected to the network plays an important role in concept of Mobile Cloud Computing. In mobile cloud computing, the mobile devices utilize the resources to migrate computation among the mobile nodes and cloud nodes. In this paper, we have combined the working of mobile device in combination with the cloud computing so that task migration can take place with less effort and applications can be utilized easily. This topic helped the purpose of application usage at the same time demonstrate the technique of integration between mobile device and cloud computing which are platform independent, means any mobile device can use this application in any heterogeneous environment.

Keywords- Cloud Node, Computation Migration, Mobile agent, Mobile Node, Overhead, Mobile cloud computing

I. INTRODUCTION

With the introduction of the latest technology in different fields, when every hour something new is released in the market users have to learn more and more to achieve success. They need to be updated else they sometimes have to pay very hard if they miss an important meeting or any task. Today number of products are available in the Market like Google Task, Outlook, Google docs and Desk.com which are built for kind of same purpose. But many of them among this cannot be easily used, and also does not support use in mobile devices. While some of the applications also require subscription. With the increase of number of users of mobile devices these applications are not available easily. There was a time when many languages were used to develop mobile applications. Some of them were J2ME used specially for mobile application, Symbian, BREW[4] etc. However it’s time for Smart phones, Tablets, Touch Screens, PDA etc. Cloud Computing through the internet provides the external users with storage, computing resources including high power computing platforms, data centers and software services. Cloud computing totally frees up the user from managing the hardware, software and data resources along with shifting these burdens to cloud service providers. Thus Cloud computing is a shared pool of computing resources that can be provisioned and released with minimal effort combining the computing power and data storage into web.

With increase in number of users for mobile devices, the ease of accessing applications from anywhere and anytime has become an important factor for mobile users. Cloud is a scalable network of nodes and applications running on cloud are nothing but cloud application[12]. Mobile devices can be used in coordination with the cloud. By connecting the mobile devices with the cloud we can form the mobile cloud computing i.e. accessing your cloud through the mobile device anywhere and anytime. Mobile cloud computing is the use of cloud computing in combination with mobile devices. This makes the use of cloud computing for processing and then storing the data on mobile devices. With this the access to the information can totally be downloaded to your mobile device as well as can be view anytime. But there is a limitation to this, as Mobile devices have less storage capacity, less battery life, less bandwidth, memory. So mobile devices are limited on resources and also may result in accidental damage or lost of device.

With these problems we cannot totally rely on mobile device also, so concept of mobile cloud computing is used. Also when the mobile is connected to the internet, the bandwidth is small, so large amount of data transmission is also not possible. The emanation of mobile agent [3] which is a model of distributed computing plays an important role in remote control, active networks, distribution information retrieval, mobile computing and other applications [2]. The mobile agents should be able to use the resources from the cloud easily, for which an efficient migration is required between the cloud and mobile device. Computation migration can take place at different levels like application level, OS level, middleware level and virtual machine level[10] but main issue is the portability. Therefore the migration should be a substantial migration between cloud node and mobile device so that migration becomes portable and also should not impose heavy overhead.

II. LITERATURE REVIEW

In mobile cloud computing, many different methods have been developed for the migration of task between cloud and mobile devices. The main feature
is portability issue as the use of mobile is concern. With the use of mobile device, mobility problem is already solved, i.e. whenever you go mobile you can access the resources from the cloud anywhere and anytime. Different methods are developed each of them resulting into some portability issue.

With the Clone Cloud method, there is duplicating of runtime environment and then executing it on device or the cloud[7]. Accessing the physical hardware of the cloud through the mobile devices increases complexity. With the use of weblets, dependency on the devices is reduced but some communication overhead is imposed.

Task mobility refers to capability to allow users to continue the operation of their tasks on different nodes which can be done using VM migration[8]. VM migration migrates the whole VM to another computer along with the files stored on shared storage which requires a lot of time transferring to another VM which is a disadvantage. With SOD, it migrates only the topmost frame of the stack while data and code are migrated on demand[5]. A new Excloud method has been developed for the middleware system which combines the SOD and VM migration. In this SOD is integrated on top of VM[6]. Object present in cloud node is different from that of mobile node as well as implementation is different. So communication overhead is imposed. A twin method hierarchy is developed which uses state capturing and state restoration methods[5]. In case if there are extra conditional branching, the overhead will be increased in the execution of the applications.

III. PRESENT WORK

The above approaches is faced with overhead problem, delay or portability issues. To overcome these issues a light weight task migration technique is used. In this case migration takes place between the cloud node and mobile devices. Depending upon the task to be migrated, the migration takes place as, from Mobile device to Cloud node, Cloud to cloud migration and then results from Cloud node back to Mobile device. The migration from mobile device to cloud node requires the use of some of the resources from the cloud through the mobile applications. Mobile applications can access the required resources from the cloud when required. The migration between two clouds requires the access of resources between clouds, load balancing and data access locality[9]. Finally the migration from cloud node to mobile device stores the result back to the mobile device at specific locations. So the result can be available at any location using the unique resources.

