MOBILE_SOCIAL_TV: MOBILE TV WITH
SOCIAL INTERACTION ON CLOUD
PLATFORM

Rajendraprasad B G¹, Mrs. Ragini Krishna²

¹ Mtech sapthagiri college of engineering
² Assistant professor Dept of CS&E
sapthagiri college of engineering

Abstract

Recently many mobile entertaining or media applications have been launched, but most popular app like Facebook, Twitter, YouTube have larger demand among users. But these Media applications are limited by the unstable wireless connectivity and limited battery lifetime of mobile devices, due to these problems the quality of service encountered by the users of the Smartphone is not up to the mark. In order to overcome from these problems cloud computing technology has been used. cloud computing technology, with its vast resources provides compensation to the limitations of the mobile devices and offers a ideal platform for the required mobile services. In this paper we review the design of the Mobile Tv with social interaction on Cloud platform(Mobile_social_Tv), which makes use of both Platform-as-a-Service(PaaS) and Infrastructure-as-a-Service cloud(IaaS) services to offer the co-viewing experience of video watching to a group of mobile users.

1. INTRODUCTION

Nowadays laptops ,Notepad and Smartphones are shipped with many microprocessor cores and GB's of RAM's, they have high computation power than Normal desktop computers of late 90's. The wide deployment of 3G,4G broadband cellular infrastructures has further increased the use of Smartphone by Common people. Every Smartphone users need the fastest technologies like 3G, Wi-Fi for fast web access & chatting. These technologies focus more on the challenging scenarios such as real-time video streaming and online gaming, for social interacting, and exchanging emails. Recently many mobile entertaining or media applications have been launched, but most popular app like Facebook, Twitter, YouTube have larger demand among users. But these Media applications are limited by the unstable wireless connectivity and limited battery lifetime of mobile devices, due to these problems the quality of service encountered by the users of the Smartphone is not up to the mark. In order to overcome from these problems cloud computing technology has been used.

Cloud computing provides low-cost, agile scalable resource supply and power efficient communication between streaming devices. Cloud can reduce load of computation and other tasks which is involved in a mobile application. This significantly reduces battery consumption of the mobile devices. It effectively utilizes the cloud computing standards to offer a co-viewing experience of video watching just as users are watching TV at their homes.

In Mobile social TV, the system effectively makes use of the cloud computing Services like Infrastructure-as-a-Service (IaaS) and Platform-as-a-Service (PaaS) to provide good quality of video streaming, a mobile users can fetch a on-demand or live video to watch from video gallery and also you can invite your family and friends for watching the video. They can also chat with each other while Watching the video. In traditional system each users uses dish TV, set boxes for digital broadcasting of channels. The CloudMoV utilizes agile resource support and the functionalities which are Platform-as-a-Service (PaaS) and Infrastructure-as-a-Service cloud (IaaS) to achieves the following goals.

Encoding Flexibility- There are various Video streaming devices which have large screen size and small screen displays, also they have various screen resolutions. The devices that support for customized media playback hardware, video playback and also support for various codec. Mobile social TV unloads the transcoding streams of different devices at real time in an IaaS cloud. A VM(virtual machine) is employed for a user in the IaaS cloud. The VM downloads the video and transcodes it into the proper formats while considering particular configurations and current connectivity quality of the Smartphone.

Spontaneous Social Interactivity-Multiple mechanisms are included in the design of Mobile social tv for concurrent viewing and social chatting with each other. First factor is efficient synchronization mechanisms in which the same portion of the video program can be able to watch by friends in the same circle and share their views and comments about video with each other. Second factor is an efficient message communication mechanism which is designed for social interactions among friends. PaaS cloud can be used for social interaction support due to its provision of powerful
underlying platforms. IaaS is the foundation of cloud computing. Some space can be taken on rent in data centres from IaaS provider to maintain and deploy services and storage

Battery Efficiency- analysis indicates that the display and network modules both 3G and Wi-Fi absorb maximum power in video streaming device To save energy consumption from the network module of video streaming devices cloud computing technologies are used. Cloud can reduce load of computation and other transcoding tasks which is involved in a video streaming application.

Portability- A prototype Mobile social tv system is implemented following the philosophy of “Write Once, Run Anywhere” (WORA): both the the back-end server modules and front-end mobile modules are implemented in Java, HTML and MySql database is used for storing user data The client module can run on any devices supporting HTML5 browser, we deploy the system on virtual machine running on the cloud and conduct thorough tests on EC2 micro instance

2. RELATED WORK
A number of mobile TV systems have emerged in past years both software and hardware developed in mobile devices. Some early systems bring the co-viewing experience to Smartphone’s on the move these things focus more on convergence of the mobile network and the television network. research has focused on documenting the demand of social communication among mobile users.

