CONTENT ADAPTATION AND SHARED STATE DISTRIBUTION FOR MULTIPLAYER MOBILE GAMES

Qasim Mehdi, Pawan Kumar, Aly Salim and Kamal Bechkoum
School of Computing and Information Technology
University of Wolverhampton
Wolverhampton, UK WV1 1SB
{Q.H.Mehdi, Pawan.Kumar, A.Salim2, K.Bechkoum}@wlv.ac.uk

Abstract:

Typically games on mobile devices are limited to standalone single player games that are not only simple but also are limited by the device capabilities on which they are being played. With the advancement of networking technology, mobile multiplayer games have started to evolve. Nevertheless, these games are played on homogenous devices that have limited functionality and performance. For performance, scalability and heterogeneity, it is important that mobile multiplayer games be played on heterogeneous devices and able to support large number of players for an immersive experience. This demands that players not only receive the content as quickly as possible but also the content be adapted to the device capabilities. Further as different devices have different computing capabilities; it seems reasonable to distribute resources dynamically among the mobile players' devices so that the overall shared state is maintained in a consistent state. In this paper we highlight the issues related to multiplayer gaming on mobile devices and provide a proposal for content adaptation and shared state distribution for multiplayer games on mobile devices based on dynamic scripting approach

Key words:

Content Adaptation, mobile games, WIMAX, WIFI, Bluetooth 2.0, QOS, interest management, dynamic scripting

1. Introduction

As more and more mobile devices grace the earth there is a need to use them in different scenarios and situations to the best of their abilities. Thus in gaming terms utilising them on multiplayer mobile gaming should maximise their potential. Mobile devices as we know them are different i.e. there are PDA's, smart phones, tablets etc. The ability to play multiplayer games over heterogeneous devices seamlessly and without the device user's knowledge (while distributing processing loads to different mobile devices as per resource availability) would not be possible. This raises issues such as content adaptation and distributed processing on mobile devices as well as networking issues that crop up within these situations especially when wireless communication is involved. These are challenging issues that need attending to. Some concepts of content adaptation are covered in some depth in the Olga report where UMA (Universal Media Access) is looked at. This is a concept that encompasses the ability of different constrained mobile devices to access rich media resources [1].

Some work has been accomplished in this area while some research is still in progress, which will be looked at later in the paper. Limitations arising from using wireless networking over mobile devices include limited battery power, constant change in device location, network traffic due to bandwidth being used by other types of communication (including Bluetooth and infra-red). As far as these limitations are concerned there has been some work carried out to optimise the use of network resources. This includes work done in areas of distribution concepts, which involves the choices of architectures and protocols suitable for mobile device multiplayer gaming for example peer to peer, client server and server network [3]. An area that does focus on some aspects of reducing these network issues includes compensatory techniques whereby messages compressed and aggregated reducing bandwidth and transmission frequencies. Another technique is Interest management which will be detailed later on in this paper [3]. Work has been carried out using peer-to-peer overlay architecture for interest management in multiplayer games especially in addressing scalability issues. In this case P2P is used with an interest management mechanism to reduce some of the side effects such as limited visibility that comes with P2P architecture [4]. There are a few limitations as far as grid computing on mobile devices is concerned. These are similar to the networking limitations and include unreliable wireless connectivity, power consumption sensitivity, and software interoperability between different devices [6]. There are some main research projects that have been undertaken thus yielding some interesting results including mobile OGSI.NET which looks at creating a mobile specification for mobile grid computing using pocket PC's running Microsoft operating system [2]. The research on mobile ad-hoc grid networks is still quite a new field. A fair bit of work in this area does not deal with gaming as a beneficiary but rather looks at distributing processes among mobile devices to reduce computing power, increase energy saving, and build efficiency in running more powerful applications on mobile devices. The seamless integration of networking protocols with grid computing principles and interest management in-order to develop a middleware that can actually provide multiplayer gaming regardless of devices hardware and software capabilities is what researchers are moving towards. This paper deals with content adaptation in multiplayer games with regards to different devices being used as such. It raises questions on how multiplayer gaming is conducted on mobile devices which are detailed in the next section. The primary aim of this paper is to try and dissect these questions raising probable solutions to be explored during the investigation. In order to do that, and in addition to these introductory notes, this paper is organised as follows. Section 2 is dedicated to giving a detailed description of the questions related to multiplayer gaming on mobile devices. Section 2.1 deals with the associated networking issues. Software architectures for online games are briefly reviewed in Section 2.2. Section 2.3 deals with scalable-shared state distribution and section 2.4 elaborates on content representation in games. Section 3 deals with content adaptation and shared state distribution on multiplayer games with a twist on how we propose to tackle the issues while Section 4 concludes the paper.

