International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 2 Issue 12, December - 2013

Implementing and Optimizing cloud resources for IPTV Services Through Virtualized IPTV Architecture

S Prakash Babu[1], Y. Gangadhar[2]

M.Tech student, Dept of CSE, Kuppam Engineering college[1],

Assoc.prof, Dept of CSE, Kuppam Engineering College[2],

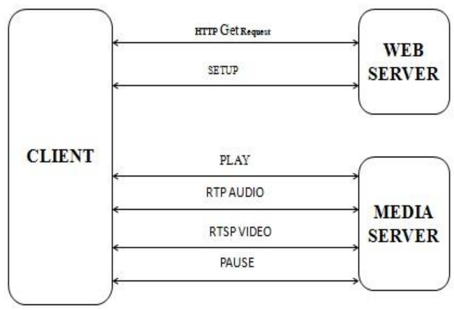
Abstract:

Cloud computing playing vital role in present Research. In this project we are implementing real time IPTV services through a virtualized IPTV architecture. Here, we are applying differences in the deadline associated with live TV versus video-on-demand (vod) to effectively multiplex these services through intelligent time shifting of service delivery. The generalized framework for computing the amount of resource needed to support multiple services without missing the deadline for any service. Here we are constructing optimization formulation generic cost function as concave function to reflect the different pricing options. This formulation gives the number of servers needed at different time instants to support these services. We implementing simple mechanism for time-shifting scheduling jobs in a simulator by virtualized cloud based services and it will take advantage as statistically multiplexing across applicants to significant cost saving to the operator.

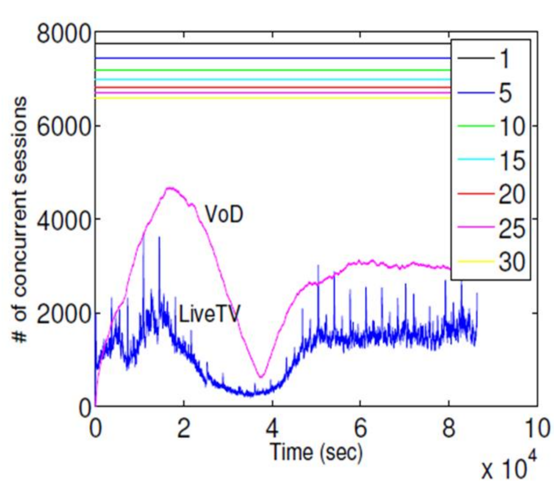
I. Introduction:

Cloud computing is a new infrastructure environment that delivers on the promise of supporting on-demand services. In IPTV services, Live TV is typically multicast from servers using IP Multicast, with one group per TV channel. Where unicast stream used to delivery video-on-demand by the service operator for each request being served by a server. When users change channels while watching live TV, For each time channel change, the user should join the multicast group associated with the channel, and wait for required data to be buffered before the video is displayed; this can take some amount of time. As a result, there have been more attempts to support instant channel change by mitigating the user perceived channel switching latency [1], [2]. With the typical ICC implemented on IPTV systems, unicast stream from the server delivery the content at an accelerated rate. the switching latency will be small when The playout buffer is filled quickly. Once the playout buffer is filled up to the playout point again the set top box reverts back to receiving the multicast stream.

A typical IPTV architecture



In this paper, we aim to take advantage of the difference in workloads of many IPTV services to better utilize the deployed servers. For example, when ICC workload is very bursty with a large peak to average ratio and VoD has a relatively steady load and imposes "not so stringent" delay bounds. It offers opportunities for the service provider to deliver the video-on-demand data in anticipation and potentially out-of-order. We seek to minimize the resource requirements for supporting the IPTV service by taking advantage of statistically multiplex across the different services.



