Agenda

Session 1: 1:00 PM to 1:50 PM

Introduction to NVIDIA Halos - Strategy for AV Safety
- Chapter 1: Overview
- Chapter 2: Design-time safety guardrails

Session 2: 2:00 PM to 2:50 PM

Guardrails for NVIDIA Halos Across the Product Life Cycle
- Chapter 3: Deployment-time guardrails
- Chapter 4: Validation-time guardrails

Session 3: 3:00 PM to 3:50 PM

Safety Regulation and Standardization in the Era of AI-Based AV
- Chapter 5: Safety regulation and standardization
- Chapter 6: From AVs to general Physical AI

Session 4: 4:00 to 5:00 PM

Navigating the High-Stakes Safety Challenges of AVs

Chapters 1 & 2: Overview & Design-Time Safety Guardrails

Marco Pavone, Sr. Director, Autonomous Vehicle Research, NVIDIA

Jonas Nilsson, Director of Safety Engineering, NVIDIA

Karl Greb, Sr. Director of Safety Engineering, NVIDIA

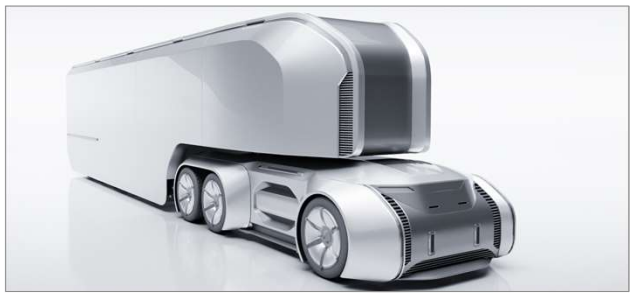
Everything that Moves will be Autonomous

Autonomous Vehicles Rely on AI

Cars

Robotaxis

Delivery Vehicles

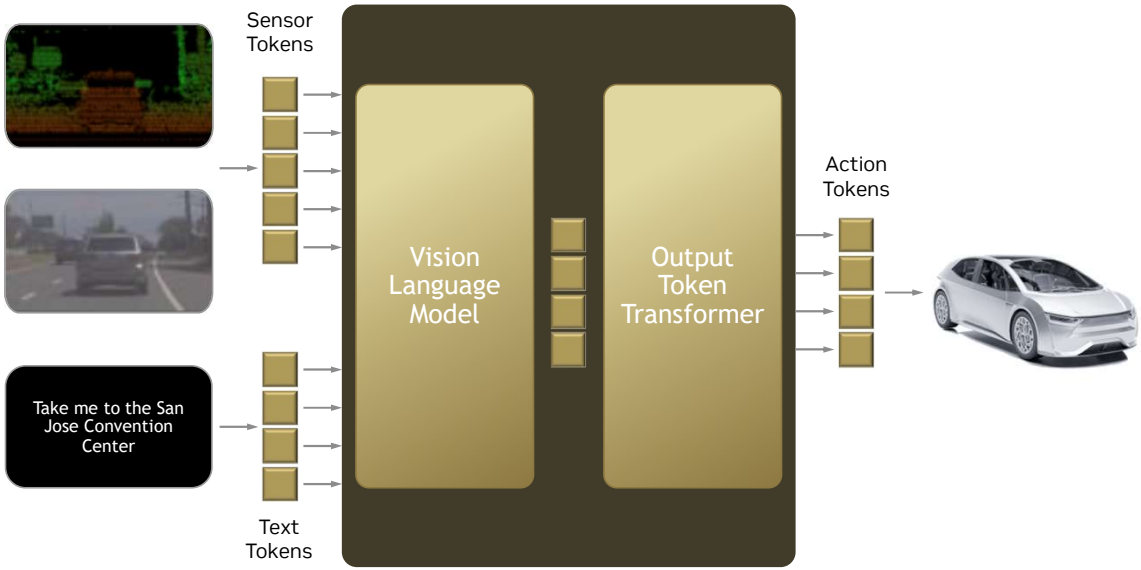


Trucks

Buses

Tractors

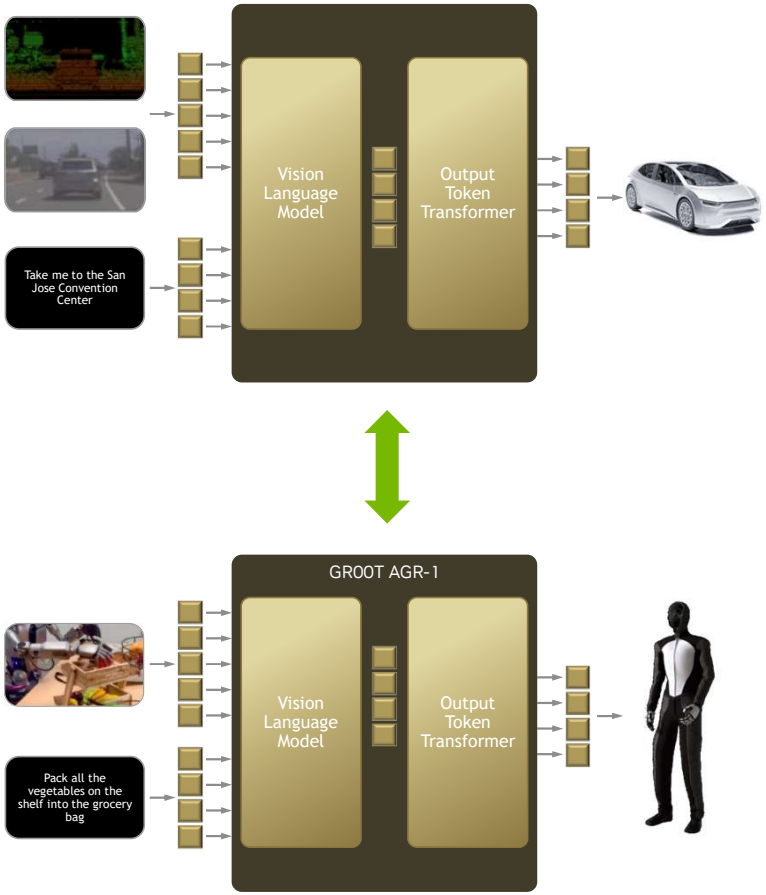
E2E and Foundation Models are Enabling New Capabilities for AVs

