Brazilian Journal of Science and Technology

a SpringerOpen Journal

RESEARCH

Open Access

CrossMark

A new approach in mobile gaming on cloud-based architecture using Citrix and VMware technologies

Rajesh Bose[*] and Debabrata Sarddar

*Correspondence:
bose.raj00028@gmail.com
Department of CSE,
University of Kalyani, Nadia,
West Bengal, India

Abstract

Without doubt the lure of high resolution videos along with a deeply engaging user interface is an irresistible attraction for almost all gamers across the world. Sales of video games are projected to cross more than US$ 110 billion in the coming years. With Internet speeds in certain countries climbing into the 3G, and the newer 4G, spectrums, multi-gaming sessions have found rising number of followers all across the globe. However, a sizeable chunk of the gaming community does not have a sustained access to high speed network. Our paper proposes a game manager which harnesses commercially available virtualization applications to offer the best game streaming experience anywhere in the world. We discuss a model which would allow users requesting multi-player game sessions to be played over the Internet, to connect to the appropriate game cloud for the fastest possible game streaming experience. End users would be able to play their favourite games connected over the Internet without having to worry about extensive hardware requirements.

Keywords: Video games, Virtualization, Multi-player game, Internet, Streaming

Background

Introduction

While the concept of cloud gaming itself is not novel, the increasingly high demands for playing games across a range of hardware platforms has driven up the market for hosting games on the cloud. The smart phone mobile market today is fast asserting itself as the bedrock for games that most users enjoy. Our proposed work aims to build on the idea of cloud computing (Zaharia 2009) and design an architecture based on the client–server game concept. The central idea behind this paper is to explore commercial tools available to host games on a server over 3G mobile networks. A mobile phone user connected on a 3G mobile network would not have to download an entire game to play. In fact, he/she would be able to request single or multi-player sessions with relative ease and without having to use up sizeable amount of memory space in order to install a game to begin playing it. We bring together two robust and eminently powerful technologies in tandem to allow mobile users on a 3G network to enjoy gaming experience without the need to install game applications on mobile devices.

Mobile cloud video gaming

As an emerging trend in the gaming industry, Gaming-on-Demand technology has become one of the most discussed topics. Gaming-on-Demand makes use of cloud computing as it fundamental basis on which the framework operates. Leading the commercial segment in this field are companies such as OnLive, Gaikai, and G-Cluster. These companies host almost all their games on their own supported cloud computing networks. These companies have adopted almost a common fundamental method to deliver game play to their respective customers. In this method, video games do not quite run on the end-user's device. Private servers hosted in cloud networks run the video games. A string of continuous frames are then transmitted over the internet to the computers, or even to interactive television sets on which the users are able to play video games. The user's responses and video game interactions are sent back to the game server hosted on the cloud, via the same network. A mobile cloud video gaming (MCVG) technology is essentially a Gaming-on-Demand model that has been adapted and put into operation for mobile devices with wireless communication capabilities. In order to support mobile gaming on clients which are not required to process game play on their own, two integral components comprising the game engine and game streaming servers in the form of virtual machines are required. Whenever a user logs on to a game server from a mobile gadget, the game content server would first confirm the connection. Subject to successful confirmation, it would then begin to initialize a game engine server and examine the dynamics affecting the quality of service (QoS) which influences the quality of mobile cloud gaming (MCG) experience. The client's account information and game data are then pulled in by the game engine server. From this point on, the processing of the game flow and corresponding user data begins. The resultant streaming output, influenced by the game logic, is continuously translated into game video. Converted by the game streaming server, the game video which is considered to be a raw format for delivery is sent through appropriate wireless communication channels to the mobile client on which the game is to be actually played. For the responses of the user to be channelled to the game engine, the user's continuous feedback using the buttons and sensors of the mobile device, is sent to the cloud server. The feedback which is continuously streamed during the course of the game play is fed into the game content server directly (Wang and Dey 2009). Fundamentally, combined with MCVG, the cloud assumes the role of a video generator which continuously communicates, and a server that streams video frames of the game being played. The mobile devices are simply controllers of the game events in play, and receive streaming videos of the game play. Owing to certain limitations imposed by features available for mobile communication networks on a viable commercial scale, video-rendering technologies, video compression, and QoS transmission controls assume enormous significance in designing of a MCVG system (Wang and Dey 2010; Cai and Leung 2012). As far as we have researched, the Android app version of OnLive was the first MCVG commercial application available on Google Play Android App market.

