ISSN 2320 – 2602
International Journal of Advances in Computer Science and Technology (IJACST), Vol. 3 No.2, Pages : 01 – 06 (2014)
Special Issue of ICCSIE 2014 - Held during February 16, 2014,Bangalore, India

Design of Techniques to Enhance the Services for Mobile Video-on-Demand Applications

D. N. Sujatha*, Girish K*, Venugopal K. R**, L. M. Patnaik***
*Department of Computer Applications
B M S College of Engineering, Bangalore-560019, India.
**Department of Computer Science and Engineering
University Visvesvaraya College of Engineering, Bangalore University, Bangalore-560001, India.
***Indian Institute of Science, Bangalore-560012, India.
dnsbms@gmail.com

***Abstract*— The current generation of mobile clients have incessant demand for services. Providing Video-on-Demand (VoD) services to these clients require a new paradigm. This paper presents an architecture for Mobile Video-on-Demand. Several schemes that reduces the load on the video server, thereby increasing the number of requests that need to be served is proposed. The *Distributed Indexing* reduces the storage load on the server and reduces the time to search for a video. The *Chaining Technique* increases the acceptance of requests and thereby reduces the rejection rate. Two types of hand-offs are defined :Video Server Hand-off and a Client Hand-off. These provide the continuity of services to the mobile clients. The *Windowing Scheme* which includes *Session Management* is proposed. This scheme control the flow of segments. All these schemes have been simulated and the results are presented.**

***Index Terms*– Chaining, Distributed Indexing, Hands-off, Session Management, Windowing.**

I. INTRODUCTION

VoD is the a brain-child of SeaChange Axiom Open Suite. It is a is used to encompass a wide range of technologies whose common objective is to enable clients to opt for videos from a central server for viewing on a television screen or a computer screen. VoD either streams content allowing the client to view the video while it is being downloaded or the video is brought in its entirety to the clients set-top box before allowing the client to view.

The hurricane in the form of advancement of wireless and mobile technology introduces a new dimension in VoD systems; it has been aptly coined Mobile Video-on-Demand (MVoD) as it provides VoD services to mobile clients. As people tend to work beyond their office desk, we can expect the next generation of wireless communication networks to include rapid deployments of independent mobile users. This ever growing market can easily be tapped by VoD providers. Mobile VoD is described as the coming together of the next generation of interactive television and a ubiquitous host. It suffers from a number of potholes which include fixed maximum bandwidth requirement, load adaptivity, clients sensitivity, limited coverage of a wireless transmission medium, clients with heterogeneous capabilities, etc.

The video server that services the request of the client has a fixed and defined bandwidth through which it can satisfy all the requests of the client. If there are more number of requests, then the client will have to wait for the bandwidth to become available. This increases the reneging factor and a number of requests are likely to be rejected. Therefore, load adaptivity is extremely important. The video server should have an optimum coverage area to stream the video to the client as the clients are always on move. The video server should also take into account the heterogeneous capability of the clients devices.

The VoD still lacks a universal standardization as the underlying technologies are relatively new. Nevertheless, many research institutes and commercial organizations have established de-facto standards and consequently, there are many operational VoD-related services available today [4]. The biggest stumbling block faced By VoD is the lack of a network infrastructure that can handle the large amounts of data required for streaming the video. Therefore there is a definite problem to provide continuous streaming of data for a ubiquitous host under the existing network infrastructure.

Motivation: The motivation behind this work is to address the challenges faced while providing VoD services to mobile clients. The challenges include continious streaming of video, support the heterogeneous capability of the client and reduce the load on the server.

Contribution: The objective of this work is to design schemes to render services to portable devices. The video search time is reduced by using distributed indexing. The session management scheme is implemented to ascertain smooth downloading of videos from one or more clients and chaining scheme reduces the load on the video server, services more number of requests with less bandwidth

Outline: The reminder of this paper is organized as follows. Section II presents literature survey in the area of mobile VoD. The system model for M-VoD System is discussed in Section III. We elucidate the usefulness of our algorithm through simulation and performance analysis in Section IV. Finally, we provide concluding remarks and future enhancements in Section V.

