

# Spanning the globe with Envoy at Stripe

Dylan Carney

[dcarney@stripe.com](mailto:dcarney@stripe.com)

What we built w/ Envoy  
Challenges  
Looking back

stripe

# A bit about Stripe

Stripe is a technology company that builds economic infrastructure for the internet.

Millions of companies in over 120 countries use Stripe to start, run, and scale their businesses.

stripe



stripe

With increased reach  
comes interesting problems

stripe

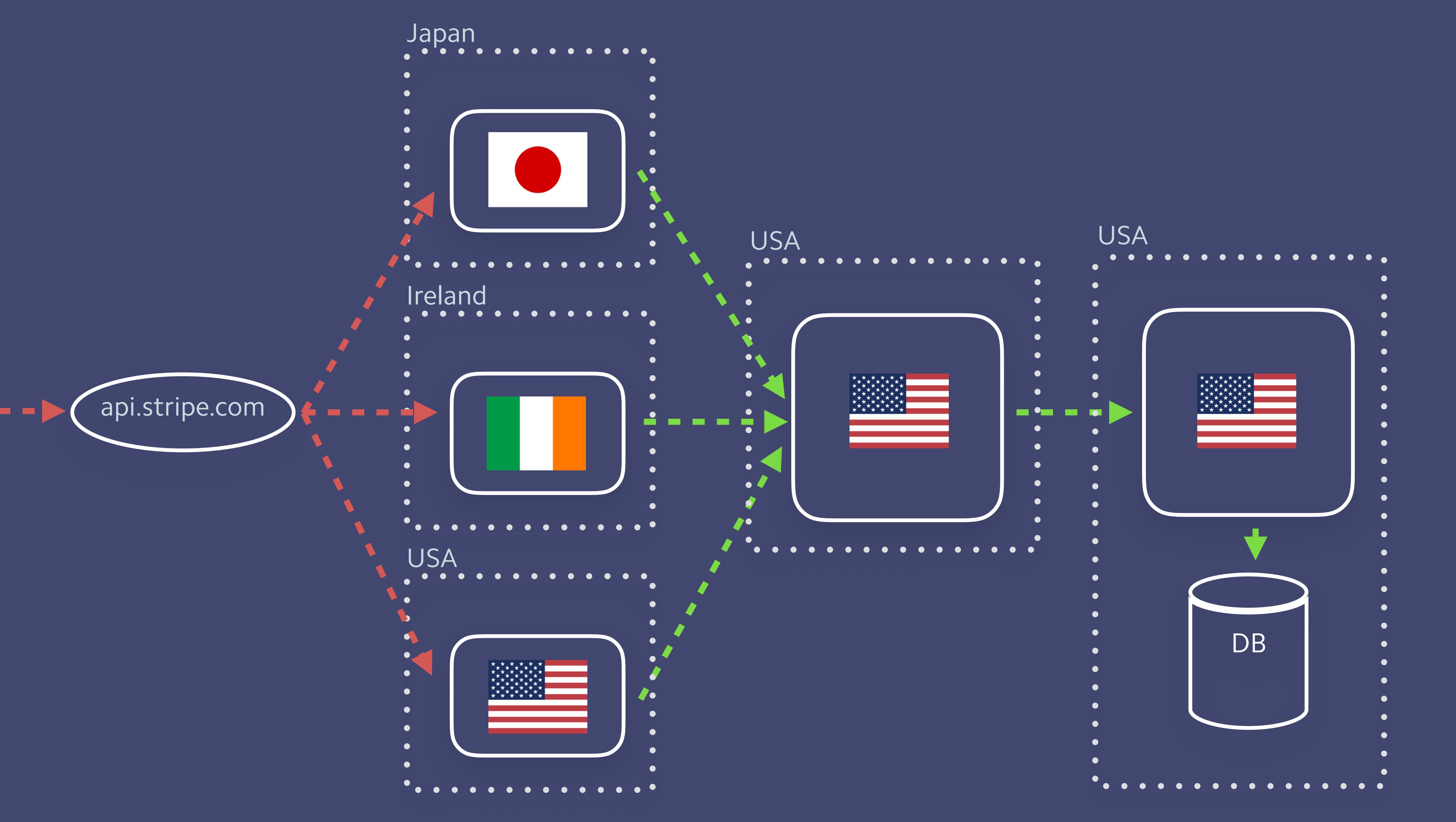
# Seattle to...

