INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume3, Issue3)
Available online at www.ijariit.com

A Virtual Cloud Computing Provider for Mobile Devices

Dhanya .K
Cambridge Institute of Technology
dhanyakodapuram@gmail.com

Prof. Preethi .S
Cambridge Institute of Technology
preethi.ise@citceh.edu.com

Abstract: A mobile device like a smartphone is becoming one of main information processing devices for users these days. Using it, a user not only receives and makes calls but also performs information tasks. However, a mobile device is still resource constrained, and some applications, especially work related ones, usually demand more resources than a mobile device can afford. To alleviate this, a mobile device should get resources from an external source. One of such sources is cloud computing platforms. Nevertheless, an access to these platforms is not always guaranteed to be available and/or is too expensive to access them. We envision a way to overcome this issue by creating a virtual cloud computing platform using mobile phones.

Keywords: Cloud computing, Mobile Phones, Ad Hoc Network, Hadoop.

INTRODUCTION

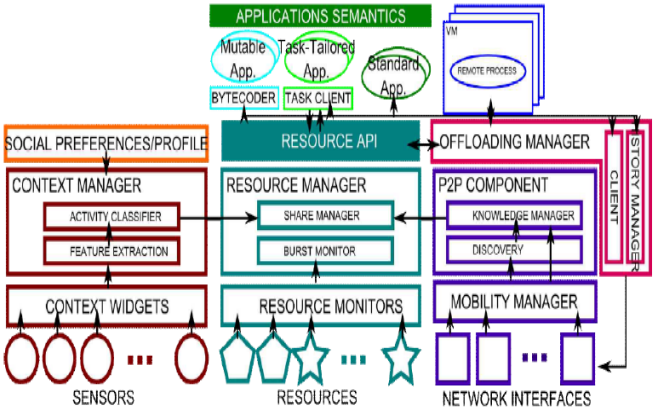
Mobile devices such as smartphones and tablets have become an essential part of our lives, because of their powerful capabilities. Users depend on their mobile devices to make calls, create and edit documents, performing image processing, access the online social networks websites (Facebook, twitter, etc.), organize meetings and make a video and audio calls. On the other hand, the current proliferation of Cloud Computing (CC) paradigm makes a big evolution in Information Technology (IT).The concept of CC relies on a network-based resource sharing to increase resource availability and to reduce the economic and management costs. The cloud is simply a collection of high-performance servers with a huge amount of storage resources connected together and accessible through the Internet. The cloud resources are provided to the users as a service in pay as you use service model. In spite of the benefits provided by the mobile smartphones, and the way they make the life easier; they have many weaknesses, such as limited battery lifetime, limited processing capabilities, and limited storage capacity. It is very important to take into account these limitations since they are hindering mobile users from doing their daily tasks in an efficient way. One solution to overcome these limitations is to integrate Cloud computing technology with Mobile Cloud computing [1]. However, today's large applications are way above the capabilities of just one smartphone. So, the first thing that comes to our mind is to offload the computation to someone else, right? But, it is usually difficult or expensive for one smartphone to share data and computing resources with another machine. Data shared through centralized services requires expensive uploads and downloads that strain wireless data networks. To overcome this, propose guidelines for a framework to create a virtual mobile cloud computing provider through ad-hoc collaboration. Coordinating smartphone data and computing would allow mobile applications to utilize the capabilities of an entire smartphone based cloud while avoiding global network bottlenecks. They argue that, in many cases, processing mobile data in place and transferring it directly between smartphones would be more efficient, cheaper and less susceptible to network limitations than offloading data and processing to remote servers. Their framework emulates a traditional cloud provider using nearby mobile devices. The proposed approach allows avoiding a connection to infrastructure-based cloud providers while bringing benefits of computation offloading. Hadoop ported on mobile devices is used for distributing of processing tasks and storage and communication is based on XMPP. But Hadoop also creates a bunch of issues when they have tried to port it.

RELATED WORK

Mobile cloud computing (MCC) is a very important technology that has many useful applications in our world. The smartphones are used in many aspects of our life, shopping on the Internet, and creating and distributing many types of files. But these devices have many limitations including short battery life time and limited storage and processing. Mobile Cloud Computing technology can help to overcome these limitations. There are many related works in the literature about mobile devices and cloud computing and their useful applications in many life aspects. J.H. Christensen presents general requirements and key technologies to achieve the vision of mobile cloud computing. The author introduces an analysis on smartphones, context awareness, cloud and restful based web services, and explains how these components can interact to create a better experience for mobile phone users. X. Luo [6],

introduces the idea of using cloud computing to enhance the capabilities of mobile devices. The main goal of this work is to show the feasibility of such implementation, introducing a new partition scheme for tasks. The best point about this paper is the considerations about using the cloud to back mobile computing. Another approach introduced by Chun and Maniatis[7] explore the use of cloud computing to execute mobile applications on behalf of the device. They propose the creation of clone VMs to run applications/services the same way that they will run on mobile devices in order to avoid inconsistencies produced to run part of a program in different architecture.

