




Using Simulation in the Safety Assurance of Autonomous Vehicles

John Redford - Co-founder, Chief Architect & VP Perception



Agenda

- About Five AI
 - Our Approach To Safety Assurance
 - The Need For Simulation
 - The High Fidelity (World And Sensors), End-to-end Approach
 - The Prediction And Planning Only Approach
 - Simulation At Multiple Qualities And Multiple Points In The Stack
 - Coverage (And Coverage Directed Test Generation)
 - Other Uses Of Simulation In AV Development
 - Conclusion
- 



About Five AI

About Five AI

- We are developing the system design and full software stack for a Level 4 Autonomous Vehicle for urban mobility.
- Founded ~4 years ago. ~150 staff. 4 scientific and 1 policy advisors.
- ~\$50M funding to date from European VCs and UK Government.
- Offices in UK: Cambridge, Bristol, Edinburgh, London, Oxford.
- Testing at Millbrook Proving Ground + surrounding public roads and more recently in Bromley and Croydon too.
- Lead partner in Streetwise project, trailing service in London in late 2019 (with Transport for London, Direct Line Group, et al)

Our Current Prototype Vehicles

14 cameras

3 LIDAR

8 RADAR

GPS+IMU

2 Xeon, 8 GPU



Public Road testing since June 2018



On roads around Millbrook Proving Ground, Bedford



On roads between Bromley and Croydon, London



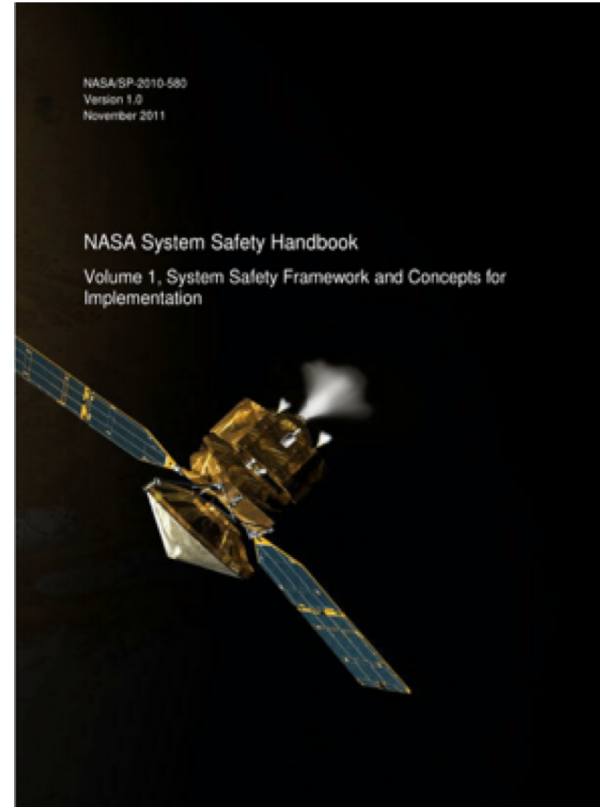
FIVE
AI



Our Approach To Safety Assurance

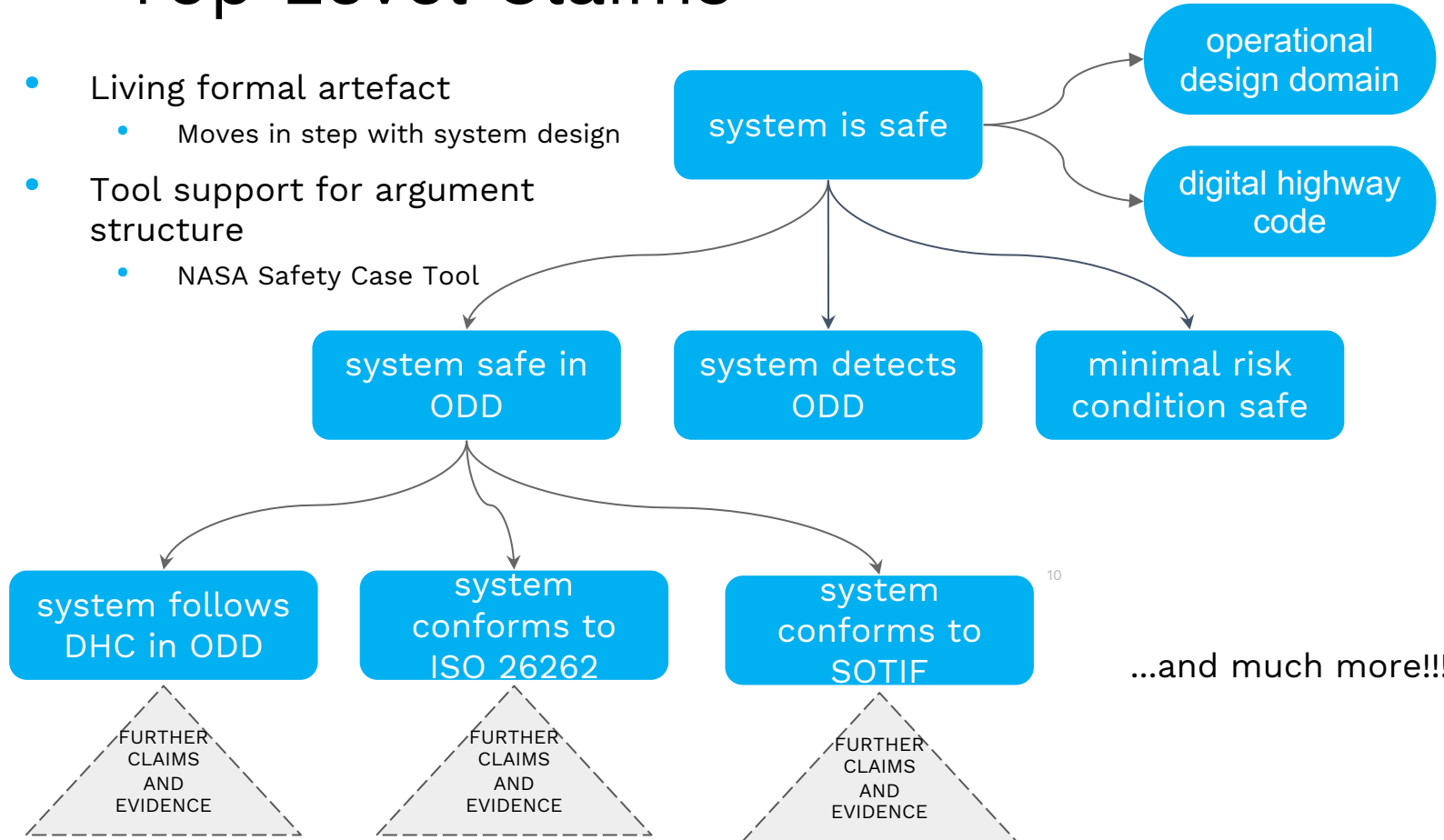
Risk Informed Safety Case

- Focusses on claims, justification and evidence
 - Seeking: a minimal tolerable level of safety
 - Aiming: to be as safe as reasonably practicable
- In the context of a specific operational design domain (ODD)
- Explicit nominal safety from a Digital Highway Code (DHC)
- Not a prescriptive standard, but
 - More than ISO 26262.
 - More than SOFIF (ISO/PAS 21448)
 - Aligns with UL4600



Top Level Claims

- Living formal artefact
 - Moves in step with system design
- Tool support for argument structure
 - NASA Safety Case Tool



