Dissecting the Applicability of HTTP/3 in Content Delivery Networks

Mengying Zhou[1], Yang Chen[1], Shihan Lin[1], Xin Wang[1], Bingyang Liu[2], Aaron Yi Ding[3]
[1]Shanghai Key Lab of Intelligent Information Processing, School of Computer Science, Fudan University, China
[2]Huawei Technologies Co. Ltd., China
[3]Department of Engineering Systems and Services, Delft University of Technology, The Netherlands
{myzhou19,chenyang,shlin15,xinw}@fudan.edu.cn, liubingyang@huawei.com, aaron.ding@tudelft.nl

2024 IEEE 44th International Conference on Distributed Computing Systems (ICDCS) | 979-8-3503-8605-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICDCS60910.2024.00091

***Abstract*—HTTP/3 (H3) has experienced significant growth and extensive adoption in various scenarios, especially in Content Delivery Networks (CDNs). Over the past few years, there have been numerous insightful studies on its deployment in industrial CDNs. However, these studies often separately analyze H3 and CDN, overlooking their synergistic integration. In this work, we explore the applicability of H3 in CDN from a holistic perspective. We analyze 325 websites hosted by seven CDN providers and identify three key characteristics where CDN align perfectly with H3's strengths. Firstly, CDN resources dominate the composition of webpages, where enabling H3 can amplify H3's benefits in connection acceleration. Secondly, CDN providers also exhibit a dominant characteristic, with the majority of CDN resources hosted by a few large providers. This phenomenon makes different webpages share the same provider. When browsing consecutively, H3 helps to skip the connection phase by resuming the connections to the same CDN provider across pages. Thirdly, H3 mitigates the congestion problem on webpages serving multiple CDN resources. This work provides a deeper insight into the applicability of H3 in large-scale distributed systems like CDNs, holding promise for informing the development and optimization of industrial H3.**

***Index Terms*—HTTP/3, Content Delivery Network, Measurement, Web Performance**

I. Introduction

HTTP/3 (H3) [1] is the latest HTTP version developed based on the QUIC transport layer protocol [2]. Compared with the previous TCP-based HTTP/2 (H2) and HTTP/1.1 (H1.1), H3 exhibits several notable advantages, such as fast connection, stream multiplexing, better adaptation to network conditions, and improved security [3]–[5]. The superiority of H3 has been widely documented in various scenarios to highlight its ability to reduce latency [6], [7], improve throughput [8], [9], and provide better resilience [4], [10] in emulation and production environments [11], [12].

CDN, as a main driver for H3, has adopted H3 technology early and widely [13], [14]. Following Google's pioneering H3 adoption, Akamai, Cloudflare, and many other mainstream CDN service providers have also released H3 support, yielding insightful reports on performance and deployment. However, these studies analyze H3 and CDN ***separately***, neglecting to deeply integrate the characteristics of CDN and H3 to investigate their ***synergy***. The performance optimization by adopting H3 derives not only from H3's strengths but also owing to specific CDN characteristics. To fill this gap, our research aims to unveil how H3 features and CDN characteristics jointly accelerate the performance, examining the synergistic effects of their integration.

Our motivations. We propose the following questions to explore the synergistic effects of H3 and CDN integration:

1) What are the characteristics of CDN usage on webpages, and what is the potential coherence with H3?
2) How do these CDN characteristics collaborate with H3's strengths (e.g., fast connection and stream multiplexing) to contribute to better content delivery?
3) What implications and insights can our findings bring to CDN providers, end users, web developers, and researchers?

Our contributions. By analyzing 325 websites in the Alexa Top list [15], we investigate the synergy between H3 and CDN and extract the root of its effectiveness: H3 streamlines repetitive processes in multiple requests to large-scale distributed systems, with CDN being a typical one.

Our measurements reveal that CDN resources constitute the majority of webpage content. This phenomenon amplifies the connection acceleration brought by H3, even at a small-scale adoption of H3. Additionally, we observe that certain CDN providers exhibit a strong dominance, with most CDN resources hosted by just a few giant providers. This dominance leads to a phenomenon where different webpages share the same providers. This "shared-provider" phenomenon can optimize web loading by resuming connections to the same CDN provider across different pages in consecutive visits. Moreover, the dominance of giant CDN providers further results in centralizing CDN resources on a few providers, posing risks of congestion problems. Leveraging H3's stream multiplexing can improve transmission efficiency in such cases. Finally, based on these findings, we summarize a series of suggestions for maximizing the benefits of H3 adoption. The key contributions of this work are summarized below:

1) By analyzing 325 websites from Alexa Top list, we

This work is supported by the National Key R&D Program of China under Grant No. 2022YFB3102901, National Natural Science Foundation of China (No. 62072115, No. 61971145, No. 61831018, No. U21A20452) and European Union's Horizon 2020 Research and Innovation programme (No. 101021808). Yang Chen is the Corresponding Author.

2575-8411/24/$31.00 ©2024 IEEE
DOI 10.1109/ICDCS60910.2024.00091

identify three characteristics of CDN that collaborate with H3's strengths: a high proportion of CDN resources in webpage content, a preference for giant providers, and centralizing a large volume of content on a few providers.
2) We reveal that the synergy between H3 and CDN arises from H3's capability to eliminate repetitive processes in multiple requests to large-scale distributed services, demonstrating H3's applicability in CDNs.
3) Based on our findings, we put forward several constructive implications for CDN providers, end users, web developers, and researchers.

II. Background and Motivation

This section first introduces the highlighted features of H3 (Section II-A) and its deployment trajectory in CDNs (Section II-B). We then discuss the contributions and limitations of previous research on H3-enabled CDNs (Section II-C). Notably, the investigation of H3 in CDNs is still in its early stages, with previous studies offering only a brief overview of its performance and deployment scale. Based on this, we present the motivation of this work, aiming to delve deeper into the synergy between H3 and CDNs (Section II-D).

A. Two Highlighted Features of H3

H3 offers several notable features. The following two are most extensively discussed in previous studies:

Fast connection establishment: H3 accelerates the connection process compared with the widely used H2 + TLS/1.2 protocol suite. H3 reduces the handshake latency from three round-trip times (RTTs) to just one RTT by adopting the latest TLS/1.3 [16] and merging its QUIC transport layer handshake into the TLS handshake [2]. Furthermore, the connection resumption mechanism in H3 allows clients to transmit data directly without the handshake process by verifying pre-shared keys [16] stored from previous connections.

Stream multiplexing: H3 addresses the Head-of-Line (HoL) blocking problem [17] in TCP-based HTTP by introducing stream multiplexing. The HoL blocking problem refers to the blocking of packets in a queue because preceding packets are lost, even though they are not logically related. This happens because TCP processes packets strictly in order. H3 uses multiple independent streams, preventing interference between each other to resolve HoL blocking.

B. H3 Adoption in Mainstream CDNs

The traffic of H3 has reached 29.5% of all the websites since May 2024[1]. With the increasing acknowledgement of H3 in the industry and the growing desire for H3 support among users, numerous CDN providers have released the availability of H3 support. This allows their customers to configure page content for access through an H3-enabled CDN. The H3 adoptions in various providers are summarized in Table I.

