
AWS
re:Invent

ROB301

Reinforcement learning and robotics

Ray Zhu

Principal Product Manager, AWS Robotics and Autonomous Services

AWS

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Agenda

Reinforcement learning (RL) overview

RL for robotics

Role of simulation

An example: Autonomous tracking and following



© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Reinforcement learning overview

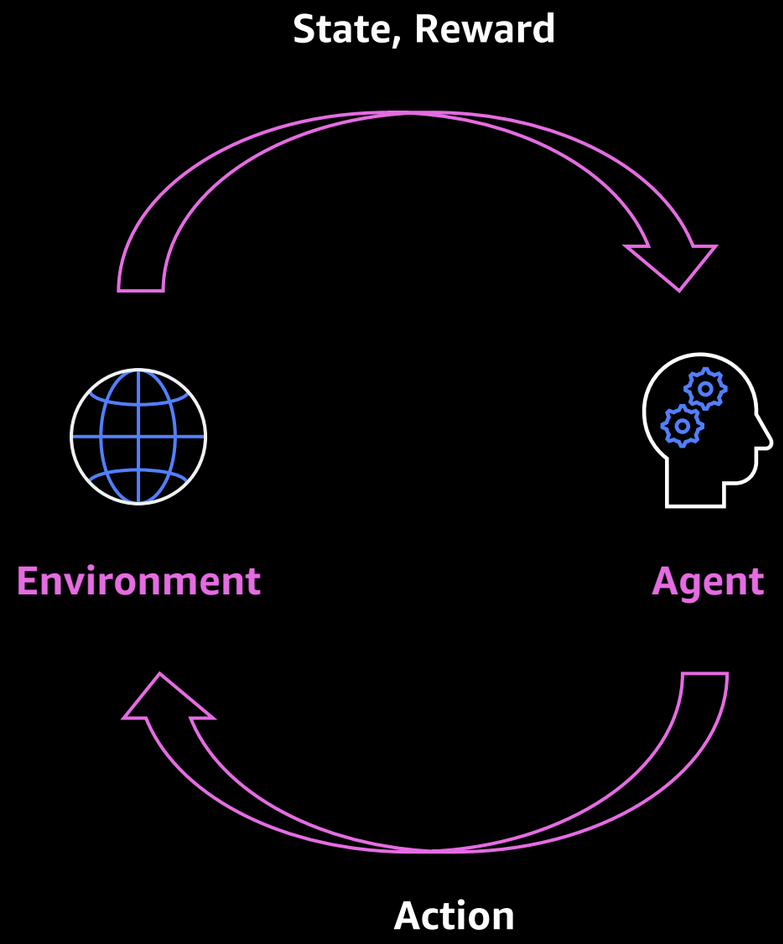


AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Reinforcement learning overview

An area of machine learning in which software **agents** take **actions** in an **environment** in order to **maximize the notion of cumulative reward** with the objective of completing a specific task

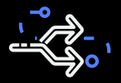


Reinforcement learning overview

WHY RL?



No existing training dataset required

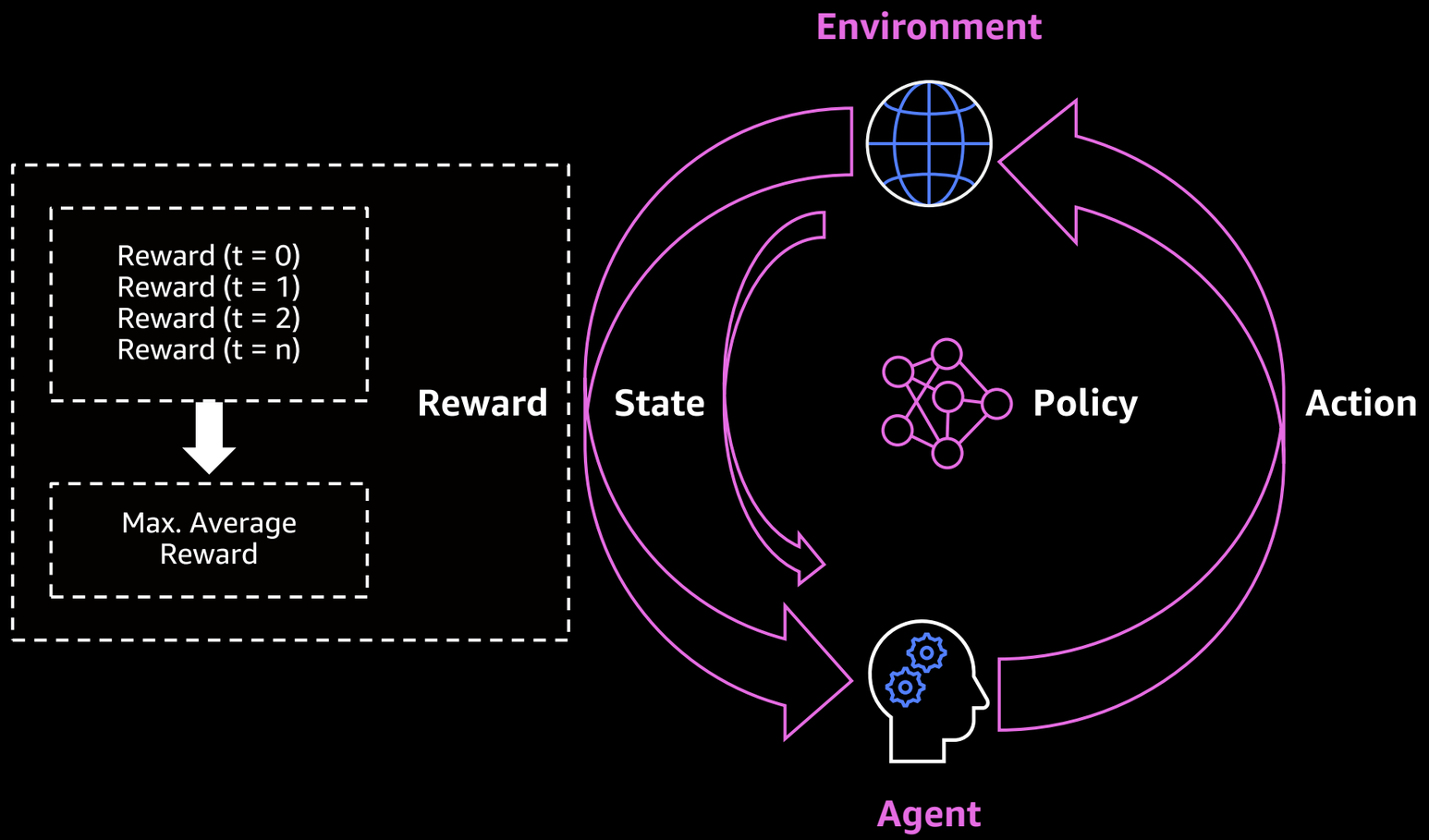


Adaptive to any environment scenario



Allows iterative training with human inputs

Reinforcement learning overview



Reinforcement learning for robotics

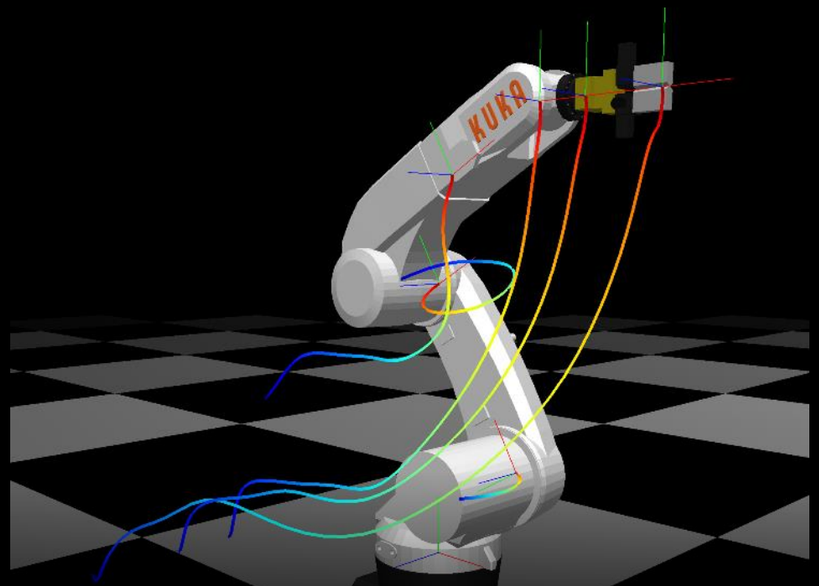


AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Reinforcement learning for robotics

