From Lab to Live.

How Riot ships production-grade AI on AWS

Wesley Kerr
Head of Tech Research · Riot Games



AGENDA

Building and shipping learned bots



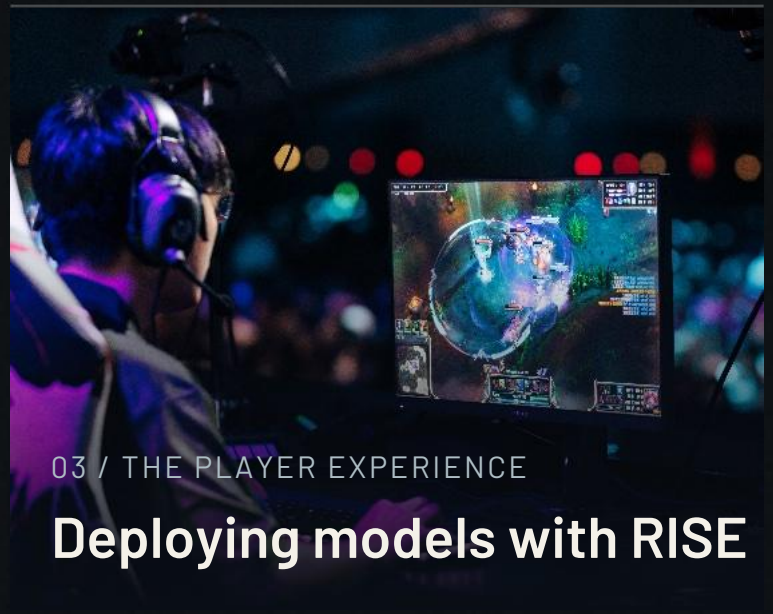
01 / THE HYPE

Learning to play



02 / THE REALITY

Getting ready for launch

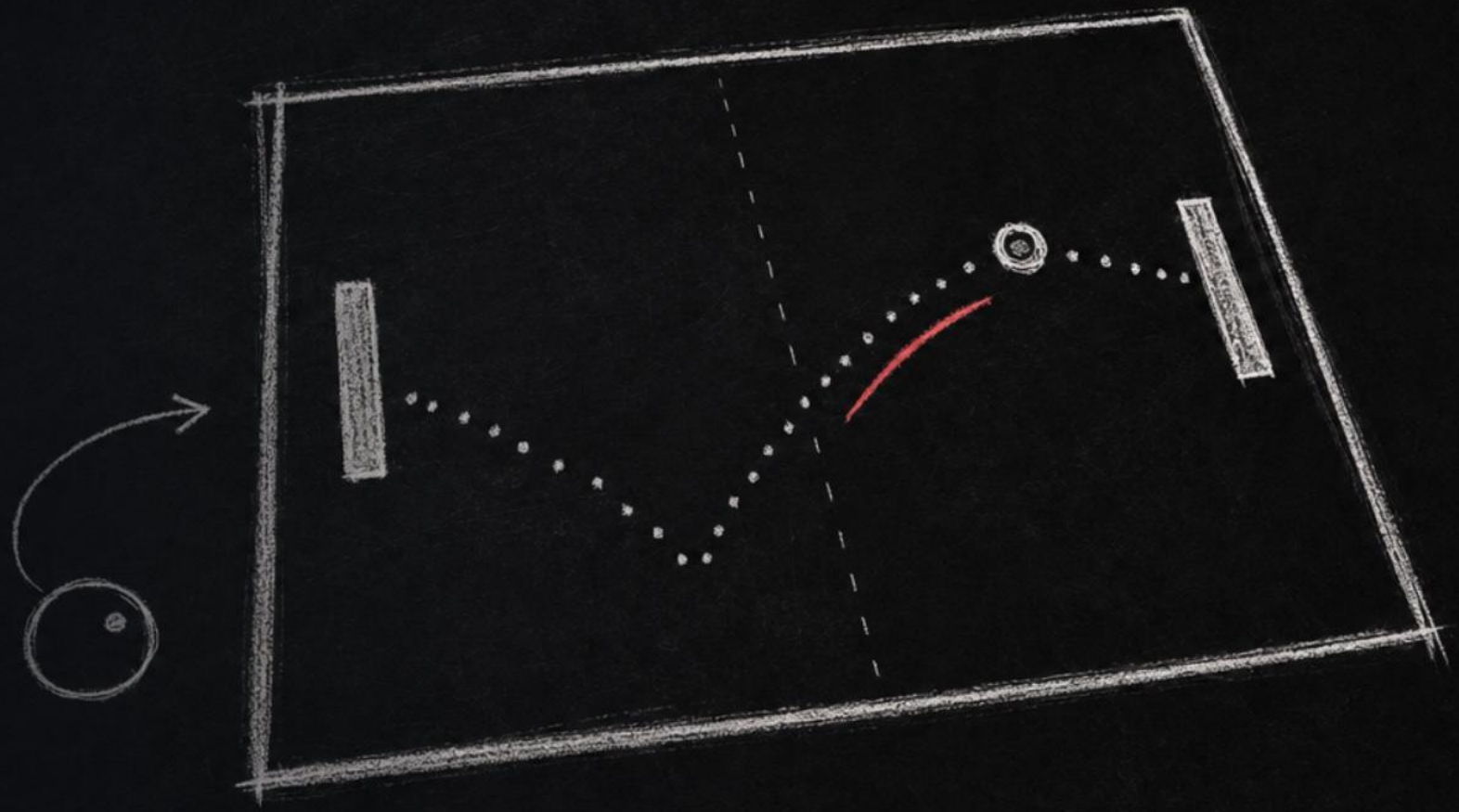


03 / THE PLAYER EXPERIENCE

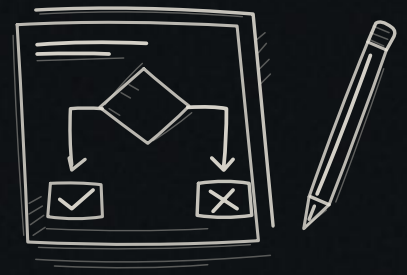
Deploying models with RISE

Act 01

The Hype



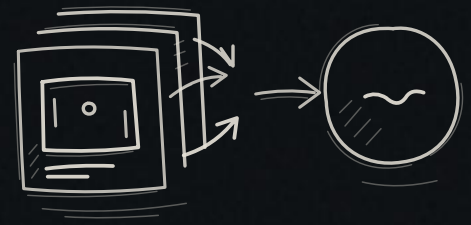
Three ways to build game AI



Designer-programmed

Explicit rules
and behaviors.

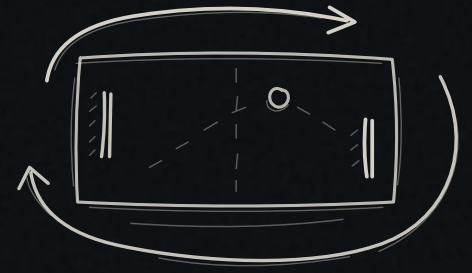
Scales with human effort



Learn from players

Examples from
player matches.

Scales with player data

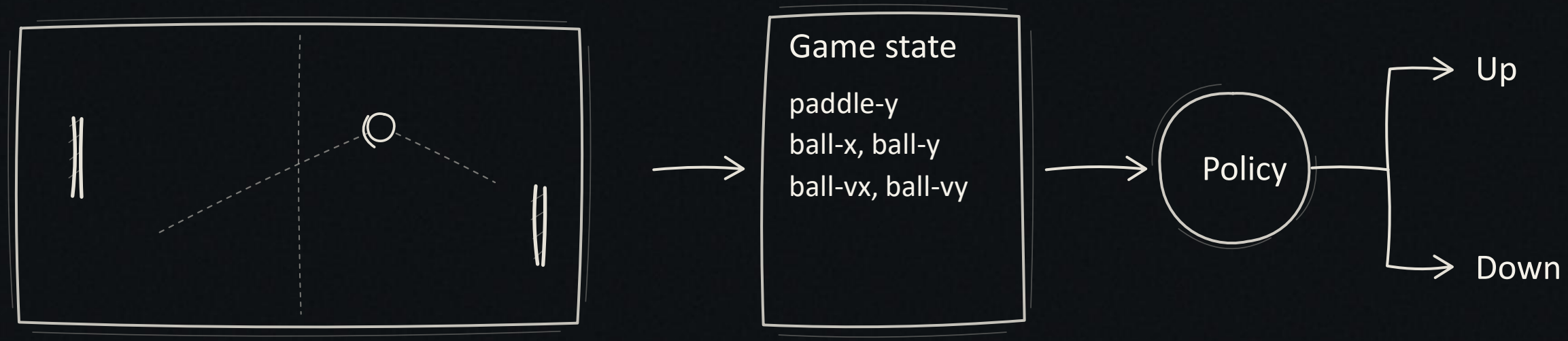


Learn from scratch

Gameplay and
reward feedback.

Scales with compute

Concrete Example: Pong

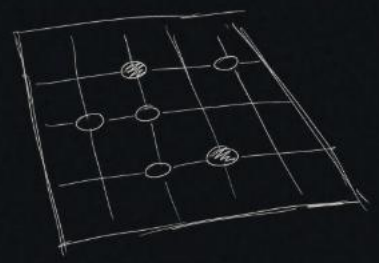


RESEARCH MILESTONES

Progress in learning to play games



Atari
DeepMind
Arcade control



AlphaGo
DeepMind
Strategic search



OpenAI Five
OpenAI · Dota 2
Team coordination



AlphaStar
DeepMind · StarCraft II
Real-time strategy



CICERO
Meta · Diplomacy
Language + strategy

Beyond Bots

Testing

Behavior Cloning
Train from human replays

Fuzz Testing
Auto-run edge cases with random input

Automated QA Tests
Bots act as control group

Artificial Opponents

Drop-In Players
Take over when players disconnect

Practice Matches
Chill vs AI at any skill level

Onboarding AI
Provide a gentle path to learn a complex game

Meta Analysis

Win rate Dashboards
Run thousands of games overnight

Experimentation
How does one change affect the meta?

Viable Strategy Discovery
Test new gameplay opportunities

Coach

Player Insights
Help players climb ranks

Practice Focus
What should I be working on?

