ScenarioScript

an open format scenario-describing language

The Simulation Platform for AI and Autonomous Technology

What is a Scenario?

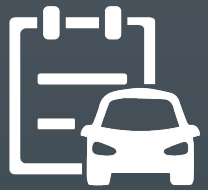
https://youtu.be/H44uQC3Iks8?t=42

Motivation: Why an Open Format for Scenarios?

“Cheaper, Faster, Safer. Together.”

- Automotive testing is transitioning to scenario-based testing
- Scenarios should not have to be remade over and over again
- An open format lets more people participate in scenario authoring
- Stakeholders want to avoid platform lock-in at this early stage in AV development

Philosophy: Where Should We Focus?



Favor the directly observable over model parameters

- Scenarios grounded in reality connect data sets, simulation, and real-world testing
- Does not force simulator developers to adopt the same models

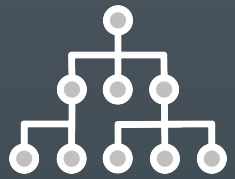
Favor extendability over implementation-specific requirements

- Modeling is unavoidable, but models are not universal
- Creating a space to carry model parameters eases adoptability

Maximize code reuse in programs using the spec

- Stakeholders will extend the format differently, but commonalities increase portability and minimize waste

Design

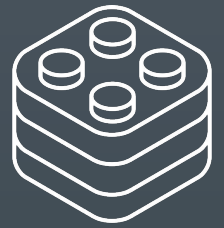


Structure elements

- Skeletal structure
- Ensures code reusability by uniformly locating data

Abstract elements

- Opportunities for adopters to customize or extend spec
- Enables customer-specific implementation and model parameters



Building blocks

- Data structures representing the most common physical quantities
- Ensures code reusability through standardization

The Spec: Top Level

```
// version of the scenario schema
"version": “v0.1”,
```

```
// unique ID for scenario
“uuid”: “43f320d4-aec5-44a6-abf1-0f545d8d2b85”,
```

```
// encapsulate environment
"environment": { },
```

```
// encapsulate dynamic actors
"actors": { },
```

```
// list static obstacles
“static_obstacles”: [],
```

```
// hold company-specific implementation details
"implementation": { }
```

The Spec: Scene

```
"environment": {
    // where the scenario takes place
    "place": { },
    // when the scenario takes place
    "time": { },
    // weather the scenario takes place in
    "weather": { }
},
"actors": {
    // list of vehicles
    "vehicles": [ ],
    // list of pedestrians
    "pedestrians": [ ]
},
"static_obstacles": [
    // static obstacles
]
```

The Spec: Vehicle

```
// a human-readable name
"name": “vehicle_000”,
```

```
// unique ID for the vehicle
“uuid”: “c8c32880-a99b-4875-a91f-1af90dcc256e”,
```

```
// the initial condition
"initial": { },
```

```
// the visual and physical properties
"asset": { },
```

```
// what the vehicle does
“intent”: { },
```

```
// company-specific implementation details for this vehicle
"implementation": { }
```

The Spec: Pedestrian

```
// a human-readable name
"name": “pedestrian_000”,
```

```
// unique ID for the pedestrian
“uuid”: “a1e0c315-0793-45d0-bf3f-1d8c53237c19”,
```

```
// the initial condition
"initial": { },
```

```
// the visual and physical properties
"asset": { },
```

```
// what the pedestrian does
“intent”: { },
```

```
// company-specific implementation details for this pedestrian
"implementation": { }
```

The Spec: Static Obstacle

```
// a human-readable name
"name": “air_conditioner_000”,
```

```
// unique ID for the obstacle
“uuid”: “ccb25291-778c-4b35-b0a7-39dc941db3ab”,
```

```
// the initial condition
"initial": { },
```

```
// the visual and physical properties
"asset": { },
```

```
// company-specific implementation details for this obstacle
"implementation": { }
```

