Florida International University
School of Computing and Information Sciences

Software Engineering/Research Focus

Final Documentation

Project Title:

Next Generation Networking for Virtual Reality and Game Player Collaborative and Multi-Player

Team Members: Gabriel Taveras & Brad Marzouka

Product Owner(s): Dr. Francisco Ortega

Mentor(s): Dr. Francisco Ortega

Instructor: Masoud Sadjadi

Abstract

The video game industry has seen a tremendous maximization in recent years due to its popularity and ambitious number of specifications: graphics, story, action and violent gameplay, etc. Of the most common and popular of the gaming categories is the ability for players to access online multiplayer platform, with a shot to compete and show off your skills against other players, on a global scale, in a collaborative environment. As the number of players begin to increase, so does the demand for the instant availability of the gaming content: reduced latency and lag time, minimal waiting queues, optimal steaming, and rendition and smoothness of in-game cut scenes. The demand for such a high-performance operation in a massive multiplayer environment requires extreme insistence in hardware and networking components within a centralized data center for exceptional gaming throughput; gaming processing will require a multitude of services and commitment to manage the flow of data, to provide the best quality of service (QoS) to the end users for the best online gaming experience. In this paper, we'll discuss about the possible implementation of SDN within a multiplayer spectrum to provide a better online experience for players, the cyber security aspect of the possibilities of advanced persistent threats (APTs) it may face, and possible implementation in future projects, to coincide with the future generations of technological advancements within the gaming industry (i.e. Virtual reality gaming).

Table of Contents

INTRODUCTION 5
CURRENT SYSTEM 6
PURPOSE OF NEW SYSTEM 6

USER STORIES
IMPLEMENTED USER STORIES 7
PENDING USER STORIES 8

PROJECT PLAN
HARDWARE AND SOFTWARE RESOURCES 9-10
SPRINTS PLAN 11
Sprint 1 11
Sprint 2 12
Sprint 3 12
Sprint 4 13
Sprint 5 13
Sprint 6 14
Sprint 7 14

SYSTEM DESIGN
ARCHITECTURAL PATTERNS 15
SYSTEM AND SUBSYSTEM DECOMPOSITION 19
DEPLOYMENT DIAGRAM 22
DESIGN PATTERNS 22

SYSTEM VALIDATION 25

GLOSSARY 26

APPENDIX 27
APPENDIX A - UML DIAGRAMS 27
Static UML Diagrams 28
Dynamic UML Diagrams .....
APPENDIX B - USER INTERFACE DESIGN 29
APPENDIX C - SPRINT REVIEW REPORTS 31
APPENDIX D - USER MANUALS, INSTALLATION/MAINTENANCE DOCUMENT, SHORTCOMINGS/WISHLIST DOCUMENT AND OTHER DOCUMENTS
..... 35

REFERENCES 39

INTRODUCTION

Video games has seen an influx in popularity in recent years, not only among young children, but amongst adults as well; we see a diverse population in the different generations of human beings jumping in to taste the reluctance of an excellent story mode, or just for sheer excitement to compete with other players from around the globe. Concurrently, with more players jumping into the multiplayer world of video gaming, the demand for instant gratification of video game content becomes the decisive action of whether or not players will want to continue progressing within the game. Nowadays, a video game's revenue predominantly comes from the online multiplayer content, and is usually the deciding factor of whether or the not the game will continue forward for future releases, additional patching, and other opportunities to release additional features, adding more of an experience to the game (e.g. sequels, unlockable items, or DLCs).

The multiplayer platform is especially remarkable due to the workload the game servers must maintain in order to establish a quality gaming service for the users on a daily basis over extended periods of time. Multiplayer gaming is stretched through a variety of margins to give an adaptable experience for gamers; for instance, depending on the game genre, the workload shall vary, as it requires achieving different functionalities to maintain availability of online for the gamers (e.g. story mode, playthroughs, player collaboration). By using Software Defined Networking (SDN) we have the ability to enable better management and control of communication, data flow, and increase packet distribution within a data center for an optimal experience in quality gaming.

Thus, we recommend the implementation of Software-Defined Networking (SDN) to be a foundation layer in next generation gaming to assist in the increase performance of a multiplayer environment. SDN provides an opportunistic methodology to reduce workloads of gaming servers, discover clear and concise network pathways to meet end-user needs and attempting to reduce the failover rate when governing multiple online players on a massive scale.

Current System

Current systems readily available to end users and enterprises consist of a network topology of a main switch that communicates requests to other network switches on the network. These switches then push sent requests to a Top of Rack switch which sends data to virtual machines where the data is used and manipulated by end users. In a scenario where the network in question needs to have very low latency, like in the case of multiplayer gaming or a high bandwidth application, this network cannot sufficiently process data adequately enough to handle the requirements of the end users.

Purpose of New System

Our new SDN network will have the capability of supporting these new low latency, high bandwidth requirements that end users need for their multi-user applications. An SDN network, by design, is equipped to handle the data requirements of current and future bandwidth applications due to the network's ability to direct and automate traffic. This guarantees a better quality of service (QoS) for data transmissions. Also, with the deployment of virtualization in many organizations today, SDN networks require the use of a different style of network that doesn't have the complexities of current network systems. With an SDN network along with the OpenFlow protocol, a network can be created to be better suited to each individual virtual machine by allowing the creation of data paths across the network to allow for faster processing and data transfer across the network.

USER STORIES

The following section provides the detailed user stories that were implemented in this iteration of the Next Generation Networking project. These user stories served as the basis for the implementation of the project's features. This section also shows the user stories that are to be considered for future development.

Implemented User Stories

Actors for Online Multiplayer Environment	Possible Outcome(s)
1. As a gamer	I want to play multiplayer games that have the lowest latency possible. Meaning, exceptional gameplay, higher bandwidth, no lag, no shutdowns, kick-outs, and little queue.
2. As a system administrator	I want to deploy a network that is easy to maintain, applicable API and accessible controls and other interfaces.
3. As a gamer	I want to play games with thorough online content and no multiplayer interruptions.
4. As a system administrator	I want to deploy a network that can sustain operations, while handling the workload of millions of players within a global spectrum.

5. As a system administrator	I want to deploy a network that is capable of supporting VR/AR gaming and other innovative future technologies to come.
6. As a system administrator	I want to see the centralized server environment distribute the number of data/packets with a smooth TCP/IP underlay, reducing RTT, and increasing QoS for customers.
7. As a CEO	I want to provide satisfaction and exceed customer expectation .