Personalized Feedback
Your replay, your breakdown

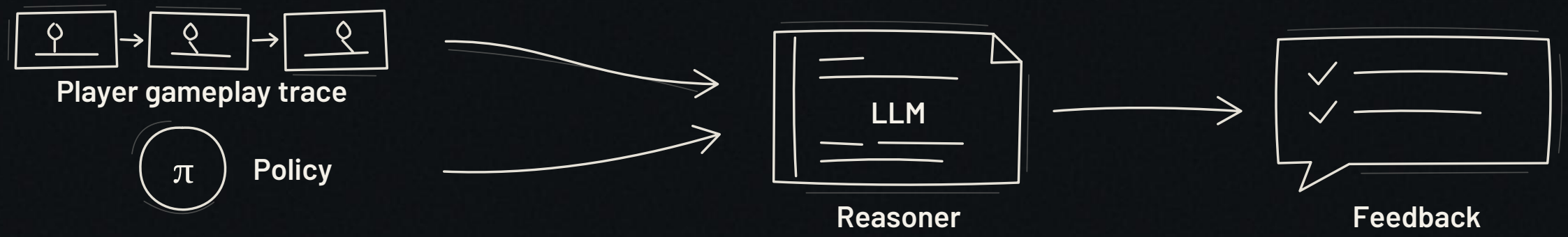
Learned AI Behavior

Using a policy as a game-specific oracle

System 1 / fast, practiced reactions



System 2 / deliberate reasoning



Beyond Bots

AI is evolving from scripted opponents into adaptive gameplay participants.

Capability Layer	What It Enables	Examples
Scripted	Reliable baseline behaviors	Tutorials
Adaptive	Contextual tactics and reactions	League
Learning	Improvement through experience	2XKO
Cooperative	Teamplay and assistance	PUBG
Persistent	Long-term social attachment	

Act 02

The Reality



aws FOR GAMES

GAMES ON AWS 2026

GS Neotek | MEGAZONE CLOUD

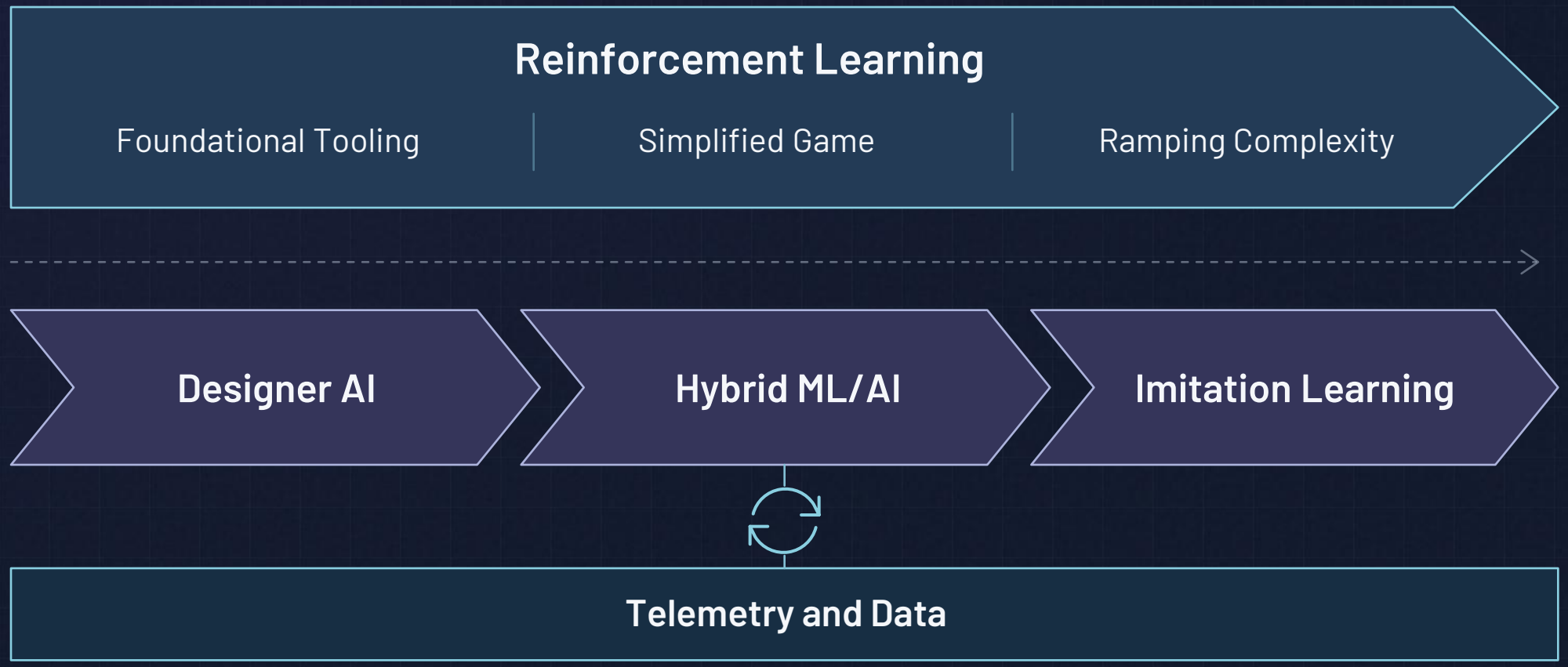




Riot focuses on session-based multiplayer games that promote long-term mastery and strategic depth, creating rich learning environments that evolve as players and metas develop.