2.1 Amazon Elastic Compute Cloud (EC2)
Amazon EC2 is a central part of Amazon's cloud computing platform, Amazon Web Services (AWS). EC2 allows users to rent virtual computers on which they can run their own computer applications. EC2 allows scalable distribution of application by providing a Web service through which a user can boot an Amazon Machine Image to create a virtual machine, which in turn calls an instance containing any desired software. Each user can build, launch, and end server instances as planned by paying the hour for active session of servers, hence the term elastic. EC2 provides users with control over the geographical location of instances that allows for latency optimization and high levels of redundancy.

Amazon EC2 is a representative IaaS and paas cloud, offering raw hardware resources including networks to users, CPU, storage and EC2 is an appropriate platform for computing intensive tasks in mobile social TV i.e., those the surrogates carry out.

2.1.1 Amazon Machine Images (AMI):
An Amazon Machine Image AMI gives the information required to start an instance in the cloud which acts as a virtual server in the cloud. You can notify an AMI when you launch an instance and also you can launch as many instances as possible from the AMI as you need.

An AMI includes the following:

- For each instances root volume a template is assigned.
- Start-up permissions that guides & control which AWS accounts can use the AMI to launch instances

2.1.2 Amazon EC2 Instances: Amazon EC2 provides each instance with a consistent and predictable amount of CPU capacity, regardless of its underlying hardware. Amazon EC2 dedicates some resources of the host computer, such as instance storage, memory and, CPU to a specific instance. Amazon EC2 utilizes other resources of the host system, such as the disk subsystem of instances and the network. If each virtual machine on a host system tries to utilize one of these shared resources as much as possible, each receives an equal amount of that resource however when a resource is not utilized properly, an instance can grab the available resources.

Each instance type provides low or high performance through shared resource. For example the type of instances with high I/O performance have a larger allocation of shared resources. The variance of I/O performance is also reduced due to Allocating a larger share of shared resources. For most applications average I/O performance is more than enough. However for applications that require more consistent I/O performance, available instance types are shown in Fig 1.

2.2 HTTP Live Streaming (HLS)
HTTP Live Streaming (also known as HLS) is an HTTP-based media streaming communications protocol implemented by Apple as part of their QuickTime and IOS software. It works by dividing the overall stream into a small sequences of HTTP-based file downloads, each download loads one small piece of an overall potentially unbounded transport stream. As and when the stream is played, the user may select one from a number of different alternate streams containing the same material encoded at a different data rates, letting the streaming session to adapt to the available data rate. At the
beginning of the session streaming it downloads an extended M3U playlist containing the metadata for the various sub-streams which are available.

<table>
<thead>
<tr>
<th>Instance Family</th>
<th>Instance Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>General purpose</td>
<td>m1.small</td>
</tr>
<tr>
<td>Compute optimized</td>
<td>c1.medium</td>
</tr>
<tr>
<td>Memory optimized</td>
<td>m2.xlarge</td>
</tr>
<tr>
<td>Storage optimized</td>
<td>hi1.4xlarge</td>
</tr>
<tr>
<td>Micro instances</td>
<td>t1.micro</td>
</tr>
<tr>
<td>GPU instances</td>
<td>cg1.4xlarge</td>
</tr>
</tbody>
</table>

Fig.1 Instance Types

2.3 Prior work
A number of mobile TV systems have been emerged in recent years both hardware and software are driven by the advancement in Smartphone's. Some early systems [1], bring the co-viewing experience to the mobile users. But they concentrate more on convergence of the mobile network and television network, than exploring the demand of social message exchange among mobile users. There are some other works dedicated to enhance social elements to television systems [13], [14], [10], S. Kosta [2] have proposed a supporting work that makes it simple for developers to migrate their Smartphone applications to the cloud. Coppens[4] try to add fast screening social interactions to TV but their design is limited to traditional broadcast programs. Y. Feng Z. Liu [6] designed and implemented a new system framework to provide the required system support to achieve spontaneous social interaction with other users in the same mobile application. Oehlberg [13] conduct a plenty of experiments on human social activities while watching different types of program. Even Though these designs are not that much suitable for in a mobile environment. Chuah [11] extend the social experiences of viewing traditional broadcast programs to mobile devices, but the quality of service is not that much feasible. Schatz et al. [12] have designed a mobile social TV system which is customized for Symbian devices and dvb-H networks for large number of users. Compared to these prior work and systems we target at a design for a generic, featuring co-viewing experiences among mobile users, portable mobile social TV framework. Our substructure is open to all Internet based video programs either live or on-demand and supports a wide range of devices with HTML5 compatible browsers installed without any other component on the devices.

3. MOBILE_SOCIAL_TV: ARCHITECTURE AND DESIGN
Figure 2 gives an overview of the architecture of "Mobile TV with social interaction on cloud platform[12]. In this an instance or a VM surrogate is created for online user in an IaaS cloud infrastructure. Virtual machine acts as a gateway between the mobile device and the video gallery. Cloud based mobile social tv can be divided into the following major functional modules:

Converter: It is nested in each VM and is responsible for converting the video stream from the video gallery into appropriate format. Before delivery to the user the video stream is further encapsulated into a proper transport format.