Pending User Stories

Actors for Future Stories	Possible Outcome(s)
1. As a gamer	I want to be able to play multiplayer VR or AR games with no systematic issues.

PROJECT PLAN

This section describes the planning that went into the realization of this project. This project incorporated the agile development techniques and as such required the sprints to be planned. These sprint planning's are detailed in the section. This section also describes the components, both software and hardware, chosen for this project.

Hardware and Software Resources

The following is a list of all hardware and software resources that were used in this project:

HARDWARE RESOURCES
• Windows 7 • Kali Linux • Ubuntu • Virtual Machines

SOFTWARE RESOURCES

- Oracle VirtualBox
- Mininet
- Openflow
- Mingle
- GitHub
- Various Google Products

Sprints Plan

Sprint 1

Description:

- Researched basic formalities relating to the cyber-security infrastructure within the AR/VR platform environments. Research is only done as initial introduction. My goal is to learn what the basics of AR/VR and cyber security involved with it.

Acceptance Criteria:

1. Initial research of AR/VR environments.
2. Research principles of Cyber-security within VR, and see how its applied.
3. Understand vulnerabilities within AR/VR and discover solutions that have been generated from security professionals.

Sprint 2

Description:

- Per request from Dr. Ortega, the initial goal for this sprint is to understand, and research methods and other procedures on why consumers would need to be aware of software vulnerabilities within the VR/AR platform. Also, with multiplayer gaming on a constant rise, conducting research on how to reduce latency in order to improve multiplayer gaming with VR headsets, and overall multiplayer gaming shall produce a phenomenal hit within the industry for gaming corporations to apply these effective protocols. Decrease in latency = better performance, constant streaming and reliable IT support.

Acceptance Criteria:

1. Research how to implement cyber-security within AR/VR platform, and why it is important for the consumer and industry; as well as, how multiplayer will correspond with them.
2. Discover how to reduce latency, and other network bandwidth obstacles to improve multiplayer performance.
3. Discover how other gaming corporations attempt to reduce lag and increase performance within a multiplayer environment.

Sprint 3

Description:

- Consensus of this sprint involves to conduct further research of SDN Networks and how it is implemented within the Multiplayer environment. SDN networks has already begun to become operational for companies such as Blizzard. We must further research this topic to see how to expand with Virtual reality and gaming.

Acceptance Criteria:

1. Differentiate between using SDN and Cloud platforms to provide a stable gaming experience.
2. Review performance delay functions to allow more effective gaming.
3. Provide more research papers on SDN topics to provide a basic foundation on where to start for the implementation of SDN and to facilitate a collaboration with the AR/VR environments.

Related Tasks:

Tasks and Purpose involves mainly researching between Cloud Computing and SDN networking, and how to apply gaming and multiplayer framework.

Sprint 4

Description:

- SDN is now an up and coming technological tool that allows network administrators to program, initialize, control, change, and manage network behavior dynamically through open interfaces and low-level functionality. These appliances benefit the reduction of actual hardware in which requires additional hardware maintenance, and space. SDN will be additionally researched, as well as Cyber-security practices used by professionals to assess future threats that virtual realities will face.

Acceptance Criteria:

1. Research professional opinions and solutions on how to reduce VR vulnerabilities, and be aware of any future malicious factors.
2. Further research of SDN within a complex multiplayer environment to reduce latency, improve performance, and ease of control for administrators.
3. Understand the concept of network trafficking and network security
4. Understand how SDN can be applied to VR/AR headset and headset gaming.

Related Tasks:

Tasks and purpose include further research and finding research papers for Dr. Ortega.

Sprint 5

Description:

- As per request from Dr. Ortega, AR/VR is a very ambiguous topic; as result, we will be primarily focusing on the applying SDN onto gaming platforms. Cyber-Security can be briefly mentioned, however it is not the primary focus.

Acceptance Criteria:

1. Conduct further research on SDN
2. Determine which gaming industry corporations actually apply SDN for multiplayer.

Sprint 6

Description:

- Combine work with research partners to begin working on research paper. Finalize remaining research in areas needed.

Acceptance Criteria:

1. Begin working on research paper to turn in at the end of the semester.
2. Finish remaining research in latency and cyber security requirements of the project.

Sprint 7

Description:

- Combine work with research partners and finish research paper and final deliverables.

Acceptance Criteria:

1. Finalize work on research paper to turn in at the end of the semester.
2. Finish final deliverables such as posters and PowerPoints to submit for grading.

SYSTEM DESIGN

This section contains information on the design decisions that went into this project. The architecture patterns are outlined and explained. The entire system is shown in a package diagram and the subsystems are explained. To conclude, the design patterns used in the project are discussed.

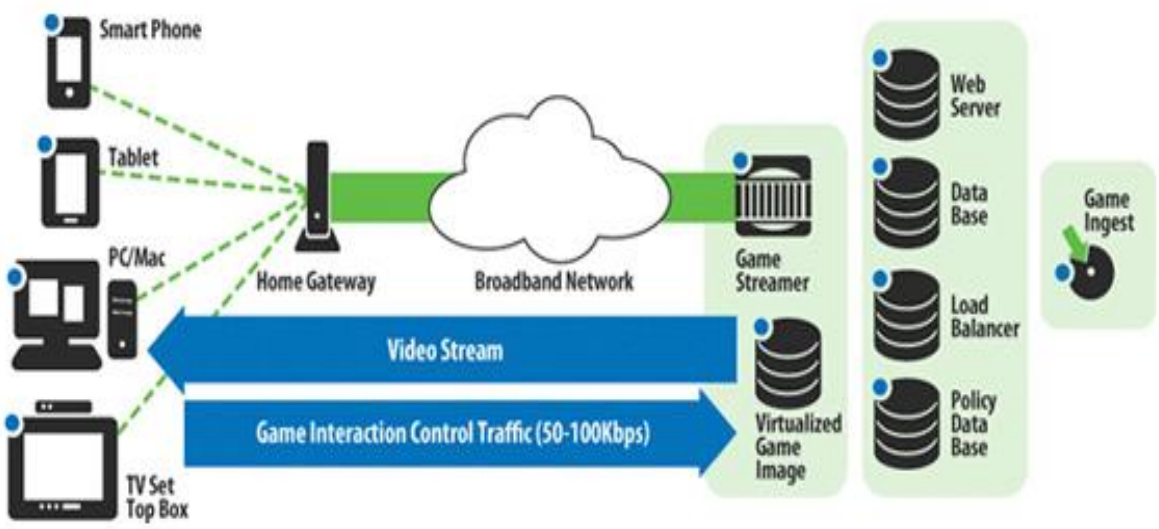
Architectural Patterns

For usage of the SDN, we can implement hypernet; which is an integrated bundle that covers all the knowledge needed to create a SDN network and deploy the software needed to run on the SDN. It can be tailored for a specific game would contain a program to devise the optimal network topology from:

