A STUDY ON MOBILE SOCIAL TV USING CLOUD COMPUTING

G. Bhaskar

J Nagaraju

M.V.Siva Prasad

O.S.Reddy

Abstract—

The recent cloud computing technology, with its rich resources to compensate for the limitations of mobile devices and connections, can potentially provide an ideal platform to support the desired mobile services. There has been a rapid reinvention in the smart phones by combining the features of a mobile phone with those of another popular consumer device, such as a personal digital assistant (PDA), a media player, a digital camera, and/or a GPS navigation unit. The most popular smart-phones today are powered by Google's Android and Apple's IOS mobile operating systems and the wide deployment of 3G broadband cellular networks. The combination of cloud computing and mobile networks to bring benefits for mobile users, network operators, as well as cloud computing providers. The design of mobile social TV system, using cloud, which can effectively utilize the cloud computing to offer a living-room experience of video watching by mobile users with spontaneous social interactions. In cloud mobile social TV, mobile users can import a live or on-demand video to watch from any video streaming site and invite their friends to watch the video concurrently, and chat with their friends while enjoying the video. It therefore blends viewing experience and social awareness among friends on the move. They can also chat with each other while enjoying the video.

Keywords- Smart phone, IaaS, Cloud Computing, Surrogate, Video Streaming

I. Introduction

A number of mobile TV systems have sprung up in recent years, driven by both hardware and software advances in mobile devices. Some early systems bring the“livingroom” experience to small screens on the move. There has been an rapid reinvention in the smart-phones by combining the features of a mobile phone with those of another popular consumer device, such as a personal digital assistant (PDA), a media player, a digital camera, and/or a GPS navigation unit. The design of a novel enhanced mobile video streaming, which can effectively utilize the cloud computing paradigm to offer a living- room experience of video watching to disparate mobile users with spontaneous social interactions. It is natural to resort to cloud computing, the newly-emerged computing paradigm for low-cost, and scalable resource supply, to support power-efficient mobile data communication with virtually infinite hardware and software resources.

II. Related work

A number of mobile TV systems have sprung up in recent years, driven by both hardware and software advances in mobile devices. Some early systems bring the living room experience to small screens on the move. But they focus more on barrier clearance in order to realize the convergence of the television network and the mobile network, than exploring the demand of “social” interactions among mobile users. Although many mobile social or media applications have emerged, truly killer ones gaining mass acceptance are still impeded by the limitations of the current mobile and wireless technologies, among which battery lifetime and unstable connection bandwidth are the most difficult ones.

We propose the design of a Cloud-based, novel Mobile social TV system. The system effectively utilizes both PaaS (Platform-as-a-Service) and IaaS (Infrastructure-as-a- Service) cloud services to offer the living-room experience of video watching to a group of disparate mobile users who can interact socially while sharing the video. To guarantee good streaming quality as experienced by the mobile users with time varying wireless connectivity, we employ a surrogate for each user in the IaaS cloud for video downloading and social exchanges on behalf of the user. The following advantage offered using the present system

- **Encoding flexibility**: Different mobile devices have differently sized displays, customized playback hardware’s, and various codec’s. Traditional solutions would adopt a few encoding formats ahead of the release of a video program. But even the most generous content providers would not be able to attend to all possible mobile platforms, if not only to the current hottest models. CloudMoV customizes the streams for different devices at real time, by offloading the transcoding tasks to an IaaS cloud.

- **Battery efficiency**: The burst transmission mechanism makes careful decisions on burst sizes and opportunistic transitions among high/low power consumption modes at the devices, in order to effectively increase the battery lifetime.
- **Spontaneous social interactivity**: Multiple mechanisms are included in the design of CloudMoV to enable spontaneous social, co-viewing experience.
- **Portability:** A prototype CloudMov system is implemented following the philosophy of "Write Once, Run Anywhere" (WORA): both the front-end mobile modules and the backend server modules are platforms implemented in "100% Pure Java" Our prototype can be readily migrated to various cloud and mobile with little effort.

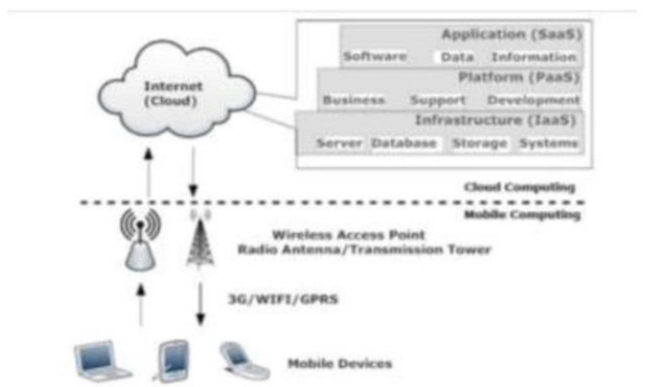

Fig1. Mobile Cloud Computing

A. ARCHITECTURE

Mobile social TV architecture is used to offer living room experience to mobile user while watching video by utilizing cloud computing services. A surrogate or a VM is created for each online mobile user in IaaS cloud infrastructure. The surrogate act as a proxy between mobile devices and video sources, providing transcoding as well as segmentation. They are also responsible for handling frequently exchanged social messages among users, shielding mobile devices from unnecessary traffic, enabling battery efficient and spontaneous social interaction. The surrogates exchange social messages via back end PaaS cloud which adds scalability and robustness to system. The design of Mobile Social TV can be divided into the following functional modules:

