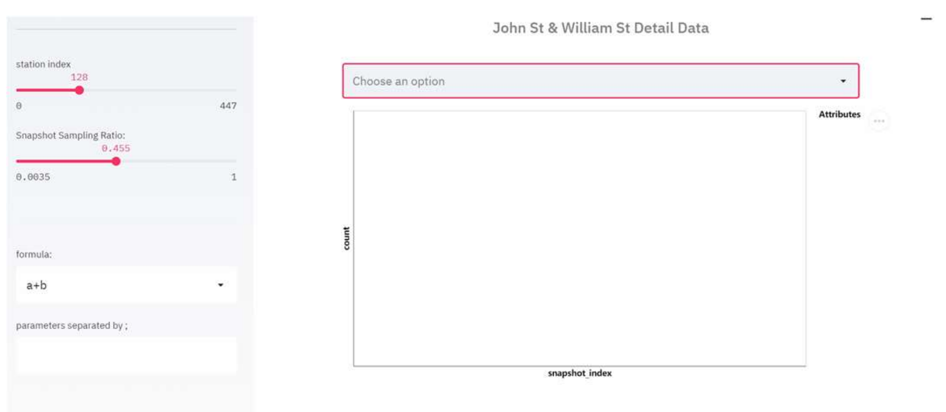
RL for Operations, 2022

Citi Bike Management in MARO with Deep RL

Sean Sinclair,
Cornell University



MARO

Multi-Agent Resource Optimization (MARO) developed by Microsoft Asia

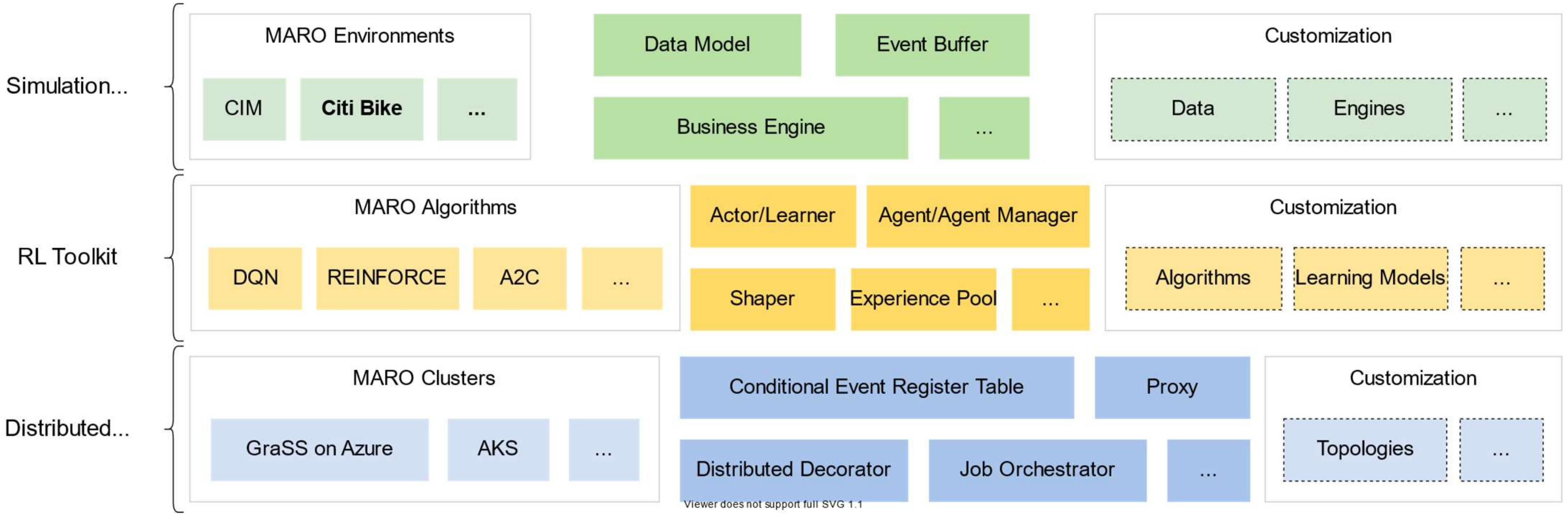
Simulators

Data-driven trace replays for:
- Container inventory management
- Bike repositioning in transportation
- VM provisioning in data centers

