A Novel Cloud based Mobile Social TV

Srilatha Pavuluri[1] and S.Sravani[2] and Kadiyala Ramana[3]

[1]PG Student, Dept. of CSE, Chiranjeevi Reddy Institute of Technology, Anantapur, Andhra Pradesh, India

[2]Assistant Professor, Dept. of CSE, Chiranjeevi Reddy Institute of Technology, Anantapur, Andhra Pradesh, India

[3]Assistant Professor, Dept. of IT, Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh, India

ABSTRACT: Cloud computing is now considered a major commanding hosting platform in numerous areas including mobile computing. Several mobile television systems have sprung up in current years, motivated by hardware as well as software progression in mobile devices. By delegating towards cloud of communications as a provision, mobile television becomes familiarized to streams intended in aid of a variety of applications. We intend CloudMoV to effortlessly make use of agile resource support as well as prosperous functionalities obtainable by Infrastructure-as-a-Service cloud and Platform-as a-Service cloud. Novel system of cloud-based social television makes easy for consumption of two most important functionalities in the direction of contributing portable users such as, a user provoke frequent friends to stare at comparable video in co-viewing by social interactions, and replacement text communication while examination. The system is capable to attain an important power saving, by opportunistically switching device among high-power as well as low-power transmission modes throughout streaming.

KEYWORDS: Transcoder, Mobile client, Gateway.

I.   INTRODUCTION

The rapid reinvention in the smart phones is the combining the features of a mobile device such as a PDA (personal digital assistant), a digital camera, a media player and a GPS navigation unit plus the features of a laptop, including web browsing, Wi-Fi, multiple microprocessor core, third party applications and giga bytes random access memories RAMs. The most popular Smartphone today are powered by Google's Android and Apple's iOS and the wide deployment of 3G broadband cellular networks. Cloud Computing is the delivery of computing resources over the internet. Cloud services allows businesses and individuals to use hardware and software that managed by the third parties at remote locations. A mobile user exploits VM technology to rapidly instantiate customized service software on a nearby cloudlet [6]. Think air[10] is the frame work for migrating Smartphone applications to the cloud. Although many media and mobile social applications have emerged, truly destroyer ones gaining mass acceptance are still impeded by the limitations of the present wire technologies and mobile devices, among which unstable wireless connection and battery life time are challenging one. In now a day's 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 apps, and emails.

Many mobile social or media applications have been launched recently, but most popular app like Twitter, Facebook have large demand among users. But there are some limitations in the current mobile and wireless technologies, in which unstable connection bandwidth and lifetime of battery are some of the problems. Cloud computing provides low-cost, agile scalable resource supply and power efficient mobile communication. 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. Cloud-MoV effectively utilizes the cloud computing to offer a living room experience of video watching just as users are watching Television at their homes. In mobile social television, mobile users can import video on demand or a live to watch from any streaming site like Vimeo, Ustream ,YouTube and invite their family and for watching the video concurrently and chat with each other while enjoying the video. In traditional system each users uses dish-TV, set boxes for digital broadcasting of channels. The Cloud-MoV utilizes agile resource support and the functionalities which are provided by both an Infrastructure-as-a-Service (IaaS) cloud and a Platform-as-a-Service (PaaS) cloud. It therefore blends social awareness and co-viewing experience among friends on the go. As opposed to traditional Television watching, mobile social Television is well suited to today's life style, where family and friends may be separated geographically but hope to share a co-viewing experience. While social Television

enabled by set-top boxes over the traditional TV systems is already available for us, it remains a challenge to achieve mobile social Television, where the at the same time as viewing experience with friends is enabled on mobile devices.

The design achieves following goals.
1. Encoding Flexibility
2. Battery Efficiency
3. SpontaneousSocial Interactivity
4. Portability
.

II. RELATED WORK

Many mobile television systems have been popular in recent years with advanced software and hard wares in mobile devices. Some systems [3] bring living room experience to the small screens on the move but they more focused on the how to deliver the content and which technology is used but they never focused on the limitations of mobile devices. Coppens added social interactions [8] to the television but the design is limited to traditional broadcast channels. Schatz designed mobile social television [14] which is for DVB-H networks and symbian devices and it not support wider audience. The design of cloud based novel mobile social television supports all internet based video programs i.e. on demand video and live TV with HTML5 compatible browsers and supports a wide range of devices, without any other mandatory components on the devices. Minimizing power consumption is the one of the major challenge for any application in mobile devices. It is able to select the correct tradeoff between application quality and energy conservation by monitoring energy supply and demand[9]. Amigo TV combines broadcast television and community with rich communication in order to leverage a social experience [1]. Cloud computing having a most powerful platform in mobile computing. In cloud computing mobile devices workload is offloaded to rich resource infrastructures in dynamic synthesis. The cloud based mobile social television is designed based on the cloud computing. Cloudmov uses both platform as a service and infrastructure as a service to offer living room experience to a group of people who interact socially while watching the video and sharing the video.

