foretellix
verifying autonomous systems

Securing America's Future Energy

Measurable Safety – A Metric Driven Approach for Safety Assessment And Rating of AVs

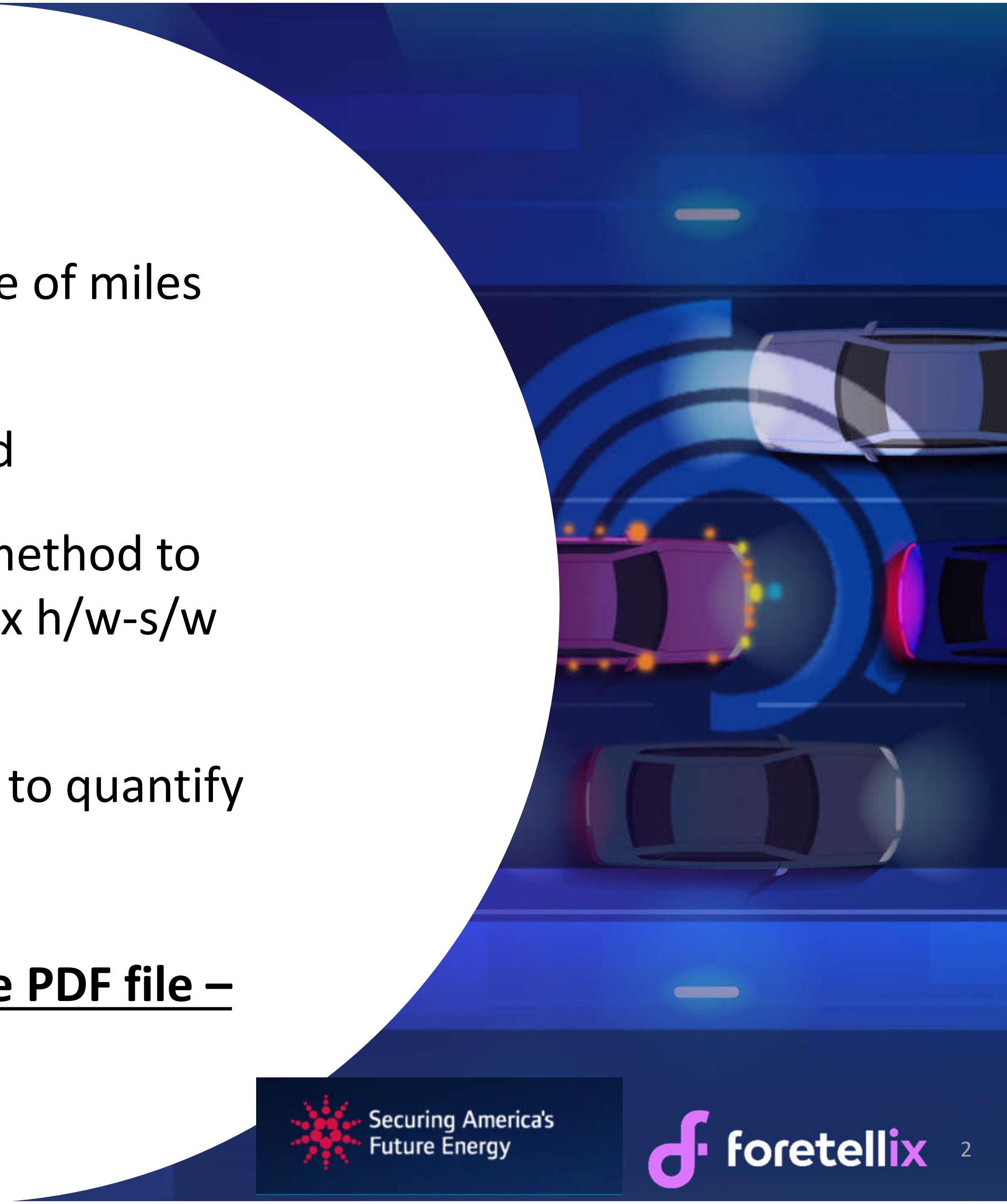
CDV – Coverage Driven verification

Gil Amid (Gil.Amid@Foretellix.com)
SAFE – Autonomous Vehicle Initiative
avi@secureenergy.org

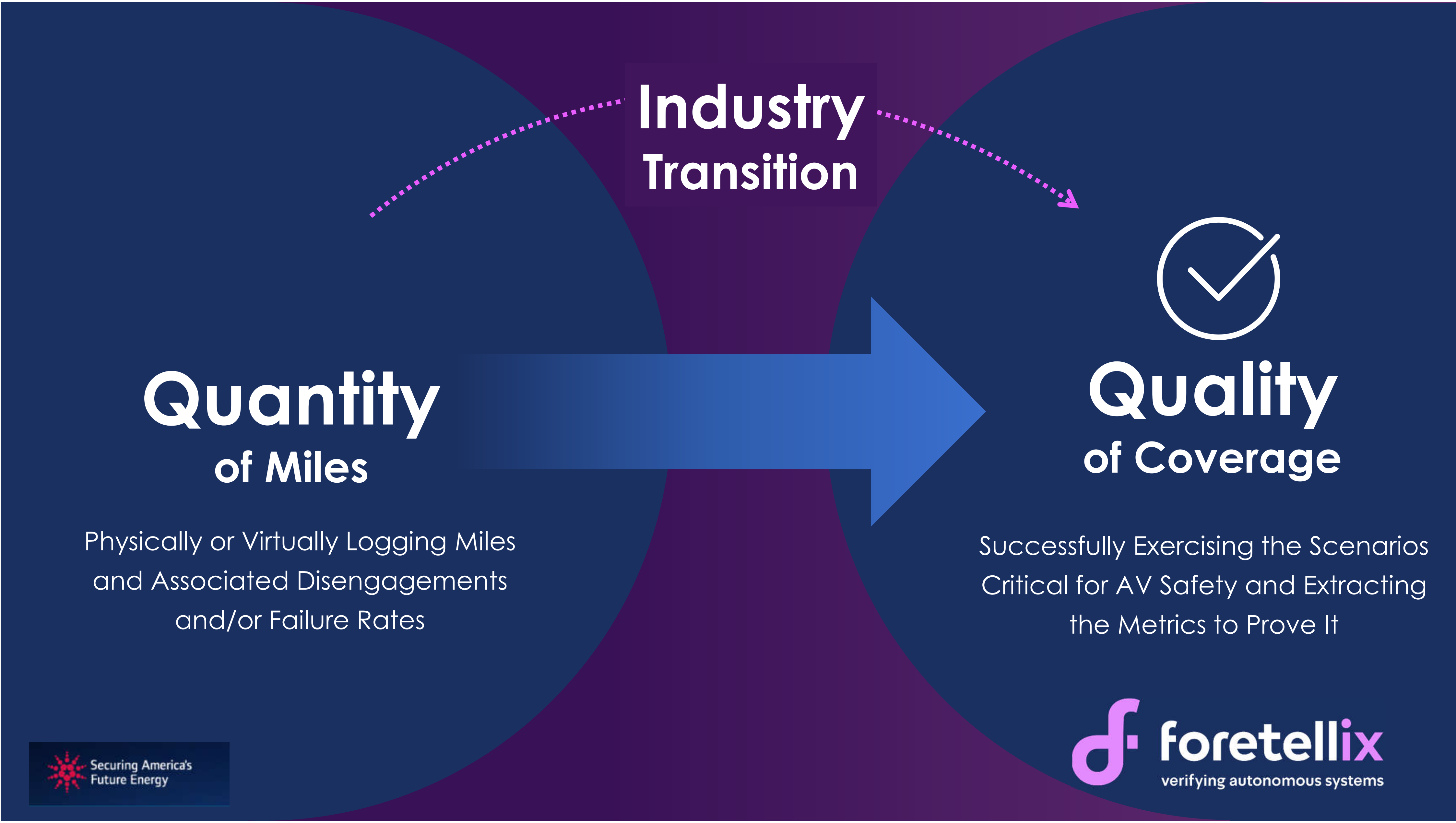
Key Messages

- AV Safety needs to be quantifiable – usage of miles and disengagement is insufficient
- AV Safety can be measured and quantified
- Coverage Driven Verification is a proven method to measure and quantify maturity of complex h/w-s/w systems
- Coverage metrics can and should be used to quantify AVs safety

<ins>**The full, detailed presentation is available in the PDF file – following the “backup slides” title.**</ins>



Securing America's Future Energy



Industry Transition

Quantity

of Miles

Physically or Virtually Logging Miles and Associated Disengagements and/or Failure Rates

Quality

of Coverage

Successfully Exercising the Scenarios Critical for AV Safety and Extracting the Metrics to Prove It

Securing America's Future Energy

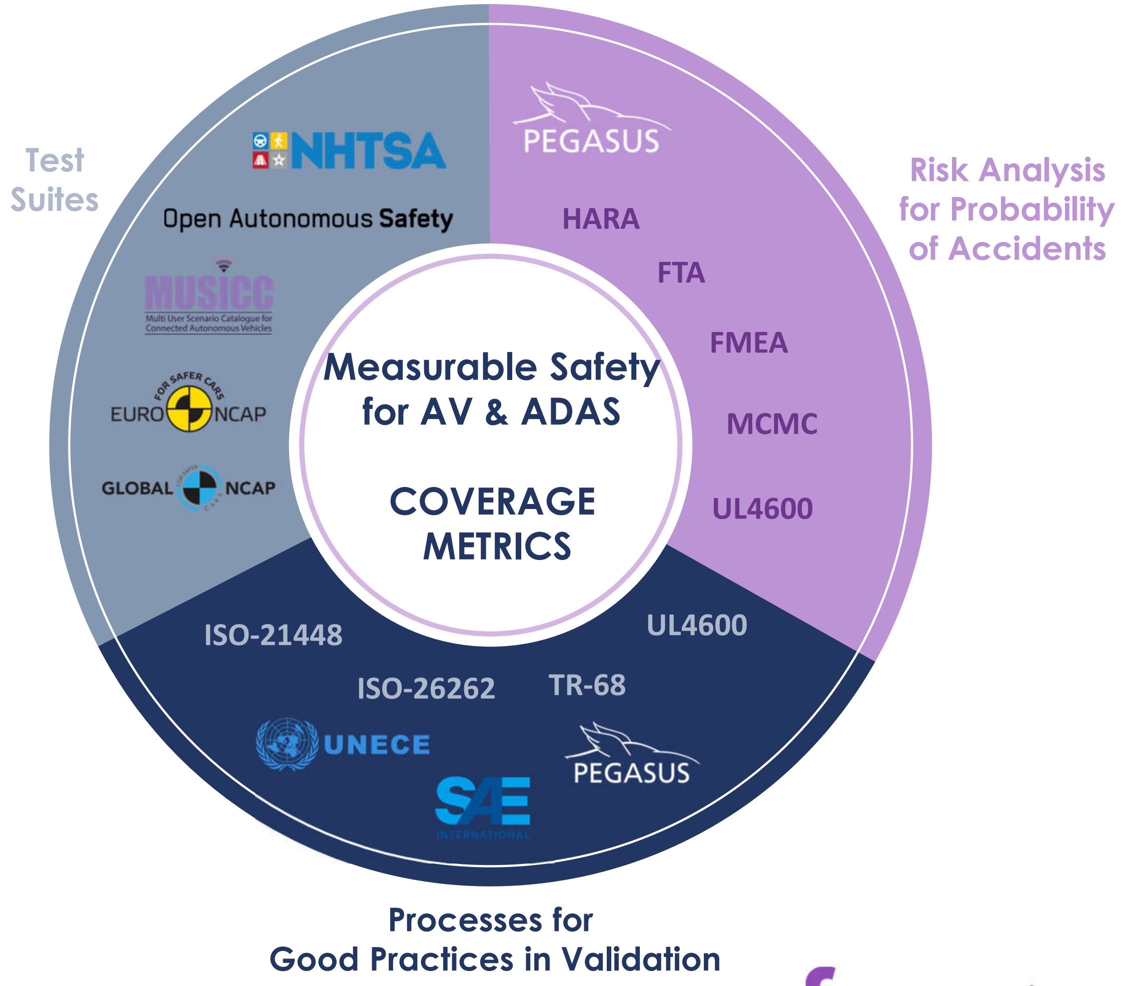
foretellix
verifying autonomous systems

Building the AV Safety Case

- Verification & validation coverage metrics are needed for enabling the body of evidence required for building the AV's safety case

- Coverage Metrics measure what actually happens and provides scenario coverage aggregation analytics & metrics

- Coverage metrics supports all existing and emerging safety standards & processes



Test Suites

NHTSA

Open Autonomous Safety

MUSICC

Multi User Scenario Catalogue for Connected Autonomous Vehicles

EURO NCAP

GLOBAL NCAP

Risk Analysis for Probability of Accidents

PEGASUS

HARA

FTA

FMEA

MCMC

UL4600

Measurable Safety for AV & ADAS

COVERAGE METRICS

ISO-21448

ISO-26262

TR-68

UL4600

UNECE

PEGASUS

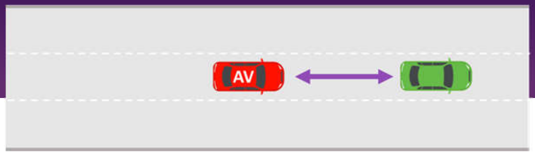
SAE INTERNATIONAL

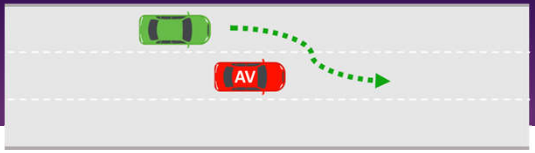
Processes for Good Practices in Validation

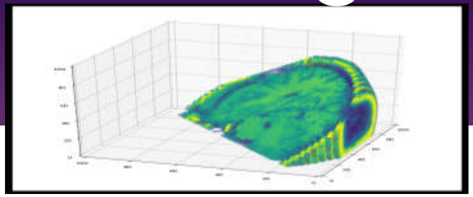
Securing America's Future Energy

foretellix

KPI/Metrics vs Coverage







- How did the AV perform within a given ODD?
- KPI/Metrics specify the specific measurements to be analyzed, given specific test conditions /ODD. Usually – “simulation output”
- **Answering:**
  - In ODD X, How did the ego perform for all test variations in the context of “cut in” ? (aggregate of all specific measurement)
  - What was TTC, when the AV was driving at 55kph, and the other player deceleration was -3 m/s^2 ? Is it above my threshold ?

- What was actually tested, out of the possible space of testing values [per ODD]
- Coverage can be measured both on test input/settings ,as well on output/results of the tests. It can be measure on one ,two, or multiple dimensions
- **Answering:**
  - For “cut in” scenario, on a road with 2 lanes and only green cars, what % of the possible AV speeds between 50KPH and 100KPH did I test ?
  - What % of the TTC space between 0 and 3S was demonstrated during all tests ?

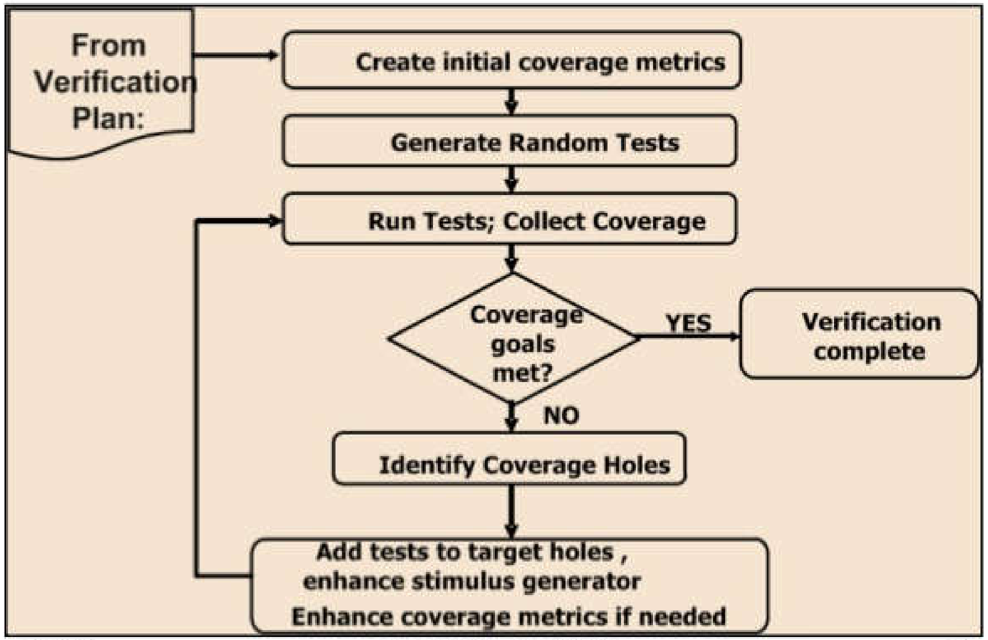
Securing America's Future Energy

foretellix

Coverage Driven Verification

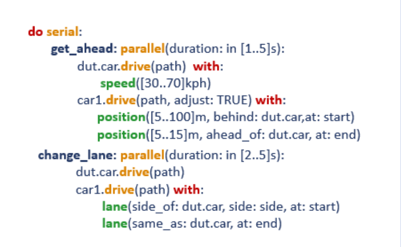
- The main method to verify complex VLSI/SOC designs: Microprocessors, GPUs, Network and cellular processors
- Method evolved in the early 90's
  - Intel's Pentium® floating point bug – ~$0.5B cost (1994)
- Main principles: Loop: Plan, test, measure and analyze metrics
- Goal is to maximize coverage
- Using Constrained Random Scenario/Test generation

Coverage Driven Verification


From Verification Plan:
Create initial coverage metrics
Generate Random Tests
Run Tests; Collect Coverage
Coverage goals met?
YES
Verification complete
NO
Identify Coverage Holes
Add tests to target holes , enhance stimulus generator Enhance coverage metrics if needed

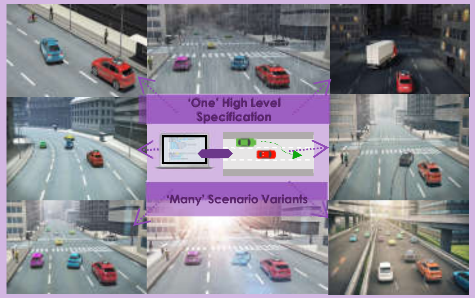
Coverage Driven Verification Methodology for Measurable Safety

Plan & Specify



```
do serial:
    get_ahead: parallel(duration: in [1..5]s):
        dut.car.drive(path)  with:
            speed([30..70]kph)
        car1.drive(path, adjust: TRUE) with:
            position([5..100]m, behind: dut.car,at: start)
            position([5..15]m, ahead_of: dut.car, at: end)
  change_lane: parallel(duration: in [2..5]s):
        dut.car.drive(path)
        car1.drive(path) with:
            lane(side_of: dut.car, side: side, at: start)
            lane(same_as: dut.car, at: end)
```

Generate



An **Iterative** flow that **adapts, refines and improves** the coverage plan using **methodology** to reach the desired confidence level

