

What C++ architecture could best support Quality-of-Experience-oriented map streaming in a multiplayer building game?

COMP130 - Computing Architecture

1707981

March 23, 2018

Low broadband speeds across the world could hinder the advancement of multiplayer gaming, particularly for traffic-hungry projects such as MMOs or building games. From a creative standpoint, player experience is decidedly the most important factor in a game's design. When downloading the game environment, low bandwidth increases download speeds and temporarily strips players of their autonomy, causing frustration. To aid this, Quality-of-Experience (QoE) techniques are proposed to mitigate the negative effects of limited bandwidth on the player experience, in a way that could be adapted to both new and existing games.

1 Introduction

The games market is continuing to grow at an immense rate, especially in developing countries. In East Asia, a growth of nearly 15% over the last year has been observed, as

shown in Fig. 1.

This presents profitable opportunities for game developers to expand into new territories. However, a lag in technological advancement, particularly broadband, raises reasonable concerns of whether they can fully exploit these opportunities. The potential of games with building elements, such as the extremely popular (as suggested by Google Trends [1]) Minecraft and Fortnite, may be hindered by a lack of bandwidth necessary to download map resources at an acceptable speed. The slow download may frustrate players by rendering them unable to access areas of the game in a timely manner.

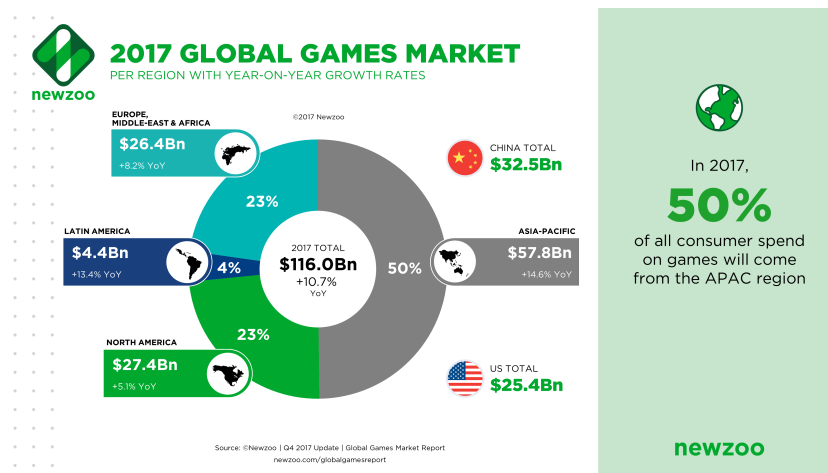


Figure 1: The games market in 2017, according to NewZoo (Source: [2])

This study aims to discover the extent of this problem, followed by proposals of architectural methods to implement systems that could mitigate the negative effects of an unreliable networking environment on the player experience. These can be described as QoE (Quality-of-Experience) improvements where the bandwidth is not improved, but instead coordinated toward where the player needs it most. The programming language in question is C++, while the hypothetical game assumed a 3D environment wherein the player can manipulate the terrain and build meshes. The goal is to propose a reasonable set of design patterns to be used for QoE-oriented map streaming in a large-scale game,

and how they could be applied to an existing game engine.

This analysis will take place over the following steps:

- Exploration into the games market and its broadband speeds
- A brief introduction to player experience as a separate concept to content quality, to gain an understanding of what should be prioritised
- A priority list of assets and features of immediate interest to the player
- A proposed technical outline of a system that could facilitate level streaming with priority on player experience.

2 The Big, Slow Market

China is the most profitable games market as of 2017 [3]. Furthermore, according to SteamSpy in March 2018, 8 of the 10 most popular Steam games in China were partly or fully multiplayer [4]. This suggests China is a valuable target for online games. However, compared to most developed countries, China's Internet speeds lag behind (figuratively speaking), at an average of 7.6Mbps [5].

Similarly, another country notorious for its online gaming community is Brazil. According to SteamSpy, 9 of the top 10 games had a multiplayer component, with 8 boasting primarily multiplayer gameplay [6]. Brazil's Internet speed averaged about the same—6.8Mbps—in 2017 [5] (an ironic failing for the world's fifth top source of DDoS attacks in early 2017 [7]).

