Impact of TCP Loss on Regional Application Performance

Ryan Gao, Prasad Tammana, Sorabh Gandhi, Matt Calder (Microsoft), Ethan Katz-Bassett (Columbia)

ABSTRACT

Content providers deploy serving infrastructure around the globe to provide low latency endpoints and caching services that are critical for good user experience. Despite efforts to improve performance by pushing content and infrastructure close to users, performance in many regions of the world still lags far behind what is observed in North America and Europe. Poor performance can result from excessive TCP loss, so understanding regional loss characteristics is a critical step toward addressing regional performance issues that degrade user experience and prevent a uniform global experience.

In this work, we examine regional characteristics of TCP loss and how they impact performance in different application workloads. Our analysis of four production applications in four countries quantify how regional differences in loss behavior contribute to differences in regional application performance.

1. INTRODUCTION

Global content providers typically rely on in-house or third-party content delivery networks (CDNs) to improve latency between clients and content. Microsoft's internal CDN, Azure Frontdoor, improves network performance for many of Microsoft's critical online services by deploying split-TCP proxies between clients and backend application servers. Like many content providers, Azure Frontdoor deploys servers around the world to reduce network and topological distance to users in order to improve performance [16].

However, even with servers close to users in developing nations, those users may experience performance much worse than that of users in developed regions. Figure 1 shows the performance difference in transfer latency between Cloud Storage (§2.1) users in Germany, Brazil, and India, relative to performance in the United States. Except for single digit percentiles (the fastest performance), performance in Germany nearly mirrors the United States. However, even with Azure Frontdoor servers in all four countries, users in Brazil and India experience significantly slower performance at the mid and upper percentiles, compared to users in the United States. At the 95th percentile, the performance of a user in the United States is twice as fast as a Brazilian user and three times as fast as an Indian user.

Poor performance decreases revenue and frustrates users of applications behind Azure Frontdoor. Global enterprise customers desire a uniform user experience across their business. Latency sensitive consumer services, such as search, where revenue is primarily driven by ads, see direct revenue loss from poor performance [15].

However, accounting for regional performance differences and attributing the causes of any single regional network difference remains challenging. The community lacks an understanding of regional differences in loss characteristics and their corresponding performance impact on different classes of popular applications, despite the fact that many Internet performance problems [2, 17, 9] manifest as TCP loss.

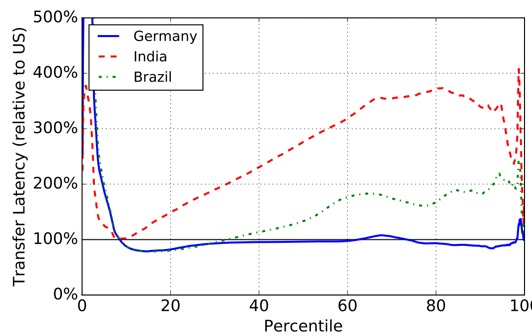

Figure 1: Cloud Storage Network Latency, relative to performance from the United States.

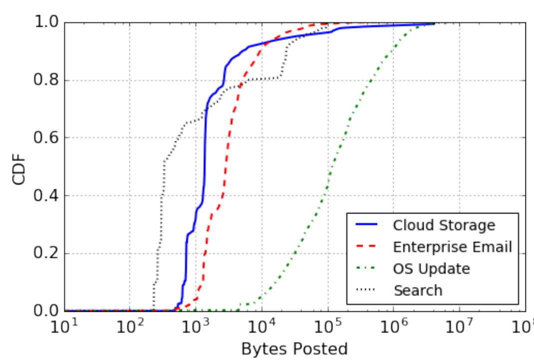

Figure 2: Response Size Distribution of Cloud Storage, Enterprise Email, OS Update, and Search traffic.

In this work, we examine the regional differences in TCP loss behavior and capture how much these differences explain regional performance differences. We focus on Brazil and India, in relation to the developed markets of Europe and North America, because of their importance to Microsoft's business and the persistent performance challenges associated with operating in those regions. Microsoft has invested heavily in both networking and compute infrastructure in these countries, including Azure Frontdoor locations and internal and public Azure data centers. India has around 250 million people with web-connected devices that make heavy use of cloud-based applications and services. The cloud market in India is expected to reach $2 billion by 2020 [20]. In 2015 alone, Microsoft brought three new Azure regions online in India [1]. As of 2016, Brazil is the largest cloud services market in Latin America, expected to reach $1.1 billion by 2017 [19].