ISSN 2320 – 2602
International Journal of Advances in Computer Science and Technology (IJACST), Vol. 3 No.2, Pages : 01 – 06 (2014)
Special Issue of ICCSIE 2014 - Held during February 16, 2014,Bangalore, India

II. LITERATURE SURVEY

In this section we present a concise description on the existing work that has been carried out in the area of Mobile VoD. Tran Et al. [1] thave proposed a three-tier architecture for mobile VoD. A broadcast protocol based on Periodic Broadcast to disseminate videos to clients is proposed. Selective-caching scheme called Dominating-Set Cache, which allows clients to play a video by exploiting nearby caches is discussed. Sato et al. [2] presents a new form of VoD called Fragmented Patching, which enables clients to move around freely even while receiving videos. In fragmented patching the patch flows are sent via broadcasting. In addition, to avoid increasing traffic due to broadcasting, the patch flows are broken down into segments, each of that is aggregated to be shared with as many multiple clients as possible.

Bellavista et al. [3] presents the distributed infrastructure of the SOMA* (Secure and Open Mobile Agents) programming environment, and particularly focuses on its mobility. SOMA allows all entities to move autonomously and to adapt to the current system situation and to the current characteristics of their points of attachment to the network. It describes the adoption of mobile code technologies to support client and terminal mobility.

In paper [4] a two-level buffering strategy to maintain streaming continuity is proposed. The Mobile agent based Ubiquitous multimedia Middleware (MUM), a dynamic and flexible infrastructure to support both streaming quality adaptation and session continuity, independently of client roaming is presented. Bellavista et al. [5] present MUM Open Caching (MUMOC), dynamic and flexible overlay infrastructure for the distributed caching of both VoD prefixes and VoD metadata. Active services can improve VoD streaming over the best-effort Internet, to primarily support prefix caching and to achieve interoperability, by providing open and standard representations of the available VoD flows in order to simplify the inter-working with legacy VoD systems.

Venkatraman et al. [6] propose a Mobile Multimedia Network that consists of several cells where each cell comprises of multimedia servers and a base-station. This paper propose a mechanism to maintain the continuity of data transfer during handoffs in mobile-lP environments and minimizes the data loss during this period. This method performs an efficient buffering of the data at the Base-station by using toggled buffers and enables it to calculate optimal playout time for multimedia applications.

In paper [7], Friend Relay, a resource-sharing framework for mobile wireless devices that offers automatic publishing, discovery and configuration, as well as monitoring and control is presented. Roy et al. [8] describes a Mobile Streaming Media Content Delivery Network (MSM-CDN) that can facilitate the access of rich multimedia streams by mobile clients on wireless networks. This paper describes some of the design requirements for such servers.

In paper [9] schemes for distribution of Object based Continuous Media, Information Quality based Distribution and Mobility Aware Continuous Media Distribution is presented. It identifies trade offs between information quality and currency of information. This paper introduces the notion of quality of information and uses it for the efficient delivery of multimedia streams in a mobile wireless environment.

Brandt et al. [10] deals with the adaptation of video on a client. They describe five different classes of adaptation mechanisms that are useful for mobile devices: temporal adaptation: reducing the frame rate, spatial adaptation: reducing the spatial resolution, quality adaptation: reducing the quality of each frame, format adaptation: changing the encoding format, structural adaptation: changing the contents of the stream

Mate et al. [11] deals with session mobility in clients. It describes the key difference between physical and service mobility is that while the first one attempts to continue the service uninterrupted even as the device moves, the latter attempts continuous service experience even if this is consumed from a different device.