Reinforcement learning algorithms have the potential to enable robots to acquire complex behaviors adaptively in their environments



Motion planning



Warehouse navigation



Object manipulation

Reinforcement learning for robotics

Use AWS RoboMaker with Amazon SageMaker to train, tune, and deploy reinforcement learning agents, and help improve operations and customer experience

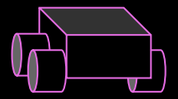
Woodside Energy



GE Aviation

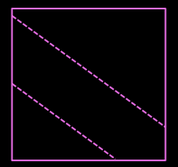


A simple example



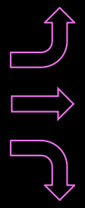
Agent

This is the robot sans programmed behavior; it merely has ability to observe its current state and compute a reward to reinforce desired behavior



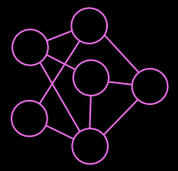
Environment

The world in which the robot (agent) will operate; the feedback from this environment is used for learning behavior



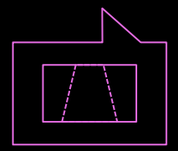
Action space

The set of actions that the robot can take; in this case, 3 discrete actions: straight, left, and right



Policy

The strategy that the robot uses to decide which action to take next

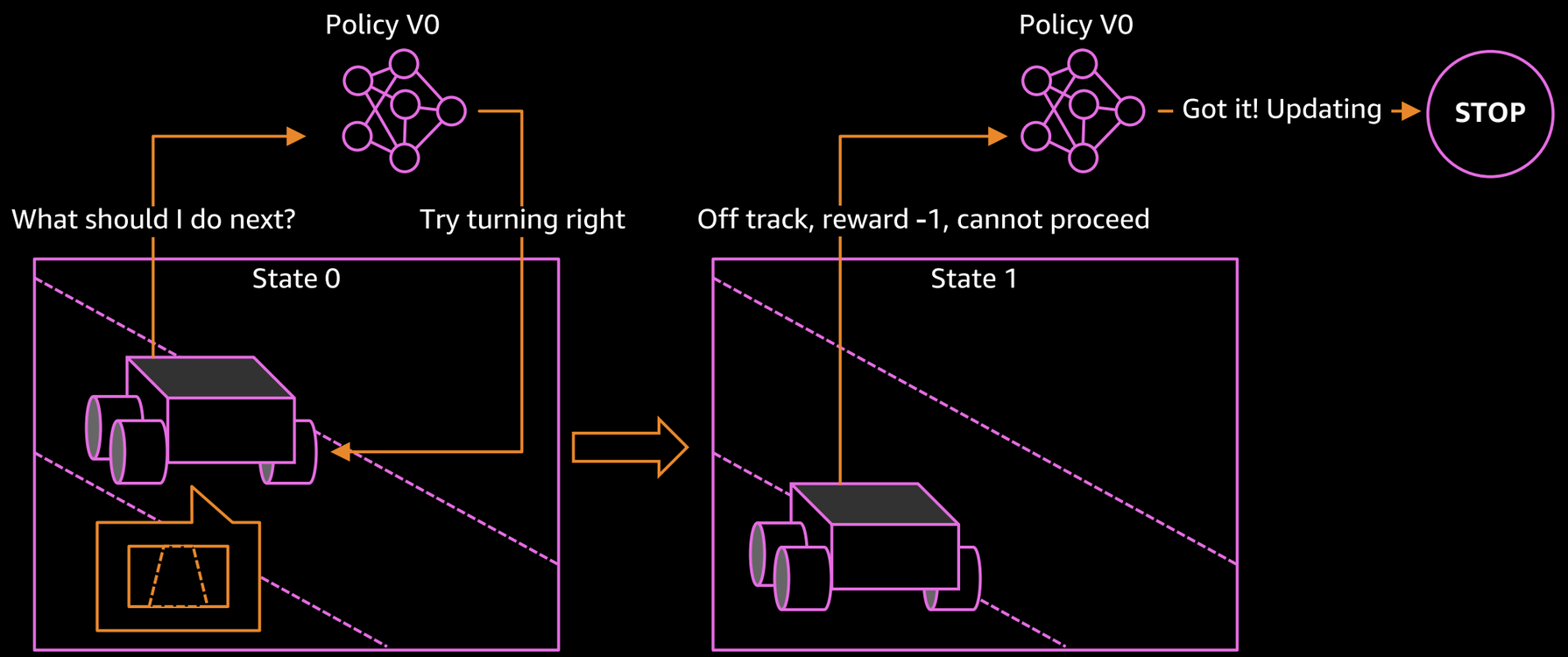


Observation space

View from the camera mounted on the front of the robot

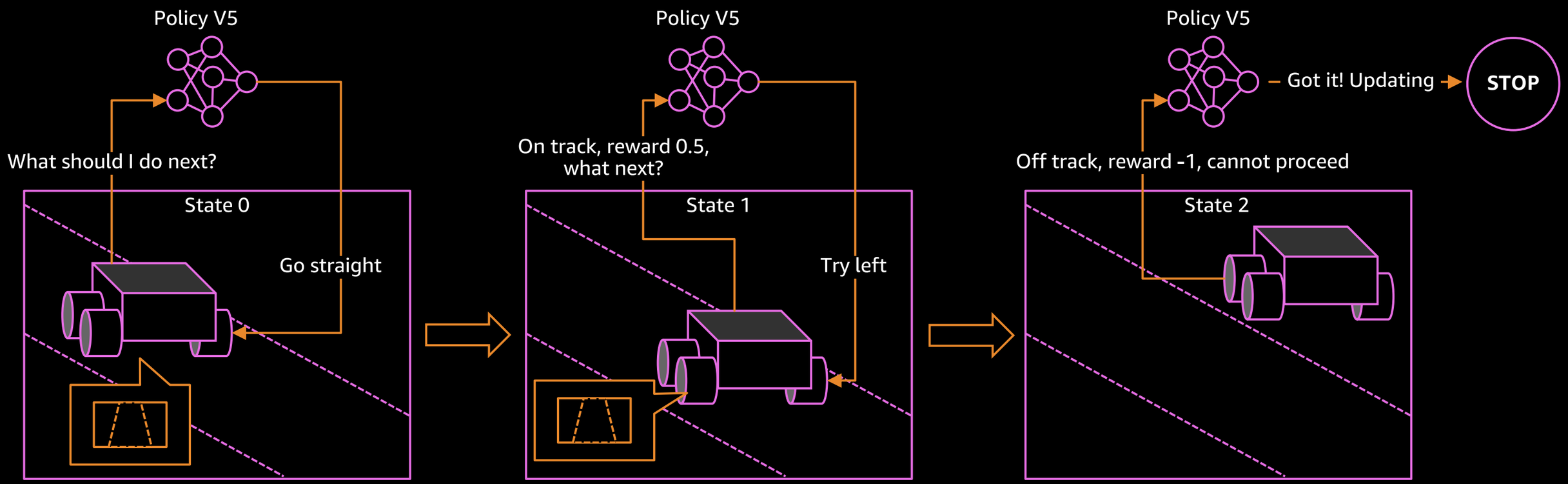
A simple example

EPOCH 0: EPISODE 1



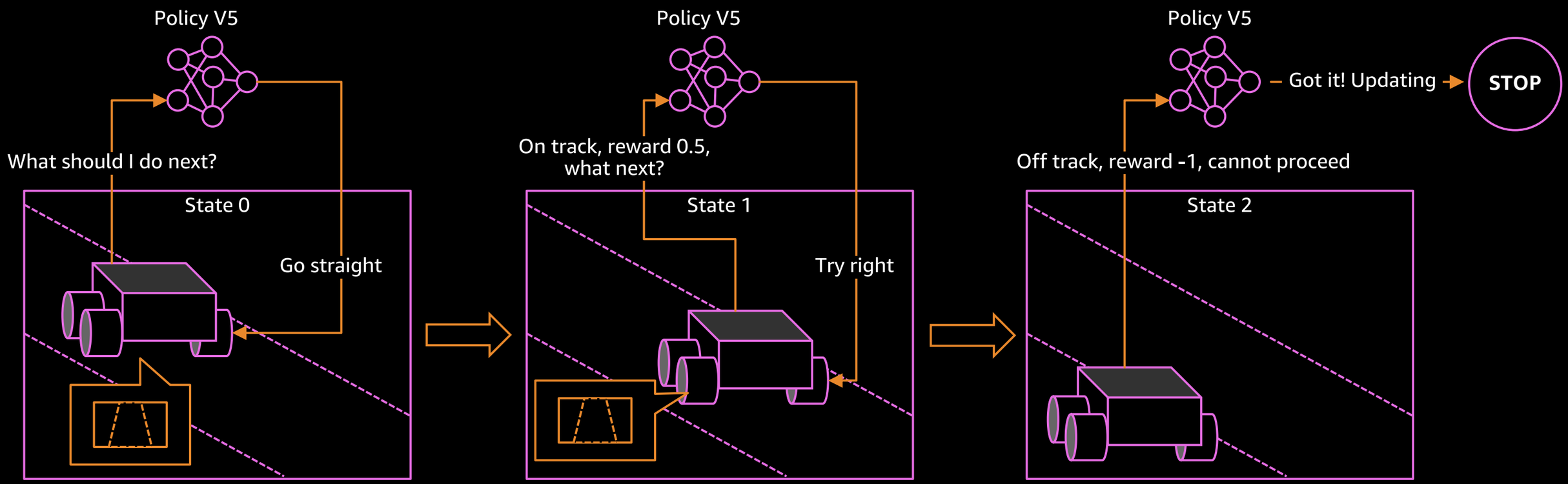
A simple example

. . . EPOCH 5: EPISODE 1



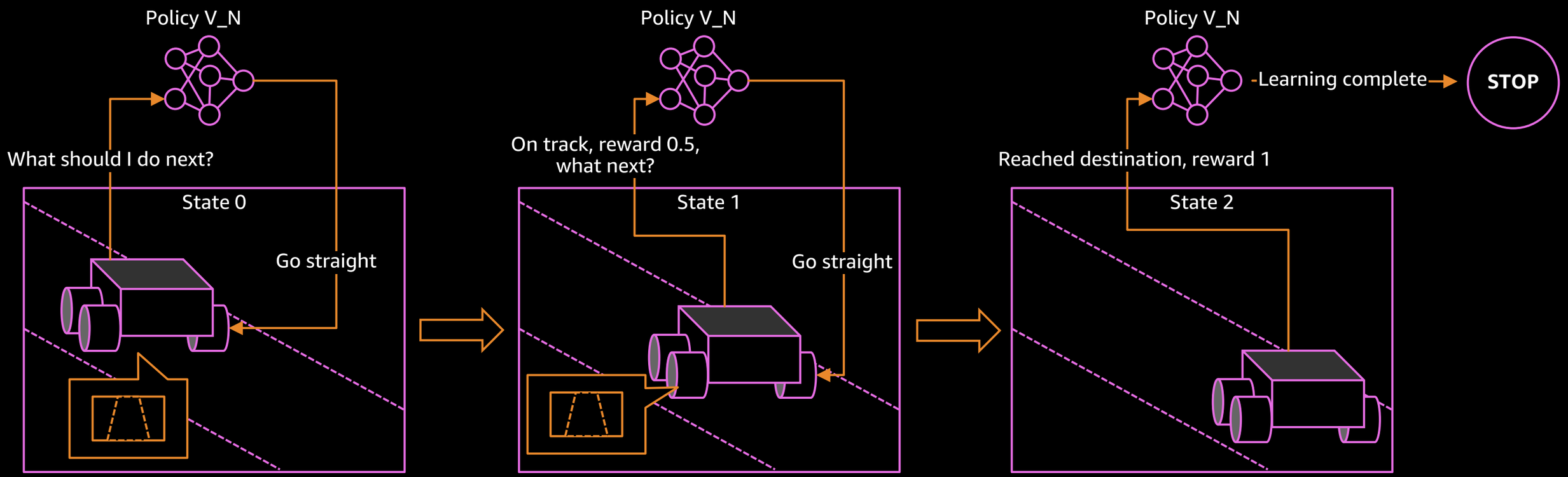
A simple example

EPOCH 5: EPISODE 2



A simple example

. . . EPOCH N: EPISODE 1



Role of simulation



© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

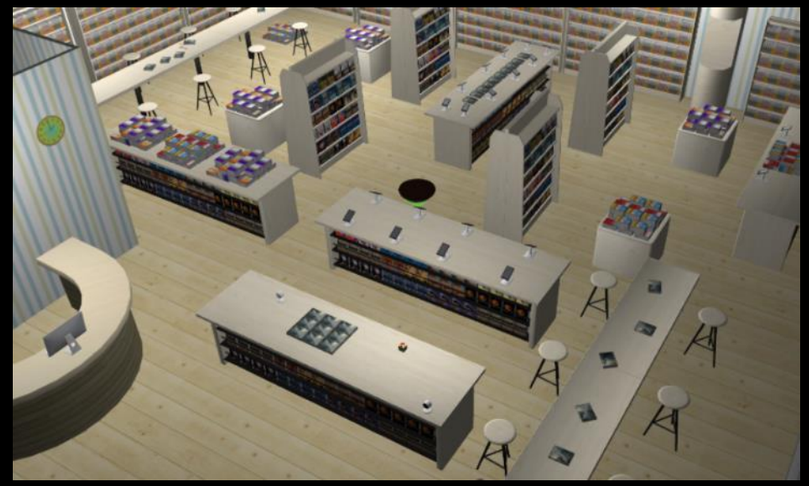
Role of simulation

More robust with virtual environments

- Easy to reset and explore various scenarios
- Easy to alter the world, introduce obstacles, change tracks, and tweak physics