GENERAL ARCHITECTURE FOR THE AD HOC MOBILE CLOUD



The process for the creation and usage of a virtual cloud provider [2] is simple: If a user is at a stable place and wants to execute a task which needs more resources than available on the device, the system listens for nodes in the vicinity. If available, the system intercepts the application loading and modifies the application in order to use the virtual cloud. To support this process, we propose the architecture shown in above figure.It consists of five main features: Application manager; Resource manager; Context manager; P2P component and; Offloading manager. The Application Manager is in charge of launching and intercepting an application at loading time and modifying an application to add features required for offloading – proxy creation, RPC support - according to the current context. The *Resource Manager* is in charge of application profiling and resource monitoring on a local device. For each application, a profile is defined in terms of the number of remote devices needed to create a virtual cloud and sensibility to privacy and amount of resources needed for the migration to happen (in average). This profile is checked by the application manager whenever an application is executed in order to determine whether an instance of the virtual provider should be created or not. The *Context Manager* wields and synchronizes contextual information from context widgets and makes it available in some way for other processes. It is composed by three subcomponents: context widgets, context manager, and social manager. The *Offloading manager* component is in charge of sending and managing jobs from the node to other remote devices, plus receiving and processing jobs sent from them. It communicates with the P2P component once a job is issued to the respective node, and waits for the results to be delivered back to the application. This component is the one in charge of detecting failures in the execution and to re-emit them. It also is in charge of creating protected spaces for the execution of the tasks coming from remote nodes. This protected spaces (represented here as a VM), are utilized to block the access to sensitive data located on the devices.

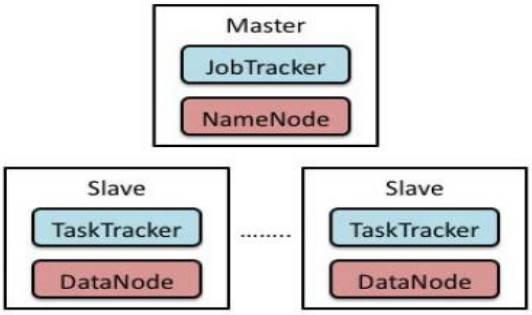
CURRENT IMPLEMENTATION

This project consists of two sub-implementations.

i. Cloud computing provider client.
ii. Ad Hoc mobile cloud framework.

Both are developed based on Hadoop, a cloud computing platform from Apache. For the former, we used Retro weaver to port a client of Hadoop to Java 1.4, ensuring that it will run on top the PhoneME4, Mysaifu5, and JamVM6, the selected target VMs for the mobile devices. For the later, we exploit the Hadoop's API and create our own implementation. Most of the classes and interfaces related to the file system were replaced by direct downloads from the source mobile device, while the map/reduce framework calls were replaced by RPC methods implemented using the Jabber-RPC extension.

HADOOP FRAMEWORK



Hadoop is an Apache open source framework written in java that allows distributed processing of large datasets across clusters of computers using simple programming models. A Hadoop frame-worked application works in an environment that provides distributed storage and computation across clusters of computers. Hadoop is designed to scale up from a single server to thousands of machines, each offering local computation and storage.

MAP REDUCE

Hadoop Map Reduce is a software framework for easily writing applications which process big amounts of data in parallel on large clusters (thousands of nodes) of commodity hardware in a reliable, fault-tolerant manner. The term Map Reduce actually refers to the following two different tasks that Hadoop programs perform:

The Map Task: This is the first task, which takes input data and converts it into a set of data, where individual elements are broken down into tuples (key/value **The Reduce Task:** This task takes the output from a map task as input and combines those data tuples into a smaller set of tuples. The reduce task is always performed after the map task pairs).