San Diego: **36 ms**

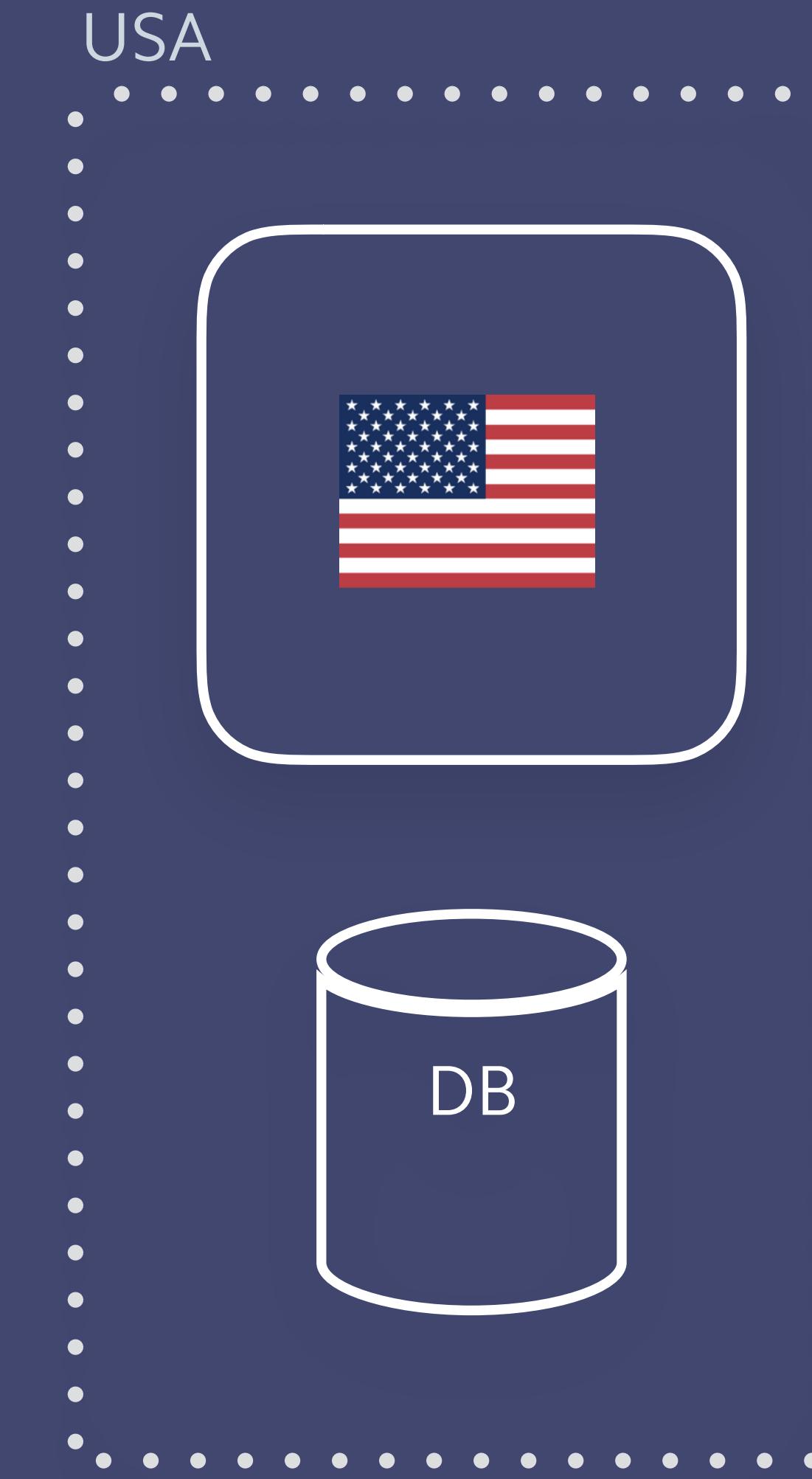
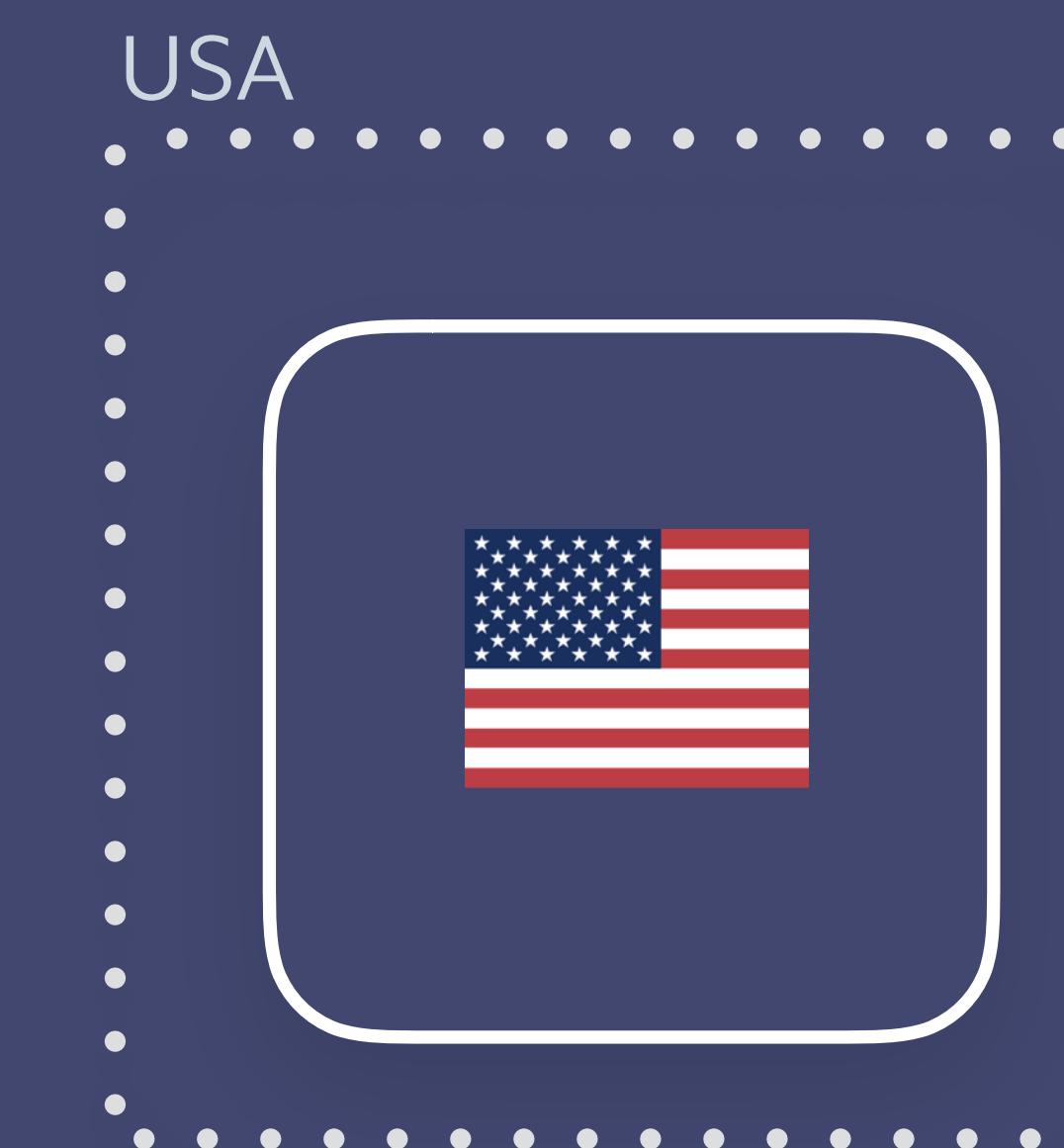
Mexico City: **67 ms**