Cloudflare was the earliest CDN service to provide H3 support in 2019 [18], [19]. As for Google, the pioneer driving force behind the promotion of H3, officially announced H3 support for its CDN service in 2021 [20]. However, researchers observed the presence of H3-enabled CDN in various Google-related web services during previous wild measurements [14], [21], [22]. This was because Google conducted experimental deployments in many of its applications very early [22], which resulted in the highest H3 deployment ratio in its CDN services [21]. Fastly was also among the early adopters of H3, making it publicly available as early as 2021 [23]. Furthermore, the LiteSpeed team founded QUIC.Cloud [24], specializing in providing H3-driven CDN services. Following this trend, many CDN service providers, such as Amazon CloudFront [25] and Meta [26], offered options to enable H3 functionality. Akamai, another service provider actively embracing H3, announced that H3 has become the default CDN configuration from 2023 [27].

[1]https://w3techs.com/technologies/details/ce-http3

TABLE I: Release year of H3 support in various CDNs and their corresponding performance reports

Provider	Release Year	Performance Report
Cloudflare	2019 [18], [19]	H3 performs 12.4% better in TTFB, but 1-4% worse in PLT than H2 [28].
Google Cloud CDN	2021 [20]	Reduce rearch latency by 2%, video rebuffer times by 9%, and improves mobile device throughput by 7% [11].
Fastly	2021 [23]	QUIC can represent an 8% increase in throughput [9].
QUIC.Cloud	2021 [24]	H3 turns TTFB from 231ms to 24ms [29].
Amazon CloudFront	2022 [25]	N/A
Meta	2022 [26]	H3 reduces tail latency by 20% and MTBR by 22% [12].
Akamai	2023 [27]	6.5% enhancement in users with TAT under 25ms; 12.7% improvement for requests exceeding 1 Mbps [5].

C. Previous Research on CDN over H3

With the widespread adoption of H3 in mainstream CDNs, both CDN service providers and researchers have measured its performance and deployment scale.

Performance Report. Several CDN service providers offered specialized reports on the performance after deploying H3, providing valuable insights about H3 adoption in production environments. According to Akamai's report [5], fast connection setup was a key factor in enhancing CDN performance, resulting in a 6.5% improvement in the percentage of users achieving a turnaround time (TAT) of less than 25ms. Furthermore, Akamai demonstrated that H3 contributes to improved throughput. The proportion of requests meeting a threshold of more than 1 Mbps showed a 12.7% improvement. Fastly also validated that QUIC presented an 8% increase in throughput [9]. The optimized congestion control algorithm and recovery mechanisms employed in QUIC assisted Meta in reducing tail latency by 20% and mean-time-between-rebuffering (MTBR) by 22% [12]. Meanwhile, a study from

Yu and Benson [4] measured the performance of production QUIC of Google, Cloudflare, and Meta. This study delved into variations in QUIC's performance in production environments due to different congestion control implementations, a similar observation also highlighted by Cloudflare [28].

Deployment Scale. In addition to performance, the scale of deployment is also a crucial aspect of the investigation. Some studies have provided valuable measurement studies on the deployment scale of H3 at different development stages. Trevisan et al. [22] pointed to CDN services as a driver for increasing H3 deployment, particularly due to the widespread deployment of Google CDN on most webpages. Targeting the RFC 9000 standardization, Zirngibl et al. [14] also emphasized that the dominance of CDNs promotes H3 deployment. Saverimoutou et al. [21] conducted the first systematic study focusing on the H3-enabled CDN deployment range and performance. They considered the impact of CDN's characteristics and examined First and Repeat modes. However, they still did not delve into other CDN characteristics and the compatibility between H3 and CDN features. For instance, the proportion of CDN resources on webpages, the sharing of CDN providers across pages, and the higher quantity of multiple CDN resources on a single page, as discussed in this paper, are all important and influential CDN characteristics in H3-enabled environments.

D. Motivation of This Work

The above findings from existing studies merely present the well-known and understood facts about H3 in CDNs. These studies analyze CDN and H3 separately, lacking a holistic consideration of the synergistic effects by deeply integrating the characteristics of CDN and H3. According to previous studies, adopting H3 can yield observable performance improvements, such as optimized connection times and page load times. However, these enhancements are not solely attributed to H3's outstanding features but also owing to certain CDN characteristics.

It is essential to explore how specific features of H3 and CDN characteristics contribute to this acceleration, and how their integration generates a synergistic effect to promote such "1+1>2" outcome. Given the clearly describing H3's features and advantages, our research aims to delve into three main aspects. Firstly, uncovering the inherent characteristics of CDN services (Question 1 in Section I). Secondly, analyzing the synergistic collaboration between H3's features and these characteristics of CDN services (Question 2). Lastly, the profound understanding gained from the above two aspects enables us to provide clear explanations for observed phenomena and practical suggestions for effectively implementing H3 in CDNs (Question 3). By answering these questions, we aim to conclude the applicability of H3 in CDNs and explore potential optimization strategies for H3-enabled CDNs.

III. Data Collection

This section presents the criteria for selecting the target webpages (Section III-A), the details of the probe setup (Section III-B), the metrics used to evaluate H3 performance in CDNs (Section III-C), and ethical consideration (Section III-D).

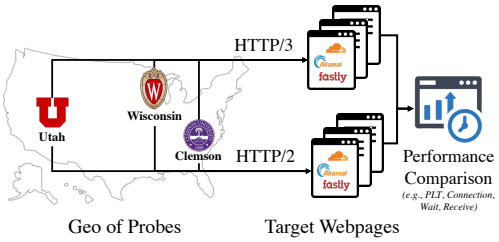


Fig. 1: The illustration of measurement

A. Webpage Selection

We create the target webpage list based on the Alexa Top 500 [15] websites. We exclude websites that are inaccessible via H3 during measurements due to incomplete H3 adoption. After this exclusion, the final list contains 325 websites. We select their landing pages as the measurement webpages.

We acknowledge certain limitations in our webpage selection, as the chosen websites may not comprehensively cover various popular lists (e.g., Tranco, Cisco Umbrella, Trexa Top) and pages with different structures (e.g., landing page vs. internal pages [30]). Nevertheless, we believe that our website selection is still representative. Alexa Top list is widely used and remains influential. Moreover, since our focus is on H3-enabled webpages and popular websites that exhibit a more active adoption of H3, it is more suitable to perform measurements on the Alexa Top list generated according to daily visitors and pageviews [31]. While different webpage selection criteria may introduce some variation in results, our findings on the H3's applicability in CDNs are still helpful and informative. Other factors are worthy of investigation, and we plan to explore these aspects in future research.

B. Probe and Collection Setup

To avoid observation bias introduced by a single vantage point, our measurements employ geographically distributed probes. We conduct distributed measurements at three vantage points on the CloudLab[2] platform, as illustrated in Fig. 1. CloudLab is a research testbed for cloud computing and networking research, offering configurable environments and various networking components. The three vantage points are hosted by the University of Utah, the University of Wisconsin-Madison, and Clemson University. Each vantage point deploys three probes, each equipped with 8 CPU cores, 128GB of memory, and running the Ubuntu 20.04 operating system.