We collected millions of measurements from production Azure Frontdoor traffic for four application customers (§2.1) in four regions. Our cross-workload analysis reveals that loss rates vary greatly across regions, as does the proportion of those losses recovered via TCP fast retransmissions versus retransmission timeouts (RTOs).

We make the following key observations:

- Overall loss rates for Brazil and India are 30-40% higher than observed in North America and Europe.
- While 40%-70% of lossy connections in Brazil, North America, and Europe recover from all losses without incurring a retransmission timeout, 60%-80% of lossy connections in India incur a costly timeout.
- At the 75th percentile, slow retransmissions are responsible for 15%-20% of the performance difference between India and the United States while for Brazil it is only 10%-12%.
- Retransmission timeouts contribute to a large portion of the performance difference between India and the United States for latency sensitive applications. But even after discarding requests from India that incurred an RTO, the 95th percentile latency is still 30% higher than in the United States.

While it is well understood that in general, network performance can be worse in developing compared to developed regions, in this work we seek to capture how this translates to regional differences in global production applications.

2. BACKGROUND

2.1 Azure Frontdoor

Azure Frontdoor is Microsoft's first-party CDN, both serving static content and offering split-TCP connection points that reverse proxy requests to their appropriate application origins. Azure Frontdoor is deployed in many dozens of locations around the world. We refer to a group of co-located Azure Frontdoor servers in a particular geographic location as an *edge*. Azure Frontdoor uses anycast redirection [10, 4] to direct users to a nearby edge. Azure Frontdoor hosts many popular Microsoft services such as Office365, OneDrive, Outlook, and Bing, which we refer to as *customer* applications.

2.2 TCP loss and retransmissions

Most widely deployed TCP variants detect loss and trigger retransmissions in two ways:

Slow retransmissions / retransmission timeouts. TCP starts a retransmission timer every time it hands outbound packets to the IP layer. If the timer expires before TCP receives an acknowledgement, then it triggers an RTO.

Fast retransmissions. Each outbound packet carries a sequence number that identifies which segment of data is being sent, and each incoming acknowledgement (ACK) carries the sequence number of the next expected data segment. When the sender receives duplicate ACKs with a sequence number that is earlier than the last sequence number sent, it can infer that at least one segment has been lost. It can then resend the lost data, without waiting for the retransmission timer to expire.

3. METHODOLOGY

3.1 Data Set

We collected server-side application and TCP-level events for responses across four Azure Frontdoor customers, each with a different application behavior and workload.

- **Cloud Storage**: Cloud file hosting service where users store arbitrary files such as music, images, and video.
- **Enterprise Email**: Corporate web email services.
- **OS Update**: Feature and security updates.
- **Search**: Bing web search engine.

We ran our collection on a single machine for each customer/ location combination. Search, Cloud Storage, and OS Update locations were San Jose, U.S.A.; Frankfurt, Germany; Sao Paulo, Brazil; and Mumbai, India. Because some customers have specific location requirements, Enterprise Email data was from a different set of locations: Sao Paulo, Brazil; Chicago, U.S.A; Amsterdam, Netherlands; and Singapore. Our analysis shows that for the U.S. West and Central, and Netherlands and Germany, traffic distributions are similar enough to get value from comparison. We don't believe that this has significant impact on our results. We avoid making comparisons between Singapore and other edge locations, but it is included for completeness.

Our data set contains only server-to-client responses (i.e. Search results but not requests). We exclude internal monitoring and bot traffic, so that our data is representative of client performance only. Table 1 shows the number of HTTP responses we recorded per edge customer and edge location. The applications differ on various attributes, including the response size and latency to the application origin. For example, we expect OS Update to deliver much larger responses than Search, so the TCP loss behavior will greatly differ. Figure 2 shows the edge-to-client response size distribution for each customer.