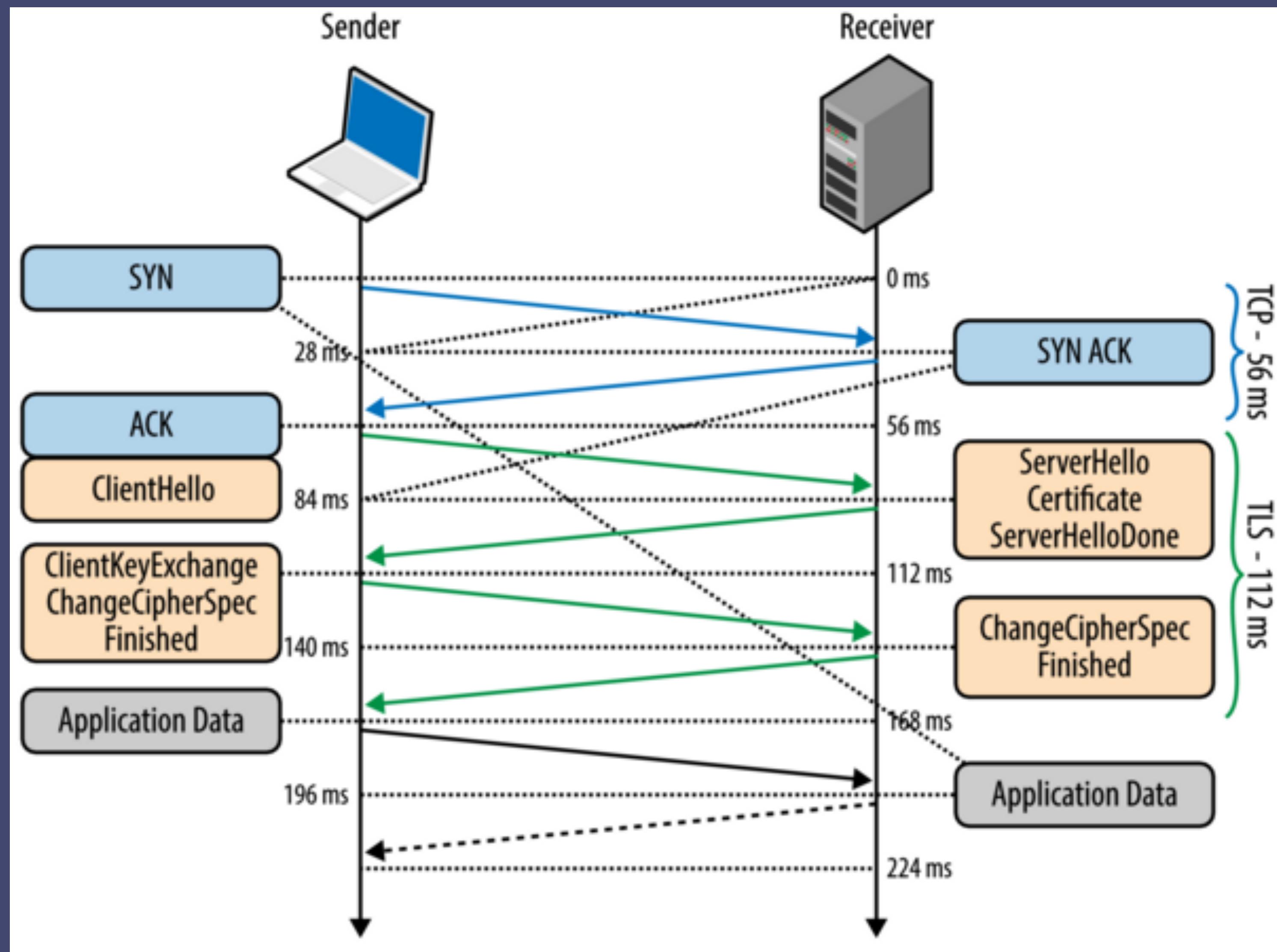
Mumbai: **275 ms**

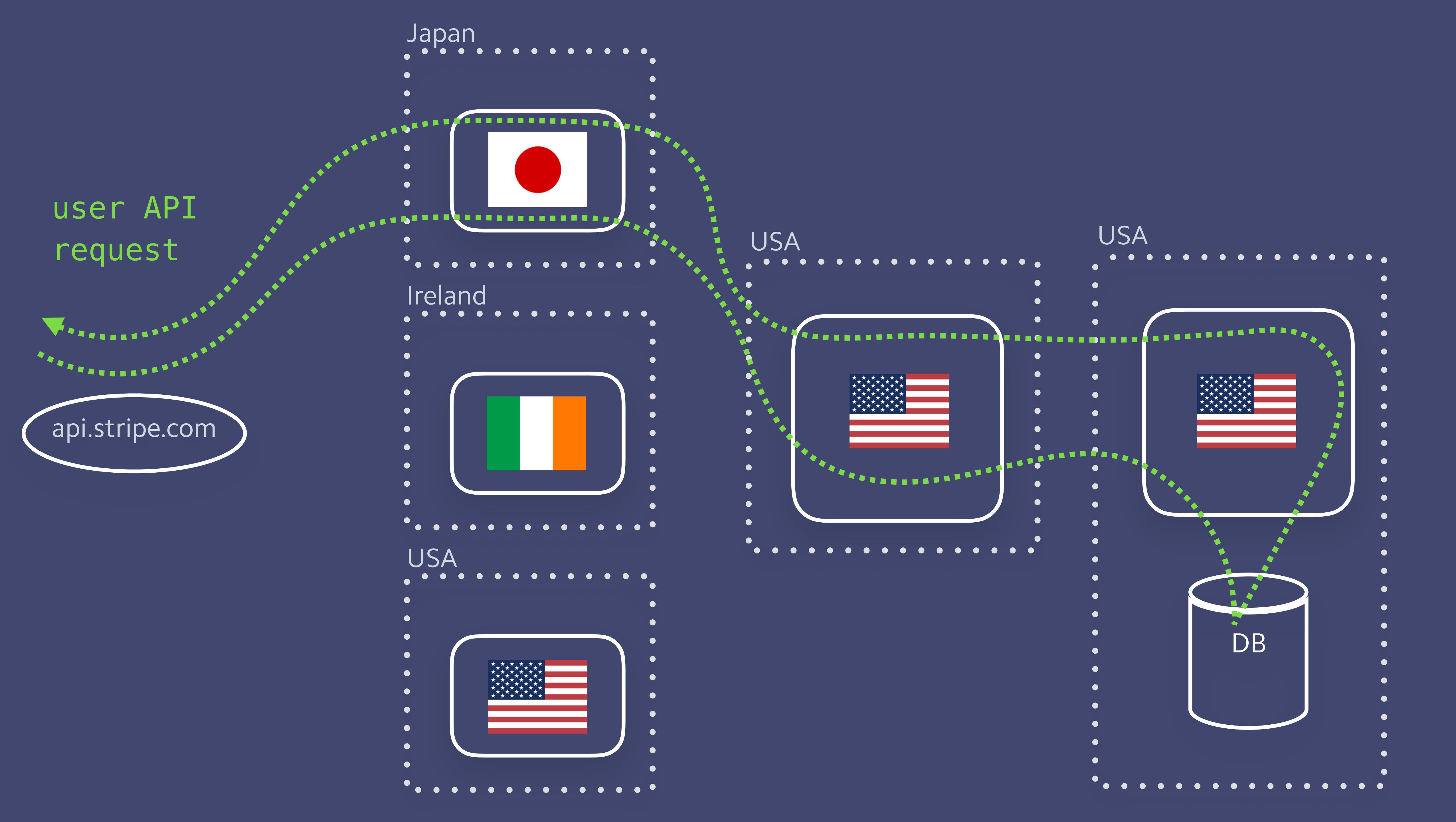
stripe



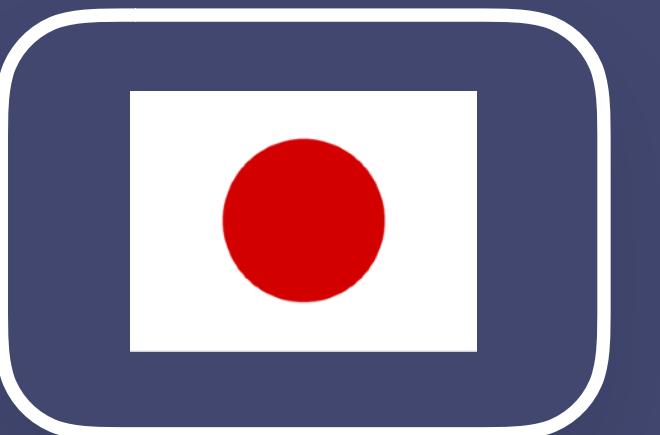
TLS negotiation







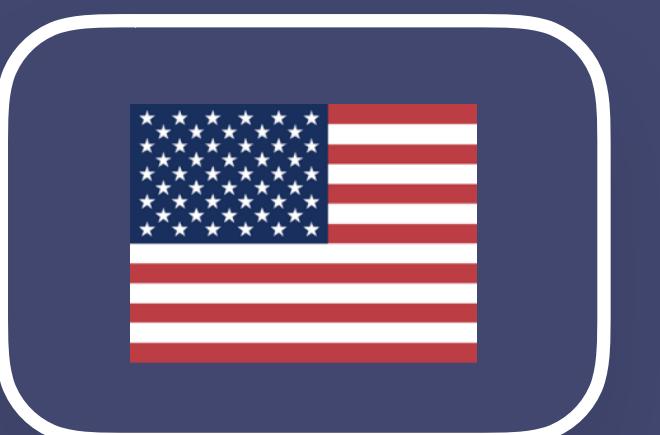
Japan



Ireland



USA



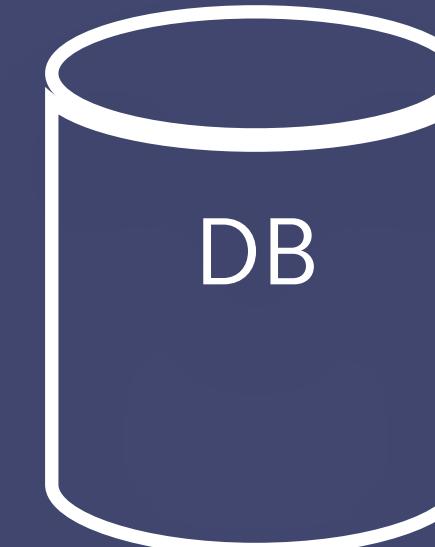
USA



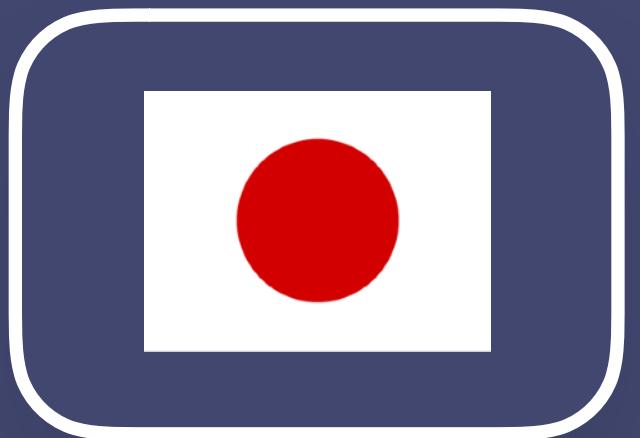