III. SYSTEM MODEL

A. Architecture

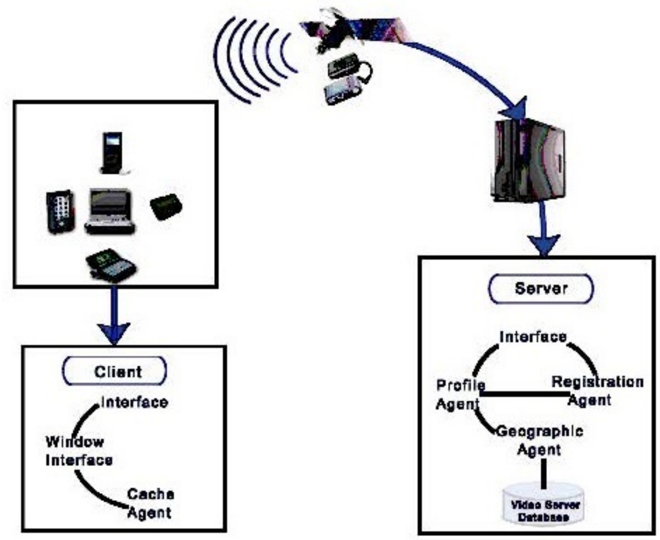

Fig. 1. The Overview of the Mobile Video-on-Demand System.

The Overview of the Mobile Video-on-Demand System is shown in Fig. 1. It includes the clients connected to the video servers on a wireless network. The server receives requests from the clients and streams the requested video to the client.

Each client uses the interface to submit the request to the video server. The interface is maintained as a window. The window manages the flow control while receiving the video streams. A cache agent is present which buffers the streams that are waiting to be processed.

The server consists of an interface that interacts with the outside world. The interface consists of profile agent, registration agent and geographic agent. Each client register

ISSN 2320 – 2602
International Journal of Advances in Computer Science and Technology (IJACST), Vol. 3 No.2, Pages : 01 – 06 (2014)
Special Issue of ICCSIE 2014 - Held during February 16, 2014,Bangalore, India

with theregistration agent of the VoD system by specifying the profile.The profile encompasses home address, care-of address, location, type of device, availibility of resources to send and receive video streams. The profile agent stores the details in the database. The geographic agent constantly keep track of the geographic location of the client.

Fig. 2. The Architecture for Mobile Video-on-Demand System.

The Architecture for Mobile Video-on-Demand System is depicted in Fig. 2. The architecture consists of video servers and clients. The video servers are connected in a peer-to-peer network. Each video server has control over a specific area referred to as domain. The clients present in a specific domain use the services of the video server of that domain. The domains of two video servers overlap each other. Therefore, when a client reaches the overlapping region an assumption is made that the client would be moving towards another video server.

A client in the domain of one video server is Hand-off to another video server when the client is moving away from the area covered by the first video server and entering the area covered by the second. It is very important to maintain continuity in video transmission during the process of handoff. In this architecture we define two types of hand-offs: Video Server Hand-off and Client Hand-off

Video Server Hand-off occurs when a client downloading a video from its video server is moving towards the domain of another video server. The domains of the video servers overlap. The downloading of the video should be continue to the video server into whose coverage area the client is moving. When the client reaches the overlapping region, a handoff is made in order to avoid interruptions. The client information, with the video information and the information on the last segment received is sent to the new video server. The new video server, checks its repository, if the video is present, it streams the video to the client. If not, it accepts the request for video and searches for the same in the neighbouring video servers and then continue streaming to the client. This ensures the continuity of video streaming to the client without the clien'ts knowledge.

Client Hand-off occurs when a client which is downloading the video from another client or more than one client moves out of the coverage area of any of the sending clients. If the client is downloading the video from one client and it leaves the coverage area of that client, then the request would be sent to the video server with the information on the last segment received and streaming of video stops. If the client is downloading from multiple clients, then if it moves from the coverage area of any of the clients, then the sending client checks for response by resending the last segment to the client. If there is no response then it remove the details of the clients and stops downloading.

B. Proposed Schemes

In this subsection, we present the proposed schemes : Chaining Technique, Windowing Scheme, Session Management and Distributed Indexing which enables the effective delivery of services to portable devices.

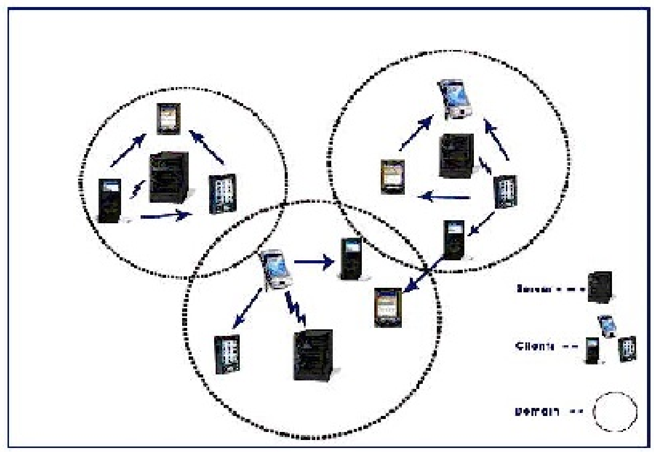

Fig. 3. Chaining Scheme for M-VoD System.