We had several constraints to manage that impacted our data collection. Since our collection is running in production environments, we needed to ensure that the collection does not impact services. We chose a fairly conservative sampling rate that collected around 250 thousand responses for each customer-location pair. Collecting that many responses took 2-3 days for three of the services, but it took 3 times longer for OS Update because it is not primarily driven by client activity.

3.2 Data Collection

We captured TCP and application layer events using Event Tracing for Windows (ETW), a kernel-level tracing feature in Windows. ETW can enable or disable event tracing dynamically, allowing detailed tracing in a production environment without requiring server or application restarts. In our collection, we use two standard Windows ETW event publishers, Microsoft-Windows-TCPIP and Microsoft-Windows-HttpService for TCP/IP and HTTP events, respectively. We also use an Azure Frontdoor-internal ETW publisher to capture split TCP timings. Each event has a high resolution timestamp and custom data fields. The TCP data for responses captured on Azure Frontdoor are correlated with detailed logs for every request that Azure Frontdoor serves. This includes information about client location and ASN, protocol used (HTTP/HTTPS), the Edge that served the request, and OS and browser information from the user-agent.

Customers on Azure Frontdoor collect End-User Latency (§ 3.3) using the W3C resource timing API [13]. This data includes information such as page load time that can be correlated with the other logs.

3.3 Metrics

Here we describe the metrics used to evaluate regional loss in our applications and how they are calculated.

Transfer Latency. The total time spent on the network transferring the response as measured from the server. The delta between first byte of response sent and final ACK received.

End-User Latency. Page load time (PLT). Client-side measure of how long a webpage took to finish rendering using the W3C Resource Timing API and reported back to Microsoft.

Slow Retransmit. We detect slow retransmissions (§ 2.2) in a response from an explicit ETW event from Windows TCP stack. We do not know how many bytes were retransmitted.

Fast Retransmit. Unlike slow retransmits, fast retransmits have no explicit ETW event in the version of Windows Server running on Azure Frontdoor. We have to infer fast retransmits by checking if the network layer transmitted more payload bytes than the application layer sent. In cases where there is no slow retransmit event in the response, we know that there must have been a fast retransmit. A limitation of our data collection scheme is that we can only categorize responses as having slow retransmits and maybe fast retransmits, or just fast retransmits. Because of this, any response which has a slow transmit gets categorized as such even it may have had fast retransmits as well. This also prevents us from establishing a third category of 'slow and fast retransmits' in our analysis. Previous work has shown that slow retransmits dominate the time taken to complete TCP requests [9], so we don't believe that rolling some fast retransmits into slow will have substantial impact on our results.

Response Retransmission Rate. The fraction of responses with particular types of loss.

4. RESULTS

Our results first focus on the performance impact of regional loss across Search, Cloud Storage, and Enterprise Email workloads as captured by response transfer latency and page load time. We then examine regional retransmission rates across our workloads by breaking down variation in total, slow, and fast retransmission ratios. To better understand the characteristics that influence retransmission rates, we examine variation across a number of different network properties.

As mentioned in §3.1 and Table 1, the Enterprise Email deployment locations differ from those of other applications: the European location is the Netherlands rather than Germany, the US location is Central US rather than West US, and the South Asia location is Singapore rather than India.

4.1 Impact of Loss on Performance

We first examine the impact that loss has on performance in different client regions across several workloads. For each application, we find the performance at the 75th and 95th percentiles relative to *all responses from that application served from the United States* at the same percentile. In other words, we are using all responses from the United States as our baseline for performance comparison. The 75th percentile (%ile) is a standard metric used for network-related performance across many Microsoft teams. The 95th%ile is important because retransmissions, especially RTOs, disproportionately impact the tail of transfers and we are interested in understanding how extreme user performance degradation is at the far end of the spectrum.

4.1.1 Transfer Latency

Transfer latency is the total time spent serving the request as measured from the server-side. Examining transfer time is important because it is a direct measure of the time an application has spent waiting for the network, as opposed to application-specific PLT which is impacted by a number of non-network factors such as external resource retrieval and page rendering capabilities [21]. Since Azure Frontdoor is application-agnostic and so is transfer latency, this is one of the key performance indicators for our service.