Tool Support: Operational Design Domain

– Simulator Must Match...

```
static scene Example is {  
  RoadStructure with {  
    Road is SingleCarriageway with {  
      CentralDividerMarking is BrokenWhiteLineCentralMarking,  
      NearSideRoadsideFeature is Pavement,  
      NearSideRoadEdge is Curb,  
      NearSideRoadsideMarking is DoubleRedLine,  
      RoadGeometry is StraightRoadGeometry,  
      RoadSurface is AsphaltSurface,  
      SpeedLimit is Thirty,  
      FarSideRoadsideFeature is Pavement,  
      FarSideRoadEdge is Curb,  
      FarSideRoadsideMarking is DoubleRedLine,  
      TrafficLanes are {  
        Lane with {  
          LaneNumber is LaneOne,  
          LaneType is IntegratedBikeLane,  
          LaneDirection is EgoDirection  
        },  
        Lane with {  
          LaneNumber is LaneTwo,  
          LaneType is IntegratedBikeLane,  
          LaneDirection is EgoDirection  
        }  
      }  
    },  
    BusStopInLane with {  
      LaneNumber is LaneOne,  
      LaneType is IntegratedBikeLane,  
      LaneDirection is EgoDirection  
    }  
  }  
}
```



```
dynamic odd {  
  for element WeatherCondition we allow [ClearCalm, Windy, LightRain]  
  for element GroundCondition we allow [DryGround, WetGround]  
  for element AirParticulateMatter we do not allow [SensorParticulate, Fog]  
  for element AlteredCondition we do not allow anything //e.g. accident, road works, etc.  
  for element RoadDescription we allow [SingleCarriageway, OneWayStreet, DualCarriageway]  
  for element SceneEnvironmentState we do not allow [SchoolArea, HomeZone, QuietZone, SharedSpace]  
  for element RoadIntersectionFeature we do not allow [StaggeredCrossroads, UnmarkedJunction,  
    LargeRoundabout, SignalledRoundabout]  
  
  //much more below
```


Tool Support: Digital Highway Code

– Simulator Test Oracle Must Match...

```
DrivingInLane : "Generic driving along a lane" {
  atomic LaneFollowing : "Safe lane positioning, moderating speed according to road layout, speed limits and progress."
  atomic VehicleDistanceModeration : "Longitudinal distance and speed moderation from vehicles in the EGO trajectory path"
    attributes Vehicle as VehicleBeingFollowed LongitudinalAction as FollowingVehicleAction
  Associated Rules {
    rule HighwayCodeDistanceModeration : "You should leave enough space between you and the vehicle in front so that you can pull up safely if it suddenly slows down or stops. "
    rule HighwayCodeGap : "You should allow at least a two-second gap between you and the vehicle in front on roads carrying faster-moving traffic and in tunnels where visibility is reduced."
    //...
  }
  atomic VRUDistanceModeration : "Longitudinal distance and speed moderation from VI
    attributes Objects as TheVRU
    relevant when Pedestrian Bicycle Motorcycle
  Associated Rules {
    rule HighwayCodeVRUModeration : "Always leave a 1.5m+ lateral gap between any
    //...
  }
  atomic BeingOvertakenInLane : "When you are being overtaken"
  attributes
    Vehicle as OvertakingVehicle
  Associated Rules {
    rule : "If a driver is trying to overtake you, maintain a steady course and speed, s
    rule : "Never obstruct drivers who wish to pass. Speeding up or driving unpredictal
    rule : "Drop back to maintain a two-second gap if someone overtakes and pulls int
  }
}
attributes
  optional LaneNumber as EgoLaneFollowingLaneNumber default LaneOne
Associated Rules {
  rule : "You MUST NOT exceed the maximum speed limits for the road and for your vel
  rule difficultGeometrySpeed : "The speed limit is the absolute maximum and does not
    the road and traffic conditions is dangerous. You should always reduce your speed
    relevant when HillRoadGeometry CrestRoadGeometry CornerRoadGeometry
//etc. etc.
```

```
static scene firstStreetFromStartToFirstTrafficLight is {
  RoadDescription with {
    SingleCarriageway with {
```

The Applicable Rules are:

- Where a single carriageway has three lanes and the road markings or signs do not give priority to traffic in either direction; use the middle lane only for overtaking or turning right. Remember, you have no more right to use the middle lane than a driver coming from the opposite direction; do not use the right lane
- Where a single carriageway has four or more lanes, use only the lanes that signs or markings indicate
- Unless road signs or markings indicate otherwise, you should use; left lane when going left; right lane when going right; most appropriate when straight ahead.
- You must not drive dangerously
- you must not drive without care and attention
- you must not drive without reasonable consideration for other road users
- You MUST NOT drive on or over a pavement, footpath or bridleway except to gain lawful access to property, or in the case of an emergency.



The Need For Simulation



The Need For Simulation

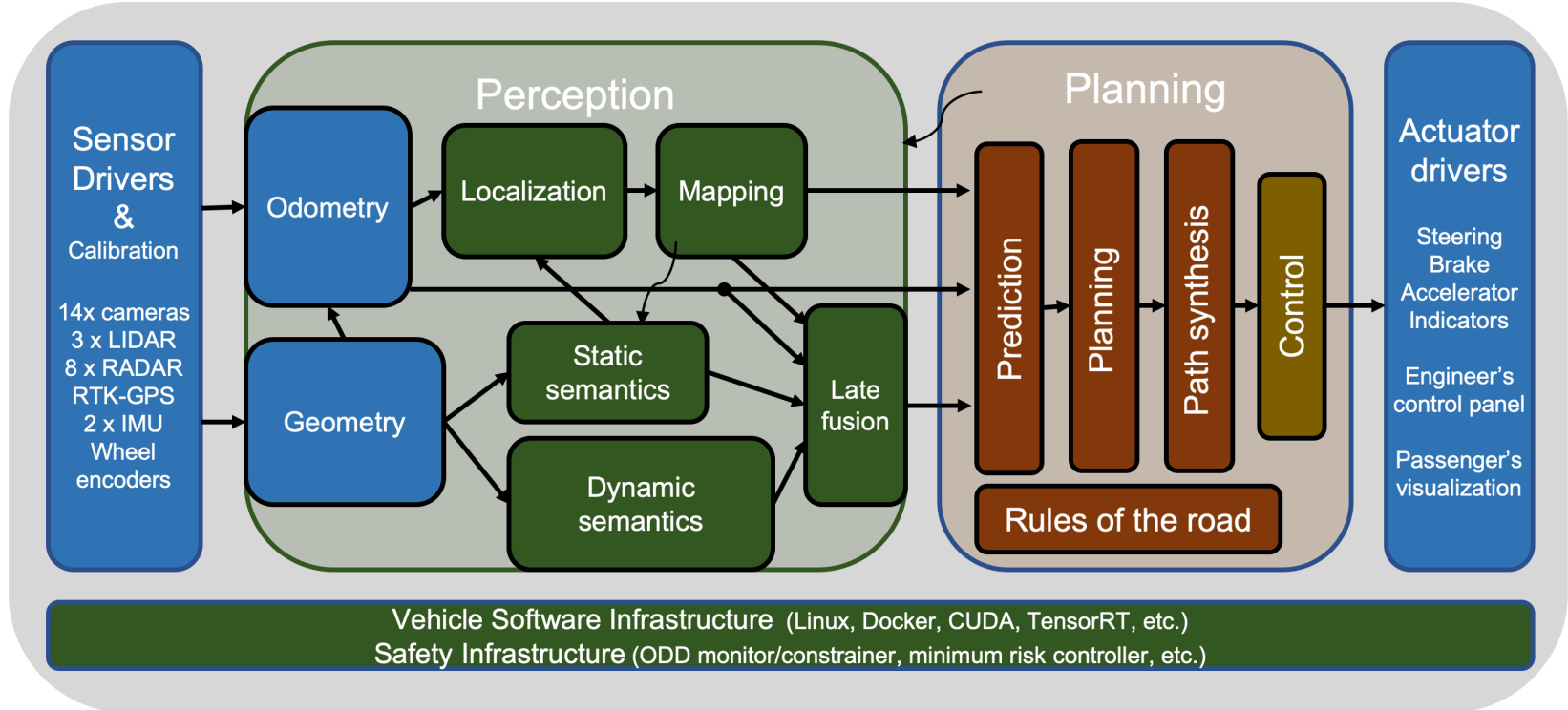
- Clearly not practical to do all testing in the physical world
 - Too many miles need to be driven to get statistically useful data on dangerous situations.
Most miles driven will be uneventful...
 - 2.4 million vehicle miles per personal injury accident
 - 80 million vehicle miles per fatal accident
 - Too dangerous and or costly to arrange tests with serious failures
 - Too costly to carry out multiple tests with slightly varying parameters.
- Department of Transport figures for 2016
 - Injured: 136,621
 - Killed: 1,792
 - All motor vehicles billion miles: 323.7
- Simulation must form a significant part of any safety case!
- Simulation is also useful in development of course
 - Generation of training data for perception.
 - Playpen for algorithm development for planning.