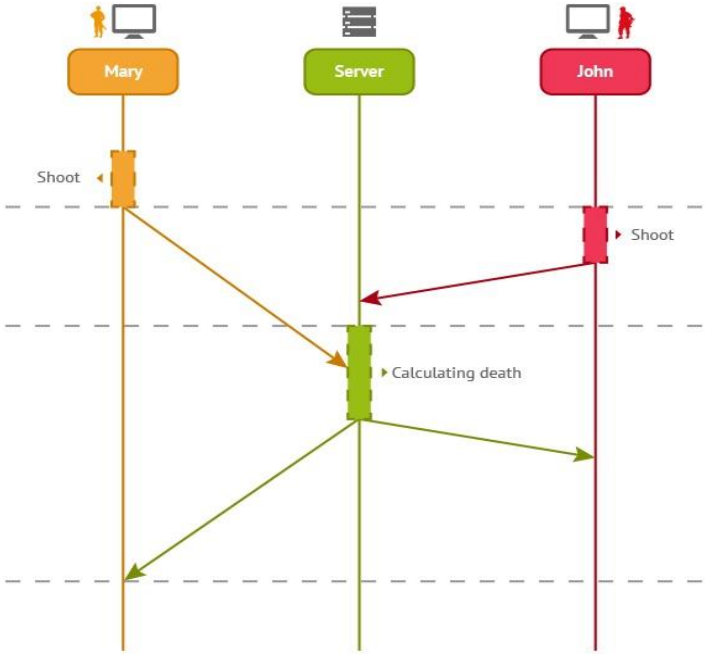
- given set of current players
- the software protocol stacks utilized on the end system
- router methods (priority queues, multicast, routing tables)

Using the hypernet can ease the tension and workload of actual NIDs, hardware servers, and other technologies used to establish a thorough network. We must take into consideration of TCP/IP traffic and RTT time to transport and distribute the data/packets to clients on a global scale.

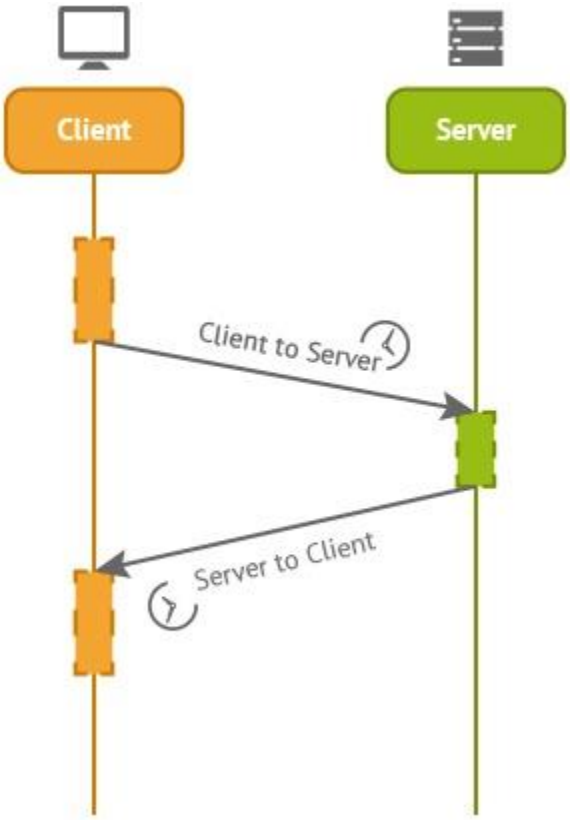
Traditional network architectures are mainly implemented through appliances (network switches, routers, and hubs). The “bottleneck” effect is when data and request from multiple devices will cause a buildup of traffic and clog the network ports, and cause the hardware components to stall based on the extreme workload.



Once the data is received, depending on the queue or wait time, the packets will eventually become processed and sent to its respective destination, while the information, or confirmation of communicated of the data link, will ping back to the original host…all within fractions of second with each other (minutes if the bottleneck is severe).



We focus on how RTT and Lag can contribute to the lackluster of effective gaming. The amount of time taken by a packet of information that is sent from a client to get to the server and return to that client, with considering the time it takes to process. If Mary has Lag and fires preemptively, she can still be executed because of the lack of network compensation; as John shoots at a later time (milliseconds later) and effectively kills Mary.



RTT = Time to send + Time to Receive

Similar to that of a TCP Handshake with SYN/ACK recognition.

System and Subsystem Decomposition

The operational framework of the SDN architecture is to manipulate frame and packet flows through the network at a large scale in a "programmable" fashion. The initial design of the SDN infrastructure us to converge the mesh of storage and network traffic onto lower latency pipes, for a smoother transition user/client interaction. The SDN model encompasses"

- Control Plane – SDN controller, allows dynamic access and administrator; able to change network switch rules, prioritize/de-prioritize, or block specific packets.
- Data Plane – Hardware switching devices; can be new or existing hardware with specialized firmware.
- Centralized Controller – centralized management for consistent policies involving virtual and physical resources.
- VEB – while implementing with virtual technology, VEB serves as the hypervisor that can be controlled by an SDN controller.
- OpenFlow – communication interface between the control and forwarding layers of SDN architecture.

Deployment Diagram

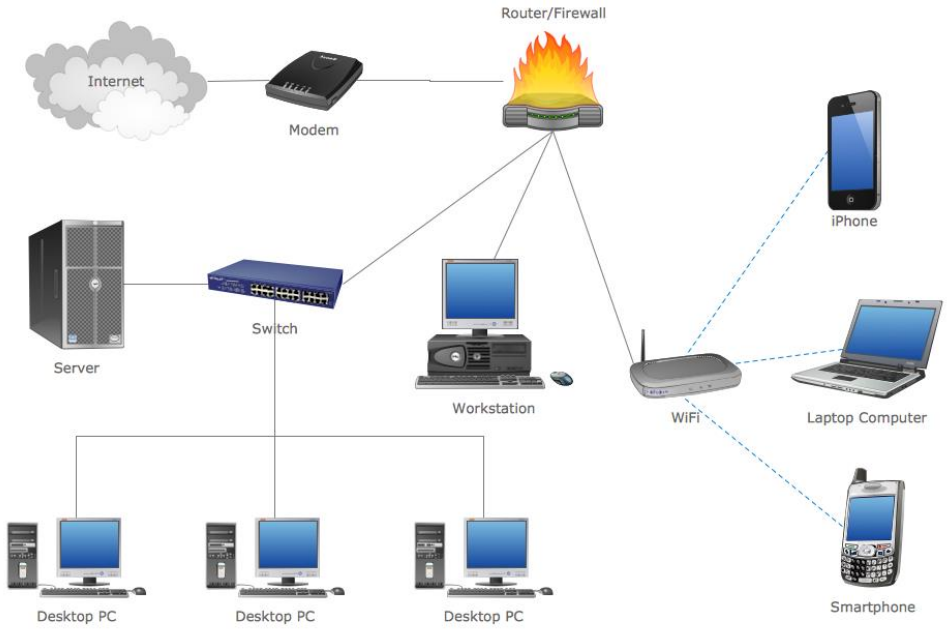


Figure 1: Basic representation of a network diagram with multiple users and devices.