For each target webpage, each probe uses a Chrome browser (version 108.0.5359.61) to access it with H3 and H2, respectively. We enable H3 access by activating the `enable-quic` option in the Chrome browser. To prevent potential interference between the two protocols, we use separate instances of Chrome for each protocol with different user data directories

[2]https://www.cloudlab.us

TABLE II: Number of requests and the percentage of total requests using different HTTP versions

Protocol	CDN		Non CDN		All	
	# Req	%	# Req	%	# Req	%
HTTP/2	14870	41.2	7215	20.0	22085	61.2
HTTP/3	9280	25.8	2462	6.8	11742	32.6
Others	3	0.01	2227	6.2	2230	6.2
All	24153	67.0	11904	33.0	36057	100

specified using the `user-data-dir` parameter. To ensure CDN resources are served from the edge CDN server rather than fetched from the origin server, we visit each webpage twice for every measurement. The first visit triggers the caching of CDN resources at the edge CDN server, and the second visit's performance is taken as the measurement result. In practice, our results show that there is no significant difference between the two visits. This is because our selected webpages are popular and frequently visited, resulting in their long-term presence in edge CDN servers. After each page visit, all connections are terminated, and all caches are cleared to ensure no potential influence between different page visits. Each probe sequentially visits the target webpages in a fixed order. The entire measurement process spans six days, from October 10, 2022 to October 15, 2022.

We analyze the performance of each visited webpage by collecting Chrome-HAR file[3]. The Chrome-HAR files contain detailed information about each resource file loaded on the page and performance metrics. To differentiate the CDN resources on webpages, we utilize LocEdge [32], an open-source tool capable of identifying CDN resources within webpages and determining the CDN service provider hosting these resources.

C. Evaluation Metrics

In this study, we analyze four standard web performance metrics to represent a typical web browsing scenario involving multiple CDN resources. One metric is at the page level, while the other three are at the file level. For page-level web performance, we adopt the commonly used Page Load Time (PLT) [33] metric. PLT represents the duration when all web resources (e.g., HTML, images, fonts, CSS) and any sub-resources to complete the loading. PLT is defined as the period from the start of the page load to the trigger of `onLoad` event[4]. Compared with visual metrics like SpeedIndex and First Contentful Paint, PLT is a more suitable evaluation metric because measuring CDN performance requires fully loading all web resources [34]. SpeedIndex and First Contentful Paint measure the time it takes to visually load a small portion of content, which may not necessarily involve CDN transmission. For file-level web performance, we consider Connection time, Wait time, and Receive time based on Cloudflare's experience[5]. Connection time corresponds to the handshake period. Wait time refers to the time from sending the first byte to receiving the first byte of the response. Receive time represents the duration of response data transmission.

[3]https://github.com/cyrus-and/chrome-har-capturer

[4]https://developer.mozilla.org/en-US/docs/Glossary/Page_load_time

[5]https://blog.cloudflare.com/a-question-of-timing

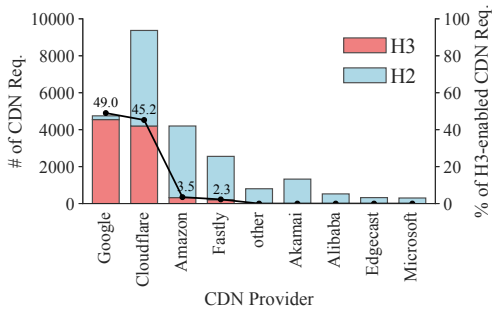


Fig. 2: H3 adoption by different CDN service providers and their market share

In this paper, we focus on the reduction benefits that H3-enabled CDNs bring to webpages compared with H2-based CDNs. To present the results more intuitively, we introduce a processed metric called $X_{reduction}$ in the subsequent experiments, calculated with $X_{H2}-X_{H3}$. Here, X can refer to any of the four metrics mentioned above. For instance, $PLT_{reduction}$ represents $PLT_{H2} - PLT_{H3}$. A positive value of $X_{reduction}$ indicates that H3 performs better, while a negative value suggests that H2 performs better.

D. Ethical Considerations

Our research involves the collection of data related to CDN traffic. In this work, we only collect data from CDN resources that are publicly accessible on webpages. We do not engage in the collection or analysis of real user traffic. The data we collect is generated through controlled automated machines and browsers. Furthermore, the average traffic to each nearby CDN server is 126.7 Kbps. This volume is negligible compared with the CDN petabyte scale traffic [35].

IV. Analysis on H3 Adoption

Before delving into the collaboration between H3 and CDN characteristics, we present an overview of the adoption status of H3 within webpage and CDN providers.

A. Adoption Rate in Webpages

Table II presents the number of requests and the percentage of total requests using different HTTP versions. The selected websites accumulate a total of 36,057 requests. In our dataset, 67.0% of requests (24,153) originate from CDN services, revealing the prevalence of CDN services in current webpage resources. The remaining 33.0% (11,904) are non-CDN resource requests provided by web services.

The usage rate of H3 in all requests stands at 32.6%. This adoption rate of H3 has notably increased compared with previous studies [14], [22]. Among the H3 requests, 78.8% are CDN requests, indicating that currently CDN services are still the primary driver for H3 deployment [13], [14]. In particular, some websites, such as youtube.com and wordpress.com, fully

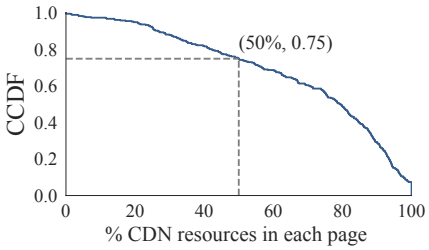
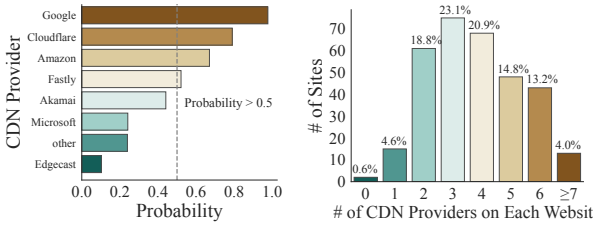


Fig. 3: The complementary cumulative distribution function (CCDF) of the percentage of CDN resources on each webpage



(a) The probability of different CDN providers appearing on webpages

(b) The number and percentage of webpages with different numbers of providers

Fig. 4: Shared giant providers across different webpages

support access using H3. A similar characteristic of these websites is their heavy reliance on various static resources hosted by CDN. H2 requests lead with the highest proportion at 61.2%. While other HTTP versions (including HTTP/1.1, HTTP/1.0, and HTTP/0.9) are rarely used, especially in CDN requests, where their contribution is less than 0.01%.

B. Adoption in CDN Providers

In Fig. 2, we illustrate the status of H3 adoption by different CDN service providers and their market share among our selected websites. Consistent with prior findings [21], [22], Google remains a key promoter in advancing H3 deployment, contributing to nearly 50% of all H3-enabled CDN resources. Additionally, Google's CDN services have almost entirely shifted towards H3 access. Meanwhile, Cloudflare serves 45.2% of H3-enabled CDN requests. Notably, its proportions of H3 and H2 are comparable, indicating the rapid deployment of H3 at Cloudflare. In contrast, Amazon, Fastly and remaining CDN providers offer limited support for H3, with their CDN services still primarily relying on H2.

V. Characteristics of CDN usage

In this section, we will answer the first research question: what are the characteristics of CDN usage on webpages, and what is the potential coherence with H3? We present three specific phenomena characterizing CDN services and discuss the potential benefits of adopting H3 within these phenomena.

A. Characteristic #1: CDN Dominates Webpage Content

CDN resource proportion in webpage content is an important factor influencing page load time. In Fig. 3, we depict the Complementary Cumulative Distribution Function (CCDF) of the percentage of CDN resources on each webpage. We observe that 75% of webpages have exceeded 50% CDN resources. This indicates that the majority of webpages are dominantly composed of CDN resources. This high proportion of CDN resources could amplify H3's strengths in fast connection.

