A picture containing tool

Description automatically generatedVanhove Louis

The Host Migration Graveyard:

A feasibility study in co-op games

Supervisor: Van Verdegem Dave

Coach: Van der Kelen Cedric

Graduation Work 2023-2024

Digital Arts and Entertainment

Howest.be

A close up of a card

Description automatically generated

Contents

[Abstract & Key words 3](#_Toc187542113)

[Preface 4](#_Toc187542114)

[List of Figures 5](#_Toc187542115)

[Introduction 6](#_Toc187542116)

[Background 6](#_Toc187542117)

[hISTORY 6](#_Toc187542118)

[DEDICATED SERVERS 7](#_Toc187542119)

[COMMUNITY-HOSTED SERVERS 7](#_Toc187542120)

[PEER-TO-PEER 8](#_Toc187542121)

[RELAY CLIENT-Server 8](#_Toc187542122)

[Personal inspiration 9](#_Toc187542123)

[Purpose of the research 9](#_Toc187542124)

[Research questions 10](#_Toc187542125)

[Research question 1: CONTEXT FEASIBILITY 10](#_Toc187542126)

[Research question 2: MIGRATION STRATEGIES 10](#_Toc187542127)

[Hypotheses 10](#_Toc187542128)

[Research question 1: CONTEXT FEASIBILITY 10](#_Toc187542129)

[Research question 2: MIGRATION STRATEGIES 10](#_Toc187542130)

[Theoretical Framework 11](#_Toc187542131)

[Research 13](#_Toc187542132)

[1. GAME PROTOTYPE 13](#_Toc187542133)

[1.1. Subtopic 1 13](#_Toc187542134)

[1.1.1. Subtopic 13](#_Toc187542135)

[2. HOST MIGRATION EXPERIMENTATION (yap eos, DISCORD LOOKUP, EXAMPLEs, …) 14](#_Toc187542136)

[1.2. Subtopic 1 14](#_Toc187542137)

[1.2.1. Subtopic 14](#_Toc187542138)

[Discussion 16](#_Toc187542139)

[Conclusion 17](#_Toc187542140)

[Future work 18](#_Toc187542141)

[Critical Reflection 19](#_Toc187542142)

[References 20](#_Toc187542143)

[Acknowledgements 21](#_Toc187542144)

[Appendices 22](#_Toc187542145)

# Abstract & Key words

**An abstract explains the outline of the paper concisely (the methods, results, etc.). Maximum length of 250 words, preferably both in English and Dutch.**

**(ENG)**

Host migration is a networking process that allows another player to take over as the game host when the original host disconnects, ensuring no loss of progress for the other players. Despite its potential to improve multiplayer stability and eliminate dedicated server costs, it is generally considered as a dead concept due to technical complexity and production cost constraints. This research explores the feasibility of implementing effective host migration strategies online co-op games made in Unity. Various approaches were researched for performance, reliability, and cost-efficiency, offering practical solutions tailored for indie developers. Not only the feasibility of host migration, but also the scalability of such an implementation was extensively tested and analyzed.

// say something about conclusions

**(NL)**

Hostmigratie is een netwerkproces waarbij een andere speler de rol van spelhost overneemt wanneer de oorspronkelijke host de verbinding verbreekt, waardoor de andere spelers geen vooruitgang verliezen. Ondanks het potentieel om de stabiliteit van multiplayer games te verbeteren en kosten voor dedicated servers te vermijden, wordt het vaak gezien als een dood concept vanwege technische complexiteit en productiekosten. Dit onderzoek verkent de haalbaarheid van effectieve strategieën voor hostmigratie in online co-op games ontwikkeld in Unity. Verschillende benaderingen zijn onderzocht op prestaties, betrouwbaarheid en kostenefficiëntie, met als doel praktische oplossingen te bieden voor indie-ontwikkelaars. Niet enkel de haalbaarheid van hostmigratie, maar ook de schaalbaarheid van een implementatie ervan was uitgebreid getest en geanalyseerd.