Package Features

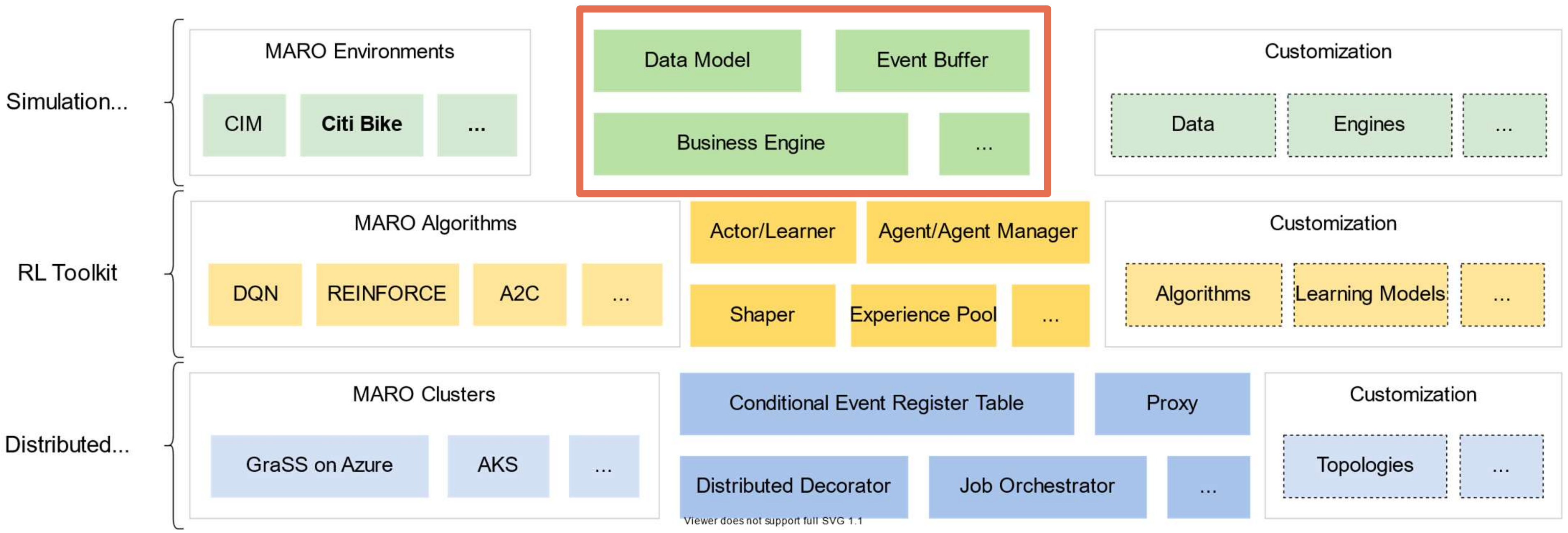
Open-source package containing:
- Simulation toolkit (not in OpenAI Gym API)
- Business engine with real-world and toy / stochastic data traces
- RL Toolkit (training, developing new algorithms)

MARO



Simulation...
RL Toolkit
Distributed...
MARO Environments
CIM
Citi Bike
...
Data Model
Event Buffer
Business Engine
...
Customization
Data
Engines
...
MARO Algorithms
DQN
REINFORCE
A2C
...
Actor/Learner
Agent/Agent Manager
Shaper
Experience Pool
...
Customization
Algorithms
Learning Models
...
MARO Clusters
GraSS on Azure
AKS
...
Conditional Event Register Table
Proxy
Distributed Decorator
Job Orchestrator
...
Customization
Topologies
...
Viewer does not support full SVG 1.1