1) Chaining Technique: /indent Fig. 3. depicts the chaining process in M-VOD System. The chaining process start as a mobile client submits the request for the video to the video server. The video server in turn request the video among the clients in the neighboring video server who might be streaming the same video. If there are clients that are downloading the same video and are in the coverage area of the client the details of these clients would be send to the video server who in turn send these details to the requesting client. If there are no such client,s then the video server would search its own domain to see if there are any clients in the coverage area of the requesting client downloading the same video. If there are clients, then the information of these clients would be send to the client giving them the option for choosing the clients the client can download from. If there are no clients in the neighboring or current domain then the video server would search for the video in its repository, if it does no, then it checks the index of the neighbors if the movie is not present in the index then the videos unique id

ISSN 2320 – 2602
International Journal of Advances in Computer Science and Technology (IJACST), Vol. 3 No.2, Pages : 01 – 06 (2014)
Special Issue of ICCSIE 2014 - Held during February 16, 2014,Bangalore, India

would be passed to the neighbors which would only check the index of its neighbors. If found in the index the video server which has the video in its repository would stream the video to the video server who in turn would stream it to the requesting client.

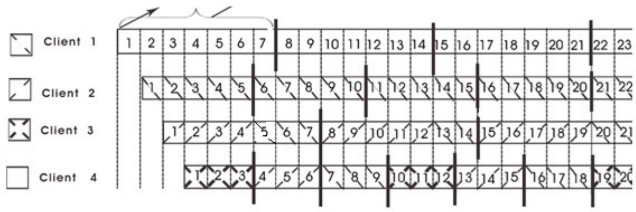

Fig. 4. Windowing Scheme for client.

2) Windowing Scheme: The windowing scheme is presented in Fig. 4. Assume that 4 clients have requested the same video at time t0, t1, t2, and t3 respectively. Client 1 receives the stream from the video server itself. It first opens a window of 7 frames (that is the maximum size it can receive). Client 2 arrives at time t1 and starts receiving the stream from Client 1. It would open a window of 5 frames. This lowers the overload on the server and service more requests and hence, reduces the latency and the reneging factor.

When Client 3 arrives it starts receiving the streams from both Client 1 and Client 2. Client 3 would open a window of 7 frames. (Maximum of window size of Client 1 and Client 2 i e 7 frames). In the above example Client 3 receives the first 4 frames from Client 2 and the next 3 from Client 1. At time t3 Client 4 requests for the same video. The video server recommends Client1 Client2 and Client 3 to Client 4. Client 4 decides to use all 3 clients. Client 4 receive the first 3 frames from Client 3, the next 3 from Client 2 and the next from Client 1. This reduces the overhead on each client. The windowing scheme is an extension of the sliding window protocol. It has been modified to assist in the delivery of multimedia data in a VoD system.

a) Session Management : Session management is the term that is used to manage the chaining scheme, when a client is downloading the video from more than one client. The requesting client would receive the QoS and streaming capacity of the potential senders. The client is empowered to choose the clients If the client does not select any client, then the request will be transferred to the video server in order to service the request. If the client selects the clients, then the minimum QoS is selected and the window size is determined. The window size is determined by the maximum capacity of the sender and receiver. If the receiver has more capacity than all the senders, then a window of the maximum capacity of the sender is opened, if not the maximum capacity of the receiver is opened. The streaming of Video starts as soon the window size is determined.

TABLE I
PARAMETERS USED FOR SIMULATION STUDY.