The Spec: Vehicle Intent

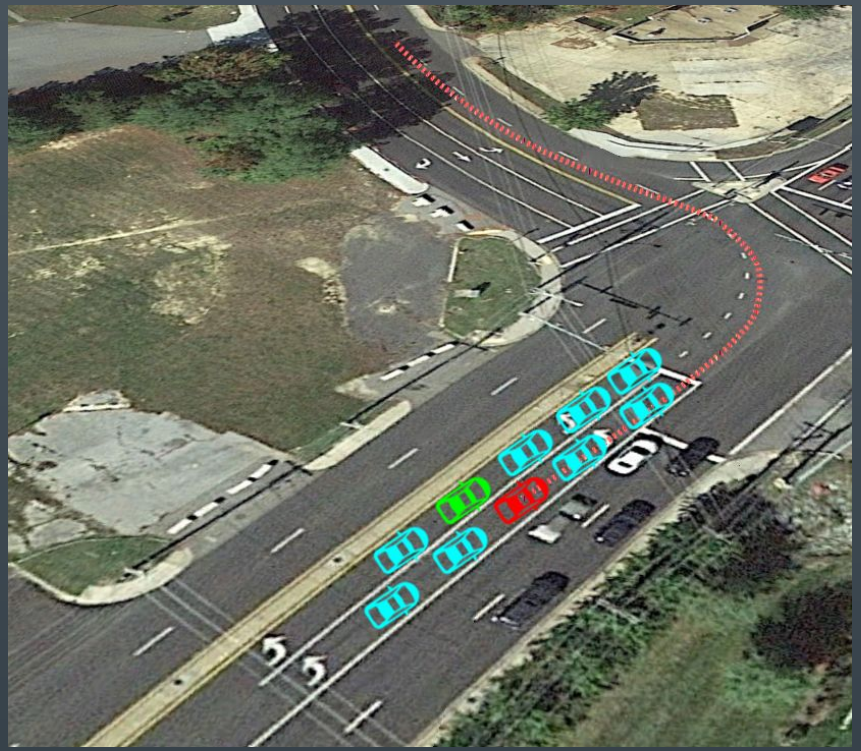
```
// a type that differentiates the intents
"type": “vehicle_intent_wander”,
```

```
// intent-specific fields (possibly abstract)
……..
```

Types:

- deactivated (i.e. parked)
- wander (improvise a route)
- enumerated route (sequence of lanes / drivelines)
- destination route (creates sequence of lanes / drivelines using a routing algorithm)
- reference trajectory (vehicle uses closed-loop controller and physical vehicle model)
- kinematic trajectory (vehicle exactly follows trajectory as function of time)
- user-controlled (vehicle is controlled externally through some interface)

Intent Illustrated Example



https://youtu.be/H44uQC3Iks8?t=42

Wander

User-controlled

Reference trajectory

Trajectory

Note: This is a concept drawing on top of Google Earth data, not a screenshot of RightWorld Studio.

The Spec: Pedestrian Intent

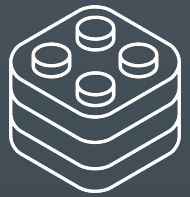
```
// a type that differentiates the intents
"type": “pedestrian_intent_wander”,
```

```
// intent-specific fields (possibly abstract)
……..
```

Types:

- deactivated (i.e. standing still)
- wander (improvise)
- destination route (navigate to point)
- reference trajectory (pedestrian uses closed-loop controller and pedestrian model)
- kinematic trajectory (pedestrian exactly follows trajectory as function of time)
- user-controlled (pedestrian is controlled externally through some interface)

The Spec: Standardization



Some infrastructure object definitions:

- coordinates, rotations
- transforms, projections
- initial condition (location, orientation, velocity, angular velocity, vehicle states)
- trajectories

Standardize

- World coordinate system (treat ENU Cartesian coordinate system as authority, use PROJ4 for others)
- Vehicle coordinate system (ISO 8855, vehicle origin on ground at rear axle center)
- Units (SI base units and radians unless explicitly called out in variable name)

How to Adopt the Format

Adopt structure elements

- choose to refactor your code OR write a converter for your content

Choose where to implement abstract elements

- simulation- and agent-level implementation, place, time, weather, intent models
- substitute abstract blocks in the schema with your object definitions

Confirm compliance

- test examples of your extended format against the schema

Share with friends!

Release

Main schema repository

- JSON schema
- “Base” and “RefCo” examples
- README

AVAILABLE TODAY!

https://github.com/righthook/scenario-script

Java reference code repository

- Shows how to implement and extend spec in Java
- Generates examples

Available next week

Publications

- Medium post
- Whitepaper

Available in coming weeks