Figure 3 and Figure 4 show the significant impact of slow retransmissions on high percentile latencies, especially in regions with poor network infrastructure like India. At the 75th%ile, all applications in India are 2x-3x worse than the US baseline. At the 95th%ile, responses served to Indian users with neither slow nor fast retransmissions are within 1.5x the US baseline. For both Cloud Storage and Enterprise Email, responses served to Brazil stay within 1x-2x of the US latencies. Brazilian users appear to be especially well served

Edge Location	Search	Date	Enterprise Email	Date	Cloud Storage	Date	OS Update	Date
US West	250,060	4/1-4/4	-	-	249,936	4/5-4/7	251,119	4/5-4/15
Germany	250,884	4/1-4/3	-	-	250,224	4/5-4/6	250,067	4/5-4/8
Brazil	249,822	4/1-4/3	269,699	4/5-4/7	249,558	4/5-4/6	249,022	4/5-4/6
India	249,234	4/1-4/3	-	-	250,282	4/5-4/8	249,792	4/5-4/7
US Central	-	-	272,837	4/1-4/3	-	-	-	-
Netherlands	-	-	274,105	4/1-4/3	-	-	-	-
Singapore	-	-	183,327	4/1-4/8	-	-	-	-

Table 1: Response counts and dates, by customer and region.

by Search, which has similar performance to the US baseline at the 75th%ile and better performance at the 95th%ile. In nearly all cases, responses for European users show better transfer latency performance than the US.

These results first demonstrate that while we targeted both Brazil and India as developing regions, India's performance can be as much as 2x worse than Brazil's. We also demonstrate the importance of application workload when evaluating network performance from the server-side.

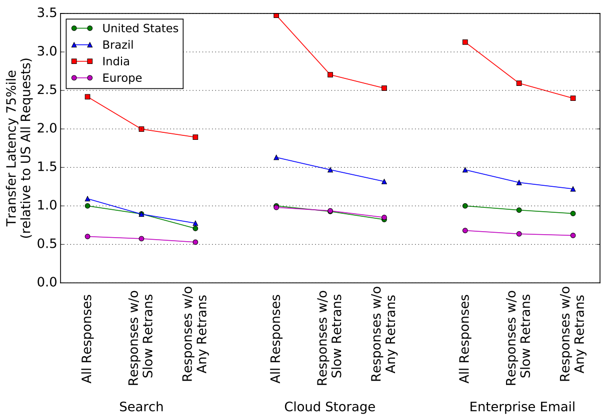

Figure 3: Transfer latency at 75th%ile for Search, Enterprise Email, and Cloud Storage for users in different regions.

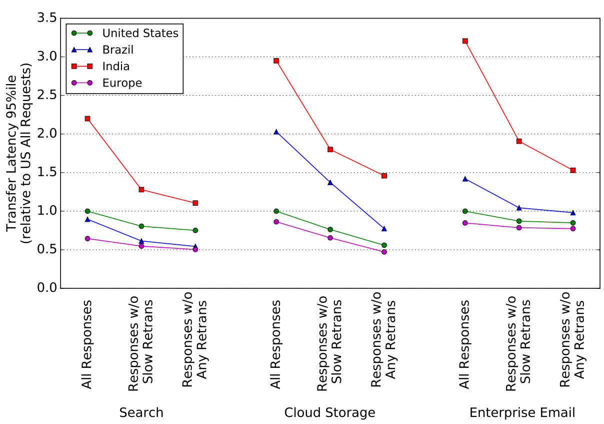

Figure 4: Transfer latency at 95th%ile for Search, Enterprise Email, and Cloud Storage for users in different regions.

4.1.2 *End-User Latency*

Azure Frontdoor customers also continuously collect client-side measurement data such as page load time (PLT) via the W3C resource timing API. This enables us to gauge the impact of retransmissions on end-user perceived latency,

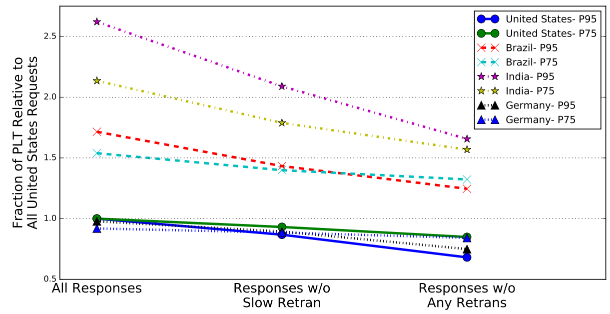

Figure 5: Page load time of Search for different client countries.