Execute & Monitor

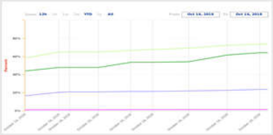
Simulation

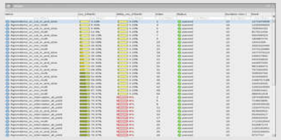
X-in-the-Loop

Test Tracks

Test Driving

Analyze

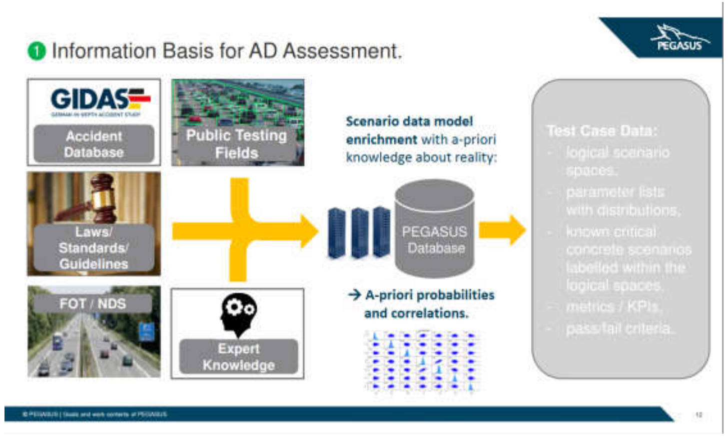




```
2.2.6.1.1.1 cut_in_and_slow                                   0%
2.2.6.1.1.2 cut_in_and_slow_no_stop_collision                 0%
2.2.6.1.1.3 side                                              100%
2.2.6.1.1.4 ego_speed                                         100%
2.2.6.1.1.5 other_speed_diff                                  [illegible]
2.2.6.1.1.6 other_distance                                    100%
2.2.6.1.1.7 cut_in_time                                       100%
2.2.6.1.1.8 other_speed_at_slow_down
2.2.6.1.1.9 slow_down_value                                   100%
2.2.6.1.1.10 slow_down_time                                   100%
2.2.6.1.1.11 ego_changed_lane                                 100%
2.2.6.1.1.12 cross__side__other_distance__other_speed_diff    25%
2.2.6.1.1.13 cross__slow_down_time__slow_down_value__ego_ch   [illegible]
2.2.6.1.1.14 ego_behavior_instance                            [illegible]
2.2.6.1.1.14.1 ego_lane_shifts                                100%
2.2.6.1.1.14.2 ego_collision                                  100%
2.2.6.1.1.14.3 ego_stopped                                    100%
```

CDV and PEGASUS method

- PEGAUS Method analyses extracted data and existing [historical] knowledge in order to create and define the required simulations space for AD assessment
- CDV complements and enhances the Pegasus approach:
    - CDV Adds the COVERAGE REQUIREMENTS as a data source for the decision process
    - Introduces constrained-random simulation generation to cover huge simulation and variation space
    - Provides methods to create unforeseeable scenarios

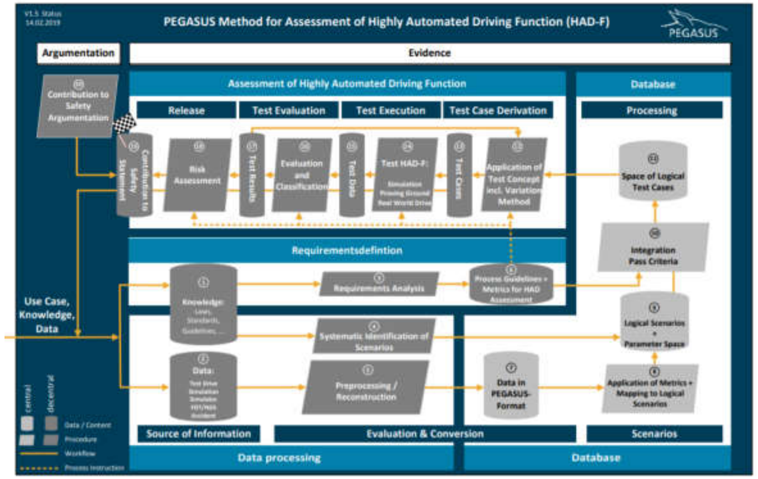

1 Information Basis for AD Assessment.

Scenario data model enrichment with a-priori knowledge about reality:

→ A-priori probabilities and correlations.

Test Case Data:
- logical scenario spaces,
- parameter lists with distributions,
- known critical concrete scenarios labelled within the logical spaces,
- metrics / KPIs,
- pass/fail criteria.

PEGASUS
GIDAS
GERMAN IN-DEPTH ACCIDENT STUDY
Accident Database
Public Testing Fields
Laws/ Standards/ Guidelines
FOT / NDS
Expert Knowledge
PEGASUS Database
© PEGASUS | Goals and work contents of PEGASUS
12

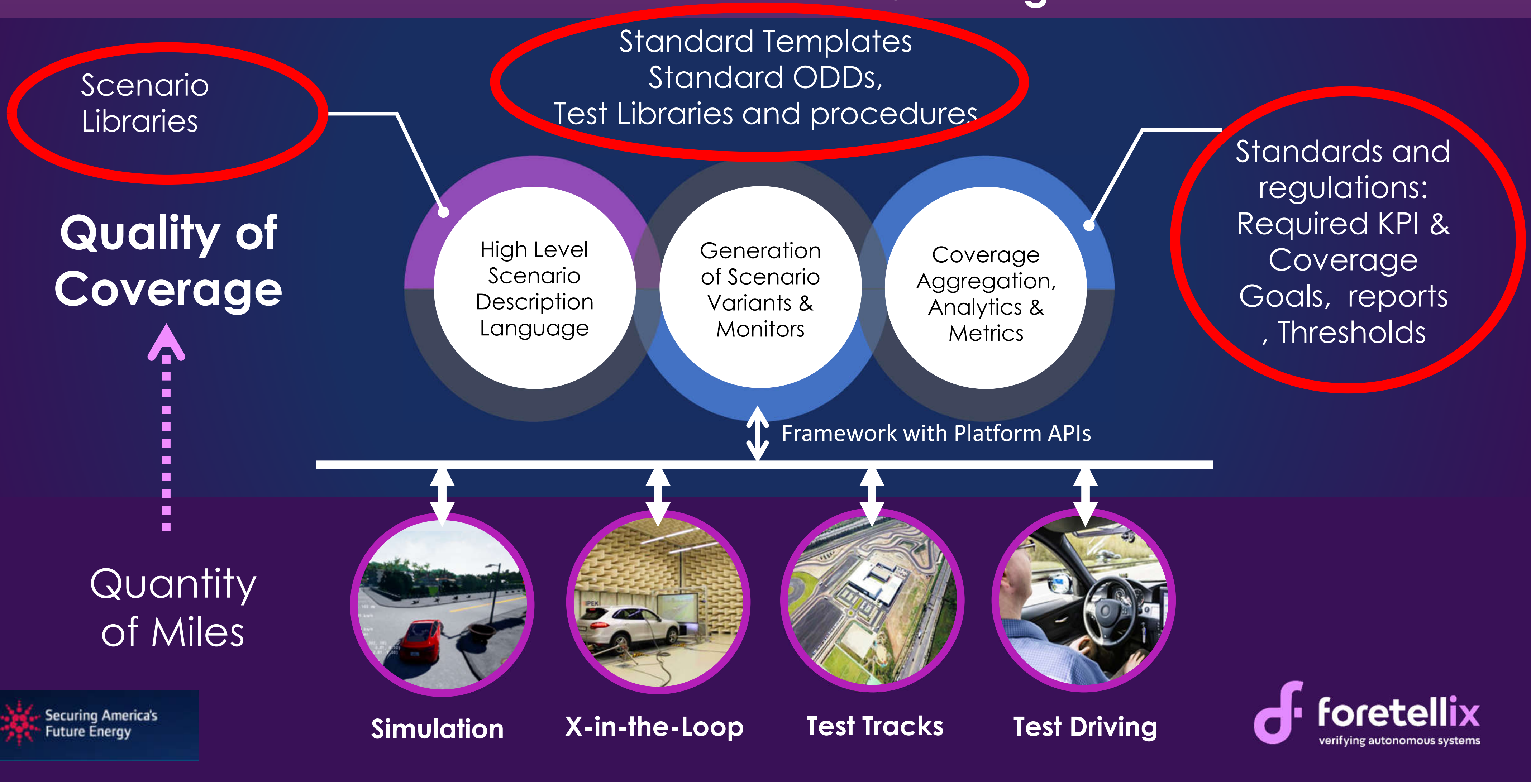

PEGASUS Method for Assessment of Highly Automated Driving Function (HAD-F)

V1.5 Status
14.02.2019
PEGASUS
Argumentation
Evidence
Contribution to Safety Argumentation
Assessment of Highly Automated Driving Function
Release
Test Evaluation
Test Execution
Test Case Derivation
Contribution to Safety Statement
Risk Assessment
Test Results
Evaluation and Classification
Test Data
Test HAD-F:
Simulation
Proving Ground
Real World Drive
Test Cases
Application of Test Concept incl. Variation Method
Database
Processing
Space of Logical Test Cases
Integration Pass Criteria
Requirementsdefintion
Requirements Analysis
Process Guidelines + Metrics for HAD Assessment
Use Case, Knowledge, Data
Knowledge: Laws, Standards, Guidelines, ...
Systematic Identification of Scenarios
Logical Scenarios + Parameter Space
Data:
Test Drive
Simulation
Simulator
FOT/NDS
Accident
Preprocessing / Reconstruction
Data in PEGASUS-Format
Application of Metrics + Mapping to Logical Scenarios
central
decentral
Data / Content
Procedure
Workflow
Process Instruction
Source of Information
Evaluation & Conversion
Scenarios
Data processing
Database

Securing America's Future Energy

The Building Blocks:

Data Driven Measurable Safety
Coverage Driven Verification



Scenario Libraries

Standard Templates
Standard ODDs,
Test Libraries and procedures

Standards and regulations: Required KPI & Coverage Goals, reports , Thresholds

Quality of Coverage

High Level Scenario Description Language

Generation of Scenario Variants & Monitors

Coverage Aggregation, Analytics & Metrics

Framework with Platform APIs

Quantity of Miles

Simulation

X-in-the-Loop

Test Tracks

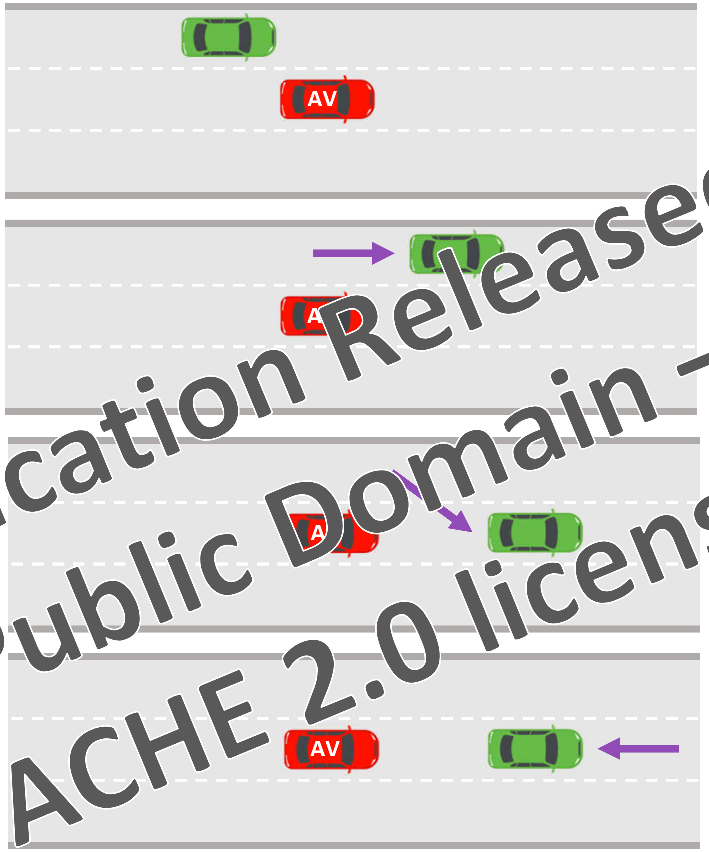
Test Driving

Securing America's Future Energy

foretellix
verifying autonomous systems

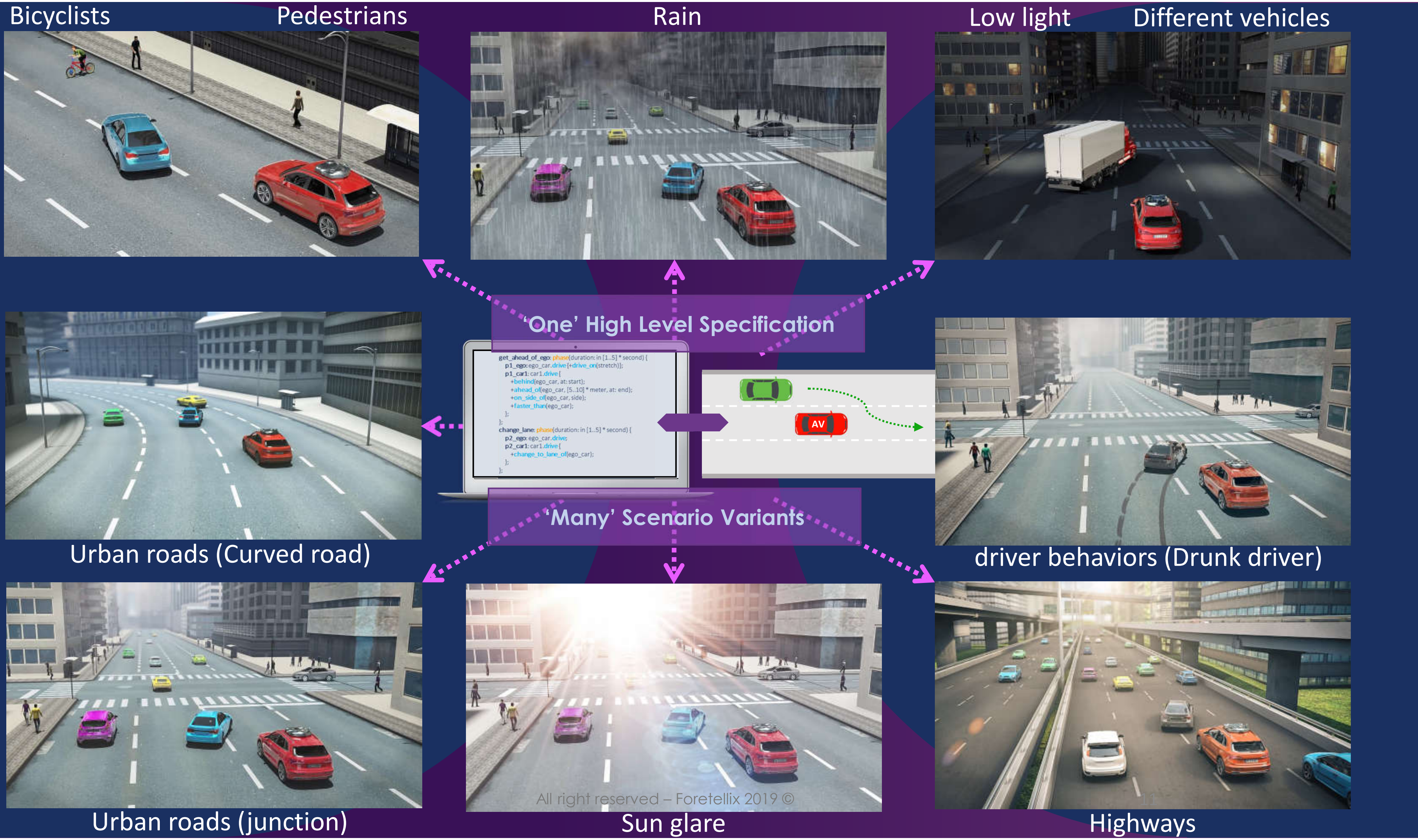
M-SDL – Cut in & Slow Scenario Example

```
scenario dut.cut_in_and_slow:

    car1: car                         # The other car
    side: av_left_right               # A side: left or right
    path:  path                       # A path in the map
   path_min_lanes(path, 2)          # Path should have at least two lanes

   do serial:
      get_ahead: parallel(duration: in [1..5]s):
          dut.car.drive(path)  with:
             speed([30..70]kph)
          car1.drive(path, adjust: TRUE) with:
             position([5..100]m, behind: dut_car,at: start)
             position([5..15]m, ahead_of: dut_car, at: end)

      change_lane: parallel(duration: in [2..5]s):
         dut.car.drive(path)
         car1.drive(path) with:
             lane(side_of: dut_car, side: side, at: start)
             lane(same_as: dut_car, at: end)

     slow: parallel(duration: in [1..5]s):
         dut.car.drive(path)
         car1.drive(path) with:
             speed_change(-[10..15]kph)
```



Example Parameters:

- From which side
- Speed of EGO
- Relative speed of green car
- Cut in aggressiveness
- Deceleration rate
- Number of lanes
- Road topology
- …

 Bicyclists Pedestrians

Rain

Low light Different vehicles

‘One’ High Level Specification

```
get_ahead_of_ego: phase(duration: in [1..5] * second) {
  p1_ego: ego_car.drive{+drive_on(stretch)};
  p1_car1: car1.drive{
    +behind(ego_car, at: start);
    +ahead_of(ego_car, [5..10] * meter, at: end);
    +on_side_of(ego_car, side);
    +faster_than(ego_car);
  };
};
change_lane: phase(duration: in [1..5] * second) {
  p2_ego: ego_car.drive;
  p2_car1: car1.drive{
    +change_to_lane_of(ego_car);
  };
};
```

‘Many’ Scenario Variants

Urban roads (Curved road)

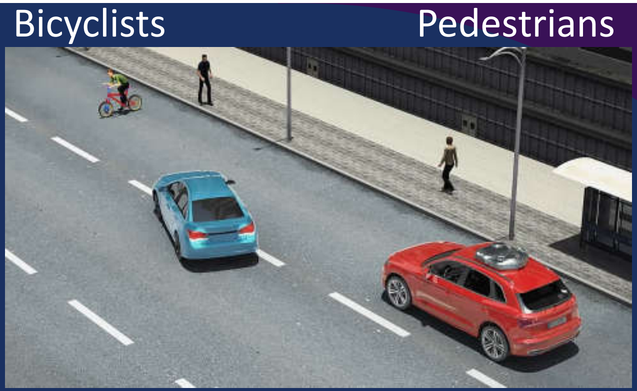
driver behaviors (Drunk driver)

Urban roads (junction)

Sun glare

Highways

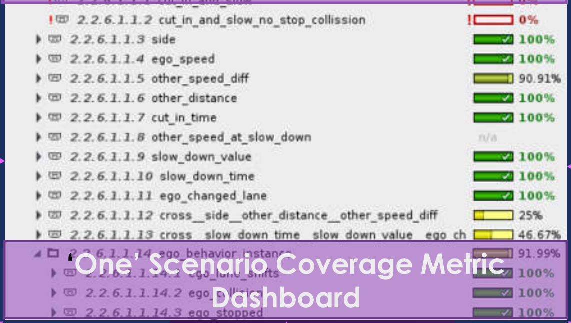
Bicyclists Pedestrians



Rain

Low light Different vehicles

‘Many’ Scenario Coverage Monitors Across Many Tests & Platforms



```
!  2.2.6.1.1.1 cut_in_and_slow                                   ! 0%
!  2.2.6.1.1.2 cut_in_and_slow_no_stop_collission                ! 0%
▸  2.2.6.1.1.3 side                                                100%
▸  2.2.6.1.1.4 ego_speed                                           100%
▸  2.2.6.1.1.5 other_speed_diff                                    90.91%
▸  2.2.6.1.1.6 other_distance                                      100%
▸  2.2.6.1.1.7 cut_in_time                                         100%
▸  2.2.6.1.1.8 other_speed_at_slow_down                            n/a
▸  2.2.6.1.1.9 slow_down_value                                     100%
▸  2.2.6.1.1.10 slow_down_time                                     100%
▸  2.2.6.1.1.11 ego_changed_lane                                   100%
▸  2.2.6.1.1.12 cross__side__other_distance__other_speed_diff      25%
▸  2.2.6.1.1.13 cross__slow_down_time__slow_down_value__ego_ch     46.67%
◢  2.2.6.1.1.14 ego_behavior_instance                              91.99%
   ▸  2.2.6.1.1.14.1 ego_lane_shifts                               100%
   ▸  2.2.6.1.1.14.2 ego_collision                                 100%
   ▸  2.2.6.1.1.14.3 ego_stopped                                   100%
```

‘One’ Scenario Coverage Metric Dashboard

Urban roads (Curved road)