B. Characteristic #2: Giant Providers Are Shared Across Different Webpages

Giant CDN providers like Akamai, Cloudflare, and Google are preferred choices for website developers when configuring CDN resources due to their widespread deployment and ability to deliver high-quality and reliable services. In Fig. 4(a), we show the probability distribution of various CDN providers appearing on webpages. The giant CDN providers are almost used on every webpage, with the probability of the top four CDN providers appearing exceeding 50%. This widespread adoption highlights the dominance of these giant providers in the CDN market.

Given the presence of giant CDN providers on almost every webpage, a phenomenon of different pages sharing the same providers has emerged. For example, both spotify.com and zoom.us have CDN resources from Amazon, Cloudflare, and Google, which means they share the usage of these three providers. Therefore, in Fig. 4(b), we investigate the number and percentage of webpages using different numbers of providers. Our results show that the vast majority of webpages (94.8%) use at least two CDN providers, indicating the prevalence of the shared-provider phenomenon. The shared-provider phenomenon not only underlines the popularity and dominance of giant CDN providers, but also presents an opportunity to explore the potential effects associated with the same providers across different pages when leveraging H3's connection resumption functionality.

C. Characteristic #3: A Large Volume of CDN Resources are Centralized on Certain CDN providers

The number of CDN resources on the page has been steadily increasing, along with the growing complication of webpages [36]. Meanwhile, a preference for giant CDN providers makes CDN resources centralized on a few CDN providers. Consequently, the large volume of CDN requests to these providers could trigger severe congestion problems for TCP-based HTTP connections.

The congestion problem primarily arises from TCP's HoL blocking, where the previous lost packets delay subsequent ones at the receiver, even if subsequent packets may arrive earlier and have no logical dependency on the lost packets. This problem is particularly severe on complicated webpages with a large number of content. Once a preceding packet is lost, the delay impact on subsequent data will accumulate as the page loads.

In Fig. 5, we analyze the number of CDN resources of

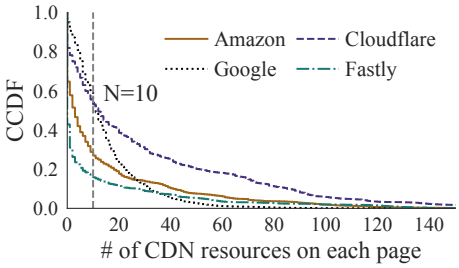
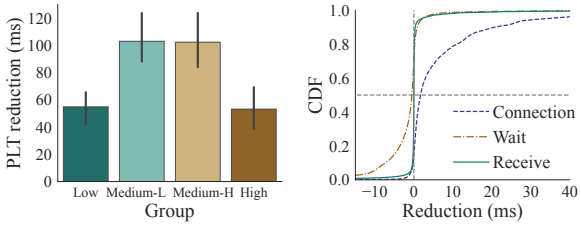


Fig. 5: The CCDF of the number of CDN resources on each webpage hosted by Amazon, Cloudflare, Google, and Fastly



(a) The PLT reduction for four groups

(b) The CDF of reduction of connection, wait, and receive

Fig. 6: The reduction of PLT, connection, wait, and receive

each webpage from four giant providers: Amazon, Cloudflare, Google, and Fastly. Our results display that each CDN provider hosts a number of CDN resources on a webpage. For the webpages using Cloudflare and Google, approximately 50% of them contain more than 10 CDN resources. This result indicates that they are highly likely to suffer from the congestion problem if they rely solely on TCP-based H2 for CDN resource delivery. H3 can mitigate congestion by utilizing stream multiplexing in its transport layer protocol (i.e. QUIC [2]), allowing logically unrelated data to be processed independently without mutual interference.

Takeaway 1: CDN services exhibit three characteristics, specifically manifested as 1) a dominant fraction of CDN resources in webpage content, 2) shared giant providers across different pages, and 3) CDN resources centralized on a few providers. Gratefully, each characteristic can be highly beneficial when adopting H3 in CDNs.

VI. H3 in Large-Scale CDNs

The characteristics explored in the previous section inspire the second research question: how do these CDN characteristics collaborate with H3's strengths to contribute to better content delivery? In this section, we first present the potential benefits of adopting H3 in CDNs. Then, we validate these opportunities and demonstrate the applicability of H3 in CDNs.

A. Potential Benefits of Adopting H3 in CDNs

In Section II, we introduce the two most discussed features of H3. Combining these two H3 features and the three CDN characteristics listed in Section V, we present the following three potential benefits when adopting H3 in CDNs.

1) As the primary constituents of webpage content, CDN resources can amplify the advantage of H3's fast connection when enabling H3 access.
2) The shared providers can promote the number of H3's resumed connections in consecutive web browsing.
3) Multiplexing streams can accelerate transmission, especially when serving a large volume of CDN resources.

B. Dominant Proportion of CDN Resources Amplifies the Benefit of H3's Fast Connection

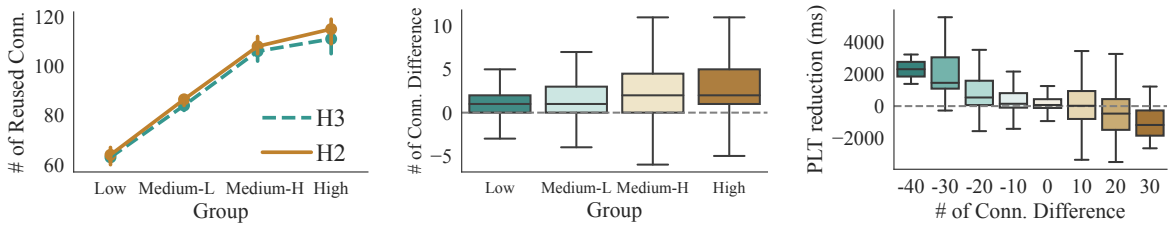
H3-enabled CDN resources reduce PLT, even at small-scale adoption. Considering the dominant proportion of CDN resources in webpage content shown in Section V-A, adopting H3 in CDN presents an opportunity: even small optimization brought by H3 will be amplified due to the large quantity of CDN resources.

We validate this opportunity by analyzing the PLT reduction of webpages. Webpages are categorized into four groups based on quartiles of the number of H3-enabled CDN resources, namely Low, Medium-Low, Medium-High, and High. Each group has an equal number of pages.

In Fig. 6(a), we plot PLT reduction for four webpage groups. Notably, all groups exhibit a positive PLT reduction, indicating that adopting H3 in CDNs will benefit all levels of webpages. Even the Low group, with limited H3 adoption, achieves a reduction of nearly 60ms. The Medium group experiences the highest reduction. However, we observe an unexpected phenomenon: as the number of H3-enabled CDN resources increases from the Medium group to the High group, the reduction benefit decreases. We thoroughly investigate and discuss this phenomenon in subsequent Section VI-C. It results from more reused HTTP connections in the High group webpages, narrowing the PLT gap between H3 and H2 and leading to less reduction benefit.

Fast connection contributes the most. The requests for HTTP resources have three main phases: connection establishment, waiting for the request to be processed, and the transmission of data. We investigate which phase contributes the most to PLT reduction. This investigation also provides clues into why the High group webpages do not achieve a significant PLT reduction.