# Preface

***A preface is a statement of the author's reasons for undertaking the work and may include personal comments that are not directly relevant to other sections of the thesis or dissertation.* No word count limit.**

This paper provides insight and analysis into the world of host migration, which is a fascinating concept over which many people hold strong opinions. With the uptick of online multiplayer games being developed in the last two decades, it is becoming more and more feasible for indie game developers to create their own multiplayer game. The standard approach for most larger studios are to use dedicated servers to ensure uninterrupted gameplay, but this approach is difficult financially for most indie game developers, including myself. I intend to make many small co-op experiences for the world to enjoy together, for a cheap price – I’m just a small indie developer after all, I cannot compete with content against AAA games. But, with no direct guarantee of sales or player revenue, paying for dedicated servers for the next few decades is an unrealistic choice which would lose me money, as if it’s a leak in my wallet forever. This sparked my interest in developing without paying for servers and letting people host their own. However, this proved to be difficult; there are very few resources out there. Personally, I’ve always strived to defy the odds, become the best and provide a net-positive influence on the world. This research could be a very valuable asset, sparking innovation in the industry through an otherwise forgotten practice. It’s time to revive the graveyard of host migration, once and for all.

I would like to express my gratitude to my coach, supervisors and partner Sara who provided guidance and support throughout this process. Special thanks to the developers and supportive community behind Mirror, especially MrGadget, whose insights contributed greatly to the technical foundation of this project.

It is my hope that this paper not only sheds light on the practicality of host migration but also inspires further research and innovation in indie multiplayer game development.

Louis Vanhove

11/01/2025

# List of Figures

**The list of figures lists the figures in the order in which they appear throughout the thesis. They may be numbered sequentially, or be subdivided following the chapters in which they appear.**

Figure 1: A picture showing something

Figure 2: A graph showing another thing

Figure 3.1: A tabel showing yet another thing, that appears in chapter 3.

# Introduction

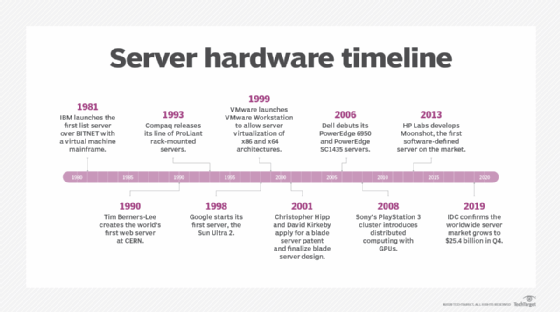
**In the introduction, you write the background of your topic and discuss the observation that spurred you on to do this research project. Explain the purpose of the paper and present your research question(s) and the hypothesis at the end of this section. This section is typically a couple of pages long**

## Background

### History

“The history of online games dates back to the early days of packet-based computer networking in the 1970s,[7] An early example of online games is MUDs, including the first, MUD1, which was created in 1978 and originally confined to an internal network before becoming connected to ARPANet in 1980.” (<https://en.wikipedia.org/wiki/Online_game#:~:text=History,-Main%20article%3A%20History&text=The%20history%20of%20online%20games,connected%20to%20ARPANet%20in%201980>.)

Since then, there has been an uptick in online games being created and played all across the world. The technologies on which these games rely have evolved greatly, together with the capabilities of computers and servers.



(<https://www.techtarget.com/searchdatacenter/feature/Dive-into-the-history-of-server-hardware>)

(<https://ourworldindata.org/data-insights/moores-law-has-accurately-predicted-the-progress-in-transistor-counts-over-the-last-50-years?utm_source=chatgpt.com>)

Let’s take a closer look at how players can connect to each other over the internet.

### Dedicated Servers

In more recent years, many games have started using dedicated servers as they have become more affordable and more powerful, offering reliability and stability. They allow for better performance (with dedicated resources and optimized hardware), as well as security and cheat prevention, allowing a controlled server to validate player actions, making sure no clients are tampering with their data. They can also provide better latency to players, given there are enough servers around the world for players to connect to. In the case the game is using Steam Datagram Relay (SDR), the latency can be improved even further through direct data funneling from relay to server, minimizing extra network hops from the network transport (usually UDP/TCP). (Valve GDC 2018 talk)

These are undeniably good features of dedicated servers, which the client-server architecture can not compete with. You cannot possibly have the same sense of security/cheat prevention, reliability or stability when you are putting the responsibility of hosting servers in the hands of your players.