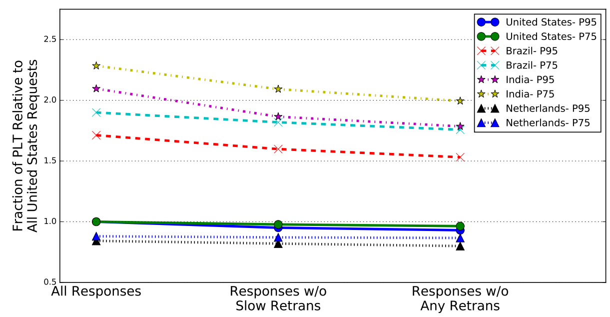

Figure 6: Page load time for Enterprise Email for different client countries.

which is a direct measure of the effectiveness of Azure Frontdoor in application acceleration.

Figure 5 for Search and Figure 6 for Enterprise Email show the performance difference in PLT if transfers with slow retransmissions are all removed, relative to all responses in the United States. Another way to view this is: what is the total impact that slow retransmits have on regional differences in end-to-end latency? For Search users in India, we can see that the relative performance impact of slow retransmissions is substantial and nearly mirrors the transfer latency results in Figures 3 and 4. Interestingly, Brazil Search users shows 1.25x-1.75x higher PLT than the US baseline whereas the transfer latency was comparable. As with transfer latency, the European results match the US quite closely.

In Figure 6, Enterprise Email PLT shows a much different behavior than Search for both India and Brazil in that the 75th%ile PLT is actually farther away from the US 75th%ile baseline than the 95th%ile. This is also reflected in the transfer latency results. We also see that Enterprise Email PLT performance appears to be impacted much less by responses with slow and no retransmissions compared to Search. Both the US and Europe see very little difference in performance between all responses and those without any retransmissions.

4.2 Variation in Retransmission Behavior

In the previous section, we looked at how loss impacts user performance in different regions and applications by excluding responses with retransmissions. Next, we examine what retransmission rates look like for responses served by different Edges.

The rates of retransmission differ substantially across workloads and edge locations, as shown in Figure 7. Edges in Europe (Germany, Netherlands) and the United States (US West, US Central) serve traffic suffering few slow retransmissions, on the order of 10% for large OS Update requests, and 2-3% for small (Search, Enterprise Email, and Cloud Storage) requests. The Indian edge serves traffic with roughly 3 times the rate of slow retransmissions, across all workloads, while the Brazilian edge falls in between the extremes. These results are in line with Figure 1 – our expectations of North America and Europe having the most reliable, least lossy Internet infrastructure, Latin America having moderately reliable infrastructure, and India having the least reliable infrastructure.

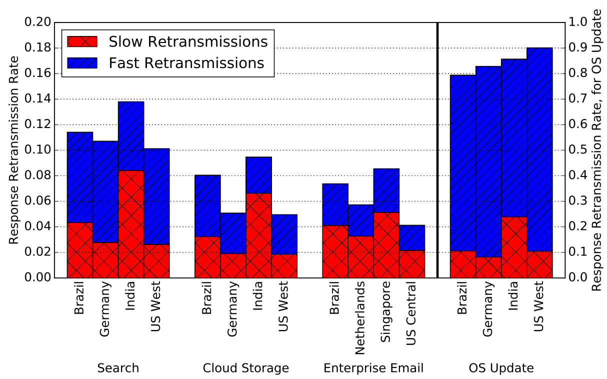

Figure 7: Request Retransmission Rate shows significant variation by customer and edge location.