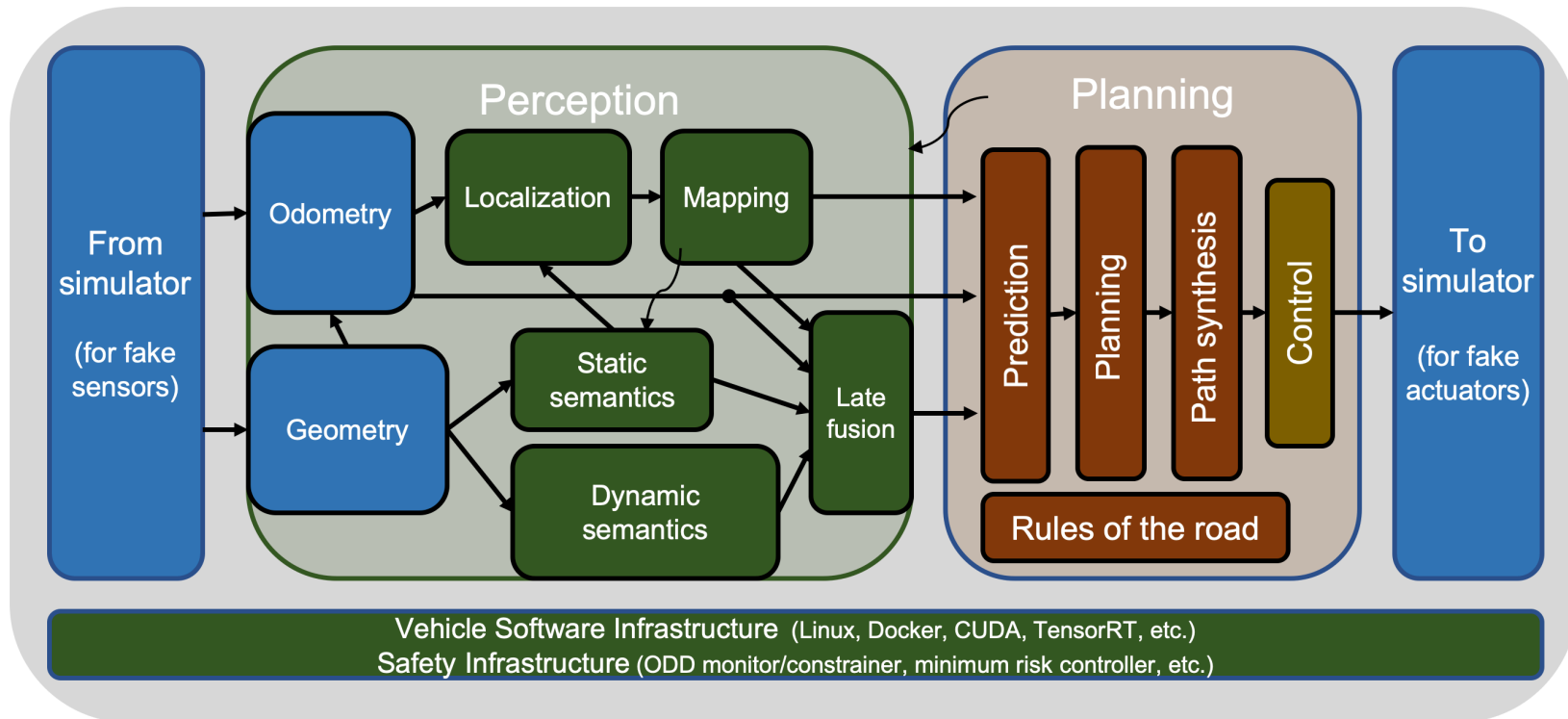
The High Fidelity (World And Sensors), End-to-end Approach



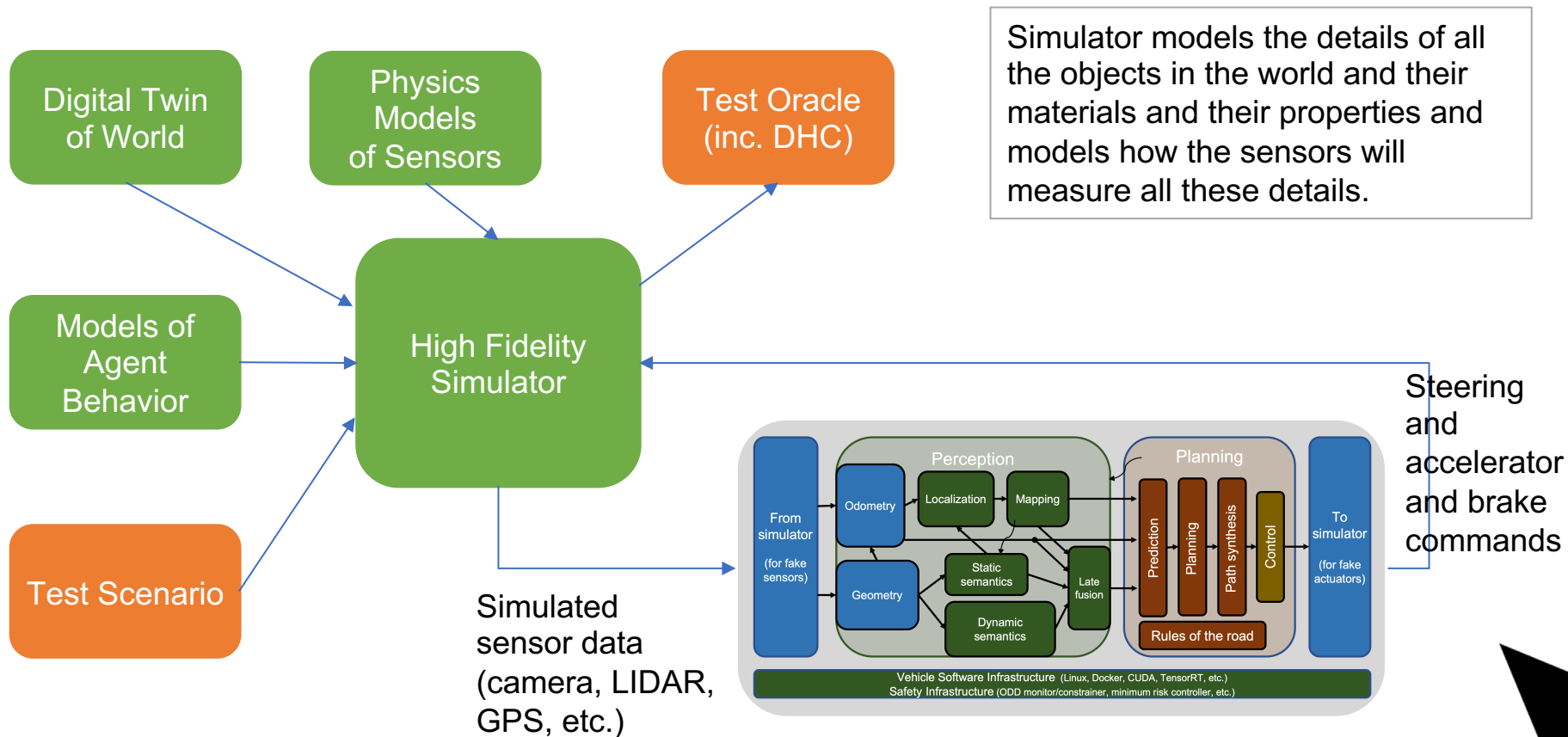
Vehicle Software Stack



Replace Drivers For Use In Simulator



High Fidelity End-to-end Approach



Careful
modelling can
provide
impressively
photorealistic
simulation of
small parts of
the real world.

