NetworkGym: Democratizing Network AI Research and Development via Simulation-as-a-Service

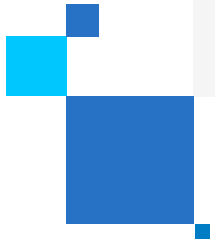
Menglei Zhang and Jing Zhu

Intel Labs/System Software Architecture (SSA), July 2023

Contributor: Pinyarash Pinyoanuntapong (UNCC)

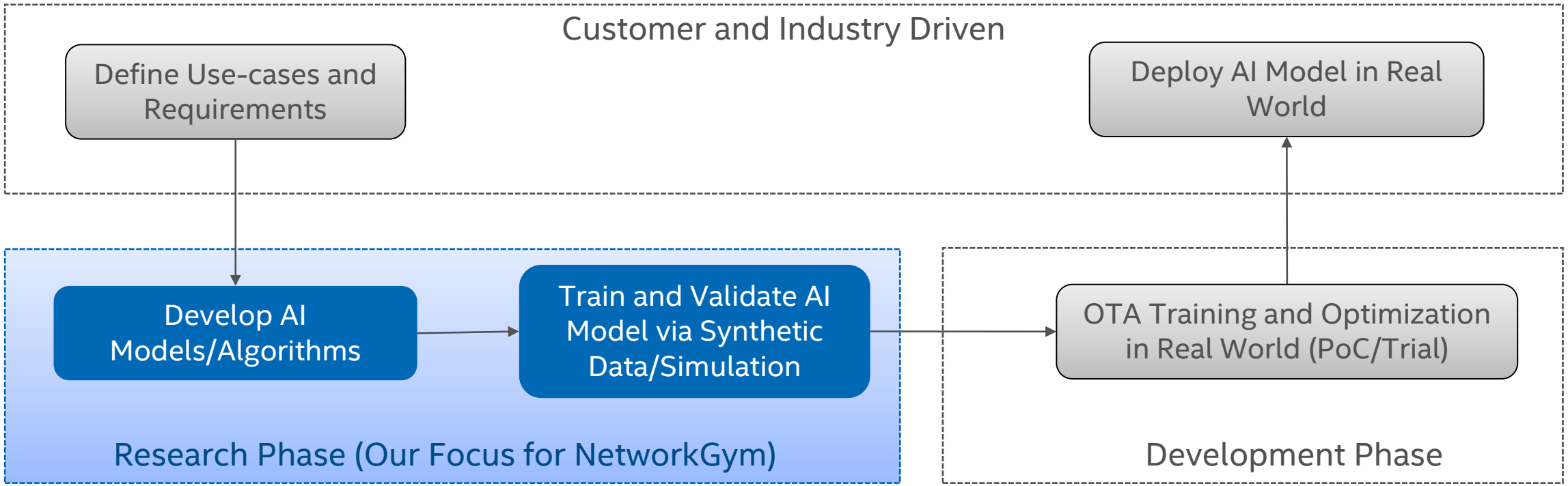
University Collaborators: Kun Yang (UVA) and Momin Haider (UCSB)

Acknowledge: Shu-ping Yeh (IL/SSA)



intel®

Network AI Models/Algorithms Development Cycle



Network AI Developer's Challenges (Why NetworkGym?)

- real-world dataset controlled by network operator, difficult to acquire, not aligned with specific usage or requirement.
- dataset by itself not enough, also need environment to train/test AI models, e.g., Reinforcement Learning, etc.

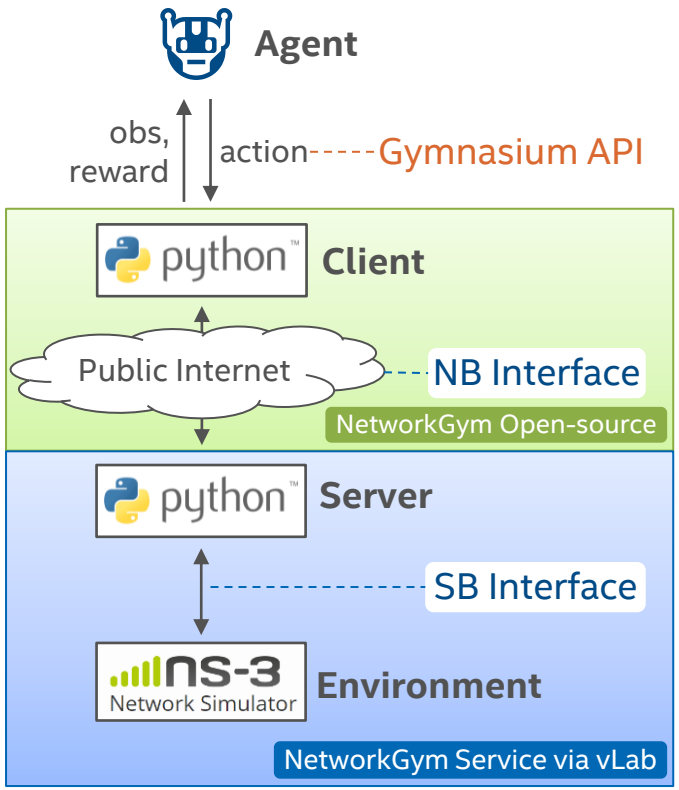
  ✓ at present, we enable 3 use cases: multi-access traffic splitting, QoS-aware traffic steering, and (cellular) RAN slicing.
- network simulation tools (e.g., ns-3, etc.) often very complex and difficult to use, especially for Network AI researcher and developer.