Speeds in the UK and US meanwhile average at over twice as much, 16.9Mbps and 18.7Mbps respectively. While this study was originally expecting a bigger discrepancy, it could surmise that the map streaming engines of large-scale player-built online games will require optimisation to compensate the lack of data bandwidth compared to, for example, disk speeds. In the Sony PS4's case, streaming speed from the disc maxes out

at 27MB/s (at a 1x speed of 36Mbps [8] multiplied by 6 [9])—a significantly higher rate than Brazil’s average Internet speed of 6.8Mbps, or 0.85 MB/s. This is only 3% of the typical disc data rate.

There is a lack of definitive research on the impact of low broadband speed in map streaming. A point of reference or scale in terms of user experience is therefore lacking, making the actual negative impacts of low bandwidth relatively uncertain. Until this is further understood, this essay will create the system around scalable broadband speed, aiming to guarantee an acceptably ‘fun’ quality even in extremely slow network conditions. However, the definition of ‘fun’ may be at odds with the industry’s tendency to create games that are bigger, more detailed, and overall more data-heavy as they evolve [10]. This prompts an exploration into what is ‘fun’, in a term more broadly described as quality-of-experience.

3 Quality of player experience

Video streaming is a heavily researched area; perhaps more than games, owing to significant dedicated R&D toward streaming services such as the Amazon FireTV [11]. Yet considering the link between video quality and quality of gameplay aesthetic—including pixel resolution, lag spike rate, and frame rate—the basic principles and goals could perhaps apply to games as well.

In the context of video, it has been documented [12] that user enjoyment is greater in a scenario of low-detail, constant frame streaming, than one of higher-detail, inconsistent frame streaming. In other words, the user is not interested in the rate of high-detailed data, but in enjoying a smoother experience.

In the context of games, this smooth experience is also shown to be necessary to retain game players. Studies have noted latency [13] [14], jitter and packet loss [14] to all have negative impacts, sometimes leading to quitting altogether. While this is not widely studied in building games (instead FPSes and MMOs), games such as [15] Fortnite,

which features both building and shooting gameplay, demonstrates that the two are not mutually exclusive.

Another study [10] noted the importance of animations in a game. Similarly, this centres around the concept of motion, and as above, suggests that graphical detail is of smaller importance than the player’s ability to gain immediate feedback. This is supported at a base psychological level by R. Ryan and E. Deci’s research on human motivation [16]. Whilst heartily citing themselves throughout their highly reputable paper, they observe a human desire to explore, apply their skills, and learn from the feedback.

These findings place high importance on the motion, control, and abilities of the player rather than visual fidelity of the background. In map streaming, this translates to the importance of giving the player base landscapes to move within and toward. This places the level mesh, and perhaps interactive objects (noting the need for motion), above textures and sounds in the priority list.

4 Quality of experience – Priorities

Based on the above findings, a preliminary priority list of asset downloads is proposed:

- Player movement, to enable initial player autonomy and feedback [16]
- Player avatar, so they can see where they are, enhancing the above
- Untextured level geometry (nearby), so the player has locations to visit
- Interactive objects, for further feedback
- Player animations and object animations to deliver the aesthetic [10]
- Untextured level geometry (distant, low-detail) for greater immersion
- Textures and sounds to further enhance and complete the aesthetic

- Additional resources.

However, this is not an exact science. For example, level textures may come in a low-quality form before their final form. This would show evidence of loading progress to the player. While it could be argued as invasive, a player would likely deem this more favourable than—for example—a loading screen, as observed in software studies [17]. Furthermore, level textures might not necessarily be streamed, unless the players should be able to draw their own.

As an additional note, the specifics of animation, player model, object and texture streaming are beyond the scope of this essay, but are noted above for the reader's interest.

5 Proposal

The proposed system base, designed to prioritise the player experience, is illustrated below as two components, the client component and server component. These represent the game and the persistent online server, respectively.

