

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 2, February 2019

Resolving Online Video Replication and Placement Issues Using Effective Resource-Saving File Management Scheme

Suraj B. Patil, Prof. Santosh T. Waghmode

M. E Student, Department of Computer Engineering, Imperial College of Engineering and Research, Pune.

Savitribai Phule Pune University, Pune, India

Department of Computer Engineering, Imperial College of Engineering and Research, Pune,

Savitribai Phule Pune University, Pune, India

ABSTRACT: Content delivery networks (CDNs) have been generally executed to give versatile cloud administrations. Such networks promote resource pooling by granting virtual machines or physical servers to be dynamically activated and deactivated as stated in current client request. This paper explores online video replication and placement issues in CDNs. An effective video provisioning plan should at the same time (i) use system resources to reduce total energy consumption and (ii) restrain replication overhead. We propose a scheme called adaptive data placement (ADP) that can dynamically place and reorganize video replicas among cache servers on subscribers' arrival and departure. Both the examinations and simulation results show that ADP can decrease the quantity of activated cache servers with constrained replication overhead. Furthermore, ADP's performance is approximate to the optimal solution.

KEYWORDS: Content Delivery Networks, Resource Management, Video Streaming, Cache Server, Adaptive Data Placement

I. INTRODUCTION

In recent years, content delivery networks (CDNs) services have been widely implemented to provide scalable cloud. Fig. 1 shows a typical local CDN whose servers are located in the same place. The request of each visitor (A) is first processed and identified by a gateway server (B) and then directed to a cache server (CS) (C) in the server farm. CSs are typically virtual machines and can dynamically provide various services by executing different contents loaded from a backhaul database (D).

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 2, February 2019

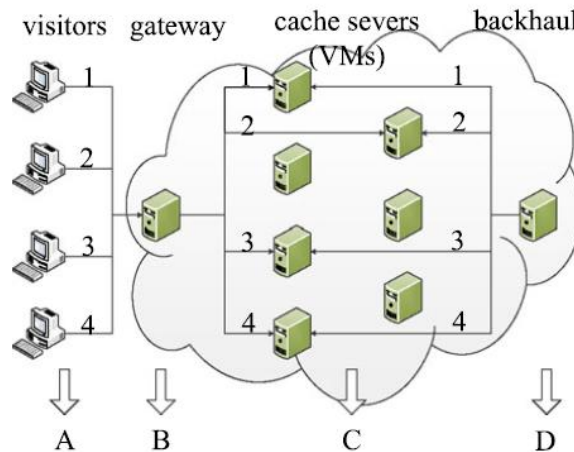


Fig. 1 Illustration of a local CDN

When a visitor arrives and requests a video clip, the system must assign a serving CS and copy a replica of the clip from the backhaul database if the CS does not cache the clip for other visitors. Because each CS has limited capability, the total number of video clips it stores and the total outgoing bandwidth of subscribers it bears are limited by its space and bandwidth constraints. Fig. 2 illustrates an example of this two-dimensional resource allocation problem, where the length and width respectively express the space and bandwidth capacity of a CS, and u and p respectively illustrate users and their requested programs.

Here, Visitors 1, 2, and 3 require Video Programs A, B, and C respectively. When User 4 is later directed to this server, an additional unit of bandwidth is required because each video is independently transmitted played by each user. However, Visitor 4 does not occupy an additional unit of storage space because Program A has already been requested by Visitor 1. Because of the many time-variant requirements of video clips, intelligently placing videos among CSs and determine their serving subscribers without violating capacity and bandwidth limits is challenging.

Typical CDN management schemes in data centers fail to address this video provisioning problem for three reasons:

1. First, rather than being separately required by a specific user/subscriber, a video clip may be simultaneously accessed by numerous online subscribers.