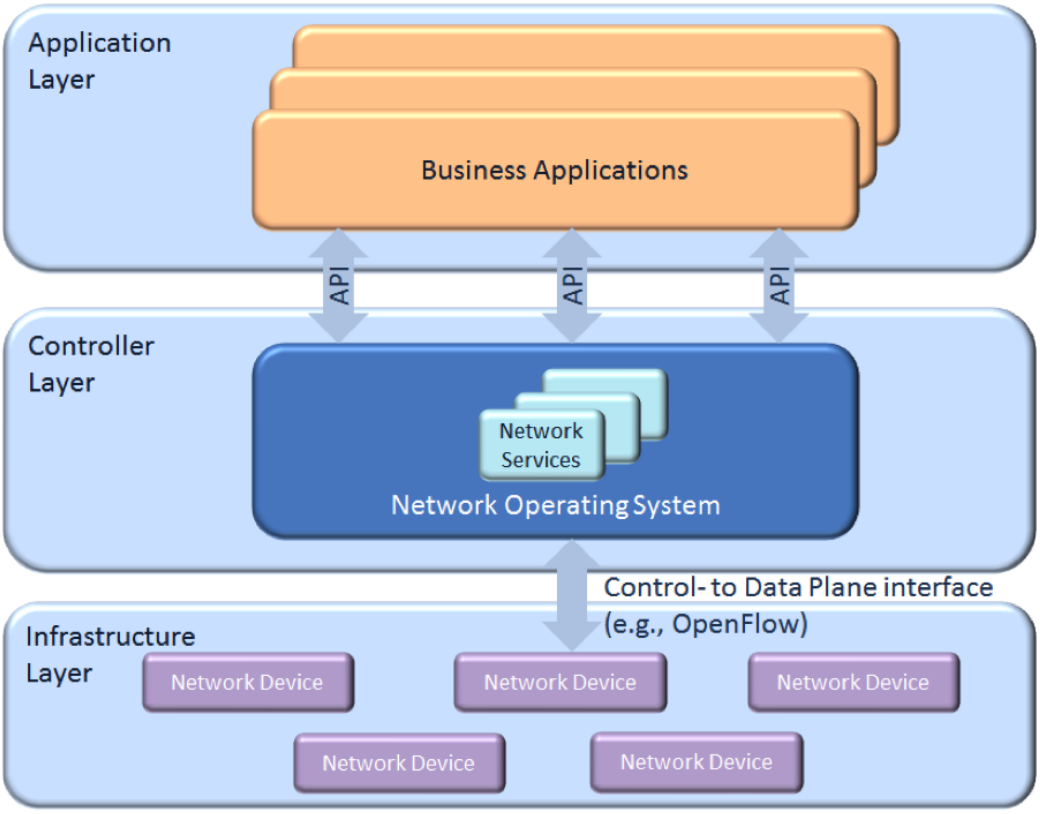


Figure 2: Standard SDN architecture

Design Patterns

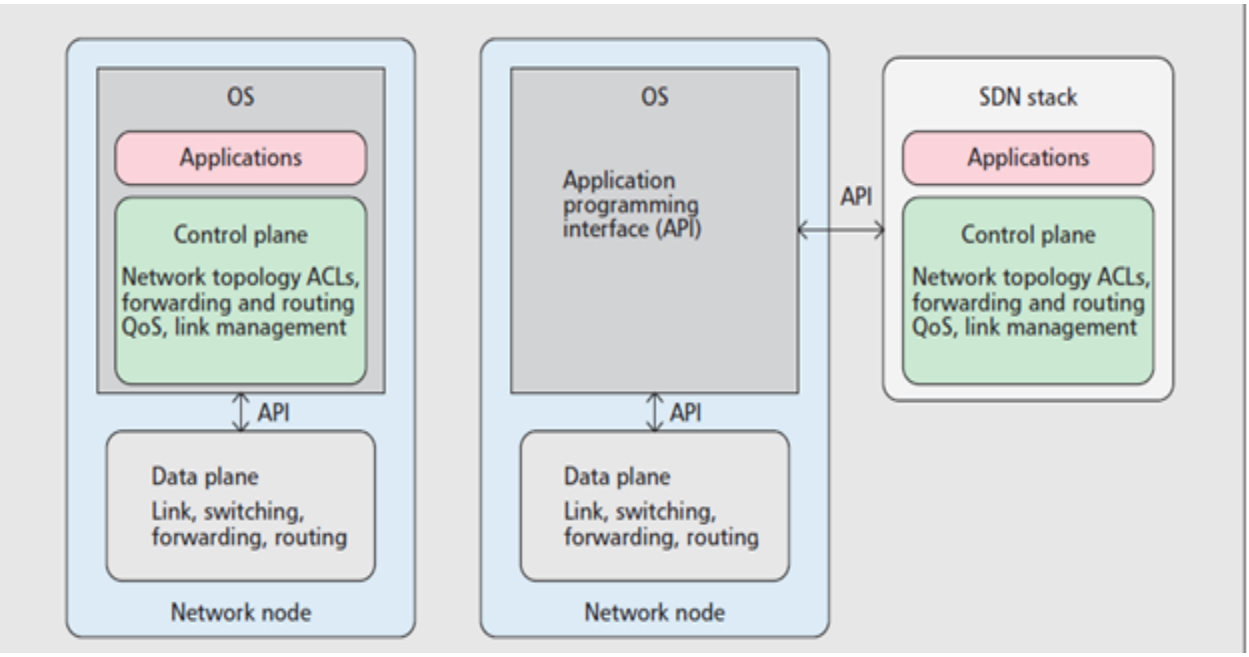


Figure 3: Overall representation and clearer illustration of SDN planes and API stacks