In Fig. 6(b), we plot the Cumulative Distribution Function (CDF) of the reduction of connection, wait, and receive times. When the median value is greater than 0, it means that using H3-enabled CDNs outperforms H2-based CDNs. We observe that the median of connection reduction value is greater than 0. This indicates that connection time reduction makes the main contribution to PLT reduction. This finding is consistent with the results observed by many companies in production environments: the fast connection establishment of H3 is the



(a) The number of reused HTTP connections with H3 and H2

(b) The number of reused connection difference under different groups

(c) The PLT reduction varying reused connection difference

Fig. 7: The relationship among the reused connection, number of H3-enabled CDN resources, and PLT reduction

main reason for performance improvement [5], [10], [12].

At the same time, the median wait reduction value is below 0, indicating that the computation overhead of H3 [37], [38] still needs to be addressed and optimized. Additionally, increasing the capacity of H3 CDN servers is essential [21]. On the other hand, the median receive reduction value being approximately 0 suggests no significant difference in data transmission performance between H3-enabled CDNs and H2-based CDNs on landing pages with small CDN resource sizes [39].

C. Reused HTTP Connections Diminish H3 Adoption Benefits

An interesting phenomenon is observed in Fig. 6(a), where webpages belonging to the High group gain the smallest reduction benefit. The reason that connection time is the primary factor contributing to the PLT reduction gives us clues. After investigating the connection phase of H3 and H2, we conclude that the fundamental reason for the less reduction lies in the high occurrence of reused HTTP connections in the High group webpages. HTTP connection reuse has a similar function as H3's fast connection, which can reduce the connection time. Therefore, for webpages with a number of reused HTTP connections, there is limited room for optimizing their connection time even by adopting advanced H3.

HTTP connection reuse under H3 and H2. HTTP connection reuse [40] refers to manipulating multiple HTTP requests on one persistent HTTP connection, avoiding establishing a separate HTTP connection for each request. This reduces the delay of HTTP connection establishment, which has a similar function as the fast connection of H3 described in Section VI-A. This reused HTTP connection occurs when multiple files are transmitted with a server, such as in the CDN scenarios.

Fig. 7(a) draws the number of reused HTTP connections with H3 and H2. As expected, the occurrence of reused HTTP connections increases with the group level. The more CDN resources there are, the higher the probability of triggering HTTP connection reuse. Notably, we observe that H2 triggers more reused HTTP connections than H3, especially in the High group.

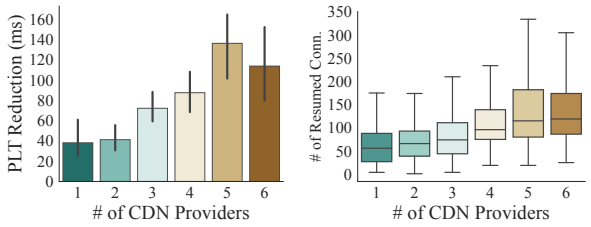
Following this, we explore the difference in the number of reused connections across different groups. We define a *reused connection difference* metric, calculated as the number of reused HTTP connections using H2 minus the number of reused HTTP connections using H3. A positive value means more reused connections when adopting H2 on this webpage, and vice versa. We determine whether a request is performed by a reused HTTP connection based on its connection time recorded in the Chrome-HAR file. If the connection time is 0, then it is a reused connection. In Fig. 7(b), we can more clearly see that H2 triggers more reused HTTP connections than H3, and webpages in the High group have the greatest difference. We could reasonably assume that the higher occurrence of HTTP connection reuse with H2 results in less PLT reduction benefit in the High group.

Reused HTTP connections diminish the benefit of using H3. Based on the above assumption, we study the relationship between the number of reused HTTP connections and PLT reduction. In Fig. 7(c), we depict the PLT reduction with different reused connection differences. It can be observed that as the difference in reused connections increases, the reduction benefit achieved by using H3 becomes smaller. Combining the information in Fig. 7(b), we can ultimately explain the phenomenon of the least PLT reduction in webpages in the High group, and conclude that reused HTTP connections would diminish the PLT reduction benefit from adopting H3.

Lessons on the turning point in performance optimization. After exploring the relationship between the PLT reduction and the number of reused HTTP connections, we would like to discuss the implications of this finding for H3 adoption.

H2 can trigger more reused connections on some pages whose connection phase is already highly optimized. For these pages, even using H3, there is limited room for optimizing connection time. Introducing another version of HTTP might even decrease the probability of connection reuse. This turning point in performance optimization in Fig. 6(a) brings a reminder to developers who switch CDN resources to H3: they need to consider the break-even point between H3 and connection reuse based on specific applications to maximize benefits, rather than adopting H3 blindly.

We think that the difference in reused HTTP connections between H2 and H3 is due to their different deployment scales [21]. According to the analysis in Section IV, only 25.8% of CDN resources currently adopt H3. Therefore, for complicated webpages with a large number of CDN resources,



(a) The PLT reduction of webpages with different numbers of used providers

(b) The number of resumed connections with different numbers of used providers

Fig. 8: Shared providers reduce PLT with resumed connections

H2, having a higher deployment density, is more likely to reuse HTTP connections compared with just started deployed H3. With many CDN servers already supporting H3 and the growth of H3 adoption in webpages, this condition may undergo a transformation in the future. We will continue to keep track of this phenomenon.

Takeaway 2: The fast connection in H3 contributes to accelerating page loading, and the dominant proportion of CDN resources amplifies such acceleration, even with a small-scale H3 adoption. However, reused HTTP connections would diminish such benefits on complicated webpages.

D. Shared Providers Reduce PLT Under Consecutive Visits

The shared-provider phenomenon defined in Section V-B provides a potential pathway for inter-page influence. Specifically, we investigate whether the connections established to CDN servers in the previous webpage visit could impact subsequent pages using the same CDN provider. By answering this question, we aim to uncover dependencies between webpages that share the same CDN provider under H3 adoption.

We perform consecutive visits where the target webpages are visited in a specific order. When visiting the next webpage, all connections are terminated, and the cache is cleared. Terminating all connections disables HTTP connection reuse, but connections can be resumed through the connection resumption mechanism in TLS/1.3 [16]. Connection resumption allows establishing connections quickly without TLS handshake using stored pre-shared keys in high-frequency communication. This benefit can be experienced across pages and in repeated visits [40], [41], and both H3 and H2 can leverage this function. However, H3 can further optimize to 0-RTT resumption [1] by integrating transport layer handshake in TLS handshake, which allows transmitting HTTP data directly. While H2 still needs to wait for 1 RTT for the TCP handshake.

More shared providers, more PLT reduction. We analyze the number of CDN providers used on webpages, including Amazon, Akamai, Cloudflare, Fastly, Google, and Microsoft. In Fig. 8(a), we plot the PLT reduction of webpages with different numbers of CDN providers. The results reveal a positive correlation between the PLT reduction and the number of used providers. Using more providers can achieve more PLT reduction. This finding confirms the presence of the shared-provider phenomenon and its positive impact on page loading.

In the context of consecutive visits, the connection resumption mechanism is the most possible factor to reduce PLT. We examine the number of resumed connections with different numbers of used providers in Fig. 8(b). As expected, using more providers corresponds to more resumed connections. This finding highlights the role of shared providers in triggering the connection resumption. Based on the two findings shown in Fig. 8, we arrive at a persuasive conclusion: For the webpages using multiple providers, the shared-provider phenomenon increases their chances of connection resumption, consequently reducing connection time and PLT.