2. Multiplayer gaming on mobile devices

For a game to be successful on mobile devices, several issues need to be addressed upfront. Depending on the genre, these may include the issues of latencies, bandwidth, scalability, real-timelines, etc. For instance, a first person shooter would demand a low latency, real-time responses whereas a role-playing game can get away with a higher latency but may require supporting several hundreds of simultaneous users. In addition, the device capabilities and networking infrastructure add constraints that need to be dealt with appropriately for a successful gaming application. Device capabilities include its operating environment such as resources available in form of processing, memory, connectivity and battery usage. Networking support in these devices may range from wireless LAN, WIMAX, GPRS and Bluetooth. Given the heterogeneity in the devices available in the market, mobile multiplayer games have been limited to homogeneous devices over fixed settings and thus are not truly mobile. For performance, scalability and heterogeneity, it is important that mobile multiplayer games are played on heterogeneous devices in a truly mobile environment and are able to support large number of players for an immersive experience. This demands addressing several key questions. Firstly, how the game content comprising representations of geometry, texture, animations, audio and video to be stored and distributed in order to provide a consistent view of the virtual world? Secondly, how to support large number of players having different devices and interacting with the virtual world without degrading the performance? Thirdly, how to distribute the shared state and the processing intensive tasks across multiple mobile devices? Fourthly, how to adapt the content and the shared state in these purely mobile and distributed games with respect to changes in device connectivity and usage?

In the following sections we briefly provide an investigation of networking issues in mobile games, software architectures for supporting online games, approaches for scalable shared state distribution and content representation in games.

2.1 Networking issues in mobile games

Multiplayer mobile gaming involves networking. It does have a big bearing on how the game performs in terms of seamless distribution of game states and messages. This may include getting round limitations like latency, bandwidth, scalability etc. These however can be minimized to an extent depending on network communication chosen (e.g. WIBREE, ZIGBEE, Bluetooth 1.0, Bluetooth 2.0 EDR, WIFI, WIMAX, and WIBRO. These are currently available standards that can be found on different mobile devices, some being more widespread than others. Mobile devices have different capabilities as far as communication with other devices is concerned. These include WIFI, Bluetooth1.2 and 2.0, WIMAX, ZIGBEE, and WIBREE etc. Mobile devices pose a few challenges in terms of seamless uninterrupted communication. Challenges faced include mobility where some devices can go out of range resulting in a reduction in the number of devices connected to the network [5]. This has its pitfalls with regards to state of the application (game state) at the time of departure from the network. ZIGBEE, WIBREE are not suitable for real-time multiplayer gaming communication and data exchange over wirelessly due to their limitations in data transfer rates as well as availability on devices too [7, 8]. Bluetooth 1.2 and WIFI have been used so far in multiplayer gaming but mostly on handheld gaming devices like Nintendo DS and PSP. However their operating range and support for mesh topology can be a problem [9, 10 & 13]. There are other issues that are related to content distribution for gaming applications over the wireless network such as different standards and limitations. Therefore the main issues that concern network aspects for multiplayer games include: scalability which allow the games to adapt to resource changes as seen fit; distribution concepts which encompass architectures for communication (client/server etc); networking resources which include the latency; bandwidth; and computational power available)[3].
However with the advent of Bluetooth 2.0 EDR and WIMAX the range, scalability and QOS (Quality of Service) have been improved. Therefore it can be argued that the best way to induce better communication and data exchange between devices during game playing is by looking at better distribution concepts and compensatory techniques including interest management and process redistribution using grid concepts [3]. The communication protocol being trialled for this investigation and prototyping is a contraption of WIMAX and Bluetooth 2.0 EDR. Bluetooth 2.0 is used as a communication protocol between different devices within range of each other forming Pico-nets that communicate through a nominated server points. The nomination process of these points would depend upon a few set rules taking into account necessary factors including power levels and computational resource. This would be a good way to tackle some issues with regards to range and strength of signal between mobile devices.

