



Webinar

How to Plan for Performance and Scale for Multiplayer Games

Speakers



Calvin Scherle

Solutions Engineer @ Cloudflare



Moderator: Alissa McAloon

News Editor and Associate Publisher @ Gamasutra

About Cloudflare

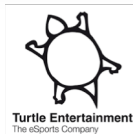
Cloudflare is a leading **security, performance, and reliability** company.

We protect and accelerate any Internet application online without adding hardware, installing software, or changing a line of code.

We have one of the world's largest networks that powers more than 20 million Internet properties, with nearly 10% of the Fortune 1,000 companies using at least one Cloudflare product.



Trusted by:

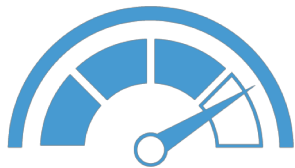


What you will learn today

- How to meet player expectations from the largest online games
- How to plan for different types of online games
- Real-world examples of successes and failures in performance and reliability
- Strategies to minimize overspending on infrastructure
- Deployment strategies for different gaming use cases
- Techniques for offloading processing load from your servers

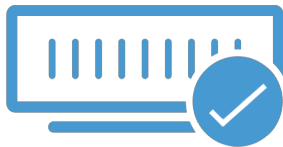


Players want their games to be...



FAST

Lag and latency can ruin a game, and loading times keep players out of their favorite activities



RELIABLE

Players want to play on their own time, which means keeping your game running 24/7 (or close to it)

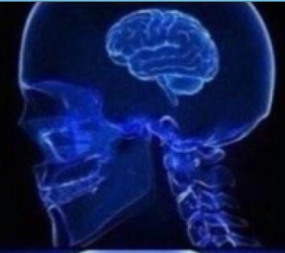


SECURE

Security is paramount for tournaments when money is on the line, while good security practices help protect player data and minimize cost