III. CLOUDMOV: ARCHITECTURE AND DESIGN

As a novel Mobile-Social TV system using cloud computing (CloudMoV), provides two major functionalities to participating mobile users: (1) Universal streaming: A user can stream a live or on-demand video from any video sources he chooses, such as a TV program provider or an Internet video streaming site, with tailored encoding formats and rates for the device each time. (2) Co-viewing with social exchanges: A user can invite multiple friends to watch the same video, and exchange text messages while watching. The group of friends watching the same video is referred to as a session. The mobile user who initiates a session is the host of the session. The architecture of CloudMoV and the detailed designs of the different modules is presented in the following.

A. Key Modules

Fig. 1 gives an overview of the architecture of CloudMoV. A surrogate (i.e., a virtual machine (VM) instance), or a VM surrogate equivalently, is created for each online mobile user in an IaaS cloud infrastructure. The surrogate acts as a proxy between the mobile device and the video sources, providing transcoding services as well as segmenting the streaming traffic for burst transmission to the user. Besides, they are also responsible for handling frequently exchanged social messages among their corresponding users in a timely and efficient manner, shielding mobile devices from unnecessary traffic and enabling battery efficient, spontaneous social interactions. The surrogates exchange social messages via a back-end PaaS cloud, which adds scalability and robustness to the system. There is a gateway server in CloudMoV that keeps track of participating users and their VM surrogates, which can be implemented by a standalone server or VMs in the IaaS cloud.

The design of CloudMoV can be divided into the following major functional modules.
1. Transcoder: It resides in each surrogate, and is responsible for dynamically deciding how to encode the video stream from the video source in the appropriate format, dimension, and bit rate. Before delivery to the user, the video stream is further encapsulated into a proper transport stream. In this implementation, each video is exported as MPEG-2 transport streams, which is the de facto standard nowadays to deliver digital video and audio streams over lossy medium.

International Journal of Innovative Research in Computer and Communication Engineering

2. *Reshaper*: The reshaper in each surrogate receives the encoded transport stream from the transcoder, chops it into segments, and then sends each segment in a burst to the mobile device upon its request (i.e., a burst transmission mechanism), to achieve the best power efficiency of the device. The burst size, i.e., the amount of data in each burst, is carefully decided according to the 3G technologies implemented by the corresponding carrier.

3. *Social Cloud:* It is built on top of any general PaaS cloud services with BigTable-like data store to yield better economies of scale without being locked down to any specific proprietary platforms. Despite its implementation on Google App Engine (GAE) as a proof of concept, our prototype can be readily ported to other platforms. It stores all the social data in the system, including the online statuses of all users, records of the existing sessions, and messages (invitations and chat histories) in each session. The social data are categorized into different kinds and split into different entities (in analogy to tables and rows in traditional relational database, respectively). The social cloud is queried from time to time by the VM surrogates.

4. *Messenger*: It is the client side of the social cloud, resid-ing in each surrogate in the IaaS cloud. The Messenger periodically queries the social cloud for the social data on behalf of the mobile user and pre-processes the data into a light-weighted format (plain text files), at a much lower frequency. The plain text files (in XML formats) are asynchronously delivered from the surrogate to the user in a traffic-friendly manner, i.e., little traffic is incurred. In the reverse direction, the messenger disseminates this user's messages (invitations and chat messages) to other users via the data store of the social cloud.

5. *Syncer:* The syncer on a surrogate guarantees that view-ing progress of this user is within a time window of other users in the same session (if the user chooses to synchronize with others). To achieve this, the syncer periodically retrieves the current playback progress of the session host and instructs its mobile user to adjust its playback position. In this way, friends can enjoy the "sitting together" viewing experience. Different from the design of communication among messengers, syncers on different VM surrogates communicate directly with each other as only limited amounts of traffic are involved.

6. *Mobile Client:* The mobile client is not required to install any specific client software in order to use CloudMoV, as long as it has an HTML5 compatible browser (e.g., Mobile Safari, Chrome, etc.) and supports the HTTP Live Streaming protocol. Both are widely supported on most state-of-the-art smartphones.