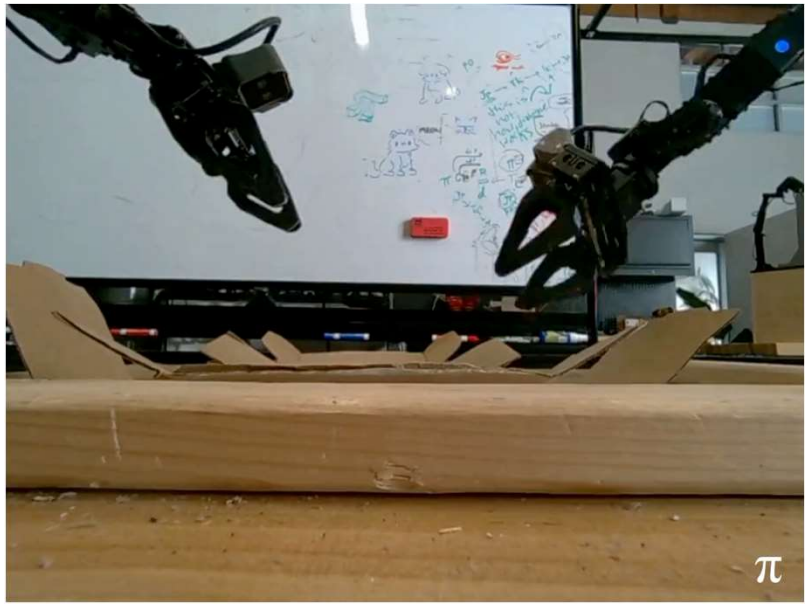


Wayve

NVIDIA E2E Foundation Model

Architectures are Converging for Physical AI





Physical Intelligence

“

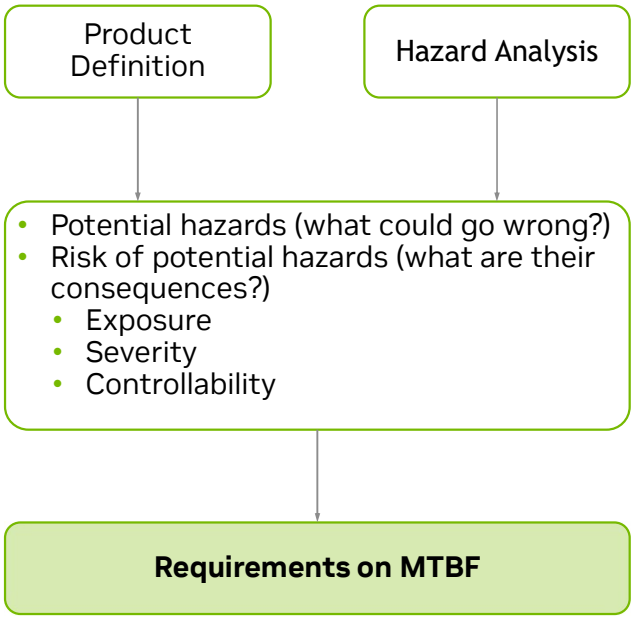
How do we assure the safe performance of E2E, possibly foundation-model-empowered AV stacks?

”

What Do We Mean by “Safe Performance”?

Defining, Meeting, and Validating MTBF Targets on AV systems

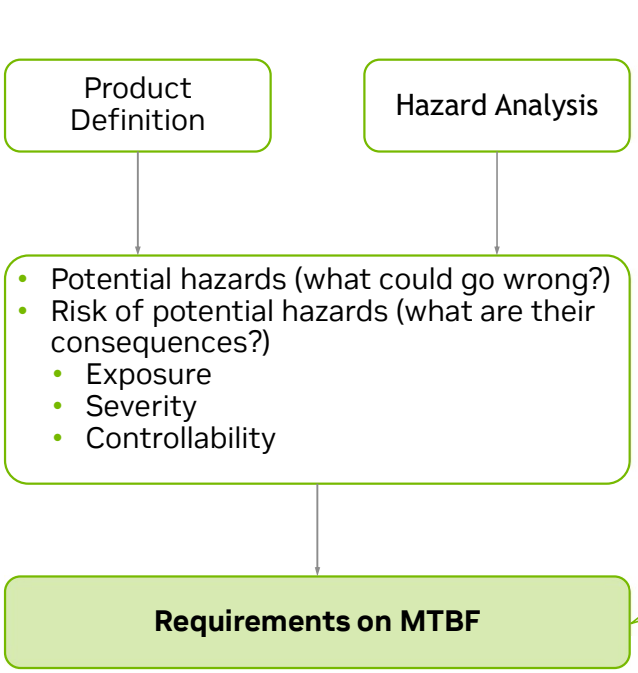
MTBF: Mean time between failures



What Do We Mean by "Safe Performance"?

Defining, Meeting, and Validating MTBF Targets on AV systems

MTBF: Mean time between failures



MTBF targets for L2++ products*

- Limiting autonomy to driving scenarios with high controllability (e.g., L2++ hands-on) can keep targets manageable
  - MTBF ~10^2 hrs, validation data up to 10^4 hrs
- Targeting greater availability (e.g. L3) means stricter system-level targets
  - MTBF ~10^7 hrs, validation data up to ~10^9 hrs

Validation data must appropriately cover input space!

How do we design a system that can achieve high MTBF targets?

How do we efficiently validate that our system meets MTBF targets?

*Estimates aligned with human fatal crash statistics, "Driving to Safety," RAND Report

Three General Principles: Diversity, Monitoring, and Evidence

Diversity

Robust Systems Engineering
Ensure multiple components with overlapping responsibilities but independent sources of information and reasoning.

Machine Learning (ML)
Collecting diverse training datasets to help ensure the models will never be surprised or encounter out-of-distribution (OOD) scenarios online.

Three General Principles: Diversity, Monitoring, and Evidence

Diversity

Robust Systems Engineering
Ensure multiple components with overlapping responsibilities but independent sources of information and reasoning.

Machine Learning (ML)
Collecting diverse training datasets to help ensure the models will never be surprised or encounter out-of-distribution (OOD) scenarios online.

Monitoring

Monitoring Self
Perform system health monitoring, from battery levels all the way up to human-operator-in-the-loop.

Monitor Environment
Perform Operational Design Domain (ODD) detection to ensure the system is well-equipped to handle the situation.

Three General Principles: Diversity, Monitoring, and Evidence

Diversity

Robust Systems Engineering
Ensure multiple components with overlapping responsibilities but independent sources of information and reasoning.

Machine Learning (ML)
Collecting diverse training datasets to help ensure the models will never be surprised or encounter out-of-distribution (OOD) scenarios online.

Monitoring

Monitoring Self
Perform system health monitoring, from battery levels all the way up to human-operator-in-the-loop.

Monitor Environment
Perform Operational Design Domain (ODD) detection to ensure the system is well-equipped to handle the situation.

Evidence

V & V
Perform Verification and Validation (V&V) across the full range of technology layers comprising the AV stack

Testing of ML Components
Evaluate ML models on test sets, with critical components evaluated on a carefully curated *safety dataset*

What is Unique to E2E from a Safety Standpoint?

Data Requirements to Achieve High MTBF Grow Significantly*

- Abstractions and decompositions lower data requirements to achieve high MTBF

- Only route to achieve high MTBF is data scaling (either synthetic or real)

Synthetic data generation can, however, mitigate this concern

What is Unique to E2E from a Safety Standpoint?