Choosing an Architecture

**SINGLE SERVER
SINGLE SESSION**



**MULTIPLE SERVERS
MULTIPLE SESSIONS**



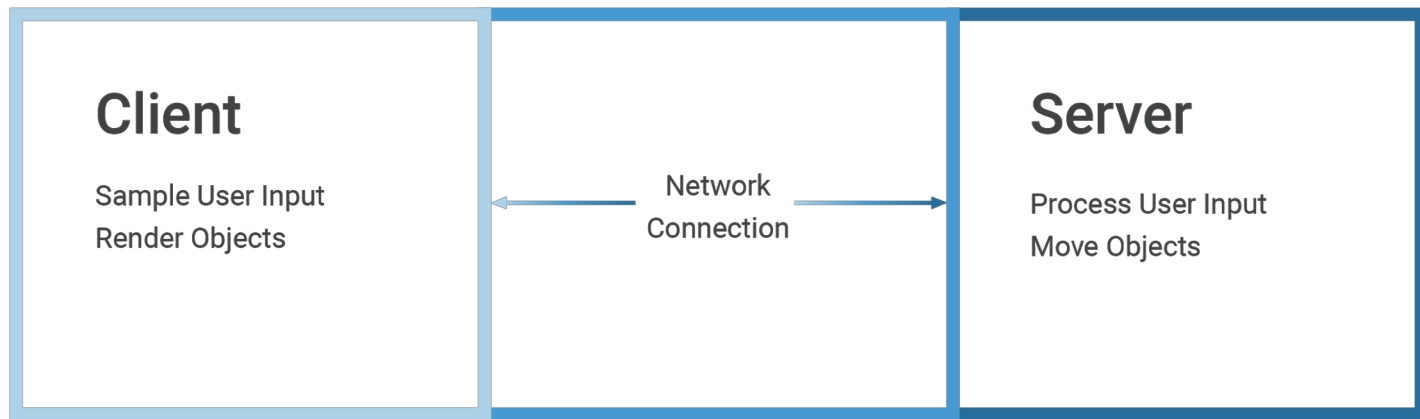
**CLOUD-HOSTED
MULTI-SESSION**



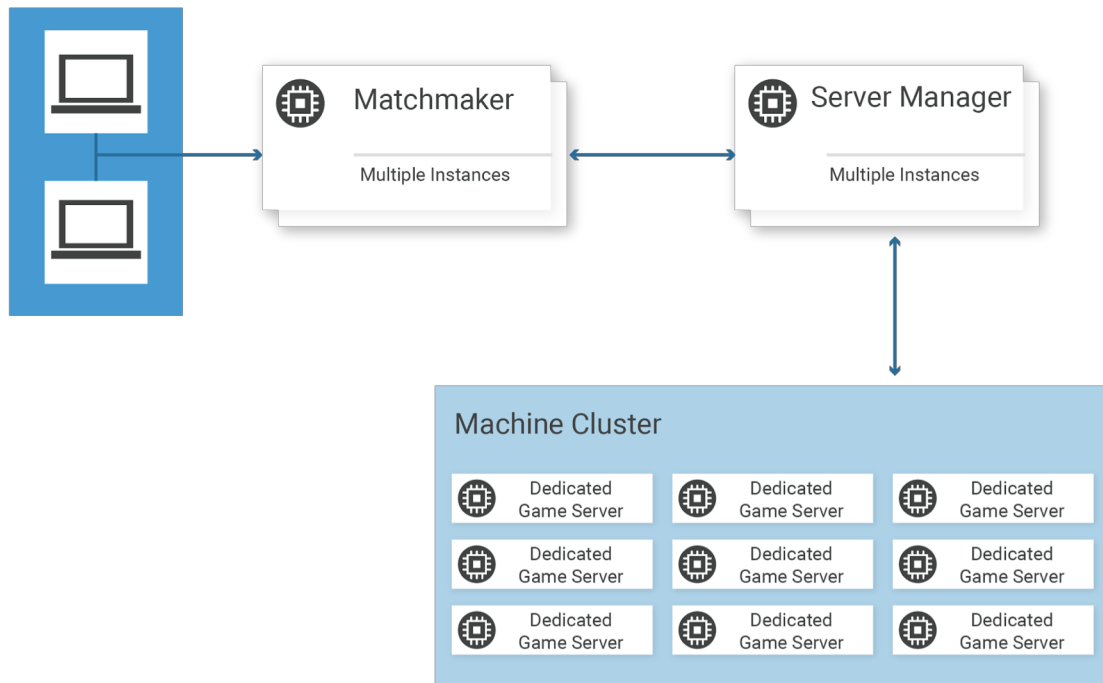
MULTI-CLOUD



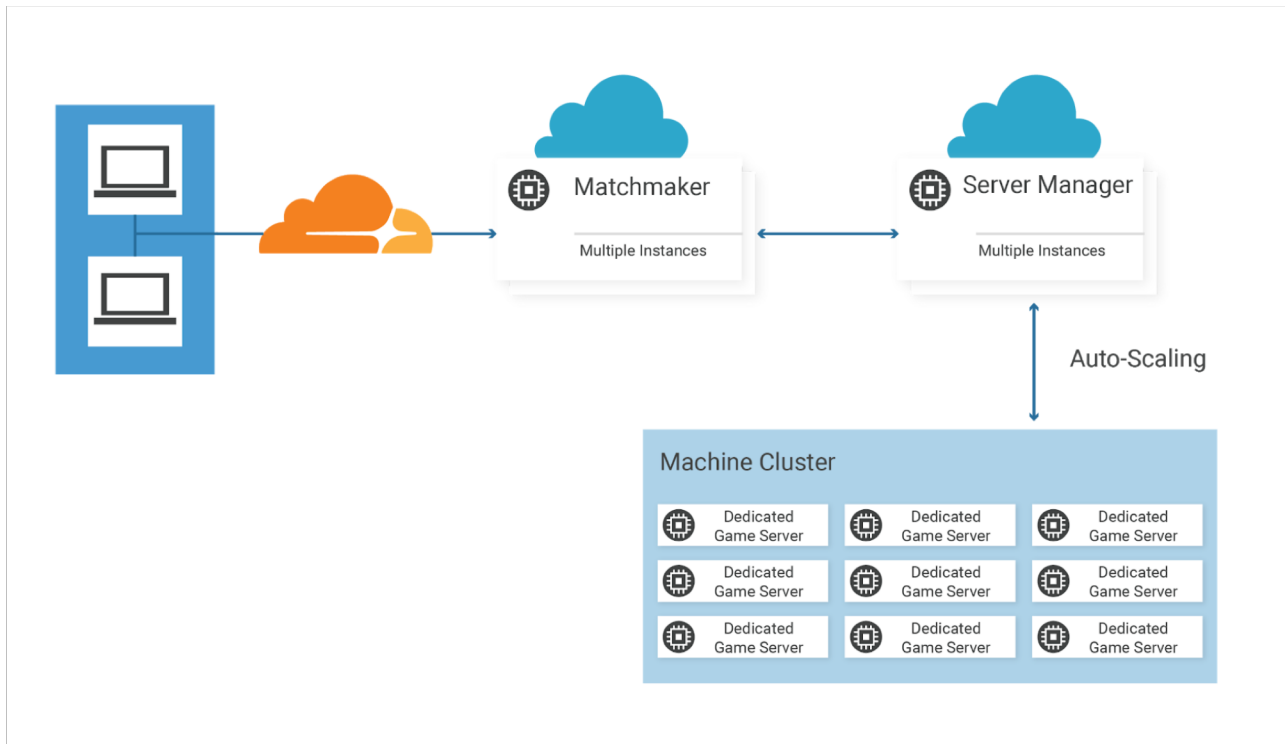
Single Server, Single Session



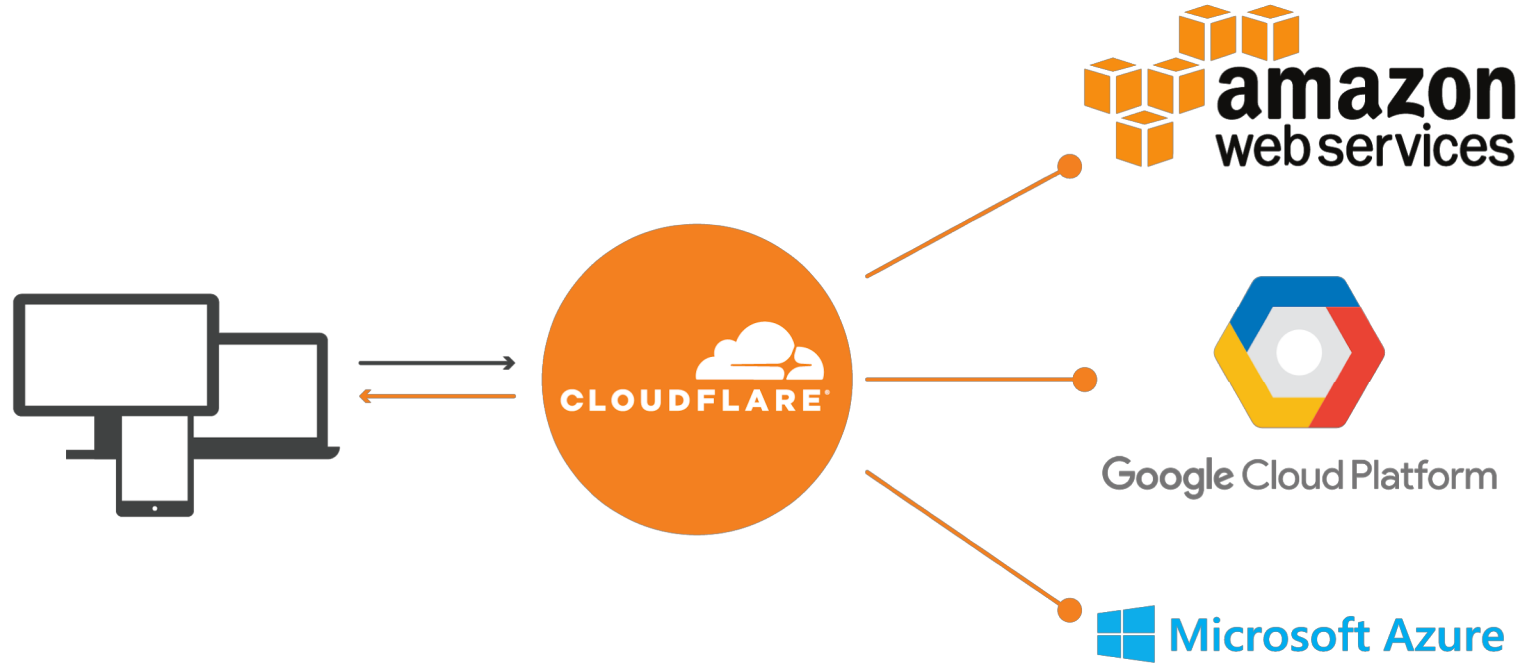
Multi-Server, Multi-Session



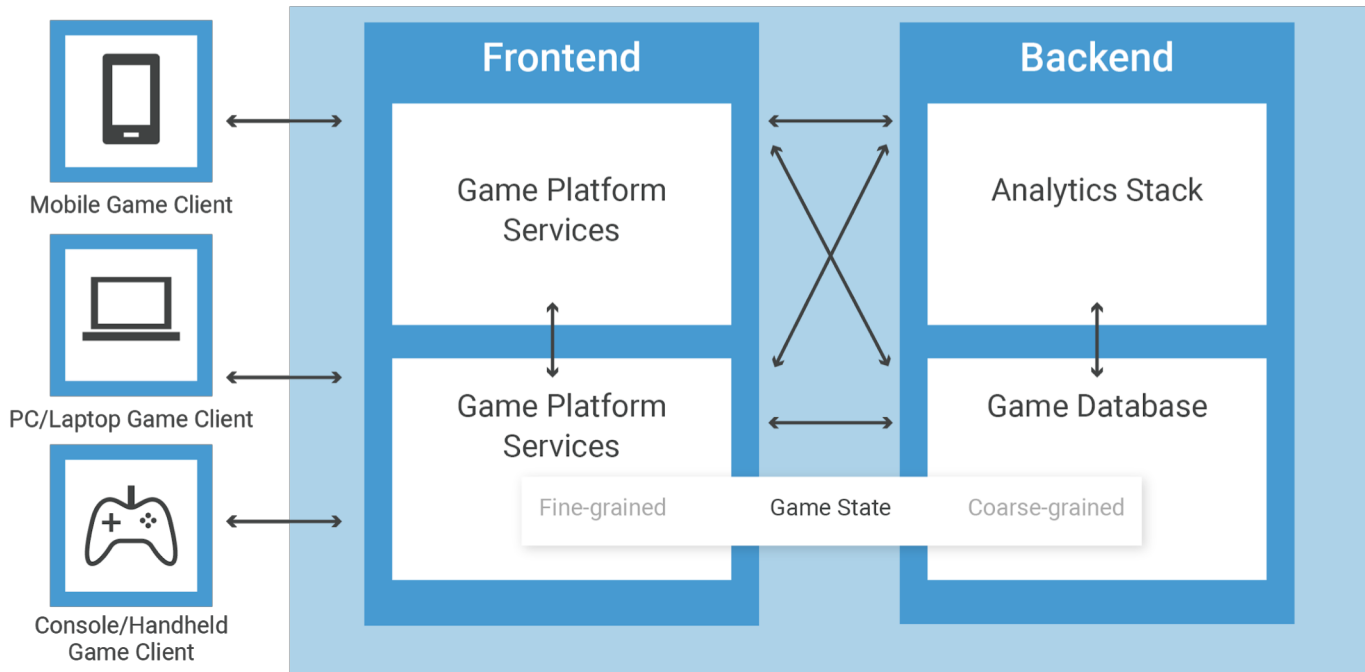