Parallel investment

Applied bot evolution and long-term RL research

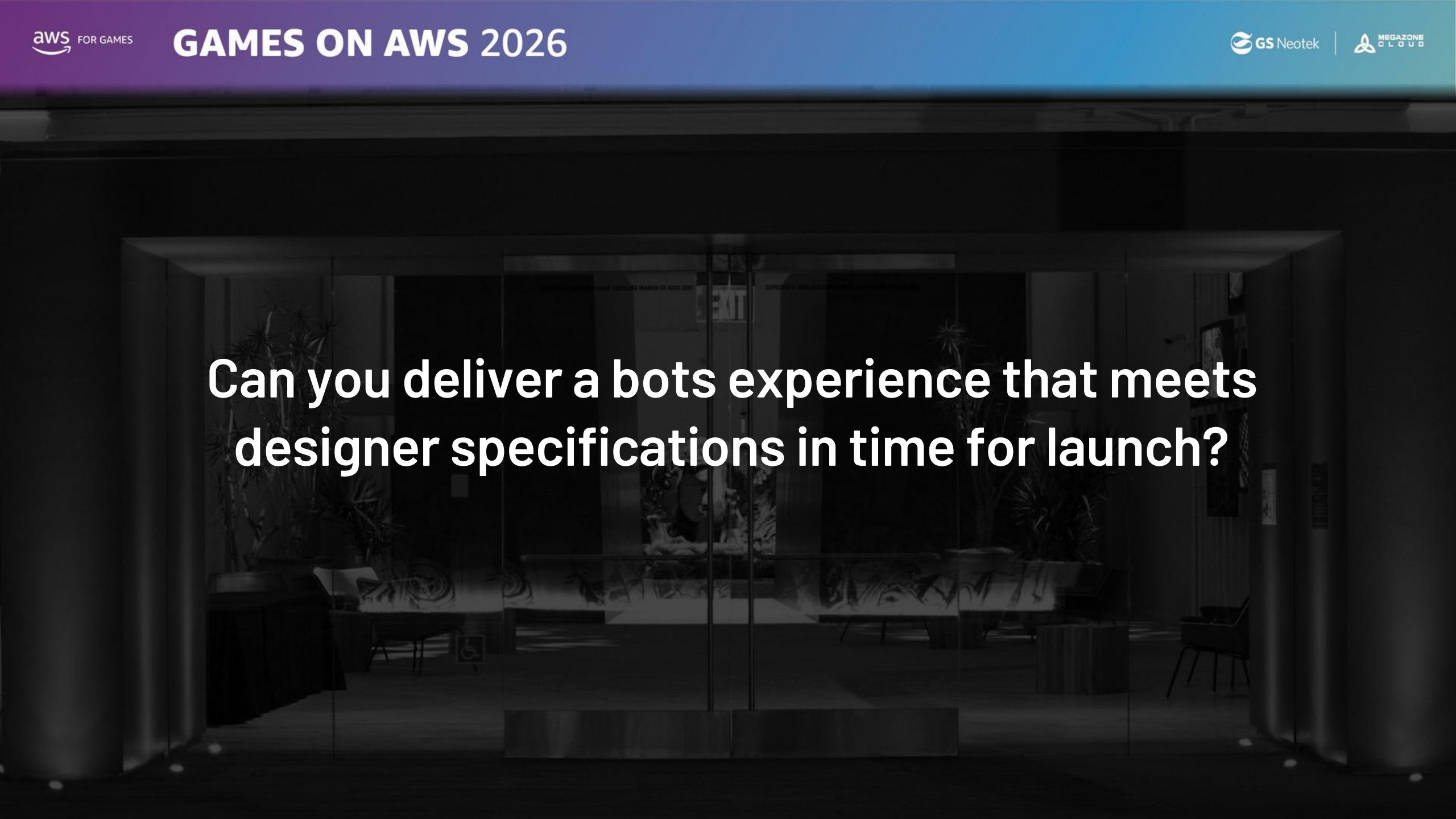


aws FOR GAMES

GAMES ON AWS 2026

GS Neotek | MEGAZONE CLOUD





Can you deliver a bots experience that meets designer specifications in time for launch?

Designer control

Example brief · Renekton as the bot

01 Farm the wave instead of forcing a fight.

02 On easy, don’t chase under the turret.

03 Give a low-health player room to escape.



Failure Modes

Things to watch out for

Reward hacking & proxy misalignment

Agents optimize the reward you give them, not the behavior you actually want.

Non-stationary data

With games as a service, the game is never done and always changing.



FAILURE MODE 02 / A CHANGING GAME

Patches change the training problem



Illustrative distributions, not measured data

Failure Modes

Things to watch out for

Reward hacking & proxy misalignment

Agents optimize the reward you give them, not the behavior you actually want.

Non-stationary data

With games as a service, the game is never done and always changing.

