Sangamesh Int. Journal of Engineering Research and Applications www.ijera.com
ISSN : 2248-9622, Vol. 4, Issue 7(Version 1), July 2014, pp.167-170

RESEARCH ARTICLE OPEN ACCESS

Delay Efficient Method for Delivering IPTV Services

Sangamesh[1], Shilpa. K. Gowda[2]
[1] ECE department, SJB Institute of technology, Bangalore
[2] Asst. Professor, ECE department, SJB Institute of technology, Bangalore

Abstract
Internet Protocol Television (IPTV) is a system through which Internet television services are delivered using the architecture and networking methods of the Internet Protocol Suite over a packet-switched network infrastructure, e.g., the Internet and broadband Internet access networks, instead of being delivered through traditional radio frequency broadcast, satellite signal, and cable television (CATV) formats. IPTV provides mainly three services: live TV, catch up TV, and video on demand (VoD).This paper focuses on delivering the live TV services by exploiting the virtualised cloud architecture of the IPTV and statistical multiplexing. The VoD tasks are prescheduled so that there will be less Instant Channel Change (ICC) delay. We select a proper scheduling algorithm for rescheduling the VoD tasks. We then implement the scheduling algorithm for pre-shifting the VoD tasks.
Key Terms: IPTV, Statistical multiplexing, Virtualization, Vo D, Live TV, Scheduling, ICC Delay

I. INTRODUCTION

As the demand for Internet-based applications grows around the world, Internet Protocol Television (IPTV) has been becoming very popular. With the recent advances in communication and computer technology, television has gone through many changes over the years. Nowadays IP based video delivery became more popular (IPTV).

Internet protocol television is defined as a multimedia services such as television/video/audio/text/graphics/data delivered using the internet protocol suite over IP based networks managed to provide the required level of quality of service and experience, security, interactivity and reliability like- Internet, instead of being delivered through traditional satellite signal and cable TV formats. IPTV means delivering enhanced video applications over a managed or dedicated network via Internet Protocol. In IPTV service, this technology is used as that of Internet Services. In this service the TV channels are encoded in IP format and delivered to TV using a Smart Electrical Electronic Device. The IP TV Service also includes Video on Demand cloud services which are similar to watching Video CDs / DVDs using a VCD / DVD/CD player. Movies, different channels, Instructional Videos and other content shall be available to customers in the IP TV Services. This IPTV is through a broadband connection. IPTV is not video over the public Internet

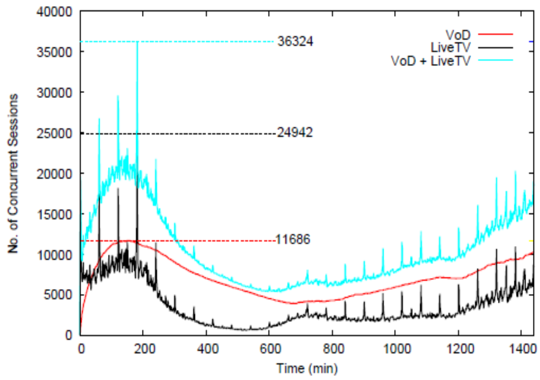

Fig1.LiveTV ICC VoD Concurrent sessions vs time, ICC burst seen every half hour

IPTV services can be classified into three main groups a) Live Television: with or without interactivity related to the current TV show, b) Time shifted Television: catch up TV (replays a TV show that was broadcast hours or days ago), start–over TV (replays the current TV shows from its beginning) and c) Video-On-Demand (VoD): browse a catalog of videos, not related to TV programming. In IPTV, Live TV is typically Multicast from servers using IP Multicast; with one group per TV channel, there are typically several hundred Channels. The consumer’s Set-Top Box “tunes” to a particular TV “channel” by joining the Multicast group for that channel. Video-on- Demand (VoD) is also supported by the service provider, with each request being served by a server using a unicast stream.