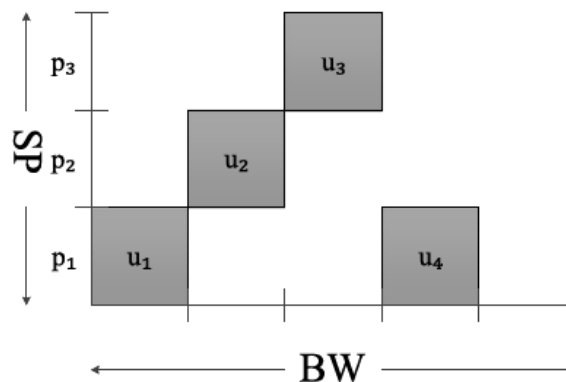


Fig. 2 Example of a CS in a CDN

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 2, February 2019

2. Second, the two resource requirements (i.e., bandwidth and storage size) exhibit different characteristics (i.e., when different visitors request the same clip, their storage requirements can be combined, whereas their bandwidth requirements are independent).
 3. Third, because video placements and server selections are conducted online as visitors arrive and depart; the migration/management of clip overhead should be limited to provide real-time responsiveness.
- Therefore, a new resource management scheme that is specifically designed for provisioning online videos in CDNs is required.

Motivation:

This paper introduces a new problem called resource-saving video placement (RSVP) and proposes a scheme called adaptive data placement (ADP). Through analysis and simulations, we demonstrate the two main advantages of ADP: (i) the worst-case performance difference between ADP and the optimal solution can be guaranteed, and (ii) the replication overhead on each arrival or departure of a visitor is limited. Because ADP is based on common assumptions, it can be applied to various types of CDNs to improve their resource and power efficiency.

Objectives:

1. Position the concept of CDN associating by providing a systematic understanding and categorization of the CDN space in terms of applications, features, and implementation techniques.
2. Provide a novel architecture for CDN associating to enable resource sharing among multiple CDNs and demonstrate the effectiveness of CDN associating.
3. Investigate enabling techniques, i.e. resource discovery, server selection, and request-redirection, in relation to CDN associating.
4. Investigate the QoS-oriented aspect of request servicing via a utility model for CDN associating and demonstrate its applicability in practice.

II. REVIEW OF LITERATURE

The paper [1] represents iAware, a lightweight interference-aware VM live migration strategy. It analytically captures the essential relationships between VM performance interference and key factors that are practically accessible through realistic experiments of benchmark workloads on a Xen virtualized cluster platform. iAware together estimates and minimizes both migration and co-location interference among VMs, by designing a simple multi-resource demand-supply model. Major experiments and complementary large-scale simulations are conducted to validate the performance gain and runtime overhead of iAware in terms of I/O and network throughput, CPU consumption, and scalability, compared to the traditional interference-unaware VM migration approaches.

In the [2] paper review the state-of-the-art research on managing the performance overhead of VMs, and summarize them under diverse scenarios of the IaaS cloud, ranging from the single-server virtualization, a single mega datacenter, to multiple geodistributed datacenters. Specifically, we unveil the causes of VM performance overhead by illustrating representative scenarios, discuss the performance modeling methods with a particular focus on their accuracy and cost, and compare the overhead mitigation techniques by identifying their effectiveness and implementation complexity.

Video conferencing, even with its stringent delay constraints, should also be provided as a cloud service, taking full advantage of the inter-datacenter network in the cloud. We design Airlift [3], a new protocol designed for the inter-datacenter network, tailored to the needs of a cloud-based video conferencing service. Airlift delivers packets in live video conferences to their respective destination datacenters, with the objective of maximizing the total throughput across all conferences, yet without violating end-to-end delay constraints. In order to simplify our protocol design in Airlift, we use intersession network coding and the concept of conceptual flows, such that the optimization problem that can be conveniently formulated as a linear program. Our real-world implementation of Airlift has been deployed over the Amazon EC2 cloud.

By improving the placement of VMs on host machines, traffic patterns among VMs can be better aligned with the communication distance between them, e.g. VMs with large mutual bandwidth usage are refer to host machines in close proximity. We formulate the VM placement as an optimization problem and prove its hardness. We design a two-tier approximate algorithm [4] that efficiently solves the VM placement problem for very large problem sizes. Given

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 2, February 2019

the significant difference in the traffic patterns seen in current data centers and the structural differences of the recently proposed data center architectures, we further conduct a comparative analysis on the impact of the traffic patterns and the network architectures on the potential performance gain of traffic-aware VM placement.

