

The two things most important to gamers (and thus developers) are real-time, high-speed play, and regular releases of new content, which require rapid development cycles. Security is often an afterthought, but it shouldn't be, given the growth of malware-as-a-service and consequent data breaches in many popular gaming services.

***Cloudflare Workers for Gaming is especially positioned to support rapid development cycles, while seamlessly integrating with Cloudflare's existing offerings, which increase the speed and security of online gaming applications.***

### Proposed features and integrations:

1. Promote the **Workers API**, **Wrangler CLI**, and **workers.dev playground** as the ideal environment for developers to to rapidly prototype, iterate, and release new games.
2. Seamless integration + support for **Twitch** streaming to increase adoption of Cloudflare-developed games and better compete with AWS gaming offering.
3. Improved integration with **WebSockets** to enable the highest possible speed within real-time games:
  - Enable Workers to act as client/server endpoints for WebSocket sessions (currently, Workers only support proxying Websockets).

### Market Exploration

#### Internal Research:

Investigate how existing game development customers (i.e. Nodecraft, Discord, Hypixel) currently use our products.

- Is there more potential for integration of our services?
- Are customers already using our other products as a substitute for this new service? Would launching WfG cannibalize our existing products? If so, quantify the trade off.

#### External Research:

Highlight WfG's unique value proposition when compared to competitors like AWS and Akamai.

- While AWS is an industry standard for server infrastructure, WfG can provide a *serverless* infrastructure for the most forward-thinking developers.
- Explore areas where competitors lag in terms of security offerings and load balancing.

*Preliminary Hypothesis:* Game developers need products that allow for faster development cycles and boost security, without compromising frame rate.

# WfG ROADMAP

Week 1-4

Exploratory research, build team, confirm project feasibility and timeline, adjusting goals as needed.

Week 3-7, ongoing

Development + continuous iteration: monitor active usership, security concerns, and success of Twitch integration.

Week 8-12, ongoing

Initial prototype + testing: test on small group of new & existing users. A/B test with existing versions of Workers and Websockets.

Week 20

Grand launch at Cloudflare Connect conference. Increase marketing + promotion of WfG with continued success.

## Goals:

## How to measure success?

*Gain traction with WfG and the Cloudflare brand in the game development industry*

- # free and Unlimited users should steadily increase, along with traffic on gaming domains
- Look for increasing conversion to paid option from larger developers. If conversion is weak, consider adjusting payment incentives (features) or pricing policy
- Marketing return on investment should increase over time

*Compete with AWS gaming offering*

- Monitor Twitch activity from WfG games
- Track conversion from AWS (and other competitors) to CfW

*Increase dependency on Cloudflare's entire product portfolio*

- Track which other Cloudflare services WfG customers are using, and identify opportunities for potential integrations

*Maintain + strengthen Cloudflare's reputation as the Internet's fastest, most secure network*

- Customer satisfaction surveys, specifically in regard to speed and security
- Measure incoming attempted data breaches and DDOS attacks, relative to latency, especially for WfG-Websocket users

## Risks:

*Lack of ownership when building out product integrations*

*Cannibalization of Cloudflare's existing products*

## Mitigants:

Build team by transitioning Websockets, etc. engineers onto WfG, leveraging their familiarity with Websockets while allowing them to fully focus on WfG.

Continuously monitor WfG customer acquisition: WfG should widen the funnel for Cloudflare's other product lines. If cannibalization is a major issue, it's a sign that Cloudflare's product portfolio needs to be more broadly positioned to all types of developers.