Cloud-Hosted Multi-Session



Multi-Cloud



Adding Platform Services



Optimizing Server Load

"The War on Lag cannot be won because our players can always bring one more ship"

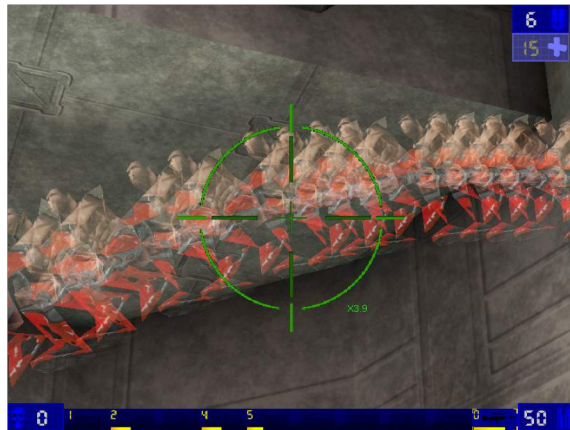
- CCP Veritas, EVE Online Dev-Blog

Tick Rate

How often the server updates the game state

Example:

- 10 clients connected - Need to process each client per tick
- 64Hz tick rate - Game updated 64 times a second
- 640 ops/sec required



Note: Basic example, may require several ops per package of client data

Optimization Goals

Real-time game server optimization typically has two goals in mind:

1. Reduce ops/tick **required**
2. Increase ops/tick **available**

"In order to bring things down to the level where the server can keep up, there's painfully few options. They boil down to two: reduce load or increase processing capacity."