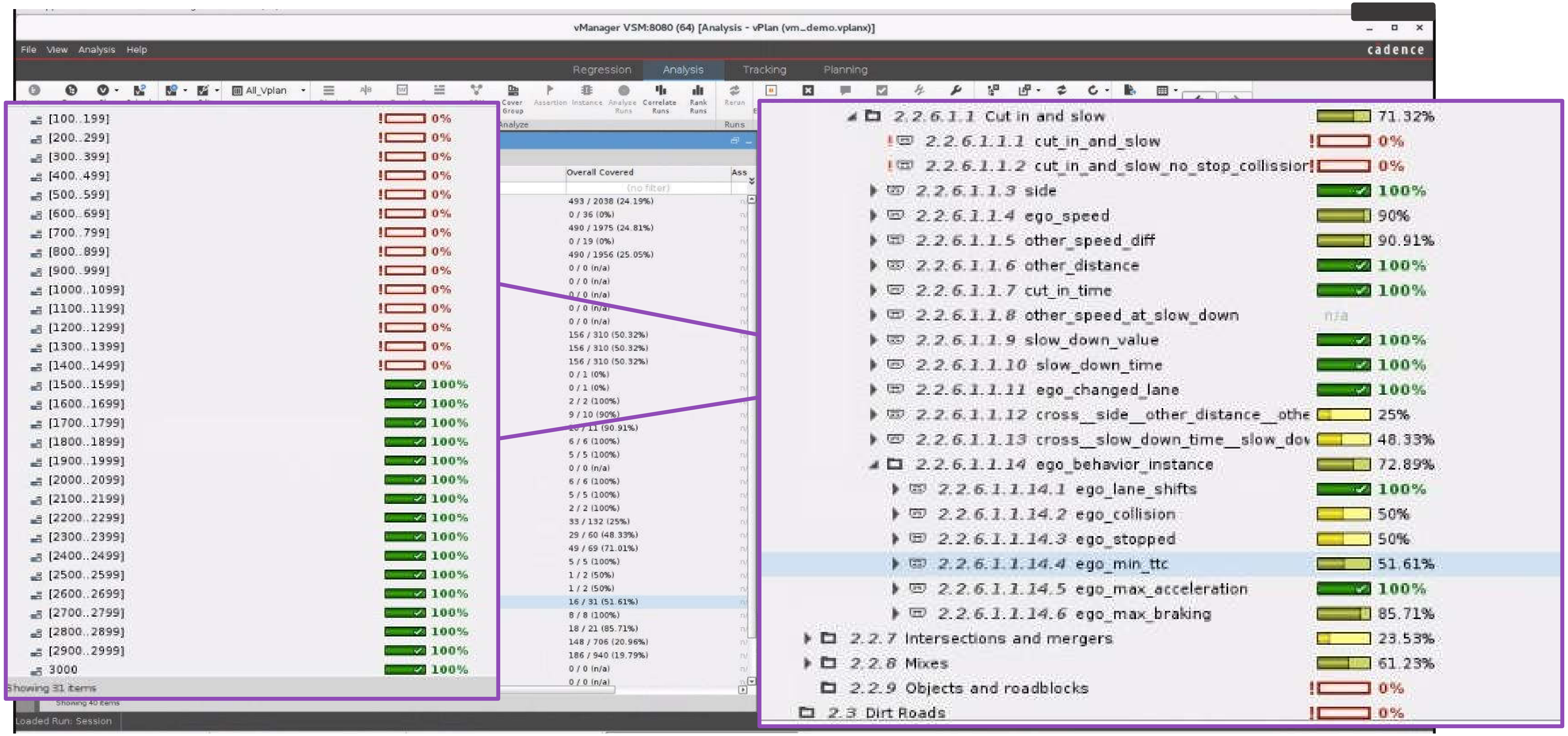
driver behaviors (Drunk driver)

Urban roads (junction)

Sun glare

Highways

Coverage analysis – TTC Coverage Gap



[100..199]	0%
[200..299]	0%
[300..399]	0%
[400..499]	0%
[500..599]	0%
[600..699]	0%
[700..799]	0%
[800..899]	0%
[900..999]	0%
[1000..1099]	0%
[1100..1199]	0%
[1200..1299]	0%
[1300..1399]	0%
[1400..1499]	0%
[1500..1599]	100%
[1600..1699]	100%
[1700..1799]	100%
[1800..1899]	100%
[1900..1999]	100%
[2000..2099]	100%
[2100..2199]	100%
[2200..2299]	100%
[2300..2399]	100%
[2400..2499]	100%
[2500..2599]	100%
[2600..2699]	100%
[2700..2799]	100%
[2800..2899]	100%
[2900..2999]	100%
3000	100%

Showing 31 items

Overall Covered	Ass
(no filter)	
493 / 2038 (24.19%)	n/
0 / 36 (0%)	n/
490 / 1975 (24.81%)	n/
0 / 19 (0%)	n/
490 / 1956 (25.05%)	n/
0 / 0 (n/a)	n/
0 / 0 (n/a)	n/
0 / 0 (n/a)	n/
0 / 0 (n/a)	n/
0 / 0 (n/a)	n/
156 / 310 (50.32%)	n/
156 / 310 (50.32%)	n/
156 / 310 (50.32%)	n/
0 / 1 (0%)	n/
0 / 1 (0%)	n/
2 / 2 (100%)	n/
9 / 10 (90%)	n/
10 / 11 (90.91%)	n/
6 / 6 (100%)	n/
5 / 5 (100%)	n/
0 / 0 (n/a)	n/
6 / 6 (100%)	n/
5 / 5 (100%)	n/
2 / 2 (100%)	n/
33 / 132 (25%)	n/
29 / 60 (48.33%)	n/
49 / 69 (71.01%)	n/
5 / 5 (100%)	n/
1 / 2 (50%)	n/
1 / 2 (50%)	n/
16 / 31 (51.61%)	n/
8 / 8 (100%)	n/
18 / 21 (85.71%)	n/
148 / 706 (20.96%)	n/
186 / 940 (19.79%)	n/
0 / 0 (n/a)	n/
0 / 0 (n/a)	n/

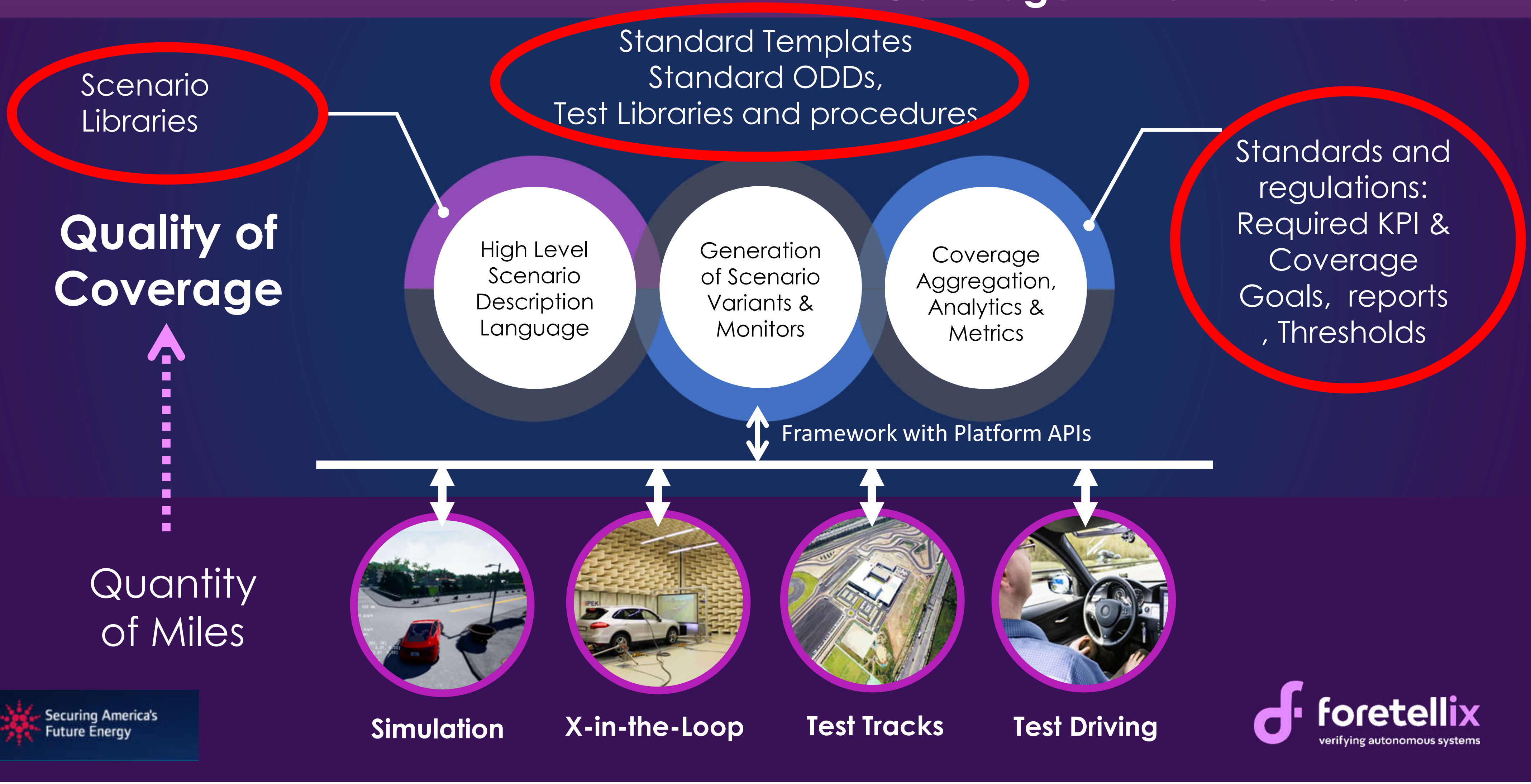
Showing 40 items

Loaded Run: Session

2.2.6.1.1 Cut in and slow	71.32%
2.2.6.1.1.1 cut_in_and_slow	0%
2.2.6.1.1.2 cut_in_and_slow_no_stop_collisior	0%
2.2.6.1.1.3 side	100%
2.2.6.1.1.4 ego_speed	90%
2.2.6.1.1.5 other_speed_diff	90.91%
2.2.6.1.1.6 other_distance	100%
2.2.6.1.1.7 cut_in_time	100%
2.2.6.1.1.8 other_speed_at_slow_down	n/a
2.2.6.1.1.9 slow_down_value	100%
2.2.6.1.1.10 slow_down_time	100%
2.2.6.1.1.11 ego_changed_lane	100%
2.2.6.1.1.12 cross__side__other_distance__othe	25%
2.2.6.1.1.13 cross__slow_down_time__slow_dov	48.33%
2.2.6.1.1.14 ego_behavior_instance	72.89%
2.2.6.1.1.14.1 ego_lane_shifts	100%
2.2.6.1.1.14.2 ego_collision	50%
2.2.6.1.1.14.3 ego_stopped	50%
2.2.6.1.1.14.4 ego_min_ttc	51.61%
2.2.6.1.1.14.5 ego_max_acceleration	100%
2.2.6.1.1.14.6 ego_max_braking	85.71%
2.2.7 Intersections and mergers	23.53%
2.2.8 Mixes	61.23%
2.2.9 Objects and roadblocks	0%
2.3 Dirt Roads	0%

Securing America's Future Energy

The Building Blocks: ✓ Data Driven Measurable Safety **Coverage Driven Verification**



Scenario Libraries

Standard Templates
Standard ODDs,
Test Libraries and procedures

Standards and regulations: Required KPI & Coverage Goals, reports , Thresholds

Quality of Coverage

High Level Scenario Description Language

Generation of Scenario Variants & Monitors

Coverage Aggregation, Analytics & Metrics

Framework with Platform APIs

Quantity of Miles

Simulation

X-in-the-Loop

Test Tracks

Test Driving

Securing America's Future Energy

foretellix
verifying autonomous systems

Summary: Measurable Safety – Coverage Metrics

- **Usage of Coverage Metrics Supplies:**
  - **Goals for testing**
  - **Threshold of quality and safe behaviors**
  - **Relative comparison between AVs**

- With Coverage Driven Verification AND Using standard templates, standard testing libraries and ODD – you have a complete certification system

Securing America's Future Energy

foretellix
verifying autonomous systems

Backup Slides

- A fully detailed presentation, with examples and SOTIF articulation.

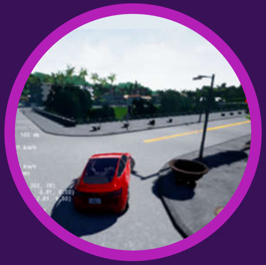
Outline

- ADS/AV Safety Primer
- Coverage Driven Verification Primer
- If time permits: Measurable Safety - CDV for SOTIF

Today:

Safe?

Quantity of Miles



Simulation

X-in-the-Loop

Test Tracks

Test Driving

Securing America's Future Energy

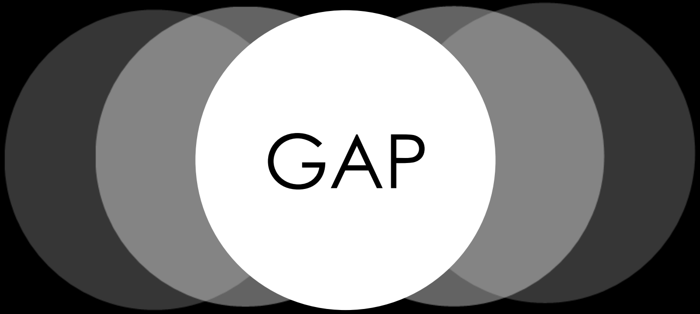
foretellix
verifying autonomous systems

The regulatory and liability prospective:

Safe?

No Standards In Place

No Rating System In Place



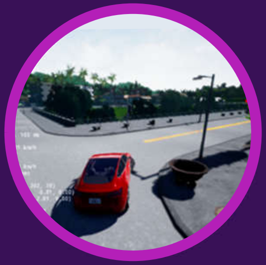
What to demand for certification?

What can be tested ?

What data can be used ?

What is “safe enough” ?

What is the required minimum ?



Simulation

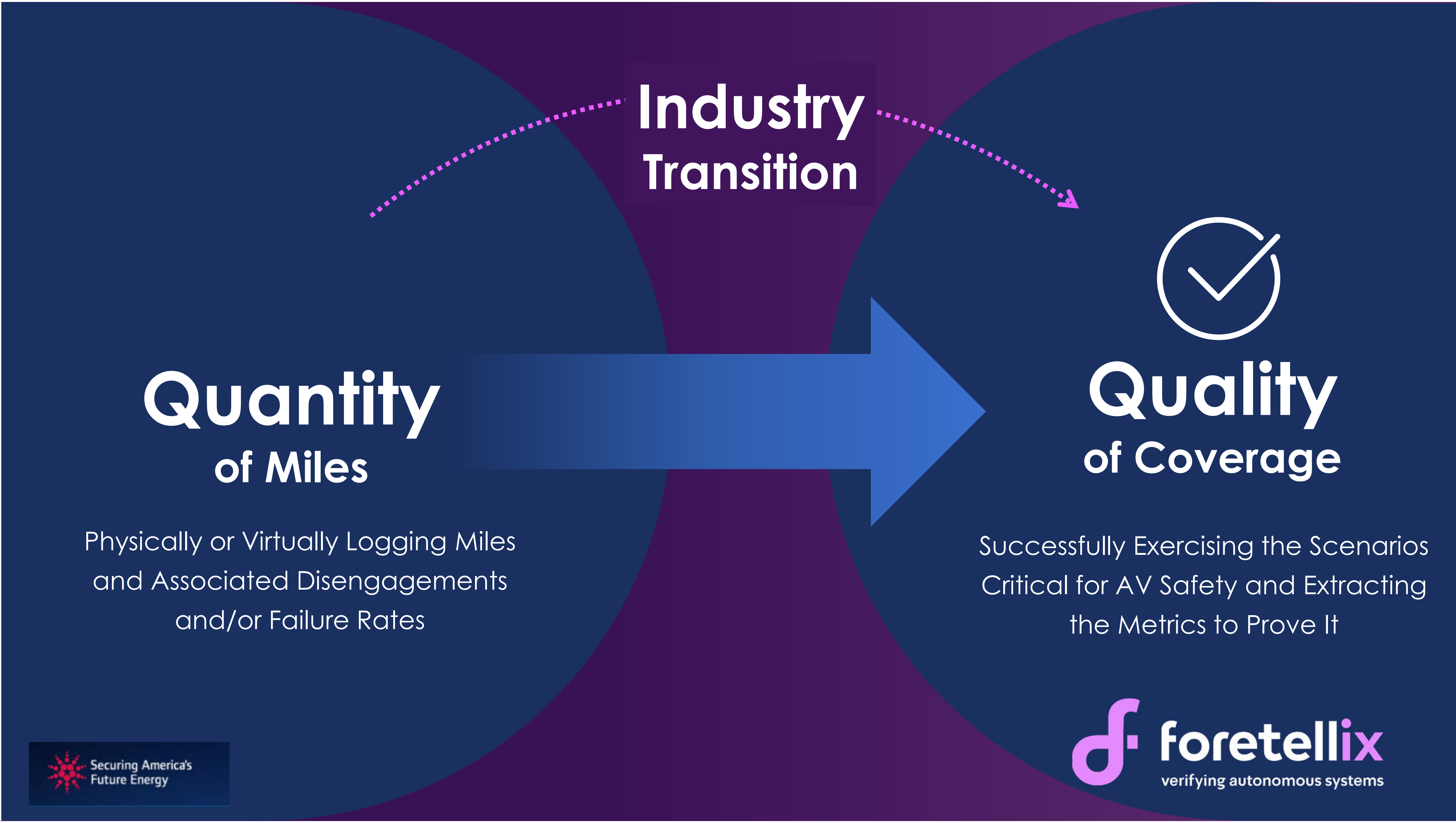
X-in-the-Loop

Test Tracks

Test Driving

Securing America's Future Energy

foretellix
verifying autonomous systems



Industry Transition

Quantity

of Miles

Physically or Virtually Logging Miles and Associated Disengagements and/or Failure Rates

Quality

of Coverage

Successfully Exercising the Scenarios Critical for AV Safety and Extracting the Metrics to Prove It

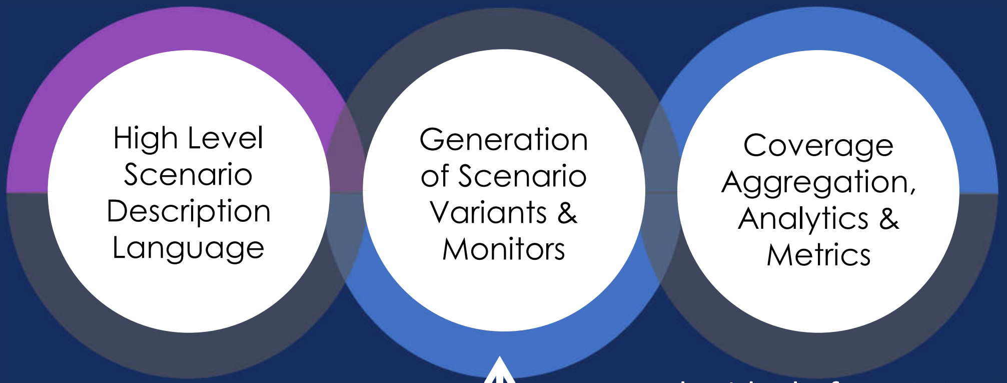
Securing America's Future Energy

foretellix
verifying autonomous systems

The transition:

Data Driven Measurable Safety

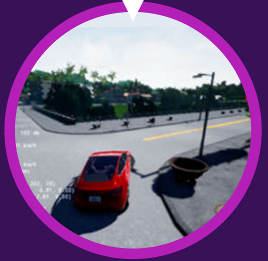
Coverage Driven Verification


High Level Scenario Description Language
Generation of Scenario Variants & Monitors
Coverage Aggregation, Analytics & Metrics

Framework with Platform APIs

Quality of Coverage

Quantity of Miles



Simulation

X-in-the-Loop

Test Tracks

Test Driving

Securing America's Future Energy

foretellix
verifying autonomous systems

A safety case is a structured argument, supported by a **body of evidence** that provides a compelling, comprehensible and valid case that a system is safe for a given application in a given operating environment

(UK Ministry of Defense 2017, page 26)