4.3 Influences on Retransmission Behavior

The previous section showed how retransmission rates and the ratio of slow and fast retransmissions vary across edge regions and applications. We now explore the factors that impact transmission rates within a region. As described in section 3.2, each request served by Azure Frontdoor is logged and joined with lots of additional metadata such as client location, client ISP, ISP network type (mobile or broadband), and OS and browser information extracted from the user-agent string. To search for factors that influence retransmission behavior, we dimension our data on this request metadata. We evaluated dozens of variables that could impact retransmission rate at a given edge, including RTT, geographic distance between client and the edge, time spent waiting for back-end response, HTTP vs. HTTPS, OS, and browser. We found that, even controlling for other factors, loss rate and slow:fast distributions varied the most depending on the client's country and the client's network type.

4.3.1 Client Country

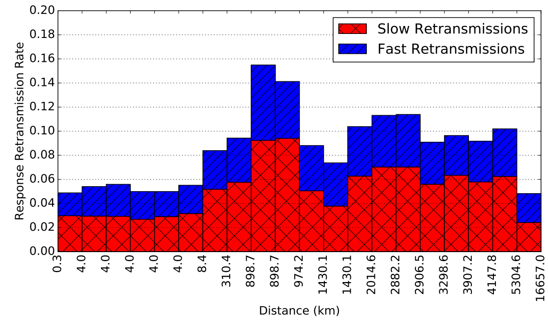

Figure 8: Retransmission rates of Enterprise Email traffic served from Singapore from clients at different distances. Distance doesn't really matter, but origin country does.

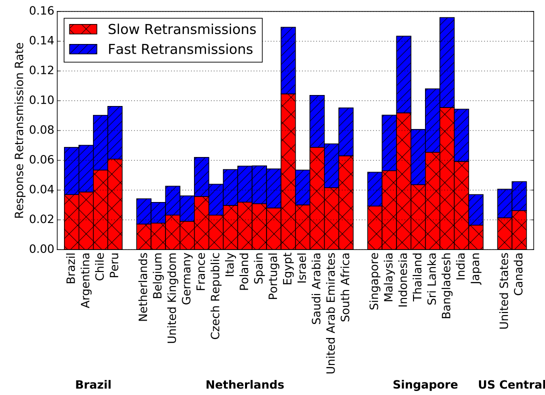

Figure 9: Retransmission rates for Enterprise Email traffic, dimensioned on client country and Edge country (bolded). Ordered by median distance.

Figure 8 shows the effects of distance on retransmission rates. The x-axis, representing distance from our Singapore edge, has been divided into 20 equal-request-count buckets. A plurality of requests originate from Singapore (about 4 km), but there is also substantial traffic from Jakarta, Indonesia (about 900 km) and Bangkok, Thailand (about 1430 km). This figure demonstrates that clients served from distant edges are not necessarily more prone to loss than nearby clients. For example, requests from Jakarta have a much higher retransmission rate than requests from Bangkok, despite the clients being closer to the edge. The last bin, largely requests from Japan, has lower retransmission rate than even requests from Singapore.

Figure 9 shows how retransmission rate varies across the highest traffic volume countries served by different edges. For Enterprise Email requests, the distribution of transferred bytes is highly consistent across all (client country, edge country) combinations, so that will not be a confounding factor. The US Central edge primarily serves North America users that have slow retransmission rates less than 3%. The Netherlands in contrast serves a very diverse set of countries including all over Europe, the Middle East, and Africa [11, 8]. Interestingly, Middle Eastern countries with good Internet infrastructure, such as Israel and the United Arab Emirates, have retransmission rates comparable to those of much closer France, while Egypt and Saudi Arabia have the highest

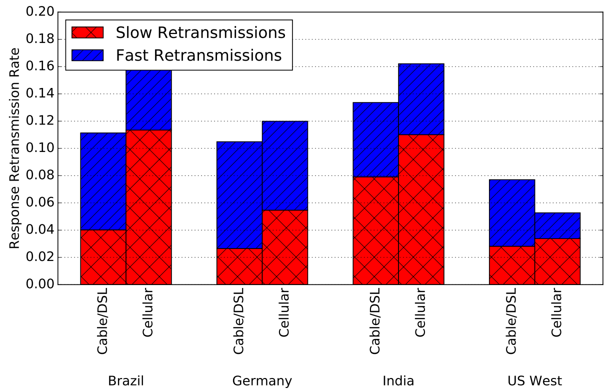

Figure 10: Retransmission rates for Search traffic, dimensioned on network type.