2.2 Software architectures for online games

Network architectures namely client/server, P2P and hybrid [4] do play an important role in multiplayer games. They form the backbone of how the game is distributed around the network of players. The manner in which data is relayed across the multiple player devices with respect to their

capabilities dictates how well the game is executed on the whole.
There are numerous architecture models used in networking including client server, peer to peer and hybrid. Client server being the more popular architecture with gaming companies due to a number of reasons such as:

- Easy to implement.
- Easy to maintain state consistency due to having a centralized server.
- Hard to cheat and hack into the game.

The client-server inherently suffers from single point failure if triggered would result in server failure and the loss of game content. High bandwidth requirement at the server would present a challenge especially with wireless networks due to their limited bandwidth capabilities [15].
The peer to peer architecture does tend to be used for some games but not as much due to a few factors that include inability to harness control over game administration due to each device having to run its own game software thus messaging other devices to relay game states and update its own too. This may prove detrimental towards mobile devices due to them having limited storage and processing power [14, 15]. It does however pose a few advantages that could be of good use for multiplayer gaming which include the elimination of one point of failure, reduction of bottleneck and reduced message latencies. However it is harder to implement the P2P architecture. One reason being its difficulty in scaling with proportion to users engaged. State consistency among players would then become an issue [15]. This may not be suitable to content adaptation due to the fact that while playing multiplayer games with different devices of different capabilities the ability to adapt content to suit each device format is crucial. It does show strain in coping with increased amount of users due to its increased messaging thus containment of users is not easily achievable. There is also the issue of computational power and battery life, which are all constrained factors on mobile devices. P2P does rely on these two factors due to the fact that there is no central repository to work with. This thus brings about some issues with regards to computing resource management and battery life management [14].
The hybrid/mirrored server architecture takes into account the advantages from both architectures and presents a sound case for deployment of multiplayer games on it. There is a more structured coupling with the hybrid system though it still has to cope with message bottleneck that does trigger network traffic. This in-turn affects other aspects of mobile device limitations.
The bandwidth requirement for P2P and client/server is nearly the same. Thus for this particular investigation the client/server architecture seems to be a more reliable approach to start with. It would suit mobile devices due to their limited capabilities and thus it can take up a lot of the processing away from them ensuring a sound gaming experience on the client side. It would be also be a lot easier to implement and work with. This will be intertwined with the use of interest management that will be detailed later in the paper.

2.3 Scalable shared state distribution

Performance and scalability in multiplayer games requires efficient distribution of the shared state. Typically, a player's node will contain some subset of the shared virtual world whose state is influenced and maintained by the player. In order to have a mutual consistent view of the virtual world, events or messages are exchanged between player nodes (either directly or indirectly through a server). However, an update occurring at one node is likely to have an immediate significance for only a subset of other nodes in the system. The techniques that exploit this *interest* of each node to minimise the number of messages sent are referred as *interest management* (IM) schemes [16].

Interest management systems have been incorporated in several large-scale distributed simulators [17,18], collaborative virtual environments [19,20,21] and multiplayer online games [22,23,24]. These have been incorporated mainly to allow systems to scale seamlessly and efficiently. The scalability in these systems is primarily related to the number of nodes that can participate and the computational complexity of the model that is being simulated e.g. in a game it could be the number of entities the game has. Without the IM system, every update or state changes at one node would need to be communicated to all the other nodes. This could significantly increase the bandwidth usage, message sent per second and computational requirements at processing these messages. However, incorporating IM systems would try to minimise the above at the expense of computational costs for its processing and thus affecting the real-time requirements of these systems and potentially reducing performance. Thus, performance and scalability of these systems mainly depend on the effectiveness of the deployed IM scheme in these systems. In our previous work [25] we presented a scalable algorithm for interest management in online games and highlighted some of the related works in the area. These would be highly appropriate for addressing the second and third questions of scalable-shared state distribution in multiplayer games on mobile devices.