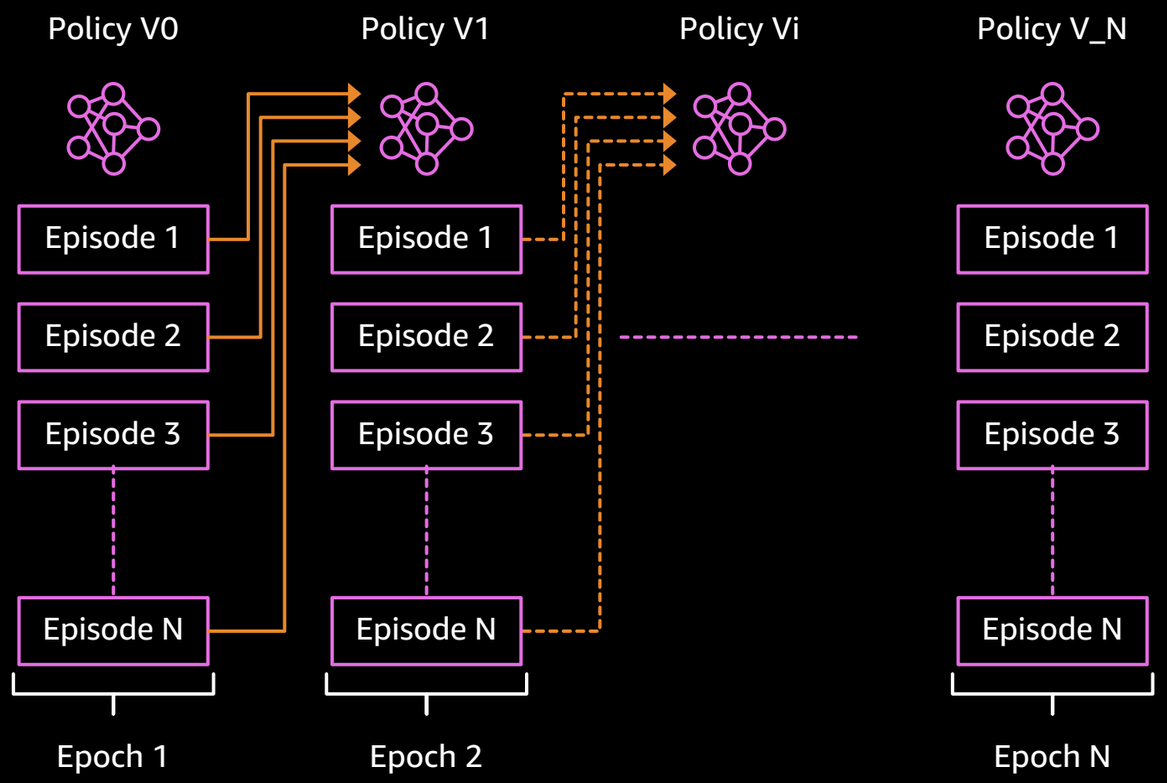


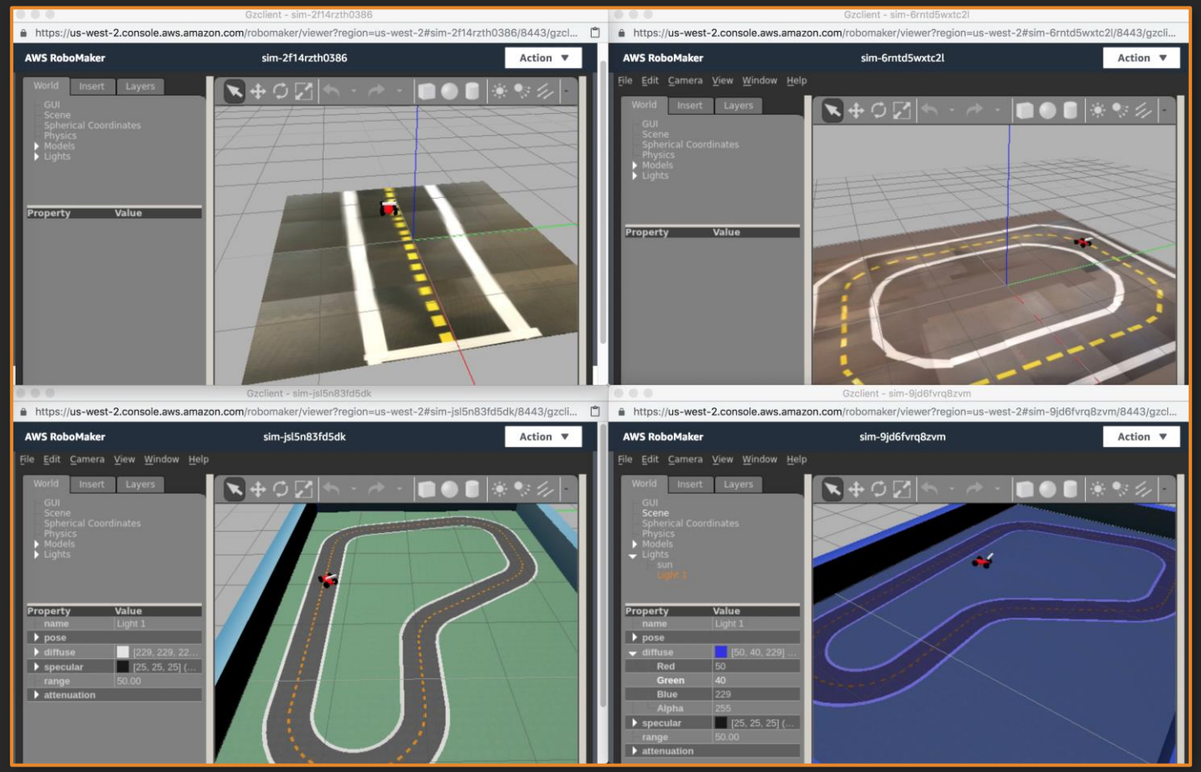


Role of simulation

Faster training

- Run a large number of simulations in parallel
- Run at faster than real-time speed