Mobile browser gaming

Browser-based games have already amassed a considerable fan following. Many of these games are to be found on social networking sites, and are enjoyed by users across the

globe. Such games have the following features (Vanhatupa 2010) which are inherent to the development and existence of the games on browser-based platforms:

1. Support for multi-player game sessions: a large number of concurrent game playing sessions can be enjoyed by users with access to browser-based games.
2. Web browser as client: games which are played on web browsers do not require software to be installed. Although some browser-based games may need appropriate browser add-ons to function properly or at all.
3. Always on: with a cloud-based game actually being hosted and processed on cloud computers, it is possible for a game session to continue and associated messages received even when a player is not connected to the Internet.
4. Different time scales: a browser-based game can be real-time, turn-based, or might execute actions at set intervals within a given time period based on preset queued commands received.
5. Every player has a unique account with which controlling of a game character role is made possible. Such a character role can be akin to that of a role-playing game, turn-based games—such as strategy games where entire nations or groups may be controlled, or real-time games. Browser-based games can be of different genres. These can range from the ones already mentioned, to manager games and social networking types. The common trait of all browser-based games is the web browser itself which acts as the container within which every browser-based game is played. Simultaneously with the evolution of mobile devices, progress is being made towards development of games that can be played on mobile browsers (mobile browser-based games or MBG).

Cloud gaming issues and challenges

A cloud gaming architecture has to integrate a wide array of cutting edge technologies in order to deliver optimum performance levels. These technologies include video streaming that offer low response times, to 3D rendering methods which are able to support extraordinary operations. Our analysis starts by examining the key design issues and considerations that most cloud game operators have focussed their attention. A typical cloud gaming architecture comprises a system or a collection of systems that work together to collect users' inputs, conveying these signals to the cloud server or servers, process the signals into relevant game actions, condense and deliver the encoded/compressed information to the game play, and then transmit the game scenes to the mobile device user. All of these operations have to take place in the least amount of time usually measured in terms of milliseconds or less. The less time it takes to complete processing these operations, smoother is the interactive session that a user would be able to enjoy. Naturally, this period of time—called as interaction delay, is to be maintained at the lowest possible mark so as to ensure a gaming experience that is vibrant and highly responsive to all connected players of a given cloud game. There are, however, certain unavoidable consequences to such a design. With shorter latency times, the entire cloud gaming system has to process vital screen rendering and video compression operations within a very narrow range of duration. In addition to this, high network latency can

leave a negative impact on the gaming experience and user interaction. Keeping this in view, we commence our discussion on designing a model for delay tolerance.

Video streaming

Streaming requirements for live video telecast and cloud gaming are near identical. In both cases, incoming video feeds have to be quickly encoded for proper viewing by users, and compressed in a very short span of time prior to transmission. Further, encoding is required and must be accomplished for a very few frames which represent a small number of the most recent video frames.

However, major differences can be observed between the two. Cloud gaming almost nearly offers no capability to buffer video frames on a client device. Live media streaming, on the other hand, does have the advantage to do so. The reason cloud gaming does not offer buffering of video is because whenever a player performs an action on the local client, the command has to travel through the Internet to the cloud system. This command must then be processed by the game engine and the resultant video frames in relation to the game generated. These video frames are then compressed by an appropriate video codec and transmitted back to the player. Considering the incredibly short duration—within 100–200 ms—it takes to complete this cycle, it becomes obvious that there is infinitesimal amount of time to buffer. In contrast, live media streaming can extend a buffer in the order of hundreds of milliseconds to even a few seconds without compromising QoS.

Our work features Citrix XenApp 6.5 with HDX features. This enables us to take advantage of superior streaming capabilities. Citrix HDX projects capabilities that deliver high definition video streaming to any device over any network. We have used Citrix HDX to help our users enjoy improved graphics performance when connecting to virtual desktops using the Citrix Independent Architecture (ICA) protocol. This technology is appropriate in applications that involve a large number of pixels changing at a high frequency rate. To make desktop virtualization both scalable and practicable regardless of low bandwidth and/or high latency wide area network conditions, ICA is included with Citrix HDX tools.

Related works

Video rendering for 3D games

Understanding the bandwidth constraints even in high-speed mobile networks for rendering 3D games has been the bedrock of the paper (Lu et al. 2015). The authors propose a revolutionary model which transforms graphics rendering factors into predictable metrics based on which a satisfactory level of user experience can be achieved. What is most significant is that the authors demonstrate how a regression analysis approach can be used for solving video bitrates leading to a positive impact on video rendering abilities of the proposed model. However, the authors conclude that in case of certain games, a 2D rendering approach may be better than 3D. The authors have identified that the asymmetric video and graphics settings may cause discomfort as a result of constant video frequency fluctuations.

