A SURVEY ON THE IMPORTANCE OF QOS PARAMETERS ACROSS NETWORKING DOMAINS

Juan Flores, Oscar Sandford, and Ben Wunderlich

University of Victoria

Abstract

Quality of service (QoS) parameters offer a huge help in measuring the performance of new network ideas and implementations. Knowing which QoS metrics to use in different scenarios can help save time and resources, leading to cost reduction and future service flexibility and scalability. We use a previously defined taxonomy of QoS parameters, divided into metrics and policies, in order to objectively evaluate parameter prioritization in online multiplayer games, peer-to-peer (P2P) file transfer systems, and video streaming services. We show that these networking domains vary greatly in terms of their motivations and design, leading to variation in consumer and infrastructural policy. As a result, the technical QoS parameters are influenced by QoS policies motivated by target user experiences. Qualitative evaluation of these parameters and their share of contribution to a networking domain's QoS is presented, and the domains themselves are assessed for shortcomings in their service implementations. We find that security is sorely lacking in priority, and that cloud computing has the ability to reinvision the QoS policies of the future. As such, we encourage periodic analysis and evaluation of these taxonomic features in a variety of networking domains, in order to establish and maintain accurate notions of quality of service.

Introduction

The design of computer networks is compelled by a drive towards a target quality of experience (QoE) and quality of service (QoS). These measures define the success of a network service in a competitive market. It is important to make a distinction between QoE and QoS. QoS is the set of characteristics in a computer networks system that affect its ability to satisfy the utilitarian needs of its users [1]. QoE focuses closer on the end-user, and includes user perception, expectation, and their specific experiences using the service [2]. To expound on that, [1] states that quality of service is viewed from a system's perspective, and QoE from a user's perspective. However, it is important to note that, at the end of the day, quality of service mechanisms are built in order to provide the end user with a solid quality of experience [3]. This study will primarily focus on quality of service, as it is easier to evaluate due to its quantitative nature. QoS requirements stem from domain and system requirements, rather than the requirements of an end user, which are difficult to quantify. We will primarily consider quality of service in this study, in the interest of isolating the analysis of a network to its technical requirements and how well companies correctly prioritize the maintenance of their system in order to best fit a generalized view of customers' needs. However, it will be necessary to address exactly how these requirements come into place, requirements that percolate from a picture of the end user's ideal experience.

Countless studies in computer networking have addressed quality of service when it comes to the development and analysis of new algorithms, measurements on the efficacy of existing solutions, as well as domain-specific requirements for specific systems. However, various domains have differing standards on quality of service for their applications. It is difficult to correctly understand what it means for an application to have a “good” quality of service, especially under the requirements of the application domain itself, without giving weights, or prioritization, to certain parameters. This study surveys a set of well-defined, domain-agnostic quality of service parameters and defines each parameter’s importance in the context of different networking domains. Such domains include online multiplayer gaming, peer-to-peer file transfer services, and video streaming.

Related Work

Software-defined networks (SDN) address the problem of static architectures by enabling programming and measurement in a dynamic setting. In order to help developers focus more on the design goals and less on tedious implementation tasks, SDN research has inspired the utilization of network emulators [4]. Network emulators are often used to test the quality of service of server architectures in a pseudo-live setting. Experiments with such emulators reveal key quality of service parameters that have been used in domain-agnostic settings. Network emulation using Mininet in order to test quality of service parameters for TCP and UDP has been used in [5, 6]. The metrics presented in [5] include throughput, delay, packet loss, and jitter. The metrics are further defined as follows:

- Throughput (total transmitted data in bits)/(total time taken in seconds)
- Delay (time required to transmit the data from sender to receiver)
- Packet loss (the number of packets not delivered to their destination)
- Jitter (the variance in latency)

The authors used these metrics in a generic setting (i.e. no specific domain) in order to compare TCP and UDP protocols. While this domain-agnostic approach makes for a good start, there are several flaws in their results. Firstly, no packet loss was experienced, and that measurement was thereby ignored. Secondly, previous studies [6, 7, 8] used round trip time (RTT) as a quality of service metric, which [5] did not consider. Another study that investigated load balancing algorithms looked at throughput, response time, and memory utilization [9]. Measuring memory utilization is effective for load balancing, but we are considering quality of service, with the focus on the end user. The weight on the server only matters if it affects the client. This paper reinforces throughput as a necessary metric, and that response time (or RTT) must be considered.

A different paper [10] proposes a framework titled AweQoS which combines both quality of service tests with security and reliability tests. The framework includes bandwidth speed measurement, delay tests and jitter tests, as well as SYN flood test, UDP flood test, and Slow HTTP flood test. In order to provide foundation for quality of service analysis in differing domains, [11] created a taxonomy for clustering specification into categories. We will reuse their specification in our approach, as well as extend it in relation to the domains we will be looking at.