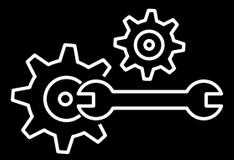
AWS RoboMaker simulation run

FULLY MANAGED INFRASTRUCTURE FOR ROBOTICS SIMULATION



Highly scalable

Concurrent simulations at cloud scale via a single API call



Fully managed

Managed robotics and simulation software stack free up engineering resources



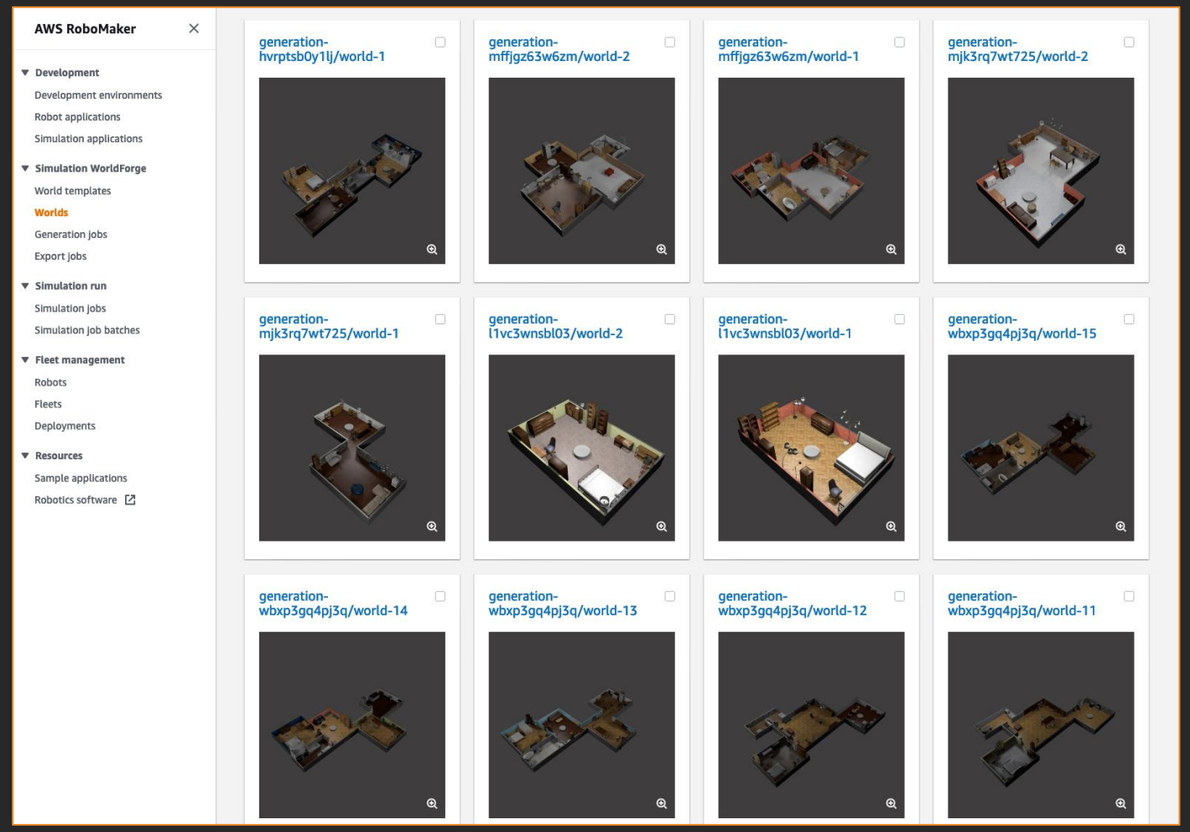
Cost-effective

Pay-as-you-go pricing at per-CPU and per-minute granularity

AWS RoboMaker Simulation WorldForge

Fully managed capability for automatic generation of virtual simulation worlds with randomization

- Out-of-the-box 3D assets and world templates
- Generation of a world within minutes vs. weeks or months
- Concurrent world generation up to hundreds of worlds
- Full integration with AWS RoboMaker simulation run



An example:
Autonomous tracking and following



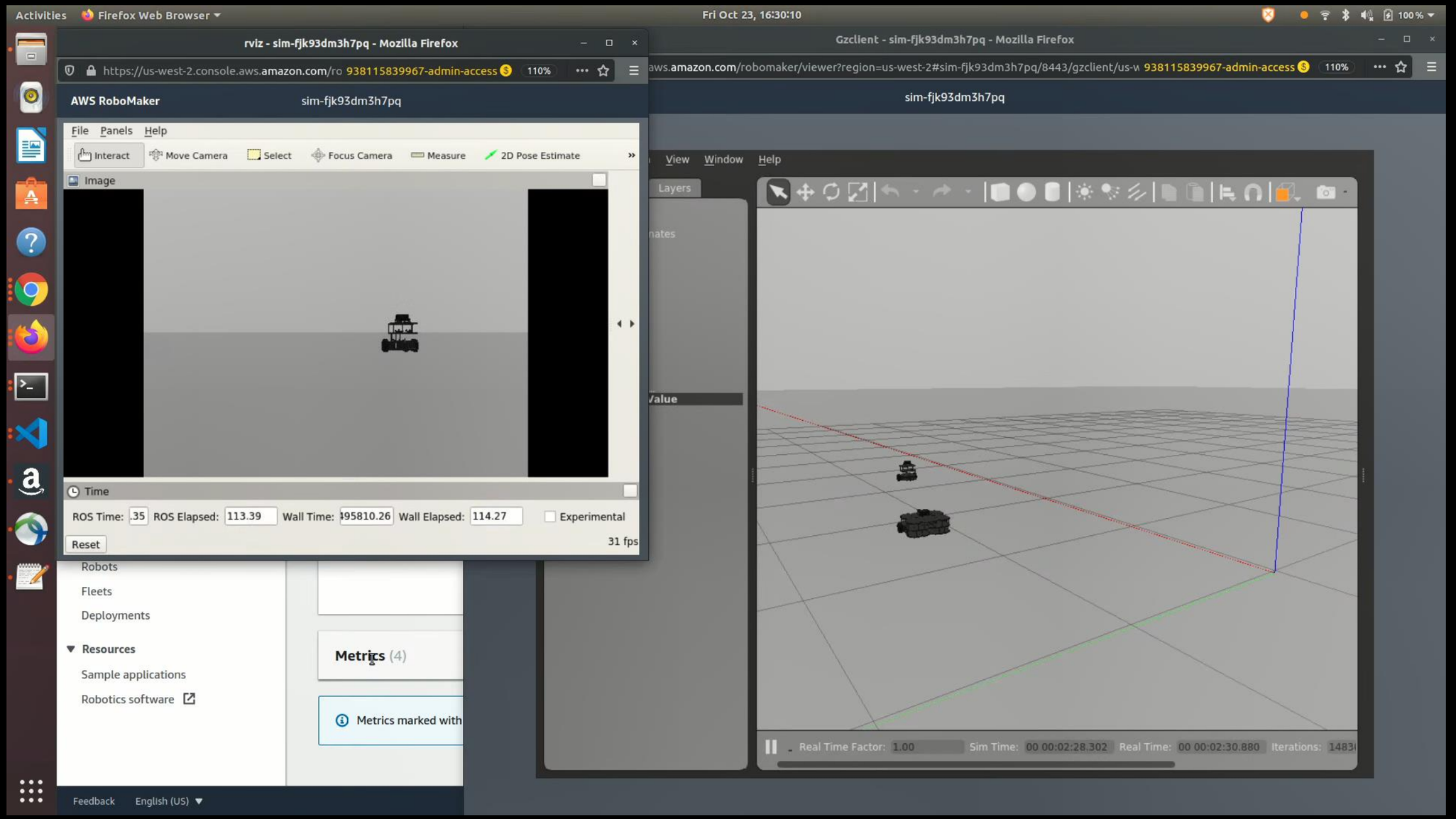
© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Autonomous tracking and following