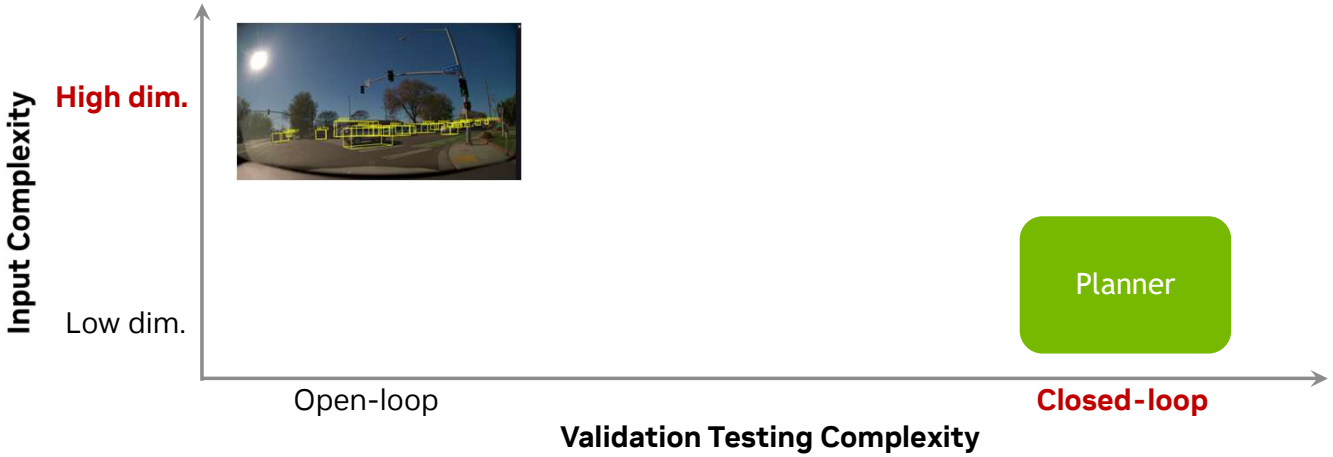
Validating Requirements is More Expensive for E2E

Modular Stack:

What is Unique to E2E from a Safety Standpoint?

Validating Requirements is More Expensive for E2E

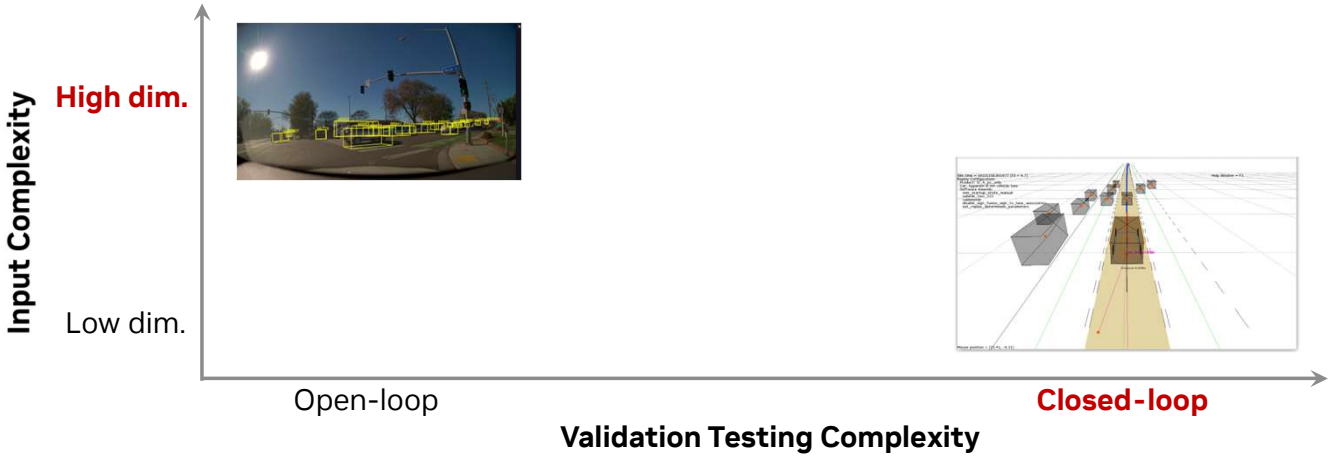
Modular Stack:



What is Unique to E2E from a Safety Standpoint?

Validating Requirements is More Expensive for E2E

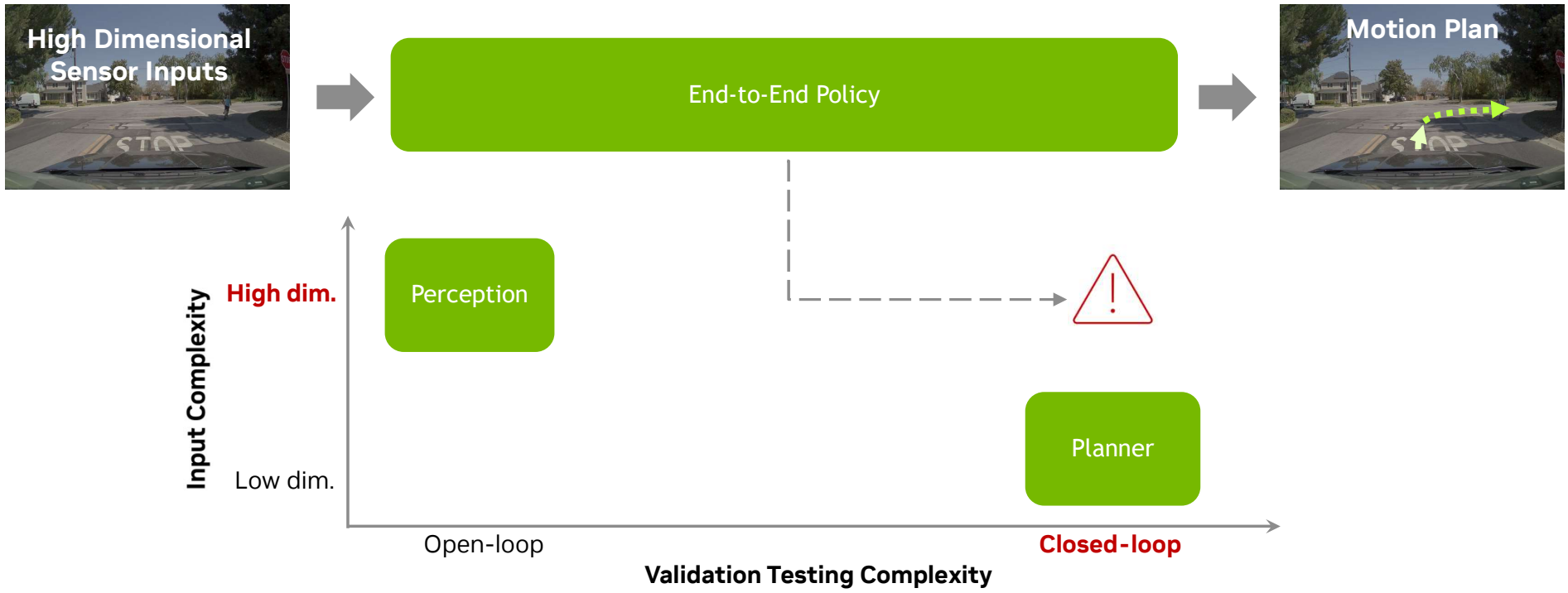
Modular Stack:



What is Unique to E2E from a Safety Standpoint?