Evaluation theatre

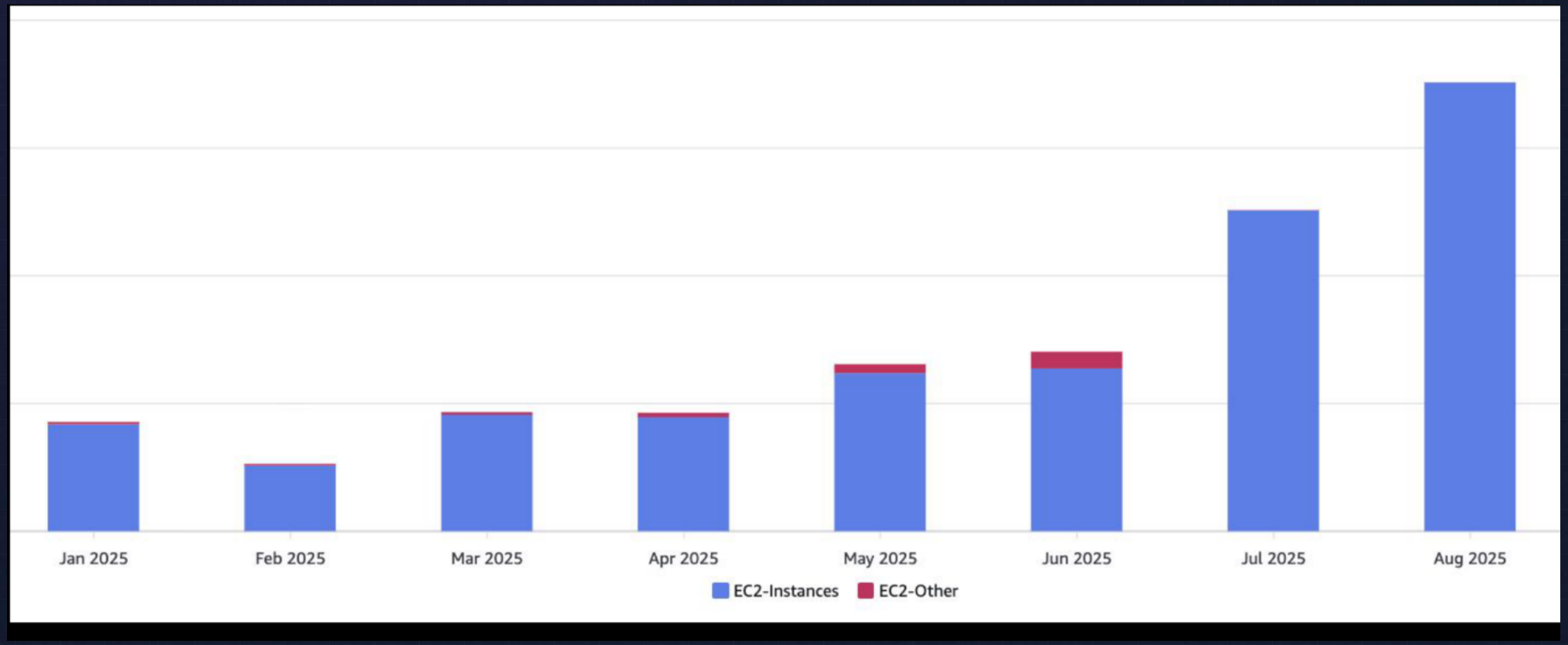
Internal benchmarks and offline metrics often don't predict real player fun or live robustness.

Infra and cost blowups

Scaling RL and self-play looks linear until it isn't.



Approaching Launch



Failure Modes

Things to watch out for

Reward hacking & proxy misalignment

Agents optimize the reward you give them, not the behavior you actually want.

Non-stationary data

With games as a service, the game is never done and always changing.

Evaluation theatre

Internal benchmarks and offline metrics often don't predict real player fun or live robustness.

Infra and cost blowups

Scaling RL and self-play looks linear until it isn't.

Griefing, exploits and degenerate play

RL agents *will* find edge cases humans haven't considered.



Training at research scale



OpenAI Five

770 ± 50 PFLOP/s-days

10 months of training



AlphaStar

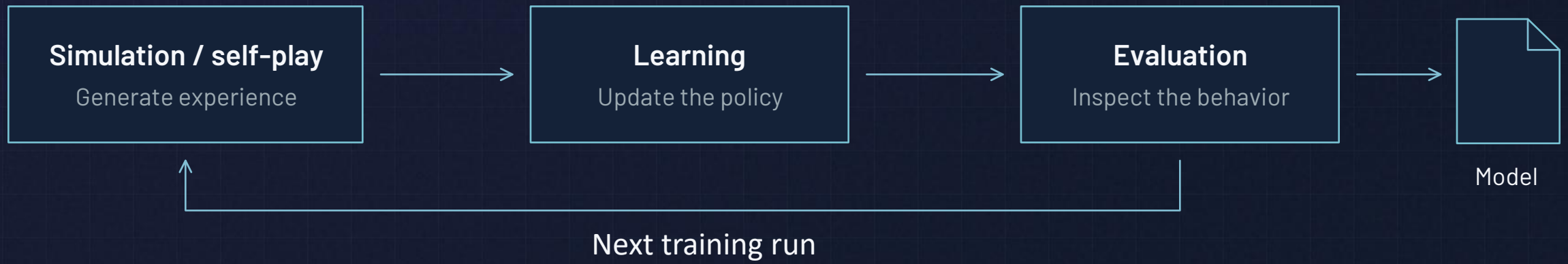
32 TPU v3 / agent

12 agents · 44 days

Reported training figures · OpenAI et al. (2019); Vinyals et al. (2019)