Source: <https://www.eveonline.com/article/introducing-time-dilation-tidi>

Optimization Techniques

Reduce required operations

Lower tick rate

Statically - Reduce tick rate of some or all servers

Partially - Reduce resolution of certain processes (e.g. physics simulation)

Dynamically - EVE Online “time dilation”

Reduce garbage input

Optimize game client (only send relevant data)

Filter traffic - Reverse Proxy

Increase available operations

Upgrade systems

Increase processing power of individual units

Increase scale

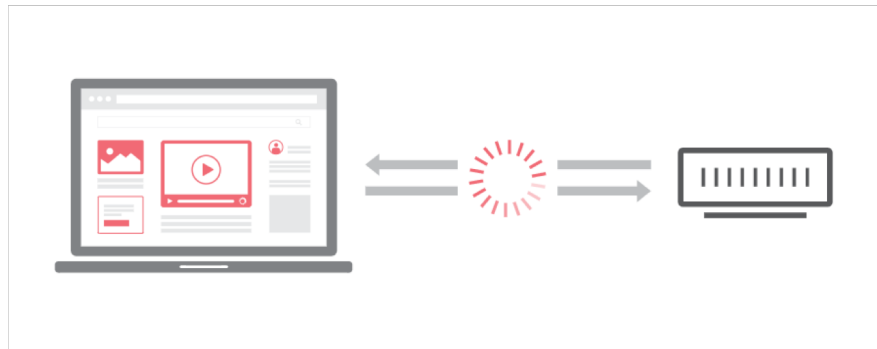
Add more servers to reduce infra load

Scale up pods/LBs

Reducing Latency

Latency Reduction

- Protocol Optimization
- Edge termination
- Geo-steering



*Pseudo latency reduction (In-game techniques, extrapolation)

Protocols for Gaming

TCP

Reliable, Slow

In-game trades, matchmaking, complex consistent game states

UDP

Fast, Inconsistent

Best for competitive games, real-time updates, when you only want the latest and speed is a must

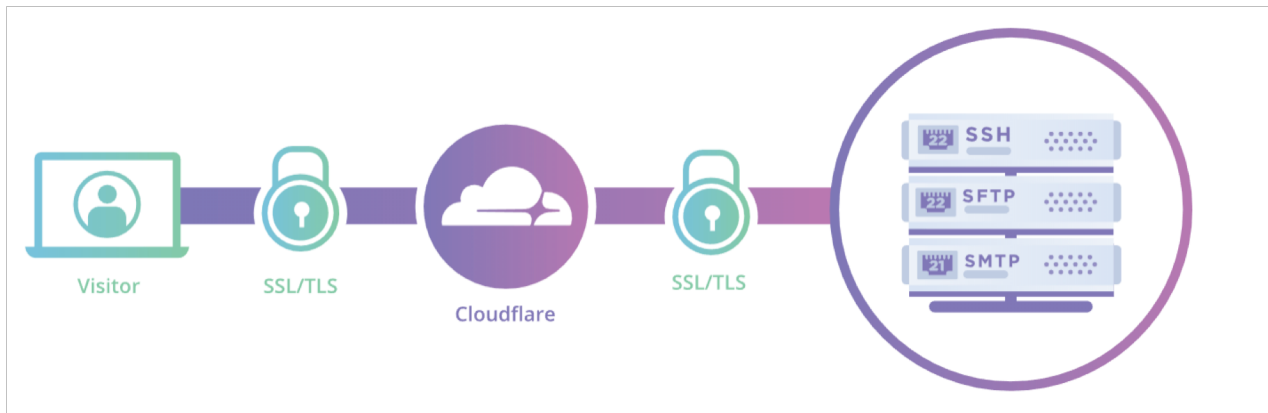
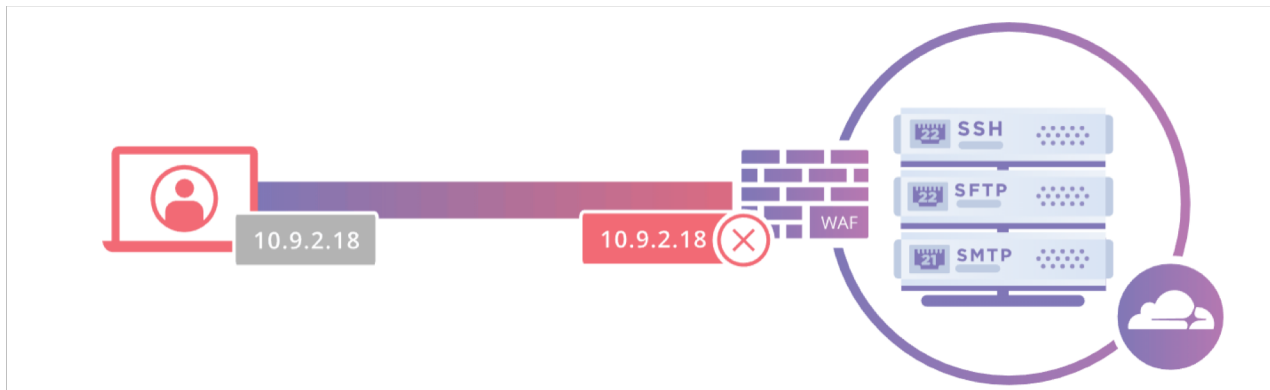
HTTP