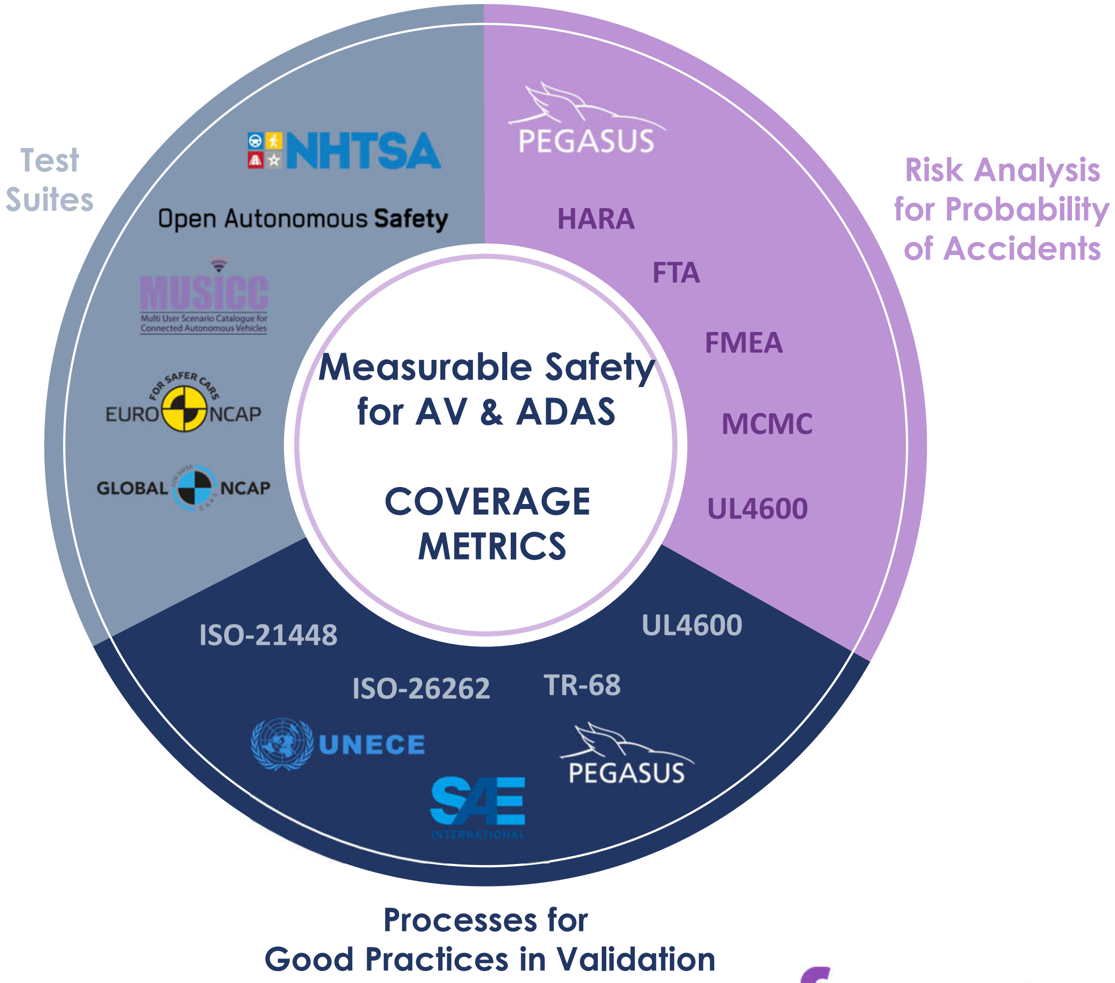
Therefore metrics should become a critical part (‘Core’) of a safety case

Securing America's Future Energy

foretellix
verifying autonomous systems

Building the AV Safety Case

- Verification & validation coverage metrics are needed for enabling the body of evidence required for building the AV's safety case
- Coverage Metrics measure what actually happens and provides scenario coverage aggregation analytics & metrics
- Coverage metrics supports all existing and emerging safety standards & processes



The transition:

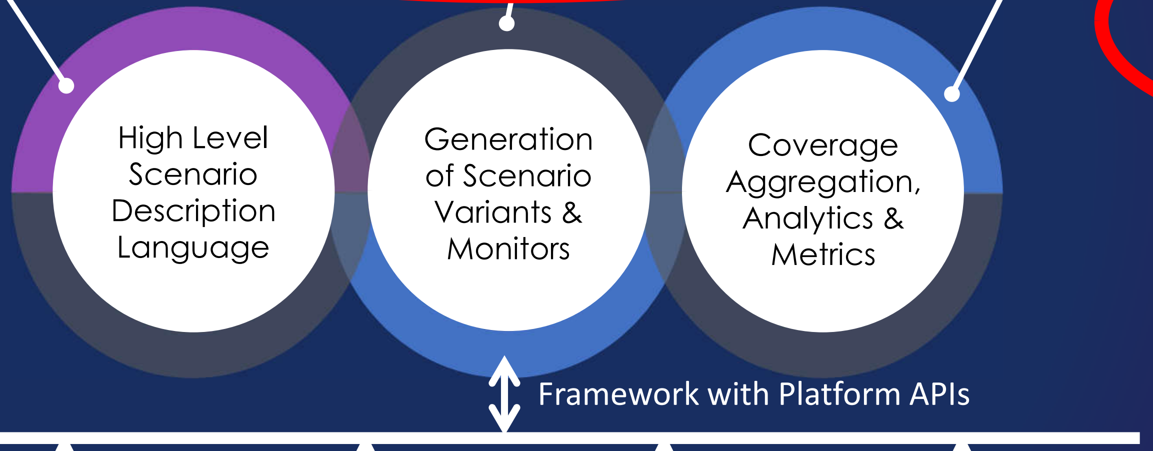
Data Driven Measurable Safety

Coverage Driven Verification

Standard Templates

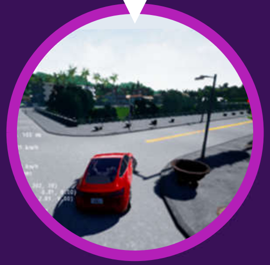
Standard

Quality of Coverage

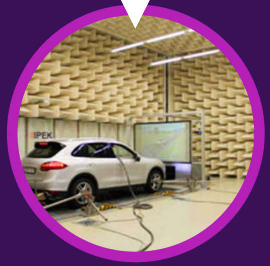

High Level Scenario Description Language
Generation of Scenario Variants & Monitors
Coverage Aggregation, Analytics & Metrics
Framework with Platform APIs

Standards and regulations

Quantity of Miles



Simulation



X-in-the-Loop

Test Tracks

Test Driving

Securing America's Future Energy

foretellix
verifying autonomous systems

The Building Blocks:

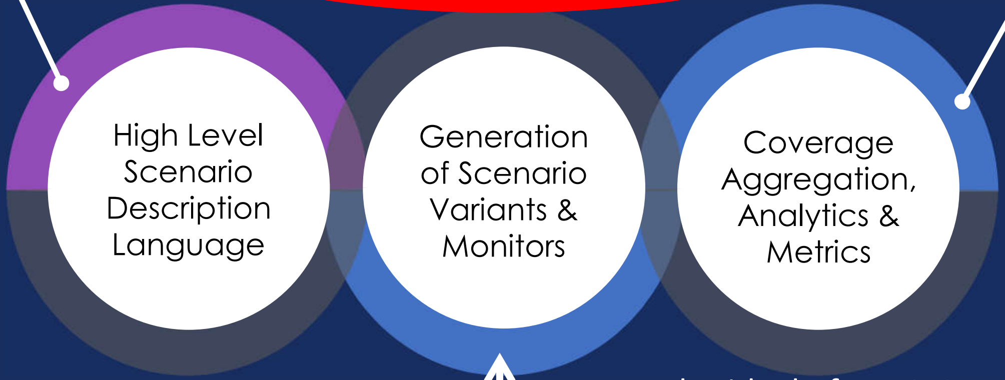
Data Driven Measurable Safety
Coverage Driven Verification

Scenario Libraries

Standard Templates
Standard ODDs,
Test Libraries and procedures

Standards and regulations:
Required KPI & Coverage Goals, reports , Thresholds

Quality of Coverage



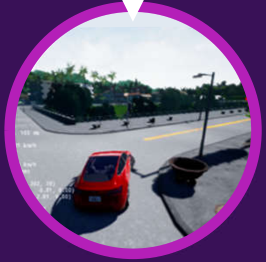
High Level Scenario Description Language

Generation of Scenario Variants & Monitors

Coverage Aggregation, Analytics & Metrics

Framework with Platform APIs

Quantity of Miles



Simulation

X-in-the-Loop

Test Tracks

Test Driving

Securing America's Future Energy

foretellix
verifying autonomous systems

Outline

- ADS/AV Safety Primer
- Coverage Driven Verification Primer
- If time permits: Measurable Safety - CDV for SOTIF

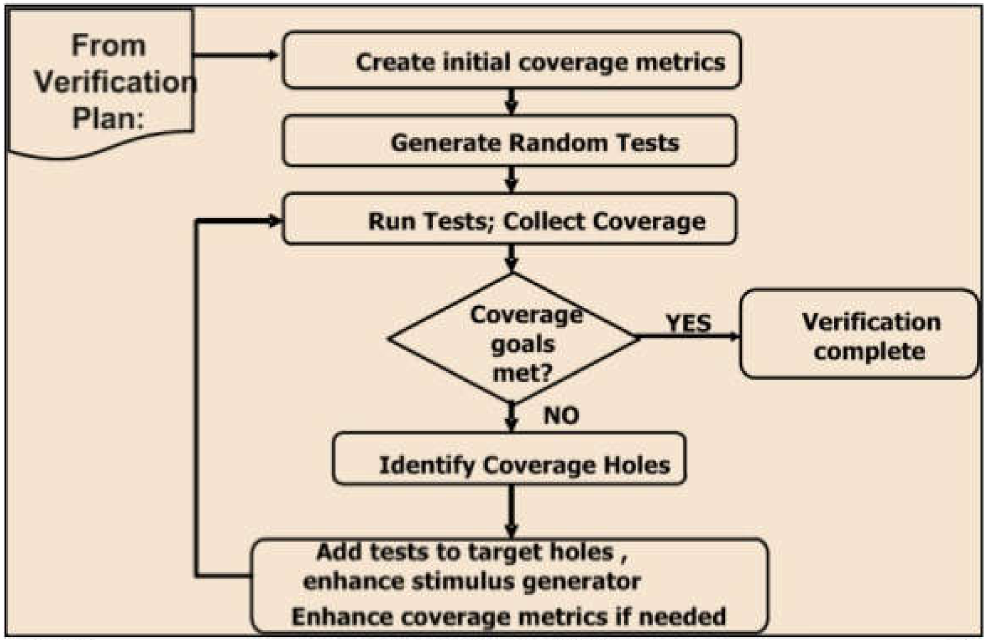
Securing America's Future Energy

foretellix

Coverage Driven Verification

- The main method to verify complex VLSI/SOC designs: Microprocessors, GPUs, Network and cellular processors
- Method evolved in the early 90’s
  - Intel’s Pentium® floating point bug – ~$0.5B cost (1994)
- Main principles: Loop: Plan, test, measure and analyze metrics
- Goal is to maximize coverage
- Using Constrained Random Scenario/Test generation

Coverage Driven Verification



From Verification Plan:

Create initial coverage metrics

Generate Random Tests

Run Tests; Collect Coverage

Coverage goals met?

YES

Verification complete

NO

Identify Coverage Holes

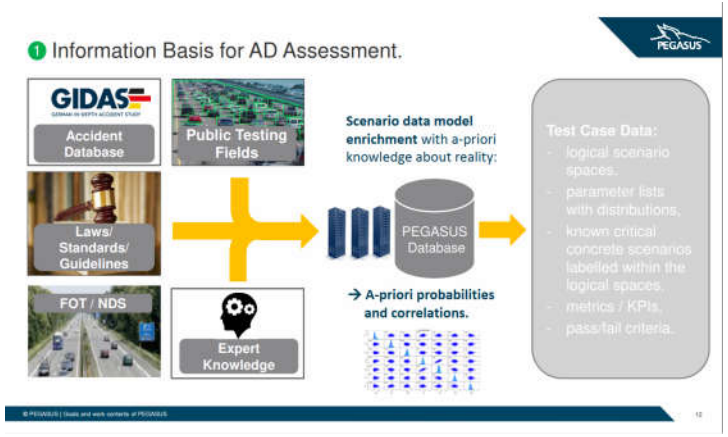
Add tests to target holes , enhance stimulus generator Enhance coverage metrics if needed

Securing America's Future Energy

foretellix

CDV and PEGASUS method

- PEGAUS Method analyses extracted data and existing [historical] knowledge in order to create and define the required simulations space for AD assessment
- CDV complements and enhances the Pegasus approach:
  - CDV Adds the COVERAGE REQUIREMENTS as a data source for the decision process
  - Introduces constrained-random simulation generation to cover huge simulation and variation space
  - Provides methods to create unforeseeable scenarios

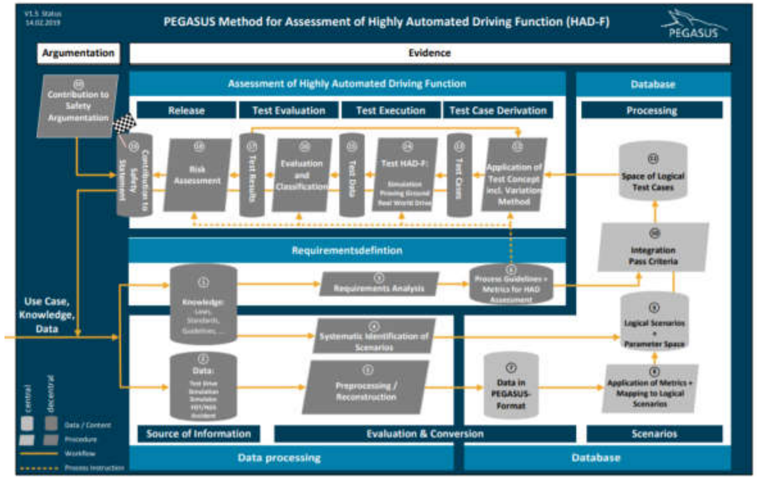


1 Information Basis for AD Assessment.

Scenario data model enrichment with a-priori knowledge about reality:

→ A-priori probabilities and correlations.

Test Case Data:
- logical scenario spaces,
- parameter lists with distributions,
- known critical concrete scenarios labelled within the logical spaces,
- metrics / KPIs,
- pass/fail criteria.

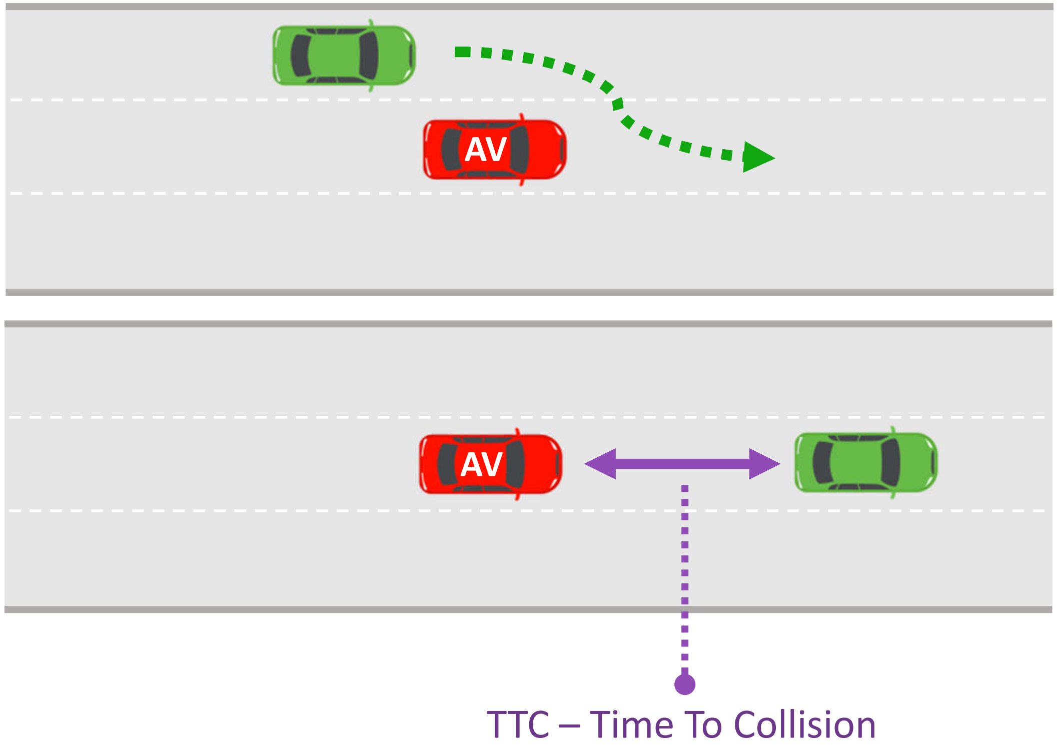


PEGASUS Method for Assessment of Highly Automated Driving Function (HAD-F)

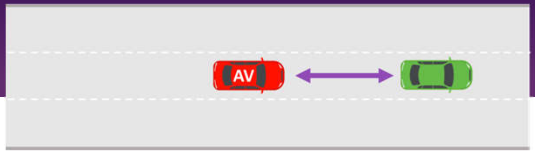
Securing America's Future Energy

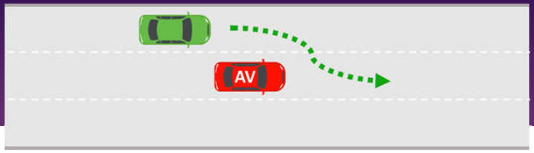
Test Scenarios and coverage explained

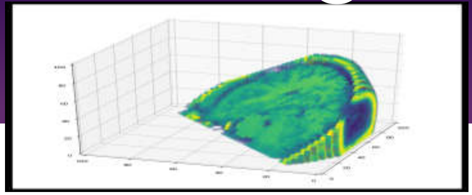
- Parameters and measurements:
  - From which side
  - Speed of EGO
  - Relative speed of green car
  - Min Time To Collision
  - Deceleration rate
  - Number of lanes
  - Road topology
  - ………….



KPI/Metrics vs Coverage







- How did the AV perform within a given ODD?
- KPI/Metrics specify the specific measurements to be analyzed, given specific test conditions /ODD. Usually – “simulation output”
- **Answering:**
  - In ODD X, How did the ego perform for all test variations in the context of “cut in” ? (aggregate of all specific measurement)
  - What was TTC, when the AV was driving at 55kph, and the other player deceleration was -3 m/s^2 ? Is it above my threshold ?

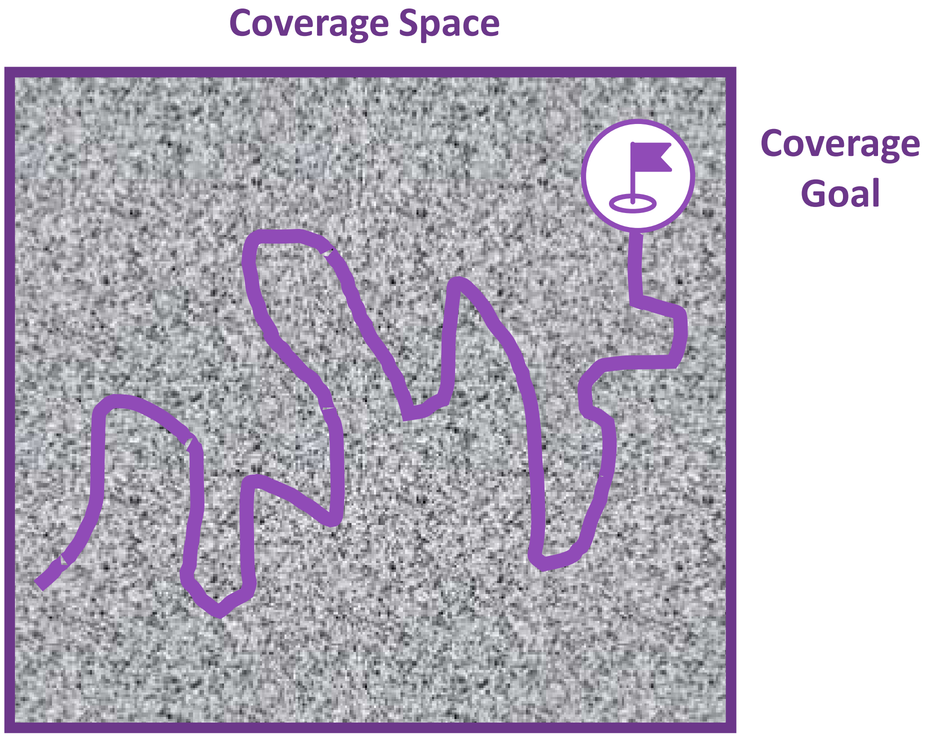
- What was actually tested, out of the possible space of testing values [per ODD]
- Coverage can be measured both on test input/settings ,as well on output/results of the tests. It can be measure on one ,two, or multiple dimensions
- **Answering:**
  - For “cut in” scenario, on a road with 2 lanes and only green cars, what % of the possible AV speeds between 50KPH and 100KPH did I test ?
  - What % of the TTC space between 0 and 3S was demonstrated during all tests ?