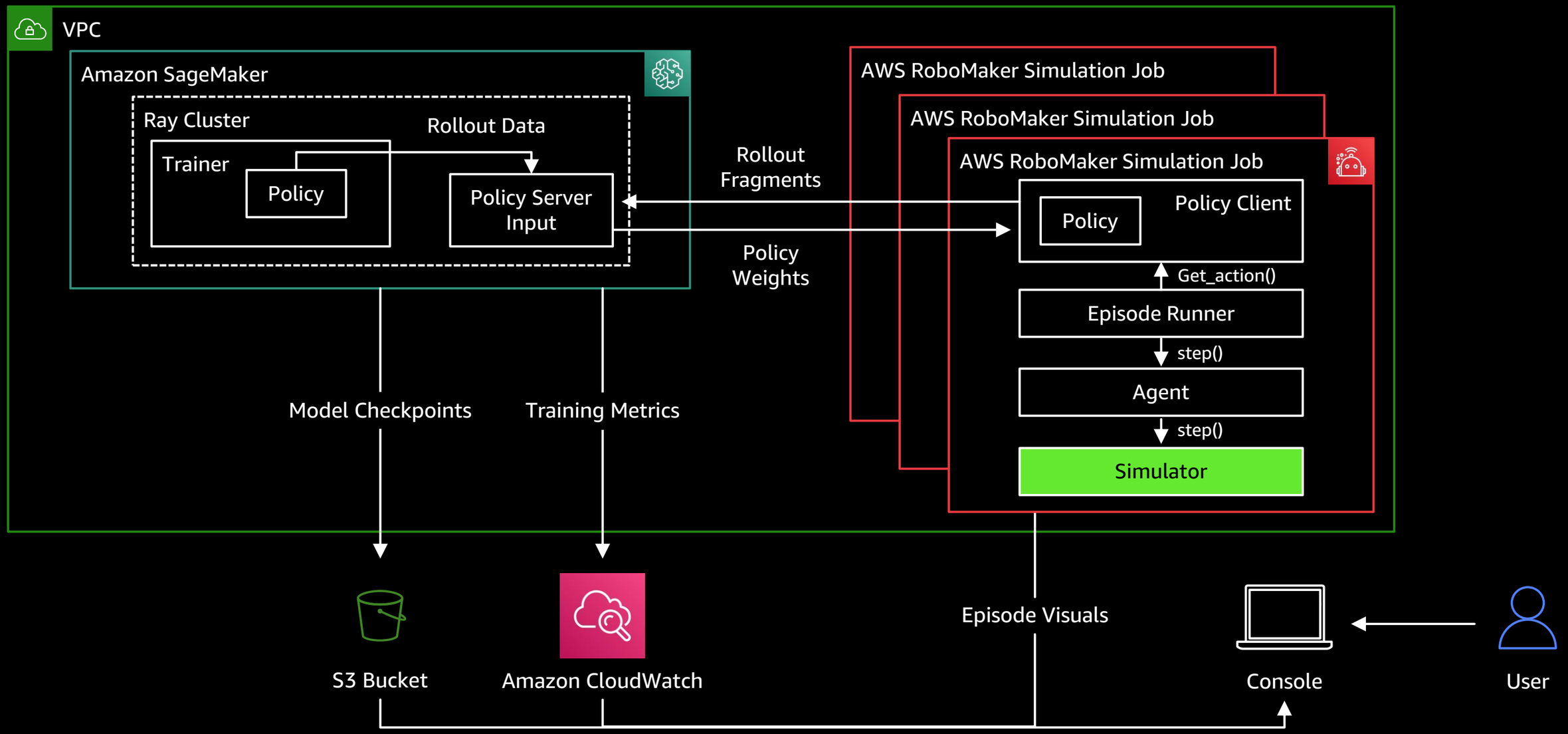
Train a robot to track and follow another robot while understanding its surroundings

- Environment: a robot to be followed
- State(s): images captured by robots' POV camera
- Action(s): 6 discrete and angled steering positions
- Reward(s): positive for decreased distance; zero or negative for increased distance