Approach

As we conclude our investigation on common quality of service metrics used in simulated and domain-agnostic settings, we compiled a set of general quality of service metrics as well as policies, according to specifications outlined in [11]. The following hierarchy of parameters is based on the general taxonomy outlined in [11], with additional notes pertaining to parameters discussed in prior related works. At this point, we make the distinction between metrics and parameters, where the former is a subset of the latter. Metrics reference the quantifiable attributes of a system's components, whereas the policies decide their behaviour [11].

Metrics cover the following attributes:

- Security (robustness against malicious action)
  - Confidentiality (information is received by intended party)
  - Integrity (information remains accurate)
- Performance
  - Timeliness (e.g. delay, latency, RTT, etc.)
  - Precision (i.e. consistency, e.g. jitter)
  - Accuracy (i.e. lack of errors, e.g. packet loss)
  - Combinations (i.e. of the above, e.g. throughput: precision over time)
- Relative Importance (i.e. cost of given service to user)

Policies involve the following:

- Levels of Service (i.e. commitment to a task, e.g. guaranteed or best-effort)
  - Availability
- Management (i.e. resource management, e.g. to accept lower QoS over no service at all)

Note the inclusion of security as a general metric. While many applications simply prioritize speed and reliability, security is key to some application domains. The security section covers two of the three points in the security triad: confidentiality and integrity. Availability in the context of QoS includes security, but is adjacent to other factors such as resource management. For that reason, availability is the main level of service parameter, where service policies set the bar for acceptable levels of services in terms of availability to the end user [3]. Reliability issues (system faults not as a result of a malicious attacker) are covered by the performance category. Readers should also note that effective bandwidth can be referred to as throughput [3], which is a combination of precision over time, which are two other metrical parameters [11].

We are now armed with the aforementioned taxonomy of quality of service parameters, from performance metrics to application-dependent policies. In the following section, we analyze the prioritization of these parameters with respect to the requirements of the networking domain, as well as its specific applications within that domain. Each metric is ranked in terms of its importance, in order to provide pseudo-quantitative supplements to our qualitative analysis. Graphics are presented in order to visualize the distribution of quantitative parameter priority with respect to the application domain. Policies, which are generally defined in the context of the application, are analyzed and evaluated by comparing and contrasting their implementations in each application domain. The metric prioritizations will be reflected in how policies are defined and enforced. These links are discussed in detail.

Domain Analysis

Online Multiplayer Games

Throughout history, games have always been a medium for social recreation as well as serious competition. Necessarily, deciding the requirements for a “good game” are complex and highly application-dependent. This is made more delicate in computer games, where the end goal is an immersive experience. In this way, the field of game design is about conveying possibilities to the player and giving them the freedom to take action on their own accord [12]. These propositions are very generalized, but we can see clear links from the demands of user experience to technical quality of service requirements. For example, an immersive experience implies that the game’s feedback should be delivered in “real time” to the player, and avoid fragmentation (such as delay, often referred to as “lag”) that betrays its artificial origins. Many games do not require a network connection, and can be played in “single-player” mode, meaning the player experience is dictated by the quality of the game software and the computer setup of the end user. However, computer games since the 1990s have begun adopting various networking features that enable competitive and social connection with tens or thousands of players, such as World of Warcraft [13]. It is necessary to understand the service demands of computer games in order to facilitate fun, high-fidelity experiences between players connecting across the globe.

A study by Bredel and Fidler investigated the link between quality of service and player performance in first person shooter (FPS) games, which are noted as having the highest QoS requirements when compared to other genres such as role-playing games and turn-based strategy games [13]. Their experiments involved the FPS game *Quake III*. Typically, most FPS games like this make use of a client-server architecture, wherein the client handles inputs and game engine renderings, and the server handles input processing, world simulation, and employs lag compensation techniques [13]. Lag compensation is done on the server-side, and is a result of the inability to reliably solve performance metrics such as delay or jitter on the two-way route from the client to the server. However, most existing lag compensation systems can unfairly enhance more-lagged players and compromise the experience of players with a smoother connection [14]. While hacking is certainly prevalent in online games, the server's handling of input processing can help mediate attacks to the integrity of a user's input, as well as the assurance that other players will receive "clean" data from the server.

Even given the instance of a smooth real-time experience in multiplayer settings, the expectations for metrics such as delay are actually more relaxed than one would expect. Studies show users initially avoiding connecting to servers with 150-180 ms round-trip times in *Quake III* and 225-250 ms in *Half-Life* [15, 16], but remaining in the server once joined, even as delay is increased [17]. However, this does not mean delay is inconsequential. Wattimena *et al.* show that timeliness metrics like delay and jitter in *Quake IV* have a pronounced negative effect on user experience, while packet loss (an accuracy metric) does not negatively affect game play up to 40% [18].

Bredel and Fidler's *Quake III* study used quantitative scoring to measure QoS parameter influence on player performance. What they found was that scoring probability decreases linearly with respect to an increase in delay, and jitter less so. Packet loss exponentially affects player performance negatively, especially when packets are dropped enroute to client from server, instead of vice versa, due to the server updating the client at a longer interval than the client transmitting to the server [13].