Parameters	Value/Range
Number of clients in a domain	10-25
Total number of clients	100
Simulation time	600 minutes
Communication range of all nodes	100 m
Server position	(100, 150)
Length of multimedia file	30
Connection setup time	1 Sec
Number of servers	3-6

3) Distributed Indexing: Distributed Indexing is used to lower the search time and the overhead on the bandwidth. All the video servers are connected to each other via a peer-to-peer network. All the video servers keep an index file containing the name of the videos present in its neighbors. When the video server receives a request from the client, it searches its repository for the video. If the video is found, it checks, if enough bandwidth is available, calculates the frame rate and video quality based on the capability of the portable device. If the video is not present in the repository, then the index files of the neighboring video servers is checked. If the index files of the neighbors do not contain the video then the video id would be sent to its neighbors.This would reduce the overhead on the server and reduce searching time. If more than one video server has the same video, then the video server with the shortest path would be calculated and the video would be streamed.

IV. SIMULATION AND PERFORMANCE ANALYSIS

The parameters used in simulation study are shown in Table 1. The Simulations starts with the client requesting for a video. The request of the client is sent to the video server covering that domain. The video server receives the details of request made by the client like client-id, video-id, geographic location of the client to enable the start of registration process. It then stores all these details of the client in the database.

Once the initialization is done, it check the neighboring video servers to find if any of its clients are downloading the same video and are in the coverage area of the requesting client. If there are any such clients, the video server stores the details of those clients. Then, the video server checks if there are any clients that are in the domain downloading the requested video and in the coverage area of the requesting client. If it finds more than one client, it stores those the details of clients and then calls Algorithm SessionManagement( )

ISSN 2320 – 2602
International Journal of Advances in Computer Science and Technology (IJACST), Vol. 3 No.2, Pages : 01 – 06 (2014)
Special Issue of ICCSIE 2014 - Held during February 16, 2014,Bangalore, India

to manage chaining scheme when the client is downloading from more then one client. If there is only one client it would call Algorithm Stream(lastsegment) to stream the video.

If it does not find any clients, it then searches its repository for the video. If the video is not present, it invokes Algorithm DistributedIndexing( ) to search for the requested video in the neighbouring video servers. If the requested video is found in more than one video server, then the nearest video server is chosen for streaming the video to the client.

If the requested video is present in the repository of the video server, it verifies for the availibility of sufficient bandwidth. If the bandwidth is currently not available, then the request is placed in a queue until the required bandwidth becomes available or till the reneging time is elapsed in which case the request is rejected. If the bandwidth is available, then the Algorithm Stream(lastsegment) is invoked.

If the client that was receiving the video from another client leaves the coverage area of that client then, it sends the information on the last segment it received to the video server. Video streaming for that client would start from the next segment. Therefore the client does not lose the contents of the video. If the client does not chose any clients to stream the video then the algorithm start with the video server searching its repository for the requested video.

Fig. 5. is a plot of the number of requests submitted

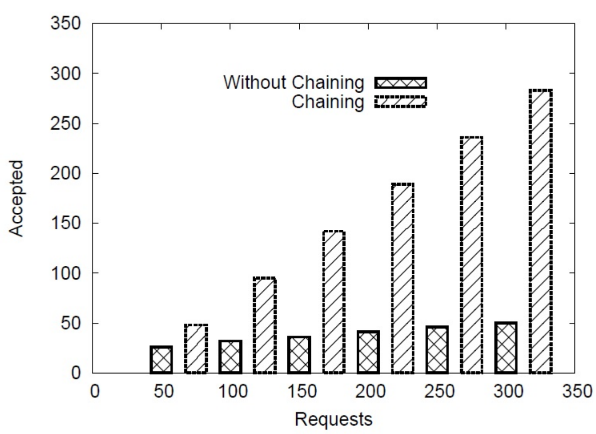

Fig. 5. Accepted.

to VoD System Vs the number of requests accepted and serviced. It evident from the graph, as the number of requests increease, thenumber of requests that are serviced by adopting chaining scheme grows exponentionally. However, the number of requests processed without adopting chaining mechanism shows a linear growth. This graph is complementary to the rejection graph shown in Fig. 7. This demostrates that using chaining mechanism more number of requests are serviced.