USA



DB



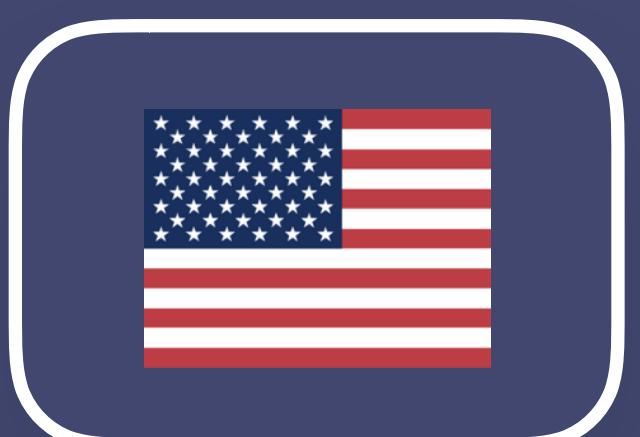
Japan



Ireland



USA



USA

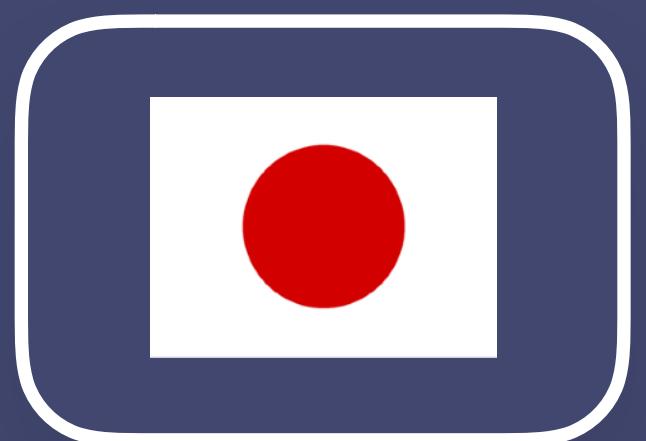


mTLS HTTP/2



stripe

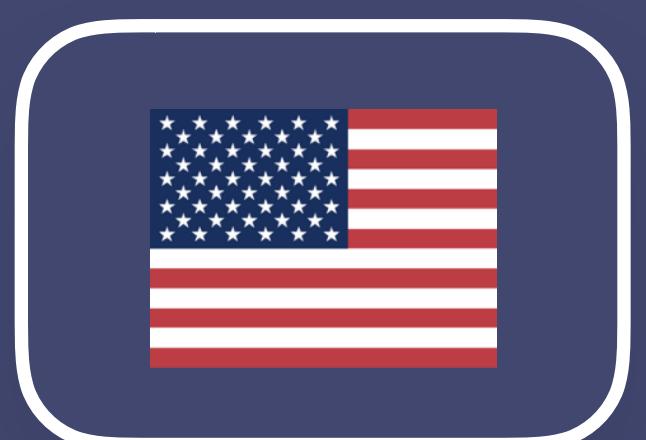
Japan



Ireland



USA

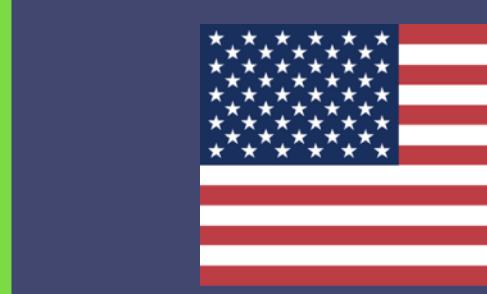


100 %

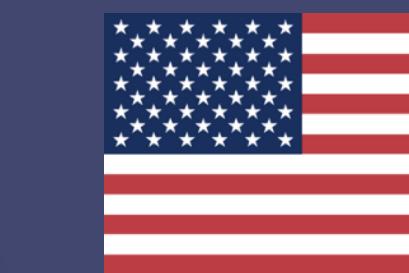
0 %