7. *Gateway:* The gateway provides authentication services for users to log in to the CloudMoV system, and stores users' credentials in a permanent table of a MySQL database it has installed. It also stores information of the pool of currently available VMs in the IaaS cloud in another in-memory table. After a user successfully logs in to the system, a VM surrogate will be assigned from the pool to the user. The in-memory table is used to guarantee small query latencies, since the VM pool is updated frequently as the gateway reserves and destroys VM instances according to the current workload. In addition, the gateway 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.
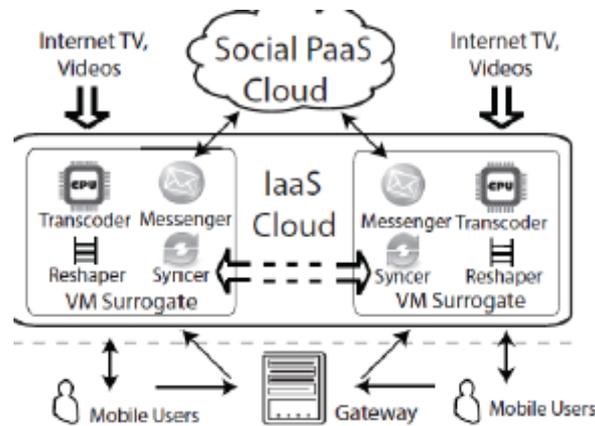
Fig.1: Architecture of CloudMoV

The key designs in CloudMov is described as following.

A. Loosely Coupled Interfaces

Similar in spirit to web services, the interfaces between different modules in CloudMov, i.e., mobile users, VM sur-rogates, and the social cloud, are based on HTTP, a univer-sal standard for all Internet-connected devices or platforms.loose coupling between users and the infras-tructure, almost any mobile device is ready to gain access to the CloudMoV services, as long as it is installed with an HTTP browser. The VM surrogates provisioned in the IaaS cloud cooperate with the social cloud implemented on a PaaS cloud service via HTTP as well, with no knowledge of the inner components and underlying technologies of each other, which contributes significantly to the portability and easy maintenance of the system. For social message exchanges among friends, CloudMoV employs asynchronous communication. All the exchanged messages are routed via the surrogates to the social cloud, which efficiently organizes and stores the large volumes of data in a BigTable-like data store. The VM surrogates query the social cloud frequently and processes the retrieved data into XML files, for later retrieval by users in an asynchronous fashion. Such a design effectively separates the mobile users from the social cloud to significantly simplify the architecture, while the extra delay introduced at the VM surrogates is ignorable, as shown in Sec. V.

B. Pipelined Video Processing

Both live streaming of realtime contents and on demand streaming of stored contents are supported in CloudMoV. Video processing in each surrogate is designed to work on the fly, i.e., the transcoder conducts real time encoding from the video source, the encoded video is fed immediately into the reshaper for segmentation and transmission, and a mobile user can start viewing the video as soon as the first segment is received. To support dynamic bit rate switch, the transcoder launches multiple threads to transcode the video into multiple bit rates once the connection speed between the surrogate and the mobile user changes. The IaaS cloud where the surrogates are deployed, represents intensive jobs.

C. Burst Transmissions

First, 3G power states, different from Wi-Fi which is more similar to the LANed Internet access, 3G cellular services suffer from the limited radio resources, and therefore each user equipment (UE) needs to be regulated by a Radio Resource Control (RRC) state machine. Different 3G carriers may customize and deploy complex states in their respective cellular networks. Different states indicate different levels of allocated radio resources, and hence different levels of energy consumptions. Second, Transmission mechanism, In CloudMoV, maximum conservation of the battery capacity of the mobile device, and design a burst transmission mechanism for streaming between the surrogate and the device is aimed. Using the HTTP live streaming protocol, the mobile device sends out requests for the next segment of the video stream from time to time. The surrogate divides the video into segments, and sends each segment in a burst transmission to the mobile device, upon such a request. When the mobile device is receiving a segment, it operates in the high-power state when there is nothing to receive, it transfers to the low-power state via the intermediate state and remains there until the next burst (segment) arrives. Third, to decide the burst size, i.e., the size of the segment

transmitted in one burst, it is necessary to take into consideration characteristics of mobile streaming and energy consumption during state transitions.