Consequently, an application's environmental impact can vary significantly depending on the geographical distribution of end-users, as electricity cost and carbon footprint per watt is location specific. In [5] paper work, we describe FORTE: Flow Optimization based framework for request-Routing and Traffic Engineering. FORTE dynamically controls the fraction of user traffic directed to each datacenter in response to changes in both request workload and carbon footprint. It allows an operator to navigate the three-way tradeoff between access latency, carbon footprint, and electricity costs and to determine an optimal datacenter upgrade plan in response to increases in traffic load.

Previous works are limited on cutting down the power consumption of the datacenters to defuse such a concern. The [6] paper represents how the spatial and temporal variabilities of the electricity carbon footprint can be fully exploited to further green the cloud running on top of geographically distributed datacenters. We jointly consider the electricity cost, service level agreement (SLA) requirement, and emission reduction budget. To navigate such a three-way tradeoff, we take advantage of Lyapunov optimization techniques to design and analyze a carbon-aware control framework, which makes online decisions on geographical load balancing, capacity right-sizing, and server speed scaling.

With the presence of fluctuating workloads in datacenters, the lifetime and reliability of servers under dynamic power-aware consolidation could be adversely impacted by repeated on-off thermal cycles, wear-and-tear and temperature rise. In [7] paper, we propose a Reliability-Aware server Consolidation strategy, named RACE, to address when and how to perform energy-efficient server consolidation in a reliability-friendly and profitable way. The focus is on the characterization and analysis of this problem as a multi-objective optimization, by developing a utility model that unifies multiple constraints on performance SLAs, reliability factors, and energy costs in a holistic manner. An improved grouping genetic algorithm is proposed to search the global optimal solution, which takes advantage of a collection of reliability-aware resource buffering, and virtual machines-to-servers re-mapping heuristics for generating good initial solutions and improving the convergence rate.

An analytical framework for characterizing and optimizing the power-performance tradeoff in Software-as-a-Service (SaaS) cloud platforms. Our objectives are two-fold: (1) We maximize the operating profit when serving heterogeneous SaaS applications with unpredictable user requests, and (2) we minimize the power consumption when processing user requests. To achieve these objectives, we take advantage of Lyapunov Optimization [8] techniques to design and analyze an optimal control framework to make online decisions on request admission control, routing, and virtual machine (VMs) scheduling. In particular, our control framework can be flexibly extended to incorporate various design choices and practical requirements of a datacenter in the cloud, such as enforcing a certain power budget for improving the performance (dollar) per watt.

The paper [9] introduces a dynamic capacity management policy, Auto Scale that greatly reduces the number of servers needed in data centers driven by unpredictable, time-varying load, while meeting response time SLAs. Auto Scale scales the data center capacity, adding or removing servers as needed. Auto Scale has two key features: (i) it autonomously maintains just the right amount of spare capacity to handle bursts in the request rate; and (ii) it is robust not just to changes in the request rate of real-world traces, but also request size and server efficiency. We evaluate our dynamic capacity management approach via implementation on a 38-server multi-tier data center, serving a web site of the type seen in Face book or Amazon, with a key-value store workload. We demonstrate that Auto Scale vastly improves upon existing dynamic capacity management policies with respect to meeting SLAs and robustness.

As virtual machines dynamically enter and leave a cloud system, it becomes necessary to relocate virtual machines among servers. However, relocation of virtual machines introduces run-time overheads and consumes extra energy, thus a careful planning for relocation is necessary. We model the relocation problem as a modified bin packing problem and propose a new server consolidation algorithm [10] that guarantees server consolidation with bounded relocation costs. We also conduct a detailed analysis on the complexity of the server consolidation problem, and give an upper bound on the cost of relocation. Finally, we conduct simulations and compare our server consolidation algorithm with other relocation methods, like First Fit and Best Fit method. The experiment results suggest an interesting trade-off between server consolidation quality and relocation cost.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 2, February 2019

Content Delivery Networks (CDNs) are effective platforms for providing various types of services. Among them, on-demand video provisioning is a popular application, allowing numerous users to arbitrarily request videos from a massive database. Video websites such as YouTube, Vimeo, and DailyMotion are examples. When a visitor arrives and requests a video clip, the system must assign a serving CS and copy a replica of the clip from the backhaul database if the CS does not cache the clip for other visitors. Because each CS has limited capability, the total number of video clips it stores and the total outgoing bandwidth of subscribers it bears are limited by its space and bandwidth constraints.