Intelligent partitioning for improving latency

Quality of Experience (QoE) is an yardstick by which online gaming experiences are measured. Noting this, the paper (Cai et al. 2013) explains the need for a partial offline execution method, and an intelligent resource allocation framework that can support, what the authors call, "click-and-play" concept. Recognizing the need to improve latency, the authors have conducted several experiments to demonstrate the feasibility of resource allocation that is aware of the fluctuating environment conditions such as networks involved, and the physical characteristics of the devices on which games are to be played. While the proposed approach holds promise, the authors do not address the actual key element of the mobile network is needed to support the intended latency objectives. Further, the model proposed relies heavily on the use of only one technology. These, in our opinion are limiting factors which detract from an otherwise an interesting concept.

Distributed gaming platform

Improving latency through a distributed cloud infrastructure (Kamarainen et al. 2014) has been the impetus for the authors to bring down latency to sub-100 ms levels. The paper proposes a model in which game servers are positioned as close as possible to the users in a bid to shorten response time. The entire concept hinges on the ability to distribute games and gaming servers in a manner such that the shortest distance between a user and the nearest gaming server is kept to the minimum. In our view, while this approach deserves to be highlighted, it is not without a flaw. In most parts of the world, a distributed architecture as the one proposed in this paper, may not be practicable considering commercial interests and technical hurdles. For instance, it may not be possible for a game manufacturer to provide servers that host games at data centers for all ISPs in a way such that users from different geographical locations, covered by these ISPs, have similar levels of user experience when playing an online game.

Measuring the quality of service of cloud gaming platforms and services

As a framework that allows assessing the quality of service achieved by a cloud gaming system, the approach (Chen et al. 2014) to tap into the flexibility of binary-level function intercepting mechanisms is an entirely novel concept. While major commercial online game service providers are extremely reluctant to share details relating to configuration of servers and the composition of users, the paper offers an invaluable tool whereby almost any online cloud gaming solution may be tuned with the aim of optimization. From a perusal of the paper, an interesting observation may be made on the subject of acceptable bandwidth range within which two of the world's most popular online cloud gaming services are able to operate to provide quality levels of user experience. Research conducted indicates that 2 Mbps is the lowest limit within which one of the better known online cloud gaming service is able to operate with slightly degraded performance.

The VNC approach towards massive multiplayer online games

Although this paper (Chuah et al. 2014) deals with the concept of green computing in the context of cloud gaming services, the authors have conducted research and

experiments with open source online gaming technology, and VNC. As correctly noted by the authors, VNC offers the advantage of platform-independent operations. However, in contrast to its more specialized open source online gaming counterpart, VNC has been found to be better suited for games which involve limited motion involvement and extremely frequent changes in game scenes.

Our literature survey sheds light on some interesting aspects which ultimately have gone on to form the bedrock of our own proposed work. We have been able to distinguish the fact that achieving a consistent lowest possible latency is a guarantee in elevating and maintaining a positive user gaming experience. This is dependent on several factors—chief of which is the average network speed. Although, literature exists on proposing new and skilful techniques to harmonize video refresh rates with physical hardware devices at both ends involving users and service providers, there is no universal framework that can offer a stable and reliable user experience over a network bandwidth that may not average any more than 3G speeds realistically available in some parts of the world.

Proposed work

The cornerstone of our proposed work is a Game Manager (GM) which is supported by a robust structure composed of Citrix XenApp and VMware virtualization technologies that offer industrial-strength reliability and high performance. The role of the GM is to accept user requests from mobile users and connect the user with a game session hosted on the nearest cloud node server best suited for the purpose. Figure 1 shows the management of game streaming sessions within a cloud system using GM. The games themselves would be installed on virtual machines hosted on VMware ESX server and published using Citrix on each cloud node. The URL link to the games would be published on a SSL-based device for a secure connection. These links would be registered