USA

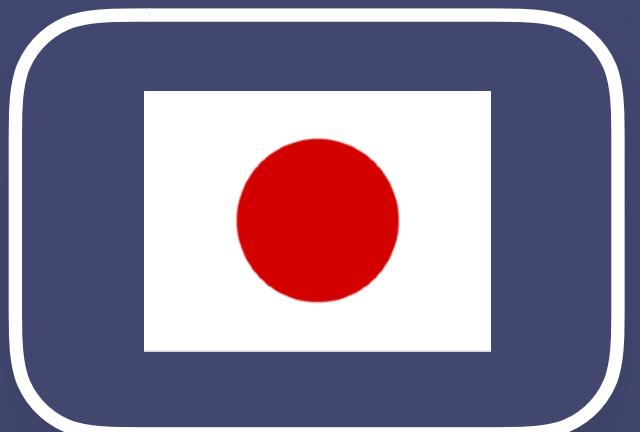


USA



stripe

Japan



Ireland



USA



USA



USA



60 %

40 %

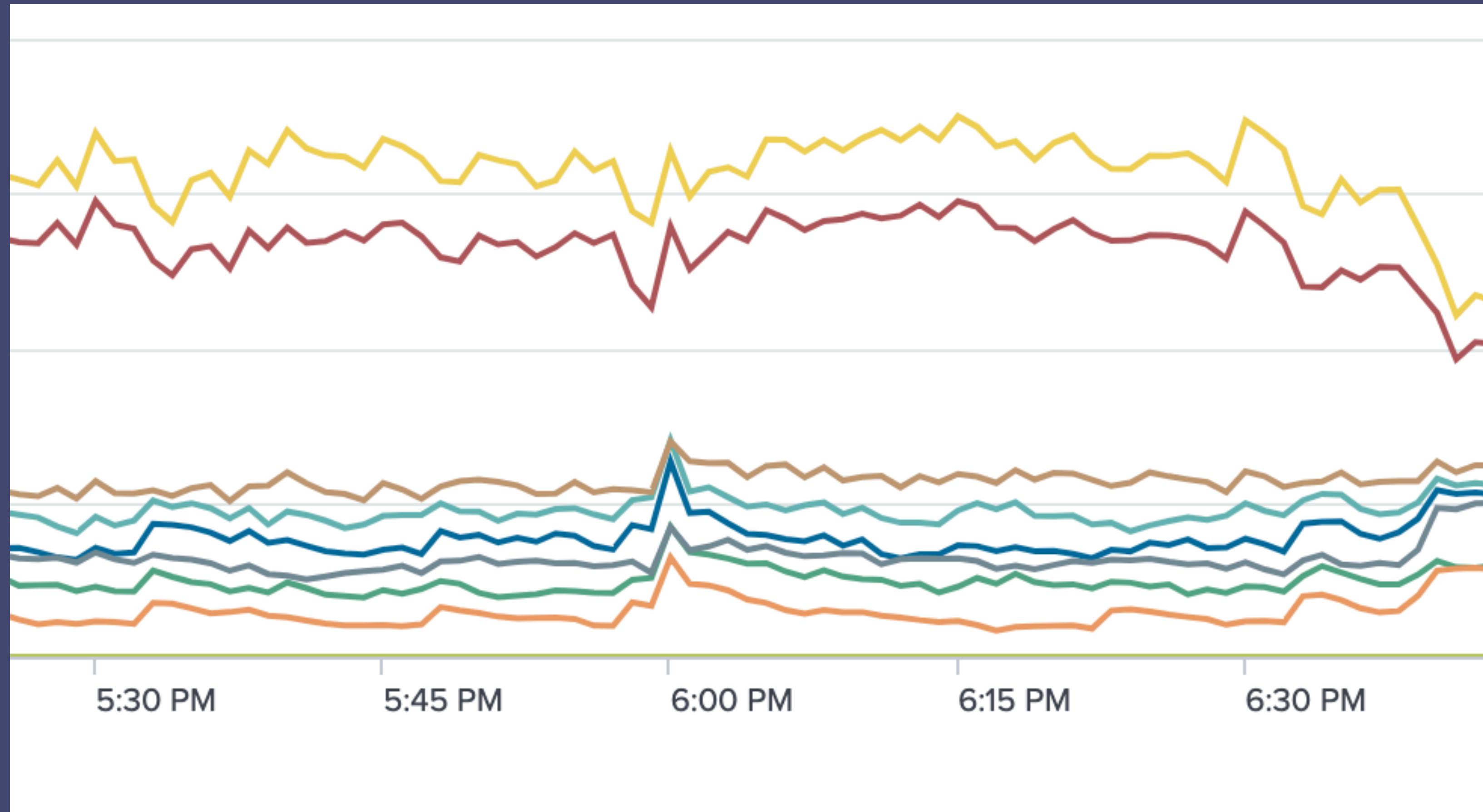


stripe

# Custom control plane

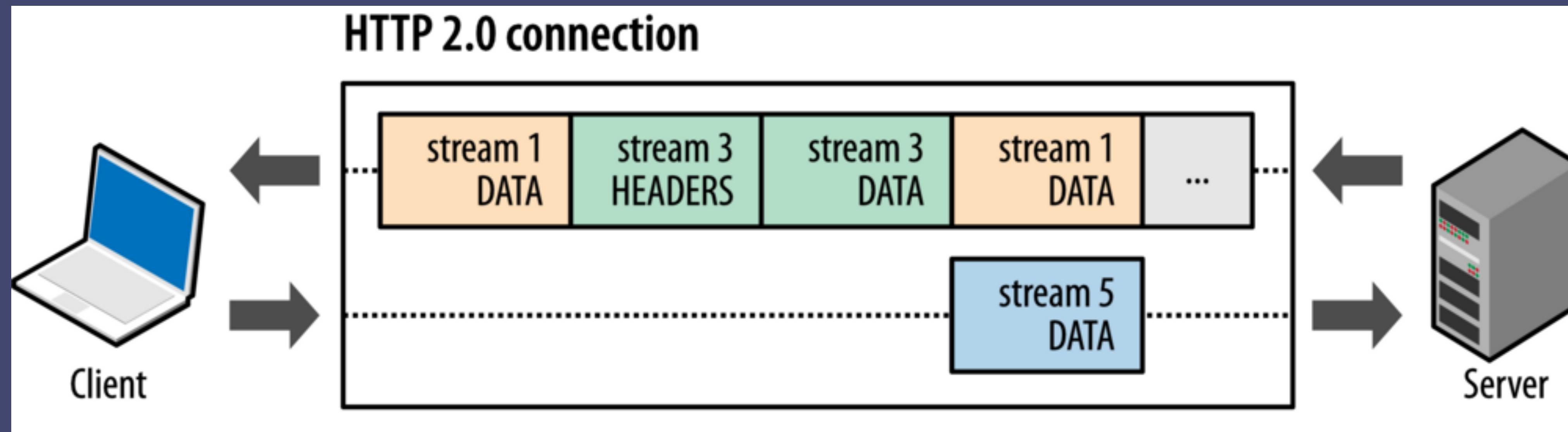
- Custom xDS services give control over routing, ramping
- Powers our blue/green deployments instead of DNS