Centimetre
level accuracy
of road edges
and building
frontages



REAL



SIMULATION

0759: SCOTCH CORNER

0707: HYDE PARK CORNER

0482: BROMPTON ROAD

0585: BEAUFORT GARDENS

person 1.00






High Fidelity Simulation Problems

- Between very difficult and currently impossible to accurately model all the aspects of the world and sensors that matter.
 - Road surface and vehicle dynamics models all possible but complex.
 - GPS, IMU, wheel-encodings models all possible (but error statistics important)
 - Visual appearance and camera lens and image sensor modelling is all reasonably well understood (although not easy to scale to large digital twins)
 - LIDAR modelling similar to cameras (although material properties not the same as for visible light, and the scanning nature of LIDAR adds complications)
 - RADAR returns are very hard to model accurately (material properties, detailed dependence on shapes, multiple reflections, etc.)
 - PERHAPS WORST - Neural networks used for visual object detection are extremely sensitive to detailed image statistics and it is an unsolved problem how to make synthetic images that behave identically to real world images...
- Large amounts of compute are needed for high fidelity simulation of the world and the sensors



High Fidelity Simulation Conclusion

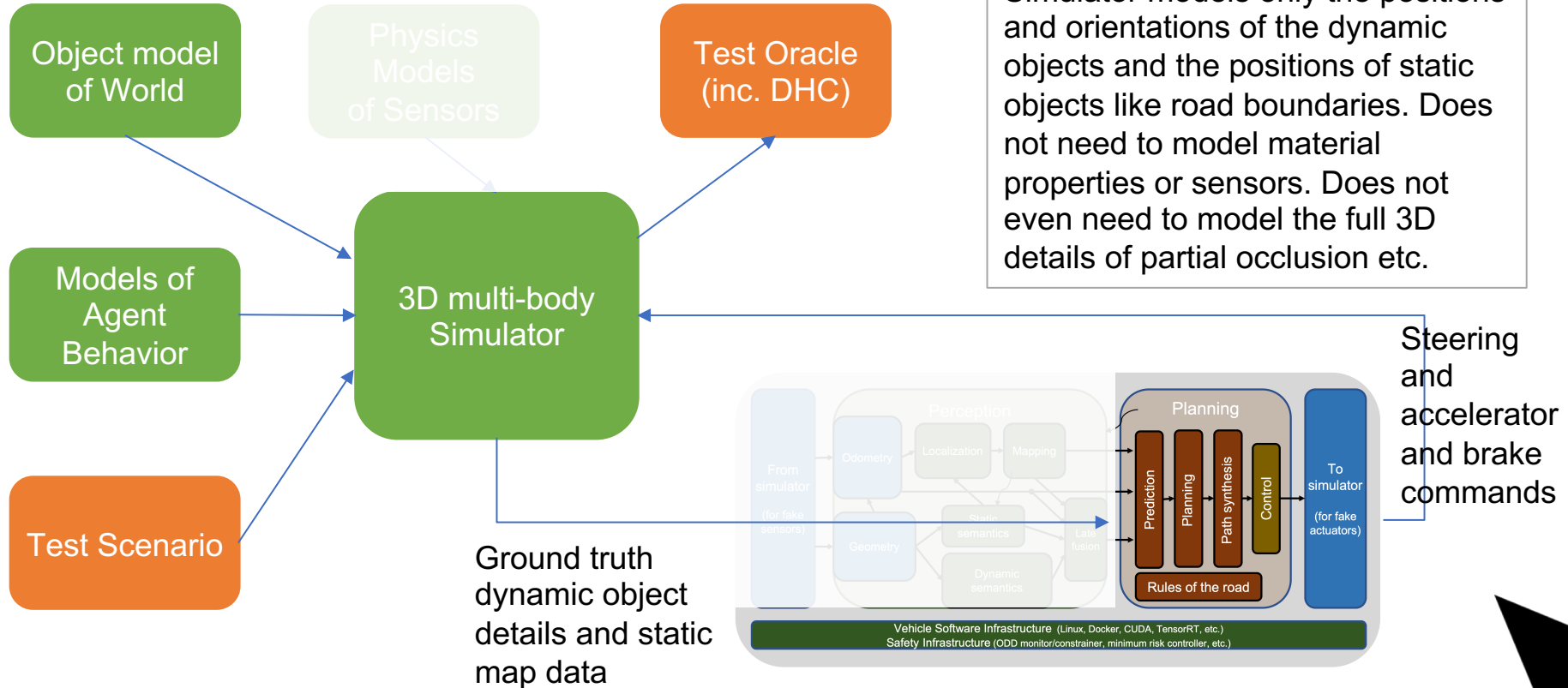
- Valuable for system integration purposes
 - Testing all parts of the stack work together as expected
 - Testing timing and bandwidth issues across whole stack
 - Valuable for testing the non-neural network parts of perception and for testing prediction and planning
 - However there are more efficient ways to do this, and
 - prediction & planning is still perturbed by perception differences
 - Not yet valuable for testing the initial stages of perception
 - The neural network parts of perception do not behave the same in the simulated world as in the real world
- 