In the concept of mobile cloud computing, Mobile cloud applications move the computing power and data storage away from the mobile devices and into powerful and centralized computing platforms located in clouds, which are then accessed over the wireless connection based on a thin native client. For accessing it over a thin native client, this migration is being done. The mobile device acts as an remote control for providing the inputs and getting back the results. The inputs are provided through the application installed on the mobile phone. The mobile device acts as an input provider and then is connected with the cloud. The working can be illustrated as under.

In this case, when the input is provided to the mobile device, it gets connected to the cloud where evaluation is performed. The mobile node acts as a subscriber and publisher to provide the input and get back the back the result. Here two different cloud servers are used. In this case when the input is provided, it checks whether the resources for the particular task execution are available. If resources availability is there, it executes the particular task on same cloud, or if resources are not available then in that case it migrates the particular task to another cloud for further execution resulting into load balancing and task distribution. On execution of task the results are migrated back to the mobile. These obtained results may be in the form of notifications on the mobile device. The performance of the task in cloud server is very fast and results are obtained within milliseconds resulting into the reduction of the delay.

This obtained notification will give information about the execution of the task. Several computation intensive tasks are taken for execution. Some of them are as calculation of the Fibonacci number using recursion, factorial, LogBase10. For a single task execution the time required may be very less, but in case when multiple task are executed, as cloud server is used the multiple task also gets executed within milliseconds. The execution consist of the process ID, conditional branching, the overhead will be increased in the execution of the applications.

Fig 1: Working and migration between cloud nodes

![Fig 1: Working and migration between cloud nodes](Image)

In this case, when the input is provided to the mobile device, it gets connected to the cloud where evaluation is performed. The mobile node acts as a subscriber and publisher to provide the input and get back the back the result. Here two different cloud servers are used. In this case when the input is provided, it checks whether the resources for the particular task execution are available. If resources availability is there, it executes the particular task on same cloud, or if resources are not available then in that case it migrates the particular task to another cloud for further execution resulting into load balancing and task distribution. On execution of task the results are migrated back to the mobile. These obtained results may be in the form of notifications on the mobile device. The performance of the task in cloud server is very fast and results are obtained within milliseconds resulting into the reduction of the delay.

This obtained notification will give information about the execution of the task. Several computation intensive tasks are taken for execution. Some of them are as calculation of the Fibonacci number using recursion, factorial, LogBase10. For a single task execution the time required may be very less, but in case when multiple task are executed, as cloud server is used the multiple task also gets executed within milliseconds. The execution consist of the process ID along with its status of execution. After execution it specifies the time required for execution along with the cloud server on which the task is executed. In this case a resource allocation algorithm is used which is as follows:

![Image]

Resource Allocation algorithm

Step 1: Check for current cloud
Step 2: Checks the availability of CPU for Cloud node 1
Step 3: Check for the CPU level per task for cloud node 1
Step 4: If CPU cycles are available then execute in cloud node 1
Step 5: If CPU cycle not sufficient, then migrate task to cloud node 2
Step 6: After migration check for the CPU availability
Step 7: Show answer and time executed in mobile

The time for execution is considered in milliseconds. As soon as a task is executed the results are obtained in mobile device. According to the availability of the cloud resources of the different cloud the task are executed or else migrated when resources for execution are not available. In this figure, the Fibonacci series is executed on cloud 1. For LogBase10 it checks the availability of resources on Cloud 1 ,but as resources are not available it migrates the task to Cloud 2 and execution takes place at Cloud2. The default cloud for execution is Cloud 1. For final task of factorial execution, it again checks for the availability of resources at Cloud 1, but again as the resources are not available the task is migrated to Cloud2 and execution takes place at Cloud 2. From the figure, it is clear that the execution takes place within milliseconds reducing the delay factor.

IV. EXPERIMENTAL RESULTS

The experimental results are shown below , the difference when the task execution takes place in a stand alone system compared with the task execution in cloud server.The results shows that the time required in a stand alone system for execution is much more as compared to the time required for execution in a cloud server.

CONCLUSION

We have studied the migration of task between mobile device to a cloud node. This paper provides the results which shows that less time is required for execution on a cloud server rather than a standalone system and also the access latency ,delay and overhead is reduced. Few of the techniques are already developed, but it requires more attention to achieve scalability, granularity in the available task migration techniques to be used with minimum overhead for future purposes. With the above developed techniques, migration between mobile and cloud can be extended to migration of bigger tasks.

REFERENCES


IEEE International Conference on advanced Information networking and Applications.

[3] Yue Jiang, Yi ting Jiang, You Ming Xia. —The design of Migration Algorithm of Mobile Agent Based on the Extended Itinerary Graph


★★★