2.4 Content representation in games

Game content mainly comprises representation of geometry (for characters, entities, levels, etc), textures, animation sequences, audio and video. All these are used to create a virtual world that the user can interact with. Several techniques have been developed for efficient representation of these elements. For geometry, these include polygonal approaches and triangulation to form 3D meshes. *Triangle soups* can be used to represent entities and geometries from semantically unrelated parts. However, these approaches require prior knowledge about device rendering capabilities and make a compromise between the number of triangles and the quality of the model. Several scalable approaches for geometry representation have also been used that adapt to fast rendering with decreased visual quality and slow rendering with high detailed model. One of these approaches is the *Level of Detailed* (LOD) [26] modelling where a model is represented with several planer meshes having different levels of granularity. Apart from rendering benefits, this approach is highly appropriate for sending entities from content servers to client devices depending on the communication infrastructure supported by the device and the available bandwidth. Further, if procedural approaches are used for moving from one detailed representation to another, this approach can further be exploited where coarser LOD is transferred first to the client device while the remaining ones are sent gradually and incorporated within the rendered mesh. This approach clearly compliments rendering and transfer of content in the wake of variable device capabilities and connectivity.

Another approach that is used in scalable geometry is based on mesh reduction algorithms and techniques. Mesh reduction algorithms reduce the polygon count of highly detailed meshes and optimise them for rendering. This can be achieved by an offline tool or can be done more dynamically at runtime based on the constraints such as distance from the viewer, frame rate, etc. These reduction approaches can be very fast and yield good approximations to the original model that can be appropriate on mobile devices that have variable degree of constraints associated with them. Other approaches to scalable geometry include representations based on curved surfaces (Bezier, splines, etc) that are more efficient and require less storage than polygonal techniques discussed above. These have been well adopted in multimedia standards such as MPEG [27]. In addition, MPEG standard incorporates special encoding and compression techniques that would be highly appropriate for their usage on mobile devices. For textures, audio and video content, several standards exist [27,28,29,30] that use state of the art encoding and compression techniques that can be used appropriately depending on the constraints associated with the devices and the quality of service requirements.

Once we have geometry representation and textures for an entity, it needs to be placed and moved in the virtual world with respect to others and/or have deformations applied on it. Animating an entity thus requires applying and updating geometric transformations such as translation, rotation and scale by either applying dynamics or key framing (based on interpolation of some function of time). Further, in case of deformations, the animation continuously changes the shape of the entity. This typically requires deformation controllers attached to the entity that influences its shape. These controllers are typically defined by a means of geometric support (such as points, lines, curves, surfaces, and volumes) and any changes made to these are reflected on to the entities to which they are attached. There exist several representation techniques for representing the actual animation data. These can be in human readable XML-based coding or have advanced compression based compact binary formats [27]. Furthermore, compact representation of animation parameters can consist of decomposing transformations into elementary motion and using quaternion or Euler angles for rotations. Other such optimisations and compression-based representation schemes have been well researched and used in games for animation that would be incorporated for content representation and their adaptation for mobile multiplayer games. In the following section we present a new approach for content adaptation in mobile multiplayer games

3 Content adaptation and shared state distribution for multiplayer mobile games

Adaptation for mobile devices has been an active research area and several projects have been developed that address one aspect or the other [31,32,33,34,35]. However, previous research work is mainly limited to generic and web based applications rather than content rich multimedia applications and games. Some provide a manual [36] and user feedback [37] mechanism for adaptation whereas other uses automatic adaptation techniques based on rule-based [38,40] and constraint based [39] approaches. For scalable multiplayer games on mobile devices we are mainly interested in adaptation of the content that is received by the client devices. At the same time we are also interested in the dynamic adaptation and distribution of computing tasks among those devices. A successful implementation of online adaptation in games demands several computational and functional requirements as suggested by Spronck [41]. These include four computational requirements of speed, robustness, effectiveness and efficiency and four functional requirements of clarity, variety, consistency and scalability. To meet these requirements, Spronck suggested '*dynamic scripting*' technique that we plan to use for content adaptation and task distribution on mobile devices. According to Spronck, dynamic scripting is an online machine –learning technique that is characterised by a stochastic optimisation technique. It maintains several rule bases one for each agent and every time an agent is generated, the rule bases are used to create a new script that controls agent's behaviour. Each rule is associated with a weight value and adaptation of the rule base proceeds by changing the weight values to reflect the success or failure of the rules in the script. The learning mechanism in this technique is inspired from reinforcement learning techniques where the weights of the rules are adapted depending on their contribution to the outcome of the script. Rules that lead to success are rewarded with weight increase whereas the rules that lead to failure are punished with weight decrease. The key to successful implementation of this approach is design of rule base from domain specific knowledge, knowledge of successful scripts and evaluation function for evaluating the success of the script.