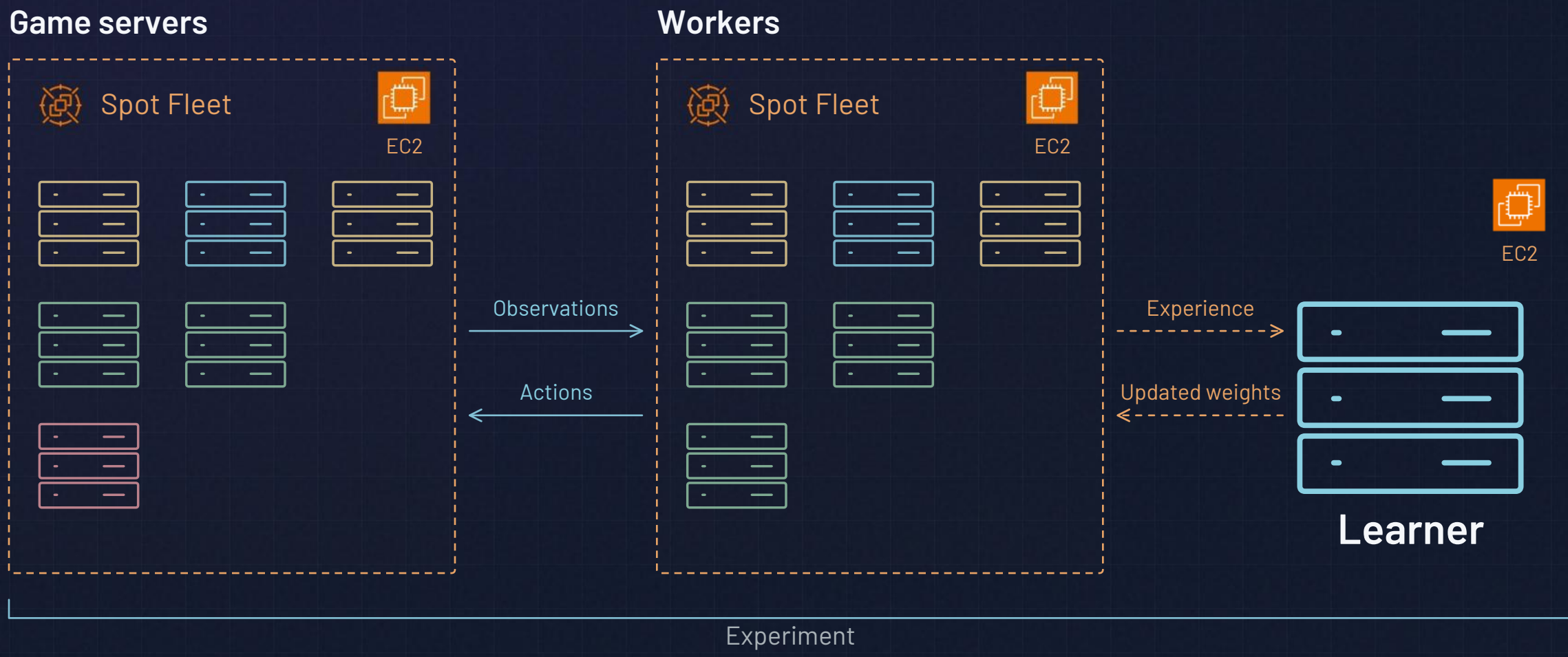
TRAINING / CONCEPTUAL WORKFLOW

Training and evaluation



Conceptual workflow · detailed connections to confirm.