Because of these advantages of dedicated servers, client-server architecture with self-hosting is rarely done by big studios anymore. This makes sense, they have made a name for themselves and have enough funding to host servers, allowing them to give their players a better experience. However, this is a fever dream that doesn’t last forever. More and more games using dedicated servers cannot keep supporting the large upkeep costs, as there are thousands of games being released every month, competing for the attention of the players. This can lead to less sales and user retention and less revenue for older games, which leaves the company to bleed money, providing servers for users who no longer make them much revenue. This is not maintainable for most studios, leading to mass server shutdowns, and unhappy users, who will even go to court (they paid for the game, you shut it down, taking away what they paid for).

(<https://www.gamesindustry.biz/ubisoft-faces-lawsuit-for-allegedly-misleading-players-by-shutting-down-the-crew>)

To put this in perspective, FIFA 22’s online servers are already shut down since November 2024. WWE 2K23’s online services are already shut down. These games have barely been out for a few years, and the massive studios supporting them choose to not sustain the upkeep of their games.

(<https://delistedgames.com/watch-list/>)

This makes sense from a business perspective, as paying for these servers for many players to use all across the world costs a lot of money. Players have high expectations of video games and their connectivity with low latency and no downtimes, even when the game gets a sudden influx of players, which the servers cannot handle.

### Community-Hosted Servers

Community-hosted servers represent a decentralized approach to multiplayer gaming, where individual players or groups host game servers independently rather than relying on developer-managed infrastructure. This model empowers gaming communities by offering greater control over server settings, game modifications, and community rules. Popularized by games like Minecraft, Counter-Strike, and ARK: Survival Evolved, community-hosted servers have fostered diverse and innovative gameplay experiences that cater to niche player interests.

For indie developers, enabling community-hosted servers can significantly reduce operational costs, as the financial burden of server maintenance and uptime shifts to the players themselves. This approach also encourages player engagement and fosters stronger community bonds, as players can create personalized gaming environments. However, community hosting introduces challenges such as inconsistent server quality, security vulnerabilities, and fragmented player bases, which may impact overall game stability and user experience.

Balancing the flexibility of community-hosted servers with performance and security considerations is essential. Implementing scalable networking solutions that allow seamless integration of community servers without sacrificing game stability can offer indie developers a viable alternative to costly dedicated servers, or a hybrid way of hosting which lowers the upkeep cost.

### PEER-TO-PEER

Peer-to-peer (P2P) networking is a decentralized multiplayer model where each player acts as both a client and a server. In this architecture, game data and state information are shared directly between players without relying on a central server. P2P networking is commonly used in smaller-scale or indie multiplayer games due to its simplicity and cost-effectiveness.

One of the primary advantages of P2P networking is its low operational cost. Since there is no need for dedicated servers, developers can reduce infrastructure expenses significantly. Additionally, P2P setups can offer lower latency in some cases, as data is transmitted directly between players without intermediary servers. This can lead to smoother gameplay experiences in local or well-connected networks.

However, P2P networking comes with notable challenges. It can suffer from connectivity issues, especially when players have varying internet speeds or are located in geographically distant regions. Security is another concern, as exposing players' IP addresses can lead to vulnerabilities like DDoS attacks or cheating. Moreover, if the host player disconnects or experiences lag, the entire game session can be disrupted, impacting the overall player experience.

To mitigate these issues, hybrid models combining P2P networking with relay servers have been introduced. Relay servers act as intermediaries, managing player connections without hosting the game state, thus improving connection stability and enhancing security. This hybrid approach allows developers to balance cost-efficiency with a more secure and stable multiplayer experience.

(<https://peerdh.com/blogs/programming-insights/understanding-peer-to-peer-networking-in-multiplayer-games?utm_source=chatgpt.com>)

### RELAY CLIENT-Server

A client-server model with a relay intermediary is a hybrid networking architecture where player data is routed through a relay server rather than direct client-to-client communication. In this setup, the relay server handles traffic between the host and clients, improving connection reliability and protecting player privacy. Unlike traditional dedicated servers that manage the entire game state, relay servers only forward data packets, reducing operational costs while offering many of the stability and security benefits of dedicated servers (Edgegap, 2023).