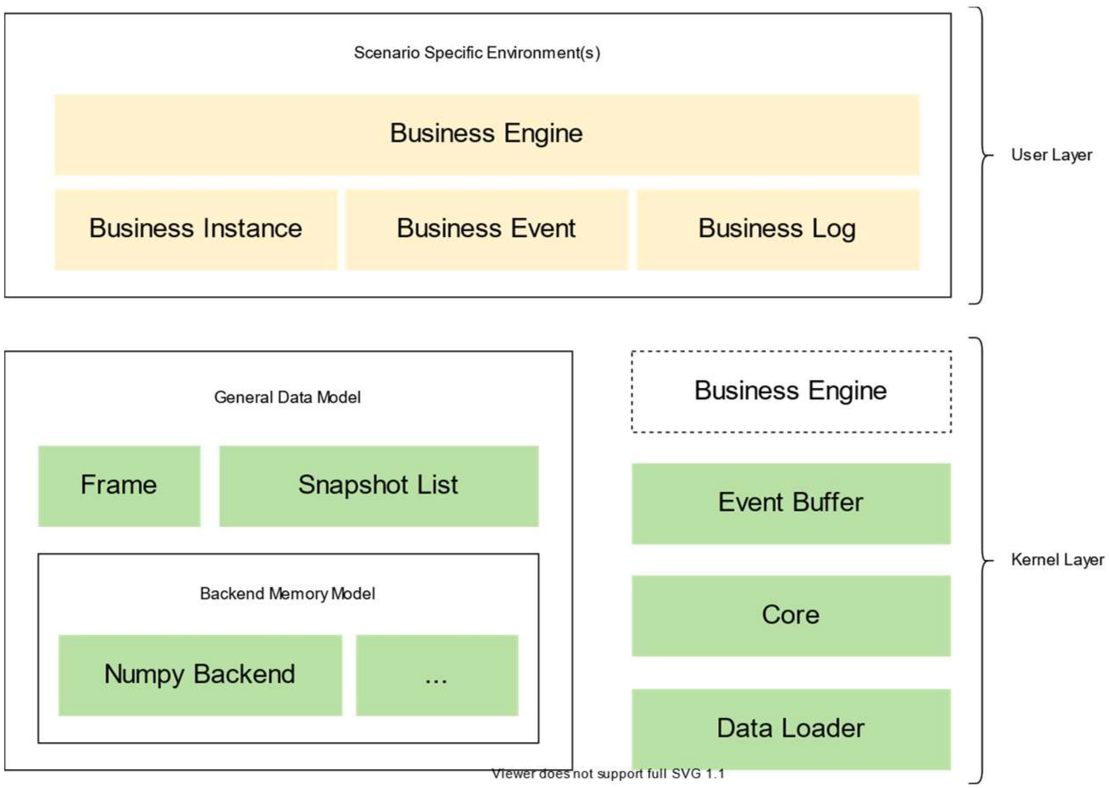
MARO



Simulation...
MARO Environments
CIM
Citi Bike
...
Data Model
Event Buffer
Business Engine
...
Customization
Data
Engines
...
RL Toolkit
MARO Algorithms
DQN
REINFORCE
A2C
...
Actor/Learner
Agent/Agent Manager
Shaper
Experience Pool
...
Customization
Algorithms
Learning Models
...
Distributed...
MARO Clusters
GraSS on Azure
AKS
...
Conditional Event Register Table
Proxy
Distributed Decorator
Job Orchestrator
...
Customization
Topologies
...
Viewer does not support full SVG 1.1

Business Engine

The MARO simulation toolkit is:
- Event driven, more friendly to business logged data
- High execution performance for real-world data (uses Cython under the hood)



Business Engine

Tick / Frame

Notion of time index:
- Most "real-world" problems do not execute in discrete time-steps
- Potential for multiple "events" per tick