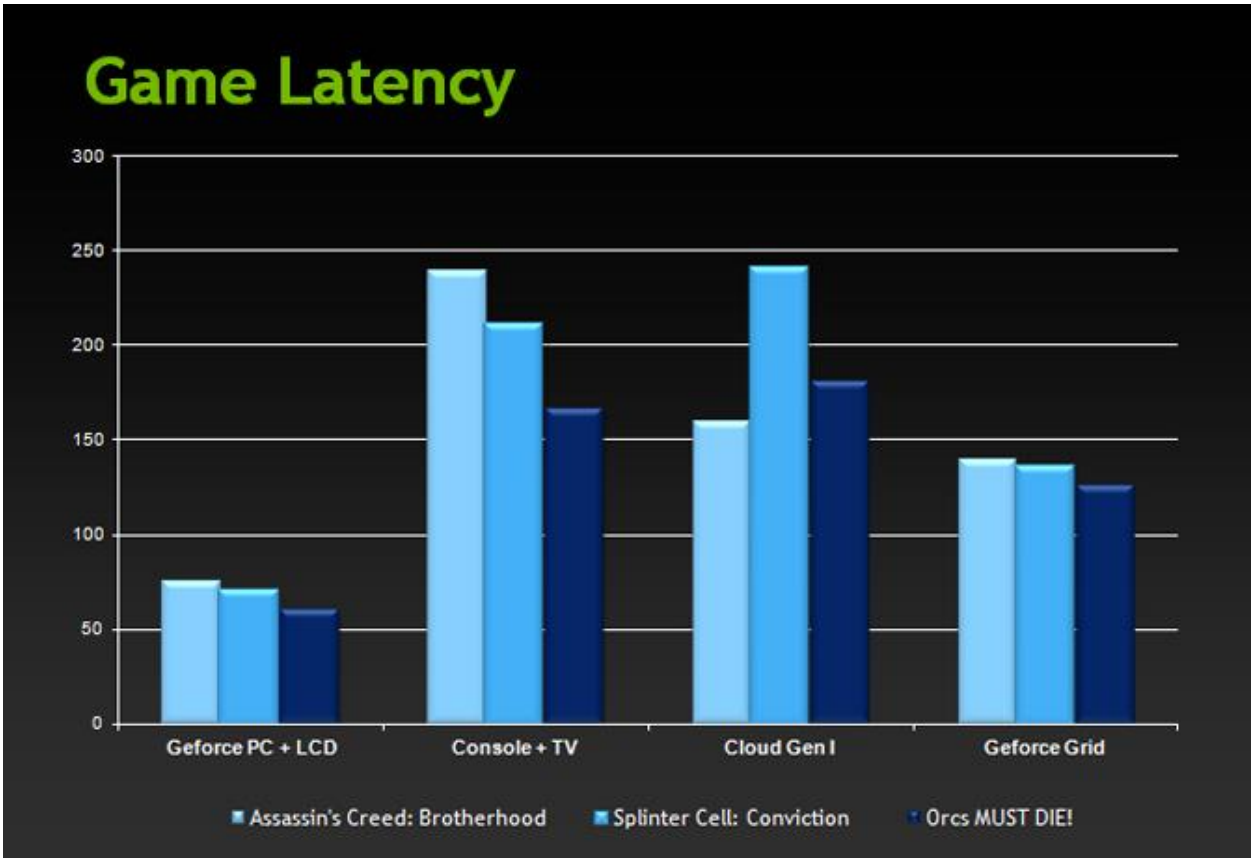


Figure 5: The effects of incorporating SDN proves that the networking capabilities can significantly have an impact on higher online gaming throughput.

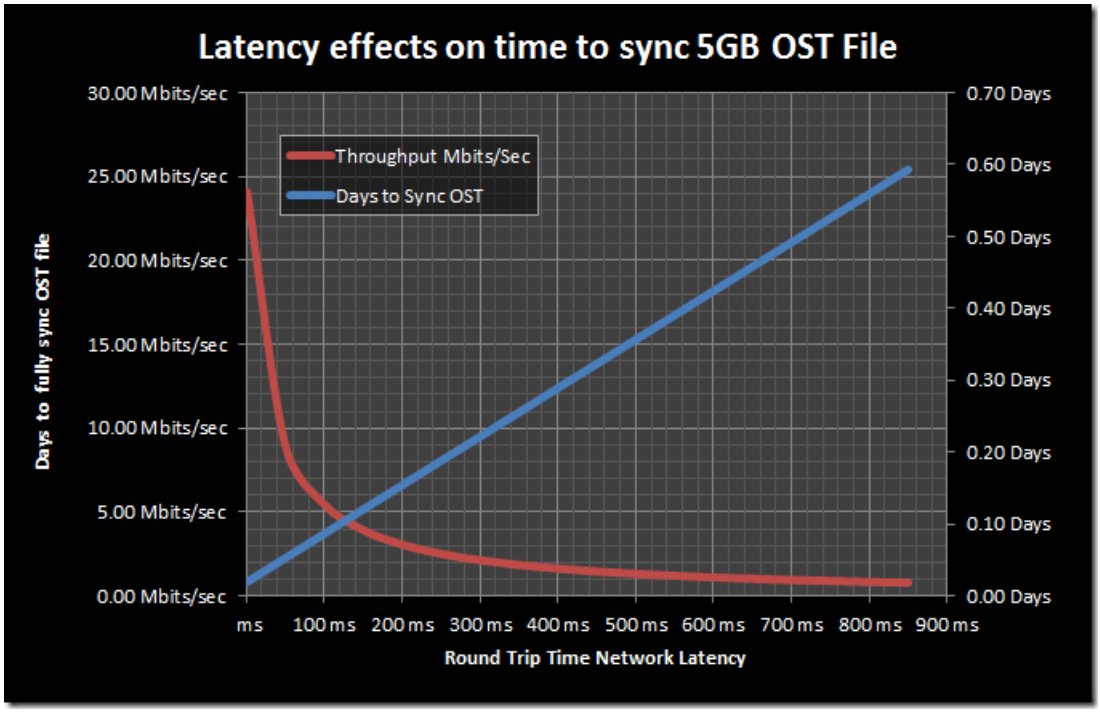


Figure 6: The RTT latency effects on SDN compared to a traditional network.

