School of Computing
FACULTY OF ENGINEERING

UNIVERSITY OF LEEDS

Cloud Gaming and Simulation in Distributed Systems

Khen Cruzat

Submitted in accordance with the requirements for the degree of Computer Science

2015/16

The candidate confirms that the following have been submitted.
<As an example>

Items	Format	Recipient(s) and Date
Deliverable 1, 2, 3	Report	SSO (DD/MM/YY)
Participant consent forms	Signed forms in envelop	SSO (DD/MM/YY)
Deliverable 4	Software codes or URL	Supervisor, Assessor (DD/MM/YY)
Deliverable 5	User manuals	Client, Supervisor (DD/MM/YY)

Type of project: ____________________

The candidate confirms that the work submitted is their own and the appropriate credit has been given where reference has been made to the work of others.

I understand that failure to attribute material which is obtained from another source may be considered as plagiarism.

(Signature of Student) ____________________

Summary

Cloud computing attempts to enable access to high-end graphics intensive games to a wider audience by using powerful data centers. The process of delivering cloud gaming includes processing the user input for the game engine then encoding this as video and streamed to the end user. Due to another added layer of complexity, other problems arise and the aim of this project is to tackle networking issues in the cloud to reduce the latency experienced by the user.
The Software-Defined Networking paradigm allows the management of network technology without the need of touching individual switches. This project will discuss the effects of using SDN for cloud gaming in a distributed system.

Acknowledgements

<The page should contain any acknowledgements to those who have assisted with your work. Where you have worked as part of a team, you should, where appropriate, reference to any contribution made by other to the project.>
Note that it is not acceptable to solicit assistance on ‘proof reading’ which is defined as the “the systematic checking and identification of errors in spelling, punctuation, grammar and sentence construction, formatting and layout in the test”;
see http://www.leeds.ac.uk/gat/documents/policy/Proof-reading-policy.pdf.

Contents

Chapter 1

Introduction

1.1 Context

Cloud computing has always been seen as a way of improving compute performance by the use of multiple computers connected to each other. The games industry has rapidly evolved along the years and a great deal of demand is apparent. Developers of games have pushed computer hardware to meet the needs of consumers for more complicated games and realism. Even with computer hardware becoming cheaper and more of a commodity, the costs of driving graphics rendering for high-end games to run at the optimal settings of 1080p at 60fps are still relatively high.

1.2 Project Aim

The aim of the project is to produce a solution which uses Software-Defined Networking to reduce the network latency in a network.

1.3 Project Objectives

- A simple game program, that is computationally expensive enough to not perform optimally on a single machine (simple flight simulator with real time procedurally generated trees).

- A simplified cloud gaming system where the game created is launched on the cloud and input on the client side in the form of button presses on the keyboard is sent to the game on the server. The game frames produced are then sent to the client's screen.

- Produce a virtual network with simulated cloud game traffic and delay. With the use of SDN, reduce latency in the network.

1.4 Deliverables

The deliverables of the project include:

- Code that demonstrates a simple game/simulation rendering graphics on a server and controlled by a client remotely.

- A manual on how to setup the client and server will be produced so the cloud gaming system can be easily setup and launched.
- Code that will create a virtual network as well as a manual on how to set it up and run the code.
- Project report that explains the problem the project is trying to solve and the schematics of the solution produced as well as an evaluation of the solution.

1.5 Project Schedule

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Chapter 2

Background Research

2.1 Problem Overview

Compute power of games can be offloaded to a server of much powerful computers and then streamed to a client with lower specification hardware such as a laptop or a mobile device. With this comes risks and shortcomings that need to be factored. One of these problems are the latency of the game. Latency can be huge factor in the gameplay such as high-paced games like first-person shooters or fighting games. The delay in pressing a button on the gamepad to seeing the action performed on the screen needs to be kept to a minimum. This idea of interaction delay tolerance being different from genre to genre of games is discussed by Shea et al. [15]. As stated above, a player of FPS games can only tolerate the least which is around 100ms whereas Role playing game (RPG) gamers can tolerate around 500 ms.

Another problem that is directly linked to delay in the system is the effect of packet loss. As stated in the *Eight Fallacies of Distributed Computing* [2], it should be assumed that latency is never zero as mentioned above as well as network is not always reliable. This means that packet loss can occur which in terms of cloud computing can mean the degradation of image quality. In the investigation conducted by Jarschel et al. [6] in which they surveyed average consumers about the importance of packet loss and delay. Generally the quality of the video streamed to the clients plays an important role as the participants were open to using such as a service if provided in good quality.

2.2 Cloud Computing

According to the National Institute of Standards and Technology [12], cloud computing is a means of providing on demand access to computing resources over the network. This should be executed with minimal management effort or service provider interaction. With the shared computing resources, cloud computing aims to process larger data and solve large scale computation.