Decision Event

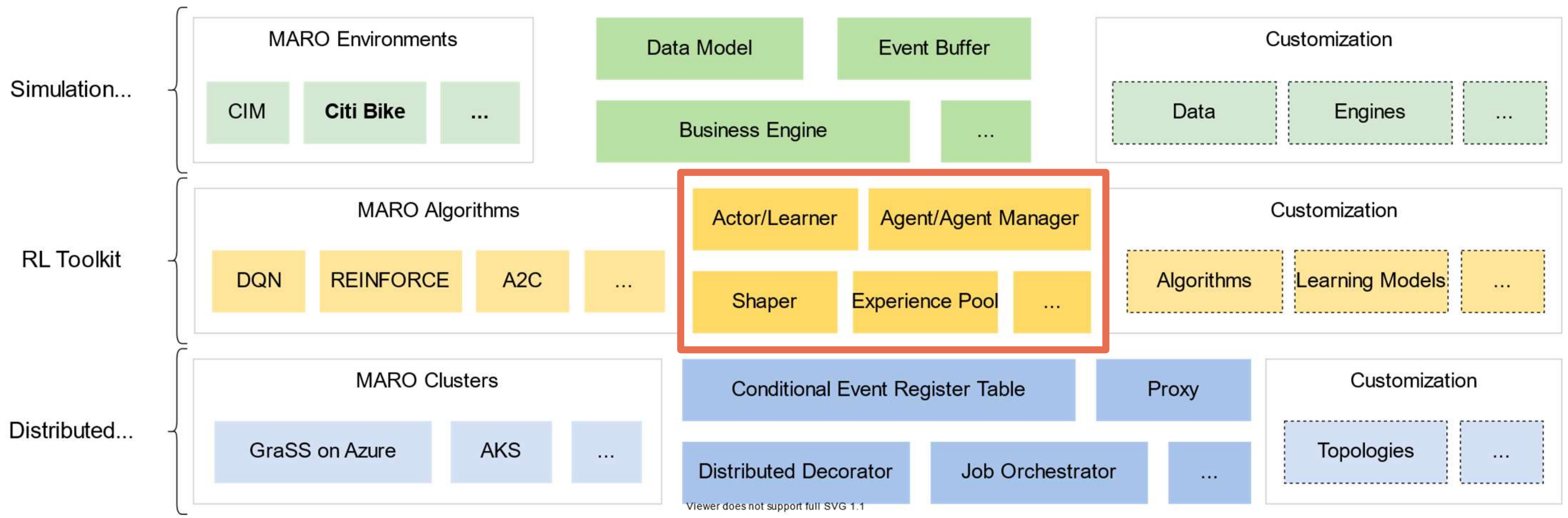
"Issue" that needs to be addressed:
- Current VM Arrival
- Empty / full bike station
- New shipping management request

(exogenous to state of system)

Snapshot List / Frame

State information on current system:
- Physical machines and available capacity
- Set of bike stations and current number of bikes
- List of shipping routes currently in progress

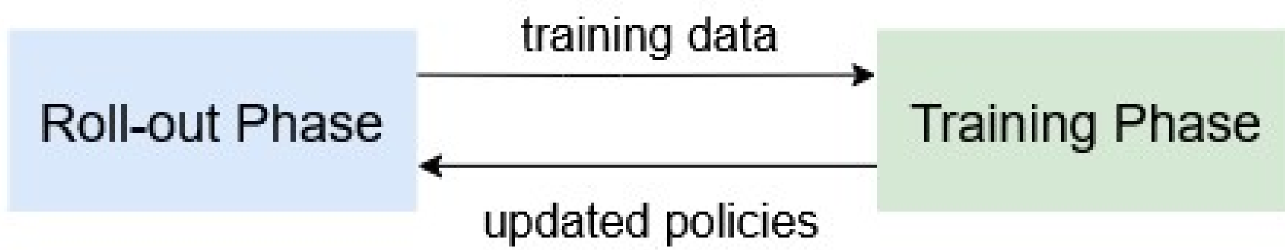
MARO


Simulation...
RL Toolkit
Distributed...
MARO Environments
CIM
Citi Bike
...
Data Model
Event Buffer
Business Engine
...
Customization
Data
Engines
...
MARO Algorithms
DQN
REINFORCE
A2C
...
Actor/Learner
Agent/Agent Manager
Shaper
Experience Pool
...
Customization
Algorithms
Learning Models
...
MARO Clusters
GraSS on Azure
AKS
...
Conditional Event Register Table
Proxy
Distributed Decorator
Job Orchestrator
...
Customization
Topologies
...
Viewer does not support full SVG 1.1