  ✓ requires zero knowledge of network simulation to train the agent.
- lack of common simulation environment with simple APIs to develop, evaluate, and benchmark Network AI models and algorithms.

  ✓ follow the standard gymnasium API for AI model training.

  ✓ additional API for network simulation configuration.

NetworkGym Overview



Agent: implements algorithms/models and interacts with Environment.

- **Gymnasium API**: interface between Agent and Environment, including action (↓), observation (↑), and reward (↑).

Client: connects Agent to the Server, configures the Environment, and enables data conversion between Gym and NetworkGym format.

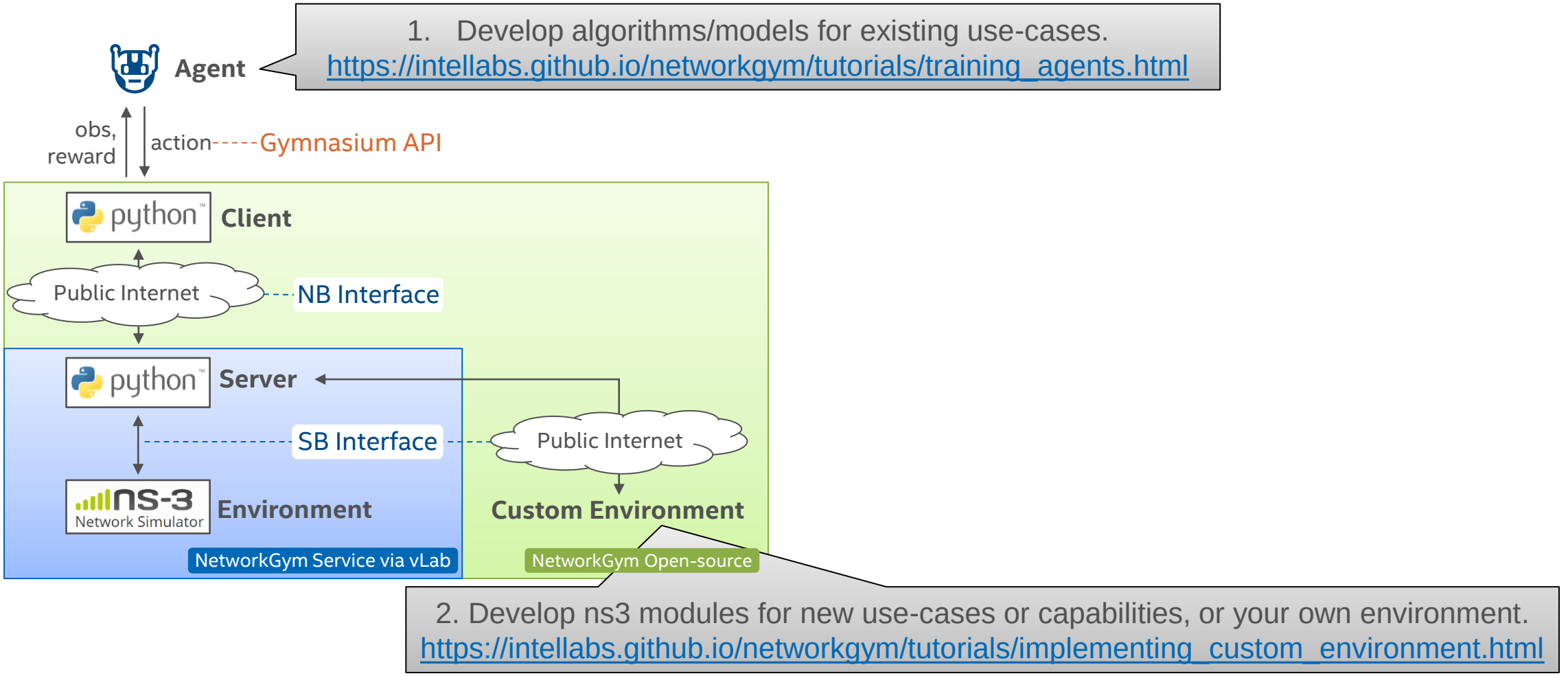
- **Northbound Interface**: interface between Client and Server, including network configuration (↓), action (↓), and network stats (↑).

Server: manages connections between Clients and Environments.

- **Southbound Interface**: interface between Server and Environment, including network configuration (↓), action (↓), and network stats (↑).

Environment: simulates or implements network environments tailored for various network AI use cases.

How to Contribute



Summary

- **NetworkGym** is a Use-Case oriented Network AI Simulation-as-a-Service Framework, Open for Contributions from the Community.
  - **Client & API** (Apachev2): connects Agent to the Server, configures the Environment, and enables data conversion between Gym and NetworkGym format via open API.
  - **Server** (IPL): manages connections between Clients and Environments.
  - **Environment** (GPLv2): simulates or implements network environments tailored for various network AI use cases.
- A PoC/Trial system is available for experimentation, supporting three use-cases: multi-access traffic splitting, QoS-aware traffic steering, and (cellular) RAN slicing.
- Interested in Collaborating and Contributing?
  - 3rd Party Network Agents: algorithms/models for the existing use-cases.
  - 3rd Party Environments: ns3 modules for new use-cases or capabilities, or your own environment.
- GitHub: https://github.com/IntelLabs/networkgym

NetworkGym