SYSTEM VALIDATION

No Testing or Code Implementation for SDN Research
Purpose: To conduct and further research on implementing SDN and other networking technologies to better enhance the outcome of multiplayer collaboration and online content gaming.
Preconditions: Setting up a Centralized Network Hub for SDN parameters
Input: Configure Network OS APIs for control of data flow
Expected Output: Network scalability Programmable network interface

GLOSSARY

SDN – Software-Defined Network – The ability for administrators to monitor and manage network behavior dynamically over open interfaces.

NID – Network Interface Device – Network device used for telecommunication purposes.

VEB – Virtual Ethernet Bridge – Integrating virtual network using a hypervisor to control SDN through a virtual machine standpoint.

RTT – Round Trip Delay Time – Length of time it takes a signal to be sent, plus time it takes to come back.

TCP/IP – Transmission Control Protocol/Internet Protocol – Helps establish and maintain a network conversation/

SYN/ACK – handshake between 2 or more network devices to acknowledge communication between the nodes.

APPENDIX

Appendix A - UML Diagrams

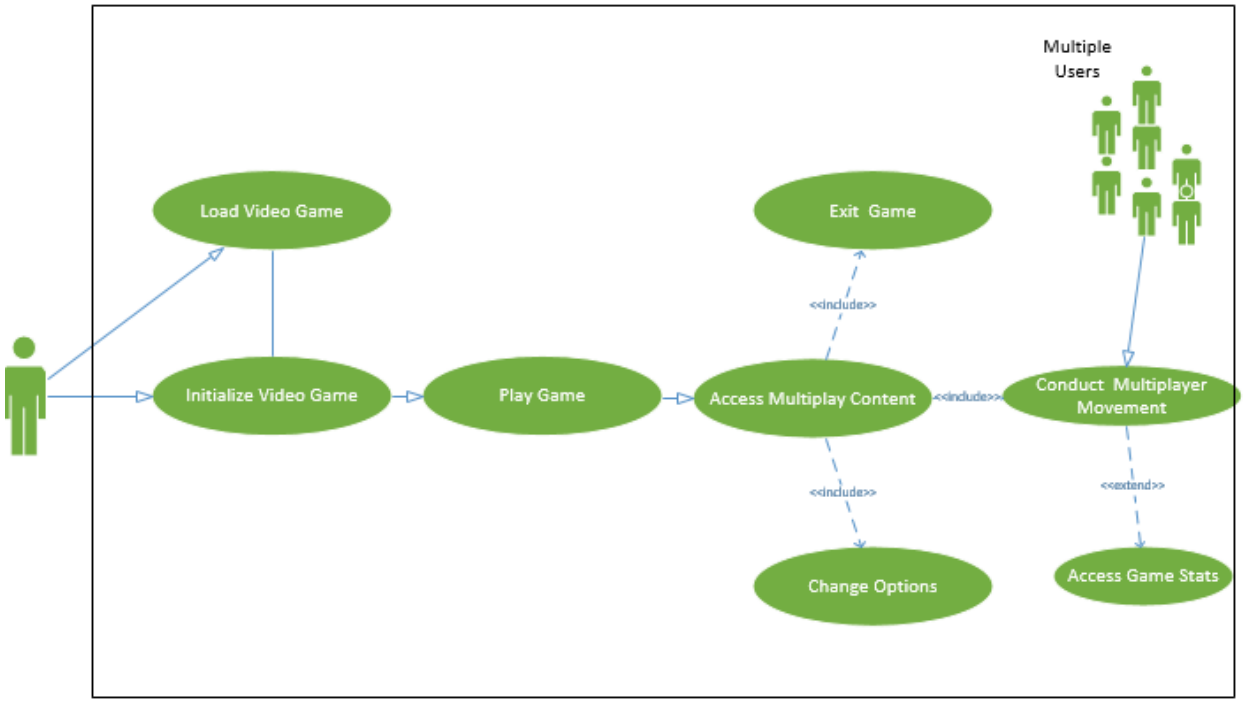


Figure 7: UML diagram displaying collaboration of player initializing contact with respective console, and online gaming collaboration.

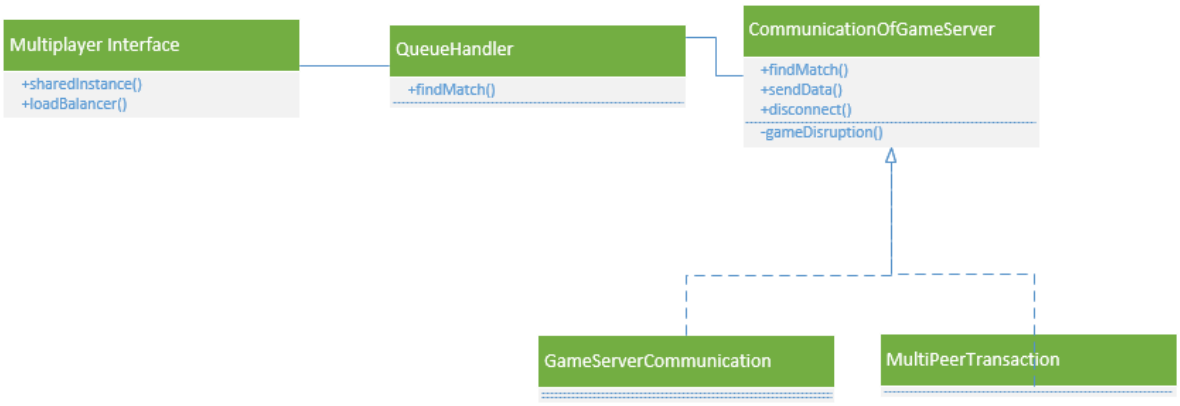


Figure 8: UML diagram showing multiplayer online interface

Appendix B - User Interface Design

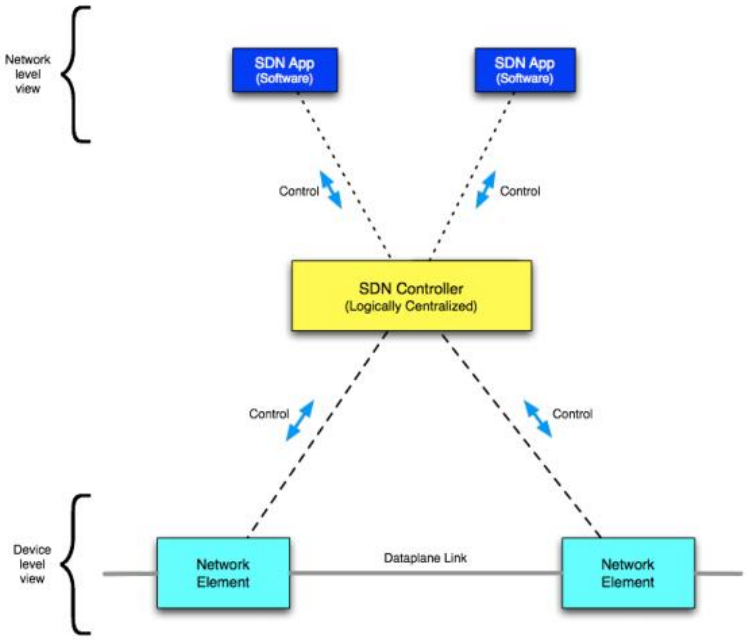


Figure 9: SDN framework and architecture. Depicts the principles of programmatic and abstracted interaction with the network. Shows direct control of forwarding behavior, defining the logical topology.

