Mobile Social TV Interaction through Cloud Computing

Ambika¹, Ramya Bathula²

¹Arjun College of Technology and Sciences, Department of Computer Science and Engineering, Hyderabad, India

²Arjun College of Technology and Sciences, Department of Technology and Engineering, Hyderabad, India

Abstract: Social TV makes watching TV an engaging rewarding experience. You can watch your favorite shows, comment on them, chat with your friends and do many more. The rapidly increasing power of personal mobile devices is providing much richer contents and social interactions to the users on the move. This trend however is suffocated by the limited battery lifetime of mobile devices and unstable wireless connectivity, making the highest possible quality of service experienced by mobile users not feasible. 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. Tough challenges arise on how to effectively exploit cloud resources to facilitate mobile services, especially those with stringent interaction delay requirements. In this paper we will make interaction to mobile services to offer the living-room experience of video watching to a group of mobile users who can interact socially while sharing the video. Here we employ a surrogate for each user in the IaaS cloud for video downloading. The surrogate performs efficient stream transcoding that matches with the present connectivity quality of the mobile user.

Keywords: IaaS, PaaS, Cloud, Surrogate, Transcode

1. Introduction

If you remember up until few years ago, watching TV used to be a social experience with family and friends gathering in a living area to watch their favorite show or movie. Now a days mobiles are shipped with multiple microprocessor cores and gigabyte RAMs; they possess more computation power than personal computers of a few years ago. On the other side, the wide deployment of 3G broadband cellular infrastructures further fuels the trend. Apart from common productivity tasks like emails and web surfing, smart phones are flexing their strengths in more challenging scenarios such as real time video streaming and online gaming, as well as serving as a main tool for social exchanges. This paper describes can invite their friends to watch the video concurrently while watching and enjoying the video. So at the end of the day you will get better content, best quality and connectivity and user can interact with your friends and family. It therefore blends viewing experience and social awareness among friends on the go. As opposed to traditional TV watching, mobile social TV is well suited to today's life style, where family and friends may be separated geographically but hope to share a co-viewing experience. Here we design mobile social Tv to seamlessly utilize agile resource support and rich functionalities offered by both an IaaS (Infrastructure-as-a-Service) cloud and a PaaS (Platform-asa- Service) cloud. Our design achieves the following goals.

- Encoding flexibility
- Battery efficiency
- Spotaneous social interactivity
- Portability

2. Existing System

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.

3. Proposed System

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-asa-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.

4. Module Description

4.1 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. 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

4.2 Social Cloud

Social network is a dynamic virtual organization with inherent trust relationships between friends. This dynamic virtual organization can be created since these social networks reflect real world relationships. It allows users to interact, form connections and share information with one another. This trust can be used as a foundation for information, hardware and services sharing in a Social Cloud.

4.3 Messenger

It is the client side of the social cloud, residing 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 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.

4.4 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.

4.5 Subscribe

In this module user can download the video. Subscribe module download video in high speed and clear video streaming. Authorized user every one download and watch the videos.

5. Process Diagram



- Only one high quality compressed video i stored
- No/Much less computations on motion estimation Fig: Process Diagram

6. Literature Survey

Literature survey is the most important step in software development process. Before developing the tool it is necessary to determine the time factor, economy n company strength. Once these things are satisfied, ten next steps are to determine which operating system and language can be used for developing the tool. Once the programmers start building the tool the programmers need lot of external support. This support can be obtained from senior programmers, from book or from websites. Before building the system the above consideration are taken into account for developing the proposed system.

7. System Requirement Specification (SRS)

- 1. Software Requirements Specification plays an important role in creating quality software solutions. Specification is basically a representation process. Requirements are represented in a manner that ultimately leads to successful software implementation.
- 2. Requirements may be specified in a variety of ways.
- 3. Representation format and content should be relevant to the problem
- 4. Information contained within the specification should be nested.
- 5. Diagrams and other notational forms should be restricted in number and consistent in use.
- 6. Representations should be revisable.

7.1 Functional Requirements

The system after careful analysis has been identified to be presented with the following requirements:

Requirement ID	Requirement Specification
CMOV_01	System should provide a provision to user one for registration.
CMOV_02	System should provide a provision for user one to login.
CMOV_03	System should provide a provision for user one to send request.
CMOV_04	System should provide a provision for user one to upload video.
CMOV_05	System should provide a provision for user one to view video.
CMOV_06	Systems should provide a provision for user to see information about video.
CMOV_07	Systems should provide a provision for user to get logout.
CMOV_08	Systems should provide a provision for second user to get login.
CMOV_09	Systems should provide a provision for second user to accept the friends request.
CMOV_10	Systems should provide a provision for second user to subscribe the video.
CMOV_11	Systems should provide a provision for second user to share comments.
CMOV_12	Systems should provide a provision for second user to get logout.