A good quality of service for online multiplayer games extends from simple performance metrics into the technical policies that enable them at scale. Clever resource management policies are necessary when it comes to serving Massively Multiplayer Online Games (MMOGs) [19]. On the one hand, the problem of availability can be solved by allocating large amounts of capital towards thousands upon thousands of machines in order to facilitate the staggering QoS requirements of so-called AAA ("triple-A") games, the blockbusters of the gaming industry. Steam, an application portal for purchasing and playing games, records great variance throughout a given day in its concurrent user counts, with a peak-to-trough delta of almost 10 million users [20, Fig. 1]. Simply keeping thousands of machines online will inevitably lead to unnecessary idling and inefficient use of resources. After data like this is collected, Steam can effectively predict times in the day of high resource requirements from their users, and plan resource allocation and load balancing accordingly.

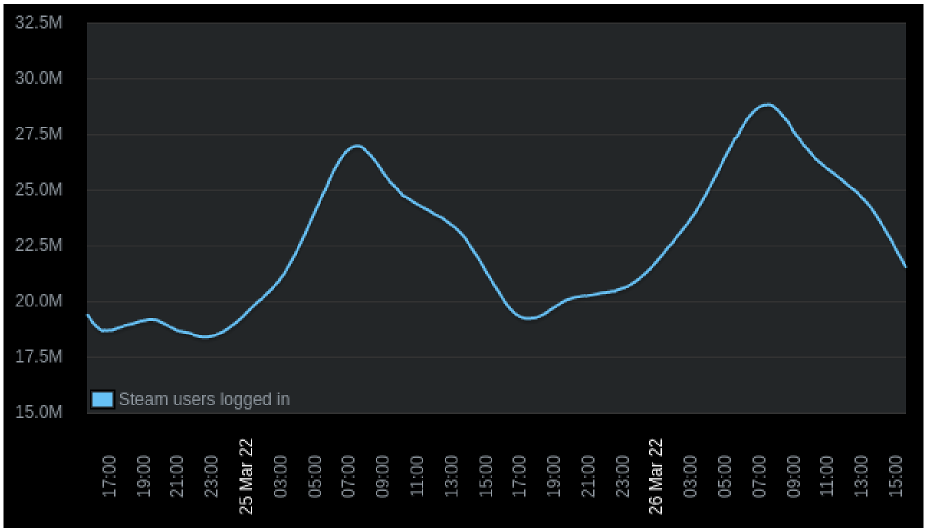

Figure 1. Steam statistics: concurrent Steam users by hour, March 25-26, 2022.

Methods for load balancing can be devised out of the area layout in games. For example, Massively Multiplayer Online Role Playing Games (MMORPG) use a technique called zoning in order to divide the game world into areas, each serviceable by a variable number of machines that can satisfy the overarching QoS requirements [19]. Some battle arenas may be large and more game resource intensive, while other peaceful areas may only need a small number of machines to adequately service its users. FPS games can employ replication, the parallelization of state update computation in heated areas where player activity is high [19]. So-called open world games, like Sea of Thieves, run separate world instances for a capped number of players. Once enough players leave an instance, the players from that instance are merged into another, more populated instance in order to save resources and encourage player interaction [21].

As evidenced, management policies to support QoS in online gaming vary depending on the structural layout of the game itself as well as the financial means of the development company, where the former can be leveraged in order to save considerable cost. Generally speaking, one player is never favoured over another in terms of resource allocation. That is, all users by themselves have equal relative importance with respect to resource distribution. What matters is locale - MMORPGs segment their world into chunks to distribute resources, while FPS games replicate the world state in order to reliably compute and distribute changes according to QoS performance demands.

Cloud gaming presents an opportunity to take much of the work from the client end and offload it to the service provider. This paradigm is obviously beneficial to the end user, as their hardware requirements will be significantly lower, in theory making the game more accessible [22]. In this way, service providers attain greater control over the quality of service in many factors, especially with regards to resource utilization by colocating multiple games on a single server [23], and the integrity of the content by reducing risk of hacking (with regards to multi-player games) or piracy (with regards to single-player games) [22]. Frame age is the time it takes for a graphics frame to be displayed on the end user's machine. In one such analysis of cloud gaming and its QoS priorities, the authors find that frame age is the most important for delivering a good user experience in cloud gaming [22]. This makes sense, as the end user is essentially receiving a video of the game they are playing. Similarly to [18], [22] finds that latency is also more impactful than packet loss in cloud gaming settings. As frame age is so important in cloud gaming, a high enough frame rate will inevitably cause traffic congestion and bandwidth issues. Methods compared in [24] show that queuing and grouping packets reduces bandwidth and leads to more predictable network behaviour and frame rate, which we have seen are the two most important performance metrics for cloud gaming.

P2P File Transfer

Peer-to-peer (P2P) applications, popularized in the late 1990s and early 2000s, broke new ground for internet architecture and usage. P2P networks and applications can be compared to the original purpose of the internet: the simple transfer of files from one end to another. The expectations for P2P file transfer tend to be milder than that of other domains such as video streaming, gaming, or real-time communication. The ideal P2P file transfer system justifies itself with two particular aspects, accuracy and relative timeliness. A user tends to expect a longer wait time depending on the size of the file being transferred. There is also little care for the inbetween progress of the file transfer, as most users will simply leave the file transfer process running in the background. As long as the wait time is not unrealistic and the final transferred file is not corrupted at any time, the user experience is generally positive.