Disadvantages of Existing System:1. Because of the many time-variant requirements of video clips, intelligently placing videos among CSs and determine their serving subscribers without violating capacity and bandwidth limits is challenging. Typical CDN management schemes in data centers fail to address this video provisioning problem.

III. PROPOSED SYSTEM

Operating a cloud-based CDN on these sites requires 3 kinds of resources: bandwidth, VM, and storage. Accordingly leasing the resources incurs 3 kinds of costs: bandwidth cost, VM rental cost, and storage cost. For simplicity, we assume that all contents are of the same size. This is also a reasonable assumption since servers divide videos into small portions with the same size for the convenience of caching. The request for a long video can also be divided into many requests for small portions. This means that the portions for a video can independently exist in different servers and can be independently served to users.

This paper introduces a new problem called resource-saving video placement (RSVP) and proposes a scheme called adaptive data placement (ADP). Through analysis and simulations, we demonstrate the two main advantages of ADP: (i) the worst case performance difference between ADP and the optimal solution can be guaranteed, and (ii) the replication overhead on each arrival or departure of a visitor is limited. Because ADP is based on common assumptions, it can be applied to various types of CDNs to improve their resource and power efficiency.

To increase the readability of the pseudo code, the updating processes of the following variables are not contained in the details of ADP: Ps, Sp, BWs, SPs, CN(s, p). These parameters can be updated based on their definitions after a subscription is added to or removed from a CS. The only exception is sOPS, which ADP must determine and change during execution. ADP is composed of two main functions: ARRIVE and DEPART, which are respectively executed when a subscription enters and leaves a system. The ARRIVE process provides fast responsiveness because the initiation of a new CS (i.e., sOPS) occurs only after, not on, a subscription's arrival. Any incoming subscription can be placed into an activated CS (SPF or the OPS) in real-time if the arrival and departure processes of previous subscriptions are completed. The DEPART procedure, which reorganizes the replica placement in CSs when a subscription u leaves the system. These two subscriptions then swap their serving CSs to enable the video clip to depart in this desired priority. The rationale here is that (i) the CSs of higher bandwidth and space utilization should be left as unchanged as possible, and (ii) less operation is required if a subscription leaves from the OPS CS.

Advantages of Proposed System:

1. To achieve high resource utilization, our proposed scheme, ADP, follows three principles: (i) it maintains only one OPS server in a system to enable most CSs to achieve at least one aspect (i.e., bandwidth or space) of full utilization; (ii) it maintains the exclusiveness of video clips (i.e., allows at most one replica for each clip) among the OPS and SPF servers to improve space efficiency, which we demonstrate in the next section; and (iii) it conducts less physical replication to limit overhead.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 2, February 2019

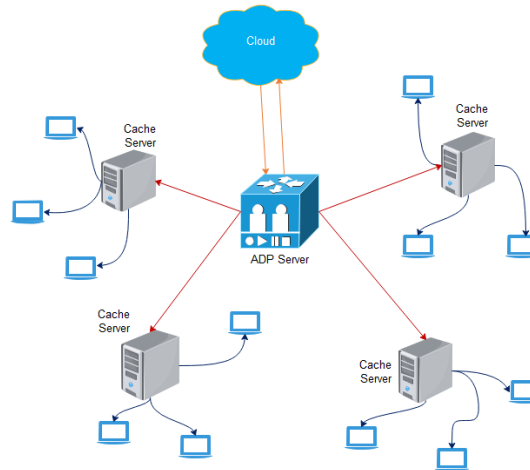


Fig.3 Proposed Architecture

IV. MATHEMATICAL MODEL

1. ARRIVE Algorithm

ARRIVE(u): decides the serving CS s_u of an incoming subscription requesting clip p_u