5.1 Server system

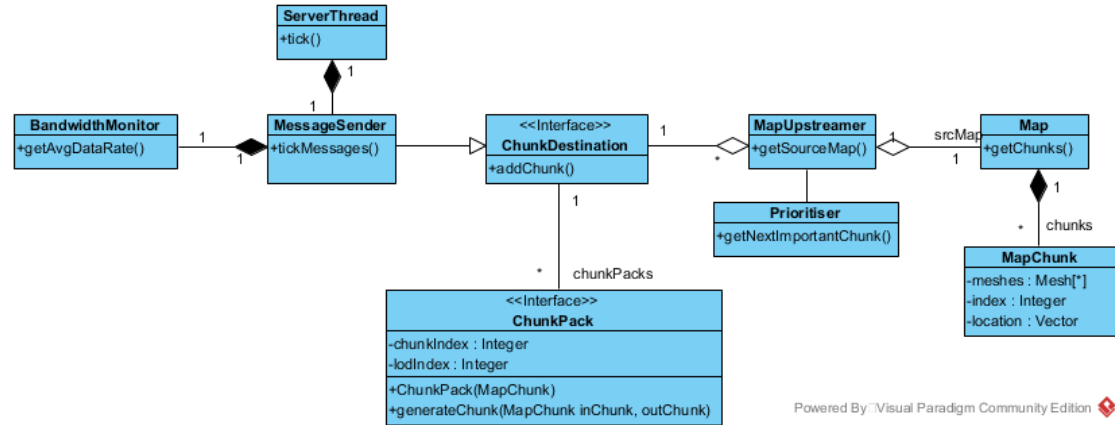


Figure 2: Partial class diagram illustrating the potential map streaming server

The proposed system would comprise the following key classes:

- **Map**: The base map class. This is shared between the client and server. By partitioning the map into several **MapChunks**, separate pieces of it can be downloaded at a time, allowing the player to roam incomplete areas in short notice.
- **MapChunk**: A portion of the map, with an index identifying the specific area of the level, and mesh information. The shape of a level chunk may vary across games, though in a building game, a uniform shape could be chosen such that there is consistency in the efficiency of building in one place over another. However, different games may use different strategies to best match their usage scenario. For example, *Sunset Overdrive* (Insomniac Games, 2014) used hexagonal chunks by design and radial chunks in practice. This was designed to facilitate the player's freedom of movement in every direction. However, an earlier game by the same company, *Fuse* (Insomniac Games, 2013), used rectangular chunks. This was

optimised for linear hall-based gameplay [18]. In a building game, the former would most likely be appropriate, as most games would likely take place in a wide virtual world so that the player can build large structures.

- **ChunkDestination:** A collection of **ChunkPacks** that may be implemented in multiple ways. In this case, it is connected to **MessageSender** for immediate transmission. In more advanced cases, it could also, at a later stage, be implemented on the client machine as a map storage manager, which could cache map data on a local machine for later use.
- **MapUpstreamer:** The MapUpstreamer is effectively a *Mediator* (described as a class that manages the collaboration between objects [19]) technically owned by neither Map nor ChunkDestination. This method was chosen so that MapUpstreamer could, if necessary, operate independently in parallel to the other tasks. An observer pattern could reasonably be used, but this typically means synchronising the data on an immediate basis [19], something which is not necessary for the time-sensitive task of prioritising the data flow.
- **BandwidthMonitor:** To achieve data rate consistency as likely desired by players (see Section 3), this class monitors bandwidth over time and adjusts the data rate to the most consistent possible. The specifics of this system, better described as bandwidth-smoothing, are beyond the scope of this essay. A broad comparison of bandwidth-smoothing techniques as illustrated by W. Feng and J. Rexford is recommended for further reading [20].
- **ChunkPack:** This interface stores components of map data in various possible class-specific ways, and provides the function *generateChunk* such that it can inject itself into the Map. See examples of subclasses in the following section.