The metrics for measuring quality of service are most applicable to the way the algorithms connect different peers in the network. There is a direct scalable relationship between the QoS performance of a P2P network, the number of peers and the implementation properties of the P2P network [25]. The paper defined the QoS as the mean time necessary to satisfy the request of each peer. The paper found the metrics that fall under the timeliness category to be an effective measurement of the performance of different P2P architectures. The researchers tested a variety of P2P architectures and placed them under different strains.

The 2004 paper [26] describes the idea of QoS routing. This idea includes the use of typical QoS metrics including, jitter, delay, and bandwidth to select a path for peers that best optimizes the network and satisfies end-user requirements.

> "QoS routing is needed to maximize the **network utilization** and improve the total **throughput** of the network." [26].

The paper first describes QoS routing when applied to IP routing algorithms, but later applies those same requirements to the routing from one peer to another peer by the P2P network routing algorithm. It identifies the difficulties that P2P has with satisfying QoS routing requirements. This includes both interconnection and interoperability of the network. The paper also identifies that security as an aspect of P2P QoS is hard to fulfill. The lack of accountability and consistent change in network peers naturally implies weak points in some systems [26].

This relation between peer selection and performance is further explored [27]. This research paper focuses on the peer selection system, and describes a possible QoS-aware service aggregation model. The model proposes to include a consistency check algorithm that surveys whether a peer interaction passes the parameters that satisfy the end-user quality of service requirements. It recommends using a dynamic peer selection tier in order to enforce this system.

> "The selection decisions are made based on the dynamic and hop-by-hop performance information such as system load, network bandwidth and delay." [27]

Similar to QoS routing mentioned before, the metrics used for performance handling are within the time-based attributes.

Despite the focus on timeliness, accuracy remains a vital attribute for P2P file transfer. Regardless of the length of time, if a transferred file is corrupted the end-user will not be satisfied. It is acceptable to have some delay from packet loss or data error during the transfer, but corrupted or fake files should be at minimal occurrence. In P2P protocols such as BitTorrent, the integrity of files must be as important if not more important than the speed of the download [28]. Another major understanding revolves around availability. For P2P to work, the network must maintain availability between peers. Interestingly, a test on BitTorrent found that there is a trade-off between availability and integrity [28]. Data integrity is easier to maintain on a more centralized peer-to-peer system. On the offhand, it relies on global components that restrict availability.

In summation, when considering metrics for measuring P2P file transfer domains, there are fewer to consider than in other domains. Despite this, it is still important that the correct metrics are considered, as the nature of P2P implies constant change in network capabilities. The main two performance attributes to consider from the hierarchy are timeliness and accuracy. These metrics are dependent and scalable to the number of peers and the algorithm properties of the P2P system. Metrics are also majorly influenced by the peer selection algorithm used by the system to create the path for retrieving data. Due to the lack of user interactions during the file transfer process, other attributes may be considered but will come secondary to timeliness and accuracy. The security attributes form the hierarchy contain the metrics which P2P has a harder time satisfying.

Video Streaming

Video streaming and video on demand (VoD) have been increasing thanks to advancements such as IPTV and LTE [29]. With this, Internet Service Providers (ISPs) have a competitive incentive to provide the best possible experience for video services. So understanding the quality of service requirements for customer satisfaction can provide a distinct advantage in ability to provide the best services to end users.

To further illustrate the importance of optimizing for quality of service metrics, it is useful to look at a 2012 study [30] that showed poor network service management results in the loss of roughly $2.16 billion per year. These huge losses can potentially be mitigated using techniques that better utilize the resources to optimize for specific applications, but for that to happen they must know what to optimize for.

It is important to recognize that QoS metrics are not always going to be sufficient to predict and control how users rate services. [31] explores the impact of various factors on Mean Opinion Score (MOS) for video streaming. In [31], MOS is compared for viewing on laptops, computers and mobile phones at 3 different video quality levels. The highest MOS is for laptops, which is followed by computers, then followed by mobile phones. Another interesting factor on MOS is seen in [31] where participants' MOS is measured at home and at a laboratory, once again at 3 quality levels. For all quality levels participants were more satisfied at home than at the laboratory. The authors speculate that this could be caused by viewers being more comfortable at home and thus have the overall experience influence their rating of the video. This illustrates that there are many factors that impact user perceptions of video service quality. However these are difficult to measure and so are largely ignored in favor of QoS metrics that can be measured and controlled by both internet service providers and video content providers.

A 2013 study [32], found that many causes of user dissatisfaction such as non-predictive buffering caused by delay, block distortions caused by packet loss, and buffer underflow caused by jitter are often sufficient to describe the quality of a video stream. From this, addressing the underlying QoS metrics that cause these sorts of technical problems is a useful way to analyze video streaming.