RL Toolkit

The RL toolkit contains:

- Linear, parallel, and distributed workflows for efficient algorithm training
- Code framework for algorithm development, training, neural network features
- Arbitrary “shaper” to take *business data* and transform into (*state*, *action reward)* information to feed into RL algorithms



Agents

The Agent specification:

- Linear, parallel, and distributed workflows for efficient algorithm training
- Code framework for algorithm development, training, neural network features
- Arbitrary “shaper” to take *business data* and transform into *(state*, *action reward)* information to feed into RL algorithms

Only requirement, specify action selection:

```
def choose_action(self, decision_event: DecisionEvent)
```

Takes as input decision event from the business engine, outputs a feasible “Action” object

Environment Sampler

Provides interface for RL algorithms + business engine

- How observations (snapshots) of environment are encoded into state vectors as input into the policy models (“state shaping”)
- How model outputs are converted to action objects defined by the business engine
- How rewards / penalties are evaluated (“reward shaping”)

Domain knowledge, deal with non-Markovian structure of real-world problems

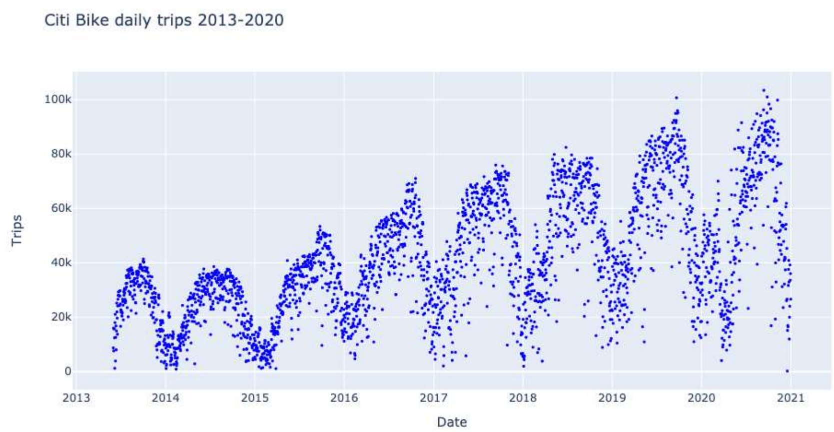
Bike Management

Simulates bike repositioning problem triggered by bike trips from CitiBike.