- Tiered failovers (priority and locality)
- Manual incident remediation
- Routing per customer, per request

# Challenges



stripe

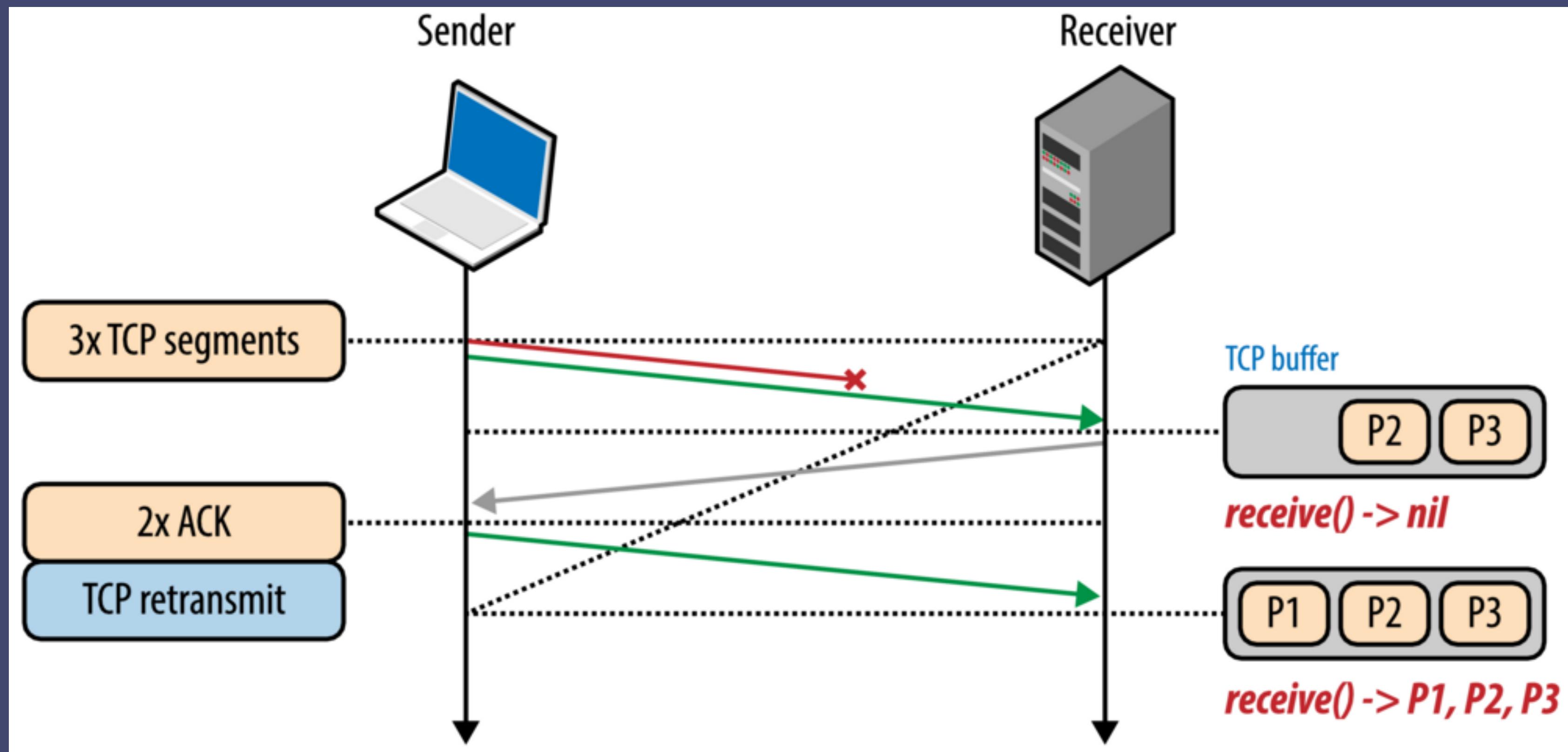
# Challenges



Grigorik, Ilya. "High Performance Browser Networking" O'Reilly 2013. [hbpn.co](http://hbpn.co) Accessed 12-Nov-2019