A 2014 study on the effects of quality of service parameters on perceived video quality [29] uses subjective tests to measure the impact of jitter, throughput, delay and packet loss on viewing different types of video content. These values are then averaged to form a cohesive view of which metrics are important.

For packet loss, as illustrated in [29], as packet loss increases, the mean user opinion score (MOS) decreases slightly at first at 1% packet loss, then stays relatively flat even up to 10% packet loss. This trend indicates that packet loss is not an especially significant metric to optimize. For jitter, as illustrated in [29], increasing jitter causes an abrupt drop in MOS, up to nearly the lowest popular score after only about 2ms of jitter. This decrease occurs regardless of the type of video being watched and thus shows that jitter is a significant component of the QoS requirements.

For throughput, as illustrated in [29], throughput has a strong positive correlation with MOS, so that when the throughput is low, the MOS is similarly low and improves as throughput increases. Notably this correlation eventually reaches a plateau where increasing throughput no longer has a significant impact on MOS. This plateau occurs around 2Mbps. This threshold is important to recognize, as spending resources to continue to improve throughput past this value will have little to no impact.

As illustrated in [29], delay is barely correlated with MOS, as even at the highest delay value of 1000ms, the MOS is only 0.25 less than the MOS at 100ms in the worst case video type, and is slightly higher than the values at 100ms for other video types. So overall delay is not an important QoS metric for video streaming.

Recapping, for video streaming

1) Jitter is very important to minimize
2) Throughput is important have relatively high
3) Packet loss is slightly important to minimize
4) Delay is not important

Discussion

After surveying various literature regarding the quality of service needs and requirements across online gaming, P2P systems, and video streaming services, it is clear that, as their service types differ, each of these domains necessarily prioritizes different technical metrics when it comes to fulfilling their quality of service requirements. Further, the literature studied exposes the policies that service providers employ in order to satisfy these metrical requirements. These policies revolve around financial constraints, resource utilization management, and fallback levels of service as a consequence of instances where the target quality of service is unreachable or violates other service policies.

In online multiplayer gaming, the end goal is to serve the player an immersive experience, in “real time”. The common usage of a client-server architecture helps to sanitize some of the interaction between players, ensuring data integrity among users and narrowing the scope of potential attacks to the service by having key information, such as the world state, be distributed from the game server. Generally, however, most security is centered around player authentication rather than data integrity. The server can only do so much for users with poor Internet connections, so it makes use of lag compensation in order to simulate a more fluid experience. This is a delicate policy, and must be carefully implemented in order to provide a fair experience for all players, regardless of connectivity strength. Timeliness is most desirable, as increased delay directly correlates to lower player performance. Jitter comes secondary with slightly weaker correlation. Packet loss matters less if it can be controlled, as it only steeply affects player performance past a certain threshold. The relative importance of these metrics is captured in Figure 2. Policies for scalable load balancing in MMOGs are critical, as users would rather not play than endure a lower quality of service. Modern cloud gaming hopes to offload some of the client-side work and solve some architectural security issues, but must still adhere to the same quality of service requirements.