TABLE III: The PLT reduction comparison of two webpage groups with different sharing degrees

Metric	High sharing group C_H	Low sharing group C_L
Avg num. of shared providers	4.16	2.58
Avg num. of resumed connection	101.64	73.74
PLT reduction (ms)	109.3	54.35

Case study: two groups of webpages with different sharing degrees. Furthermore, we conduct a case study to illustrate the benefits obtained from the shared providers. Specifically, we compare the PLT reduction of two webpage groups with different sharing degrees after adopting H3.

The two groups are constructed based on our target webpages. We extract the domains of all CDN resources used on webpages, and remove outlier webpages whose domains are not used by any other webpages. We extract 58 domains in total, and each remaining webpage is represented by a vector with 58 elements, where each element is a binary indicating whether the corresponding domain appears on that webpage. Based on their vector, we employ the k-means algorithm [42] to divide these webpages into two groups. The first group, denoted as C_H, represents the high-sharing scenario, with an average number of used providers of 4.16. Another group, denoted as C_L, represents the low-sharing scenario, with a lower average number of used providers of 2.58. We also calculate the average number of resumed connections for each group. The high-sharing group C_H has an average of 101.64 resumed connections, while the low-sharing group C_L has an average of 73.74 resumed connections.

We execute consecutive visit measurements on both groups, respectively. The results are presented in Table III. The high-sharing group achieves a greater reduction in PLT with 109.3ms, while the low-sharing group C_L has a smaller reduction of 54.35ms. This suggests that the higher the degree of sharing, the more acceleration is obtained through H3 adoption. These findings highlight that the integration of H3 and CDN can yield optimization with the help of shared providers across various pages, particularly in the common scenario of consecutive web browsing.

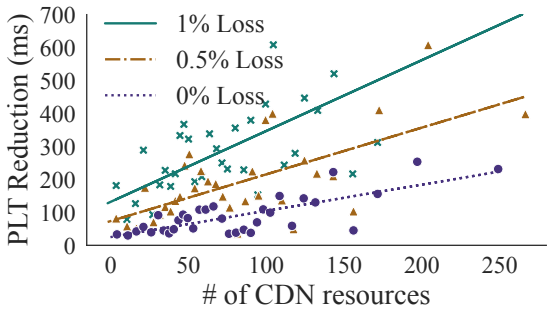


Fig. 9: The PLT reduction with the number of CDN resources under different network loss conditions

Takeaway 3: There is a phenomenon of giant CDN providers being shared across different pages. This shared-provider phenomenon can accelerate page loading by triggering connection resumption of H3 during consecutive webpage browsing. Moreover, the higher the degree of sharing among these browsed pages, the more significant the optimization becomes.

E. Stream Multiplexing Mitigates Congestion Problem

The stream multiplexing of H3 is a feasible solution to mitigate the congestion problem [7], [33]. Higher numbers of CDN resources or increased network loss rates can raise the risk of congestion. Thus, we investigate the achieved PLT reduction varying CDN resource quantities and network loss conditions. We use Traffic Control[6] utility to simulate different loss rate environments.

Fig. 9 shows the results and corresponding fitted curves. As the number of CDN resources on each webpage rises, the PLT reduction gradually increases, consistent with Trevisan et al.'s work [22]. Please note that, since CDN resources are typically small [39], with 75% being below 20KB, H3's encryption delay and its effects on transmission performance can be neglected compared with congestion delay.

Moreover, PLT optimization becomes more evident with higher network loss rates. The slope parameter of the curve with a 1% loss rate is 2.15, significantly higher than those of the curves with 0.5% and 0% loss rates, which are 1.42 and 0.80, respectively. These observations demonstrate the effectiveness of H3 in alleviating congestion, with its efficacy increasing as either the number of CDN resources or the network loss rate rises.

Takeaway 4: Multiple CDN resources centralized on a few providers increase the risk of congestion. Gratefully, H3's stream multiplexing mitigates this problem, particularly in scenarios with numerous CDN resources and a high network loss rate.

[6]https://man7.org/linux/man-pages/man8/tc.8.html

VII. Implications

Building upon the listed takeaways in Sections V and VI, this section will answer the third research question: what implications and insights do these findings have on different roles? We provide several suggestions for CDN providers, end users, web developers, and researchers to maximize the benefit of the H3-enabled CDN, improve user experience, and tackle potential challenges.

CDN providers. CDN providers can prioritize H3 access and invest in upgrading their various infrastructures to support H3, given the acceleration that H3 brings to CDN services. This allows their customers and end users to benefit from the advanced features of H3 for an optimized user experience. However, considering the performance optimization turning point discovered in Section VI-C, CDN providers should develop the most effective HTTP version strategies for different applications and businesses. Conducting performance testing and analysis of various services can help identify suitable candidates for these strategies. Additionally, we advocate for collaboration among CDN providers to drive standardization, development, and optimization of H3.

End users. It is evident that end users can easily experience webpage loading acceleration by simply enabling H3. Therefore, we recommend that end users select web browsers that support H3 to fully enjoy the enhanced browsing experience. The latest versions of mainstream browsers, such as Chrome, Safari, Firefox, and Edge, have already supported H3[7]. Users can easily receive the benefits of H3 by just updating browsers to the latest versions. Moreover, users' preference for accessing CDN resources with H3 can incentivize CDN providers to further optimize H3.

Web developers. Takeaway 3 reveals the correlation between the number of used CDN providers and page load optimization. Based on this finding, web developers can refine their CDN provider selection strategies to fully leverage the shared-provider phenomenon and increase the likelihood of connection resumption. Meanwhile, web developers can consider implementing a hybrid HTTP access approach where H3 is selectively used for specific content or pages that would benefit the most.

Researchers. Given the incomplete deployment of H3 at present, the benefits of using H3 on complicated webpages may not be guaranteed. Researchers can fix this drawback by developing an adaptive protocol selection tool that adjusts flexibly based on different conditions [43]. This allows end users to enjoy the advantages of the latest technologies while ensuring backward compatibility when necessary. This study only reveals how H3 improves performance in large-scale distributed services like CDNs. It is recommended that researchers explore H3's adaptability in various services and further refine its implementation.

[7]https://caniuse.com/http3

VIII. Related Work

A. Measurement Studies of H3

H3 [1] is a promising next-generation application layer protocol and has gained significant attention from academia and industry. A series of experiments conducted in controlled environments demonstrated that H3 improves the transmission performance of multiple files under high loss rates and high latency conditions [6], [7], [43]. Meanwhile, several industrial reports from leading companies like Akamai [5], Cloudflare [28], Fastly [9], and Google [11] also demonstrated the benefits of adopting H3 for web loading acceleration. Additionally, Yu and Benson [4] compared the performance of H2 and H3 by examining the production web services of Google, Meta, and Cloudflare. As H3 deployments expand, there is a need for a comprehensive approach to detect H3 traffic. Zirngibl et al. [14] presented three methods for identifying H3 traffic. This work not only measured the proportion of H3 traffic but also provided valuable insights into its current deployment status.

Our study measures the adoption rate and performance of H3 in CDNs. The findings reveal an expanding deployment of H3 in CDNs and display H3's ability to accelerate content delivery for CDN services.