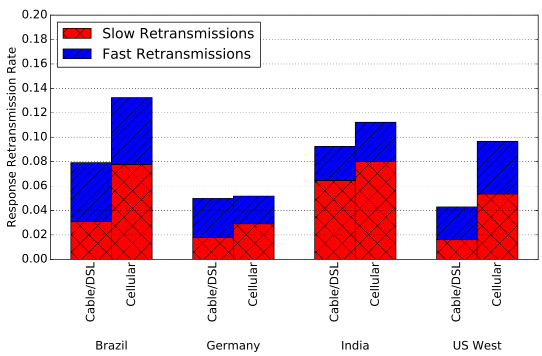

Figure 11: Retransmission rates for Cloud Storage traffic, dimensioned on network type.

retransmission rates served by the edge. Singapore, in comparison to the Netherlands, mostly serves countries which see higher slow retransmission rates. Two-thirds of countries have rates of 4% or more and half have rates of 6% or more. Japan is an interesting exception to this group, by having one of the lowest slow retransmission rates out of all countries. The Brazil edge shows 2 distinct bands of slow retransmission rates, one corresponding to Brazil and Argentina, and the other to Chile and Peru.

4.3.2 Network Type

Mobile/cellular network performance is a well-researched area [12]. User-devices may have particularly lossy link layer connections from 3G, 4G, or LTE service. We infer the end client's network type from their IP address using the Maxmind GeoIP2 Connection Type database. In Figure 10 and Figure 11, we compare the retransmission fractions for Search and Cloud Storage traffic from clients on a "Cable/DSL" connection and clients on a "Cellular" connection. Across all regions, we observe increased slow retransmission rates in cellular clients. For Search, both Brazil and India have very high loss and slow retranmission rates in cellular networks, but in India the difference between Cable/DSL and cellular is only 3%, whereas in Brazil it is 7%. For Cloud Storage, retransmission rates are overall lower across all countries than they are for Search. Cellular clients in the United States see particularly high loss rates (both slow and fast retransmissions) compared to cable/DSL clients there and cellular clients in Germany.

5. DISCUSSION

Based on our findings around the variation of TCP loss in different regions, we believe that providing regional and per-AS server-side TCP settings can improve regional performance. Windows Server natively supports different TCP configurations on a per-prefix basis. These network optimizations would complement common application layer practices such as image resizing and low video bitrate stream that attempt to optimize content for low bandwidth or high latency clients. Our initial focus would be on initial congestion window and maximum transmission unit (MTU) settings.

Previous work such as Google's argument to increase TCP's initial congestion window [7] failed to look at the impact of specific regions. This work also reflected a different time in Internet history, where sophisticated mobile devices and cloud compute had not yet impacted services in the way they have in 2016.

As a follow up to this work, we plan to identify a segment of the worst performing users in India and Brazil, investigate the client TCP settings observed from the server-side, and deploy customized TCP settings per customer for these clients to improve performance.

6. RELATED WORK

The challenges of Internet performance in developing nations have been examined before [11, 8, 6] with an emphasis on Africa. This work has typically focused on the importance of modernizing interconnection [5] and improving peering efficiency [22]. TCP packet loss and retransmission behavior has been heavily studied [14, 3, 17, 18] but this is the first work we are aware of that examines regional properties with cross application analysis. Prior work characterizing a CDN similar to Azure Frontdoor found that slow transmissions can be very costly compared to fast retransmissions [9]. They propose several techniques for reducing packet loss and converting RTOs into fast retransmits. Our work instead focuses on the regional behavior of TCP retransmissions with a focus across four production applications.

7. CONCLUSION

Deploying serving infrastructure in close proximity to users still leaves developing nations with much worse performance than developed nations. In an effort to capture the factors responsible for these gaps, we have examined the regional differences of TCP loss with a cross-application analysis. We have found that in Brazil and India, higher retransmit rates and the larger proportion of RTOs as compared to fast retransmits both contribute to a significant portion of the performance gap with the United States and Europe. For several applications, India may still see a 25-50% performance gap, even if all RTOs are eliminated.