This cloud computing model can be separated in to three different service models: Software as a Sevice (SaaS), Platform as a Service (PaaS), Infrastructure as a Service (IaaS) [5]. 'SaaS' model attempts to eliminate the need to install and run the application on the user's system. An example of this is the Microsoft Office 365 package which provides productivity software through the web browser. 'PaaS' model provides the consumer a computing platform using the cloud infrastructure to allow running and building their own applications. The consumer does not need to manage the "underlying cloud infrastructure including networks, servers, operating systems or storage [12]". 'IaaS' model provides the capability to control the processing, storage and networks to an extent. The consumers will have to install the OS images and related application software on the cloud infrastructure [9].

A disadvantage stated by Grossman [4] is that since cloud services are often remote, it can suffer from latency and bandwidth realted issues. Data centers can be physically located anywhere in the world, so the number of router hops from the client to the server may attribute to the latency. As well as distance, delays are also introduced by network hardware and error correction on data packets.

2.3 Cloud Gaming

Cloud gaming is new technology that can be seen as an alternative by having the games run remotely on a server and then streamed to the user. Performing computations remotely as with streaming games remotely is believed to gain traction in the future in the same way how streaming videos and audio have become ubiquitous through services such as Netflix and Spotify. NVIDIA's GRID Cloud Gaming advancements have shown that this is becoming the case. As stated by Mariano in *Is cloud gaming the future of the gaming industry* [11], cloud gaming is increasingly becoming an attractive option for consumers as higher end games can then work on simpler, cheaper clients as well as with devices that they may already own also known as thin clients.

These thin clients are responsible for displaying the game frames rendered on the cloud server side in the form of video frames. Also, it has to collect and process the game control inputs from the user and send these to cloud to be registered as inputs on the game engine. According to Shea et al [15], cloud gaming would be of great benefit to the game industry as it would open the user base to the thin clients. For example the recommended specifications to run the 2015 Game of the Year title Witcher 3[3] would require a system that has [13]:

- CPU: Intel Core i7 3770 3.4 GHz / AMD FX-8350 4 GHz
- GPU: GeForce GTX 770 / AMD Radeon R9 290
- RAM: 8GB

A system that has these components would cost around £400 and this does not include peripherals such as keyboard, mouse and monitor. The latest tablets and laptops can barely if at all meet the recommended specifications to run the game natively. Furthermore, games will have to deal with running on different hardware architectures and operating systems. Cloud gaming would enable consumers with lower end devices to run and play graphically intensive games.

In the paper *Cloud Gaming: A Green Solution to Massive Multiplayer Online Games* [1] it mentions that NVIDIA has introduced SHIELD which is a mobile gaming device that can be connected to a desktop PC with a compatible NVIDIA GPU and stream gameplay to the device via 802.11n WiFi. Another feature is the ability to connect to one of NVIDIA's data centres to play games from their selection of stream-ready games. One of the benefits of this service is the convenience of not having to wait for the download and installation of the game as you simply pick a game and instantly start playing. The service also boasts gameplay performance of up to 1080p at 60fps which are deemed by gamers to be the target performance.