Messenger: It is the client side of the social cloud residing in each surrogate in the Infrastructure as a service cloud. The Messenger periodically inspect the social cloud for the social data on behalf of the mobile user and pre-processes the data into a light-weighted format like plain text files at a considerably less frequency. The plain text files are asynchronously delivered from the surrogate to the user in a less traffic pathway. In the other way the messenger disseminates this user’s messages (invitations and chat messages) to other users via the data store of the social cloud.

Social Cloud: The social cloud is structured over the general (Paas)Platform as a service cloud. This framework is compatible to other platforms. It stores all the user credentials in the system including the statuses of all users in the user list and messages (friend request and chat histories) and records of the existing sessions in each chat session. The user credentials are categorized into different kinds and formed into different columns and rows in a table in relational database, respectively. The social cloud database is inquired from time to time by the VM instance.

Gateway: The gateway provides verification services for users to login to the Mobile_social_TV system and stores users information in a MySQL database and it also saves the information about the available instances in the IaaS cloud memory.
memory table. After a user successfully login into the system, a Virtual machine will be assigned from the available pool of VM to the user. To guarantee small query latencies the in-memory table is used, because the instance pool is updated periodically as the gateway preserves and destroys VM instances based on the workload. Along with that the gateway also also stores each user’s friend list in a plain text file (in XML formats) which is immediately uploaded to the surrogate after it is assigned to the user.

Subscribe: In this module user can download the video in required format, This module downloads video in high speed and clean video streaming. Every authorized user can download and watch the videos.

4. MOBILE_SOCIAL_TV: PROTOTYPE IMPLEMENTATION
We have implemented a mobile social TV system, and deployed it Amazon EC2 clouds, which is the most widely used public (PaaS) platform as a service and (IaaS) Infrastructure as a service cloud platforms. Amazon EC2 is a central part of Amazon.com’s cloud Amazon Web Services (AWS). EC2 allows users to rent an instance on cloud to run their applications. Amazon EC2 allows scalable distribution of applications by providing a Web service through which a user can boot an Amazon Machine Image to create a virtual machine which intern calls an instance containing desired software. A user can terminate, create, and launch server instances as and when it is needed and pay as much as you use the service hence the name elastic. EC2 provides users with control over the divisional location of instances that allows for latency improvement and huge levels of redundancy.

The user first connects to the login page of Mobile TV with social interaction on cloud platform. He is assigned a instance from the Instance pool only after the user successfully logs in through the gateway. Then the user is automatically redirected to the assigned VM surrogate, and welcomed by a portal page. Upon user login the gateway collects the device configuration information by examining the User-Agent header values and this information will be sent to its VM for making the decision of video encoding. The user can enter the title of the video or live broadcast he chooses to watch on the Subscribe tab of the page after he clicks the Subscribe button the address of the video is sent to the VM surrogate which downloads the stream on behalf of the user, encodes and sends properly converted video segments to the user. When the user requests for a video the playlist is first displayed and a video is selected from the playlist the client starts to play the video as soon as he receives the video. While watching a video the user can check out his friends status on the Friends tab and send request to one or more friends to join him in viewing the video. When a user receives an friend request from a user the profile pictures of those who have sent request will be highlighted on the Friends tab and decides to join the session also even he can choose to watch from the start or catch up with the viewing progresses. Users in the same session can exchange opinions and comments on the Chat tab where new chat messages can be entered and the chat history of the session is shown. The Info tab shows an abstract of the video as edited by the session host.

Fig.2 Architecture of a Mobile_social_Tv.

Fig.3 client UI of Mobile_social_Tv

Fig.4 Friend and Chat tabs. (a) Friend tab (b) Chat tab
5. CONCLUSION
We conclude results prove the superior performance of Mobile_social_TV, in terms of efficiency, timely social-interaction, transcoding and scalability. In Mobile_social_TV a mobile user can fetch a live or on-demand video to watch from video gallery and can request his friends and family to watch the video and chat with their friends simultaneously while enjoying the video. In the current prototype we do not enable sharing of encoded streams among surrogates of different users. In future work such sharing can be enabled and carried out in a peer-to-peer fashion, e.g., the surrogate of a newly joined user may fetch the transcoded streams directly from other surrogates, if they are encoded in the format/bit rate that the new user wants.

REFERENCES

Rajendraprasad B.G received B.E degree in Computer Science Engineering from Siddaganga institute of technology, tumkur and currently doing M.Tech degree in Sapthagiri College of Engineering.

Mrs. Ragini Krishna completed M.Tech and currently working as Assistant Professor in Sapthagiri College of Engineering.