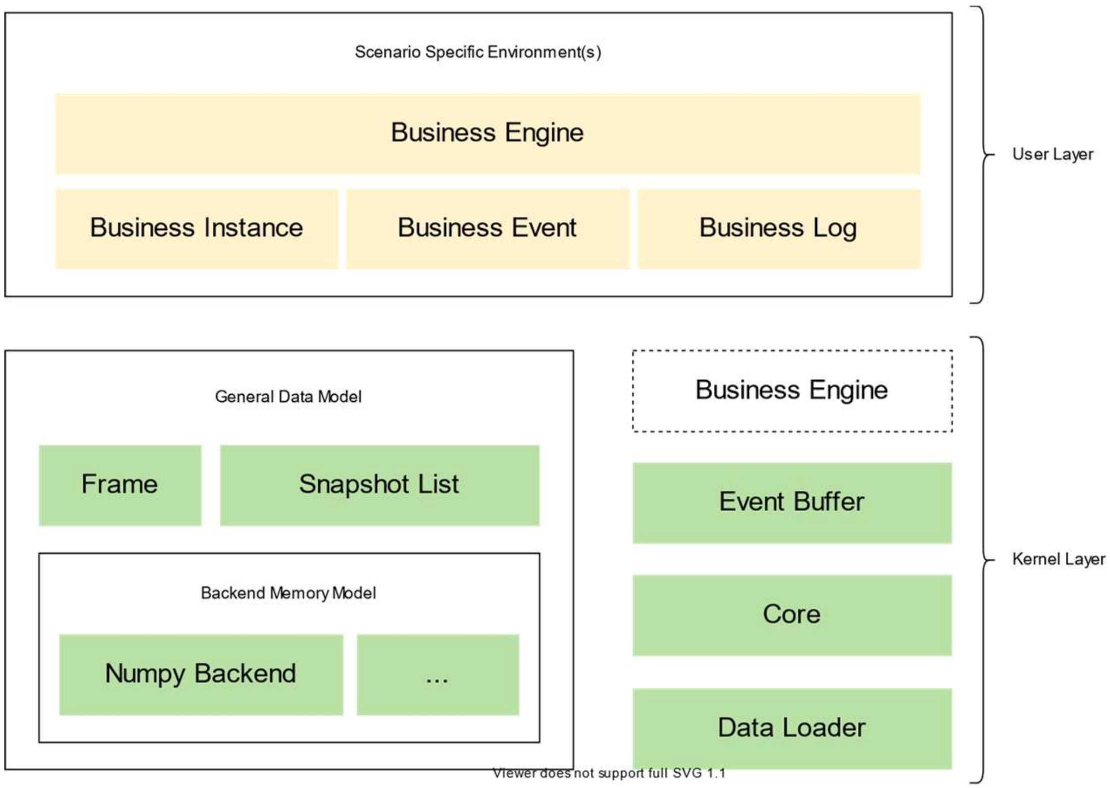
CitiBike:

- New York City's bike sharing system
- Fleet of bikes locked in a network of docking stations throughout city
- Bikes can be unlocked and returned to any other station in system

Demand for bikes and docks dynamically changes throughout the day + season



Bike Management



Decision Event

Two scenarios:

- *Supply*: Too many bikes in the corresponding station
- *Demand*: Too few bikes in the corresponding station

Action

Indicate the station ID and:

- *Departure Station*: ID of source station of bikes
- *To Station*: ID of destination station
- *Number*: Quantity of bikes to move

Bike Management

The snapshot list / frame contains state information:

Event Payload

Previous events:

- *System Actions*: Whether bikes have been rebalanced or delivered between stations and their number
- *Demand Actions*: Trip information with pickup and dropoff location, and times