stripe

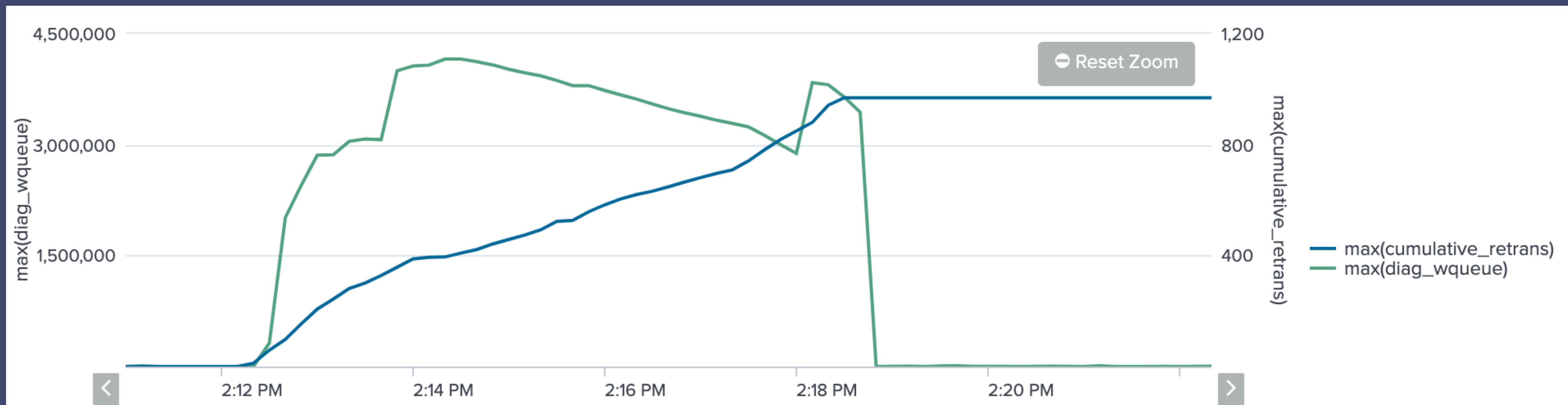
# Challenges



Grigorik, Ilya. "High Performance Browser Networking" O'Reilly 2013. [hbpn.co](http://hbpn.co) Accessed 12-Nov-2019