Hadoop Might not be the Answer

Hadoop implements much of the core functionality needed for ad-hoc clouds, including global data access, distributed data processing, scalability, fault-tolerance, hardware interoperability and data-local computation. Since Hadoop is mainly designed for deployment on many wired ethernet connected servers, the major problem is how to change Hadoop framework to run on a simple network of mobile devices. So, is Hadoop the solution for creating an ad-hoc cloud using mobile devices. Hadoop achieves poor results when is ported on top of a MANET (Mobile Ad-Hoc Network) because is not aware of mobility aspect – what happens if a node goes down but another one could come in to replace it during a Map-Reduce run. Hadoop is designed for Ethernet. Running it on mobile devices in a WiFi network achieves poor Performance.

Another issue is that Hadoop suffers from low performance with small files since it is programmed to create a new JVM (Java Virtual Machine) per each map processing. Anyway, they have found a workaround for this behavior but it triggered another issue: multiple small files deal with memory problems since Hadoop [5] creates one record in the data table for each file. Though concatenation of input files might solve the problem, it is not always the right solution (i.e. image compression). Moreover, during start up, each Data Node scans its file system and provides the Name Node with the information about stored files. Thus, it implies that the more files there are, the longer it would take to scan and transmit information.

One of the biggest issues of Hadoop is that it relies a lot on disk writes, notably on HDFS (Hadoop Distributed File System), which handles both reading the input data and also writing the output data. On top of a wireless environment, Hadoop's file system can be damaging in many cases. HDFS implements and accelerates the data replication between the nodes using pipelines. This makes network I/O instead of disk I/O the application's bottlenecks. However, in the article's experimented-version the interfaces related to the file system where replaced by direct downloads from the source mobile device.

Job creation process in Hadoop is heavy as it involves a lot of things. Had the input data being large, then we would see Hadoop doing better overall. Otherwise, the overhead of starting the cluster could be bigger enough to get better run-times by using a local written C/C++ program. The question that arises is: for what amount of data should Hadoop be really feasible for porting the framework on mobile devices. However, we believe is not enough for 100KB.

COMPARATIVE STUDY

Mobile cloud computing helps to avoid the following problems:

1. Mobile devices are resource constrained, usually, demand more resources than a mobile device can afford. Solve this problem with the help of cloud computing platform such as Amazone EC2, Micro Soft Azure etc.
2. Accessing cloud computing platform is very expensive for mobile devices and also we cannot predict the network availability. Mobile Adhoc network solves this problem and also maintaining the benefits of computational offloading.
3. Performance can be improved by sharing of resources and parallelisation of operations.
4. Cloudlet help in supporting resource intensive and interactive mobile applications by providing powerful computing resources to mobile devices with lower latency.

5. Hyrax, a mobile cloud computing client that allow mobile devices to use cloud computing platform efficiently.

CONCLUSION

In this paper, we present the motivation and preliminary design for a framework to create Ad Hoc cloud computing providers. This framework takes advantage of the pervasiveness of mobile devices, creating a cloud among the devices in the vicinity, allowing them to execute jobs between the devices. The work presented here is preliminary and creates the foundation for future work. Hadoop is not the right choice when it comes to building such an infrastructure. It is worthless using this framework on mobile ad hoc networks due to many probability issues mentioned earlier.

REFERENCES

1. Hoang T. Dinh, Chonho Lee, Dusit Niyato, and Ping Wang, A Survey of Cloud Computing: Architecture, Applications, and Approaches, Wirel. Commun. Mob Comput. 2011.
2. Gonzalo Huerta-Canepa and Dongman Lee, A Virtual Cloud Computing Provider for Mobile Devices, Proceedings of the1st ACM Workshop on Mobile Cloud Computing & Services: Social Networks and Beyond, 2010.
3. R. Balan, M. Satyanarayanan, S. Park, T. Okoshi, Tacticsbased remote execution for mobile computing, Proceedings of the 1st International Conference on Mobile Systems, Applications and Services, ACM, 2003
4. C. Clark, K. Fraser, S. Hand, J. Hansen, E. Jul, C. Limpach, I.Pratt, A. Warfield, Live migration of virtual machines, Proceedings of the 2nd conference, Symposium on Networked Systems Design & Implementation—Volume 2, USENIX Association, 2005.
5. Hadoop's small files problem,http://www.cloudera.com/blog/2009/02/ the-small- files problem.
6. X. Luo, "From Augmented Reality to Augmented Computing: Look at *Cloud-Mobile convergence, International Symposium on Ubiquitous Virtual Reality, vol. 0, 2009.*
7. B. Chun and P. Maniatis, "Augmented Smartphone Applications through Clone Cloud Execution," *HOTOS workshop*, USENIX, 2009.
8. Niroshinie Fernando, Seng W. Loke and Wenny Rahayu, Mobile cloud computing: A survey, Future Generation Comp. Syst, 2013.