(subset of TCP)

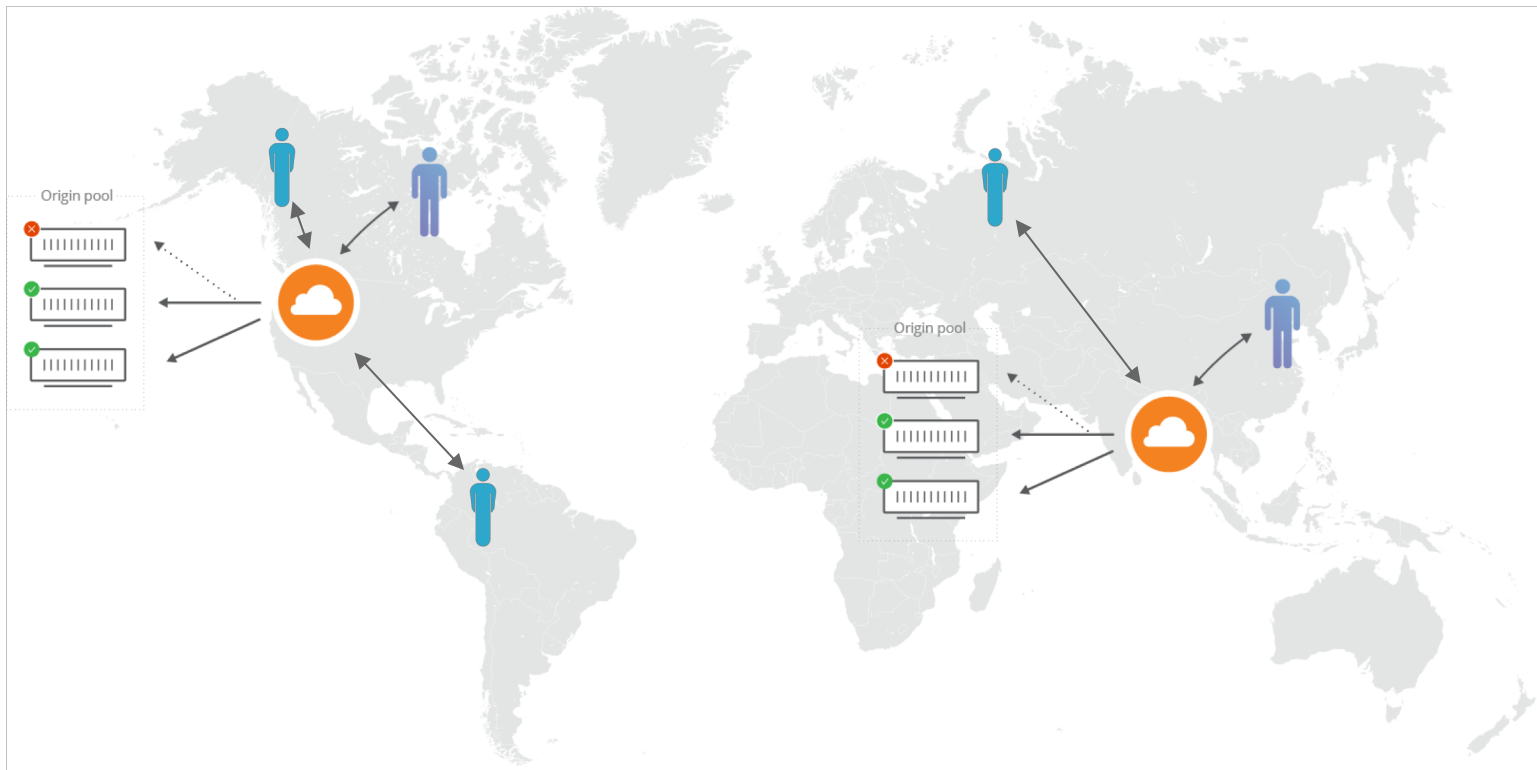
Compatible, Secure

Primarily for platform services, APIs, web-based games