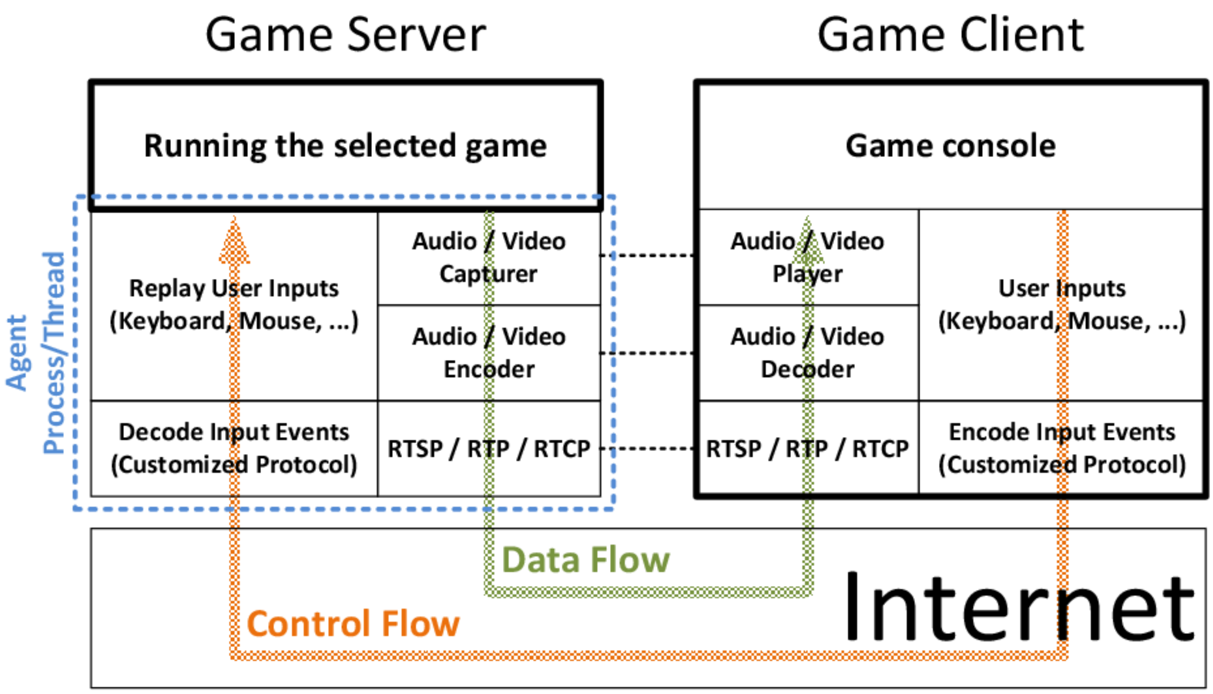

Figure 2.1: Cloud gaming system architecture

A simple cloud gaming architecture consists of several procedures which adds on top of the game engine process as shown in Figure 2.1. The client would send user inputs through some form of controller like a keyboard, mouse, or gamepead to the thin client. The game client then encodes these commands so it can easily be sent to the game server via network. When the server receives the user inputs, it simulates them so the game that is running recognises them. The game renders the frames produced as a result of the inputs and are captured and encoded to video frames. This is done so it can be easily streamed to the client through the use of Real Time Streaming Protocol (RTSP) which is a network control protocol that manages delivery of data with real time properties [14]. The thin client uses RTSP to receive the video frames and decodes them to be displayed on a video player. The user then sees the results of the button presses sent from the game running on the server side.

Due to the many different processes involved in the architecture of a cloud gaming system, managing latency has become a problem. A traditional gaming system already experiences latency and as shown in Figure 2.2, this arises from the game pipeline and display lag. The game pipiline latency is the amount of time it takes for the game to comput and render a frame and the display lag is the time it takes to display the frames on the screen. Display lag can be caused by display's scaler since current display's have a fixed resolution and expensive image processing such as dynamic contrast and motion interpolation [16]. Cloud gaming introduces latency from capturing/encoding game frame to video frame, network and decoding on the client side.

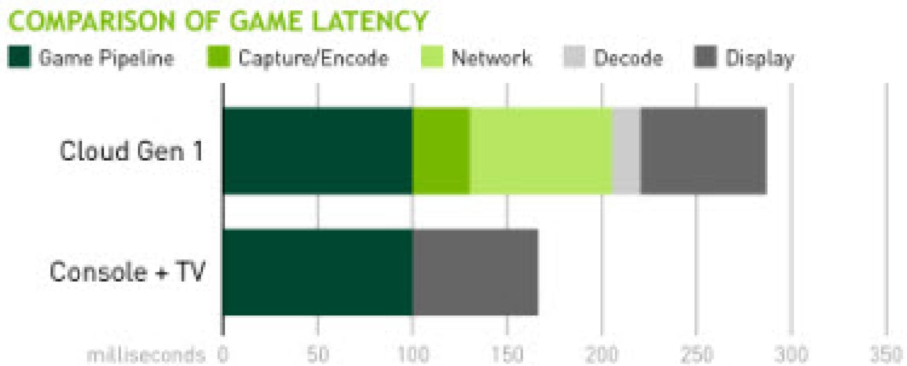

Figure 2.2: Latency in cloud gaming

2.4 Latency Mitigation

One method of latency mitigation in the network is to simply move the server closer to the clients. This means that traffic will have to travel less distance therefore latency will decrease. Unfortunately, this solution is not feasible since building and maintaining data centres are expensive. Video streaming services such as Netflix and YouTube use buffering which loads video data before playing the video in order for continuous playback. This cannot be done for live cloud gaming so other solutions to improve experiences of latency sensitive games must be explored.

2.4.1 Software-Defined Networking

As stated by Kirkpatrick [8], software-defined networking (SDN) is a new networking architecture that allows programmers to quickly reconfigure and define network usage. Whilst signifcant advances have been in other areas of technology, networking has not been able to evolve in the same pace.