One major advantage of using relay servers is enhanced security. By masking IP addresses of clients, relay servers prevent direct exposure to potential DDoS attacks and other security threats (Edgegap, 2023). Relay networks also help resolve NAT traversal issues, allowing players behind strict firewalls to connect smoothly. This approach combines the affordability of peer-to-peer networking with the security and reliability closer to that of dedicated servers.

However, routing data through relay servers can introduce additional latency compared to direct P2P connections. The performance impact depends on the server's location relative to the players and the efficiency of the relay infrastructure. Despite this trade-off, relay intermediaries provide a cost-effective and scalable solution for indie developers seeking to enhance multiplayer stability without investing in full-scale dedicated servers.

A big benefit of this method is that your players can play together from any region around the world,

(<https://edgegap.com/blog/what-are-relays-servers-for-multiplayer-games-a-peer-to-peer-networking-guide>)

### Personal inspiration

I am a game programmer and developer, who is looking to make many small games and publish them for the world to enjoy. Since 2013 I’ve been inseparable from video games, starting out on my father’s iPhone with Minecraft: Pocket Edition and eventually getting the Java edition on my very first laptop. I learned to speak English through Minecraft, made many friends and made lots of fun memories. It sparked a passion inside of me for multiplayer games, especially games where you can play together with friends or strangers by your side. I had quickly fallen in love with the co-operative and competitive genre, leading me to spend much of my free time online. I started playing competitively in Counter-Strike, reaching the highest rank on both normal matchmaking and FaceIt (Level10). As well as being top500 in Overwatch open queue and competing in Fortnite’s first seasons, being in the top 150 players in Belgium. However, playing at a high level required time to practice, while much of my time goes to developing games nowadays. For this reason, I prefer playing co-op games with my friends from time to time, and it made me wonder if I could build such a game. There have been more and more independent game developers suddenly making co-op games that many players have loved (e.g Muck, Lethal company). I wondered if I could do the same, but quickly realised I had no budget or funding for servers. After some researching, I decided to take the plunge and start learning Mirror Networking for Unity. This led me to becoming a better developer, after which I decided to go to Unite Barcelona (2024). I decided to follow as many multiplayer talks as possible, to learn as much as possible for my future co-op games. There was a talk by Paolo Abela, a senior software engineer at Unity and online game consultant, at which I asked him whether he thought host migration was a necessity for good user experience. I had previously always heard it was a dead concept with no future which I shouldn’t bother with, but he shared his thoughts, detailing how it was important and I was fascinated. I went home and looked for research on it, but there was nothing official to be found, just a few example repositories and proprietary solutions from big companies. I finally had the chance to make a difference in this industry, however small it was.

### Purpose of the research

The primary aim of this research is to evaluate the feasibility of implementing effective host migration strategies in online multiplayer co-op games developed with Unity. This study focuses on identifying and comparing various host migration approaches based on performance, reliability, and cost-efficiency. The goal is to provide practical, scalable solutions that indie developers can adopt to enhance player retention and game stability without incurring the high costs of dedicated servers.

## Research questions

### Research question 1: CONTEXT FEASIBILITY

In which contexts is host migration a feasible solution in game development for online multiplayer co-op games?

### Research question 2: MIGRATION STRATEGIES

What are the most effective strategies for implementing host migration in a client-server architecture for online multiplayer co-op games for indie studios using Unity?

## Hypotheses

### Research question 1: CONTEXT FEASIBILITY

I suspect that the only contexts this will be a viable solution for is in co-op games where cheating does not matter at all, so no leaderboard mechanics or anything like that present. When the host disconnects and the new host fills that role, I presume the data of the lobby could be manipulated to skip waves, kill other players, teleport to unreachable locations, skip to the end of the game instantly, etc.