Fig. 1 Pictorial representation of Game Manager centric mobile cloud gaming architecture
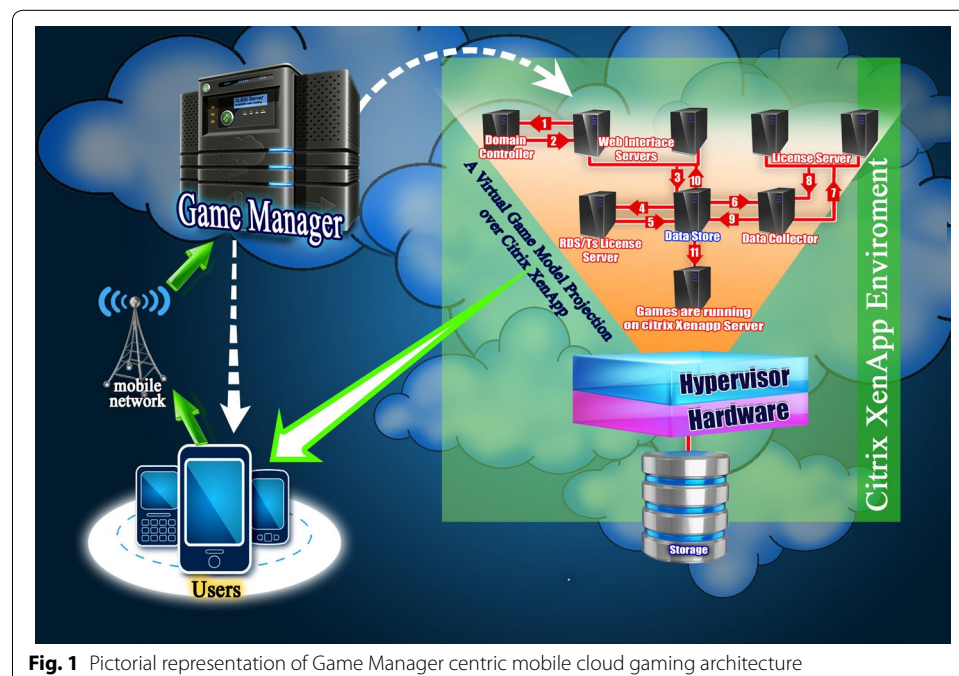
with our proposed GM. In other words, each cloud game node would have its own set of URLs registered on the GM. The GM would maintain an active database consisting of the records of these URLs and response times to each of the cloud nodes. When a mobile user sends a request to play a game, the GM seeks out the best fit cloud node in terms of best-fit response times and availability of the game session or sessions in case of multi-game requests. Once the URL has been identified by the GM, this link is sent to the user and a game session is directly established between the mobile device and the cloud node on which the game is published. Once a user disconnects or terminates a game session, an appropriate feedback is sent from the mobile device to the GM notifying that the resource is free and that the URL resource is available and ready to serve other incoming requests.

We present the entire architecture of virtualizing game server with help of VMware ESX server and eventual publication of application through Citrix. It is a very sturdy design for cloud-based games. In this design we have installed Citrix XenApp inside a distributed architecture scheme on a cluster of virtual machines. These virtual machines are hosted and managed on a VMware ESXi server. We install the games on Citrix XenApp and publish the URLs of the games. Figure 1 shows the pictorial representation of Game Manager centric Mobile Cloud Gaming Architecture These URLs are in turn made public through a SSL-based device for secure connection. Next, these URLs are then registered with our proposed GM. In this manner, each and every cloud node registers its published game links with the GM. The game cloud nodes routinely send periodic information to the GM to keep its table of records and historical data updated at all times.

The updated table enables the GM to select the appropriate game server based on user demands and resources available. While playing, resources allocated for the client remain active. When a customer terminates the game or is disconnected, a message with the request for disconnection is sent to the GM (the first case, a request to disconnect) or status reports are no longer received (in the second case, inadvertent disconnection).The GM maintains another list of active sessions and updates its state whenever it receives the report from the client. Upon receiving a request to close the session or when stipulated time limit has been exceeded, the GM updates its session table accordingly, so that resources are released and the same can then be used and made available for subsequent requests. In this manner, games are processed on servers controlled by the service provider and client devices need only basic computer capacity and a network access to play. With the introduction of WiFi communication being an integral component of most mobile devices, network performance and, therefore, users' game play experience also undergo changes. Telecommunication companies are now able to focus their efforts towards pushing more application and game services that would have otherwise been a challenge.

Since, our proposed model has been designed on Citrix XenApp 5, the internal workings require explanation (Isha and Kant 2013) in this context. The designated numbers 1 through to 11, in Fig. 1, represent the individual processes that take place when a request is placed at Citrix's web interface server. Briefly, these steps are as follows:

- The Citrix web interface transmits the request to an Active Directory (Domain Controller) server.
- The credentials of the user is matched by the Active Directory Server and returned to the web interface servers.
- The Web Interface servers then look up for the game application at the Data Store.
- Subsequently, the request is passed on to the Remote Desktop Services/Terminal Services servers where the licenses are verified.
- Following verification, the Data Store receives the confirmation from the Remote Desktop Services/Terminal Services servers.
- Data Store then transmits the request to the Data Collector.
- It is the job of the Data Collector to communicate with the license servers to check whether the game application licenses are valid.
- After verification, the Data Store receives the authenticated request from the License Servers.
- The Data Store then sends a confirmation message to the Web Interface server which allows access to the user to play the game from the mobile server.