Securing America's Future Energy

foretellix

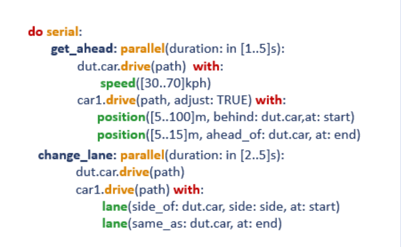
CDV
it’s the journey that matters

Exploring the coverage space,
While seeking to reach specific goals



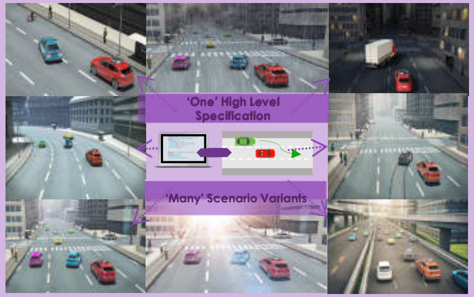
Coverage Driven Verification Methodology for Measurable Safety

Plan & Specify



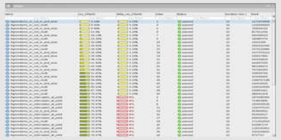
```
do serial:
    get_ahead: parallel(duration: in [1..5]s):
        dut.car.drive(path)  with:
            speed([30..70]kph)
        car1.drive(path, adjust: TRUE) with:
            position([5..100]m, behind: dut.car,at: start)
            position([5..15]m, ahead_of: dut.car, at: end)
  change_lane: parallel(duration: in [2..5]s):
        dut.car.drive(path)
        car1.drive(path) with:
            lane(side_of: dut.car, side: side, at: start)
            lane(same_as: dut.car, at: end)
```

Generate



An **Iterative** flow that **adapts, refines and improves** the coverage plan using **methodology** to reach the desired confidence level

Analyze



Execute & Monitor

Simulation

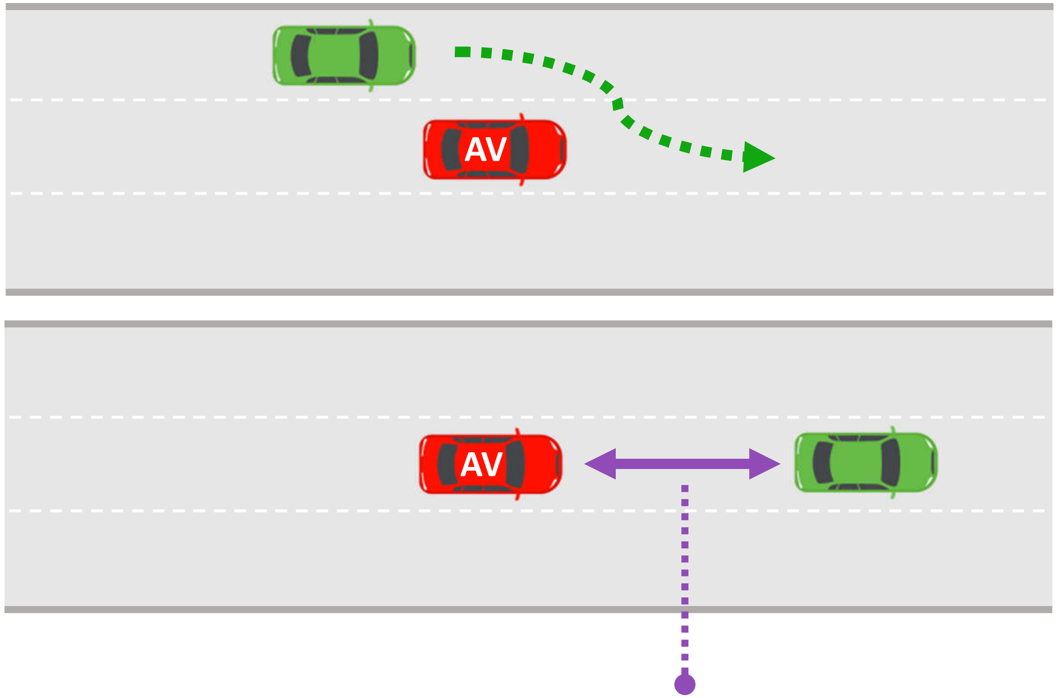
X-in-the-Loop

Test Tracks

Test Driving

Test Scenarios and coverage explained

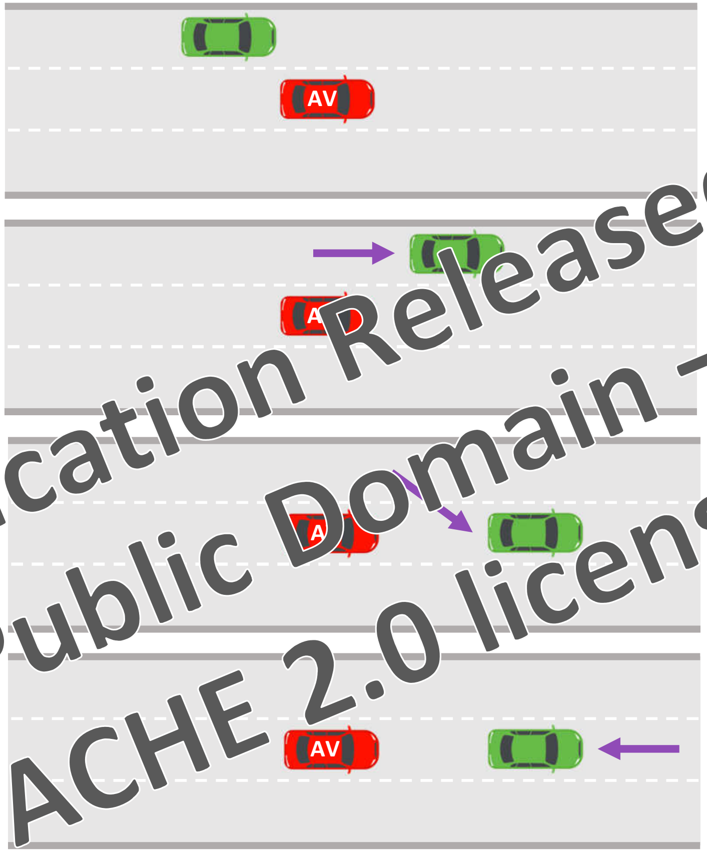
- Parameters and measurements:
  - From which side
  - Speed of EGO
  - Relative speed of green car
  - Cut in aggressiveness
  - Deceleration rate
  - Number of lanes
  - Road topology
  - ………….



TTC – Time To Collision

M-SDL – Cut in & Slow Scenario Example

```
scenario dut.cut_in_and_slow:

   car1: car                      # The other car
   side: av_left_right             # A side: left or right
   path:  path                    # A path in the map
  path_min_lanes(path, 2)       # Path should have at least two lanes

  do serial:
      get_ahead: parallel(duration: in [1..5]s):
           dut.car.drive(path)  with:
               speed([30..70]kph)
           car1.drive(path, adjust: TRUE) with:
               position([5..100]m, behind: dut_car,at: start)
               position([5..15]m, ahead_of: dut_car, at: end)

      change_lane: parallel(duration: in [2..5]s):
           dut.car.drive(path)
           car1.drive(path) with:
               lane(side_of: dut_car, side: side, at: start)
               lane(same_as: dut_car, at: end)

     slow: parallel(duration: in [1..5]s):
          dut.car.drive(path)
          car1.drive(path) with:
              speed_change(-[10..15]kph)
```

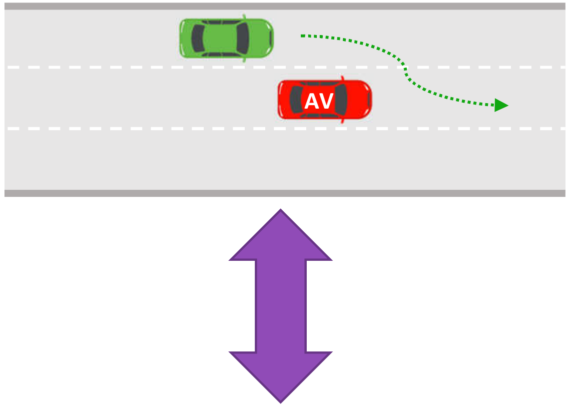


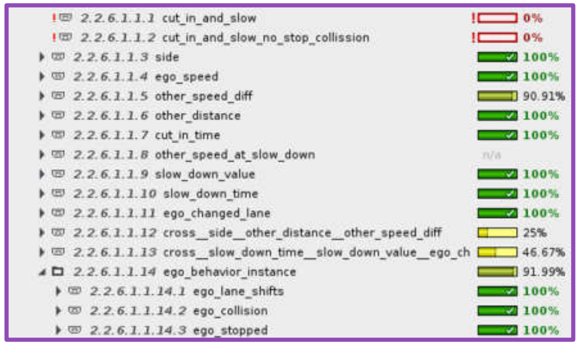
Example Parameters:

- From which side
- Speed of EGO
- Relative speed of green car
- Cut in aggressiveness
- Deceleration rate
- Number of lanes
- Road topology
- …

Specifying Scenario Coverage Metrics

```
# Original cut in and slow definition
do serial:
      …
# Coverage definitions
 cover(side)
 !ego_speed:= sample(ego_car.speed, @change_lane.start)
 cover(ego_speed, unit: kph, range: [10..100], every: 10)
 !other_speed_diff:= sample(car1.speed - ego_car.speed,
@change_lane.start)
cover(other_speed_diff, unit: kph, range: [1..20], every: 5)
!ego_min_ttc:= sample(min_ttc(),  @end)
 cover(ego_min_ttc, unit: ms, range: [0..3000], every=100)
```



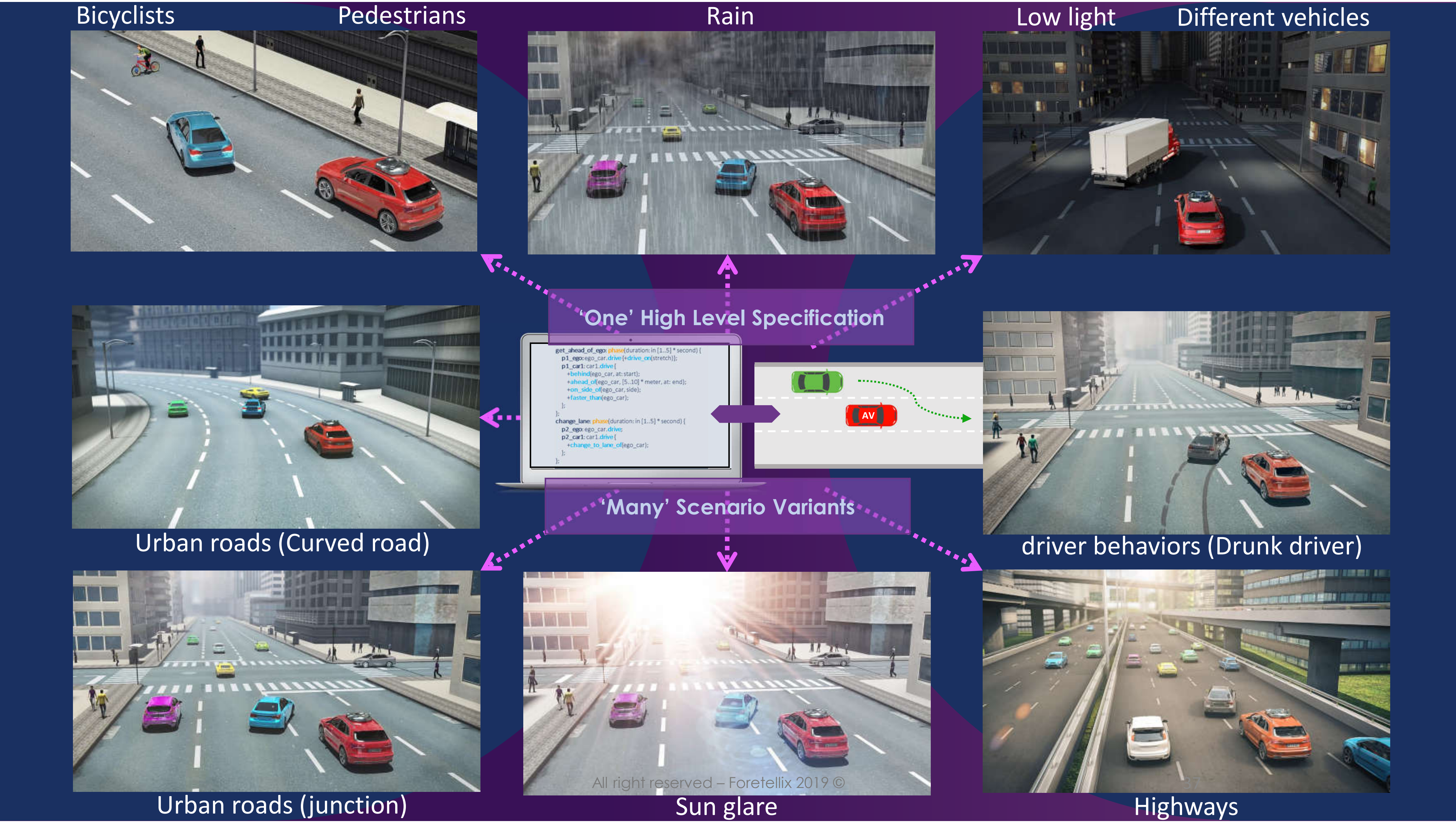


```
! 2.2.6.1.1.1 cut_in_and_slow                                        ! 0%
! 2.2.6.1.1.2 cut_in_and_slow_no_stop_collission                     ! 0%
▸ 2.2.6.1.1.3 side                                                   ✓ 100%
▸ 2.2.6.1.1.4 ego_speed                                              ✓ 100%
▸ 2.2.6.1.1.5 other_speed_diff                                         90.91%
▸ 2.2.6.1.1.6 other_distance                                         ✓ 100%
▸ 2.2.6.1.1.7 cut_in_time                                            ✓ 100%
▸ 2.2.6.1.1.8 other_speed_at_slow_down                                 n/a
▸ 2.2.6.1.1.9 slow_down_value                                        ✓ 100%
▸ 2.2.6.1.1.10 slow_down_time                                        ✓ 100%
▸ 2.2.6.1.1.11 ego_changed_lane                                      ✓ 100%
▸ 2.2.6.1.1.12 cross__side__other_distance__other_speed_diff           25%
▸ 2.2.6.1.1.13 cross__slow_down_time__slow_down_value__ego_ch          46.67%
◢ 2.2.6.1.1.14 ego_behavior_instance                                   91.99%
   ▸ 2.2.6.1.1.14.1 ego_lane_shifts                                  ✓ 100%
   ▸ 2.2.6.1.1.14.2 ego_collision                                    ✓ 100%
   ▸ 2.2.6.1.1.14.3 ego_stopped                                      ✓ 100%
```

‘Many’ Scenario Coverage Monitors Across Many Tests & Platforms
‘One’ Scenario Coverage Metric Dashboard

Securing America's Future Energy

foretellix



Bicyclists

Pedestrians

Rain

Low light

Different vehicles

'One' High Level Specification

```
get_ahead_of_ego: phase(duration: in [1..5] * second) {
  p1_ego: ego_car.drive {+drive_on(stretch)};
  p1_car1: car1.drive {
    +behind(ego_car, at: start);
    +ahead_of(ego_car, [5..10] * meter, at: end);
    +on_side_of(ego_car, side);
    +faster_than(ego_car);
  };
};
change_lane: phase(duration: in [1..5] * second) {
  p2_ego: ego_car.drive;
  p2_car1: car1.drive {
    +change_to_lane_of(ego_car);
  };
};
```

'Many' Scenario Variants

Urban roads (Curved road)

driver behaviors (Drunk driver)

Urban roads (junction)

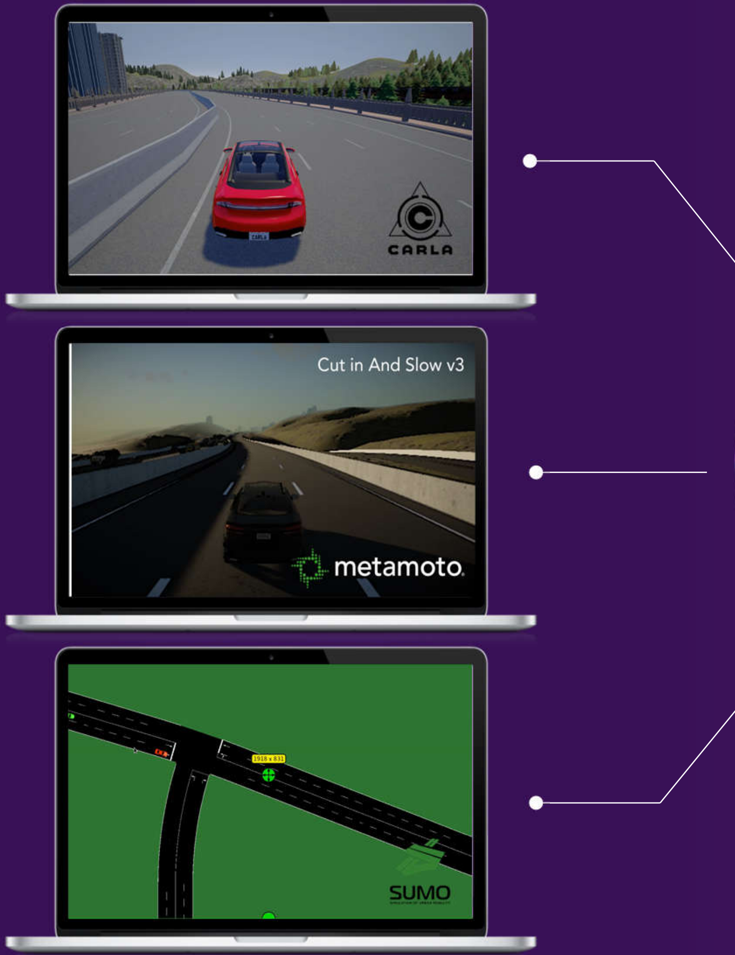
Sun glare

Highways

Portability Across Testing Platforms

- Generate scenarios and run them on multiple testing platforms
- Collect coverage from multiple simulators, X in a loop, Test tracks and street driving
- Correlate between real world and simulated coverage data

