

## Distributed robotics simulator with Unreal Engine

Larry Ng(Rapyuta Robotics) Yu Okamoto (Rapyuta Robotics)

> https://rclue.readthedocs.io https://rapyutasimulationplugins.readthedocs.io



## Rapyuta Robotics is the pioneer in cloud robotics



2

### Why use Unreal Engine(UE)

- Many tools have already been developed by the Game Industry
  - Photorealistic rendering
    - data generation for machine learning
    - demo/marketing
  - Game assets are available. <u>Epic marketplace</u>.
  - Blueprint Editor
  - Can get talent from Game Industry
  - Physics engine





FF IIV Remake https://www.jp.square-enix.com/ffvii\_remake/about/index.html



Fortnite https://www.epicgames.com/fortnite/en-US/home

### rclUE is a ROS 2 client library for UE4

- ROS 2 Nodes are implemented as UE Actors
- Allows for Publishers/Subscribers etc to be defined inside UE4
- Supports: Ubuntu, ROS 2 Foxy, UE 4
- ROS components can be accessed from Blueprint
- no bridge node









### **Unreal Networking: Replication and RPC**



Replication: Copy State from Server to Clients every tick

RPC: Call a function in server from the client (and Vice Versa)



#### Client

- Player Controller receives Player Input
- 2. Player Controller moves local Actor

#### Server

- 1. Server Side Player Controller applies new state to Server side Actor
- Other clients
- Server Actor State is replicated to all other Clients

### Our Use of Unreal Networking



#### Replication: Duplicate Position of Server's Robots to all other Clients

RPC: Move Server's copy of the Robot



#### Client

- 1. Robot ROS Node receives ROS Topic
- 2. Robot moves based on cmd\_vel
- 3. Pose and Velocity sent to Network Controller

#### Server

- 1. Server Side Network Controller moves relevant robot to Pose
- Other clients
  - Server's Robot's Pose are replicated to other clients

### **Distributed Simulator architecture**





### Goal: Simulating large number of robots

- Separate Viewer from server
  - GPU is only in the Viewer client, reduces costs significantly
- Separate Robot software from server
   ROS 2 Node only exist in client
  - Sensor simulation at client
- No delay between Robot app and simulator ROS 2 Node
  - However, delays can occur when syncing a large number of clients to the server

### Standalone vs distributed simulation





### Standalone vs Distributed Simulation - 10 robots

• Standalone Instance specs

| Туре       | CPU<br>(Cores) | RAM (GB) | GPU<br>(Yes/No) | Instances | Cost (¥/month) |
|------------|----------------|----------|-----------------|-----------|----------------|
| Standalone | 8              | 56       | Yes             | 1         | ~\$550         |

• Distributed Simulation Instance specs (5 robots per Robot Client)

| Туре          | CPU<br>(Cores) | RAM (GB) | GPU<br>(Yes/No) | Instances | Cost (¥/month) |
|---------------|----------------|----------|-----------------|-----------|----------------|
| Server        | 2              | 4        | No              | 1         | ~\$35          |
| Robot Client  | 2*2            | 4*2      | No              | 1*2       | ~\$35*2        |
| Viewer Client | 2              | 4        | Yes             | 1         | ~\$200         |
| Totals        | 8              | 16       | Yes             | 4         | ~\$305         |







link: <u>https://roscondemo-kqdbp.ep-r.io/</u> backup video: <u>https://drive.google.com/file/d/18XCkvZkeP\_FZLiUbffVMac\_EE5yiEcyu/view?usp=sharing</u>

## **For Further Information**

Check out our documentation!

#### https://rclue.readthedocs.io



Or!

#### Check us out at our booth



https://rapyutasimulationplugins .readthedocs.io





## Thank you!

### We're Hiring! hiring-tyo\_sig@rapyuta-robotics.com

## **For Further Information**

Check out our documentation at:

https://rclue.readthedocs.io https://rapyutasimulationplugins.readthedocs.io

Or!

Check us out at our booth

### Standalone vs Distributed Simulation - 10 robots



• Standalone Instance specs

| Туре       | CPU<br>(Cores) | RAM (GB) | GPU<br>(Yes/No) | Instances | Cost (¥/month) |
|------------|----------------|----------|-----------------|-----------|----------------|
| Standalone | 8              | 56       | Yes             | 1         | ~¥83,000       |

• Distributed Simulation Instance specs (5 robots per Robot Client)

| Туре          | CPU<br>(Cores) | RAM (GB) | GPU<br>(Yes/No) | Instances | Cost (¥/month) |
|---------------|----------------|----------|-----------------|-----------|----------------|
| Server        | 2              | 4        | No              | 1         | ~¥5,000        |
| Robot Client  | 0.4*10         | 0.8*10   | No              | 1*2       | ~¥5,000*2      |
| Viewer Client | 2              | 4        | Yes             | 1         | ~¥30,000       |

### **Unreal Network: Replication and RPCs**





### Why use Unreal Engine(UE)

- Photorealistic rendering
  - data generation for machine learning
  - demo/marketing
- Game assets are available. <u>Epic marketplace</u>.
- Physics engine: PhysX\*
- Editor :
  - $\circ$  GUI $\rightarrow$  easy/intuitive to edit environment.
  - Blueprint  $\rightarrow$  Visual scripting. Creating logic from GUI
  - $\circ$  Behaviour Tree  $\rightarrow$  Human/non robot simulation



\*UE5 uses Chaos physics engine \*List of Unreal Engine games

> Behaviour tree editor https://docs.unrealengine.com/4.27/en-US/InteractiveExperiences/ArtificialIntelligence/Behavio rTrees/BehaviorTreesOverview/



#### FF IIV Remake

https://www.jp.square-enix.com/ffvii\_remake/about/index.html



## More About Pick Assist Robots

## Another Thing About Pick Assist Robots

## A Bit More About Pick Assist Robots

## Also About Pick Assist Robots

## One More Thing..

## And Finally





### Rapyuta Robotics is a spin-off from ETH Zürich, which has produced many famous researchers including Einstein.





### ラピュタロボティクス株式会社は、 アインシュタインをはじめ、数々の著名研究者を輩出している チューリッヒ工科大学(ETH Zürich)からスピンオフした 大学発ベンチャー企業です。



### Index

A
B
C
D



List



A
B
C
D



## A great testimonial about Rapyuta Robotics' cool and amazing pick assist robots.

"

By Someone



## Divider



## Divider



## Divider

# rapyuta robotics

## Making Robots More Accessible



## Thank you!

# rapyuta robotics

## ロボットを便利で身近に

# robotics

Making Robots More Accessible

## **rapyuta** robotics