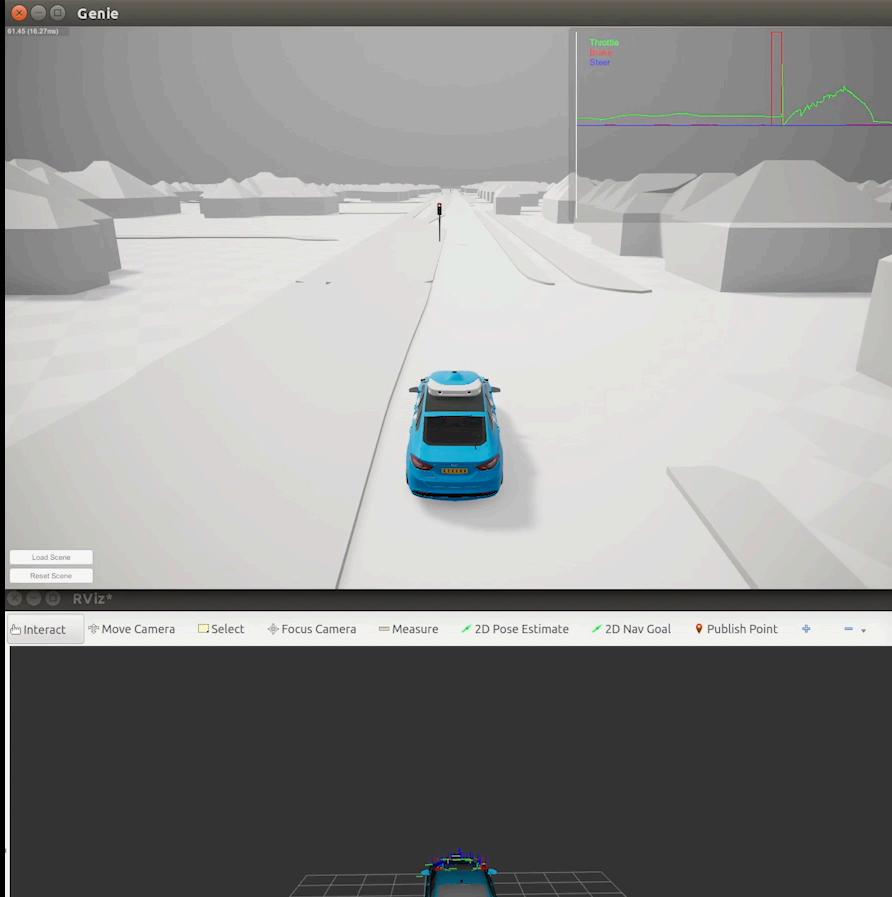


The Prediction And Planning Only Approach



The Prediction + Planning Only Approach



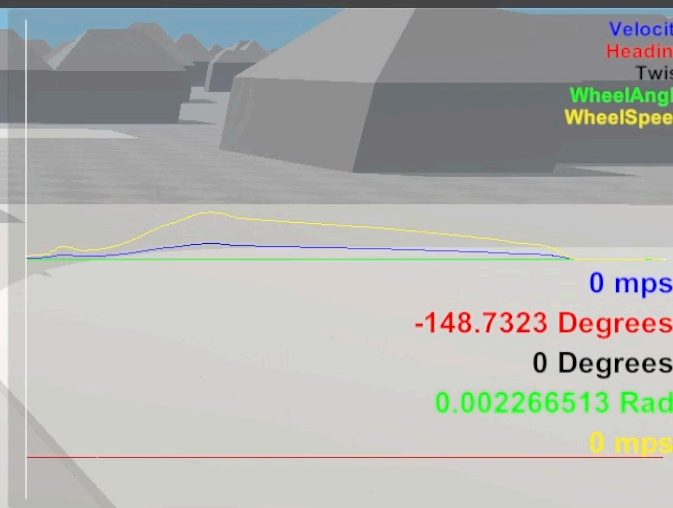


```

simeng... x roscoe... x simeng... x simeng... x simeng... x
otion_planning/trajectory
[1548340696.724942505, 11.165279829]: Waiting for a publisher of trajectory
otion_planning/trajectory
[1548340697.725563271, 12.166666964]: Waiting for a publisher of trajectory
otion_planning/trajectory
[1548340698.689351738, 13.130262442]: waitForService: Service [/driveable_p
driveable_path] is now available.
[1548340698.696566133, 13.137706920]: waitForService: Service [/driveable_p
driveable_path] is now available.
[1548340698.726155506, 13.166749364]: Waiting for a publisher of trajectory
otion_planning/trajectory
[1548340698.801656141, 13.242182016]: [Time To Collision] Not computing tim
lision since odom was not received
[1548340698.887471497, 13.328206433]: [Path Planner] Not requesting traject
e odom was not received
[1548340698.934844366, 13.375142514]: [Time To Collision] Both TF and Odom
n received. Looking for objects...
1548340698.991943, 13.432367]: [traffic_light_node]: DETECTIONS ON
[1548340699.019774670, 13.439905041]: Setting Up Test
[1548340699.292844233, 13.734071305]: VehicleController: Transition from DB
ed to enabled.
[1548340699.295101484, 13.736188399]: goThroughGreenWithoutStopping - Ego-V
eached 0/3 checkpoints and current Behaviour 1
[1548340704.325123974, 18.766128120]: goThroughGreenWithoutStopping - Ego-V
eached 0/3 checkpoints and current Behaviour 1
[1548340708.695470043, 23.136469706]: [Path Planner] Switching to use waypo
nal speeds. Current velocity 12.15 waypoint velocity 12.12. Current accel
.58 waypoint acceleration: -0.02
1548340709.312144, 23.752444]: [traffic_light_node]: DETECTIONS OFF
[1548340709.325631773, 23.766199999]: goThroughGreenWithoutStopping - Ego-V
eached 1/3 checkpoints and current Behaviour 1
[1548340714.325133034, 28.766298467]: goThroughGreenWithoutStopping - Ego-V
eached 2/3 checkpoints and current Behaviour 1
1548340714.881311, 29.322043]: Controller Spawner couldn't find the expecte
ller_manager ROS interface.
1548340719.115184, 33.556107]: [traffic_light_node]: DETECTIONS ON
[1548340719.325292489, 33.766356415]: goThroughGreenWithoutStopping - Ego-V
eached 2/3 checkpoints and current Behaviour 1
[1548340724.358252042, 38.799404294]: goThroughGreenWithoutStopping - Ego-V
eached 2/3 checkpoints and current Behaviour 1
1548340725.823071, 40.264091]: [traffic_light_node]: DETECTIONS OFF
[1548340727.957998697, 42.398667413]: All checkpoints reached on traffic li
ck!
[1548340727.958109203, 42.398667413]: VehicleController: Transition from DB
d to disabled.
[1548340727.980546178, 42.420957977]: Waiting for Traffic Lights Integratio
o be ready...
[1548340727.993064331, 42.433683751]: [Path Planner] Switching to use waypo
nal speeds. Current velocity 12.13 waypoint velocity 12.11. Current accel
on: -0.02
[1548340727.993064331, 42.433683751]: Setting Up Test
[1548340727.993064331, 42.433683751]: [traffic_light_node]: DETECTIONS ON
[1548340727.993064331, 42.433683751]: Switching light to red
[1548340727.993064331, 42.433683751]: VehicleController: Transition from DB
d to enabled.
[1548340727.993064331, 42.433683751]: stopAtRedGoAtGreen - Ego-Vehicle Reac
Behaviour 1
[1548340734.383675237, 48.823849580]: stopAtRedGoAtGreen - Ego-Vehicle Reac
checkpoints and current Behaviour 2

```

traffic light tests: red light, green light



The Prediction + Planning Only Approach

Pros and Cons

- Pro - Much more compute efficient compared to high fidelity simulation
 - No need for detailed simulation of appearance of the world
 - No need for detailed modelling of sensors
 - No need to run the perception part of the vehicle stack (which tends to be the most computationally costly part of the stack)
- Pro - Provides an isolated test of the prediction and planning part of the vehicle stack.
- Major Con - The prediction and planning testing is unrealistic since its input is as if the perception system is perfect, which in the the real world it is of course not.