Example Simulators



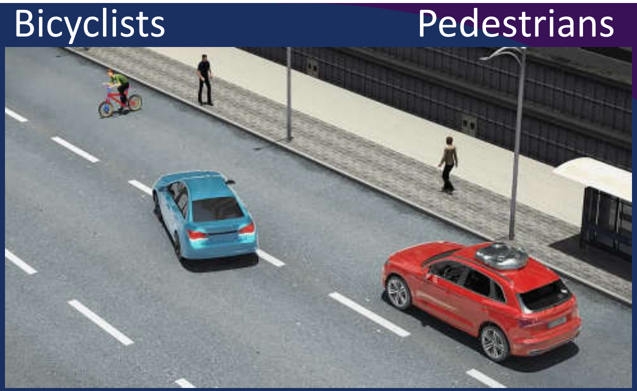
Simulation

X-in-the-Loop

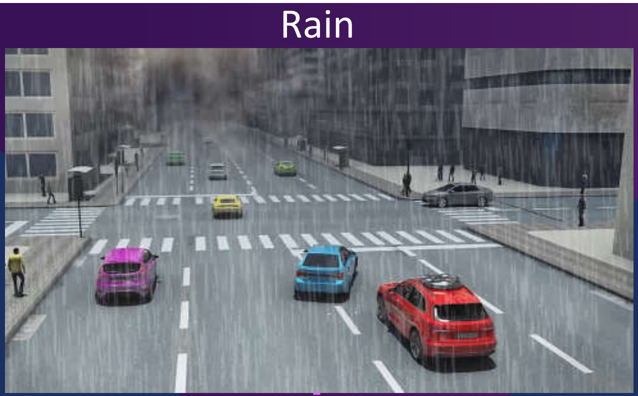
Test Tracks

Test Driving

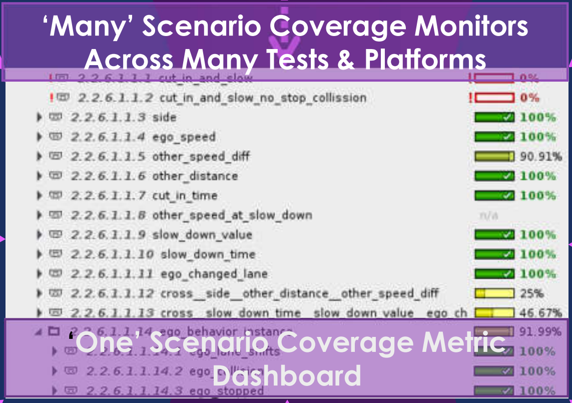
Bicyclists Pedestrians



Rain



Low light Different vehicles



‘Many’ Scenario Coverage Monitors Across Many Tests & Platforms

```
! 2.2.6.1.1.1 cut_in_and_slow                                  ! 0%
! 2.2.6.1.1.2 cut_in_and_slow_no_stop_collission               ! 0%
▸ 2.2.6.1.1.3 side                                             ✓ 100%
▸ 2.2.6.1.1.4 ego_speed                                        ✓ 100%
▸ 2.2.6.1.1.5 other_speed_diff                                   90.91%
▸ 2.2.6.1.1.6 other_distance                                   ✓ 100%
▸ 2.2.6.1.1.7 cut_in_time                                      ✓ 100%
▸ 2.2.6.1.1.8 other_speed_at_slow_down                           n/a
▸ 2.2.6.1.1.9 slow_down_value                                  ✓ 100%
▸ 2.2.6.1.1.10 slow_down_time                                  ✓ 100%
▸ 2.2.6.1.1.11 ego_changed_lane                                ✓ 100%
▸ 2.2.6.1.1.12 cross__side__other_distance__other_speed_diff     25%
▸ 2.2.6.1.1.13 cross__slow_down_time__slow_down_value__ego_ch    46.67%
◢ 2.2.6.1.1.14 ego_behavior_instance                             91.99%
    ▸ 2.2.6.1.1.14.1 ego_lane_shifts                           ✓ 100%
    ▸ 2.2.6.1.1.14.2 ego_collision                             ✓ 100%
    ▸ 2.2.6.1.1.14.3 ego_stopped                               ✓ 100%
```

‘One’ Scenario Coverage Metric Dashboard

Urban roads (Curved road)

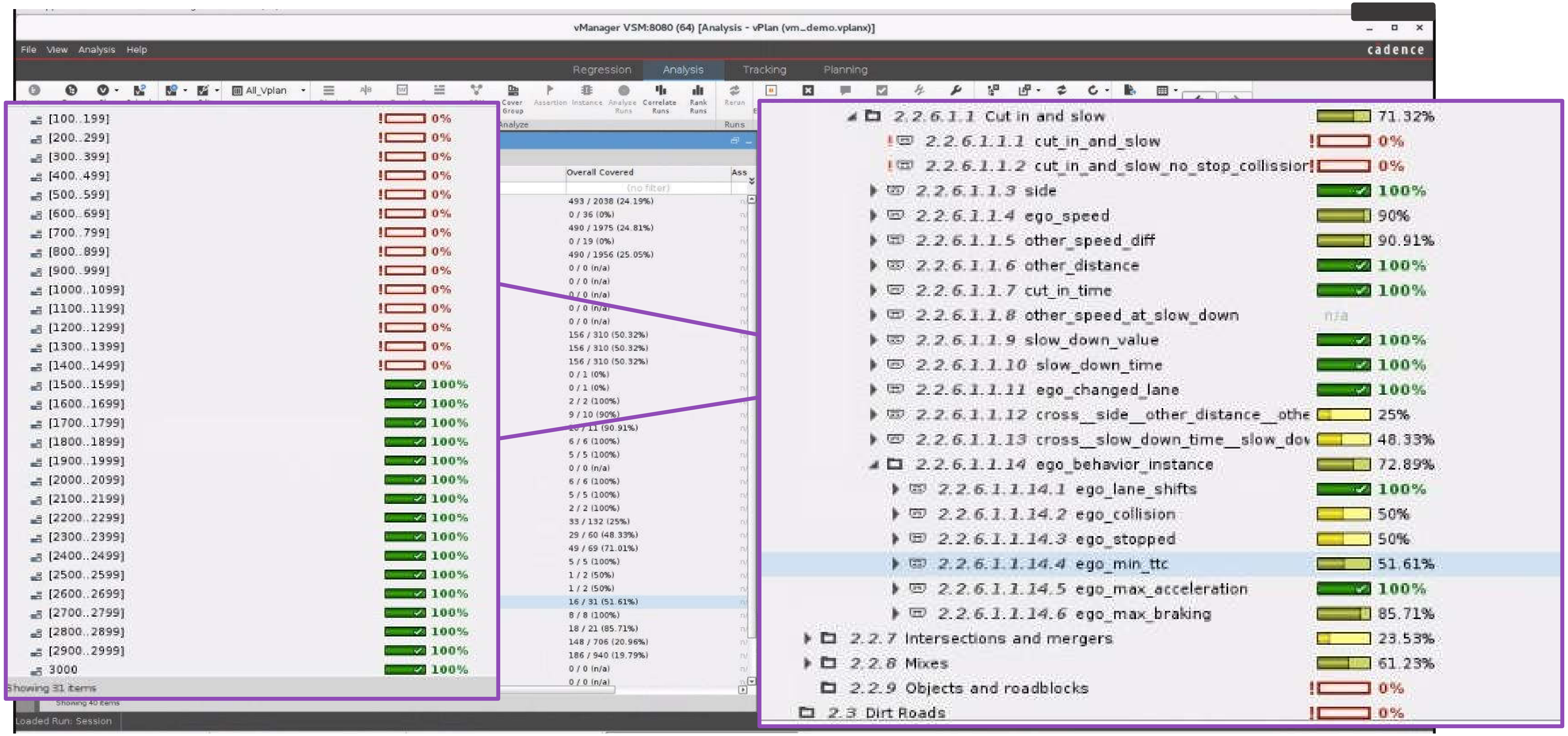
driver behaviors (Drunk driver)

Urban roads (junction)

Sun glare

Highways

Coverage analysis – TTC Coverage Gap



[100..199]	0%
[200..299]	0%
[300..399]	0%
[400..499]	0%
[500..599]	0%
[600..699]	0%
[700..799]	0%
[800..899]	0%
[900..999]	0%
[1000..1099]	0%
[1100..1199]	0%
[1200..1299]	0%
[1300..1399]	0%
[1400..1499]	0%
[1500..1599]	100%
[1600..1699]	100%
[1700..1799]	100%
[1800..1899]	100%
[1900..1999]	100%
[2000..2099]	100%
[2100..2199]	100%
[2200..2299]	100%
[2300..2399]	100%
[2400..2499]	100%
[2500..2599]	100%
[2600..2699]	100%
[2700..2799]	100%
[2800..2899]	100%
[2900..2999]	100%
3000	100%

Showing 31 items

Overall Covered	Ass
(no filter)	
493 / 2038 (24.19%)	n/
0 / 36 (0%)	n/
490 / 1975 (24.81%)	n/
0 / 19 (0%)	n/
490 / 1956 (25.05%)	n/
0 / 0 (n/a)	n/
0 / 0 (n/a)	n/
0 / 0 (n/a)	n/
0 / 0 (n/a)	n/
0 / 0 (n/a)	n/
156 / 310 (50.32%)	n/
156 / 310 (50.32%)	n/
156 / 310 (50.32%)	n/
0 / 1 (0%)	n/
0 / 1 (0%)	n/
2 / 2 (100%)	n/
9 / 10 (90%)	n/
10 / 11 (90.91%)	n/
6 / 6 (100%)	n/
5 / 5 (100%)	n/
0 / 0 (n/a)	n/
6 / 6 (100%)	n/
5 / 5 (100%)	n/
2 / 2 (100%)	n/
33 / 132 (25%)	n/
29 / 60 (48.33%)	n/
49 / 69 (71.01%)	n/
5 / 5 (100%)	n/
1 / 2 (50%)	n/
1 / 2 (50%)	n/
16 / 31 (51.61%)	n/
8 / 8 (100%)	n/
18 / 21 (85.71%)	n/
148 / 706 (20.96%)	n/
186 / 940 (19.79%)	n/
0 / 0 (n/a)	n/
0 / 0 (n/a)	n/

Showing 40 items

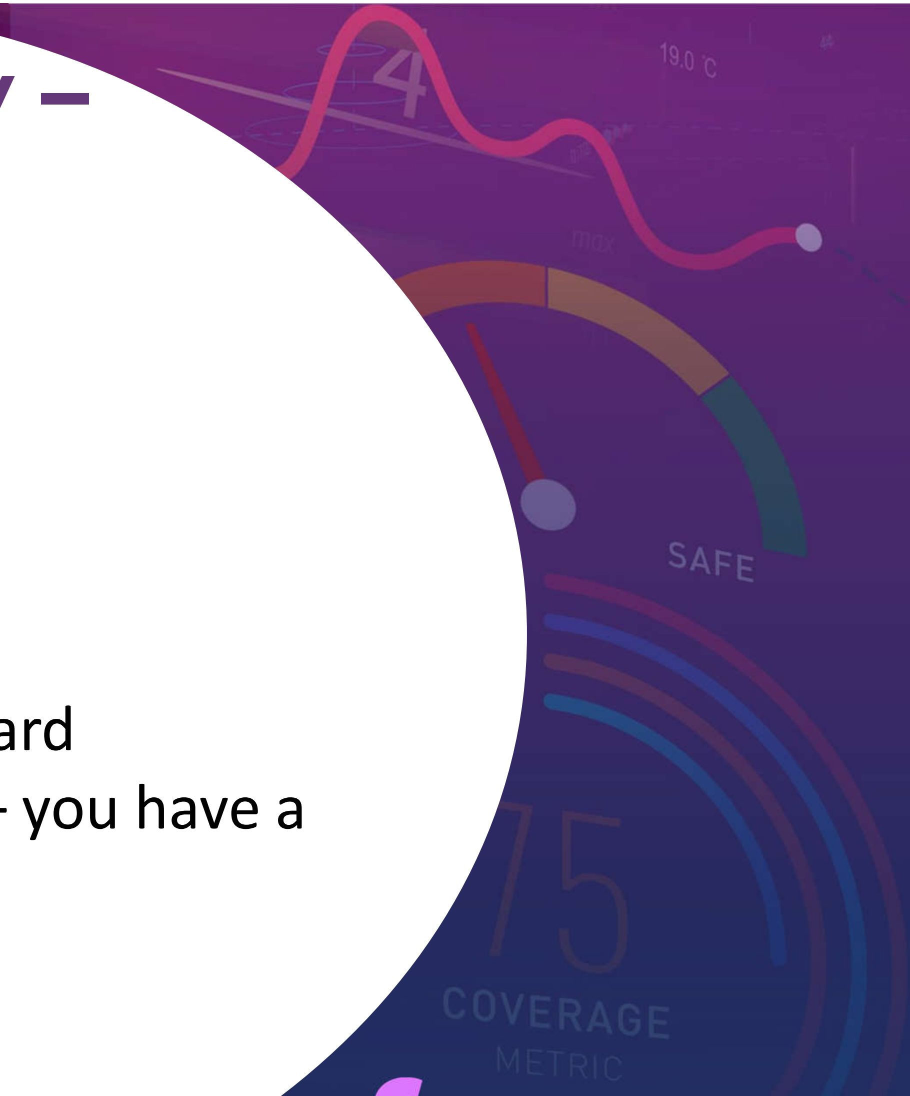
Loaded Run: Session

2.2.6.1.1 Cut in and slow	71.32%
2.2.6.1.1.1 cut_in_and_slow	0%
2.2.6.1.1.2 cut_in_and_slow_no_stop_collisior	0%
2.2.6.1.1.3 side	100%
2.2.6.1.1.4 ego_speed	90%
2.2.6.1.1.5 other_speed_diff	90.91%
2.2.6.1.1.6 other_distance	100%
2.2.6.1.1.7 cut_in_time	100%
2.2.6.1.1.8 other_speed_at_slow_down	n/a
2.2.6.1.1.9 slow_down_value	100%
2.2.6.1.1.10 slow_down_time	100%
2.2.6.1.1.11 ego_changed_lane	100%
2.2.6.1.1.12 cross__side__other_distance__othe	25%
2.2.6.1.1.13 cross__slow_down_time__slow_dov	48.33%
2.2.6.1.1.14 ego_behavior_instance	72.89%
2.2.6.1.1.14.1 ego_lane_shifts	100%
2.2.6.1.1.14.2 ego_collision	50%
2.2.6.1.1.14.3 ego_stopped	50%
2.2.6.1.1.14.4 ego_min_ttc	51.61%
2.2.6.1.1.14.5 ego_max_acceleration	100%
2.2.6.1.1.14.6 ego_max_braking	85.71%
2.2.7 Intersections and mergers	23.53%
2.2.8 Mixes	61.23%
2.2.9 Objects and roadblocks	0%
2.3 Dirt Roads	0%

Summary: Measurable Safety – Coverage Metrics

- **Usage of Coverage Metrics Supplies:**
  - **Goals for testing**
  - **Threshold of quality and safe behaviors**
  - **Relative comparison between AVs**

- With Coverage Driven Verification, Using standard templates, standard testing libraries and ODD – you have a complete certification system



Backup Slides

- Applying CDV for SOTIF
- Misc. additional information

Safety Of The Intended Functionality (SOTIF)

“Absence of unreasonable risk due to hazards resulting from functional insufficiencies of the intended functionality or from reasonably foreseeable misuse by persons”

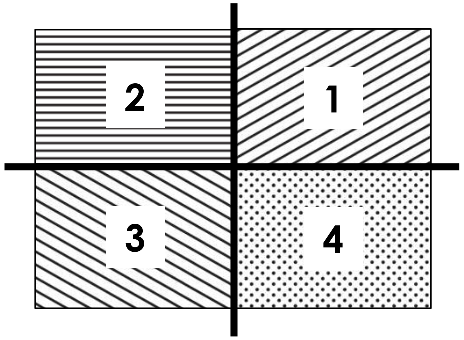
- SOTIF (ISO 21448) is dealing with Safety of Autonomous Systems, and provides guidance on design, verification, and validation measures
- SOTIF breaks down the possible scenario space to 4 categories
- “The ultimate goal is to evaluate the safety in **area 2 and area 3** and to provide an argument that these areas are **sufficiently small and the resulting residual risk is acceptable”**

▤ Known, hazardous scenarios (Area 2)

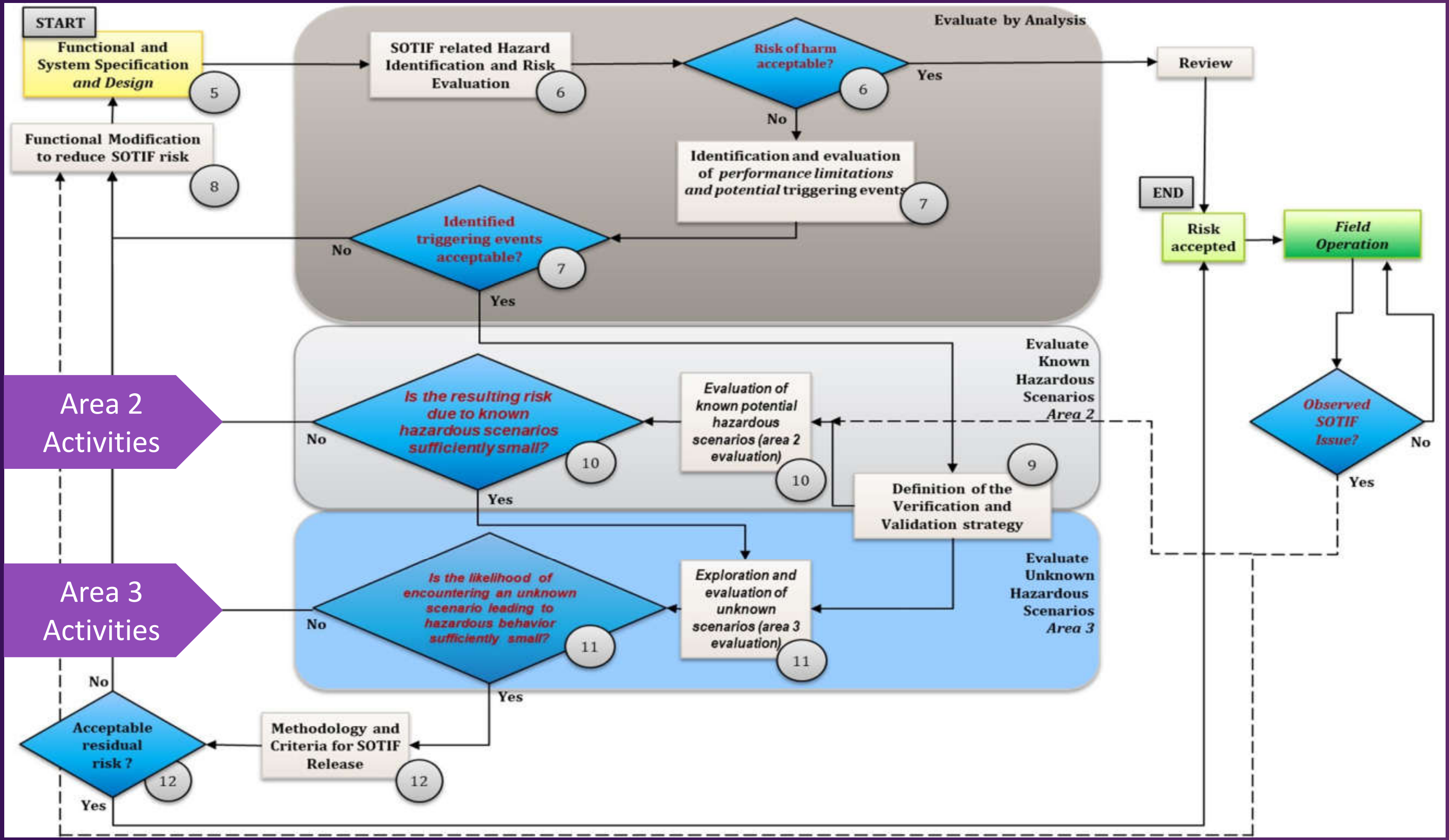
▨ Known, not hazardous scenarios (Area 1)

▧ Unknown, hazardous scenarios (Area 3)

░ Unknown, not hazardous scenarios (Area 4)

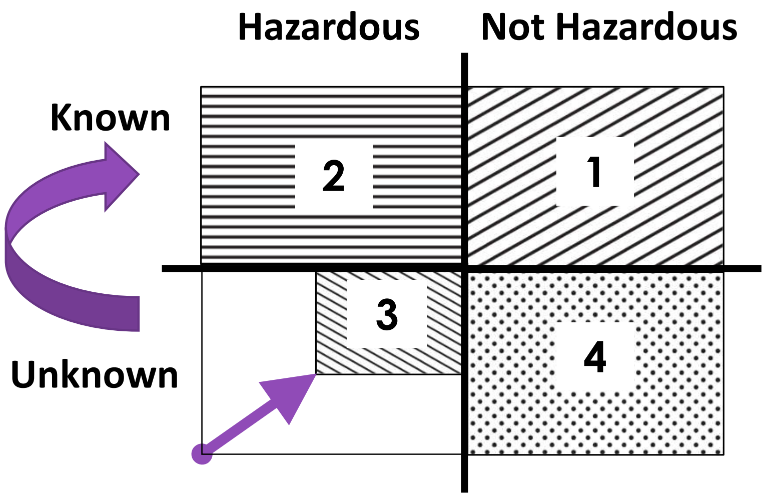


Flowchart of the ISO 21448 activities


START
Functional and System Specification and Design
5
SOTIF related Hazard Identification and Risk Evaluation
6
Evaluate by Analysis
Risk of harm acceptable?
6
Yes
No
Review
Functional Modification to reduce SOTIF risk
8
Identification and evaluation of performance limitations and potential triggering events
7
END
Risk accepted
Field Operation
Identified triggering events acceptable?
7
No
Yes
Evaluate Known Hazardous Scenarios Area 2
Area 2 Activities
Is the resulting risk due to known hazardous scenarios sufficiently small?
10
No
Yes
Evaluation of known potential hazardous scenarios (area 2 evaluation)
10
9
Definition of the Verification and Validation strategy
Observed SOTIF Issue?
No
Yes
Evaluate Unknown Hazardous Scenarios Area 3
Area 3 Activities
Is the likelihood of encountering an unknown scenario leading to hazardous behavior sufficiently small?
11
No
Yes
Exploration and evaluation of unknown scenarios (area 3 evaluation)
11
No
Acceptable residual risk ?
12
Yes
Methodology and Criteria for SOTIF Release
12

Challenge #1 - Area 3: You don’t know what you don’t know!