Station Details

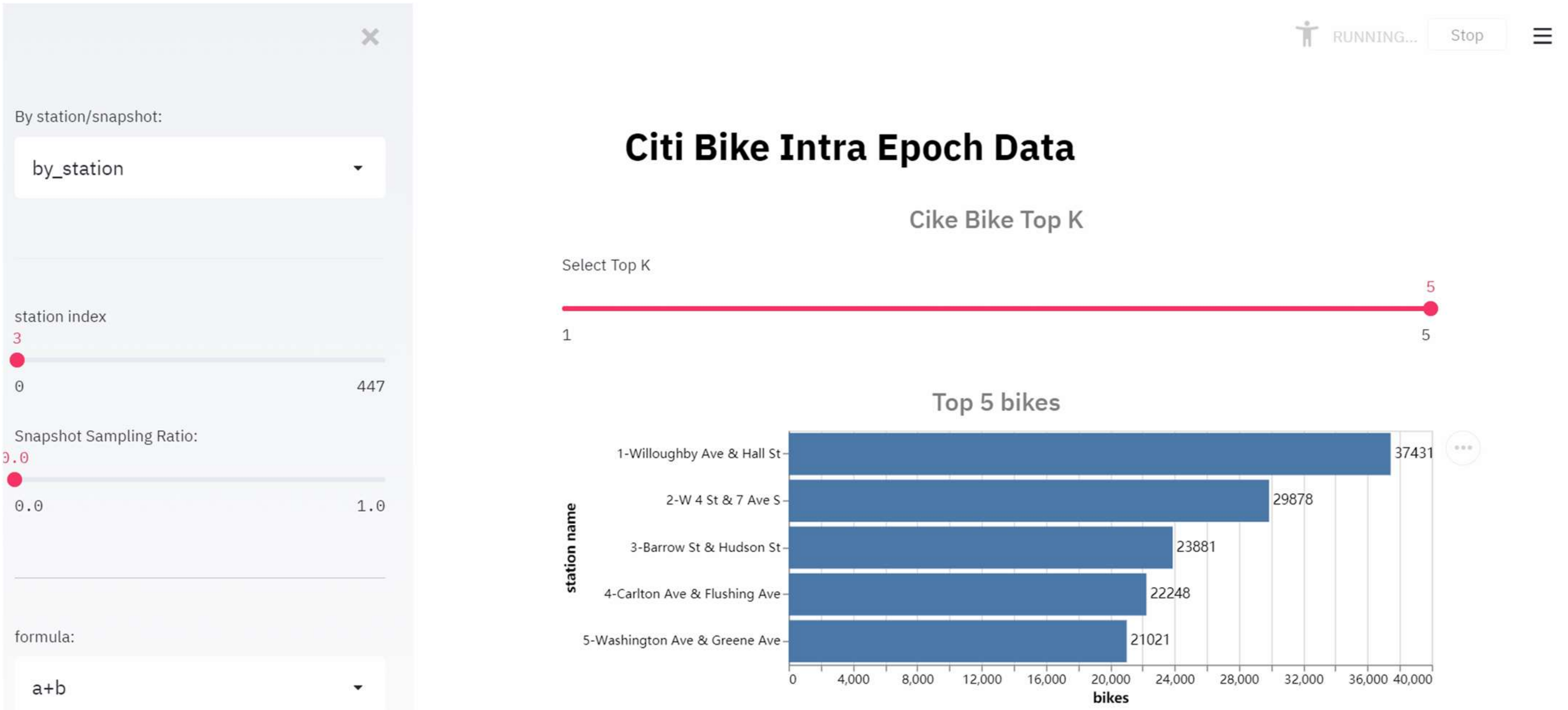
For each station includes their ID and:

- *Bikes*: Current number of bikes
- *Capacity*: Maximum capacity
- *Extra Cost*: Whether station has extra fees associated

Weather information, holiday information, weekdays, …….

We will explore this first before developing RL algorithms

Bike Management



Input Driven (‘Exogenous’) MDP

$$S = X \times \Xi$$

State Space Decomposition

‘Endogenous’ (System State) and ‘Exogenous’ (Arrival State)

System State (X) - Capacity

Exogenous State (Ξ) - Arrival

Input Driven ('Exogenous') MDP

$$S = X \times \Xi$$

State Space Decomposition

'Endogenous' (System State) and 'Exogenous' (Arrival State)

System State (X) - Capacity

Current capacity at each docking location, bikes currently in transit, etc

Exogenous State (Ξ) - Arrival

Current bike trip request

Plan for Today

Nonparametric RL

- "Nonparametric" function approximation
- Strong guarantees across:
  Sample complexity, *space complexity*, *storage complexity*
- "Zooming dimension"

Hindsight Learning

- Exogenous MDPs as model for OR problems
- Use of *Hindsight Planning* oracle for algorithm design
- Empirical results in VM allocation with Microsoft Azure

Tree-Partitions

- Implement tree-based adaptive discretization from nonparametric RL algorithms
- Use ORSuite to test on "continuous Ambulance routing"

DeepRL for CitiBike

- Implement basic deepRL algorithms for CitiBike planning system
- Use MARO – Multi Agent Resource Optimization package from Microsoft Asia
- Visualizations!

Plan for Today

Understand Model

- Explore “topology” (aka scenarios)
- Explore “snapshot list” (aka state information)
- Develop + test heuristic greedy policy
- Look at implementation for Online LP based algorithm

DeepRL for CitiBike

- Implement basic Deep RL algorithms by implementing loss function
- Adjust “environment sampler” to interpret between business engine + policy models

References

[MARO]

- Documentation: https://maro.readthedocs.io/en/latest/index.html
- GitHub: https://github.com/microsoft/maro/

References

https://github.com/seanrsinclair/RLinOperations