B. Applications of QUIC

QUIC, a representative transport layer protocol [2], is fundamental for H3's ability to improve service performance and has been adopted in various applications. One of the earliest QUIC applications was DNS over QUIC (DoQ) [44], which had recently been standardized by the Internet Engineering Task Force (IETF). Li et al. [45] enhanced the security and speed of DoQ resolvers by exploring QUIC's connection migration. Meanwhile, Kosek et al. [38] highlighted the remarkable response speed but an overwhelming response packet size of DoQ compared with existing encrypted DNS solutions. Additionally, the IETF is also promoting the standardization for QUIC Load Balancers [46]. Zhou et al. [3] leveraged QUIC Transport Parameters Extension [16] and Server's Preferred Address function [2] to enable fine-grained CDN resource allocation, ensuring faster responses and better meeting user requirements.

We discover that H3 and CDN work together effectively because H3 has the ability to remove repetitive processes in multiple requests. This functionality is also present in QUIC. Therefore, the benefits of H3 on CDNs can also be generalized to other large-scale distributed deployment systems based on QUIC.

C. Measurement Studies of CDNs

CDN [47] is one of the most critical network infrastructures for ensuring content delivery, and it has made significant progress over the past two decades. Well-known giant CDN providers, including Akamai, Cloudflare, Microsoft Azure, Google Cloud CDN [37], [48]–[50], provide large-scale CDN services. In addition to commercial CDN services, companies like Meta [12] have developed their own self-operated CDNs to cater to their substantial data transmission needs. Guo et al. [51] explored the optimization of reducing the synchronization traffic within self-operated CDNs using social network information.

The effectiveness of CDNs has been extensively demonstrated in numerous studies [37], [50], [52]. With developing networking techniques, CDN's performance varies with different protocols. Saverimoutou et al. [21] evaluated the influence of diverse Internet protocols on CDN performance. Shreedhar et al. [37] investigated the download performance of H3 in cloud storage and video streaming scenarios.

In this paper, we have identified three crucial characteristics of CDN services on a range of representative websites. Specifically, CDN resources constitute the major portion of web content, and a small number of large CDN providers dominate resource provision across a diverse range of webpages.

IX. Conclusion and Future Work

In this work, we explore H3's adaptability in CDNs by analyzing various representative websites. We unveil the synergy between H3 and CDNs, attributing it to H3's capacity to eliminate redundant processes in multiple requests. The dominant proportion of CDN resources on webpages accelerates the connection phases with H3's fast connection advantage. Even a small-scale adoption of H3 can achieve a significant PLT reduction. The dominance of CDN providers also gives rise to the phenomenon of sharing providers and serving multiple CDN resources on a single page. Connection and data transmission can be more efficient by leveraging H3's connection resumption mechanism and stream multiplexing. We conclude that three critical characteristics of CDNs align perfectly with H3's strengths, resulting in an optimization in page loading.

Despite providing a series of helpful findings, further research is still needed to better understand the deployment of H3 in CDNs. 1) Exploring the impact of other optimization strategies. Apart from the advanced H3, other optimization strategies, such as browser rendering and CDN load balancing, could also play important roles in web performance. Considering the impact of these issues together is a constructive task for future research. 2) Expanding target websites. Our measurements are conducted on a limited number of landing pages of websites, which may not fully characterize the complexity of the entire Internet. As CDN providers increasingly adopt H3, it becomes essential to expand the set of target websites for more comprehensive results. 3) Globally distributed measurement probes. In this work, all three measurement vantage points are located in the United States. It would be useful to conduct measurements from geographically diverse vantage locations to obtain a global view.

References

[1] M. Bishop, "HTTP/3," RFC 9114. Available: https://www.rfc-editor.org/rfc/rfc9114.html, 2022, accessed: 2024-05-01.

[2] J. Iyengar and M. Thomson, "QUIC: A UDP-Based Multiplexed and Secure Transport," RFC 9000. Available: https://www.rfc-editor.org/rfc/rfc9000.html, 2021, accessed: 2024-05-01.

[3] M. Zhou, T. Guo, Y. Chen, J. Wan, and X. Wang, "Polygon: A QUIC-Based CDN Server Selection System Supporting Multiple Resource Demands," in *Proc. of Middleware*, 2021.

[4] A. Yu and T. A. Benson, "Dissecting Performance of Production QUIC," in *Proc. of WWW*, 2021.

[5] "Deliver Fast, Reliable, and Secure Web Experiences with HTTP/3," Available: https://www.akamai.com/blog/performance/deliver-fast-reliable-secure-web-experiences-http3, 2023, accessed: 2024-05-01.

[6] A. M. Kakhki, S. Jero, D. Choffnes, C. Nita-Rotaru, and A. Mislove, "Taking a Long Look at QUIC: An Approach for Rigorous Evaluation of Rapidly Evolving Transport Protocols," in *Proc. of IMC*, 2017.

[7] P. Megyesi, Z. Krämer, and S. Molnár, "How quick is QUIC?" in *Proc. of ICC*, 2016.

[8] P. K. Kharat, A. Rege, A. Goel, and M. Kulkarni, "QUIC Protocol Performance in Wireless Networks," in *Proc. of ICCSP*, 2018.

[9] "QUIC vs TCP: Which is Better?" Available: https://www.fastly.com/blog/measuring-quic-vs-tcp-computational-efficiency, 2020, accessed: 2024-05-01.

[10] "Why Fastly loves QUIC and HTTP/3," Available: https://www.fastly.com/blog/why-fastly-loves-quic-http3, 2019, accessed: 2024-05-01.

[11] A. Langley, A. Riddoch, A. Wilk, A. Vicente, C. Krasic, D. Zhang, F. Yang, F. Kouranov, I. Swett, J. Iyengar, *et al.*, "The QUIC Transport Protocol: Design and Internet-scale Deployment," in *Proc. of SIGCOMM*, 2017.

[12] "How Facebook is bringing QUIC to billions," Available: https://engineering.fb.com/2020/10/21/networking-traffic/how-facebook-is-bringing-quic-to-billions, 2020, accessed: 2024-05-01.

[13] J. Rüth, I. Poese, C. Dietzel, and O. Hohlfeld, "A First Look at QUIC in the Wild," in *Proc. of PAM*, 2018.

[14] J. Zirngibl, P. Buschmann, P. Sattler, B. Jaeger, J. Aulbach, and G. Carle, "It's Over 9000: Analyzing Early QUIC Deployments with the Standardization on the Horizon," in *Proc. of IMC*, 2021.

[15] Amazon.com Inc., "The Top 500 Sites on the Web," https://www.alexa.com/topsites, 2021, accessed: 2022-05-01.

[16] M. Thomson and S. Turner, "Using TLS to Secure QUIC," RFC 9001. Available: https://www.rfc-editor.org/rfc/rfc9001.html, 2021, accessed: 2024-05-01.

[17] M. Scharf and S. Kiesel, "Head-of-line Blocking in TCP and SCTP: Analysis and Measurements," in *Proc. of GLOBECOM*, 2006.

[18] "QUIC — Cloudflare," Available: https://cloudflare-quic.com, 2023, accessed: 2024-05-01.

[19] A. Ghedini and R. Lalkaka, "HTTP/3: the past, the present, and the future," Available: https://blog.cloudflare.com/http3-the-past-present-and-future, 2019, accessed: 2024-05-01.

[20] "HTTP/3 gets your content there QUIC, with Cloud CDN and Load Balancing," Available: https://cloud.google.com/blog/products/networking/cloud-cdn-and-load-balancing-support-http3, 2021, accessed: 2024-05-01.