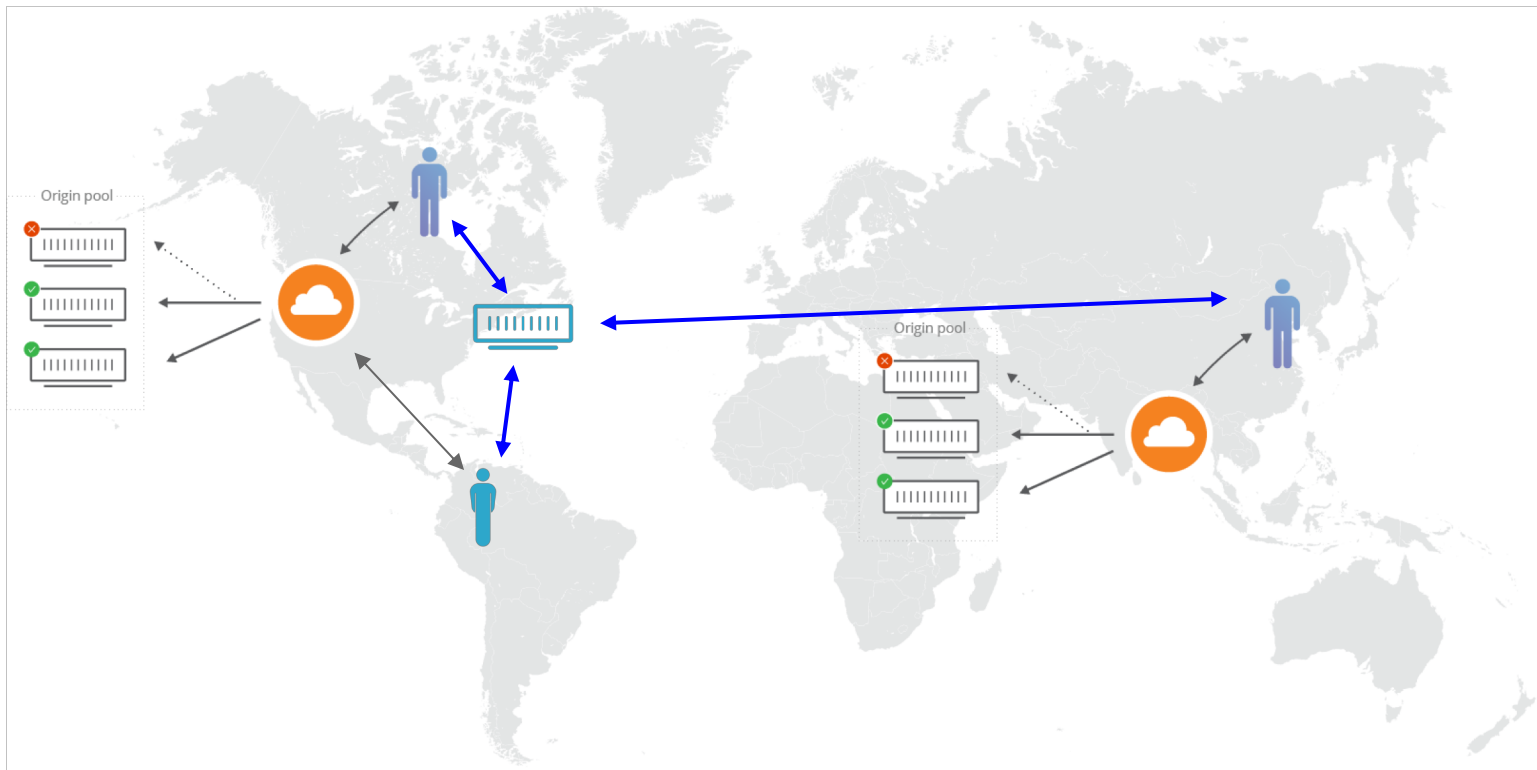
Edge SSL Termination



Geo-Steering



Geo-Steering w/ Centralized Matchmaking



The Game Must Go On
(When Things Go Wrong)