concurrent sessions needed with a 30 sec hold time for ICC - Operational trace

In a virtualized environment, ICC is managed by a set of VMs (typically, a few VMs will be used to serve a different popularity channels). Other VMs would be created to handle VoD requests. With the ability to spawn VMs quickly [3], we believe we can shift servers (VMs) from VoD to handle the ICC demand in a matter of a few seconds. Note that by being able to predict the ICC bursts (channel change behavior can be predicted from historic logs as a result of LiveTV show timings. The channel changes usually occurs for every half hour). In anticipation of the ICC load, we provide accelerate delivery of VoD content to users' STBs and shift the VoD demand away from the ICC burst interval. This will ensure that VoD users will not notice impairment in their delivered quality of service as the playout can be from the local STB cache. We also show that for any server tuple with integer entries inside the server-capacity region, an earliest deadline first (EDF) strategy can be used to serve all requests without missing their deadlines. This is an extension of previous results in the literature where the number of servers are fixed at all times [4]. The server-capacity region is formed by linear constraints, and thus this region is a polytope.

II. Proposed work:

In this paper the IPTV service such as LiveTV, VoD ,Graphics, VoIP, Text etc are delivered using the Internet. The combination of IPTV, VoIP and Internet access services is known as Triple play. The combination of triple play and mobile voice services leads to the Quadruple. Where these services are delivered by corporate LANs and Business Networks.

Advantages

1. User easily buys the channel using internet.
2. User receives the signal from set-top box.
3. No wires needed.

Disadvantages

1. More cost and limited usage only.
2. Some time make a signal problem in unconditional weather.

MODULE DESCRIPTION:

1. **Deadline Constraints and Scheduling**
2. **User Complaint**
3. **Optimization**

1. Deadline Constraints and Scheduling

Each channel pack has some scheduling and deadline constraints . The deadline Constraints Provide the limited period of time to the channels pack. User using the Channels packs within Period. Suppose your channel period time is finished that time Automatically you loss the channel Pack. Also admin provided alert message to user two Days before in channel pack period using Deadline constraints.

Types of constraints:

1. Flexible constraints. This is a default type of constraint in project. It means that a task can start As Soon As Possible

2. Semi-flexible constraints. A task must begin or end on later than the defined date

3. Inflexible constraint. A task must begin or end on a certain date.

2. User Complaint

In this module we give complaint to the complaint box and post the complaint. Then admin view the complaint then take the action to that complaint. Finally users view that complaint status.

Along with this we are providing default complaints which helps to understand easily by consumer and administrator to avoid confuse.

3. Optimization

In the module user select the cheap and best channel pack. In this project optimization there are three methods.

Linear Cost Function:

In this cost function the cost is directly propositional to the total number of resources such that cost function increases as the number of servers increases .

Piecewise Linear Convex Cost Function:

In this cost function at a particular number of servers the cost increases as the number of servers increases leads to linear cost function.

Exponential Cost Function:

In this cost function the cost decreases as the number of server increases .

III Result and analysis:

Where new user can get register by providing his details. If the user is already registered then he can sign in by his

username and password.

The logged in user can buy his interested period channel pack. Where each channel pack has different period and cost. The user can buy video files of his interested.

The user can watch the channels or video file. Where he can also provide comments and complaints which can be viewed by the administrator later.

The admin authorized to insert the number channels which can be viewed by customer. He can access the channel booking details of the particular users. he able to change the video files depends of the particular user complaints.

IV. Conclusions:

In this project we are using intelligent time-shifting of load to better utilize deployed resources for Instant Channel Change and VoD delivery. Here concluded that an in future enhancement that we are providing this IPTV services through social networking to improve performance and security while keeping standard and simple procedures and protocols.

.

V. References:

[1] 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.

[2] "Microsoft tv: Iptv edition," http://www.microsoft.com/tv/IPTVEdition.mspx.

[3] 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.

[4] J. A. Stankovic, M. Spuri, K. Ramamritham, and G. C. Buttazzo, *Deadline Scheduling for Real-Time Systems: Edf and Related Algorithms*. Norwell, MA, USA: Kluwer Academic Publishers, 1998.

VI. Author Profiles:

S.PRAKASH BABU received the B.Tech Degree in **Information Technology** from Jawaharlal Nehru University, Anantapur in 2011. Pursing M.Tech(CSE) in Dept of Computer Science&Engineering **Kuppam Engg College, Kuppam** under Jawaharlal Nehru University, Anantapur.