AWS Partnership



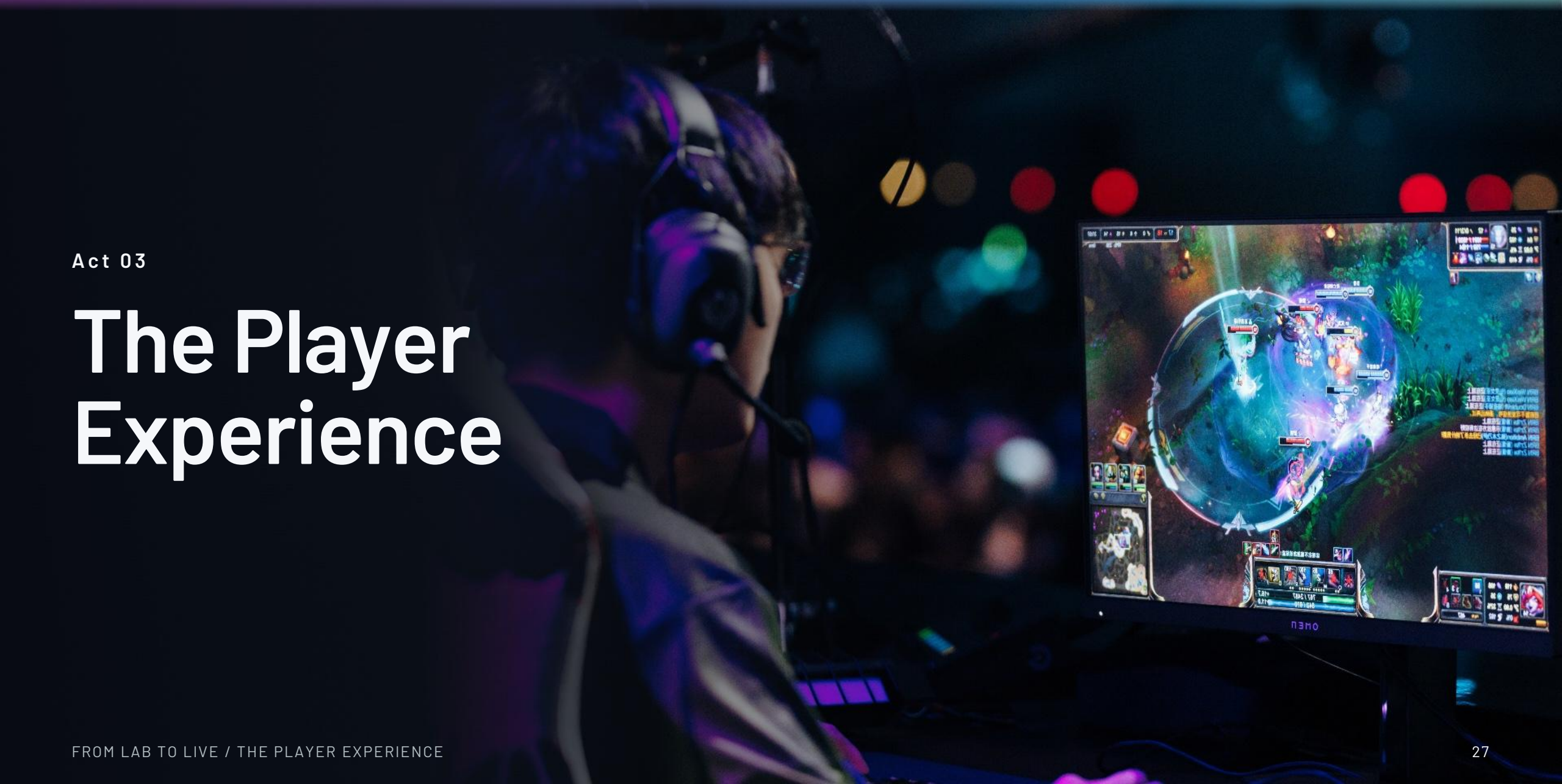
HOW RESEARCH PARTNERS

Engagement model

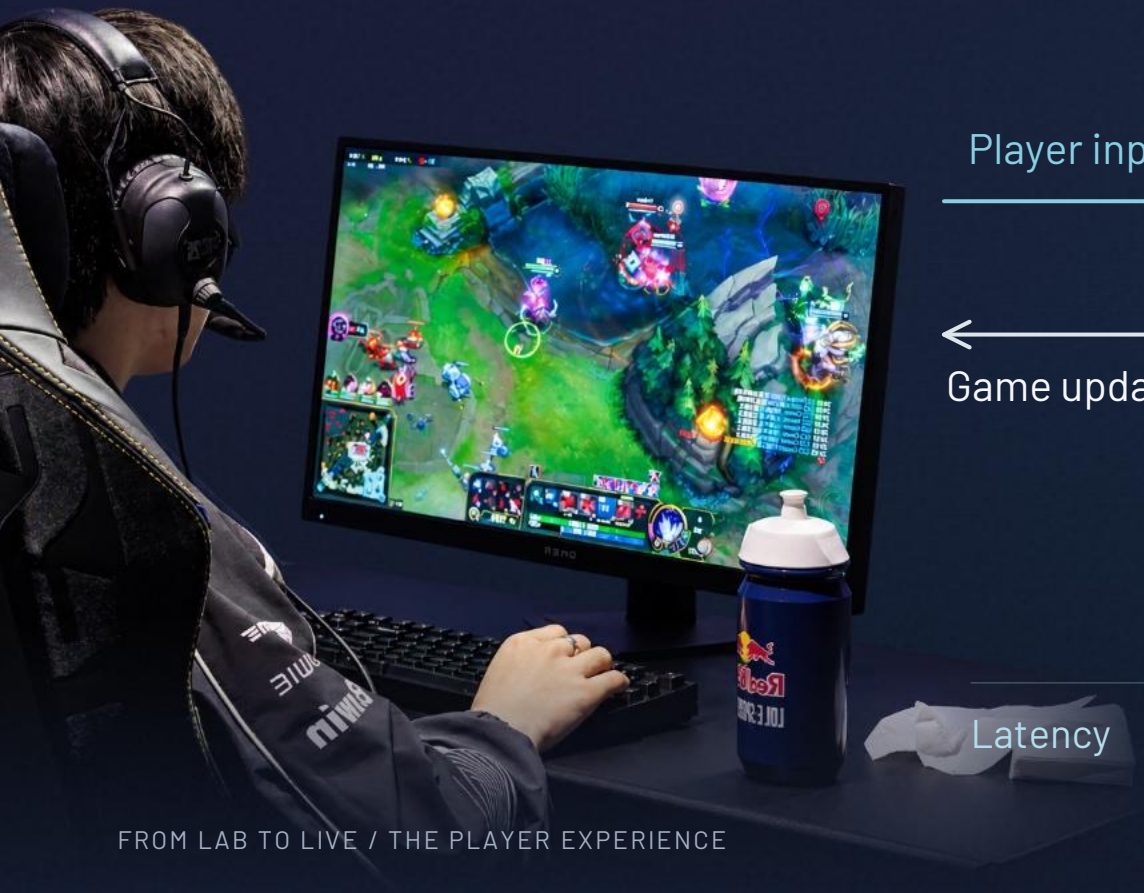


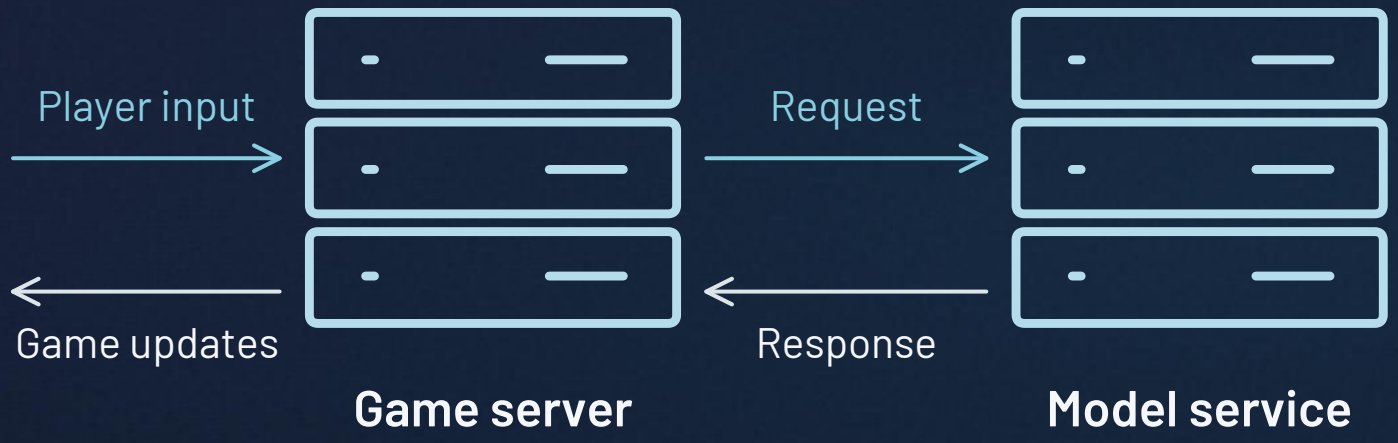
Act 03

The Player Experience



Model serving





Latency | Availability | Regions | Serving cost

Scaling Inference

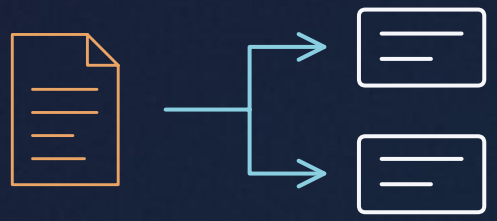
Riot's model serving platform



Serving

KServe + Triton

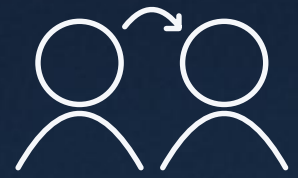
CPU and GPU models.



Why

Portability

Across runtimes
and hardware.



The tradeoff

Riot owns operations

Reliability, scaling,
and integration.