Citrix XenApp allows application access from a number of assorted devices (Isha and Kant 2013). This aids in flexibility, enhancing security, and scalability. The Citrix infrastructure shown in our diagram is an adaptation from the research (Isha and Kant 2013) conducted in resolving resource utilization issues.

Algorithm

1. Game Manager stores all the information about the game cloud nodes, viz., IP Address/URL Address, shortest node distance, available game list including single user games and multi user games, online available game users, active session lists.
2. After receiving the request from client the GM would scan its Table.
3. User selects the game and game type, and this request is sent to the GM.
4. If user selects a single user game then.
    4.1   General Manager would search for available suitable game cloud node for this request.
5. If the user choose multi user game then.
    5.1   General Manager would ask the user to select the available online game user from his table, and GM would now take the request and search the available game cloud node.
6. All cloud nodes will send a information to GM at timed intervals such that at any given instant, the following data are readily accessible and available based on which the GM would be able to route user requests, accordingly.
    6.1   Bandwidth capacity.
    6.2   Shortest cloud node distance.
    6.3   Time to live.
    6.4   Available resources and their capacity.
7. Both of the above information sets varies time to time and also from one area to another based on the following parameters:

    7.1   Number of cloud regions available to serve the game request.

    7.2   Strength of bandwidth of identified cloud regions.

    7.3   Shortest nodal distance within the cloud regions that can effectively reduce turnaround time between user's PC and the cloud region serving the game application at HA levels.

8. Now for a given point in time, compare resources and the corresponding capacities.

9. All cloud nodes are sorted in descending order based on nodal points that have the highest resources to serve the game requests.

10. If the resource of a set of cloud nodes are available.

    10.1 choose the maximum resource capacity from set of cloud nodes.

    10.2 Compare the bandwidth. Choose the highest bandwidth.

    10.3 Choose the shortest distance cloud node from set of cloud nodes to handover point.

        10.3.1 If resource capacities of two Cloud nodes to handover are the same,

            10.3.1.1 Compare the bandwidth. Choose the highest bandwidth between them.

            10.3.1.2 Consider the shortest distance cloud node from set of cloud nodes to handover.

            10.3.1.3 Compare the recent TTL (time to live) histories of two cloud nodes.

            10.3.1.4 Compare the number of available virtual machines supporting game modes at HA levels, in each cloud nodes.

    10.4 Else.

        10.4.1 Wait for refresh time for getting available resources.

11. Repeat step 10 every time while choosing a new cloud storage node to handover.

12. After choosing the nodes that can satisfy the user request, the GM would send back the user a URL Address or IP Address for the appropriate game.

13. User would need only to click on the link and make a connection with the virtual game sever directly.

14. Virtual server processes the remote view of the graphical desktop and sends it to the users via a secure link.

15. Users now start the game and pass the keyboard commands to control the game for the virtual server.

16. If a customer terminates the game or is disconnected, the cloud node will then sends a message to the GM with the request for disconnection notify the game session has ended. Resources of the virtual machine can then be released and become available for provision for a new session.

17. To complete the cycle, the GM updates its table for accepting further incoming requests to establish game connections.

Results

We run a game on an ESX based VM. The VM is running a Citrix XenApp version 6.0. The game is installed and published by this XenApp virtual server (Fig. 2).

    This section portrays the results of implementing the work. The effects were analyzed (Fig. 3) with respect to network bandwidth requirements for an interactive game session, latency times for commands sent from the client device, and the frame count per second