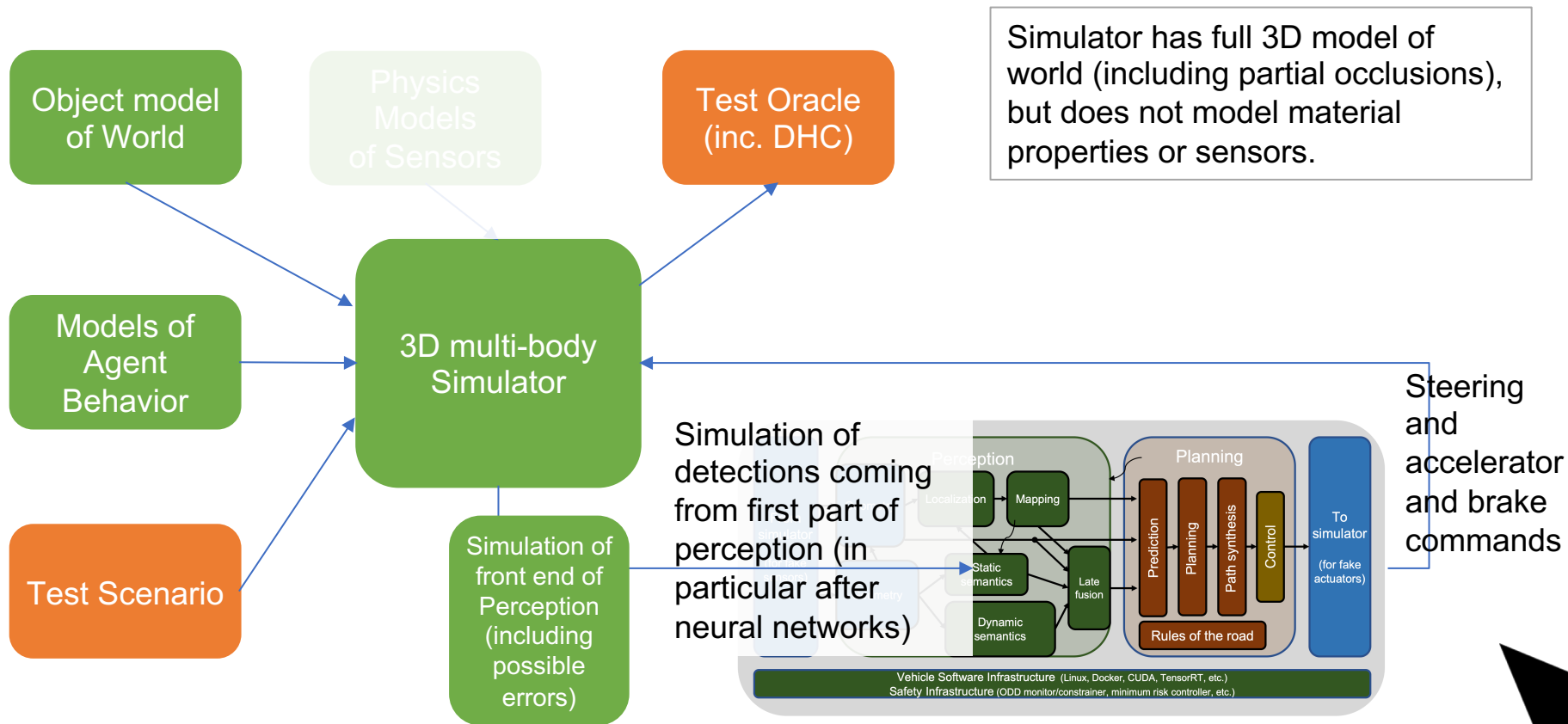
Simulation At Multiple Qualities And Multiple Points In The Stack



Example Levels Of Fidelity

- E.g. consider localisation, in a system that uses LIDAR localisation. There are many possible levels of simulation fidelity:
 1. Model the LIDAR reflectivity of all objects in the simulated world and their surface angles relative to the LIDAR beams. Model the rotating nature of the scan of the LIDAR over time and the stream of samples generated.
 2. Model the distance to objects in the simulated world and not worry about the exact details of reflectivity of materials. Still model the scanning nature.
 3. Model a fixed in time snap-shot of the LIDAR point cloud (ignoring scanning)
 4. Model the position in the LIDAR map that this point cloud would generate. I.e. don't model LIDAR, just model the 3D pose of the simulated vehicle in the simulated world.
- All these levels of simulation have value for different purposes!
- Each sensor or detection algorithm has similar options!