Deploying a model

Game team

Platform team







Model + config

Location · Runtime
Pre / post-processing

Build + deploy

Retrieve dependencies
Package the model

Endpoint

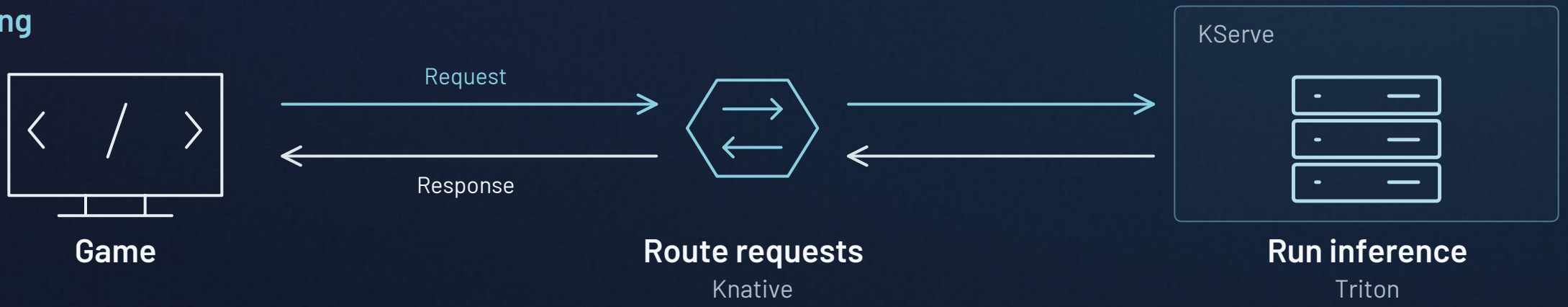
Check status
Validate inference

Deployment and serving

Deployment



Serving



Amazon EKS

From lab to live

Models can play your game.

Training is expensive. AWS helps us manage the cost.

Engage design early to shape the behavior.

Use the right deployment model to reach players.

Aim for a fun, responsive player experience.

aws FOR GAMES

Thank you

GS Neotek | MEGAZONE CLOUD