5.2 Client system

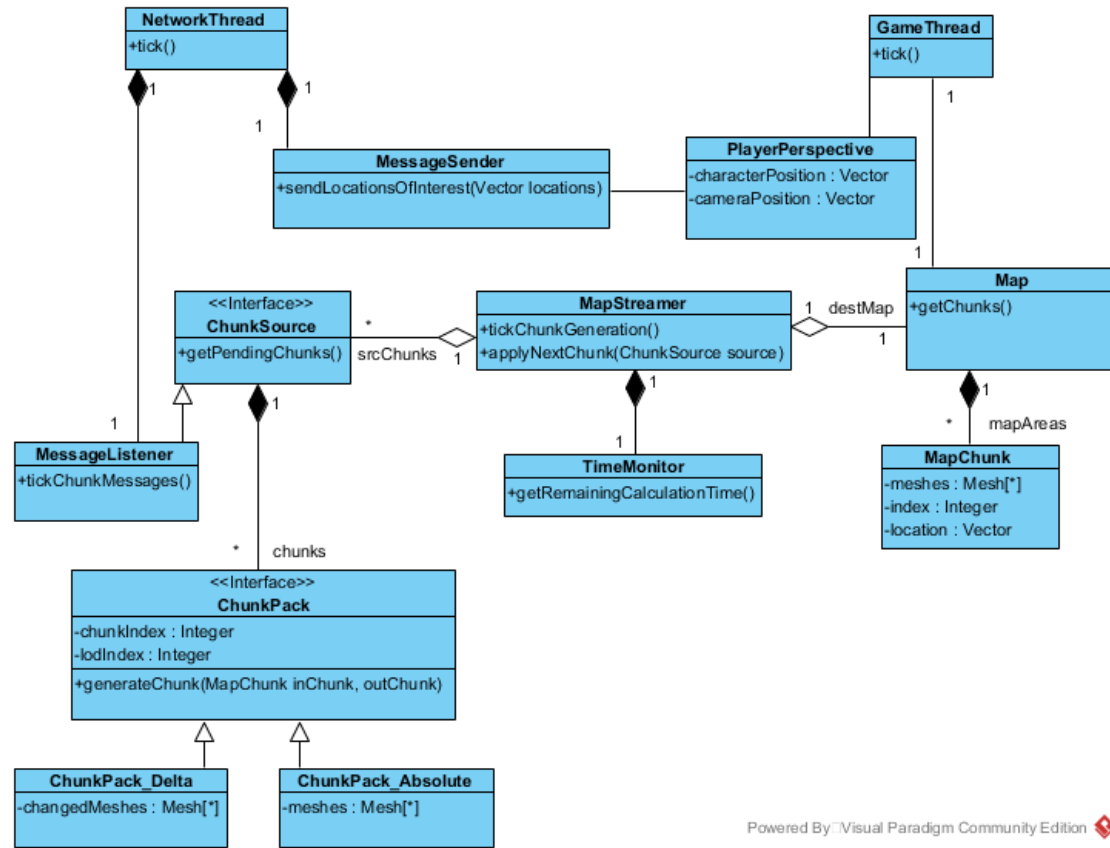


Figure 3: Partial class diagram illustrating the potential map streaming client

To support the independence of the background download process, which focuses on delivering data in a time-sensitive manner, this process is divided into two concurrent threads. This system comprises the following key classes, some of which are shared with the Server system:

- **MapStreamer**: Responsible for collecting **ChunkPacks** and transferring them to the physical **Map** when it deems appropriate. This is another example of a

Mediator. It relies on **TimeMonitor** to deliver the chunks smoothly.

- **TimeMonitor**: To ensure a smooth experience, this determines the amount of processing time to dedicate to chunk generation. This will be dependent on the game's other systems, and there are many factors to consider—such as used CPU time, network speed, the time it takes to generate an average chunk—to optimise this delivery time.
- **PlayerPerspective**: Responsible for informing the server where the player is in the map, so that the server can prioritise chunk delivery.
- **ChunkData_Absolute**: Contains a complete map chunk at a particular level of detail.
- **ChunkData_Delta**: Contains a map chunk in terms of changes that have occurred since an earlier version. This will be useful for player building events; for example, when a player constructs a wall. Contrast with rebuilding a mesh from scratch, this may return only meshes that have been added or removed, saving time and memory.

6 Conclusion

A system architecture focusing on delivering data to the player in a consistent [12] [14], immediately interactive [16], and smooth [10] [14] way is proposed to optimise the player experience. While the development environment is implied to be C++, it is likely that this architecture could be applied to most modern object-oriented languages.

There is much room for further research and expansion, including but not limited to compression, variable chunk scaling, texture streaming, gameplay element streaming and sound streaming techniques. A more detailed study could describe the particular methodology in the forms of flowcharts and pseudocode to help formulate the model.

Furthermore, lack of research on the impact of bandwidth on building games leaves the true fidelity and efficiency requirements of this system to be quite undefined. Observing the popularity of Fortnite today, which now matches Minecraft as of March 2018 [1], the game industry is likely preparing to meet a new demand for games with building elements. Should this happen, their development should hope to drive this area of research forward.