Validating Requirements is More Expensive for E2E

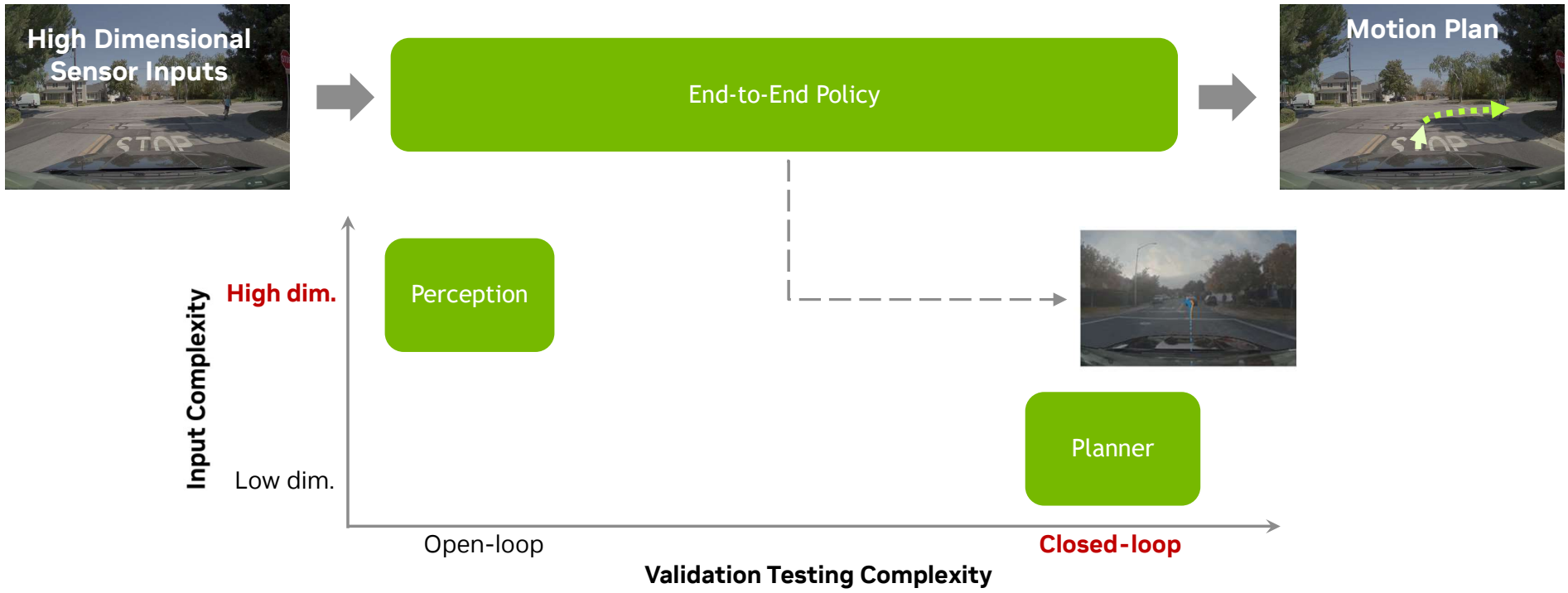
End-to-end Stack:



What is Unique to E2E from a Safety Standpoint?

Validating Requirements is More Expensive for E2E

End-to-end Stack:



What is Unique to E2E from a Safety Standpoint?

Validating Requirements is More Expensive for E2E

End-to-end Stack:

End-to-End Policy

Perception

Planner

Input Complexity

High dim.

Low dim.

Open-loop

Closed-loop

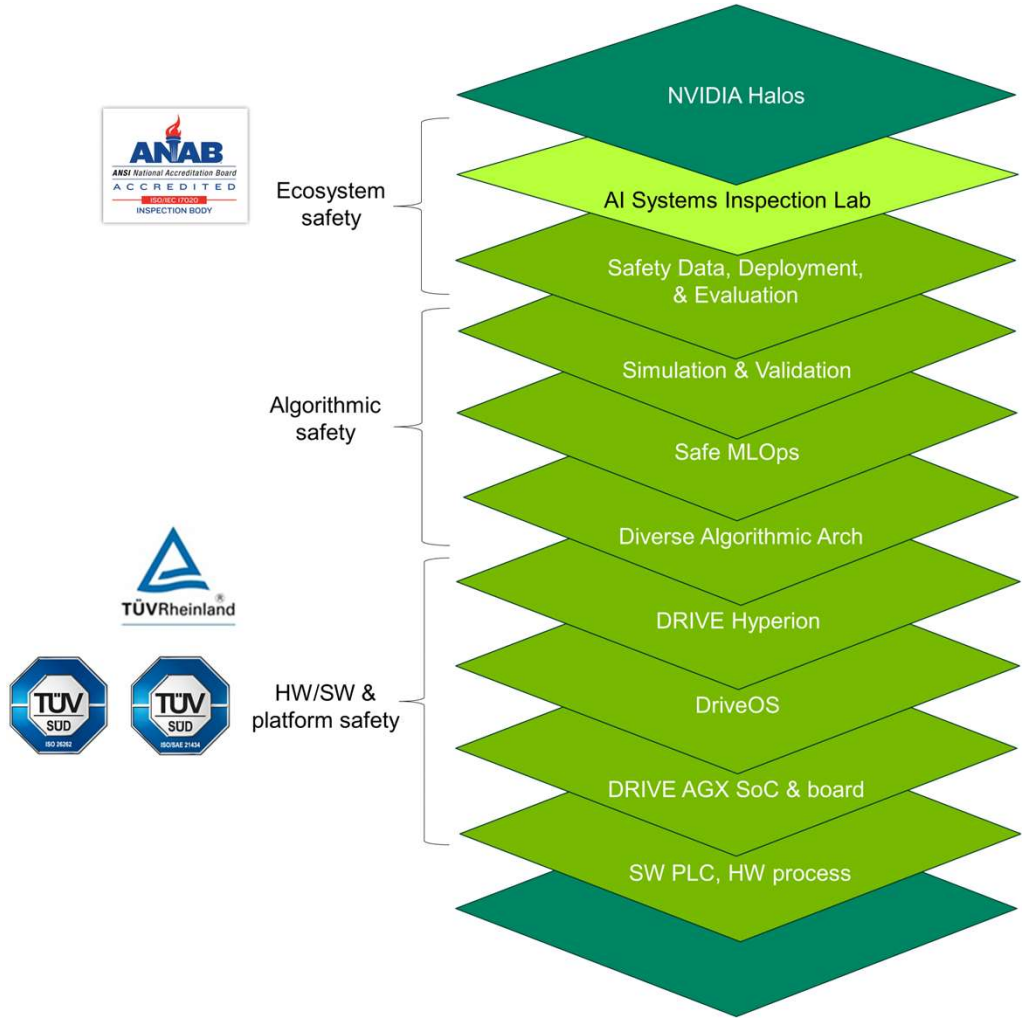
Validation Testing Complexity

data re-use might be much more challenging!!