- ▤ Known, hazardous scenarios (Area 2)
- ▨ Known, not hazardous scenarios (Area 1)
- ▧ Unknown, hazardous scenarios (Area 3)
- ░ Unknown, not hazardous scenarios (Area 4)

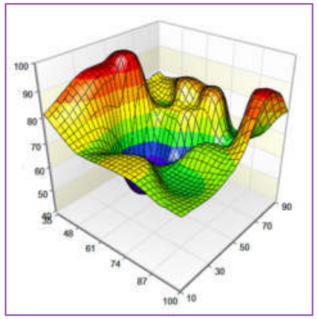


Challenge #2 - Overall: The rules of the game have changed!

Cars/Vehicles evolved from mechanical systems to complex software-controlled systems

Mechanical hazards vs. Software hazards

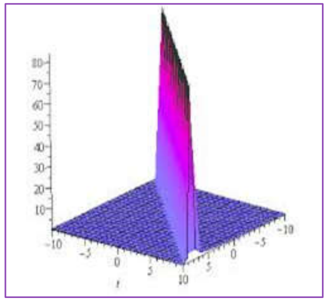
- Continuous, analog-like phenomena
- Statistical methods can help predict failure triggers

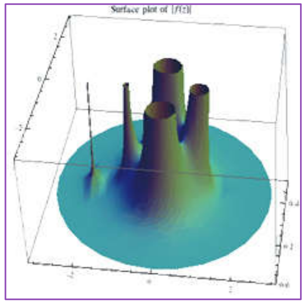


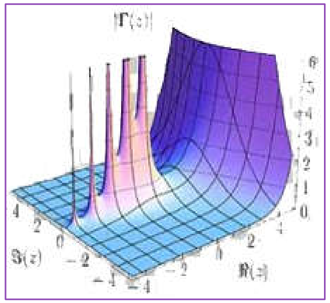
- Systematic and Singular

OR

- Random by nature







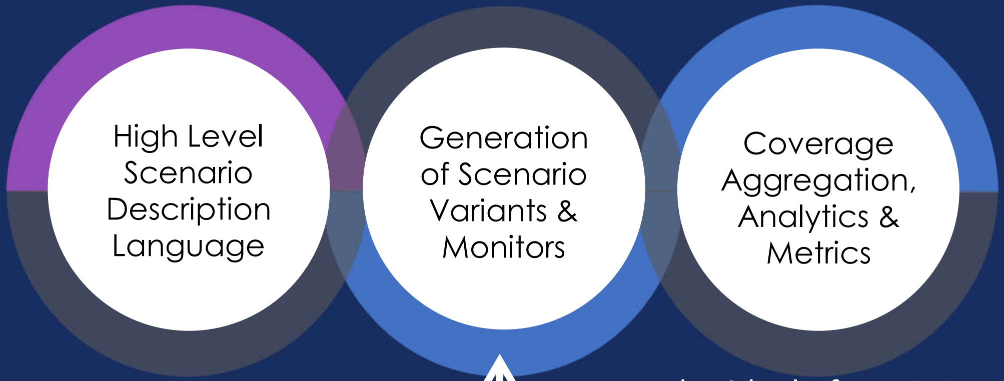
FTA (1962) , FMEA (1949) , HACCP (1960s') and other methods that evolved to analyze mechanical hazards are less effective.

(ISO-26262 integrity assumptions ? Performance Limitation ?)

CDV is a SOTIF enabler

Data Driven Measurable Safety

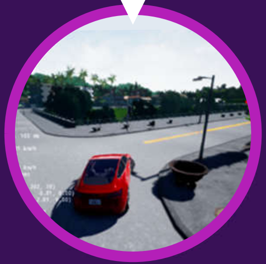
Coverage Driven Verification


High Level Scenario Description Language
Generation of Scenario Variants & Monitors
Coverage Aggregation, Analytics & Metrics

Framework with Platform APIs

Quality of Coverage

Quantity of Miles



Simulation

X-in-the-Loop

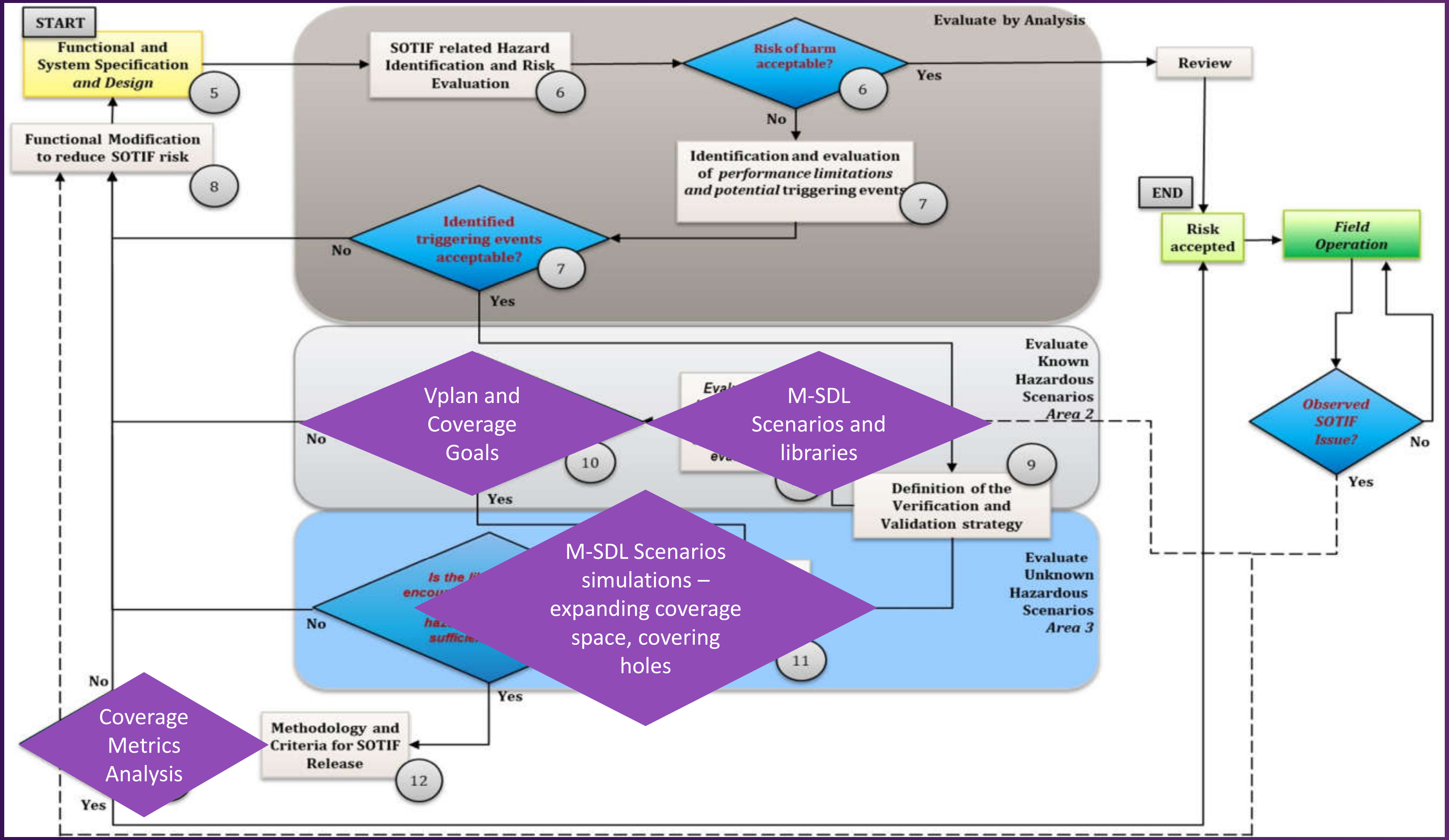
Test Tracks

Test Driving

Securing America's Future Energy

foretellix
verifying autonomous systems

Flowchart of the ISO 21448 activities

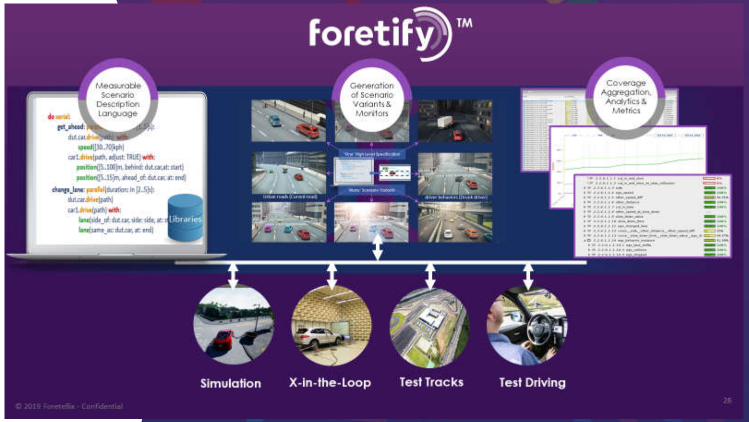

START
Functional and System Specification and Design
5
SOTIF related Hazard Identification and Risk Evaluation
6
Evaluate by Analysis
Risk of harm acceptable?
6
Yes
No
Review
Functional Modification to reduce SOTIF risk
8
Identification and evaluation of performance limitations and potential triggering events
7
END
Identified triggering events acceptable?
7
No
Yes
Risk accepted
Field Operation
Evaluate Known Hazardous Scenarios Area 2
Vplan and Coverage Goals
M-SDL Scenarios and libraries
No
10
9
Yes
Definition of the Verification and Validation strategy
Observed SOTIF Issue?
No
Yes
Evaluate Unknown Hazardous Scenarios Area 3
M-SDL Scenarios simulations – expanding coverage space, covering holes
No
11
Yes
No
Coverage Metrics Analysis
Methodology and Criteria for SOTIF Release
12
Yes

Securing America's Future Energy

foretellix

foretify™ & SOTIF

- Foretify™ is an **automation and analysis tool,** implementing the **Coverage Driven Verification methodology**
- Foretify™ provides **a systematic approach** to reduce both **area 2 and area 3**
- Foretify™ supports the SOTIF process, intended for reaching **acceptable levels of risk**


foretify™
Measurable Scenario Description Language
Generation of Scenario Variants & Monitors
Coverage Aggregation, Analytics & Metrics
Libraries
Simulation
X-in-the-Loop
Test Tracks
Test Driving

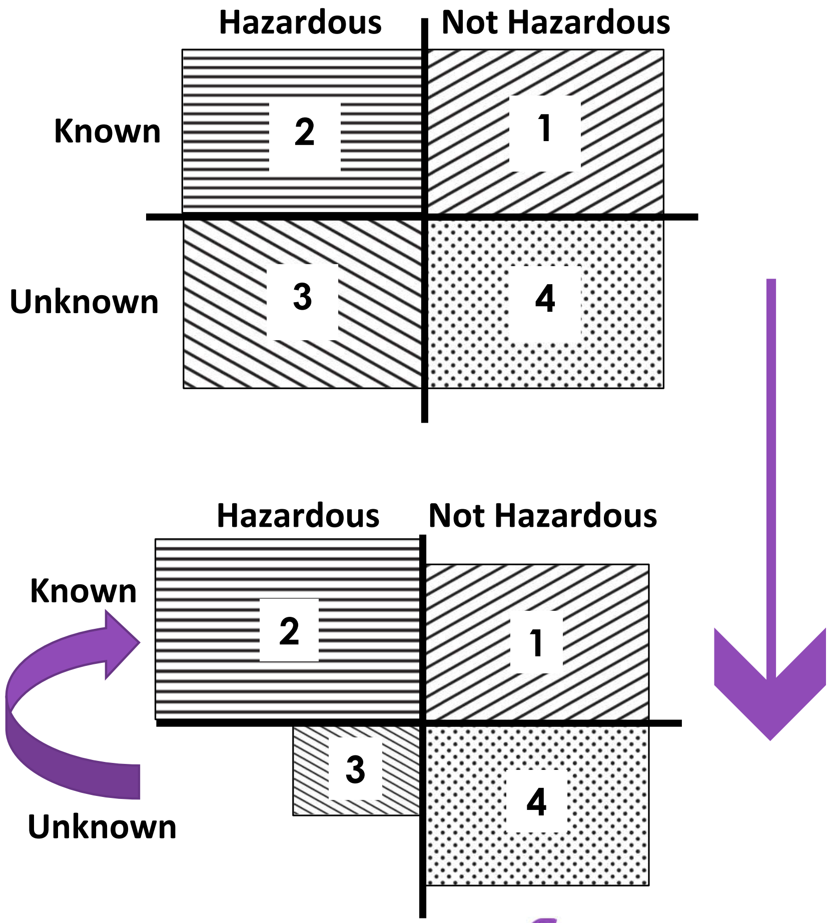
```
do serial:
    get_ahead: [illegible] [1..5]s):
        dut.car.drive(path) with:
            speed([30..70]kph)
        car1.drive(path, adjust: TRUE) with:
            position([5..100]m, behind: dut.car,at: start)
            position([5..15]m, ahead_of: dut.car, at: end)
    change_lane: parallel(duration: in [2..5]s):
        dut.car.drive(path)
        car1.drive(path) with:
            lane(side_of: dut.car, side: side, at: s
            lane(same_as: dut.car, at: end)
```

© 2019 Foretellix - Confidential

28

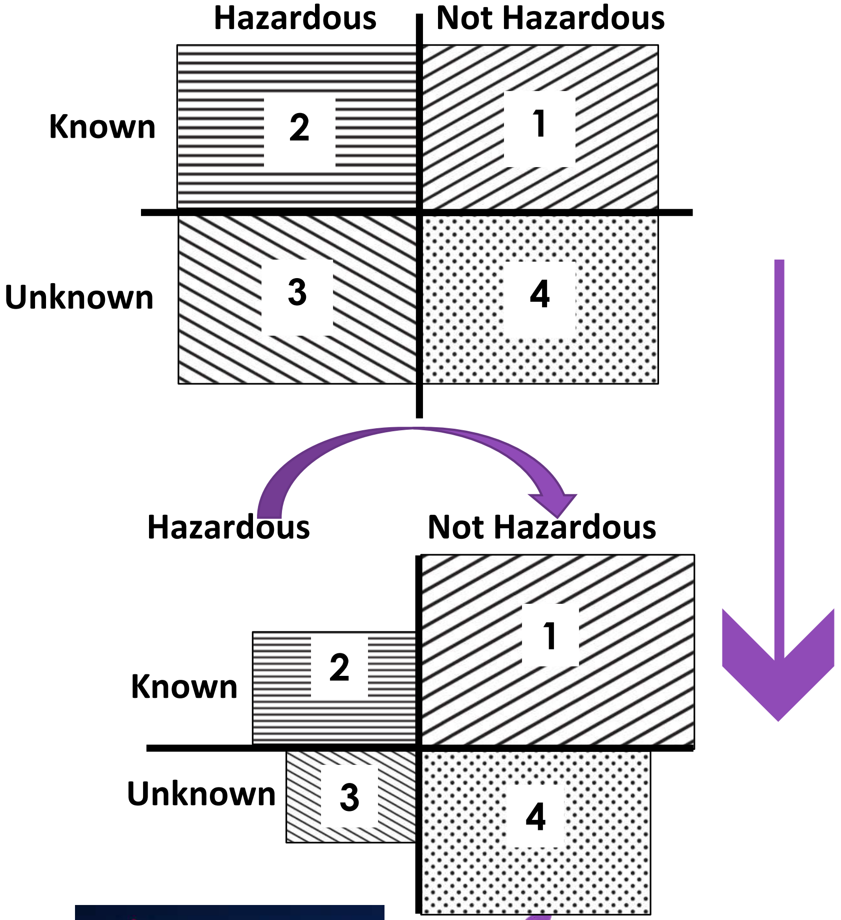
foretify™ & SOTIF-Unknown to Known (area 3)

- Intelligent automation for verification of millions of unknown hazardous scenarios
  - Generate millions of meaningful core and edge cases that are well inside the unknown space (e.g. interaction between systems)
- Exploration the unknown space using
  - Metric based coverage aggregation and analysis (e.g. Coverage holes)
  - Coverage Driven methodology to (e.g. Mixing)

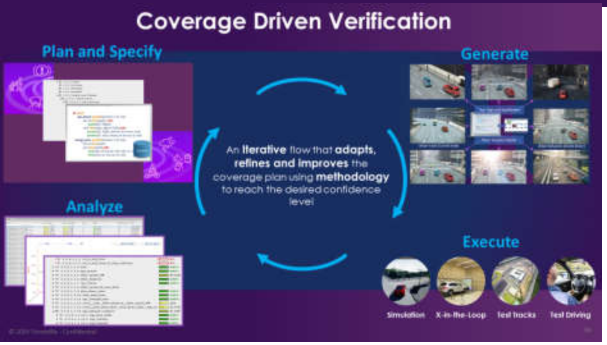


foretify™ & SOTIF-Hazardous to Not Hazardous (area 2)

- Estimate & reduce residual risk by using various techniques (e.g MCMC) on top of a framework of existing scenarios
- Correlate & calibrate between real and virtual testing platforms using Foretify, which is running on the different environments
- Validate bug fixes using Coverage driven Verification
- Monitor recordings / data logs / simulations and split into scenarios to help obtain ODD scenario statistics
  - For verifications of bugs that are statistical / ODD related in nature
  - For measuring and quantifying risk



foretify™ – The full SOTIF flow



Coverage Driven Verification

Plan and Specify

Generate

An **Iterative** flow that **adapts, refines and improves** the coverage plan using **methodology** to reach the desired confidence level