Similar to mobile phones shifting to the world of smartphones with the help of APIs (application program interface), in an SDN environment, applications can communicate with network switches through an API. The API can be used to quickly reconfigure the resources of the network to accommodate the needs of the applications being executed. This main benefit of using SDN is also discussed in *Improving network management with software defined networking* [7]. Kim et al mentions that network operators will not need to configure all the network devices indivdually to make network behaviour changes, but instead make network-wide traffic forwarding decisions. The SDN controller is used for this and would have global knowledge of the state of the network.

SDN consists of two planes, the data and control plane. The data plane also known as the forwarding plane is the part of the network that carries user traffic by forwarding them to the next hop along the path to its destination in accordance to the logic in the control plane. The control plane has command where the traffic is sent by creating the routing tables and is also repsonsible for managing connections between switches, handling errors and exceptions.

2.4.2 Speculative Execution: Outatime

Another form of latency mitigation that is being explored is Microsoft's Outatime which uses *speculation to enable low-latency continuous interaction for mobile cloud gaming* [10].

Outatime basically predicts multiple possible frame outputs that may appear in the future of the game's render scene on the client side. It has to predict what frames may be needed at least a full end host round trip time (RTT) ahead of time the client actually produces game input controls.

Outatime was extensively tested

2.5 Related Works

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Chapter 3

Design

3.1 System Model

Chapter 4

Implementation

4.1 Implementation

Chapter 5

Evaluation

5.1 Evaluation

Chapter 6

Conclusion

6.1 Conclusion

References

[1] S.-P. Chuah, C. Yuen, and N.-M. Cheung. Cloud gaming: a green solution to massive multiplayer online games. *Wireless Communications, IEEE*, 21(4):78–87, 2014.

[2] P. Deutsch. The eight fallacies of distributed computing. *URL: http://today. java. net/jag/Fallacies. html*, 1994.

[3] GameCentral for Metro.co.uk. The witcher 3 wins game of the year at golden joysticks 2015, 2015. [Online]. [Accessed 15 April 2016]. Available from: http://metro.co.uk/2015/10/30/the-witcher-3-wins-ultimate-game-of-the-year-at-golden-joysticks-2015-5471812/.

[4] R. L. Grossman. The case for cloud computing. *IT professional*, 11(2):23–27, 2009.

[5] Y. Jadeja and K. Modi. Cloud computing-concepts, architecture and challenges. In *Computing, Electronics and Electrical Technologies (ICCEET), 2012 International Conference on*, pages 877–880. IEEE, 2012.

[6] M. Jarschel, D. Schlosser, S. Scheuring, and T. Hoßfeld. An evaluation of qoe in cloud gaming based on subjective tests. In *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2011 Fifth International Conference on*, pages 330–335. IEEE, 2011.

[7] H. Kim and N. Feamster. Improving network management with software defined networking. *Communications Magazine, IEEE*, 51(2):114–119, 2013.

[8] K. Kirkpatrick. Software-defined networking. *Communications of the ACM*, 56(9):16–19, 2013.

[9] A. Kumawat. Cloud service models (iaas, saas, paas) + how microsoft office 365, azure fit in, 2013. [Online]. [Accessed on 22 April 2016]. Available from: http://www.cmswire.com/cms/information-management/cloud-service-models-iaas-saas-paas-how-microsoft-office-365-azure-fit-in-021672.php.

[10] K. Lee, D. Chu, E. Cuervo, J. Kopf, A. Wolman, Y. Degtyarev, S. Grizan, and J. Flinn. Outatime: Using speculation to enable low-latency continuous interaction for mobile cloud gaming. *GetMobile: Mobile Computing and Communications*, 19(3):14–17, 2015.

[11] B. Mariano and S. G. Koo. Is cloud gaming the future of the gaming industry? In *Ubiquitous and Future Networks (ICUFN), 2015 Seventh International Conference on*, pages 969–972. IEEE, 2015.

[12] P. Mell and T. Grance. The nist definition of cloud computing. 2011.

[13] D. Piner. Witcher III system minimum and recommended system requirements and upgrade guide, 2016. [Online]. [Accessed 15 April 2016]. Available from: http://www.tentonhammer.com/guides/witcher-iii-system-minimum-and-recommended-system-requirements-and-upgrade-guide.

[14] H. Schulzrinne, A. Rao, and R. Lanphier. Real time streaming protocol, 1998. [Online]. [Accessed 20 April 2016]. Available from: https://www.ietf.org/rfc/rfc2326.txt.

[15] R. Shea, J. Liu, E. Ngai, and Y. Cui. Cloud gaming: architecture and performance. *Network, IEEE*, 27(4):16–21, 2013.

[16] A. Soomoro. What is input lag: The breakdown, 2013. [Online]. [Accessed 21 April 2016]. Available from: http://www.displaylag.com/what-is-input-lag-the-breakdown/.

Appendices

Appendix A

External Material

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Appendix B

Ethical Issues Addressed