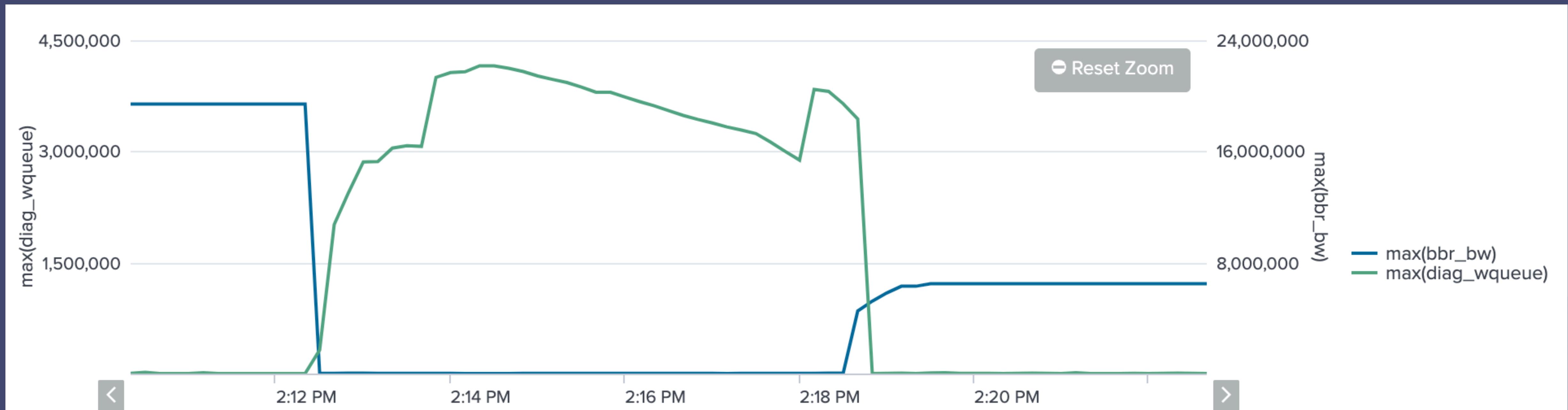
stripe

# Challenges



stripe

# Challenges

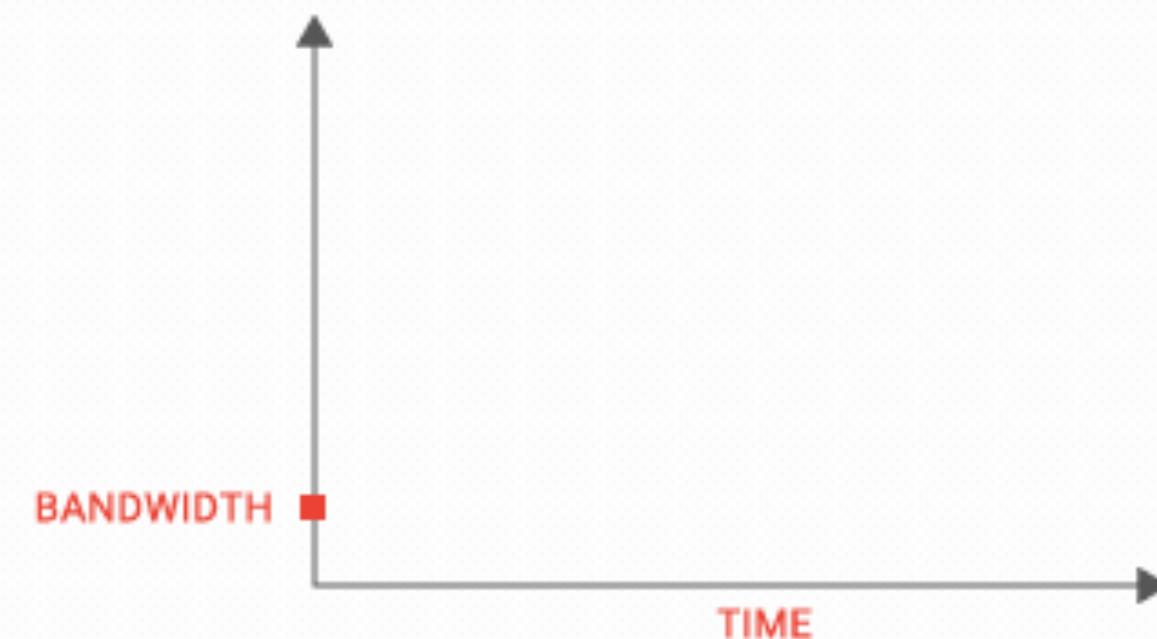


stripe

# Challenges

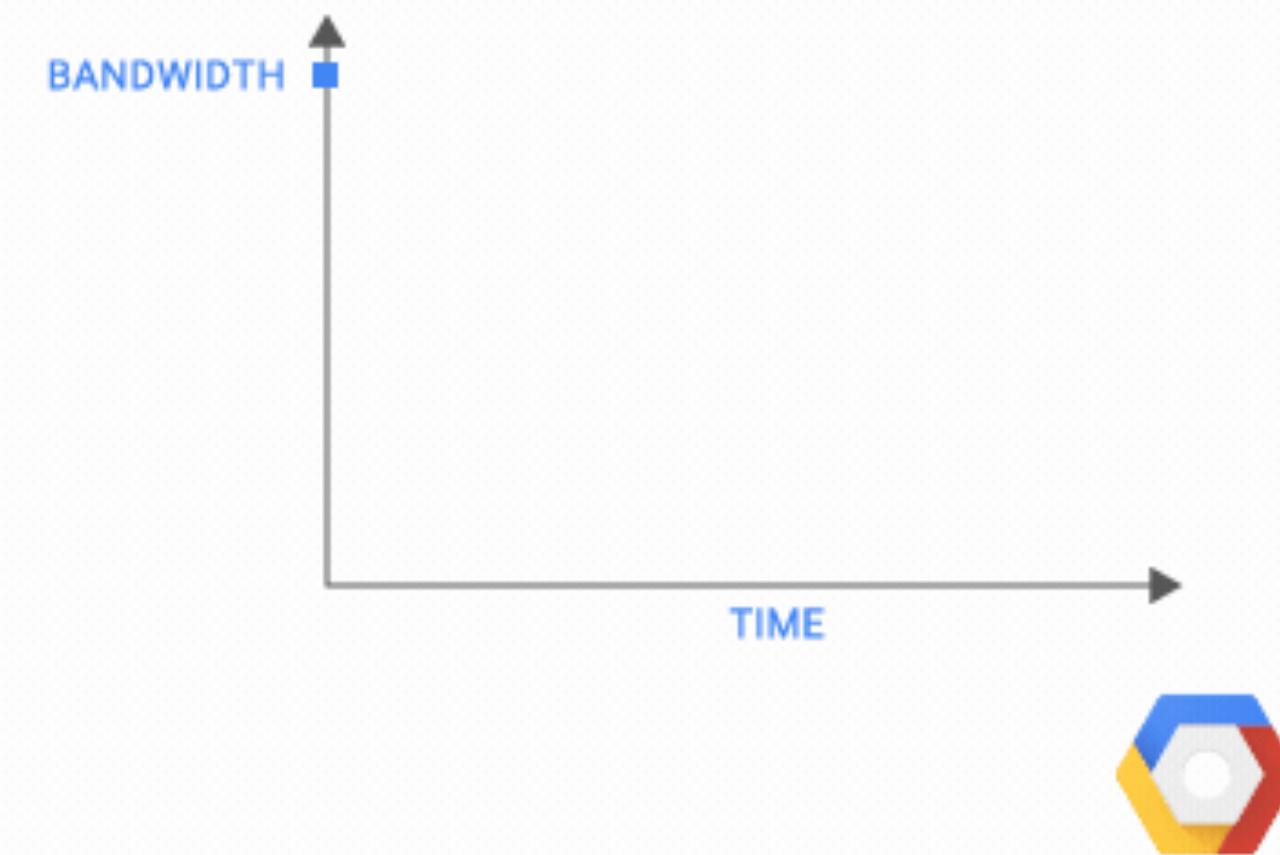
## TCP before BBR

Today's Internet is not moving data as well as it should. TCP sends data at lower bandwidth because the 1980s-era algorithm assumes that packet loss means network congestion.



## TCP BBR

BBR models the network to send as fast as the available bandwidth and is 2700x faster than previous TCPs on a 10Gb, 100ms link with 1% loss. BBR powers google.com, youtube.com, and apps using Google Cloud Platform services.



stripe

# Custom tweaks

- HTTP/2 “head-of-line blocking”
- Increased numbers of upstream TCP connections

[`foobar.stripe.com`,  
`foobar.stripe.com`,  
`fizzbuzz.stripe.com`,  
`fizzbuzz.stripe.com`]

# Custom tweaks

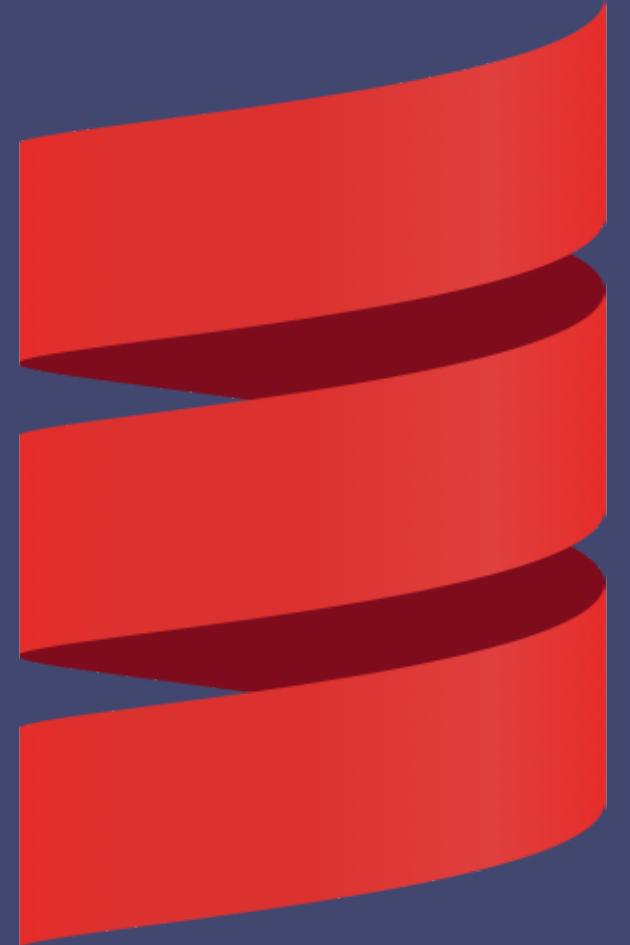
- HTTP/2 “head-of-line blocking”
- Increased numbers of upstream TCP connections
- BBR congestion control, fq packet scheduler
- TCP keepalives on all upstreams

# Pros

- Backhaul connections outlive application process restarts
- Observability, aka all the metrics
- Fixes are propagated to all locations at once, vs upgrading/migrating applications
- Network behavior can be changed w/o application code changes

stripe

# Teams are polyglot



stripe

# Cons

- More moving pieces (a lot more!)
- Timeouts, access logs, etc. all need to be kept in sync with the rest of the stack
- Bugs are outside the domain of your application/developers
- Operability knowledge amongst team members
- Implementations of HTTP protocols differ from the rest of the stack (e.g. nginx, Go, haproxy)



stripe

# Thank you!

dcarney@stripe.com

stripe