Analyze

Execute

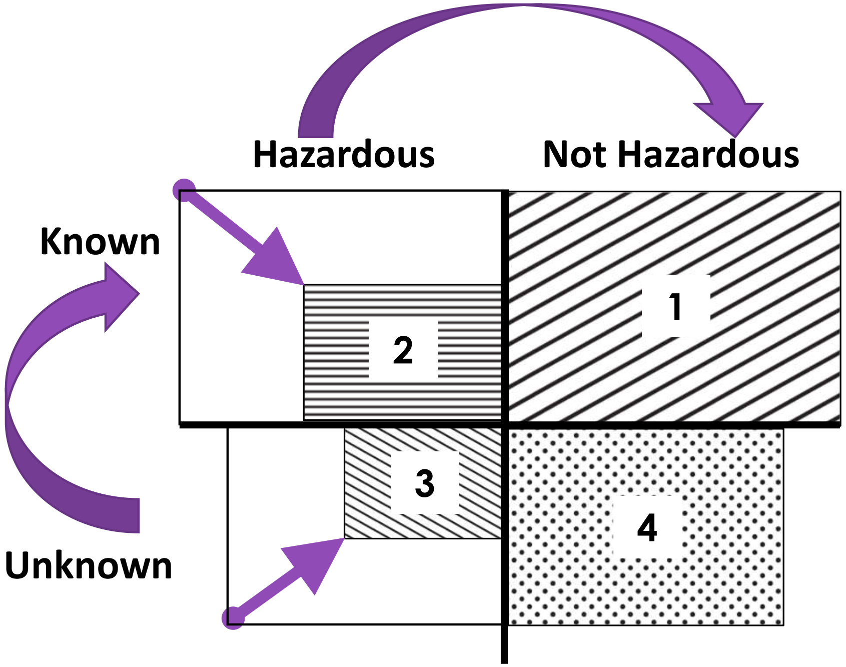
Simulation

X-in-the-Loop

Test Tracks

Test Driving

- ▤ Known, hazardous scenarios (Area 2)
- ▨ Known, not hazardous scenarios (Area 1)
- ▧ Unknown, hazardous scenarios (Area 3)
- ⠿ Unknown, not hazardous scenarios (Area 4)



foretellix

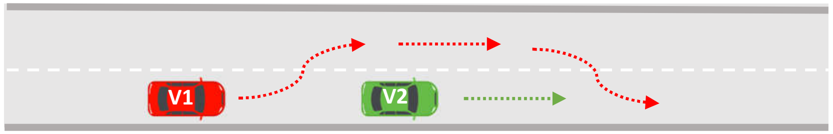
verifying autonomous systems

Misc Info

Securing America's Future Energy

Measurable Scenario Description Language (M-SDL)

- Open, Non-proprietary
- One of the leading candidates in the ASAM OpenScenario 2.0 standardization effort
- A good combination of
  - Power: Ability to write currently-unimagined scenarios
  - Readability: For both simple *and* complex scenarios
  - composability: Critical enabler to maintain readability
- Portable across different execution platforms (simulation, X in a loop, test tracks and street driving) , across levels of abstraction (very direct to very abstract), across use cases etc.
- Enables a constrained-random coverage-driven verification
- Extensible
- Can be visualized and textualized
- Dual – make the scenario happen but also monitor for its occurrence



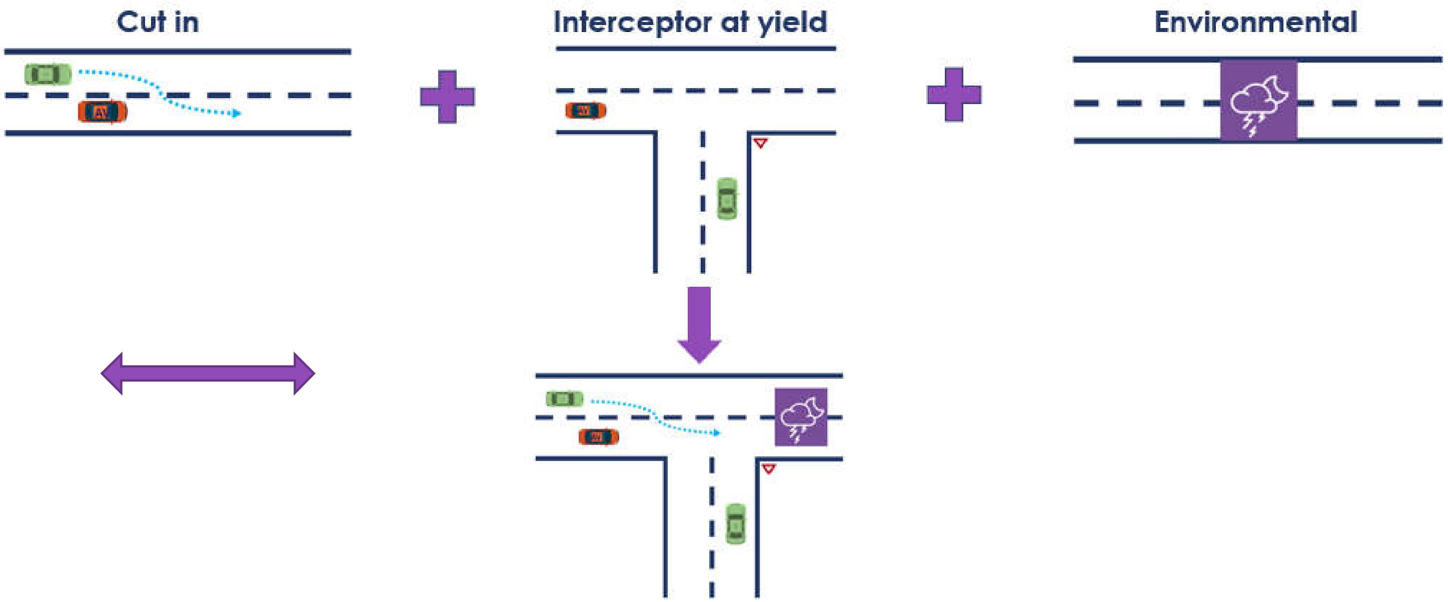
```
scenario traffic.overtake:
    v1: car # The first car
    v2: car # The second car
    p: path

    do parallel(duration: [3..20]s):
        v2.drive(p)
        serial:
            A: v1.drive(p) with:
                lane(same_as: v2, at: start)
                lane(left_of: v2, at: end)
                position([10..20]m, behind: v2, at: start)
            B: v1.drive(p)
            C: v1.drive(p) with:
                lane(same_as: v2, at: end)
                position([5..10]m, ahead_of: v2, at: end)
```

Securing America's Future Energy

foretellix

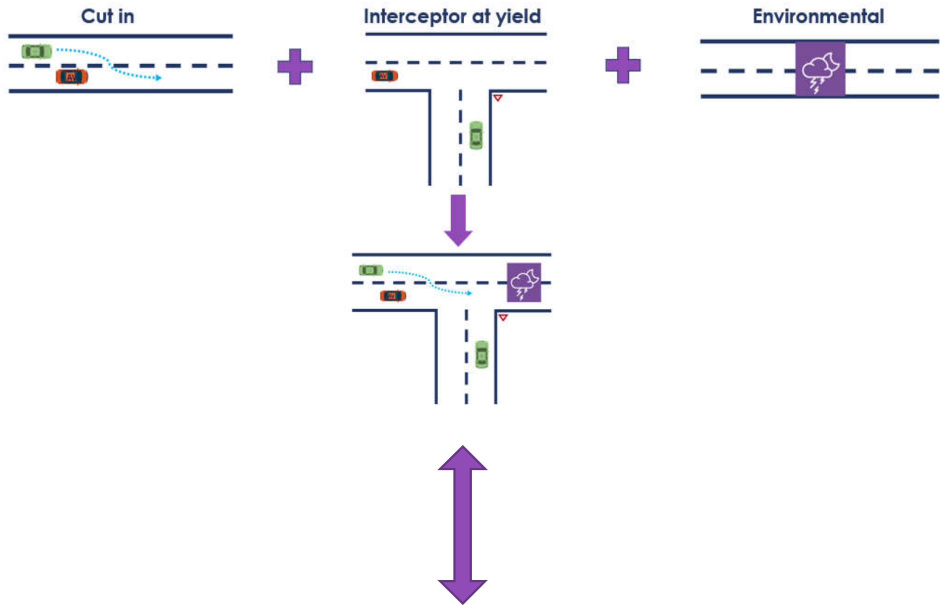
CDV Methodology: Mixing Scenarios

```
scenario dut.mix_dangers:
      do mix:
         cut_in_and_Slow()
         interceptor_at_yield()
         set_weather(rain)
```



- Create many more meaningful scenarios and extend your coverage by mixing and overlaying different scenarios
- Create Combinations of Combinations of edge cases and scenarios a human cannot think about
- Use a proven methodology together with innovative ML to improve coverage of critical safety metrics

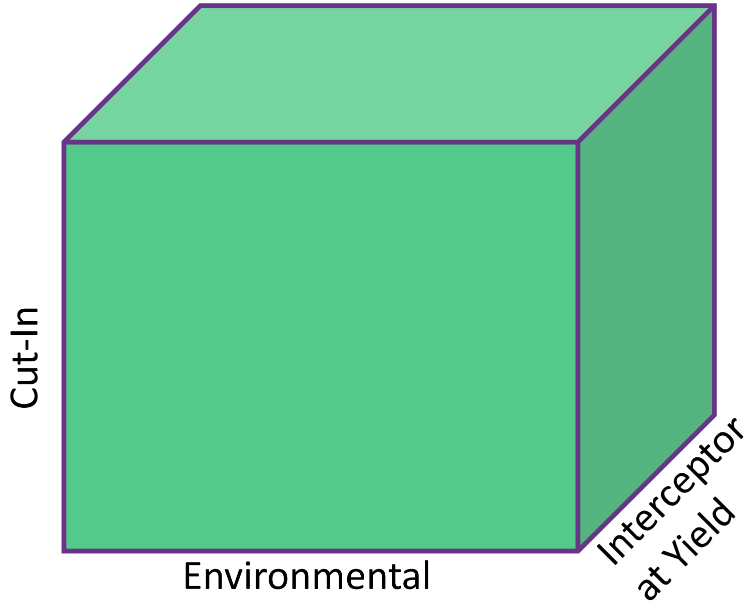
Scenario Mixing Example – Cut-in & Interceptor at Yield & Environmental



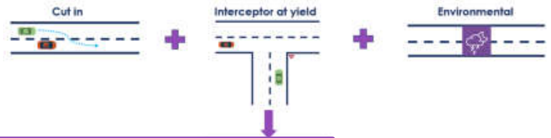
```
scenario dut.mix_dangers:
      do mix:
         cut_in_and_Slow()
         interceptor_at_yield()
         set_weather(rain)
```

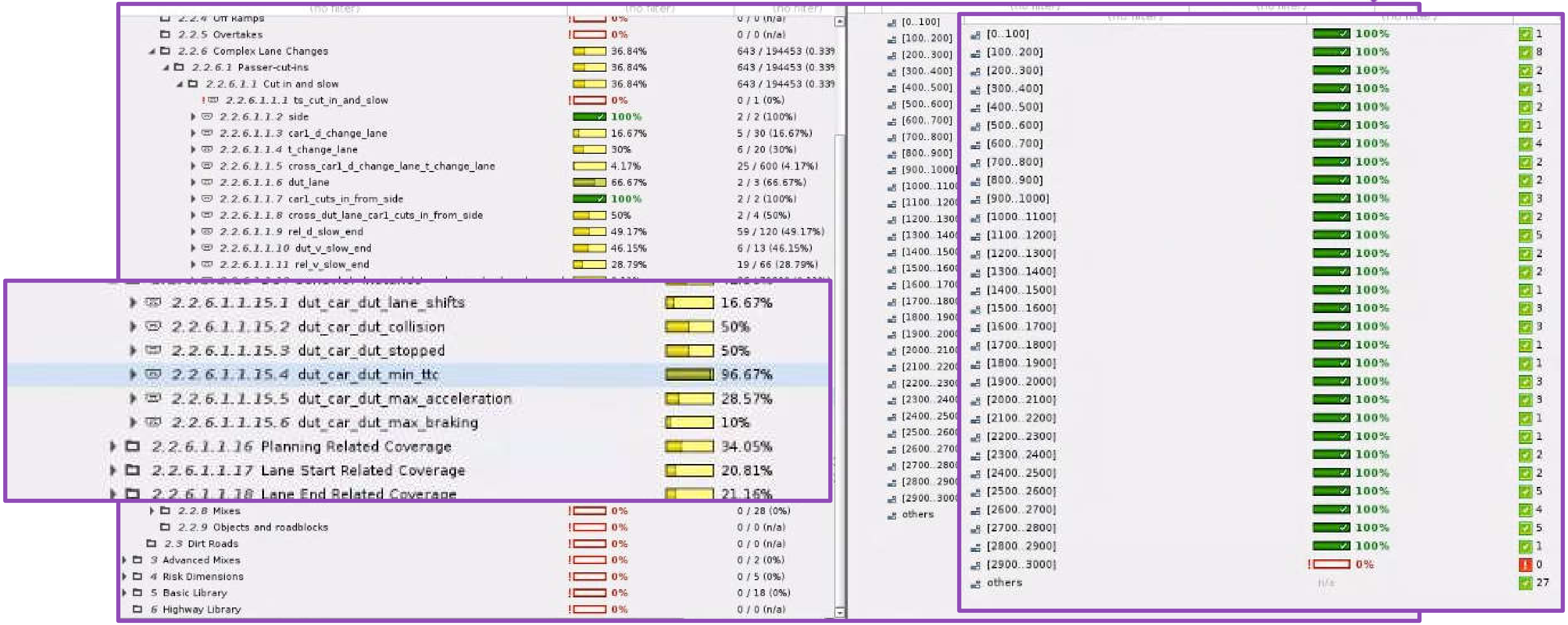
- Mixing scenarios expands the verification scope to a volume that is the product of individual verification scopes' volumes

- Constraint random exploration of the 'mixed volume' enables bug space exploration beyond human imagination and capacity to implement



After Mixing: TTC Coverage Improved





(no filter)	(no filter)	(no filter)
2.2.4 Off Ramps	! 0%	0 / 0 (n/a)
2.2.5 Overtakes	! 0%	0 / 0 (n/a)
2.2.6 Complex Lane Changes	36.84%	643 / 194453 (0.33%
2.2.6.1 Passer-cut-ins	36.84%	643 / 194453 (0.33%
2.2.6.1.1 Cut in and slow	36.84%	643 / 194453 (0.33%
! 2.2.6.1.1.1 ts_cut_in_and_slow	! 0%	0 / 1 (0%)
2.2.6.1.1.2 side	✓ 100%	2 / 2 (100%)
2.2.6.1.1.3 car1_d_change_lane	16.67%	5 / 30 (16.67%)
2.2.6.1.1.4 t_change_lane	30%	6 / 20 (30%)
2.2.6.1.1.5 cross_car1_d_change_lane_t_change_lane	4.17%	25 / 600 (4.17%)
2.2.6.1.1.6 dut_lane	66.67%	2 / 3 (66.67%)
2.2.6.1.1.7 car1_cuts_in_from_side	✓ 100%	2 / 2 (100%)
2.2.6.1.1.8 cross_dut_lane_car1_cuts_in_from_side	50%	2 / 4 (50%)
2.2.6.1.1.9 rel_d_slow_end	49.17%	59 / 120 (49.17%)
2.2.6.1.1.10 dut_v_slow_end	46.15%	6 / 13 (46.15%)
2.2.6.1.1.11 rel_v_slow_end	28.79%	19 / 66 (28.79%)

2.2.6.1.1.15.1 dut_car_dut_lane_shifts	16.67%
2.2.6.1.1.15.2 dut_car_dut_collision	50%
2.2.6.1.1.15.3 dut_car_dut_stopped	50%
2.2.6.1.1.15.4 dut_car_dut_min_ttc	96.67%
2.2.6.1.1.15.5 dut_car_dut_max_acceleration	28.57%
2.2.6.1.1.15.6 dut_car_dut_max_braking	10%
2.2.6.1.1.16 Planning Related Coverage	34.05%
2.2.6.1.1.17 Lane Start Related Coverage	20.81%
2.2.6.1.1.18 Lane End Related Coverage	21.16%

2.2.8 Mixes	! 0%	0 / 28 (0%)
2.2.9 Objects and roadblocks	! 0%	0 / 0 (n/a)
2.3 Dirt Roads	! 0%	0 / 0 (n/a)
3 Advanced Mixes	! 0%	0 / 2 (0%)
4 Risk Dimensions	! 0%	0 / 5 (0%)
5 Basic Library	! 0%	0 / 18 (0%)
6 Highway Library	! 0%	0 / 0 (n/a)

[0..100]
[100..200]
[200..300]
[300..400]
[400..500]
[500..600]
[600..700]
[700..800]
[800..900]
[900..1000]
[1000..110
[1100..120
[1200..130
[1300..140
[1400..150
[1500..160
[1600..170
[1700..180
[1800..190
[1900..200
[2000..210
[2100..220
[2200..230
[2300..240
[2400..250
[2500..260
[2600..270
[2700..280
[2800..290
[2900..300
others

(no filter)	(no filter)	(no filter)
[0..100]	✓ 100%	✓ 1
[100..200]	✓ 100%	✓ 8
[200..300]	✓ 100%	✓ 2
[300..400]	✓ 100%	✓ 1
[400..500]	✓ 100%	✓ 2
[500..600]	✓ 100%	✓ 1
[600..700]	✓ 100%	✓ 4
[700..800]	✓ 100%	✓ 2
[800..900]	✓ 100%	✓ 2
[900..1000]	✓ 100%	✓ 3
[1000..1100]	✓ 100%	✓ 2
[1100..1200]	✓ 100%	✓ 5
[1200..1300]	✓ 100%	✓ 2
[1300..1400]	✓ 100%	✓ 2
[1400..1500]	✓ 100%	✓ 1
[1500..1600]	✓ 100%	✓ 3
[1600..1700]	✓ 100%	✓ 3
[1700..1800]	✓ 100%	✓ 1
[1800..1900]	✓ 100%	✓ 1
[1900..2000]	✓ 100%	✓ 3
[2000..2100]	✓ 100%	✓ 3
[2100..2200]	✓ 100%	✓ 1
[2200..2300]	✓ 100%	✓ 1
[2300..2400]	✓ 100%	✓ 2
[2400..2500]	✓ 100%	✓ 2
[2500..2600]	✓ 100%	✓ 5
[2600..2700]	✓ 100%	✓ 4
[2700..2800]	✓ 100%	✓ 5
[2800..2900]	✓ 100%	✓ 1
[2900..3000]	! 0%	! 0
others	n/a	✓ 27

After another CDV iteration and mixing between a Cut In scenario and an interceptor Scenario coverage **of the min TTC Metric was improved**

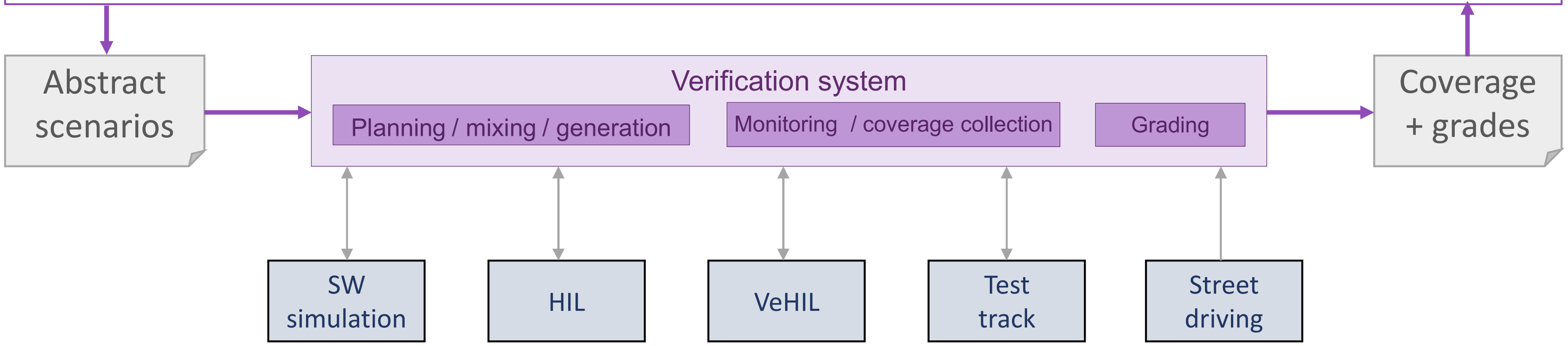
Using automation and coverage to track those risk dimensions

Risk dimensions (as a coverage plan)

- Driving scenarios
- Topology
- Locale
- Weather
- Other participants
- Emergency situations

- Human behavior
- ML limitations
- Locale change
- Mode confusions
- Sensing issues
- Planning issues

- Arbitrary subsystem failures
- Interactions with other AVs
- Production / maintenance variability
- Behavior during / after accidents
- **Mixing any of the above**
- …



Abstract scenarios

Verification system

Planning / mixing / generation

Monitoring / coverage collection

Grading

Coverage + grades

SW simulation

HIL

VeHIL

Test track

Street driving

Securing America's Future Energy