Introducing Halos

A Full-Stack Comprehensive Safety System for Autonomous Vehicles

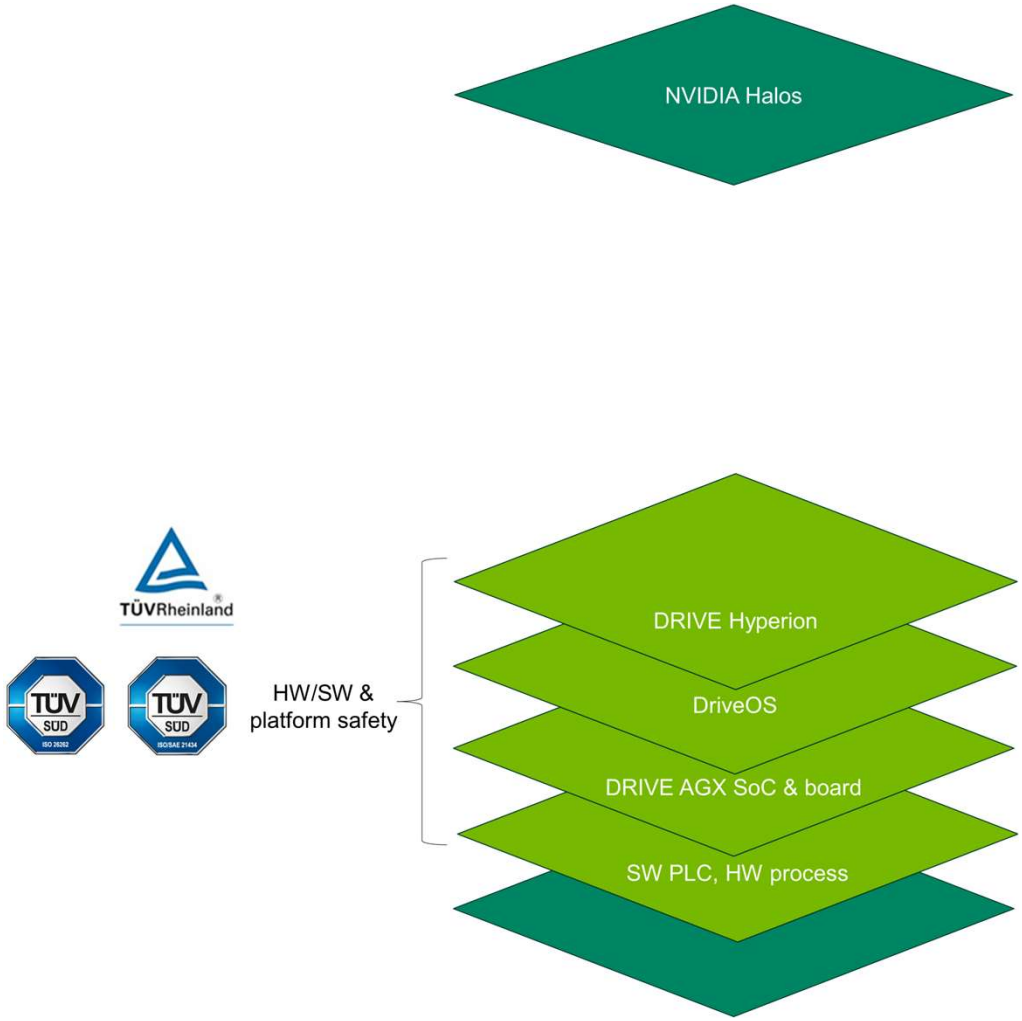
- Halos is a full-stack comprehensive safety system for Autonomous Vehicles that unifies safety elements from vehicle architecture to AI models.
- It comprises HW and SW elements, tools, models, and the design principles for combining them to safeguard AI-based, end-to-end AV stacks.



Introducing Halos

A Full-Stack Comprehensive Safety System for Autonomous Vehicles

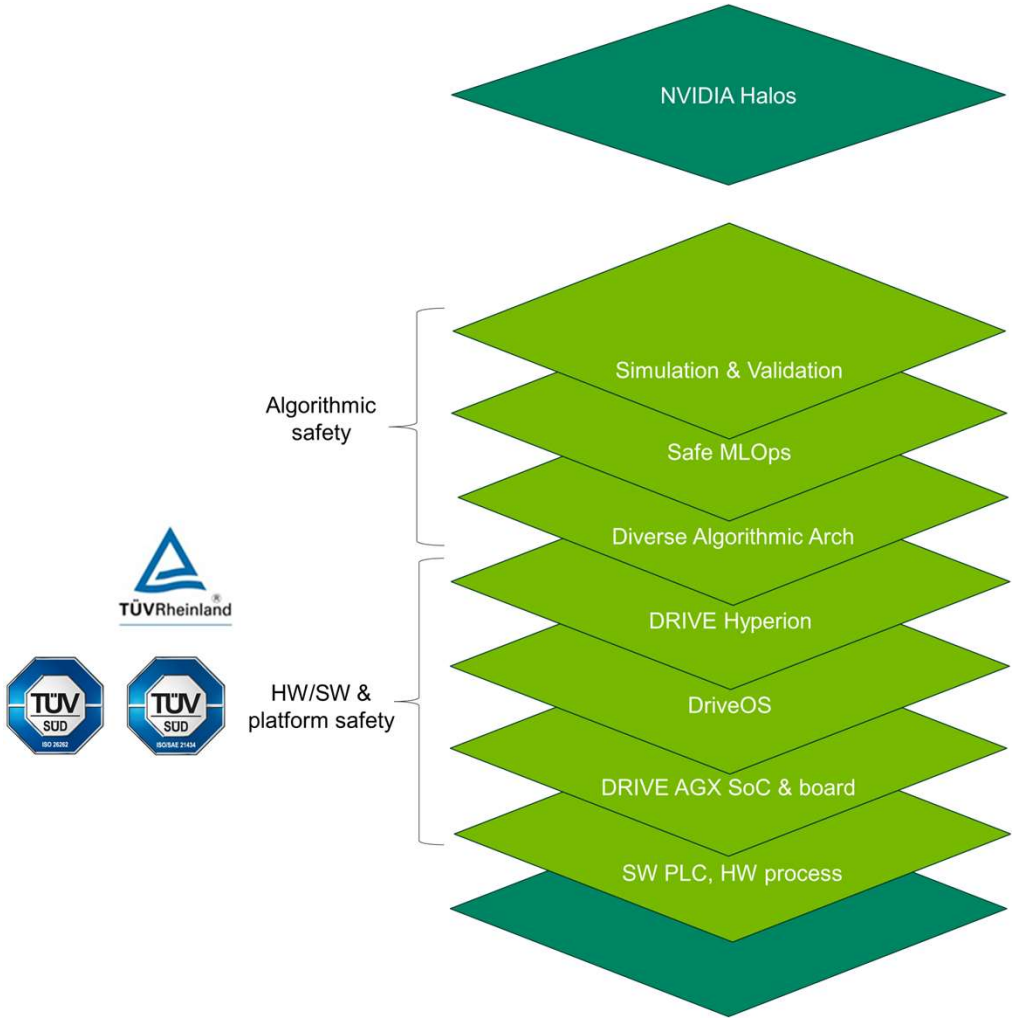
- Halos is a full-stack comprehensive safety system for Autonomous Vehicles that unifies safety elements from vehicle architecture to AI models.
- It comprises HW and SW elements, tools, models, and the design principles for combining them to safeguard AI-based, end-to-end AV stacks.