In recent years, cloud storage service has become a faster profit growth point by providing a comparably low-cost, scalable, position-independent platform for client’s data. Cloud computing has recently changed the landscape of Internet based computing,Whereby a shared pool of configurable computing resources (networks, servers, storage) can be rapidly provisioned and released to support multiple services within the same infrastructure. Due to its nature of serving computationally intensive applications, cloud infrastructure is particularly suitable for content delivery applications. Typically LiveTV and VoD services are operated using dedicated servers.

Cloud based IPTV works on the principal of on demand delivery and real time scheduling in which there is a pool for all the resources which are allocated from Cloud service providers. Virtualized cloud-based services can take advantage of statistical multiplexing across applications to yield significant cost savings to the operator. However, achieving similar benefits with real-time services can be a challenge.

In virtualized environment, ICC is managed by a set of VMs. The number of such VMs created would be driven by the predictor (note that a (small) number of VMs would typically be assigned to each distinct channel).Similarly, for the VoD service, the cloud configures a number of VMs based on the currently active VoD sessions, and will adapt to meet user demand. When a physical server complex is shared for these services, it is desirable to minimize the total number of VMs deployed (thereby the resources used) to satisfy all these requests. The provisioning approach described above effectively uses virtualization to achieve this minimization of resource usage.

II. Typical IPTV Architecture

This section gives a brief understanding about the working of typical IPTV architecture. This architecture is used for providing the different services.

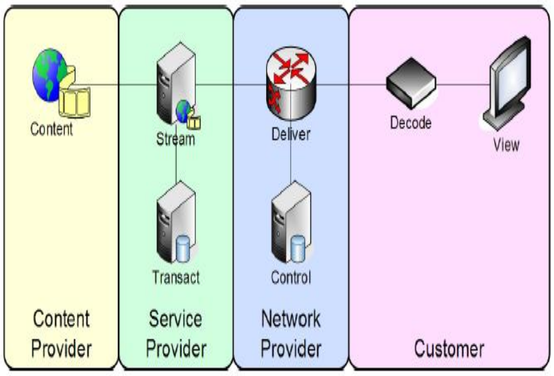

Fig 2 Typical IPTV Architecture

The main elements of IPTV are:

- TV Head-end: where live TV channels are encoded, encrypted and delivered in the form of IP multicast streams.
- VOD platform: where on-demand video assets are stored and served when a user makes a request in the form of IP unicast stream.
- Interactive portal: allows the user to navigate within the different IPTV services, such as the VOD catalog.
- Delivery network: the packet switched network that carries IP packets (unicast and multicast).
- Home gateway: the piece of equipment at the user's home that terminates the access link from the delivery network.
- User's set-top box: the piece of equipment at the user's home that decodes and decrypt TV and VOD content and displays it on the TV screen.

These are the most commonly used elements for delivering the IPTV services, namely LiveTV and Video on demand.

III. Methodology

We consider the deadlines associated with the Live TV and VoD. Live TV have time constraint but VoD can be delayed or advanced .We take advantage of virtualization and statistical multiplexing which are implemented in the cloud based IPTV architecture.

We studied different scheduling algorithms for pre scheduling the VoD tasks which have to be serviced at the burst period.

III.a Virtualization of cloud resources

Virtualization gives the ability to run multiple operating systems on a single physical system and share the underlying hardware resources.Virtualization plays a major role in the cloud computing technology, normally in the cloud computing, users share the data present in the clouds like application etc, but with virtualization users shares the Infrastructure. the main usage Virtualization Technology is ,Normally cloud providers provide the applications with the standard versions to their cloud users, for suppose if the next version of that application is released, then cloud provider has to provide the latest version to their cloud users and practically it is possible but it is more cost expensive. By using virtualization, all severs and the software application which are required by other cloud providers are maintained by the third party people, and the cloud providers has to pay the money on monthly or annual basis.