The utilisation of bandwidth in M-VoD system with and with out chaining is depicted in Fig. 6. It can be clearly seen in the graph that the consumption of bandwidth in M-VoD system with chaining is less than the consumption of bandwidth in M-VoD System without chaining. As more

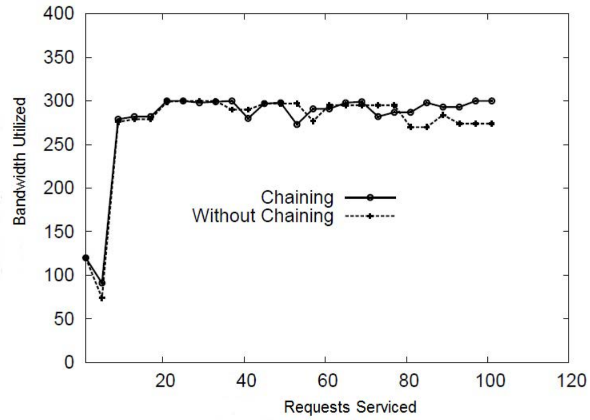

Fig. 6. Bandwidth Utilized.

number of requests comes in, more number of clients would download the video using the chaining mechanism, Hence, the consumption of bandwidth on the video server is reduced enabling it to service more number of requests.

The number of requests that are rejected is depicted in

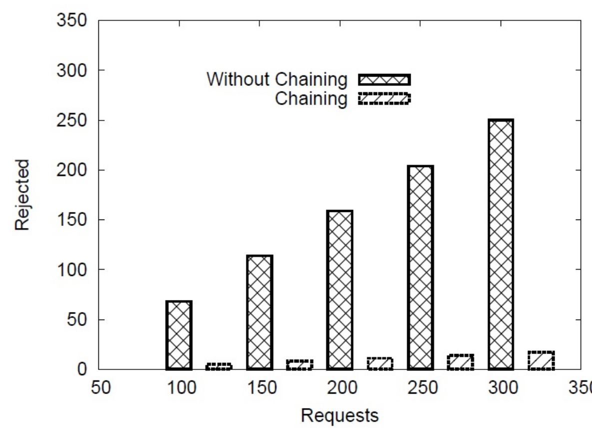

Fig. 7. Rejected.

Fig. 7. The requests get rejected as they are not accepted and serviced in time. The requests were not accepted as they exceed the reneging time, resources like bandwidth was not available at the time of submission of request or in the worst case, the requested video was not available. There is a clear difference in the ratio of the requests received verses the requests rejected between the M-VoD system with chaining and without chaining. It can been seen in the graph that 0.035 % of video requests get rejected in an M-VoD system that adopted chaining. This in contrast to M-VoD System that does not adopt chaining mechanism, where the number of requests rejected increses with the number of requests. In the former case the clients requesting the videos are chained and therefore the load on the video server is reduced. Hence, video server processes more number of requests.

Fig. 8. shows the load on the video server. Initally, when

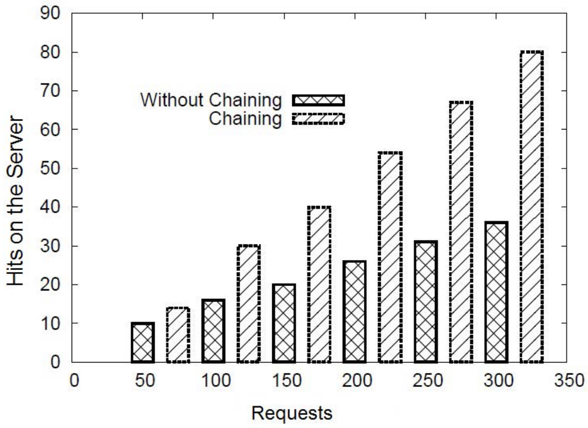

Fig. 8. Server Load.

the number of requests is 50, the number of hits on the M-VoD system with and without chaining is almost the same. However, when the number of requests are 300, the number of hits are more in M-VoD with chaining, than without chaining. The point of interest in any system is behavior during high load. This clearly demonstrates the superiority of M-VoD System with chaining.