Fig. 2 The game running on Citrix XenApp that is installed on VM
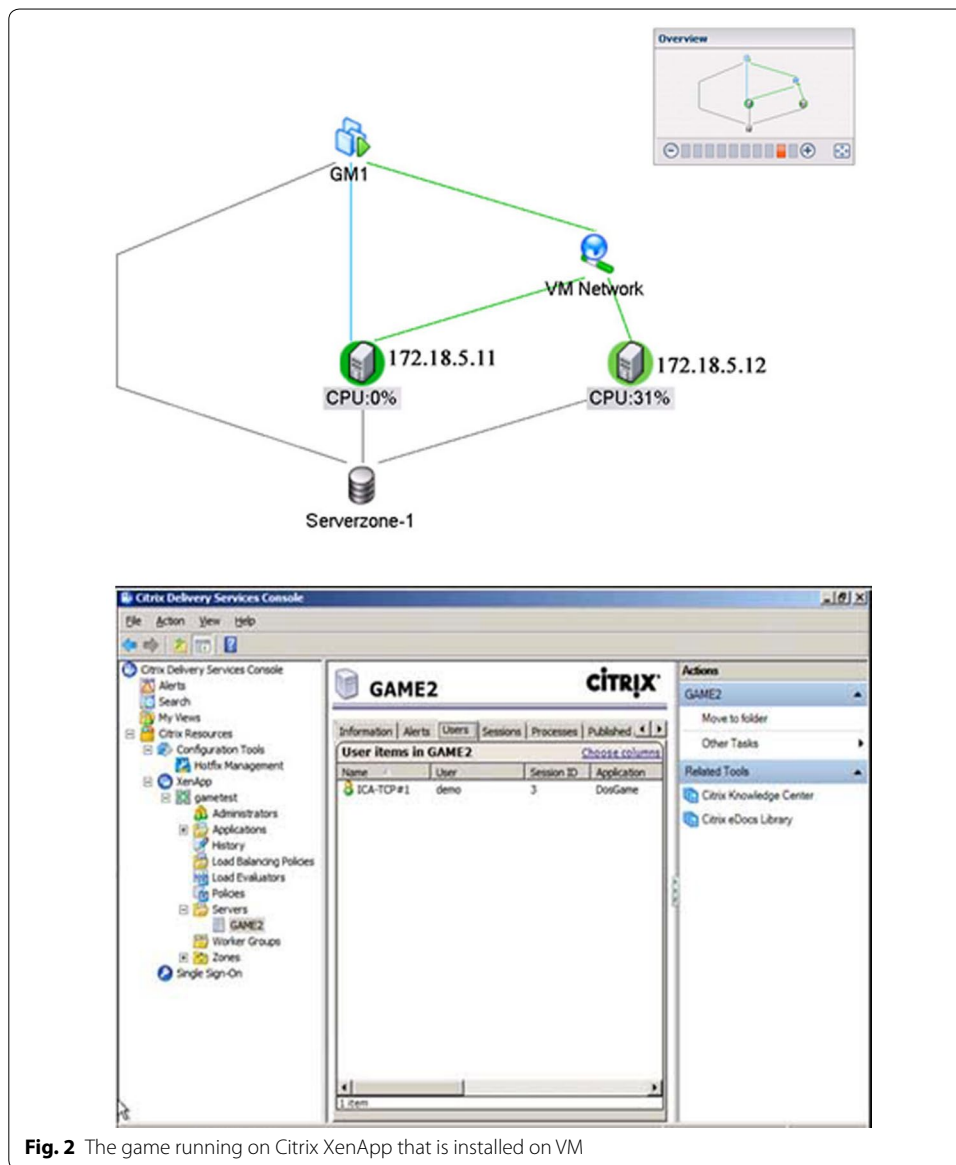
obtained. For displaying of the game scenes, the bandwidth use was examined vis-à-vis the amount of bytes transmitted per second to the client by the server. Each game screen which is ready for transmission is sent to the client in packets of 512 bytes each. Implementing a screen resolution size of 320 × 240 pixels, every image is sent into segregated packets numbering 28 of 512 bytes per packet. This translates into 14 kb per image transmitted. To achieve an optimum performance of 30 frames per second, it requires a transmission rate of around 420 kb/s. That translates into almost 3.3 Mb/s requirements for network bandwidth support capability. A figure that is acceptable considering capacity of a local network as shown in Fig. 4.