Table 1: Requirement ID and Specification

International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064 Impact Factor (2012): 3.358

7.2 Non-functional requirements

- 1. Usability
- 2. Reliability
- 3. Performance
- 4. Supportability
- 5. Implementation

8. System Design

Data Flow Diagram / Use Case Diagram / Flow Diagram

The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of the input data to the system, various processing carried out on these data, and the output data is generated by the system.





search Friends Send Reques Upload video Q Vuew Video USER1 user2 Share command Info abt video Accept reque Subscribe vide Figure 8.2: Use Case Diagram USERlogin() uplcadvideo *iewvideo* subscribevideo commends abt video viewinfoalt video send fid req accfrdreq Liter process() Login() Username Password checkvalid)



alid)



Figure 8.4: Activity Diagram

Volume 3 Issue 9, September 2014 <u>www.ijsr.net</u> Licensed Under Creative Commons Attribution CC BY



Figure 8.5: Sequence Diagram

9. System Testing

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

9.1 Types of tests

9.1.1 Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

9.1.2 Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

9.1.3 Functional test

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

- Valid Input: identified classes of valid input must be accepted.
- **Invalid Input:** identified classes of invalid input must be rejected.
- Functions: identified functions must be exercised.
- **Output:** identified classes of application outputs must be exercised.
- **Systems/Procedures**: interfacing systems or procedures must be invoked.

9.1.4 System Test

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

9.1.5 White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

9.1.6 Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

9.1.7 Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064 Impact Factor (2012): 3.358

9.1.8 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

9.1.9 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements

10. Results and Discussion

The following are the results obtained of our work. The figure 10(a) represents the sign up page of the new user. The figure 10(b) represents how to upload image for the profile picture. The figure 10(c) represents the friends list. The figure 10(d) represents how to comment on uploaded image. The figure 10(e) represents how to search the video. The figure 10(f) represents how to upload video. The figure 10(g) represents how to play uploaded video. The figure 10(h) represents how to comment on uploaded video.





Figure 10(c)



Figure 10(d)



Figure 10(e)



Figure 10(f)

Volume 3 Issue 9, September 2014 <u>www.ijsr.net</u> Licensed Under Creative Commons Attribution CC BY



Figure 10(g)



11. Conclusion

We conclude results prove the superior performance of CloudMov, in terms of transcoding efficiency, timely social interaction, and scalability. Here social mobile users can import a live or on-demand video to watch from any video streaming site, invite their friends to watch the video concurrently, and chat with their friends while enjoying the video and even u can search the video and can comment on the video.

12. Future Enhancement

In the current prototype, we do not enable sharing of encoded streams (in the same format/bit rate) among surrogates of different users. In our 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

- M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for vm-based cloudlets in mobile computing," IEEE Pervasive Computing, 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. of 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 INFOCOM'11, 2011, pp. 201 205.

- [4] T. Coppens, L. Trappeniners, and M. Godon, "AmigoTV: towards a social TV experience," in Proc. of EuroITV, 2004.
- [5] N. Ducheneaut, R. J. Moore, L. Oehlberg, J. D. Thornton, and E. Nickell, "Social TV: Designing for Distributed, Sociable Television Viewing," International Journal of Human-Computer 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. of USENIXATC, 2010.
- [7] What is 100% Pure Java, http://www.javacoffeebreak.com/faq/faq0006.html.
- [8] 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, January 2008.
- [9] K. Chorianopoulos and G. Lekakos, "Introduction to social tv: Enhancing the shared experience with interactive tv," International Journal of Human-Computer Interaction, vol. 24, no. 2, pp. 113–120, 2008.
- [10] M. Chuah, "Reality instant messaging: injecting a dose of reality into online chat," in CHI '03 extended abstracts on Human factors in computing systems, ser. CHI EA '03, 2003, pp. 926–927.
- [11] R. Schatz, S. Wagner, S. Egger, and N. Jordan, "Mobile TV becomes Social - Integrating Content with Communications," in Proc. of ITI, 2007.
- [12] R. Schatz and S. Egger, "Social Interaction Features for Mobile TV Services," in Proc. of 2008 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, 2008.
- [13] J. Flinn and M. Satyanarayanan, "Energy-aware adaptation for mobile applications," in Proceedings of the seventeenth ACM symposium on Operating systems principles, ser. SOSP '99, 1999, pp. 48–63.
- [14] W. Yuan and K. Nahrstedt, "Energy-efficient soft realtime cpu scheduling for mobile multimedia systems," in Proceedings of the nineteenth ACM symposium on Operating systems principles, ser. SOSP '03, 2003, pp. 149–163

Author Profile



Ambika received Bachelor of Engineering degree from Poojya Doddappa Appa College of Engineering Gulbarga Karnataka India affiliated to VTU Belgaum and pursuing Master of Technology (M.Tech) (2012-2014) in Computer science and Engineering from

Arjun College of Technology and Sciences Hyderabad India affiliated to JNTU Hyderabad India. Cloud Computing is the field of interest.



Ramya Bathula is presently working as Associate Professor in Department of Computer Science and Engineering, Arjun College of Technology And Sciences, Batasingram, Hayath nagar, R. R. Dist. She received her B. Tech degree in Computer Science and

Engineering from Swami Ramanand Thirde Institute of Science & Technology in 2008, M. Tech degree in Information Technology from ASTRA Aurora Scientific Technology & Research & Academy in 2010.