III.b Statistical Multiplexing

Two forms of multiplexing are commonly used today: time-division multiplexing and statistical multiplexing. Time-division multiplexing is providing a fixed amount of bandwidth for each incoming stream. The packets from each incoming stream are placed into one or more timeslots in the combined stream. Generally, this allocation can be adjusted to accommodate streams that require variable bandwidth, or in systems where the allocation cannot be changed rapidly or while the system is in use.

Statistical multiplexing allocates bandwidth to input channels in response to their needs; high-speed channels receive a larger amount of the overall network capacity. The systems can be configured with a maximum and a minimum bit rate for each tributary stream. Virtualized cloud based service can take advantage of statistical multiplexing across applications.

III.c Earliest Deadline First

Earliest deadline first (EDF) or least time to go is a dynamic scheduling algorithm used in real-time operating systems to place processes in a priority queue. Whenever a scheduling event occurs (task finishes, new task released, etc.) the queue will be searched for the process closest to its deadline. This process is the next to be scheduled for execution. EDF is an optimal scheduling algorithm on preemptive uniprocessors, in the following sense: if a collection of independent jobs, each characterized by an arrival time, an execution requirement and a deadline, can be scheduled (by any algorithm) in a way that ensures all the jobs complete by their deadline, the EDF will schedule this collection of jobs so they all complete by their deadline. With scheduling periodic processes that have deadlines equal to their periods, EDF has a utilization bound of 100%.

EDF can guarantee that all deadlines are met provided that the total CPU utilization is not more than 100%. Compared to fixed priority scheduling techniques like rate-monotonic scheduling, EDF can guarantee all the deadlines in the system at higher loading.

However, when the system is overloaded, the set of processes that will miss deadlines is largely unpredictable (it will be a function of the exact deadlines and time at which the overload occurs.) This is a considerable disadvantage to a real time systems designer. The algorithm is also difficult to implement in hardware and there is a tricky issue of representing deadlines in different ranges (deadlines must be rounded to finite amounts, typically a few bytes at most). If a modular arithmetic is used to calculate future deadlines relative to now, the field storing a future relative deadline must accommodate at least the value of the (("duration" {of the longest expected time to completion} * 2) + "now"). Therefore EDF is not commonly found in industrial real-time computer systems.

Instead, most real-time computer systems use fixed priority scheduling (usually rate-monotonic scheduling). With fixed priorities, it is easy to predict that overload conditions will cause the low-priority processes to miss deadlines, while the highest-priority process will still meet its deadline. There is a significant body of research dealing with EDF scheduling in real-time computing; it is possible to calculate worst case response times of processes in EDF, to deal with other types of processes than periodic processes and to use servers to regulate overloads.

In this paper we use this algorithm to properly schedule the VoD tasks based on the deadlines associated with each task.

IV. Implementation

We implement the EDF algorithm for rescheduling the VoD tasks. We use MATLAB tool for simulating the project. The results are shown in the next section.

V. Results and Discussions

The figure below shows the tasks before applying the EDF algorithm and after applying the algorithm.