Transmission rates were down close to ten frames per second, from the network, for a mobile client. This, therefore, has prompted for improvements and changes that are still required to be made. For satisfactory results during gaming sessions, the importance of
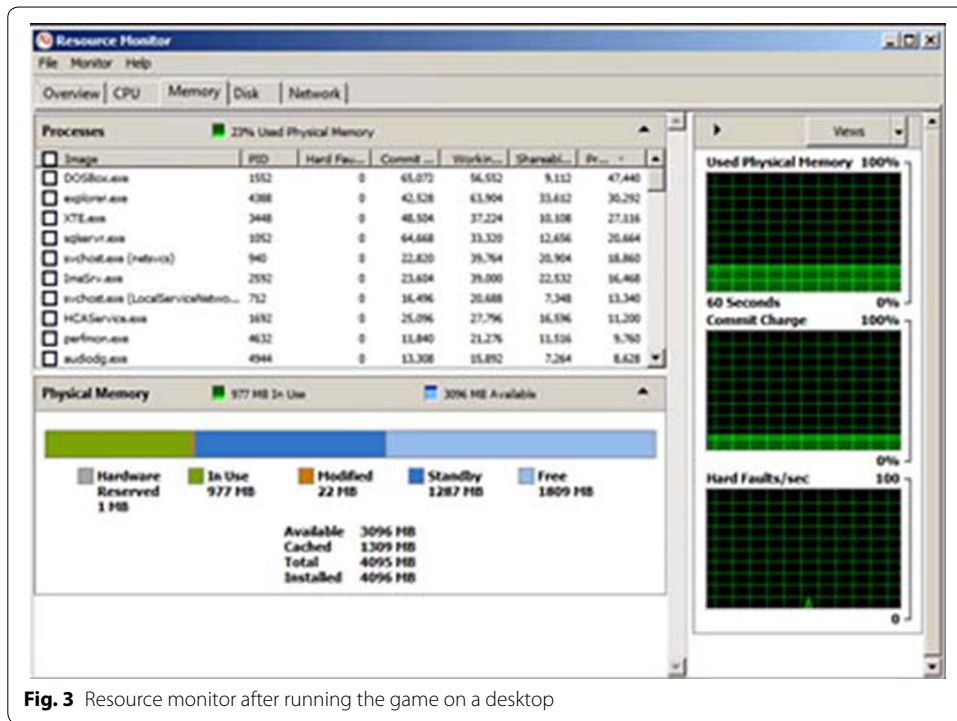
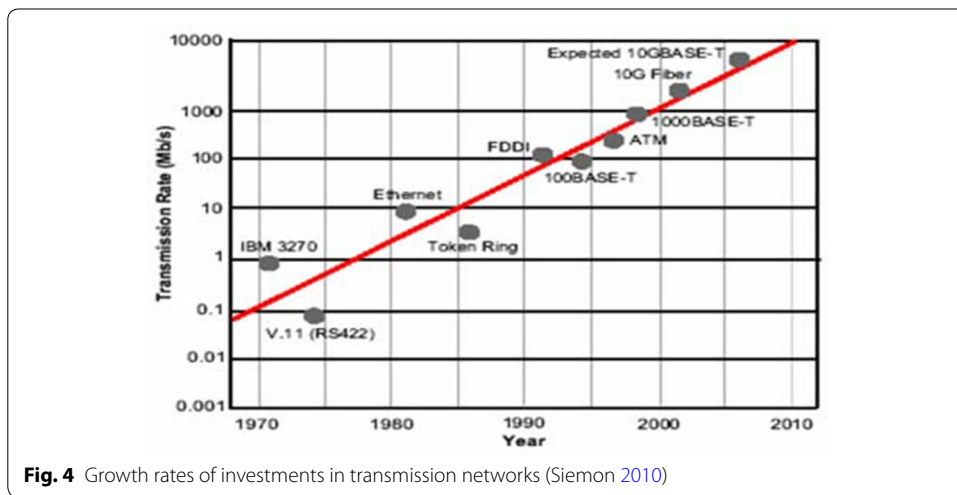**Fig. 3** Resource monitor after running the game on a desktop

Fig. 4 Growth rates of investments in transmission networks (Siemon 2010)

system response times in relation to user inputs is just as great as the number of frames per second that is rendered. A response time is defined as the period from the point of time that a user input is transmitted over the network, to the point where the result is rendered on the client device post processing of the user commands by the server subject to the logic of the game. While the desktop client was subject to a delay averaging 30 ms in response times, the mobile version reflected a pattern that ranged between 800 and 1000 ms.

Our experiments were purposely conducted on devices that can support 320 × 240 pixels. We intended to find out how the use of Citrix XenApp technology and ICA would support a possible mix such as the one discussed in our paper. We are of the opinion that

further research in tuning Citrix technologies could be initiated in this direction. While this delay of 800–1000 ms delay can be improved upon in future works, it may not be out of place to mention here that such delays can be attributed to real-life average speeds experienced in networks which claim 3G speeds in certain regions.

The values depicted here can be improved upon in future by exploiting features provided by mainstream mobile operating systems and investigating more effective methods to send and receive data. Tests discussed in our work were performed based on the reliable and proven TCP layer and packet switching network. It is assumed, on the basis of documented features, that use of UDP, which though does not guarantee transmission of packets reliably—can lead to significantly improved outcomes insofar as frame transmission rates per second delivered over a network is concerned. Response times, too, are expected to be better.