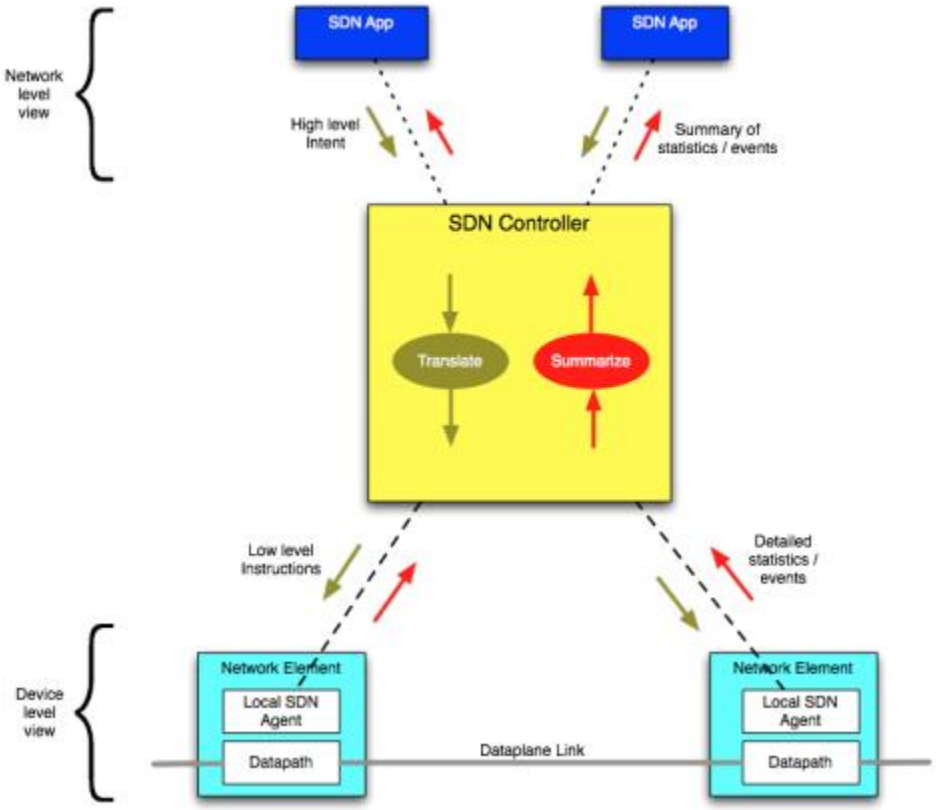


Figure 10: Depicts key aspects of logically centralized control: translating the high level intent into lower level instructions, and summarizing low level events and statistics to form higher level information

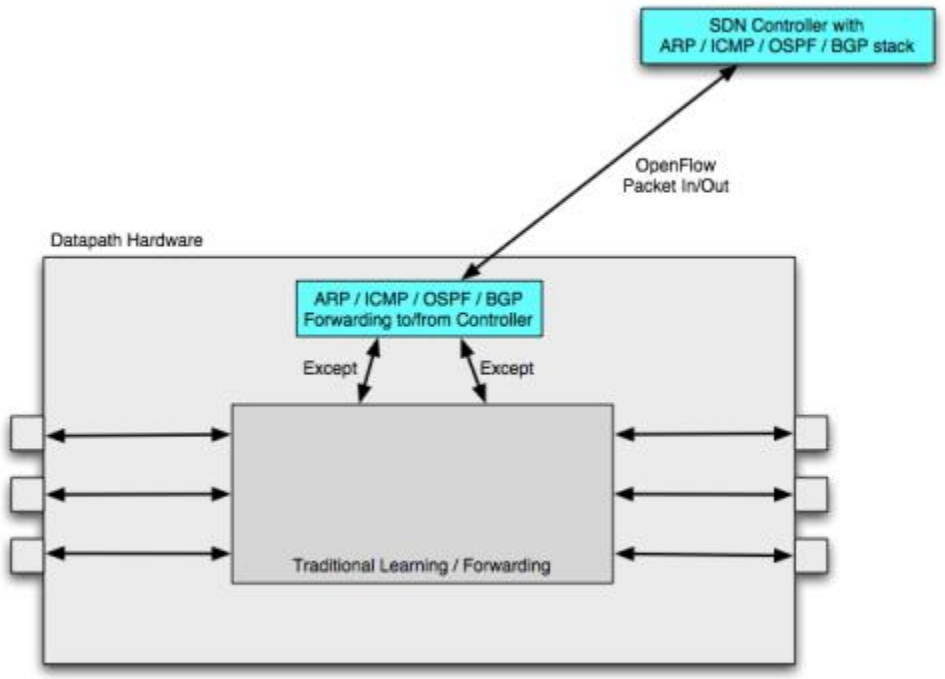


Figure 11: Traffic needs to be encapsulated to transit the traditional network, leading to the concept of overlay on and off ramps (performing tunnel origination/termination, i.e. encapsulation/decapsulation),

Appendix C - Sprint Review Reports

Meeting Times

Meeting 1

Attendees: Gabriel Taveras, Bradley Marzouka
Product Owner: Dr. Francisco Ortega
Start time: 1500 Hours

End time: 1600 Hours

Discuss the overview concept and generalization of what is required for the future implementation of SDN within the gaming platform environment. Also, how and why should we care about CyberSecurity when dealing with online gaming

- Discuss importance of CyberSecurity in networking platform
- Become aware of VR/AR technology
- Discover how to reduce latency
- What is bandwidth? And how does it differ from latency?

Our results will be discussed in future meetings and how to provide foundational first step to provide to the gaming community.

Meeting 2

Attendees: Gabriel Taveras, Bradley Marzouka
Product Owner: Dr. Francisco Ortega
Start time: 1500 Hours
End time: 1600 Hours

Conduct further research on how the Gaming Community can benefit from the usage of SDN and other technologies to provide a more fluid gaming experience. Also, VR/AR is new standard of technology, it is important to try and implement onto VR/AR platforms as well to be prepared for future generation of gaming.