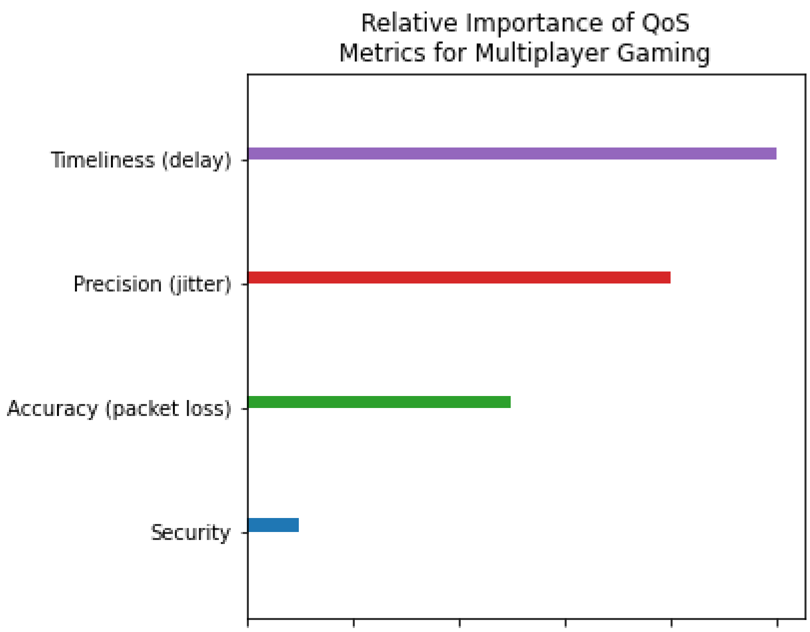

Figure 2. Relative Importance of QoS Parameters for Multiplayer Gaming

Peer-to-peer applications do not have such "real-time" requirements, but their performance is still evaluated by their accuracy and timeliness. When it comes to accuracy, file corruption must be avoided, even at the cost of delay. In this sense, accuracy is most prioritized in P2P systems. Users must be assured of the file's integrity, as well as its confidentiality, the notion that the recipient of the file is the intended receiver. We note that these security requirements are hard to fulfill in P2P due to dynamic network conditions and a lack of accountability, and are usually considered secondary to accuracy. The relative importance of these metrics, pertaining to P2P file sharing systems, is captured in Figure 3. Peer selection is a crucial mechanism in fulfilling the QoS requirements of a network, and policies for peer selection are devised based on accuracy, reliability, and fairness. Users in P2P networks have varying levels of relative importance, and the idea of fairness should be prioritized in order to balance resources and provide ample quality of service. Much like multiplayer gaming uses lag compensation to give players a connectivity baseline, helping weak peers should ensure universal availability and quality of service.

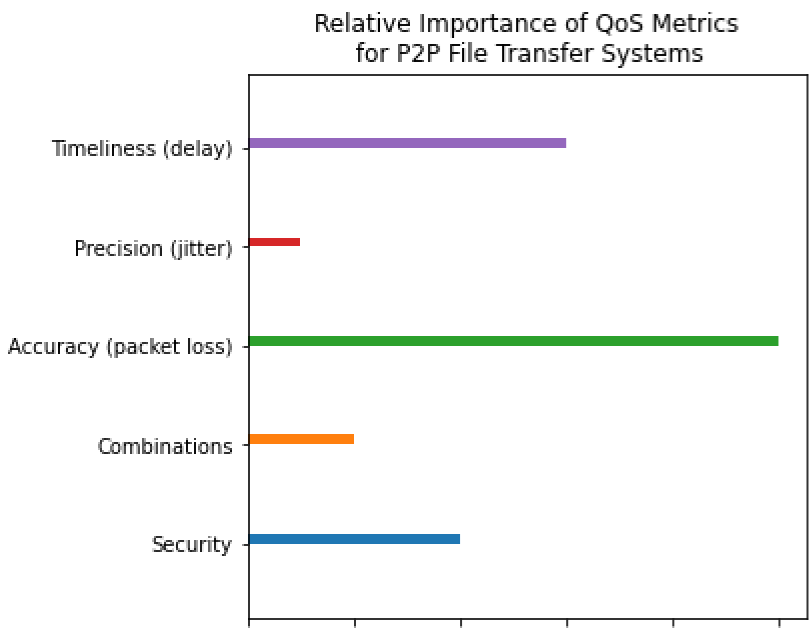

Figure 3. Relative Importance of QoS Parameters for P2P File Transfer Systems

Evaluating the importance of quality of service metrics for video on demand is often concerned with the Mean Opinion Score (MOS) of individuals when related to user experience. Minimal packet loss seems to have a negligibly negative effect on MOS. Jitter leads to a significant drop in MOS. Increasing throughput is more linearly correlated with increasing MOS, but faces severely diminishing returns after a certain threshold. Therefore high precision is chiefly essential, as well as maximizing throughput to a certain threshold in order to save resources. Overall delay is not as important. The relative importance of these metrics with respect to video streaming quality of service is captured in Figure 4.

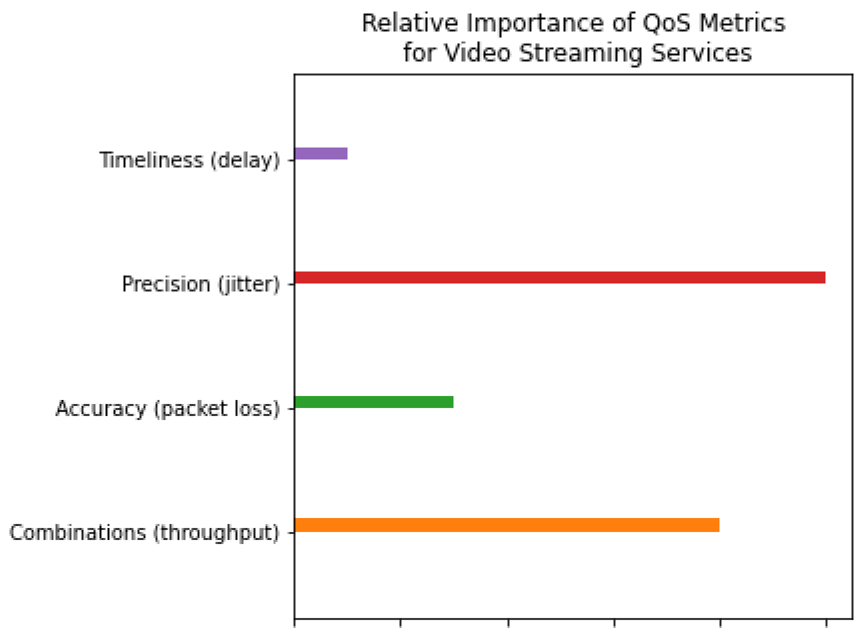

Figure 4. Relative Importance of QoS Parameters for Video Streaming Services

Conclusions

As we have shown, quality of service requirements vary across each of the surveyed networking domains. Developers of P2P systems need not consider factors important for multiplayer gaming and video streaming like jitter, and transmission accuracy generally takes precedence over transmission security.