Conclusion

Our model aims at delivering flicker-free content video gaming experience that is supported using Citrix and VMware technologies. However, the prime underlying mechanism of selecting the best fit server from the cloud ecosystem is our proposed GM. Using commercial available tools at our disposal, we have demonstrated how GM can help in choosing the appropriate published link and deliver game content to mobile devices. Future work could involve a more in-depth study on how to quickly identify cloud resources that can deliver optimum content for mobile device users of a given region. Today, 3G networks are still to be a standard across all third world countries. In coming days, studies could also progress on how to deliver qualitative better gaming experience on mobile devices over 2G speeds. However, till such time development work and studies are conducted in that direction, the enormous amount of market potential would remain fully unutilized.

In its present form, our proposed framework lays groundwork for future research using Citrix and VMware technologies. In our view, this model shows promise and is a harbinger for further advanced studies in the field. The scope of the GM, at this point, hinges on availability of 3G networks. But future advancements in lowering effects of latency while delivering high resolution games on 2G, could help mobile service providers to provide overall satisfactory mobile cloud gaming experience.

Authors' contributions
RB and DS contributed to this work in the manuscript preparation. Both authors read and approved the final manuscript.

Authors' information
Rajesh Bose is currently pursuing Ph.D. from Kalyani University. He is an IT professional employed as Senior Project Engineer with Simplex Infrastructures Limited, Data Center, Kolkata. He received his degree in M.Tech. in Mobile Communication and Networking from WBUT in 2007. He received his degree in B.E. in Computer Science and Engineering from BPUT in 2004. He has also several global certifications under his belt. These are CCNA, CCNP-BCRAN, and CCA (Citrix Certified Administrator for Citrix Access Gateway 9 Enterprise Edition), CCA(Citrix Certified Administrator for Citrix Xen App 5 for Windows Server 2008). His research interests include cloud computing, wireless communication and networking. Debabrata Sarddar, Assistant Professor in the Department of Computer Science and Engineering, University of Kalyani, Kalyani, Nadia, West Bengal, INDIA. He has done Ph.D. at Jadavpur University. He completed his M. Tech. in Computer Science and Engineering from DAVV, Indore in 2006, and his B.E. in Computer Science and Engineering from NIT, Durgapur

in 2001. He has published more than 75 research papers in different journals and conferences. His research interest includes wireless and mobile system and WSN, cloud computing.

References
Armbrust M, Fox A, Griffith R, Joseph A, Katz R, Konwinski A, Lee G, Patterson D, Rabkin A, Stoica I, Zaharia M (2009) Above the clouds: a Berkeley view of cloud computing. Disponível http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-28.pdf
Cai W, Leung V (2012) Multiplayer cloud gaming system with cooperative video sharing. In: Proceeding of the 2012 international workshop on emerging issues in clouds (in conjunction with 4th IEEE international conference on cloud computing technology and science), IEEE, Taipei, CloudCom 2012
Cai W, Zhou C, Leung V, Chen M (2013) A cognitive platform for mobile cloud gaming. In: IEEE International Conference on Cloud Computing Technology and Science 2013
Chen K-T, Chang Y-C, Hsu H-J, Chen D-Y, Huang C-Y, Hsu C-H (2014) On the quality of service of cloud gaming systems. IEEE Trans Multimed 16(2):480–495
Chuah SP, Yuen C, Heung NM (2014) Cloud gaming: a green solution to massive multiplayer online games. IEEE Wirel Communications. 21(4)
Isha KC (2013) Highly Available XenApp Cloud. Int J Inform Technol Converg Serv (IJITCS) 3(3):13–19
Kamarainen T, Siekkinen M, Xiao Y, Yla-Jaaski A (2014) Towards pervasive and mobile gaming with distributed cloud infrastructure. In: IEEE Network and Systems Support for Games (Net Games), 2014 13th Annual Workshop
Lu Y, Liu Y, Dey S (2015) Cloud mobile 3D display gaming user experience modeling and optimization by asymmetric graphics rendering. IEEE J Sel Top Signal Process 9(3):517–532
Siemon (2010) The cabling life cycles and the laws of communications networks. pp 1–12. http://www.siemon.com/br/whitepapers/10g-assurance.asp
Vanhatupa J-M (2010) Browser games: the new frontier of social gaming. In: Recent trends in wireless and mobile networks, vol 84: Communications in computer and information science. Springer, Berlin, pp 349–355. doi:10.1007/978-3-642-14171-3-30
Wang S, Dey S (2009) Modeling and characterizing user experience in a cloud server based mobile gaming approach. In: Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE, New York, 30 2009–Dec 4 2009, pp 1–7
Wang S, Dey S (2010) Rendering adaptation to address communication and computation constraints in cloud mobile gaming. In: Global Telecommunications Conference (GLOBECOM 2010), 2010. IEEE, New York, Dec 2010, pp 1–6