1: if $(\exists \text{ CS } s: s \in S^{SPF} \text{ and } P_s \ni p_u)$ { //find a SPF CS that contains p_u

2: $s_u \leftarrow s$;

3: }else{ //all SPF CSs do not contain p_u

4: $s_u \leftarrow s_{OPS}$; //add the subscription into the OPS server

5: if $(p_u \notin P_{s_{ops}}) P_{s_{ops}} \leftarrow P_{s_{ops}} + \{p_u\}$; //copy program p_u from backhaul

6: if $(BW_{s_u} == 0 \text{ or } SP_{s_u} == 0) s_{OPS} \leftarrow \text{new server}$; }

2. DEPART Algorithm

DEPART(u): reorganizes the file placement of CSs when a subscription u leaves the system

1: //find a subscription u' which takes over u's position

2: if $(\exists \text{ subscription } u': p_{u'} = p_u \text{ and } s_{u'} = s_{OPS})$ break;

3: else if $(\exists \text{ subscription } u': p_{u'} = p_u \text{ and } s_{u'} = s^{BWF})$ break;

4: else if $(\exists \text{ subscription } u': p_{u'} = p_u \text{ and } s_{u'} = s^{SPF})$ break;

5: else $u' \leftarrow u$;

6: swap $s_u, s_{u'}$;

7: $S_{TYPE} \leftarrow \text{type_of}(s_u)$; //save the type of s_u before leaving

8: disconnect u from s_u ;

9: //Manage s_u according to its S_{TYPE}

10: switch S_{TYPE} {

11: case OPS: break;

12: case BWF: leavefromBWF(s_u); break;

13: case SPF: leavefromSPF(s_u); break;

14: case FUL: leavefromFUL(s_u); break;

15: }

16: /* check OPS state */

17: if $(BW_{s_u} SP_{s_u} > 0)$ { // s_u has both available bandwidth and space

18: deactivates s_{OPS} ;

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 2, February 2019

```
19:  $s_{OPS} \leftarrow s_u$ ;  
20: }
```

V. EXPERIMENT RESULT

Experimental evaluation showing the Resource-saving file management scheme for online video provisioning on Content Delivery Networks (CDNs). This project is implemented with the showing simulation with the help of Java RMI framework. In this simulation shown 1 ADP Server, Multiple Cache Servers and list of users. Before running the ADP server, we need get the CSs from “backup” folder. These CSs are having video files. That ADP server displays the how many CSs are available in the system and also it displays how many videos have each CS. After run the ADP server, run the user. Then after Start the simulation and it will send the request by the list of users to the CSs. These lists of users are created randomly. In my result 3 CSs are get requests from the list of users and CSs are provide the videos for users. The ADP Server will display the status of the users request arrival or assigning and depart from CSs. Here, users depart from CSs is after completion of their file downloading/playing.

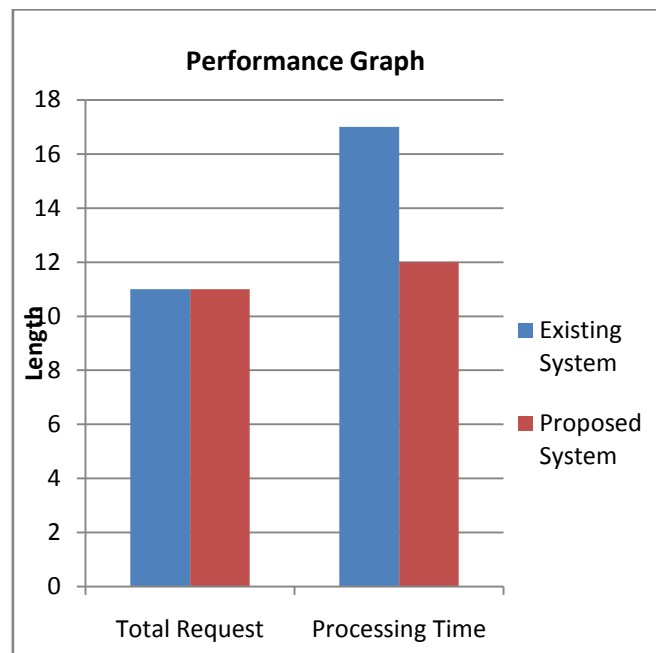


Fig. 4 Comparative graph of total request and processing time

View the Performance graph in the ADP server. In this graph we can see the total request and processing time.