For our work, we draw an analogy for implementing dynamic scripting for content adaptation and shared state distribution. The key here is that the devices are considered as agents where we specify rules based on device capabilities, its usage environment and networking constraints (Figure 1). These domain specific characteristics will be evaluated to select the desired quality of service and content representation most suitable for the device that gets adapted as its usage is changed (e.g. variation in connectivity, battery status, variable load, etc). Specifying and updating these constraints would update the weights associated with rules and would therefore augment the adaptation procedure by generating appropriate

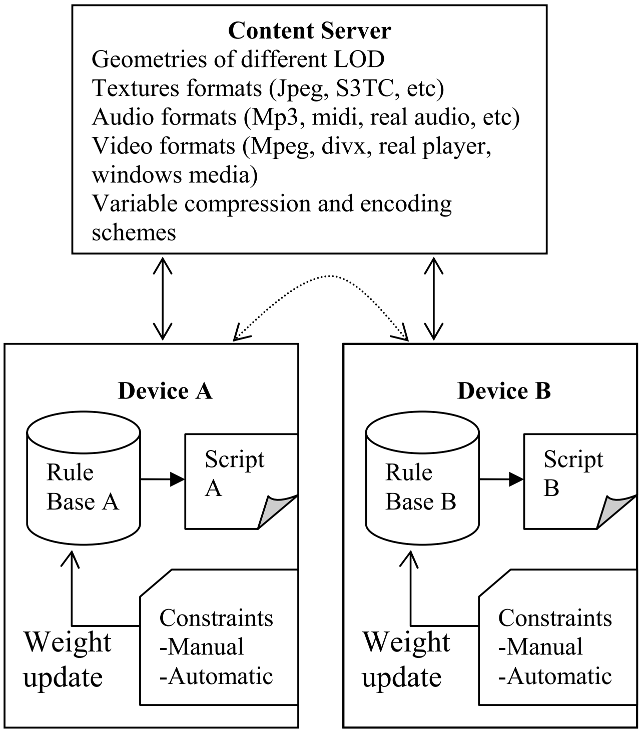

Figure 1 Dynamic scripting for content adaptation and shared state distribution on mobile devices

scripts. For truly mobile games, clients would be moving around in the real world as well as in the virtual world. This movement would clearly affect the underlying network topology. Given such a dynamic networking topology (this may comprise mirror content servers based on client-server approach and sending game update based on P2P approaches) that has to adapt and react to maintain the shared state and consistency of the game, the dynamic scripting approach would aid in selecting the appropriate shared state distribution strategy and decisions for dynamic interest management for efficient distribution, scalability and load balancing. Dynamic scripting has been proven to be very efficient for online adaptation of behaviours in games and requires very few trials for learning and therefore is a good candidate for development of a prototype system to evaluate content adaptation and state distribution for mobile multiplayer games.

4 Conclusions

This paper presented some of the issues related to multiplayer games on heterogeneous mobile devices. We presented several questions that we are trying to address in the context of multiplayer gaming on mobile devices and briefly provide an investigation of networking issues in mobile games, software architectures for supporting online games, approaches for scalable shared state distribution and content representation in games. Further, we discussed the use of dynamic scripting approach as an integral part of the adaptation system that will be used for content adaptation and shared state distribution for mobile games. This would allow for selecting appropriate representation, encoding and transmission scheme for the content depending on the device capabilities and constraints associated with the device. Further evolution of the scripts would be used for efficient shared state distribution and dynamic adaptation of the networking topology. In future we plan to investigate the set of constraints that would form part of the rule base and how weights are assigned to these. We plan to develop a prototype system to test the effectiveness of this technique for mobile devices. This would include some level of detail modelling for geometry, using client-server approach for transferring geometry and using client-sever or peer to peer approaches for sending game updates and maintaining shared state. We also plan to test several protocols such as Bluetooth 2.0, wifi and Wimax for communication between the devices.