E.g. Partial Perception Approach

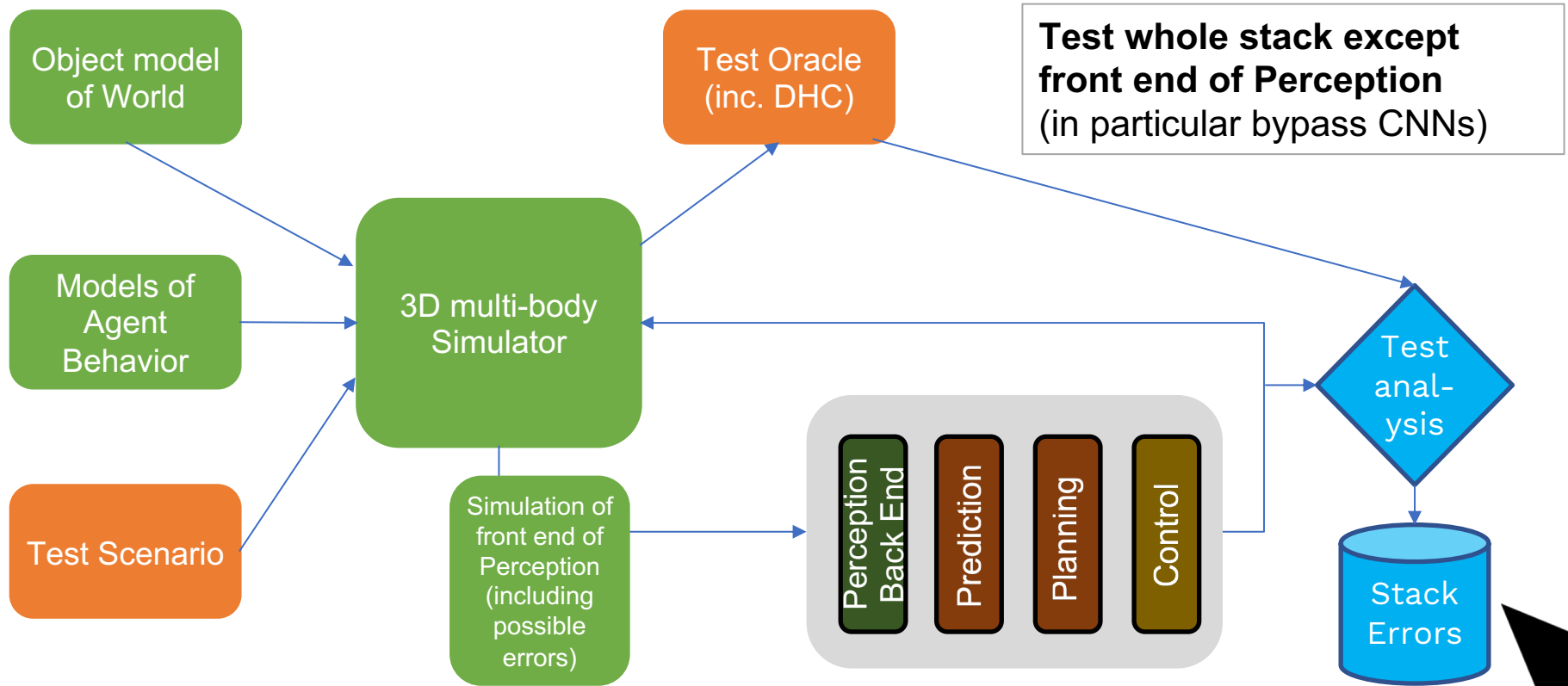


E.g. Partial Perception Approach

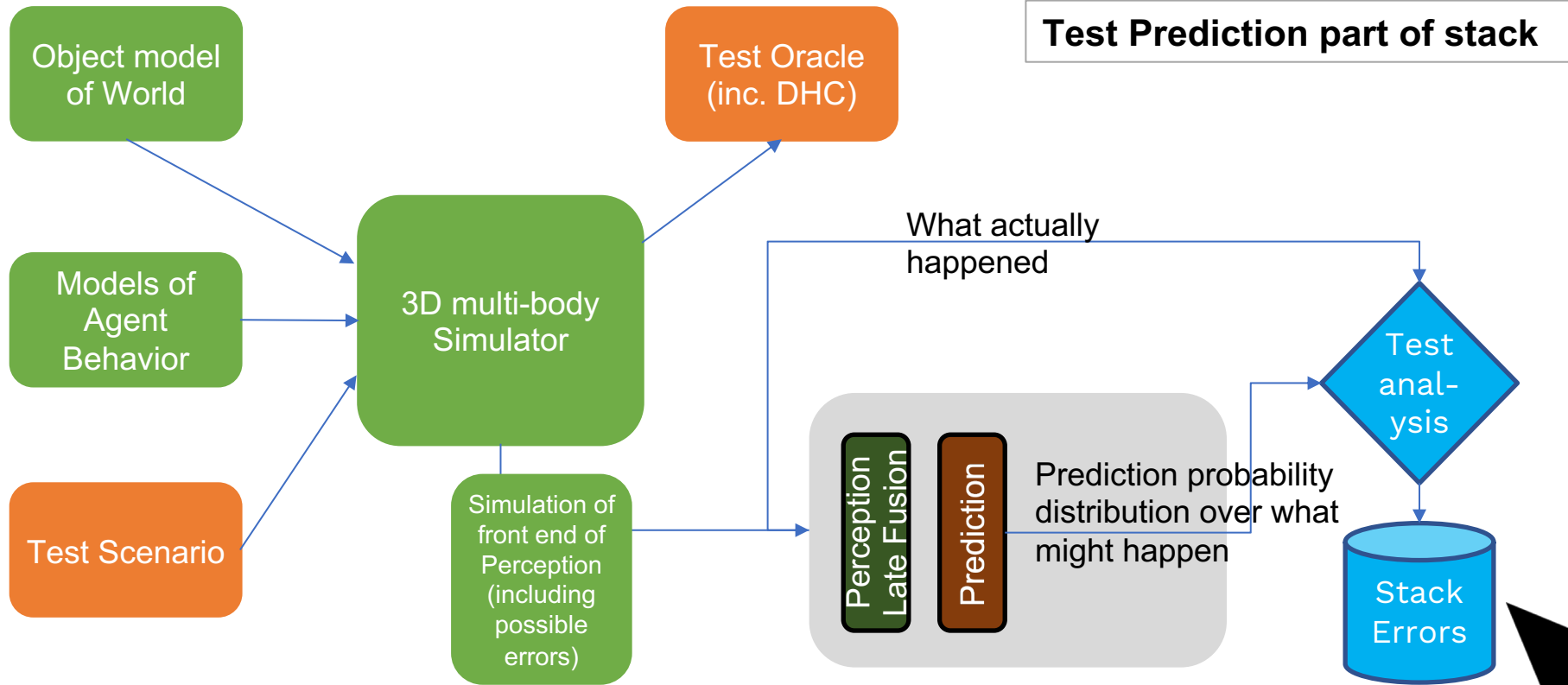
Pros and Cons

- Pro – Still much more compute efficient compared to high fidelity simulation
 - Still no need for detailed simulation of appearance of the world
 - Still no need for detailed modelling of sensors
 - No need to run the CNN part of the perception part of the vehicle stack
- Pro – Tests the fusion and tracking stages of perception
 - Have the ‘simulation of the front end of perception’ only simulate the frame at a time performance of the initial stages of perception. Leave the fusion over time and fusion across sensors stages of perception in the system under test.
- Pro – Provides realistic input data to prediction and planning.
- Pro + Con – Has statistically the same errors as the real world, but not exactly the same errors

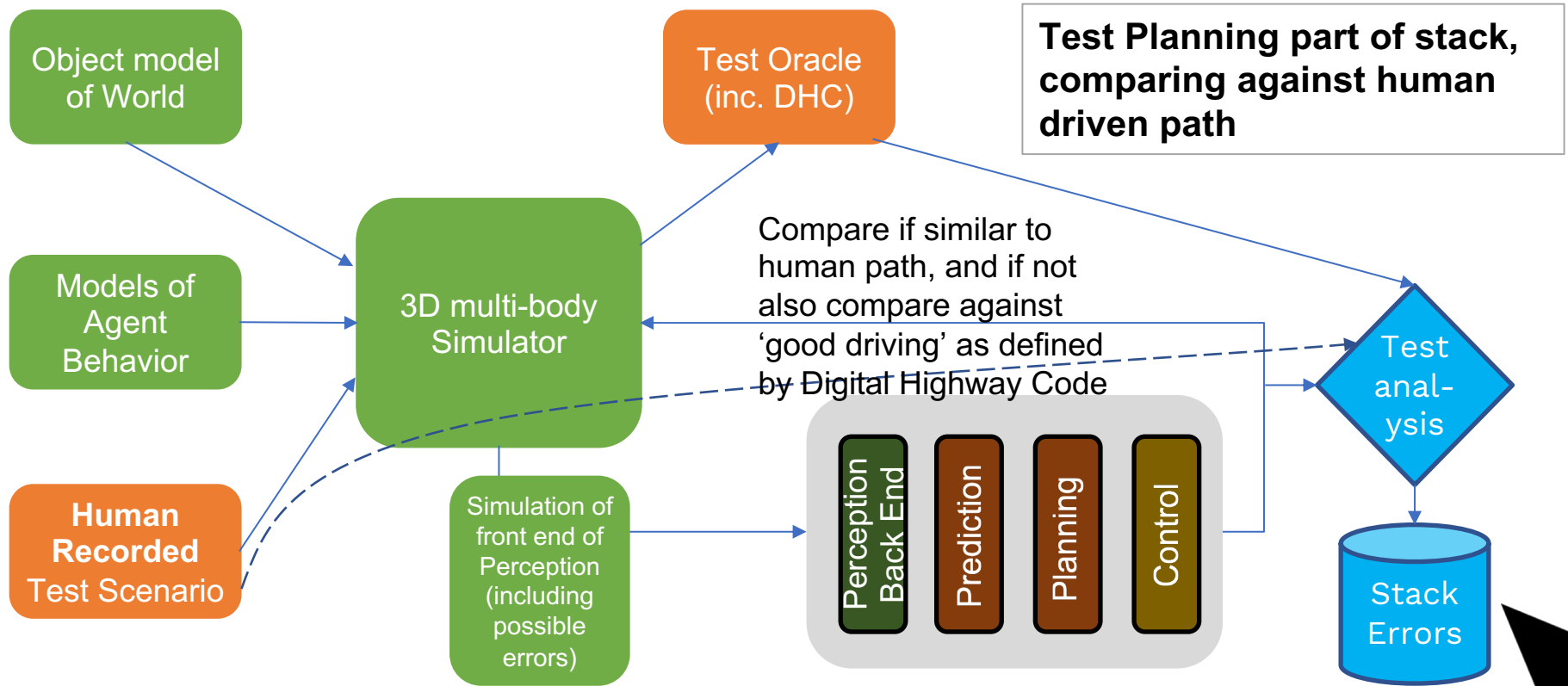
Many Useful Test Configurations (1)




Many Useful Test Configurations (2)



Many Useful Test Configurations (3)