I also hypothesize that the scope of the game cannot be too large as a lot of data syncing might take a long time and not be synced properly. Games with lots of content like Realtime Strategy Games (RTS) would probably be a bad fit,

### Research question 2: MIGRATION STRATEGIES

**Hypothesis 1: Ordered list reconnection**

Designating an ordered list of new hosts when starting the game (that just started, not the lobby) synced across clients. When the host disconnects, the clients reconnect to the next host in line. The client which will become the host knows about this as the list is ordered and synced. I assume this approach would work best, as there’s no unnecessary bandwidth usage - but this only works if the new host manages to save the information to reconstruct the scene. This might prove difficult as when the host disconnects, the connection to the server cuts off.

**Hypothesis 2: Interval state synchronization**

**Implementing regular state synchronization with a secondary client will maintain game state consistency during host migration. This could be done with a tick rate system, sending over all necessary information to the players to reconstruct the scene. This data has to be persistent though, so it does not get lost when the host leaves. I assume this approach will be the easiest and cheapest to implement, but this will affect latency and there might be a short pause while data gets deserialized and the scene gets reloaded. Some data might be lost between intervals, depending on how short the ticks are. Sending over entire saves will not be possible, it will have to be incremental (latency impact, can’t send over megabytes of data every few ticks).**

**Hypothesis 3: Co-host (free)**

Designating a co-host when the game starts, which acts like a fail-safe for the game. This co-host is chosen when the game starts (not the lobby, the actual game) by best ping, and is sent all of the information the server receives. This would double the network traffic for this user, but ensure that no data is lost.

**Hypothesis 4: Relay intermediary (free)**

For indie studios running their online multiplayer game on Steam, the transport can be sent through a Steam relay. This makes it so that data is networked with Steam as a middleman, simply passing through the data to other clients, allowing for a seamless, free host migration. This procedure could be done with any free relay service.

**Hypothesis 5: Dedicated server backup (paid fallback case)**

 If the previous approaches fall through, using a dedicated server or a cloud-based relay which re-routes the network to some listen server might be the only way to solve this. This would be paid, which would not be great for indie studios, but if it’s the only possible way, it’s something to consider for the UX. This is the way Photon (PUN2) handles it behind the scenes, so it is definitely possible - but dedicated servers cost money and this might not be the best option for indie studios.

# Theoretical Framework

**In the literature review, you present the secondary research you have conducted. You detail the background of your topics and write about the concepts that are relevant to the study. Assume that not every reader has the same skillset or -level as you do! This section typically requires a substantial amount of references and can be a lengthy section that requires a considerable amount of pages.**

Things to talk about:

### Player experience: (reviews here)

The most important reason to consider host migration is to ensure that your players have a good experience playing your game; you could make the most fun game possible, but if all of the players lose their rewards because the host leaves the game, that’s an issue. Such an experience is inevitable when you make a game where your players host their own lobbies, and function as the server, without host migration implemented. No matter how well you design your game around this, people’s internet will cut out, computers will crash, or sometimes the host will just want to stop playing the game; he’s not obligated to be there.

This begs the question: Is it actually important? Do players really care about this?

9999 Potential data analysis here from steam reviews 9999

Some examples of negative reviews, publicly posted on the Steam page of Liar’s bar, a game about bluffing released in October 2024, made with Mirror, using Client-Server architecture without host migration. (<https://store.steampowered.com/app/3097560/Liars_Bar/>)

Afbeelding met tekst, schermopname, software, Multimediasoftware

Automatisch gegenereerde beschrijving Afbeelding met tekst, schermopname, software, Multimediasoftware

Automatisch gegenereerde beschrijving

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving

Afbeelding met tekst, schermopname, Lettertype

Automatisch gegenereerde beschrijving Afbeelding met tekst, schermopname, software, Multimediasoftware

Automatisch gegenereerde beschrijving

This is of course just an extract from the page, the overwhelming majority of the reviews are positive, but most of the negative reviews revolve around cheating or the host leaving. Cheating is something we’ll discuss later (what niches is this architecture viable). I believe the developer could have benefitted from using dedicated servers.

For the host migration, you can see many players in the reviews so upset about the host leaving, that even though they’ve played this small game for many hours, it bothers them enough to write a negative review. Again, this can be attributed to the game design (player dies and has to wait for the game to finish) – but this is something that could have been prevented with host migration.