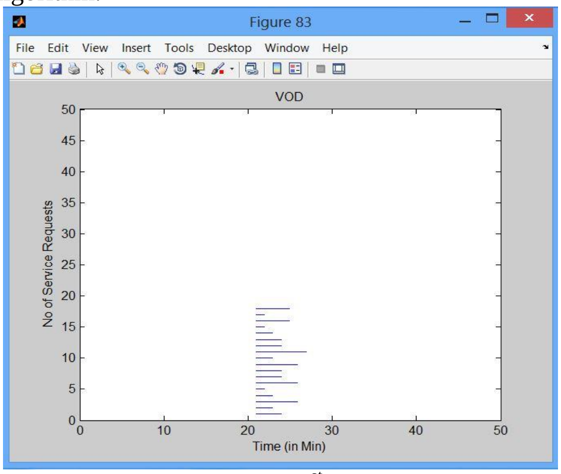

Fig 3. VoD frame at the 21st time instant, before applying the EDF algorithm

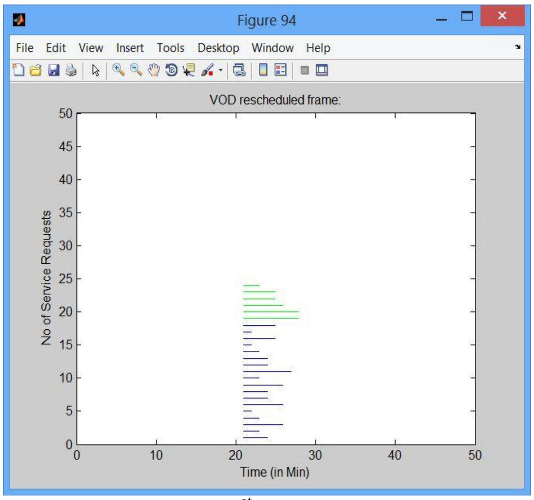

Fig 4. VoD frame at 21st time instant,after applying the EDF algorithm

The blue lines indicate the VoD tasks at that time instant. The green lines indicate the rescheduled tasks which earlier were scheduled at 29th and 30th time instant. Here we have shown only frames for the 21st time instant.

VI. Conclusion

By the results shown above we can see that the VoD tasks can be shifted or scheduled so as to reduce the ICC delay. We can in turn can use the resources efficiently by the use of rescheduling the tasks. Still different methods can be implemented for reducing the ICC delay.

VII. Acknowledgment

I would like to articulate my profound gratitude and indebtedness to my project guide Shilpa K Gowda. (Assoc.Prof, Dept. of ECE, SJBIT) who is always being a constant motivation and guiding factor throughout the project time in and out as well. It has been a great pleasure for me to get a opportunity to work under her.

References

[1] Vaneet Aggarwal, Vijay Gopalakrishnan, Rittwik Jana, K. K. Ramakrishnan, and Vinay A. Vaishampayan," *Optimizing Cloud Resources for Delivering IPTV Services through Virtualization*", In Proc. IEEE Int. Conf. Communication Systems and Networks (COMSNETS), Jan. 2012.

[2] V.Aggarwal, V.Gopalakrishnan, R.Jana, K. K. Ramakrishnan, and V. Vaishampayan, "*Exploiting Virtualization for Delivering Cloud-based IPTV Services*," in *Proc. of IEEE INFOCOM (mini-conference)*, Shanghai, April 2011.

[3] D. Banodkar, K. K. Ramakrishnan, S. Kalyanaraman, A. Gerber, and O. Spatscheck, "*Multicast instant channel change in IPTV system,*" in Proceedings of IEEE COMSWARE, January 2008.

[4] H. A. Lagar-Cavilla, J. A. Whitney, A. Scannell, R. B. P. Patchin, S. M.Rumble, E. de Lara, M. Brudno, and M. Satyanarayanan, "*SnowFlock:Virtual Machine Cloning as a First Class Cloud Primitive*," ACM Transactions on Computer Systems (TOCS), 2011.

[5] A. Dan, D. Sitaram, and P. Shahabuddin, "Scheduling Policies for an On-Demand Video Server with Batching," in Proc. of ACM Multimedia, San Francisco, CA, October 1994, pp. 15–23.

[6] K. K. Ramakrishnan, R. Doverspike, G. Li, K. Oikonomou, and D. Wang, "*IP Backbone Design for Multimedia Distribution: Architecture and Performance,*" May 2007, proceedings of IEEE INFOCOM.

[7] R. Urgaonkar, U. Kozat, K. Igarashi, and M. J. Neely, "*Dynamic resource allocation and power management in virtualized data centers,*" in Proceedings of IEEE IFIP NOMS, March 2010.

[8] C. L. Liu and J. W. Layland, "*Scheduling Algorithms for Multiprogramming in a Hard Real Time Environment,*" Journal of the ACM, vol. 20, no. 1, pp. 46–61, 1973