Introducing Halos

A Full-Stack Comprehensive Safety System for Autonomous Vehicles

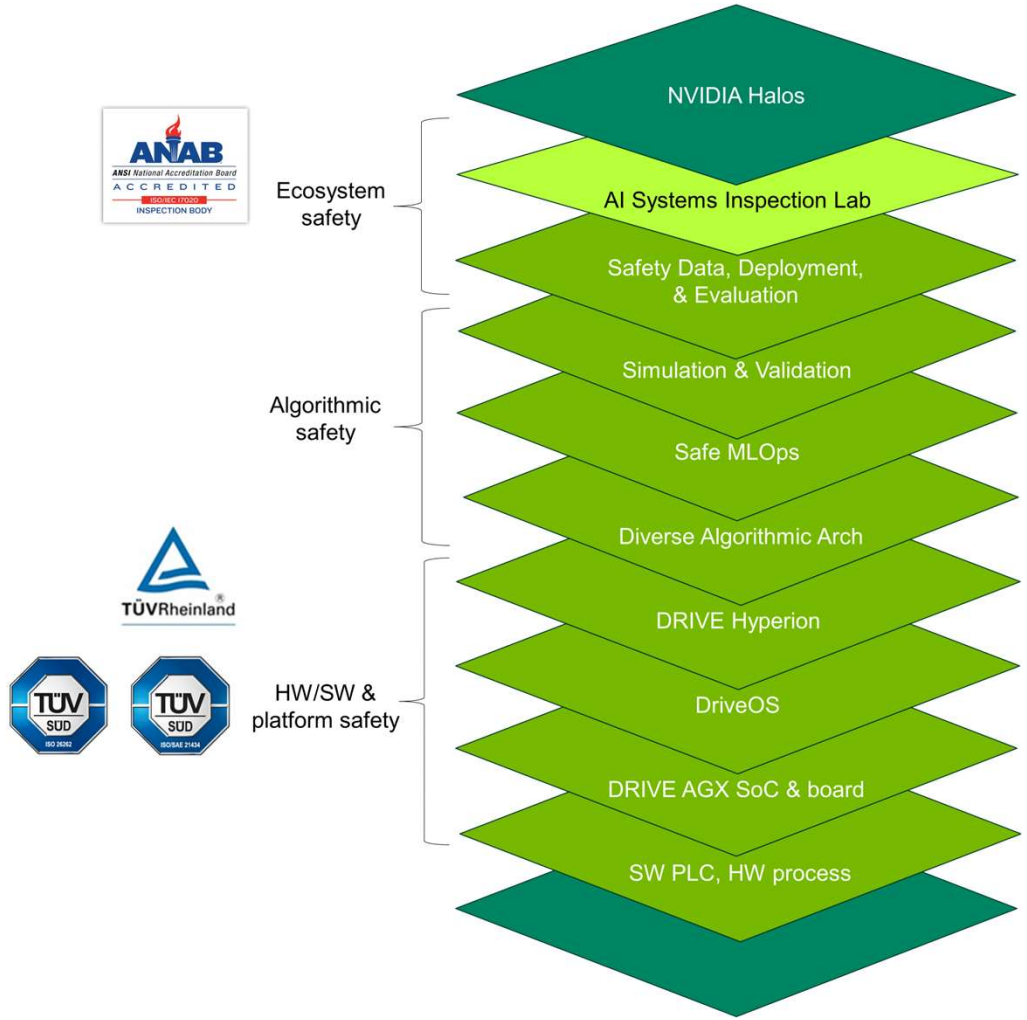
- Halos is a full-stack comprehensive safety system for Autonomous Vehicles that unifies safety elements from vehicle architecture to AI models.
- It comprises HW and SW elements, tools, models, and the design principles for combining them to safeguard AI-based, end-to-end AV stacks.



Introducing Halos

A Full-Stack Comprehensive Safety System for Autonomous Vehicles

- Halos is a full-stack comprehensive safety system for Autonomous Vehicles that unifies safety elements from vehicle architecture to AI models.
- It comprises HW and SW elements, tools, models, and the design principles for combining them to safeguard AI-based, end-to-end AV stacks.



What Makes NVIDIA Halos Unique?