References

[1] Alborghetti etal. 2004. "A united scalable framework for on-line gaming". http://www-artemis.int-evry.fr/Artemis/Research/OLGA/OLGA0.html (accessed 10th October 2006).

[2] Chu C. D., Humphrey M. 2004. "Mobile OGSI.NET: grid computing on mobile devices.*Fifth IEEE/ACM International Workshop on Grid Computing.*"pp. 182 – 191.

[3] Smed J, Kaukaranta T, Hakonen H. 2002. "Aspects of networking in multiplayer computer games." *The Electronic Library*, 20(2):87–97.

[4] Yu A, Vuong S.T, 2005. "MOPAR: a mobile peer-to-peer overlay architecture for interest management of massively multiplayer online games. *Proceedings of the international workshop on Network and operating systems support for digital audio and video NOSSDAV '05*" pp. 99 -104.

[5] Budke D, Farkas K etal. 2006. "Real-Time multiplayer game support using QoS mechanisms in mobile ad-hoc networks. *WONS 2006 : Third Annual Conference on Wireless On-demand Network Systems and Services"*. Pp 32 – 40.

[6] Phan T, Huang L, Dulan C. 2002. "Challenge: Integrating Mobile wireless devices into the computational grid. *Proceedings of the 8th annual international conference on Mobile computing and networking"*. Pp 271 – 278.

[7] Andersson A, Thoren M. 2005. "Zigbee, a suitable base for embedded wireless development?" http://db.s2.chalmers.se/download/masters/master_030_2005.pdf

[8] Wibree.com http://www.wibree.com (accessed 15th October 2006).

[9] Wikipedia http://en.wikipedia.org/wiki/Bluetooth (accessed 16th October 2006).

[10] Sreenivas H, Ali H. 2004."An Evolutionary Bluetooth Scatternet Formation Protocol. *Proceedings of the Proceedings of the 37th Annual Hawaii International Conference on System Sciences"*.

[11] Riera S M, Wellnitz O, Wolf L. 2003. "A Zone-based gaming architecture for ad-hoc networks. *Proceedings of the 2nd workshop on Network and system support for games (NETGAMES '03)"*. Pp 72 – 76.

[12] Abichar Z, Peng Y, Chang J M. 2006. "Wimax: The emergence of wireless broadband. *I.T. professional.*"pp 44 – 48.

[13] Fourty N, Val T, Fraisse P, Mercier J. 2005. "Comparative analysis of new high data rate wireless communication technologies "from WIFI to WIMAX. *Joint International Conference*