Social cloud:

Built on top of any general PaaS cloud services with Big Table-like data store to yield better scalability without being locked down. Stores all social data in system, including online status of all users, records of existing sessions, and messages (invitation and chat history) in each session. The social data are categorized into different kinds and entities. The social cloud is queried from time to time by VM surrogates.

Messenger:

Client side of social cloud residing in each surrogate in IaaS cloud. On behalf of mobile user, messenger periodically queries social cloud for social data and pre-processes the data into light weighted format (Plain text files) with lower frequency. The plain text files are asynchronously delivered from surrogate to the user in traffic friendly manner. In reverse direction, the messenger disseminates this message (invitation and chat messages) to other users via data store of social cloud. Synchronizer on each surrogate guarantees that viewing progress of this user is within a time window of other user in same session. To achieve this, synchronizer periodically retrieves current playback progress of session host and instruct mobile user to adjust its playback position. Synchronizer on different VM surrogates communicate directly with each other as only limited amounts of traffic are involved.

Transcoder:

Resides in each surrogate who is responsible for dynamically deciding how to encode the video stream from video source in appropriate format, dimension and bit rate. Each video is exported as MPEG-2 transport streams, which is the default standard nowadays to deliver digital video and audio streams over lossy medium. Reshaper in each surrogate receive encoded transport stream from transcoder, chops it into segments and then send each segment in burst to mobile device upon its request, to achieve best power efficiency of device. The burst size is carefully decided and implemented by corresponding carrier.

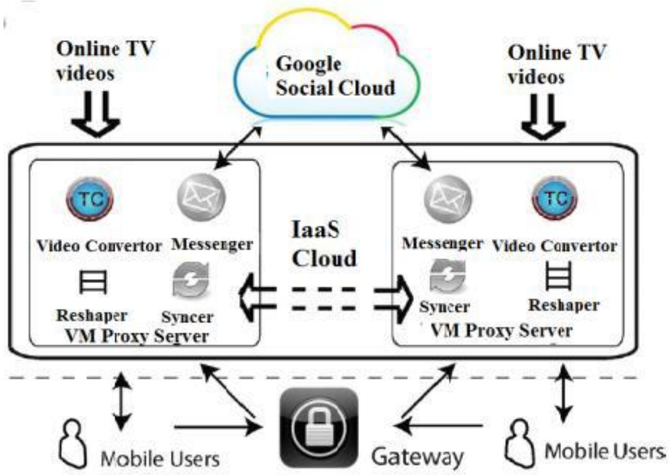

Fig. 1: Architecture of Mobile CloudTV

Gateway:

Provides authentication services for user to log in to the system, and stores user credentials in permanent table of SQL database it has installed. It also stores information of pool of currently available VMs in IaaS cloud in another in memory table. After user log into system, VM surrogates will be assigned from the pool to the user. The in memory table guarantees small query latencies since VM pool is updated frequently as gateway reserves and destroys VM instance according to current workload. It also stores each user's friend list in plain text file which is immediately uploaded to surrogate after it is assigned to user. Mobile client is not required to install any specific software as long as it has an HTML5 compatible browser (e.g., Mobile Safari, Chrome,etc.) and supports HTTP Live streaming protocol. Both are widely supported on most smartphones.

Interfaces:

Interfaces between different modules are based on HTTP, a universal standard for all Internet connected devices or platforms. Almost any mobile device is ready to gain access to services in Mobile Cloud TV, as long as it is installed with HTTP browser. The VM surrogates provisioned in IA as cloud cooperate with social cloud implemented on Papas cloud service via HTTP, with no knowledge of the inner components and underlying technologies of each other, which contributes significantly to portability and easy maintenance of system. For social message exchanges among friends, Mobile Cloud TV employs asynchronous communication. All the exchanged messages are routed via surrogates to social cloud, which effectively organizes and stores large volume of data. The VM surrogates query social cloud frequently and process retrieved data into XML, for later retrieval by users in an asynchronous fashion. Such a design effectively separates mobile users from social cloud to significantly simplify architecture while extra delay introduced is ignorable.

Pipelined video processing:

Video processing on each surrogate is designed to work on fly, i.e., the transcoder conducts real time encoding from video source, the encoded video is fed immediately into the reshaper for segmentation and transmission and mobile user can start viewing video as soon as first segment is received. To support dynamic bit rate switch, transcoder launches multiple threads to transcoder video to multiple bit rates once the connection between surrogate and mobile user changes.