AV Safety Leadership from Research to Engineering

15,000+

Engineering years invested in vehicle safety to date

21 Billion+

Safety transistors safety assessed

7,000,000

Lines of safety-assessed code

2,000,000

Daily end-to-end integration tests for validation

22,000+

Platform safety monitors

20,000+

Hours of safety dataset

1,000+

Patents filed

240+

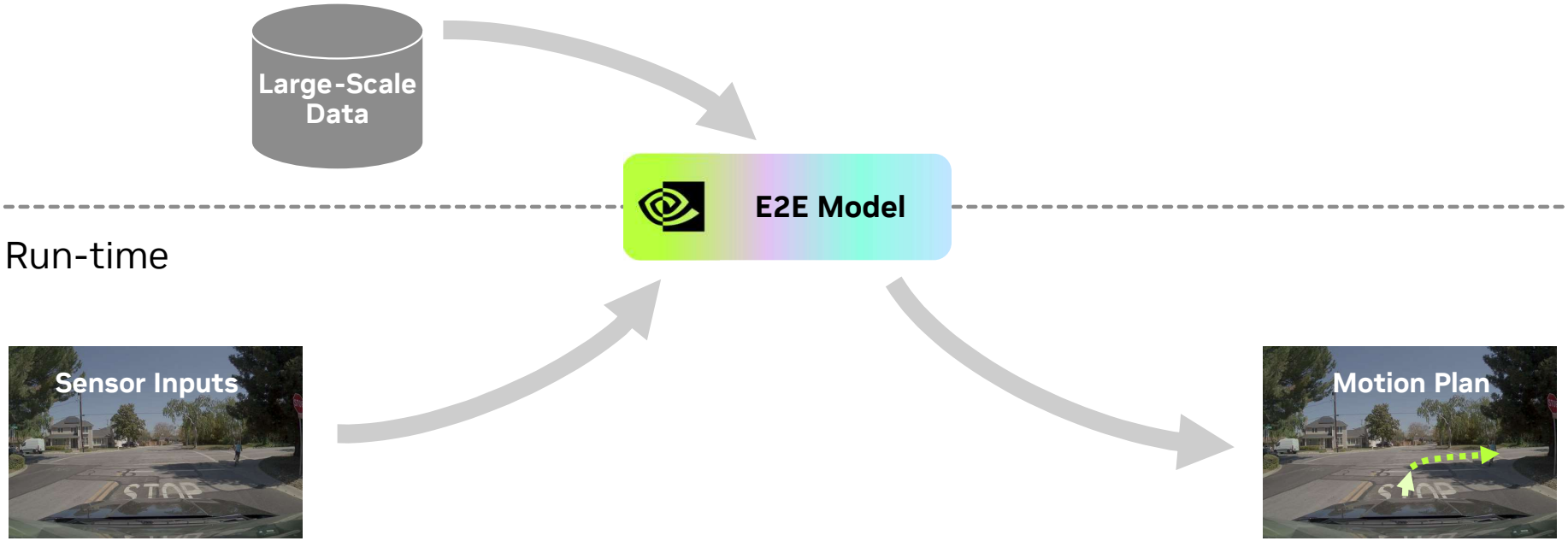
Research papers published on AV safety

30+

Certificates and assessment reports issued

Synoptic View of NVIDIA Halos AV Safety Day

Design-time



Synoptic View of NVIDIA Halos AV Safety Day

Design-time

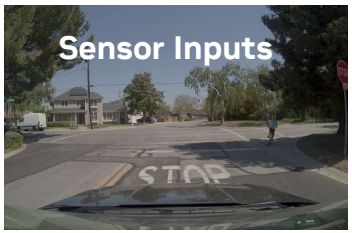
Large-Scale Data

Design-time safety (Chapter 2):
- Safety architecture
- AI train-time safety
- AV platform safety
- Data flywheel and processes

E2E Model

Validation-time guardrails (Chapter 4):
- Metrics
- Coverage – top-down
- Coverage – bottom-up
- Data flywheel and processes

Run-time



Deployment-time guardrails (Chapter 3):
- Run-time monitoring – HW
- Run-time monitoring – SW
- Arbitration
- Data flywheel and processes

Safety Regulation and Standardization in the Era of AI-Based AV (Chapter 5):
- Standardization challenges
- Regulatory challenges
- NVIDIA AI Systems Inspection Lab

From AVs to general Physical AI (Chapter 6):
- How Halos extends to Physical AI
- NVIDIA IGX elements
- Outside-in safety

Synoptic View of NVIDIA Halos AV Safety Day

Design-time

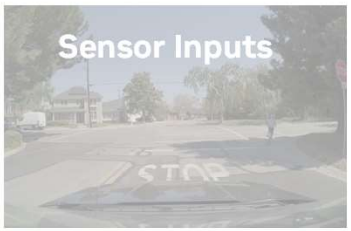
Design-time safety (Chapter 2):
- Safety architecture
- AI train-time safety
- AV platform safety
- Data flywheel and processes

Large-Scale Data

E2E Model

Validation-time guardrails (Chapter 4):
- Metrics
- Coverage – top-down
- Coverage – bottom-up
- Data flywheel and processes

Run-time



Deployment-time guardrails (Chapter 3):
- Run-time monitoring – HW
- Run-time monitoring – SW
- Arbitration
- Data flywheel and processes

Safety Regulation and Standardization in the Era of AI-Based AV (Chapter 5):
- Standardization challenges
- Regulatory challenges
- NVIDIA AI Systems Inspection Lab

From AVs to general Physical AI (Chapter 6):
- How Halos extends to Physical AI
- NVIDIA IGX elements
- Outside-in safety

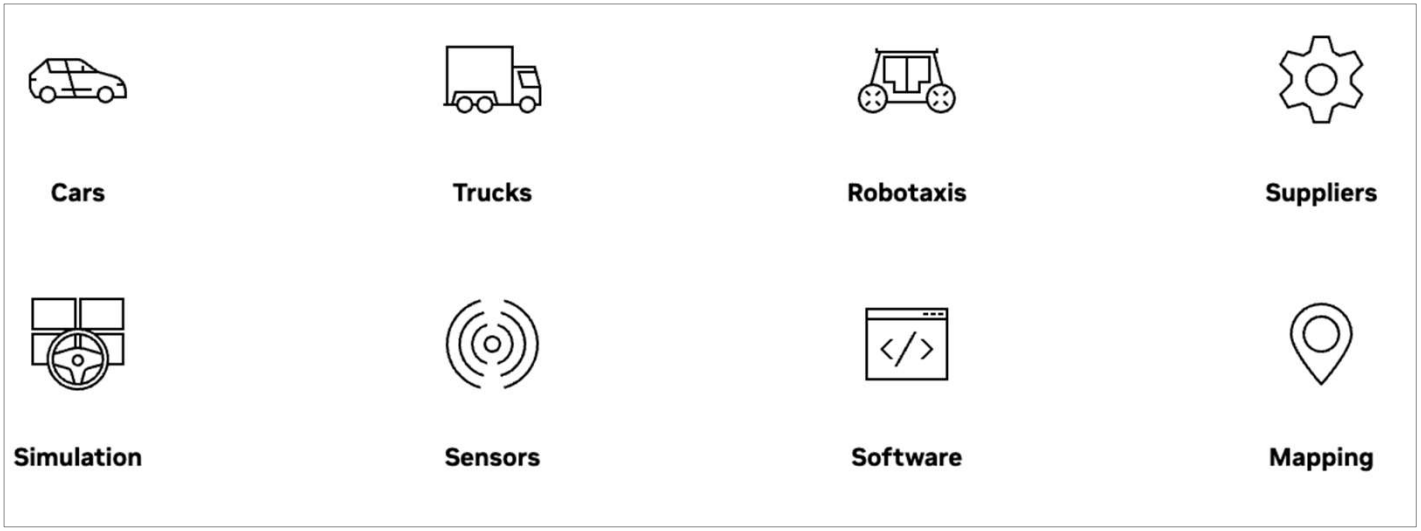
Halos Elements

Full-Stack System for Autonomous Vehicle Safety

HW/SW and Platform Safety	Algorithmic Safety	Ecosystem Safety	AI Systems Inspection Lab
• Safety assessed HW (SoC and reference board) • Safety certified DriveOS • Safety assessed base platform • NVIDIA DRIVE AGX Hyperion™ • DriveOS Linux for Safety *(future offering)*	• Libraries for safety data loading and accelerators • API for safety data creation, curation, reconstruction • NVIDIA Omniverse™ and Cosmos for AV Simulation Blueprint to train, test, and validate AVs • Diverse AV stack that combines a modular stack and E2E AI models	• Safety data with diverse, unbiased data • Continual improvements through a safety data flywheel	• Leadership in AV safety standardization and regulation • First of its kind to be accredited by ANAB, Inspects and verifies the integration of partners' products with Halos' safety elements

Partners Using Halos System

All NVIDIA DRIVE Partner Ecosystem Members are Part of Halos



Leading robotaxi companies, OEMs, industry safety pioneers, mapping and simulation companies, and software and sensor providers worldwide are using Halos systems to deliver autonomous vehicle safety at all levels of automation