Autonomous tracking and following



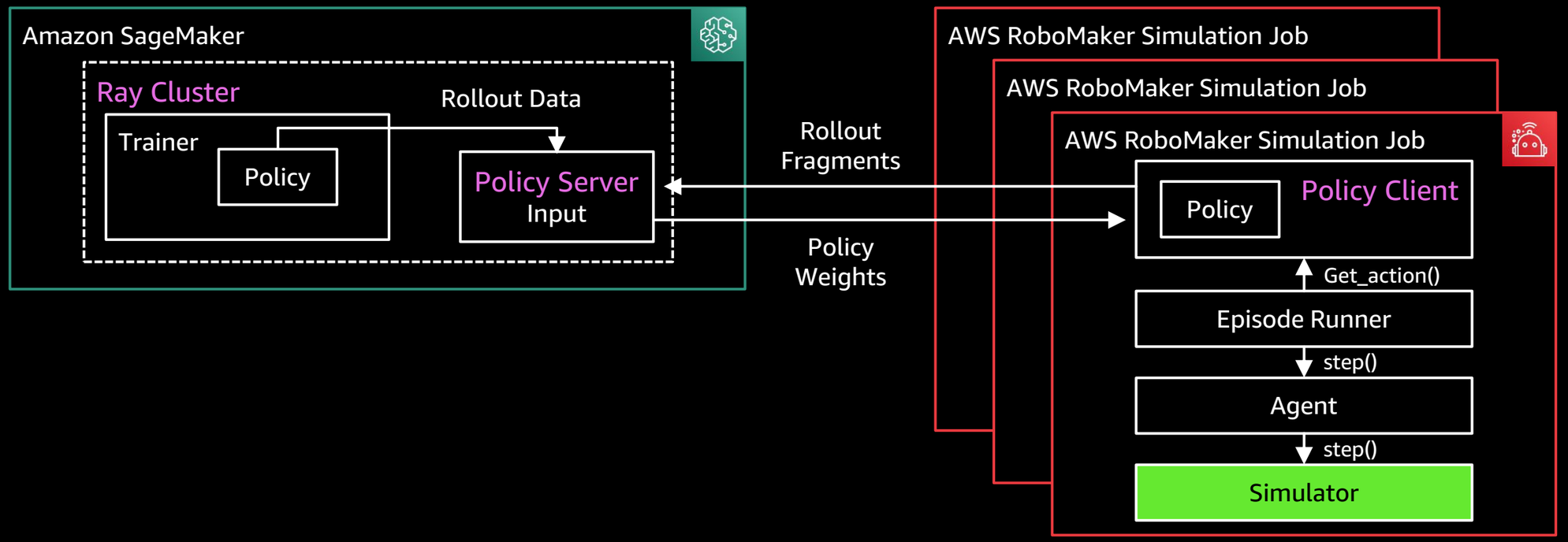
Autonomous tracking and following



Autonomous tracking and following

Ray RLlib

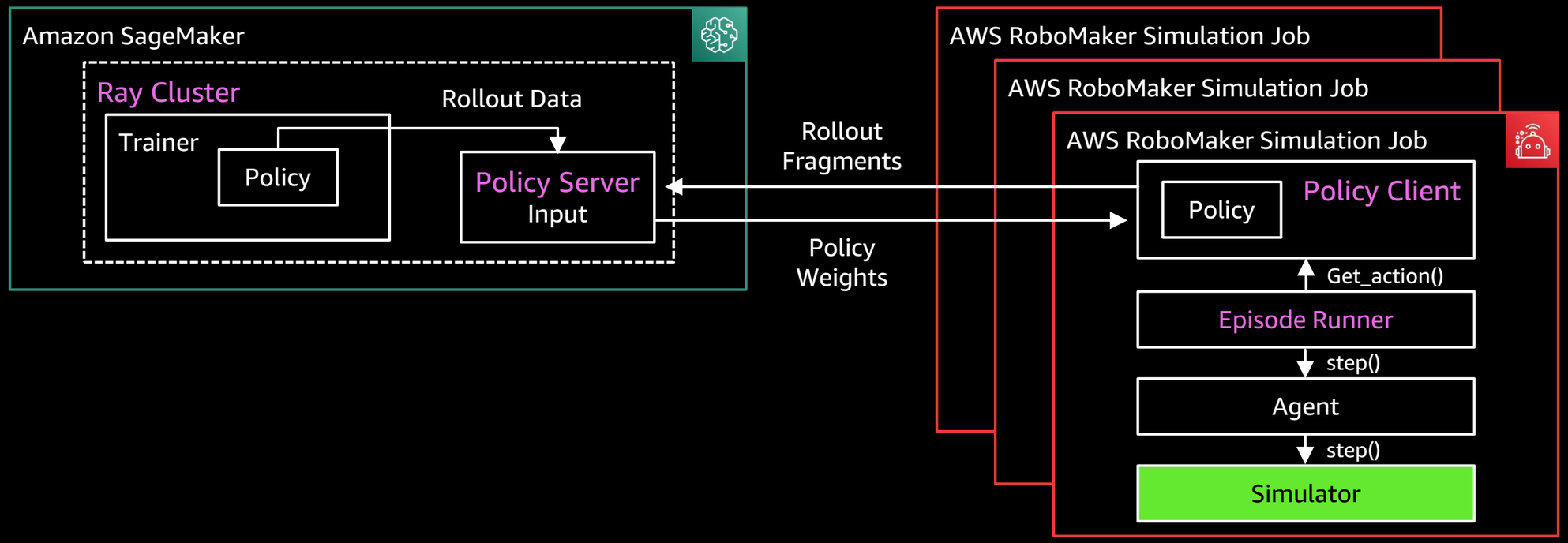
Distributed RL training



Autonomous tracking and following

Episode runner

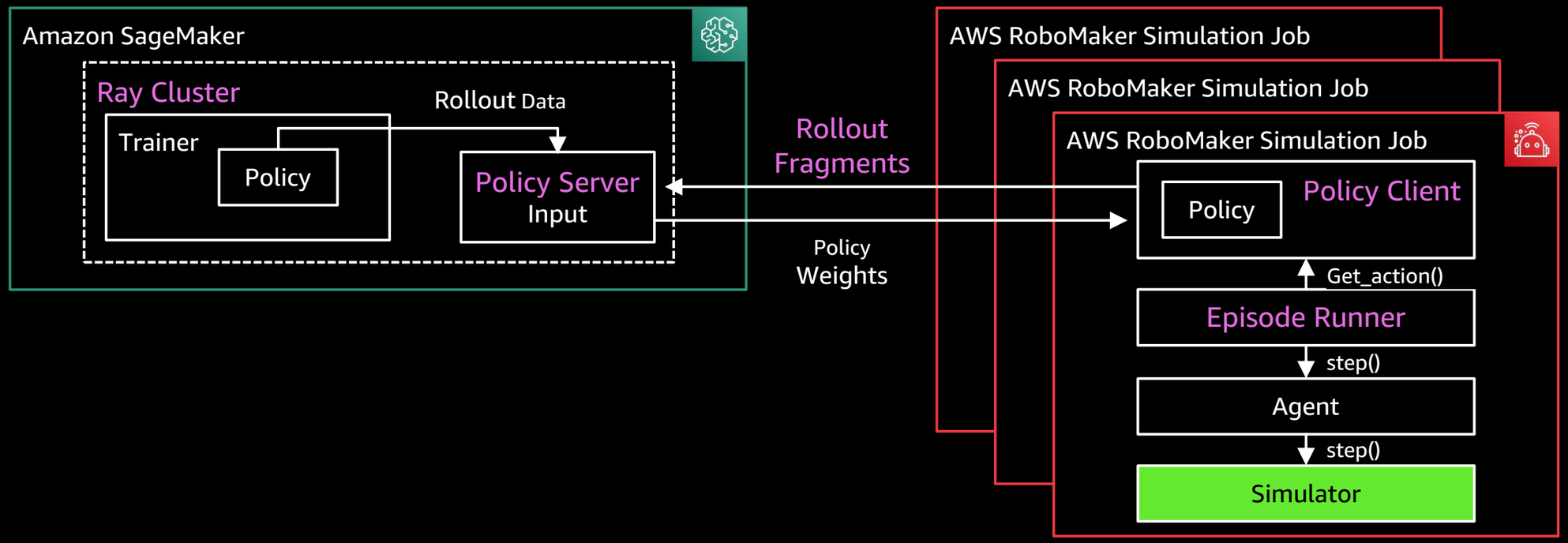
Data communication between agent and policy



Autonomous tracking and following

Rollout fragments

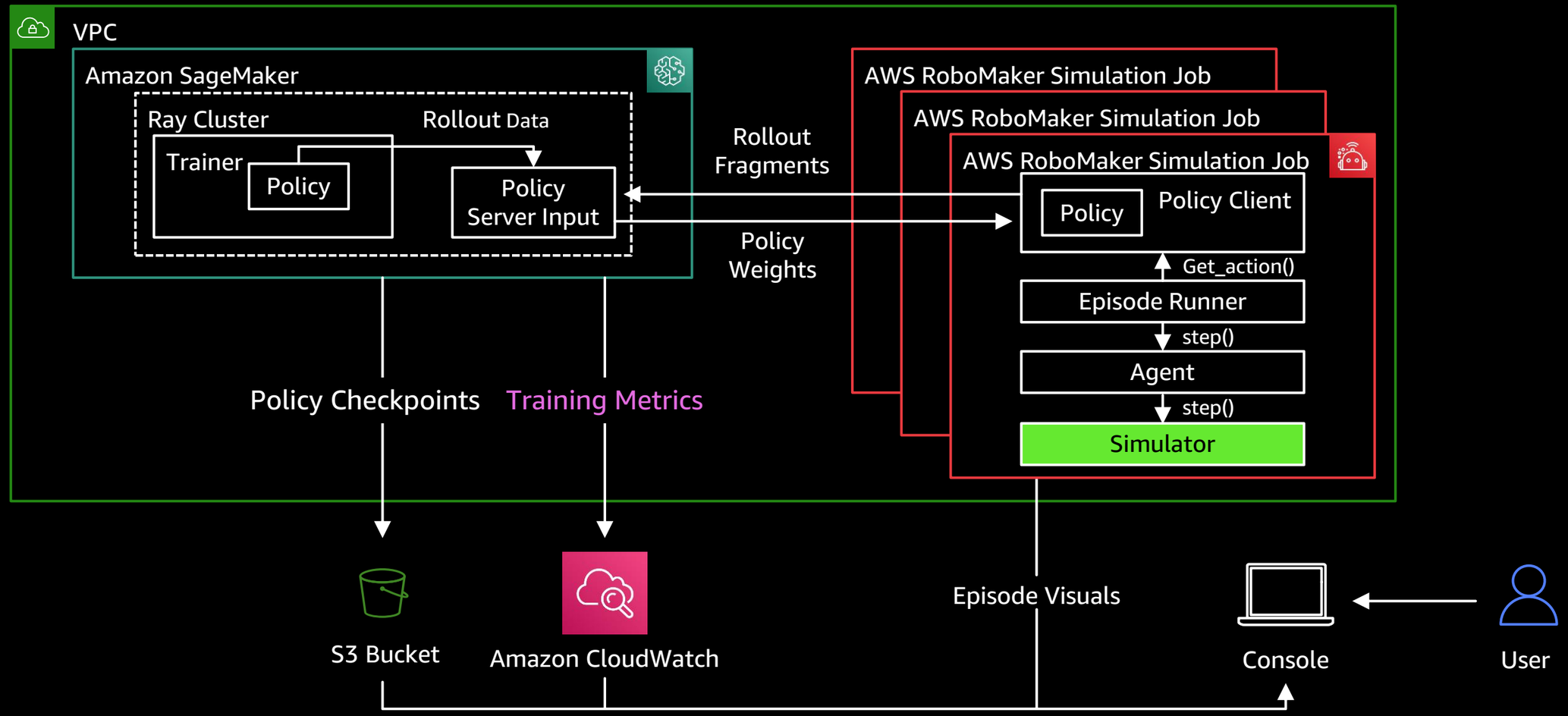
A collection of state, action, and reward data