D. VM Surrogates

All the VM surrogates are provisioned from Amazon EC2 web services and tracked by the gateway.. To implement all the video processing related tasks using ANSI C, to guarantee the performance is proposed.
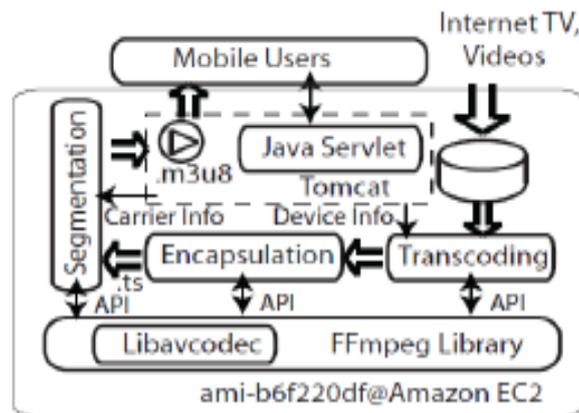

Fig. 2: Streaming architecture in each customized VM image

In particular, FFmpeg together with libav-codec as the groundsill library to develop the transcoding, segmentation and reshaping modules on the VM surrogates is needed and also a Tomcat web server (version 6.5) to serve as a Servlet container and a file server on each surrogate is needed. Once a VM surrogate receives a video subscription request from the user, it downloads the video from the source URL, and processes the video stream by transcoding and segmentation, based on the collected device configurations by the portal. Fig. 2 shows the streaming architecture in our customized VM image.

E. Data Models in the Social Cloud

GAE(Google Application Engine) is mainly used as the back-end data store to keep the transient states and data of CloudMoV, including users' online presence status, social messages (invitation and chat messages) in all the sessions. With Jetty as the underlying Servlet container, most Java-based applications can be easily migrated to GAE, under limited usage constraints, where no platform-specific APIs are enforced for the deployment. GAE provides both its Java Persistence adapter and a set of proprietary low-level APIs to map the relational data. Once a user logs in to the system and enters the URL of a video to watch, a session ID is generated for the new session (corresponding to viewing of this video), by combining the user's "username" in the system with the time stamp when the session is created. The gateway delivers an HTTP request to a Servlet listener running on GAE, to notify it that an entry for the newly joined user should be added, with the user's "username" as the key and other information (URL of the subscribed video, the session ID, etc.) as the value. Whenever a user decides to join a session hosted by his friend upon invitation, his VM surrogate switches to download the video of the session, and at the same time sends an HTTP request to the social cloud, for updating the session ID in this user's entry to the new one. If the user wishes to synchronize his playback progress with that of the session host, his VM surrogate synchronizes with the session host to maintain the playback "currenttime" value (HTML5 property).

The social cloud maintains a "Logs" entry for each existing session in CloudMoV, with the session ID as the primary key and an array list as the value, which corresponds to individual messages in this session. When a user in a session posts a comment, this message is first sent to his VM surrogate, which further injects the message into the social cloud via another Servlet listener. The message is stored as a "Message" entry in the social cloud, with the message content as the value, and an auto-generated integer as the key. Entries "Logs" and "Message" are annotated by a @OneToMany relationship, to facilitate the data management. VM surrogates of users in the same session send periodical HTTP query requests to the social cloud for the latest comments from others. The default interval for retrieval of new comments is 10 seconds. The retrieved messages are stored and updated on the surrogates, which

process them into well-formed XML formats for efficient parsing at the user devices. The user devices retrieve the XML files from the surrogates at a lower frequency (with default interval 1 minute), in order to minimize the power consumption and the traffic. Fig. 3 presents social message exchanges among a mobile user, his VM surrogate, and the GAE. A large number of entries in the social cloud becomes outdated very soon, since users may switch from one session to another, quit the system, and so on, a cron job behind the scene every 10 minutes to clear those outdated entries is launched. For example, for sessions of which everybody has left, their "Logs" entries and all the associated "Message" entries are deleted in a single transaction.
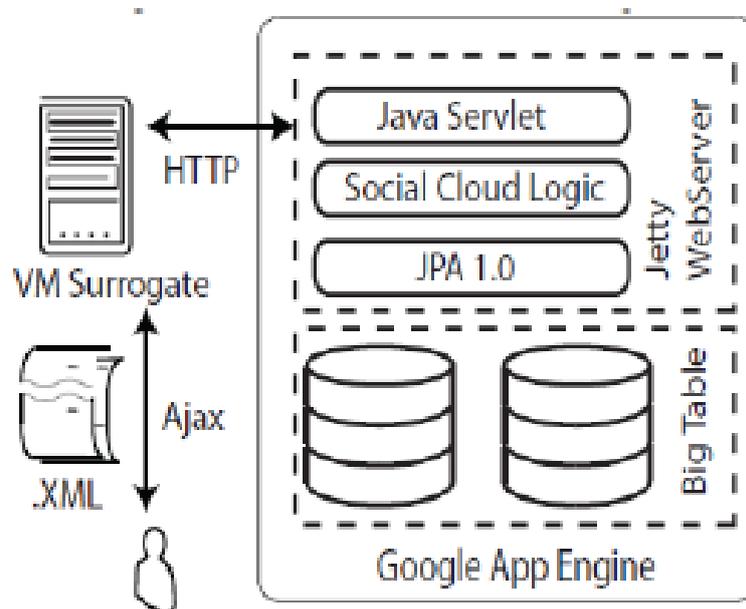