8. REFERENCES

[1] Azure Regions. https://azure.microsoft.com/en-us/regions/, October 2016.

[2] H. Balakrishnan et al. Tcp behavior of a busy internet server. Technical report, Technical Report CSD-97-966, University of California at Berkeley.

[3] H. Balakrishnan, V. N. Padmanabhan, S. Seshan, M. Stemm, and R. H. Katz. Tcp behavior of a busy internet server: Analysis and improvements. In *INFOCOM'98*.

[4] M. Calder, A. Flavel, E. Katz-Bassett, R. Mahajan, and J. Padhye. Analyzing the performance of an anycast cdn. In *IMC 2015*.

[5] N. Chatzis, G. Smaragdakis, and A. Feldmann. On the importance of internet exchange points for today's internet ecosystem. 2013.

[6] M. Chetty, S. Sundaresan, S. Muckaden, N. Feamster, and E. Calandro. Measuring broadband performance in south africa. In *Symposium on Computing for Development 2013*.

[7] N. Dukkipati, T. Refice, Y. Cheng, J. Chu, T. Herbert, A. Agarwal, A. Jain, and N. Sutin. An argument for increasing tcp's initial congestion window. *ACM SIGCOMM CCR*.

[8] R. Fanou, P. Francois, and E. Aben. On the diversity of interdomain routing in africa. In *PAM 2015*.

[9] T. Flach, N. Dukkipati, A. Terzis, B. Raghavan, N. Cardwell, Y. Cheng, A. Jain, S. Hao, E. Katz-Bassett, and R. Govindan. Reducing web latency: the virtue of gentle aggression. In *ACM SIGCOMM CCR*.

[10] A. Flavel, P. Mani, D. Maltz, N. Holt, J. Liu, Y. Chen, and O. Surmachev. FastRoute: A Scalable Load-Aware Anycast Routing Architecture for Modern CDNs. In *NSDI '15*.

[11] A. Gupta, M. Calder, N. Feamster, M. Chetty, E. Calandro, and E. Katz-Bassett. Peering at the internet's frontier: A first look at isp interconnectivity in africa. In *PAM 2014*.

[12] J. Huang, F. Qian, Y. Guo, Y. Zhou, Q. Xu, Z. M. Mao, S. Sen, and O. Spatscheck. An in-depth study of lte: effect of network protocol and application behavior on performance. In *ACM SIGCOMM CCR*.

[13] A. Jain, J. Mann, Z. Wang, and A. Quach. W3C Resource Timing Working Draft. http://www.w3.org/TR/resource-timing/.

[14] D. Lin and H. Kung. Tcp fast recovery strategies: Analysis and improvements. In *INFOCOM'98*.

[15] G. Linden. Make data useful. http://sites.google.com/site/glinden/Home/StanfordDataMining.2006-11-28.ppt, 2006.

[16] E. Nygren, R. K. Sitaraman, and J. Sun. The Akamai Network: A Platform for High-performance Internet Applications. *SIGOPS '10*.

[17] S. Rewaskar, J. Kaur, and F. D. Smith. A performance study of loss detection/recovery in real-world tcp implementations. In *ICNP 2007*.

[18] P. Sun, M. Yu, M. J. Freedman, and J. Rexford. Identifying performance bottlenecks in cdns through tcp-level monitoring. In *ACM SIGCOMM workshop on Measurements up the stack*.

[19] U.S. Department of Commerce. 2016 Top Markets Report Cloud Computing: Brazil Case Study. http://www.trade.gov/topmarkets/pdf/Cloud_Computing_Brazil.pdf, 2016.

[20] U.S. Department of Commerce. 2016 Top Markets Report Cloud Computing: India Case Study. http://www.trade.gov/topmarkets/pdf/Cloud_Computing_India.pdf, 2016.

[21] X. S. Wang, A. Balasubramanian, A. Krishnamurthy, and D. Wetherall. Demystifying page load performance with wprof. In *NSDI 13*.

[22] D. Weller, B. Woodcock, et al. Internet traffic exchange: Market developments and policy challenges. Technical report, 2013.