### Networking:

There are quite a few topics to cover over networking; host migration is simple on paper but not an easy concept under the hood, it takes quite a lot of effort, time and knowledge to create. We will discuss the necessary topics briefly, but this is not a paper which will explain the entirety of game networking to you. For this, I can recommend the paper “Building Multiplayer Games in Unity using Mirror Networking” by Dylan Engelbrecht. This was a great read and how I learned most of the networking topics, with not much background knowledge.

#### Packets

* Ports
* Portforwarding
* IP address
* IPV4
* NAT
* TCP vs UDP
* KCP
* Steam transport
* NAT
* ISP:

Dedicated servers:

Client-server architecture:

Peer-2-Peer:

Edgegap: (dedicated servers)

Networking choices in Unity (Photon, NGO, UNET, Fishnet, Mirror)

Mirror:

* Commands
* Rpcs
* Syncvars
* Order of events

Unity:

* DDOL

Csharp:

* Reflection
* Code injection
* Inheritance

Cost analysis

Host migration basic explanation:

# Research

**In the research section, you detail the elements of your experiment(s), the tests, objects you will test upon and subjects you will test with, the data gathering, data cleaning or feature extraction, measurements, … and you present the results obtained in an objective manner for each of the tests you conducted.**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

## GAME PROTOTYPE

### Subtopic 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

### Subtopic

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

## HOST MIGRATION EXPERIMENTATION (yap eos, DISCORD LOOKUP, EXAMPLEs, …)

### Subtopic 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

### Subtopic

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

# Discussion

**In this section, you offer an interpretation of the results you obtained and try to relate them to the theoretical framework you presented. This is typically not a very long section, but obviously one of the most important ones.**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

# Conclusion

**In this section, you ascertain the demonstrable outcomes of your study and outline the merits of the project for the academic field and the discourse community. This is typically not a very long section, but obviously also one of the more important ones.**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

# Future work

**This section is sometimes standalone, sometimes incorporated in the conclusion. It looks at the shortcomings of the study, alternative strategies, and what could be the next course of action in the research field. This is typically not a very long section.**

 What if the backup host leaves at the same time as the host leaves?

Late joiners?

# Critical Reflection

**This section is typically associated with a bachelor paper, not other forms of serious writing. It allows the student to reflect on the learning outcomes, both academically and in terms of personal growth.**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis.

Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

# References

**In this section, you list all the references you made in alphabetical order; consequently adhere to the referencing style you have chosen.**

Casey Raes (2014), Processing (second edition).

Saccade. (n.d.). In Wikpedia. Retrieved November 6 2016 from <https://en.wikipedia.org/wiki/Saccade>

Sarah Northway (2016) A year in VR Northway [Powerpoint slides] from <https://www.gdcvault.com/play/1024631/A-Year-in-VR-A>

# Acknowledgements

**In this section, you can thank people who contributed to your work in a meaningful way.**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis. Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.

# Appendices

**In many cases, there are items that were developed for a research paper that can’t go into the actual paper in full. Things such as code, art pieces, output of statistical analysis, questionnaires, … In this section, you can present these elements; use the first page to list and number the items, then paste them sequentially. If some items are too large, you can store them online, and link to them. Common practice is to keep those links active at least one year after the publication of the thesis.**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vestibulum ac quam nec arcu semper dignissim. Nulla quam magna, varius sit amet pharetra et, dictum quis elit. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Proin ullamcorper, ipsum sit amet scelerisque rhoncus, leo quam rhoncus elit, sit amet ullamcorper tellus nisi eget sapien. Suspendisse potenti. Ut non justo viverra, tempus felis vitae, elementum mi. Morbi at dui sed lacus fringilla condimentum. Duis non odio ac arcu volutpat vehicula eu et turpis. Praesent vitae magna ante. Nulla in orci lacus. Donec quis vestibulum mi. Sed ipsum sapien, pretium maximus purus sed, bibendum consequat lectus. Aliquam porttitor dolor eu gravida vulputate. Vestibulum ut urna eget massa tincidunt ultricies. Morbi hendrerit sapien at diam tincidunt semper. Aliquam ut quam dictum quam maximus tempor sed at felis.