- Find out more information of SDN amongst the VR/AR platform and how it can be implemented
- Further research on cybersecurity and how it can benefit online community
- Find potential threats that may increase risk within SDN platform
- Further overall efficiency on SDN networking
- How do we increase network reliability?
- How can we control networking with SDN controllers and programmable APIs?

Our results will be discussed in future meetings and how to provide foundational first step to provide to the gaming community.

Meeting 3

Attendees: Gabriel Taveras, Bradley Marzouka
Product Owner: Dr. Francisco Ortega
Start time: 1500 Hours
End time: 1600 Hours

With SDN being implemented, we must further research on the technologies surrounding its payload. Software tools, such as Mininet and OpenFlow, can help achieve this with great success within the gaming industry.

- Practice using Mininet on a Linux environment
- Research more of using OpenFlow
- What Cyber threats do we face?
- Expand on potential risks and mitigation techniques to counter malicious activities on a SDN network.

Our results will be discussed in future meetings and how to provide onto SDN platforms.

Meeting 4

Attendees: Gabriel Taveras, Bradley Marzouka
Product Owner: Dr. Francisco Ortega
Start time: 1000 Hours
End time: 1130 Hours

Semester wrap-up of how we can implement SDN onto a multiplayer platform. We must finalized research material and begin research paper that discusses the overall goal of the SDN platform in multiplayer gaming. Because of the little resources found on networking with AR/VR technologies, we will be excluding that section of research paper, and

- Find out more information of SDN amongst the VR/AR platform and how it can be implemented
- Further research on cybersecurity and how it can benefit online community
- Find potential threats that may increase risk within SDN platform
- Further overall efficiency on SDN networking

Our results will be discussed in future meetings and how to provide foundational first step to provide to the gaming community.

Appendix D - User Manuals, Installation/Maintenance Document, Shortcomings/Wishlist Document and other documents

USER MANUAL

SDN Architecture needs to the following to be addressed properly:

1. Simplicity and Expressiveness – need to be introduced to more domain models
2. Applicability – Precise and clear when discussing scope of compatibility for existing networks. Also, understand Best Practice!
    a. Needs to be supported by:
        i. wide-area networks (WAN)
        ii. Transport services
        iii. Data center networks
        iv. Intra-site chaining
        v. Residential IP services

b. Evolution over Time – maintain availability and become susceptible to react to different scenarios. Also, identifying problems, requirements, and prioritized work

3. Interworking – scalable, supports hybrid usage, and remain compatible
4. Interoperability
5. Scalability – scalability requirements on the following:
    a. Controller Plane – distributed system spanning across multiple physical platforms
    b. Network Element – applying heterogeneous processors with internal load balancing, cascading, and clustering to achieve performance (increase throughput, lower the latency)
    c. Individual functions – provisions to wide range capabilities and permit negotiations, especially with different regions of the globe.
6. Security – Implement network security and other parameters, such as:
    a. Authentication of communication entities
    b. Access control enforcement
    c. Privacy by encryption
    d. Auditing
    e. Certification of Assurance
    f. Proper mitigation techniques and countermeasure
    g. Appropriate planning and documentation
7. Resilience and Fault Tolerance – Capable of error handling, transient conditioning handling, and High Availability Support
8. Management and Monitoring - Arrangements for pairing Network Elements and SDN Controllers as well as by making provision for performance monitoring of the SDN Controller/Network Element interfaces. Further concerns include configuration, bootstrapping, and in-band control of individual Network Elements while addressing the buildout, repair and maintenance of networks

Installation

Mininet is a network emulator which creates a network of virtual hosts, switches, controllers, and links. Mininet hosts run standard Linux network software, and its switches support OpenFlow for highly flexible custom routing and Software-Defined Networking.

Mininet supports research, development, learning, prototyping, testing, debugging, and any other tasks that could benefit from having a complete experimental network on a laptop or another PC.

Mininet is ideal for implementing a mock SDN network simulator

1. Install Ubuntu Server (or any other Linux) onto a VM. NEVER INSTALL ON HOST MACHINE!
2. Update Server - download/install system updates
    a. *Sudo apt-get update*
    b. *Sudo apt-get upgrade*
    c. *Sudo apt-get distupgrade*
3. Install required software – install git
    a. *Sudo apt-get install gi*t
4. Install Mininet 2.2 from Source code –
    a. *Git clone git://github.com/mininet/mininet*
5. Identify host-only adapter on the VM
    a. *Ifconfig eth1*

Shortcomings/Wishlist

Although this research project is only on version 1.0, we succeeded to identify most of the functionality features of SDN and how implementing in a multiplayer environment can greatly increase the online gaming experience, compared to using traditional networking technologies.

In future VIP Next Generation and Multiplayer Collaboration for semesters to come, we would like to see the following possible renditions for future work:

- Future research studies committed to the implementation of SDN for Virtual/Augmented Reality gaming.
- Expand upon the usage of using OpenFlow to easily control the network flow from the forwarding plane of a network switch.
- Apply Network function virtualization (NFV) to collaborate with SDN for better throughput between the client and server interactions.
- Focus more on RTT and Speed testing to discover how to transfer data packets faster
- Practice more with Mininet
- Discover other methodologies to reduce latency and lag, and increase bandwidth for a more admirable online gaming content and multiplayer interactions (focus primarily with coherence with SDN).

REFERENCES

1.) Network Functions Virtualization (NFV) white papers and other documents: http://www.etsi.org/technologies-clusters/technologies/nfv

2.) SDN Architecture v1.0: https://www.opennetworking.org/images/stories/downloads/sdn-resources/technicalreports/TR_SDN_ARCH_1.0_06062014.pdf

3.) SDN Architecture Overview v1.1: https://www.opennetworking.org/images/stories/downloads/sdn-resources/technicalreports/TR_SDN-ARCH-Overview-1.1-11112014.02.pdf

4.) Install Mininet 2.2 [Source Code]: http://www.brianlinkletter.com/how-to-install-mininet-sdn-network-simulator/

5.) Network Revolution: Software Defined Networking and Network Virtualization: http://wikibon.org/wiki/v/Networking_Revolution:_Software_Defined_Networking_and_Network_Virtualization

6.) Concept Draw for Network Topology: http://www.conceptdraw.com/samples/network-diagram

7.) OpenNet Summit, SDN: Transforming Network to Accelerate Business Agility: http://opennetsummit.org/archives/apr13/site/why-sdn.html