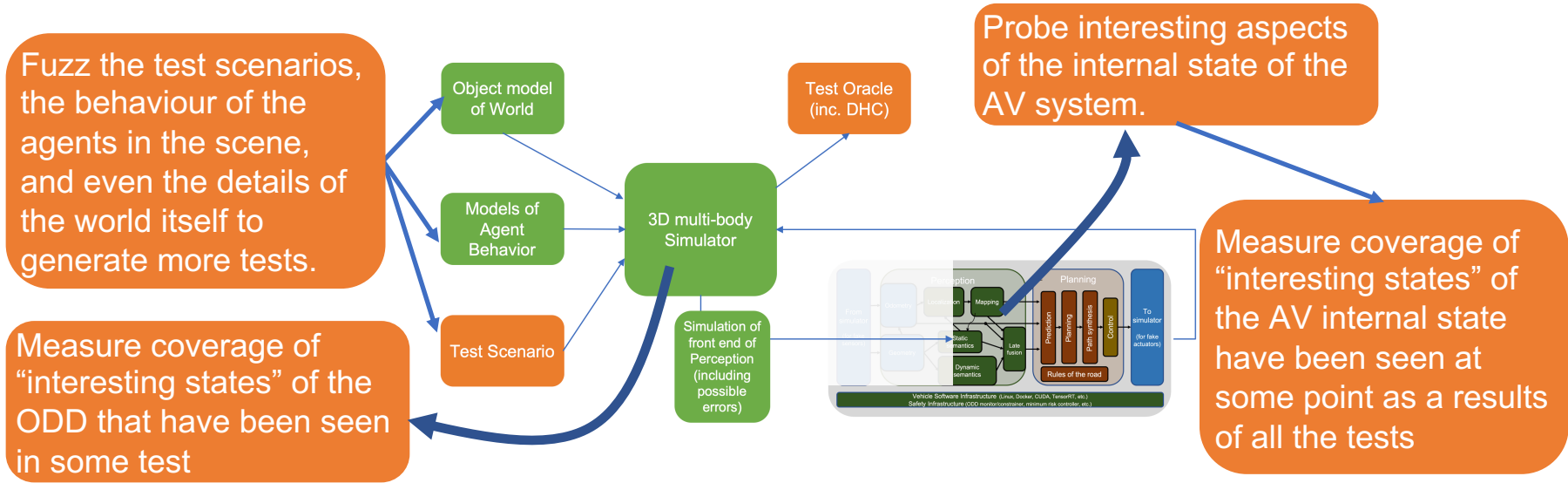


Coverage (And Coverage Directed Test Generation)

Can Target Several Types of Problems...

- Can use simulation to target different types of failure cases:
 - Cases in the environment likely to be dangerous for all AV systems
 - E.g. blinding sunlight into cameras, bizarre behaviour by other drivers, etc, etc, etc.
 - **I.e. coverage of the ODD** (guided by ontology of the objects and behaviours in the ODD)
 - Cases in the environment that trigger failures in a specific AV design that are likely to be dangerous
 - E.g. patterns of detection and lack of detection of objects that cause errors in an object tracking module, etc. etc. etc.
 - **I.e. coverage of the internal state of the AV system software** (guided by knowledge of the important modules and weaknesses in the AV system software design)
- Can use test data and scenarios generated from
 - The real world, real accident reports, etc.
 - Random 'fuzzing' around real world scenarios
 - Hand crafted scenarios.
 - Randomly generated scenarios.
 - Directed random scenario generation
 - AI guided generation of scenarios

External and Internal State Coverage



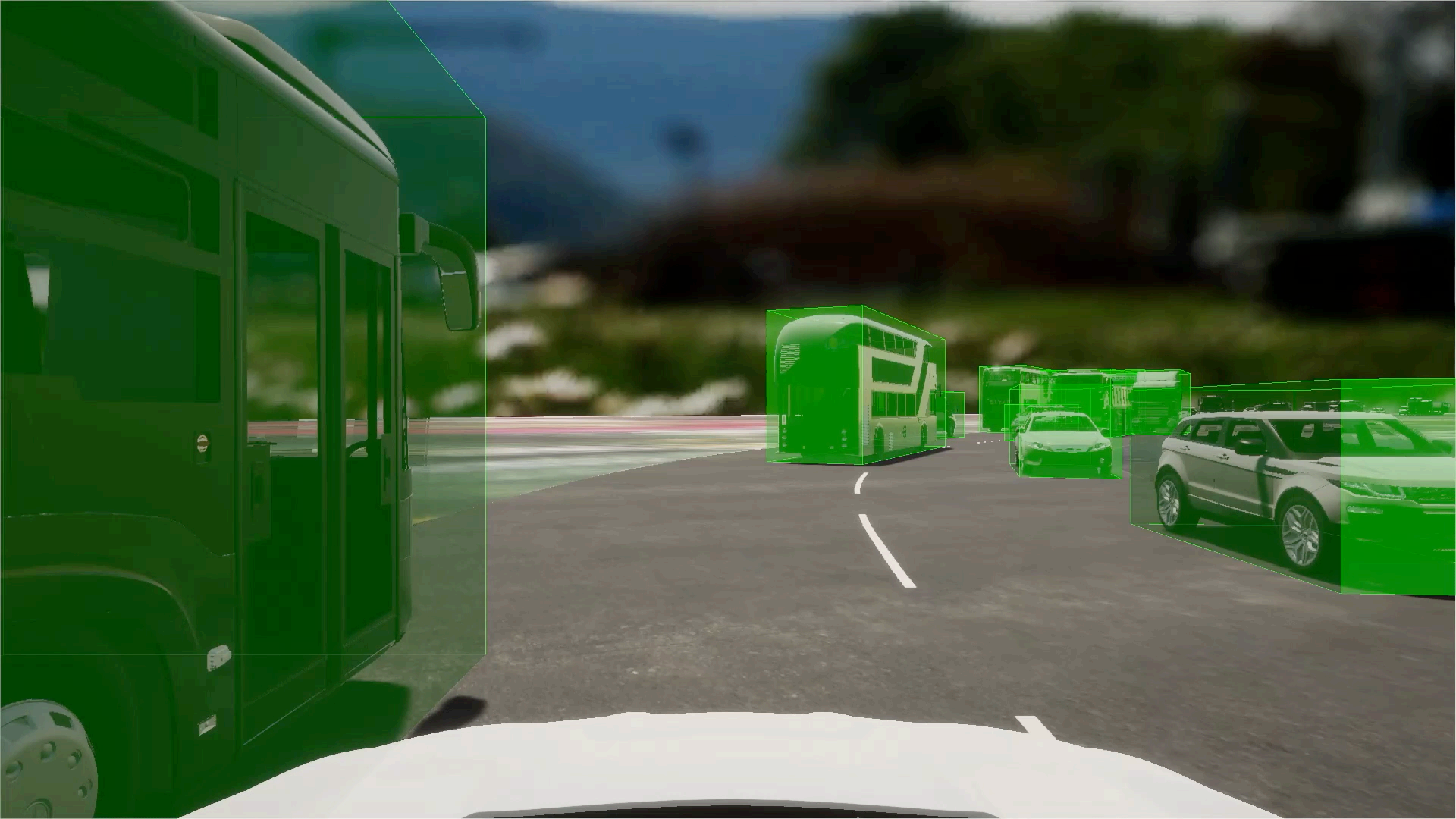
- The external state in the ODD is infinite and the internal state of the AV system is near infinite, so it is obviously impossible to check all states have been seen.
 - Coverage metrics need to be defined by people intimately familiar with the ODD and intimately familiar with the internal structure, and likely weaknesses, of the AV software.
- Coverage directed test generation is another wrapper again around the above test infrastructure and imposes another significant set of requirements on simulators...



Other Uses Of Simulation In AV Development

Training Data Generation





Visualisation

6.5 m/s

Velocity

Steering Angle

Autonomy State

0,00 1,77 3,54 5,31 7,08 8,85 10,62 12,39 14,16 15,93 17,70 19,47 21,24 23,01 24,78 26,55 28,32 30,09 09:00:05

08/04/19



Conclusion

Conclusion

- **Safety Case** – Safety assurance of a complex system like autonomous vehicles should be a huge set of carefully constructed, justifiable, reviewable arguments.
- **Simulation is essential** – Testing purely in the real world is not practical due to the huge number and variety of individually rare problem situations.
- **High fidelity modelling is useful but inadequate** – In practice it is impossible to model the real world and the sensors sufficiently accurately to guarantee to generate exactly the same errors as in the real world.
- **Low fidelity modelling of just planning is useful but inadequate** – Testing prediction and planning without relevant sensor and perception errors won't generate the same errors as in the real world.
- **Simulation is needed at multiple levels of fidelity, with test data inserted at multiple points in the vehicle software stack, and all simulation data needs to have realistic error distributions.**
- Simulation technology is useful for many other aspects of development.



Thank You

john@five.ai