Throughout this study we noticed that security is often of little or no consideration in each of these domains. The application domains we surveyed do not have security as a standard, as opposed to financial and medical applications, where validating this requirement is mandatory. We believe users should demand more from their service providers in the way of tightened security with respect to the confidentiality, integrity, and availability of communication transactions, as well as the ability to simultaneously provide a “good enough” quality of service to its end users. Users should be treated equally, adaptive infrastructure should be built in order to alleviate strain on service providers as well as hardware requirements for end users.

To this end, futures involving cloud computing are already coming into fruition. However, it is important to realize that, while these distributed systems promise to save bandwidth and maximize resource utilization, the quality of service policies must adapt around the quantitative metrical requirements of their respective networking domain. These requirements will change as the user experience demands evolve, but we have also shown that, for the most part, the requirements themselves do not change, but the scale simply grows. Therefore, more careful attention must be made to require quality of service policies to strictly adhere to their consumer promises.

As these policies evolve, it is important to continuously survey the state of quality of service across many different networking domains in order to maintain a clear picture of what quality of service means to each domain. We hope to see future studies replicate our survey, adding additional networking domains and evaluating the prioritization of the metrics and allocation of policies given in the taxonomy accordingly.

References

[1] Mart´ın Varela, Lea Skorin-Kapov, and Touradj Ebrahimi, "Quality of service versus quality of experience," Quality of Experience, pp. 85–96, Mar 2014.

[2] Markus Fiedler, Tobias Hossfeld, and Phuoc Tran-Gia, "A generic quantitative relationship between quality of experience and quality of service," IEEE Network, vol. 24, no. 2, pp. 36–41, 2010.

[3] X. Xiao, Technical, Commercial and Regulatory Challenges of QoS: An Internet Service Model Perspective. 2008.

[4] Rogerio Le ´ ao Santos de Oliveira, Christiane Marie ˜ Schweitzer, Ailton Akira Shinoda, and Ligia Rodrigues Prete, "Using mininet for emulation and prototyping software-defined networks," in 2014 IEEE Colombian Conference on Communications and Computing (COLCOM), 2014, pp. 1–6.

[5] Akhilesh Sharma and Aakanksha Sharma, "Qos parameter analysis of tcp and udp traffic over open flow enabled software defined network," in Proceedings of International Conference on Data Science and Applications, Mukesh Saraswat, Sarbani Roy, Chandreyee Chowdhury, and Amir H. Gandomi, Eds., Singapore, 2022, pp. 13–25, Springer Singapore.

[6] Pinkey Chauhan and Mithilesh Atulkar, "Achieving enhanced network performance in udp and tcp traffic of software defined networking by selecting java based controllers decisively," International Journal of Innovative Technology and Exploring Engineering, vol. 9, no. 7, pp. 268–274, May 2020.

[7] SM Shamim, Shahadat Shisir, Ahosanul Hasan, Mehedi Hasan, and Arafat Hossain, "Performance analysis of different open flow based controller over software defined networking," Global Journal of Computer Science and Technology, 2018.

[8] Majed Althobyani and Xingang Wang, “Implementing an sdn based learning switch to measure and evaluate udp traffic,” Computers & Electrical Engineering, vol. 66, pp. 342–352, 2018.

[9] Himanshi Babbar, S. Parthiban, G. Radhakrishnan, and Shalli Rani, “A genetic load balancing algorithm to improve the qos metrics for software defined networking for multimedia applications,” Multimedia Tools and Applications, 2022.

[10] Zhibin Liu, Ziang Liu, Yuanyuan Huang, Xin Liu, Xiaokang Zhou, and Rui Zhou, “A research of distributed security and qos testing framework,” in 2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech), 2020, pp. 174–181.

[11] B. Sabata, S. Chatterjee, M. Davis, J.J. Sydir, and T.F. Lawrence, “Taxonomy for qos specifications,” in Proceedings Third International Workshop on ObjectOriented Real-Time Dependable Systems, 1997, pp. 100–107.

[12] E. Adams and J. Dormans, Game mechanics: Advanced game design. Berkeley, CA: New Riders, 2012.

[13] M. Bredel and M. Fidler, "A measurement study regarding quality of service and its impact on multiplayer online games," 2010 9th Annual Workshop on Network and Systems Support for Games, 2010, pp. 1-6, doi: 10.1109/NETGAMES.2010.5679537.

[14] S. W. Lee and R. K. Chang, “Enhancing the experience of multiplayer shooter games via advanced lag compensation,” Proceedings of the 9th ACM Multimedia Systems Conference, 2018.

[15] G. Armitage, “An experimental estimation of latency sensitivity in multiplayer Quake 3,” in Proc. of IEEE ICON, Sept. 2003.

[16] T. Henderson, “Latency and user behaviour on a multiplayer game server,” in Proc of NGC, Nov. 2001

[17] T. Henderson and S. Bhatti, “Networked games: a QoS-sensitive application for QoS-insensitive users?” in Proc. of RIPQoS, Aug. 2003.