V. CONCLUSION AND FUTURE WORK

We have proposed an architecture that provides VoD services to portable devices. Portable devices have gone through an evolutionary process from just being able to communicate to the age of downloading and viewing of multimedia streams. To facilitate this our architecture is reduces the load on the video server by using a *Chaining technique*. This ensures that even if there are a large number of requests, the number of requests serviced is high.

To reduce the load on the video server, the videos are distributed among the video servers. The *Distributed Indexing scheme* reduces the search time for a video. The video server hand-off and the client hand-off will ensure continuity in streaming the video. The client is given the option of selecting the client/ clients to download the video, as most clients have different capabilities.

We have implemented *Session Management* to make sure that downloading the video from one or more clients goes on smoothly. Multiple clients are preferred for downloading the video in order to avoid the load on a single client.

Further enhancements can be made in providing a secure transmission. The battery life, power consumption, heterogeneity of clients are to be studied. In future, clients themselves become content distributors enabling video services to be provided to other clients thus entering the area of social networking

REFERENCES

[1] Duc A. Tran, Minh Le, Kien A. Hua, "MobiVoD: A Video-on-Demand System Design for Mobile Ad hoc Networks", In the proceedings of the 2004 IEEE International Conference on Mobile Data Management (MDM 04), 2004, pp. 212-223.

[2] Sato, Michiaki Katsumoto, Tetsuya Miki,"Fragmented Patching: New VOD Technique That Supports Client Mobility", In the proceedings of the 19th International Conference on Advanced Information Networking and Applications (AINA 05), vol. 1, 2005, pp. 527-532.

[3] Paolo Bellavista, Antonio Corradi, Cesare Stefanelli, "A Mobile Agent Infrastructure for the Mobility Support", In the proceedings of the 2000 ACM symposium on Applied Computing - Vol. 2, Italy, 2000, pp. 539-545.

[4] Paolo Bellavista, Antonio Corradi, Lucas Foschini, "Java-based Proactive Buffering for Multimedia Streaming Continuity in the Wireless Internet", In the proceedings of the Sixth IEEE International Symposium on World of Wireless Mobile and Multimedia Networks (WoWMoM 05), 2005, pp. 448-450.

[5] Paolo Bellavista, Antonio Corradi, Lucas Foschini, "MUMOC: an Active Infrastructure for Open Video Caching", In the Proceedings of the First International Conference on Distributed Frameworks for Multimedia Applications (DFMA'05), 2005, pp. 64-71.

[6] P. Venkataram, R. Rajavelsamy, S. Laxmaiah, ""A Method of Data Transfer Control during Handoffs in Mobile-IP based Multimedia Networks, In the ACM SIGMOBILE Mobile Computing and Communications Review, Vol. 5 , Issue 2 (April 2001), 2001, pp. 27-36.

[7] Hillary Caituiro-Monge, Kevin Almeroth, Maria del Mar Alvarez-Rohena, "Friend Relay: A Resource Sharing Environment Framework for Mobile Wireless Devices", In the proceedings of 4th international workshop on wireless mobile applications and services on WLAN hotspots 2006 (WMASH 06), Los Angeles, CA, USA, September 29-29, 2006, pp. 20-29.

[8] Sumit Roy, Jhon Ankcorn and Susie Wee, "Architecture of a Modular Streaming Media Server for Content Delivery Networks", In the proceedings of the 2003 International Conference on Multimedia and Expo - (ICME '03) - Vol. 03, 2003, pp. 569-572.

[9] Sandeep K. S. Gupta, Martin Reisslein and Arunabha Sen, "Towards Information Quality Based Distribution of Multimedia Streams in Mobile Environments", In the proceedings of the 8th IEEE Workshop on Future Trends of Distributed Computing Systems (FTDCS- 01), 2001, pp. 25-30.

[10] Jens Brandt, Lars Wolf, "A Gateway Architecture for Mobile Multimedia Streaming", In the conference proceedings of European Symposium on Mobile Media Delivery (EuMob06), September, 2006.

[11] Sujeet Mate, Umesh Chandra, IgorD. D. Curcio, "Movable-Multimedia: Session Mobility in Ubiquitous Computing Ecosystem", In the proceedings of the 5th international conference on Mobile and ubiquitous multimedia, Stanford, California, Article No.8, 2006.