Autonomous tracking and following

Training metrics

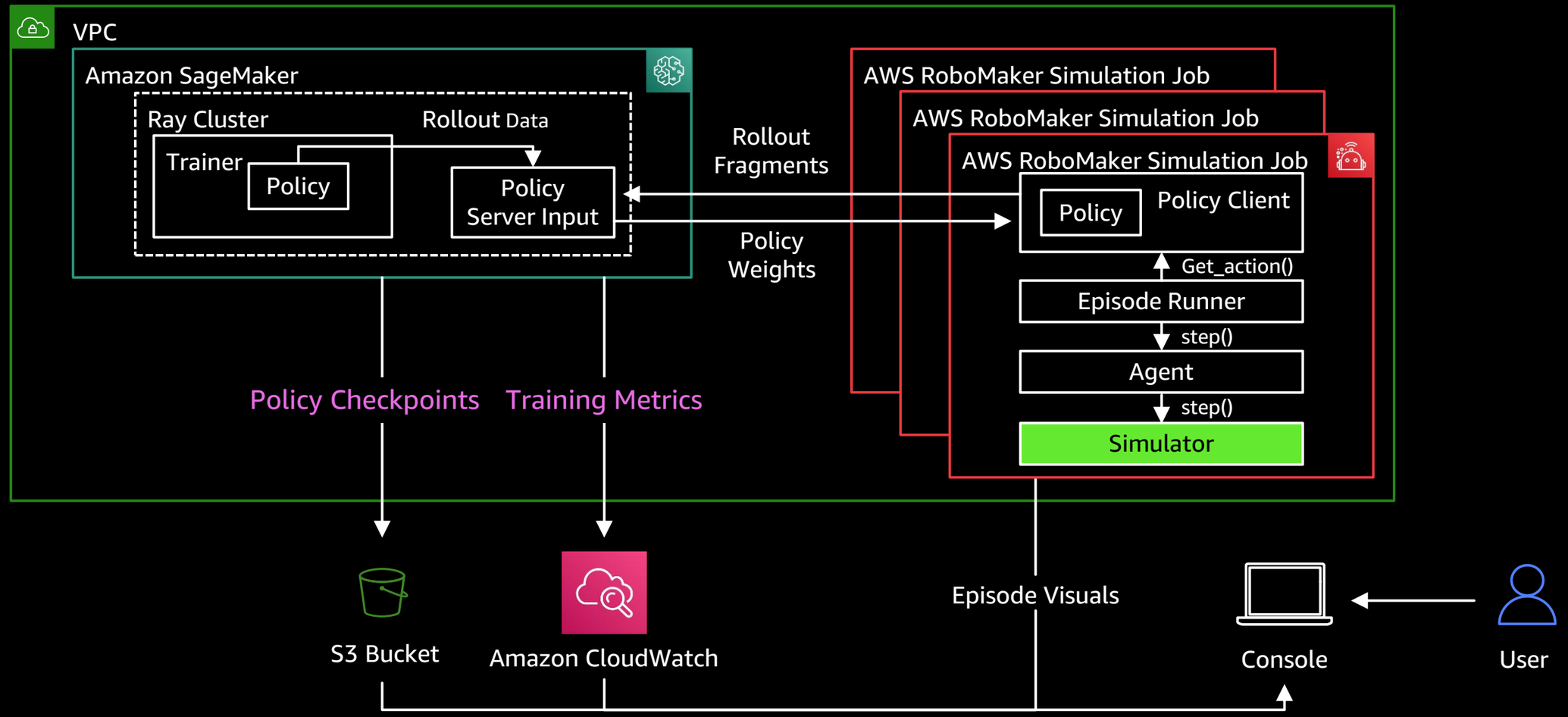
Rewards plot, episode lengths plot, etc.



Autonomous tracking and following

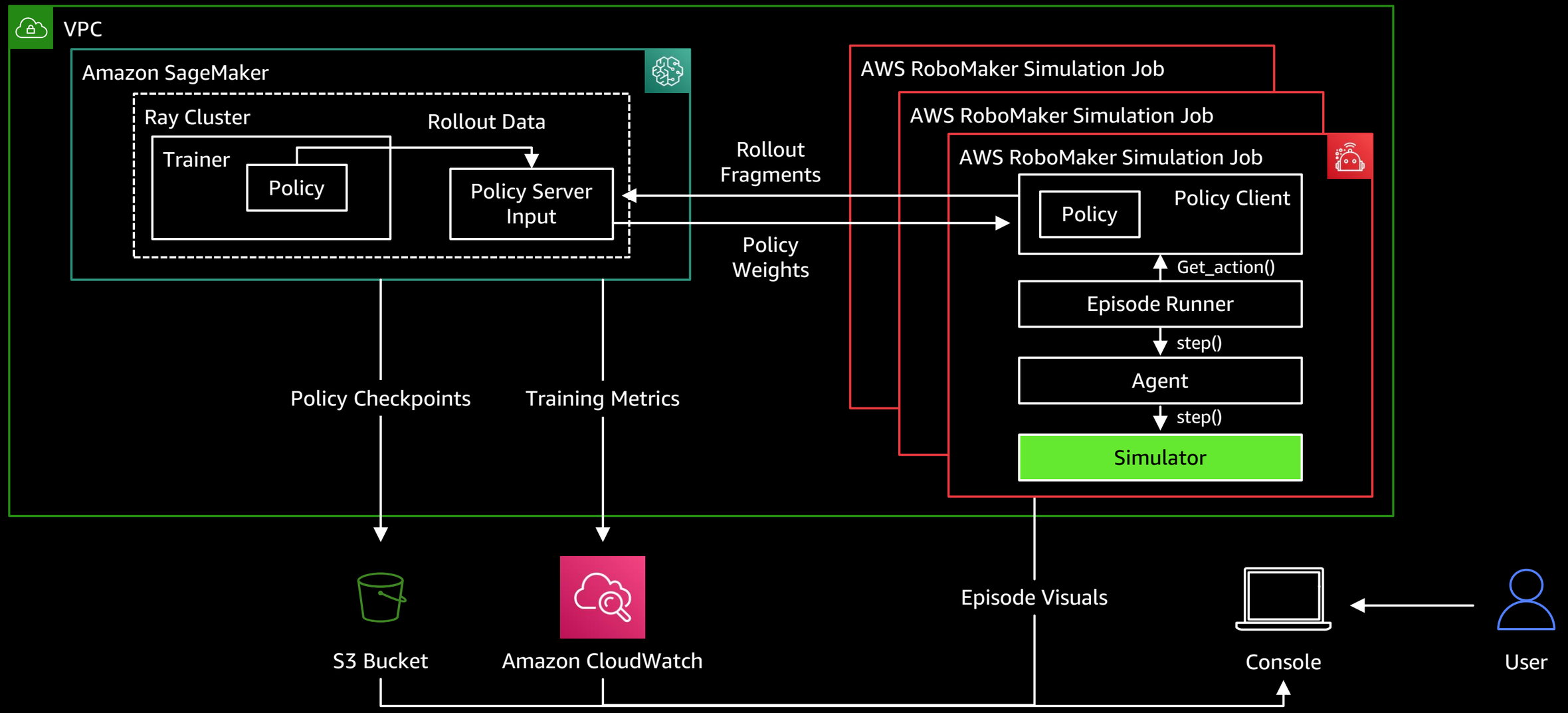
Policy checkpoints

Trace back and restart historical policies



Autonomous tracking and following

New: Kubernetes (Argo and Kubeflow) based orchestration and workflow for RL



Thank you!

Ray Zhu

rayzhu@amazon.com

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.



Please complete the session survey

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.