[21] A. Saverimoutou, B. Mathieu, and S. Vaton, "Influence of Internet Protocols and CDN on Web Browsing," in *Proc. of NTMS*, 2019.

[22] M. Trevisan, D. Giordano, and A. S. Khatouni, "Measuring HTTP/3: Adoption and Performance," in *Proc. of MedComNet*, 2021.

[23] "Making loveholidays 18% faster with HTTP/3," Available: https://tech.loveholidays.com/making-loveholidays-18-faster-with-http-3-1860879528a7, 2021, accessed: 2024-05-01.

[24] "QUIC.cloud CDN is Production Ready!" Available: https://www.quic.cloud/quic-cloud-cdn-production-ready, 2021, accessed: 2024-05-01.

[25] C. Yun, "HTTP/3 Support for Amazon CloudFront," Available: https://aws.amazon.com/blogs/aws/new-http-3-support-for-amazon-cloudfront, 2022, accessed: 2024-05-01.

[26] T. Ingale, "Watch Meta's engineers discuss QUIC and TCP innovations for our network," Available: https://engineering.fb.com/2022/07/06/networking-traffic/watch-metas-engineers-discuss-quic-and-tcp-innovations-for-our-network, 2022, accessed: 2024-05-01.

[27] "HTTP/3 is added by default to a new Ion property," Available: https://techdocs.akamai.com/ion/changelog/may-15-2023-support-for-http3, 2023, accessed: 2024-05-01.

[28] S. Tellakula, "Comparing HTTP/3 vs. HTTP/2 Performance," Available: https://blog.cloudflare.com/http-3-vs-http-2, 2020, accessed: 2024-05-01.

[29] "Switch from Cloudflare for Better TTFB," Available: https://www.quic.cloud/quic-cloud-for-better-ttfb, 2023, accessed: 2024-05-01.

[30] W. Aqeel, B. Chandrasekaran, A. Feldmann, and B. M. Maggs, "On Landing and Internal Web Pages: The Strange Case of Jekyll and Hyde in Web Performance Measurement," in *Proc. of IMC*, 2020.

[31] K. Ruth, D. Kumar, B. Wang, L. Valenta, and Z. Durumeric, "Toppling Top Lists: Evaluating the Accuracy of Popular Website Lists," in *Proc. of IMC*, 2022, pp. 374–387.

[32] R. Huang, M. Zhou, T. Guo, and Y. Chen, "Locating CDN Edge Servers with HTTP Responses," in *Proc. of SIGCOMM Poster and Demo Sessions*, 2022.

[33] M. Rajiullah, A. Lutu, A. S. Khatouni, M.-R. Fida, M. Mellia, A. Brunstrom, O. Alay, S. Alfredsson, and V. Mancuso, "Web Experience in Mobile Networks: Lessons from Two Million Page Visits," in *Proc. of WWW*, 2019.

[34] R. Netravali, V. Nathan, J. Mickens, and H. Balakrishnan, "Vesper: Measuring Time-to-Interactivity for Web Pages," in *Proc. of NSDI*, 2018.

[35] J. Yang, A. Sabnis, D. S. Berger, K. Rashmi, and R. K. Sitaraman, "C2DN: How to Harness Erasure Codes at the Edge for Efficient Content Delivery," in *Proc. of NSDI*, 2022.

[36] P. Biswal and O. Gnawali, "Does QUIC make the Web faster?" in *Proc. of GLOBECOM*, 2016.

[37] T. Shreedhar, R. Panda, S. Podanev, and V. Bajpai, "Evaluating QUIC Performance Over Web, Cloud Storage, and Video Workloads," *IEEE Transactions on Network and Service Management*, vol. 19, no. 2, pp. 1366–1381, 2022.

[38] M. Kosek, L. Schumann, R. Marx, T. V. Doan, and V. Bajpai, "DNS Privacy with Speed? Evaluating DNS over QUIC and its Impact on Web Performance," in *Proc. of IMC*, 2022.

[39] P. Gill, M. Arlitt, Z. Li, and A. Mahanti, "Youtube Traffic Characterization: A View From the Edge," in *Proc. of IMC*, 2007.

[40] S. Singanamalla, M. T. Paracha, S. Ahmad, J. Hoyland, L. Valenta, Y. Safronov, P. Wu, A. Galloni, K. Heimerl, N. Sullivan, C. A. Wood, and M. Fayed, "Respect the ORIGIN! A Best-case Evaluation of Connection Coalescing in The Wild," in *Proc. of IMC*, 2022, pp. 664–678.

[41] S. Cook, B. Mathieu, P. Truong, and I. Hamchaoui, "QUIC: Better For What And For Whom?" in *Proc. of ICC*, 2017.

[42] J. MacQueen, "Some Methods for Classification and Analysis of Multivariate Observations," in *Proc. of the fifth Berkeley Symp. Math. Statist. Probability*, 1967.

[43] M. Zhou, Z. Li, S. Lin, X. Wang, and Y. Chen, "FlexHTTP: An Intelligent and Scalable HTTP Version Selection System," in *Proc. of EuroMLSys*, 2022.

[44] C. Huitema, S. Dickinson, and A. Mankin, "DNS over Dedicated QUIC Connections," RFC 9250. Available: https://www.rfc-editor.org/info/rfc9250, 2022, accessed: 2024-05-01.

[45] X. Li, Y. Chen, M. Zhou, T. Guo, C. Wang, Y. Xiao, J. Wan, and X. Wang, "Artemis: A Latency-Oriented Naming and Routing System," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 12, pp. 4874–4890, 2022.

[46] M. Duke and N. Banks, "QUIC-LB: Generating Routable QUIC Connection IDs," Available: https://datatracker.ietf.org/doc/draft-ietf-quic-load-balancers, 2023, accessed: 2024-05-01.

[47] J. Dilley, B. Maggs, J. Parikh, H. Prokop, R. Sitaraman, and B. Weihl, "Globally distributed content delivery," *IEEE Internet Computing*, vol. 6, no. 5, pp. 50–58, 2002.

[48] A.-J. Su, D. R. Choffnes, A. Kuzmanovic, and F. E. Bustamante, "Drafting Behind Akamai: Inferring Network Conditions Based on CDN Redirections," *IEEE/ACM Transactions on Networking*, vol. 17, no. 6, pp. 1752–1765, 2009.

[49] A. Flavel, P. Mani, D. A. Maltz, N. Holt, J. Liu, Y. Chen, and O. Surmachev, "FastRoute: A Scalable Load-Aware Anycast Routing Architecture for Modern CDNs," in *Proc. of NSDI*, 2015.

[50] V. K. Adhikari, Y. Guo, F. Hao, V. Hilt, Z.-L. Zhang, M. Varvello, and M. Steiner, "Measurement Study of Netflix, Hulu, and a Tale of Three CDNs," *IEEE/ACM Transactions On Networking*, vol. 23, no. 6, pp. 1984–1997, 2014.

[51] T. Guo, Y. Ma, M. Zhou, X. Wang, J. Wu, and Y. Chen, "SocialCache: A Pervasive Social-Aware Caching Strategy for Self-Operated Content Delivery Networks of Online Social Networks," in *Proc. of ICC*, 2023.

[52] L. Wang, V. Pai, and L. Peterson, "The Effectiveness of Request Redirection on CDN Robustness," in *Proc. of OSDI*, 2002.