Enhancing Reliability

Peak traffic mitigation strategies:

- Queueing
- Staged maintenance/failover
- Elastic scaling
- (Semi) permanent scaling

	Pros	Cons
Queuing	<ul style="list-style-type: none"> • Never go over capacity • Buys time to scale up/down to meet demand • Measure queues for real time concurrent player data • Avoid buying capacity to meet temporary demand 	<ul style="list-style-type: none"> • Players must wait, from minutes to hours • Potential party breakups • Implementation effort and cost for queueing logic/infrastructure
Staged Maintenance/Failover	<ul style="list-style-type: none"> • No downtime • Low average cost for usage-based service 	<ul style="list-style-type: none"> • Requires complex infrastructure (multi-cloud and/or multi-cluster) • High cost at peak
Elastic Scaling	<ul style="list-style-type: none"> • Cost effective • Dynamically meet demand 	<ul style="list-style-type: none"> • Requires scaling infrastructure • Sudden spikes can overwhelm infrastructure
(Semi) Permanent Scaling	<ul style="list-style-type: none"> • Most reliable if demand is met 	<ul style="list-style-type: none"> • May incur downtime • Potential wasted cost

War Stories

Destiny 2: Shadowkeep

Bungie simultaneously released a major expansion for Destiny 2, migrated the game from Battle.net to Steam, and made the game free to play for all new players on October 1, 2019

Difficulties:

- 24-hour pre-release maintenance
- Queues lasting upwards of 1 hour
- 2-hour emergency maintenance
- ~1 month of no new players pre-migration





Posted by u/trick_m0nkey 21 days ago 🏆 🥇 🥈 3

1.9k

What Bungie has accomplished is impressive (A Sysadmin Perspective)

Discussion

I want to point out that whoever manages the Bungie infrastructure, at some point some months ago they had directives that looked like this:

1. We are going to give the option to migrate users from 3 different platforms (Xbox, PS, Battle.net) to a single unified platform (Steam)
2. We are then going to decommission all primary characters off Battle.net and migrate them over to Steam within a 24 hour window
3. Simultaneously during this migration, we're going to deploy a major revision of the product to production
4. At the end of the upgrade and migration window, we will be expecting an exponential and explosive growth of new registered users, so we need to make sure everything scales too.
5. All hands on deck for the migration to squash bugs live.

If anyone here has ever experienced something like this before in their professional careers, they gotta know how much work this is, and how much teamwork and coordination is required. It's amazing to me that today everything seems to be working just fine. It's not uncommon for this kind of thing to knock orgs offline for a week. Sometimes to the point to where you gotta roll everything back and try again.

Whoever manages the Bungie Sysadmin/Devops team, hats off to you and your team. I hope you all get that PTO approved and you get a chance to sleep.

Also, if you could write a whitepaper about how it all had to happen, I would *love* to read it. I bet it would make for a great learning tool on how to do such a large scale migration.

EDIT: Appreciate the gold. Go hug a sysadmin today, they suffer in silence.

Apex Legends

Respawn Entertainment surprise launched Apex Legends on February 4, 2019 to critical acclaim and massive player interest, using a flexible multi-cloud infrastructure.

Successes:

- Over one million players on second day of launch with 60 players per session
- No downtime for updates due to flexible infrastructure
- Netcode developed by a team of three people



Notable stories



- Niantic delays Pokemon: Go global rollout after players experience **failed connections and massive lag**
- Valve hosts The International 2015 - the world's largest Dota 2 tournament with a \$18 million USD prize pool that year. Matches are **suspended for several hours due to DDoS attacks against Valve servers**
- World of Warcraft Classic releases in 2019 to **server queue times upwards of 2 hours**

Questions?

Contact us: cfmarketing@cloudflare.com

Learn more: cloudflare.com/gaming