[18] A. F. Wattimena, “Performance modeling of interactive gaming,” Vrije Universiteit Amsterdam, Tech. Rep., May 2006.

[19] V. Nae, R. Prodan and T. Fahringer, "Cost-efficient hosting and load balancing of Massively Multiplayer Online Games," 2010 11th IEEE/ACM International Conference on Grid Computing, 2010, pp. 9-16, doi: 10.1109/GRID.2010.5697956.

[20] Valve Corporation, “Steam: Game and player statistics,” Welcome to Steam, 2022. [Online]. Available: https://store.steampowered.com/stats. [Accessed: 26-Mar-2022]. 10.1109/LCOMM.2011.080811.111160.

[21] Rare, “Sea of Thieves.” [Online]. Available: https://www.seaofthieves.com/

[22] S. Flinck Lindström, M. Wetterberg and N. Carlsson, "Cloud Gaming: A QoE Study of Fast-paced Single-player and Multiplayer Gaming," 2020 IEEE/ACM 13th International Conference on Utility and Cloud Computing (UCC), 2020, pp. 34-45, doi: 10.1109/UCC48980.2020.00023.

[23] Y. Li et al., "Towards Minimizing Resource Usage With QoS Guarantee in Cloud Gaming," in IEEE Transactions on Parallel and Distributed Systems, vol. 32, no. 2, pp. 426-440, 1 Feb. 2021, doi: 10.1109/TPDS.2020.3024068.

[24] J. Saldana, J. Fernandez-Navajas, J. Ruiz-Mas, J. I. Aznar, L. Casadesus and E. Viruete, "Comparative of Multiplexing Policies for Online Gaming in Terms of QoS Parameters," in IEEE Communications Letters, vol. 15, no. 10, pp. 1132-1135, October 2011, doi:

[25] F. Gianfelici, "Measurement of quality of service (QoS) for peer-to-peer networks," IEEE Symposium on Virtual Environments, Human-Computer Interfaces and Measurement Systems, 2005., 2005, pp. 6 pp.-, doi: 10.1109/VECIMS.2005.1567580.

[26] Linnolahti, Janne. "QoS routing for P2P networking." HUT T-110.551 Seminar on Internetworking. 2004.

[27] Xiaohui Gu and K. Nahrstedt, "A scalable QoS-aware service aggregation model for peer-to-peer computing grids," Proceedings 11th IEEE International Symposium on High Performance Distributed Computing, 2002, pp. 73-82, doi: 10.1109/HPDC.2002.1029905.

[28] Pouwelse, J., Garbacki, P., Epema, D., Sips, H. (2005). The Bittorrent P2P File-Sharing System: Measurements and Analysis. In: Castro, M., van Renesse, R. (eds) Peer-to-Peer Systems IV. IPTPS 2005. Lecture Notes in Computer Science, vol 3640. Springer, Berlin, Heidelberg. https://doi.org/10.1007/11558989_19

[29] P. Paudyal, F. Battisti and M. Carli, "A study on the effects of quality of service parameters on perceived video quality," 2014 5th European Workshop on Visual Information Processing (EUVIP), 2014, pp. 1-6, doi: 10.1109/EUVIP.2014.7018383.

[30] Duraimurugan, S., Avudaiammal, R. & Vincent, P.M.D.R. Enhanced QoS Through Optimized Architecture for Video Streaming Applications in Heterogeneous Networks. Wireless Pers Commun 118, 1655–1673 (2021).

https://doi-org.ezproxy.library.uvic.ca/10.1007/s11277-021-08109-8

[31] H. Gahbiche Msakni and H. Youssef, "Is QoE estimation based on QoS parameters sufficient for video quality assessment?," 2013 9th International Wireless Communications and Mobile Computing Conference (IWCMC), 2013, pp. 538-544, doi: 10.1109/IWCMC.2013.6583615.

[32] M. Li, “Qoe-based performance evaluation for Adaptive Media Playout Systems,” Advances in Multimedia, vol. 2013, pp. 1–7, 2013.

Appendix

Contributions by Oscar Sandford:

- Set up project structure on GitHub, formatted report template, project website
- Written and editorial contributions to project proposal documents
- Old project: set up simulation environment, driver code
- Report: overall editing, leading contributions to introduction, related works, approach, multiplayer gaming, discussion, and conclusion sections
- Presentation: set up slides, composed slide content/script, presented one third of content

Contributions by Juan Flores:

- Written and editorial contributions to project proposal documents
- Wrote and posted weekly update reports on project website
- Report: edited citations, contributions to introduction, peer to peer networks and discussion sections.
- Presentation: composed slide content/script, presented one third of content
- Recorded and edited video presentation for project

Contributions by Ben Wunderlich:

- Planning and editorial contributions to project proposal documents
- Old project: helped debug driver code
- Report: contributions to introduction, approach, video streaming sections
- Presentation: composed slide content/script, presented one third of content