Fig. 3: Social message exchanges via Google App Engine.

IV. CONCLUSION

We conclude results prove the superior performance CloudMo V, in terms of efficiency, timely social interaction, transcoding and scalability. In CloudMoV, a mobile users 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

[1]. J. Santos, D. Gomes, S. Sargento, R. L. Aguiar, N. Baker, M. Zafar, and A. Ikram, "Multicast/broadcast network convergence in next generation mobile networks," Comput. Netw., vol. 52, pp. 228–247, Jan. 2008.
[2]. S. Kosta, A. Aucinas, P. Hui, R. Mortier, and X. Zhang, "Thinkair: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading," in Proc. IEEE INFOCOM, 2012.
[3]. K.Ramana, Dr.A.Subramanyam, and K.suresh, "A Survey on Cloud Computing and Service Oriented Architecture", VSRD International Journal (IJCSIT), November 2011.
[4]. Z. Huang, C. Mei, L. E. Li, and T. Woo, "Cloudstream: Delivering high-quality streaming videos through a cloudbased SVC proxy," in Proc. INFOCOM'11, 2011, pp. 201–205
[5]. NoSQL Date Base. [Online]. Available: http://nosql-database.org/.
[6]. HTTP Live Streaming. [Online]. Available: http://tools.ietf.org/html/ draft-pantos-http-live-streaming-01.
[7]. Z. Liu, Y. Feng, and B. Li, "Socialize spontaneously with mobile applications," in Proc. IEEE INFOCOM, 2012.
[8]. W. Zhang, Y.Wen, Z. Chen, and A. Khisti, "QoE-driven cachemanagement for http adaptive bit rate (abr) streaming over wireless networks," in Proc. IEEE Globecom, 2012.
[9]. J. Flinn and M. Satyanarayanan, "Energy-aware adaptation for mobile applications," in Proc. 17th ACM Symp. Operating Syst. Principles, 1999, SOSP '99, pp. 48–63.

[10]. N. Ducheneaut, R. J. Moore, L. Oehlberg, J. D. Thornton, and E. Nickell, "Social TV: Designing for distributed, sociable television viewing," Int. J. Human-Comput. Interaction, vol. 24, no. 2, pp. 136–154, 2008.

[11]. J. Santos, D. Gomes, S. Sargento, R. L. Aguiar, N. Baker, M. Zafar, and A. Ikram, "Multicast/broadcast network convergence in next generation mobile networks," Comput. Netw., vol. 52, pp. 228–247, Jan. 2008.

[12]. K. Chorianopoulos and G. Lekakos, "Introduction to social TV: Enhancing the shared experience with interactive TV," Int. J. Human- Comput. Interaction, vol. 24, no. 2, pp. 113 120,

[13]. Yu Wu, Zhizhong Zhang, Chuan Wu," CloudMoV: Cloud-Based Mobile Social TV"IEEE 2013

[14]. N. Ducheneaut, R. J. Moore, L. Oehlberg, J. D. Thornton, and E. Nickell, "Social TV: Designing for distributed, sociable television viewing," Int. J. Human-Comput. Interaction, vol. 24, no. 2, pp. 136–154, 2008.

[15]. K. Ramana, T.Hari Krishna, C.V.Lakshmi Narayana, and M.Sankara Prasanna Kumar, "Comparative Analysis on Cloud Computing and Service Oriented Architecture", International Journal of Advanced Research in Technology, September 2011

[16]. N. Ducheneaut, R. J. Moore, L. Oehlberg, J. D. Thornton, and E. Nickell, "Social TV: Designing for distributed, sociable television viewing," Int. J. Human-Comput. Interaction, vol. 24, no. 2, pp. 136–154, 2008.

[17]. M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for VM-based Cloudlets in mobile computing," IEEE Pervasive Comput., vol. 8, pp. 14–23, 2009