References

- [1] T. Kim. <https://www.cnbc.com/2018/03/15/fortnite-is-becoming-biggest-game-on-internet.html>, Mar. 2018. Accessed: 16/03/2018.
- [2] “Key resources.” <https://newzoo.com/resources/>, Oct. 2017. Accessed: 16/03/2018.
- [3] C. Cui and P. Harding-Rolls, “China - largest games market continues its growth story.” <https://technology.ihs.com/593405/china-largest-games-market-continues-its-growth-story>, June 2017. Accessed: 13/03/2018.
- [4] “China - Country Stats - SteamSpy.” <http://steamspy.com/country/CN>, Mar. 2018. Accessed: 18/03/2018.
- [5] “Q1 2017 State of the Internet Connectivity Report.” <https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/q1-2017-state-of-the-internet-connectivity-report.pdf>, May 2017. Accessed: 20/03/2018.
- [6] “Brazil - Country Stats - SteamSpy.” <http://steamspy.com/country/BR>, Mar. 2018. Accessed: 20/03/2018.

- [7] “Q4 2017 State of the Internet Security Report.” <https://www.akamai.com/us/en/multimedia/documents/state-of-the-internet/q4-2017-state-of-the-internet-security-report.pdf>, Feb. 2018. Accessed: 19/03/2018.
- [8] “Blu-ray FAQ.” <http://www.blu-ray.com/faq/>. Accessed: 22/03/2018.
- [9] “Tech specs (PS4).” <https://www.playstation.com/en-gb/explore/ps4/tech-specs/>. Accessed: 22/03/2018.
- [10] E. Andersen, Y.-E. Liu, R. Snider, R. Szeto, and Z. Popović, “Placing a value on aesthetics in online casual games,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI ’11, (New York, NY, USA), pp. 1275–1278, ACM, 2011.
- [11] I. Lunden. <https://techcrunch.com/2012/07/23/amazon-ramps-up-global-expansion-opens-massive-media-rd-center-in-london/>, July 2012. Accessed: 23/03/2018.
- [12] D. C. Robinson, Y. Jutras, and V. Craciun, “Subjective video quality assessment of http adaptive streaming technologies,” *Bell Labs Technical Journal*, vol. 16, pp. 5–23, Mar. 2012.
- [13] S. Zander and G. Armitage, “Empirically measuring the qos sensitivity of interactive online game players,” in *Australian Telecommunications Networks & Applications Conference*, pp. 8–10, Dec. 2004.
- [14] K. T. Chen, P. Huang, and C. L. Lei, “Effect of network quality on player departure behavior in online games,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 20, pp. 593–606, May 2009.
- [15] “Most Played Core PC Games (US & Europe).” <https://web.archive.>

org/web/20180323094757/https://newzoo.com/insights/rankings/
top-20-core-pc-games/, Mar. 2018. Accessed: 23/03/2018.

- [16] R. M. Ryan and E. L. Deci, “Intrinsic and extrinsic motivations: Classic definitions and new directions,” *Contemporary Educational Psychology*, vol. 25, no. 1, pp. 54 – 67, 2000.
- [17] J. Hohenstein, H. Khan, K. Canfield, S. Tung, and R. Perez Cano, “Shorter wait times: The effects of various loading screens on perceived performance,” in *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA ’16, (New York, NY, USA), pp. 3084–3090, ACM, 2016.
- [18] E. Ruskin, “Streaming in sunset overdrive’s open world.” Game Developer’s Conference, 2015.
- [19] E. Gamma, R. Helm, R. E. Johnson, and J. M. Vlissides, “Design patterns: Abstraction and reuse of object-oriented design,” in *Proceedings of the 7th European Conference on Object-Oriented Programming*, ECOOP ’93, (London, UK, UK), pp. 406–431, Springer-Verlag, 1993.
- [20] W.-C. Feng, P. P. Mishra, and K. K. Ramakishnan, “A comparison of bandwidth smoothing techniques for the transmission of prerecorded compressed video,” in *INFOCOM ’97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Driving the Information Revolution., Proceedings IEEE*, vol. 1, pp. 58–66 vol.1, Apr 1997.