VI. CONCLUSION

In this work, we examine an online video placement scheme for superior utilization and energy-saving in cloud delivery networks. We introduce a new problem that dynamically places incoming video subscribers to CSs to limit the number of active machines as well as the replication overhead. This problem considers both transmissions bandwidth and storage space constraints and is modeled. It can therefore be applied effectively to various types and scales of CDNs. By classifying servers into different types, our proposed ADP scheme places and reorganizes video subscriptions on their arrival and departure.

International Journal of Innovative Research in Science, Engineering and Technology

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Visit: www.ijirset.com

Vol. 8, Issue 2, February 2019

Through analysis, we demonstrate the effectiveness of ADP regarding performance and overhead. The worst-case overhead of ADP is limited, and the performance difference to the optimum is bounded. The outstanding performance of ADP is also evidenced by the simulations. The results show that ADP significantly outperforms the compared scheme under various conditions and maintains performance approximate to the optimal solution. In addition, the replication overhead of the system is also limited. To the best of our knowledge, ADP is the only scheme that addresses this placement problem and provides all the mentioned advantages.

REFERENCES

- [1] F. Xu, F. Liu, L. Liu, H. Jin, B. Li, and B. Li, "iAware: Making live migration of virtual machines interference-aware in the cloud," IEEE Trans. Comput., vol. 63, no. 12, pp. 3012–3025, Nov. 2014.
- [2] F. Xu, F. Liu, H. Jin, and A. V. Vasilakos, "Managing performance overhead of virtual machines in cloud computing: A survey, state of the art, and future directions," in Proc. IEEE, 2014, vol. 102, no. 1, pp. 11–31.
- [3] Y. Feng and B. Li, "Airlift: Video conferencing as a cloud service using inter-datacenter networks," in Proc. IEEE Int. Conf. Netw. Protocols, 2012, pp. 1–11.
- [4] X. Meng, V. Pappas, and L. Zhang, "Improving the scalability of data center networks with traffic-aware virtual machine placement," in Proc. IEEE INFOCOM, 2010, pp. 1–9.
- [5] P. X. Gao, A. R. Curtis, B. Wong, and S. Keshav, "It's not easy being green," in Proc. ACM SIGCOMM, 2012, pp. 211–222.
- [6] Z. Zhou, F. Liu, Y. Xu, R. Zou, H. Xu, J. C. S. Liu, and H. Jin, "Carbon-aware load balancing for geo-distributed cloud services," in Proc. IEEE Modelling, Anal. Simul. Comput. Telecommunication Syst., 2013, pp. 232–241.
- [7] W. Deng, F. Liu, H. Jin, X. Liao, H. Liu, and L. Chen, "Lifetime or energy: Consolidating servers with reliability control in virtualized cloud datacenters," in Proc. IEEE CloudCom, 2012, pp. 18–25.
- [8] Z. Zhou, F. Liu, H. Jin, B. Li, B. Li, and H. Jiang, "On arbitrating the power-performance tradeoff in SaaS clouds," in Proc. IEEE INFOCOM, 2013, pp. 872–880.
- [9] Gandhi, M. H.-BALTER, and R. Raghunathan, "AutoScale: Dynamic, robust capacity management for multi-tier data centers," ACM Trans. Comput. Syst., vol. 30, no. 4, p. 14, Nov. 2012.
- [10] Y. Ho, P. Liu, and J. Wu, "Server consolidation algorithms with bounded migration cost and performance guarantees in cloud computing," in Proc. IEEE Utility Cloud Comput., 2011, pp. 154–161.