Burst transmission:

3G cellular services suffer from limited radio resources, and therefore each user equipment (UE) needs to be regulated by Radio Resource control (RRC)state machine. Different states in 3G indicates different level of allocated radio resources and hence different level of energy consumption. To make our implementation easier, consider three basic states in our design, which are commonly employed by many carriers, namely CELL-DCH (dedicated physical channel in both uplink and downlink), CELL-FACH (no dedicated channel but UE is assigned in uplink) and IDLE in decreasing order of power levels. UE may stay at a high-power state (CELL-DCH) for data transmission even the data rate is very low. A 3G carrier commonly transfer UE from high power state to low power state for releasing radio channels allocated to this UE to other users. For example, if a UE working at a high power state does not incur any data traffic for a preconfigured period of time then the state of UE will be transferred to a low power. When volume of data traffic rises, UE" wakes up "from a low power state to high power.

III. Working Schedule

All mobile devices installed with HTML5 compatible browsers can use Mobile cloud TV services, as long as the HTTP Live Streaming (HLS) protocol is supported. The user first connects to the login page of MEC. After the user successfully logs in through the gateway, he is assigned a VM surrogate from the VM pool (the hostnames of available VMs, e.g.ec2-50-16-xx-xx.compute-1.amazonaws.com, are main tained in an in-memory table of a MySQL database deployed in the gateway). Then the user is automatically redirected to the assigned VM. Upon user login, the portal collects the device configuration information by examining the "User-Agent" header values, and this information will be sent to its surrogate for decision making of the video encoding formats. The user can enter the URL of the video or live broadcast he wishes to watch, on the "Subscribe" tab of the portal; after he clicks the "Subscribe" button, the address of the video is sent to the VM surrogate, which downloads the stream on the user's behalf, transcoder and sends properly encoded segments to the user. From the surrogate to the mobile device, the video stream delivered using HLS is always divided into multiple segments, with a playlist file (.m3u8) giving the indices. When the mobile client subscribes to a video, the playlist is first downloaded and individual segments are requested by the client in the following. A playlist file may become out-dated if new contents are generated, e.g. in case of a live broadcast. In that case, the mobile client needs to download the playlist again to keep the indices updated. The client starts to play the video as soon as the first segment is received. When watching a video, the user can check out his friends' status (online or offline, which video they are currently watching) on the "Friends" tab and invite one or more friends to join him in watching the video. When a user receives an invitation from a friend (profile pictures of those friends who have sent invitations will be highlighted on the "Friends" tab) and decides to join the session, he can choose to watch from the start, or catch up with the viewing progresses of others by clicking the "Sync" button, which triggers the Syncer functionality in the surrogate. 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. All the data (.xml files) updates are delivered in an asynchronous manner based on AJAX techniques without the need of reloading the portal page.

IV. CONCLUSION

Binding cloud computing with mobile computing helps for providing on demand services in multimedia network. Using Cloud computing services (mainly IaaS) developer deployed an application which allow user to access online videos from internet and also to chat with friends to exchange their comments about video(Social network).By offloading computational work in cloudlet, mobile power is extended to provide better performance. A surrogate for each mobile user can be accessed in case of any failure in framework and it improves the reliability. Mobile cloud TV is constructed in peer to peer fashion where data can be shared among mobile users. When a user request for a video and if it's available in nearby user surrogate, it can be accessed without duplicating the component and thereby reducing overhead. In addition to on-demand cloud resources (e.g., pay a fixed price for certain resources), there are auction based cloud resources whose availabilities are controlled by the current bidding prices and the auctions market. By employing one surrogate for each mobile user, achieve ultimate scalability of the system, and mobility to serve a multitude of mobile devices anywhere, anytime through the channel Internet regardless of environments and platforms .The power states in commercial 3G cellular networks, this propose an energy-efficient burst transmission mechanism that can effectively increase the battery lifetime of user devices. In this case, achieving the optimal sever selection will be more challenging and it should be focused in future development

V. References

[1] 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

[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] Z. Huang, C. Mei, L. E. Li, and T. Woo, "Cloudstream: Delivering high-quality streaming videos through a cloud-based SVC proxy," in Proc. INFOCOM‟ 11, 2011, pp. 201–205.W.-K. Chen, Linear Networks and Systems (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.

[4] T. Coppens, L. Trappeniners, and M. Godon, "AmigoTV: Towards a social TV experience," in Proc. EuroITV, 2004.

[5] 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.

[6] A. Carroll and G. Heiser, "An analysis of power consumption in as smartphone," in Proc. USENIXATC, 2010.

[7] 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.

8] 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, 2008.

[9] M. Chuah, "Reality instant messaging: Injecting a dose of reality into online chat," in CHI ‟ 03 Extended Abstracts on Human Factors in Computing Syst., 2003, ser. CHI EA ‟ 03, pp. 926–927.