on Autonomic and Autonomous Systems and International Conference on Networking and Services - (icas-icns'05) ". Pp 66.
[14] Cronin E, Filstrup B, Kurc A. 2001. "A Distributed Multiplayer game server system." http://warriors.eecs.umich.edu/games/papers/quakefinal.pdf (accessed 20th October 2006).
[15] Pellegrino J D, Dovrolis C. 2003. "Bandwidth requirement and state consistency in three multiplayer game architectures. *Proceedings of the 2nd workshop on Network and system support for games (NETGAMES '03)*". Pp 52 – 59.
[16] Singhal S, and Zyda M. 1999. *Netwroked Virtual Environments: Design and Implementation*. Addision Wesley
[17] Morse K. 2000. "An Adaptive, Distributed Algorithm for Interest Management"; *PhD Thesis*, University of California, Irvine
[18] US Defence Modelling and Simulation Office. 1998. High Level Architecture (HLA)- Interface Specification, version 1.3
[19] Macedonia M, Zyda M, Pratt D, Brutzmann D and Barham P. 1995. "Exploiting Reality with Multicast Groups: A Network Architecture for Large-Scale Virtual Environments"; *IEEE Computer Graphics and Applications,* 15(3): 38-45
[20] Miller D and Thorpe J A. 1995. "SIMNET: The Advent of Simulator Networking", *Proc. of IEEE,* 83(8): 1114-1123
[21] Greenhalgh C and Bendford S. 1995. "MASSIVE: A Distributed Virtual Reality System Incorporating Spatial Trading", *Proc. of 15th International conference on distributed computing systems (DCS 95)*, IEEE Computer Society, 27-35
[22] Epic Games 1999. *The Unreal Networking Architecture*. World Wide Web, http://unreal.epicgames.com/Network.htm
[23] Yu A and Vuong S T. 2005. "MOPAR: A Mobile Peer-to-Peer Overlay Architecture for Interest Management of Massively Multiplayer Online Games", *in proc. of International Workshop on Network and Operating systems Support for Digital Audio and Video,* pp: 99-104
[24] Liu E, Yip M and Yu G. 2005. "Scalable Interest Management for Multidimensional Routing Space", *in proc. of the ACM symposium on Virtual Reality Software and Technology,* pp: 82-85
[25] Kumar P and Mehdi Q. 2006. "Recursive Interest Management For Online Games", *in proc. of 8th International Conference on Computer Games (CGAMES),* Louisville, KY, USA
[26] Clark J H 1976. "Hierarchical Geometric Models for Visible Surface Algorithms", Communications of the ACM, 19-10, 547-554, October 1976.
[27] MPEG (Moving Picture Experts Group) – 4 [Online]. Available: http://www.chiariglione.org/mpeg/standards/mpeg-4/mpeg-4.htm
[28] ISO/IEC 15444-1:2000: Information technology – JPEG 2000 image coding system – Part 1: Core coding system.
[29] Real Video: [Online] Available: http://en.wikipedia.org/wiki/RealVideo
[30] DivX: [Online] Available: http://en.wikipedia.org/wiki/Divx
[31] Katz R H 1994. Adaptation and mobility in wireless information systems. *IEEE Personal Communications*, 1(1):6–17, 1994.
[32] E. de Lara, D. S. Wallach, and W. Zwaenepoel. 2001 Puppeteer: Component-based adaptation for mobile computing. In *Proceedings of the 3rd USENIX Symposium on Internet Technologies and Systems*, San Francisco, California, Mar. 2001.
[33] Fox A., Gribble S. D, Chawathe Y, and Brewer E. A.1998. Adapting to network and client variation using infrastructural proxies: Lessons and perspectives. *IEEE Personal Communications*, 5(4):10–19, Aug. 1998.
[34] Lum W. Y.and. Lau F. C 2002. A context-aware decision engine for content adaptation. *IEEE Pervasive Computing*, 1(3):41–49, July 2002.
[35] Narayanan D., Flinn J., and Satyanarayanan M. 2000. Using history to improve mobile application adaptation. In *Proceedings of the 3rd IEEE Workshop on MobileComputing Systems and Applications*, Monterey, California, Dec. 2000.
[36] WAP Forum. Wireless application protocol architecture specification, Apr. 1998. Available at: http://www.wapforum.org/what/technical/arch-30-apr-98.pdf.
[37] Mohomed I, Cai J H, Chavoshi S, Lara E 2006. Context-Aware Interactive Content Adaptation. *In Proceedings of the 4th International conference on Mobile systems, applications and services*, Uppsala, Sweden
[38] Smith J. R., Mohan R., and. Li C.-S 1998. Content-based transcoding of images in the Internet. In *Proceedings of the IEEE International Conference on Image Processing*, Chicago, Illinois, Oct. 1998.
[39]. Smith J. R, Mohan R., and Li C.-S.1998. Transcoding internet content for heterogeneous client devices. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, Monterey, California, May 1998.
[40] Bickmore T. W.and Schilit B. N. 1997. Digestor: Device-independent access to the World Wide Web. *Computer Networks and ISDN Systems*, 29(8–13):1075–1082, 1997.
[41] Spronck P. 2005. "Adaptive Game AI"; *PhD Thesis*, University of Maastricht, The Netherlands