View All NVIDIA DRIVE Partner Ecosystem Members

Chapter 2a: Design-Time Safety

Safety Architecture, AI Train-Time Safety, and Data Flywheel & Processes

Jonas Nilsson, Director of Safety Engineering, NVIDIA

Synoptic View of NVIDIA Halos AV Safety Day

Design-time

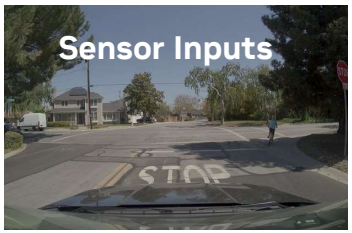
Design-time safety (Chapter 2):
- Safety architecture
- AI train-time safety
- AV platform safety
- Data flywheel and processes

Large-Scale Data

E2E

Validation-time guardrails (Chapter 4):
- Metrics
- Coverage – top-down
- Coverage – bottom-up
- Data flywheel and processes

Run-time



Deployment-time guardrails (Chapter 3):
- Run-time monitoring – HW
- Run-time monitoring – SW
- Arbitration
- Data flywheel and processes

Safety Regulation and Standardization in the Era of AI-Based AV (Chapter 5):
- Standardization challenges
- Regulatory challenges
- NVIDIA AI Systems Inspection Lab

From AVs to general Physical AI (Chapter 6):
- How Halos extends to Physical AI
- NVIDIA IGX elements
- Outside-in safety

Why use E2E in AVs?

User Experience Drives Adoption

Product Goals:

I. Human-like driving behavior in complex situations

What Does It Mean to be Safe?

Safe Driving Behavior is Necessary But Not Sufficient

Product Goals:

I. Human-like driving behavior in complex situations

II. Safe

Behave Safe

- See far enough
- Brake hard enough
- Correct decisions

Fail Safe

- Very few bugs
- Behave safely when something fails

Verified Safe

- Testable system (In finite time)

Can E2E be Safe Without Guardrails?

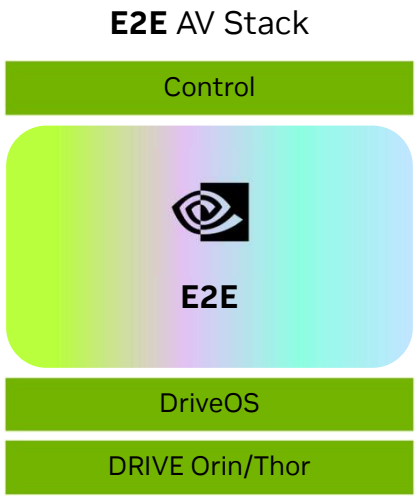
The Key Challenges For a Naïve E2E System

Fail Safe:

- No monitor or backup if something goes wrong

Verified Safe:

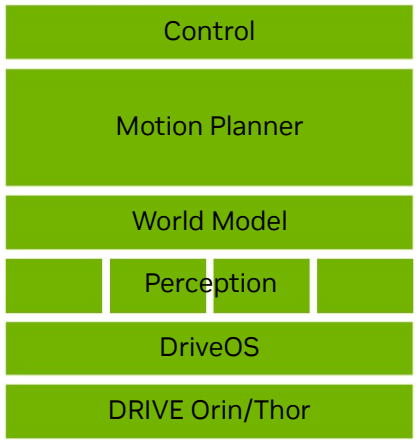
- Not possible to decompose safety requirements
- Sufficient testing only with:
  - massive mileage
  - validated E2E simulation environment



Why are Modular Stacks used in Safety-critical Systems?

Diversity, Redundancy, Subsystem Testing

Modular AV Stack



Fail Safe:

- Diverse and redundant algorithms, DNNs, sensors

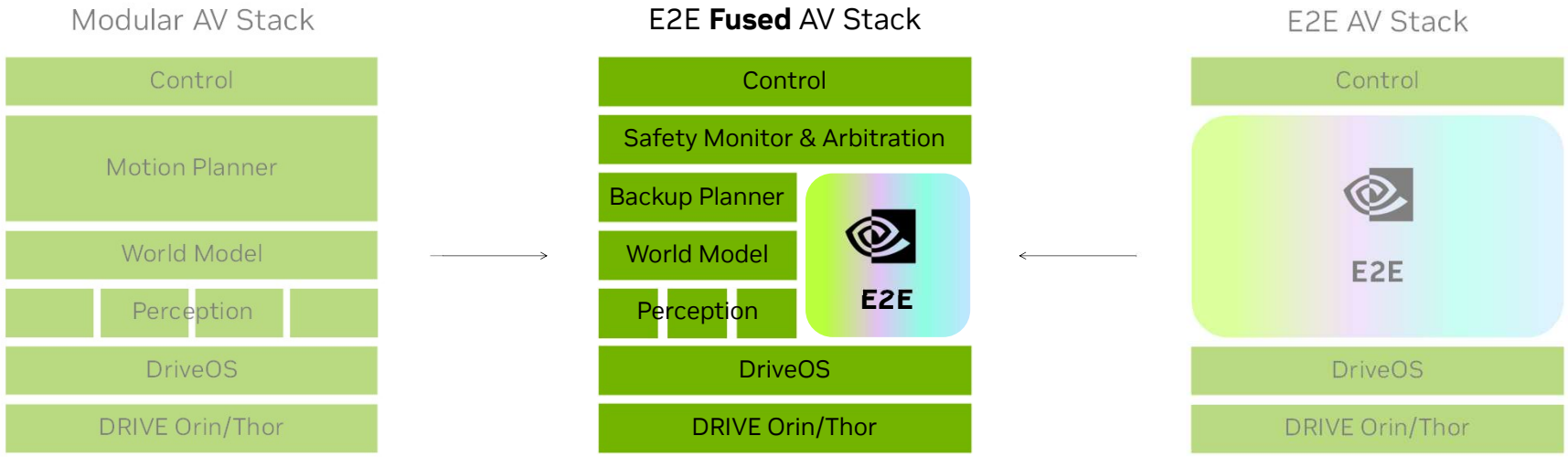
Verified Safe:

- Decompose safety requirements between subsystems
- Subsystem testing
  - Perception in open-loop
  - Motion planner with (simple) world model simulation

But, developing good driving behavior in complex situations is challenging compared to E2E approaches

Fusing E2E with a Modular Stack

Can We Bring the Best From Both Worlds?



Fail Safe:

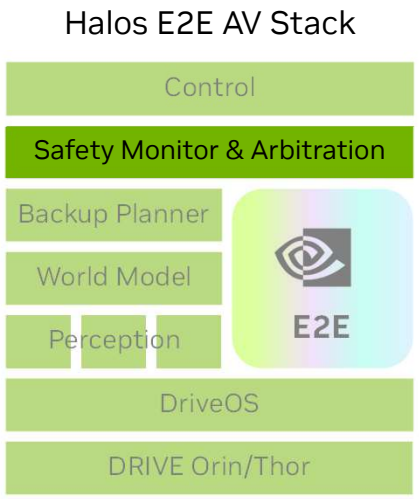
- Modular stack monitor E2E to prevent severe mistakes
- Modular stack acts as backup to bring vehicle to safe state

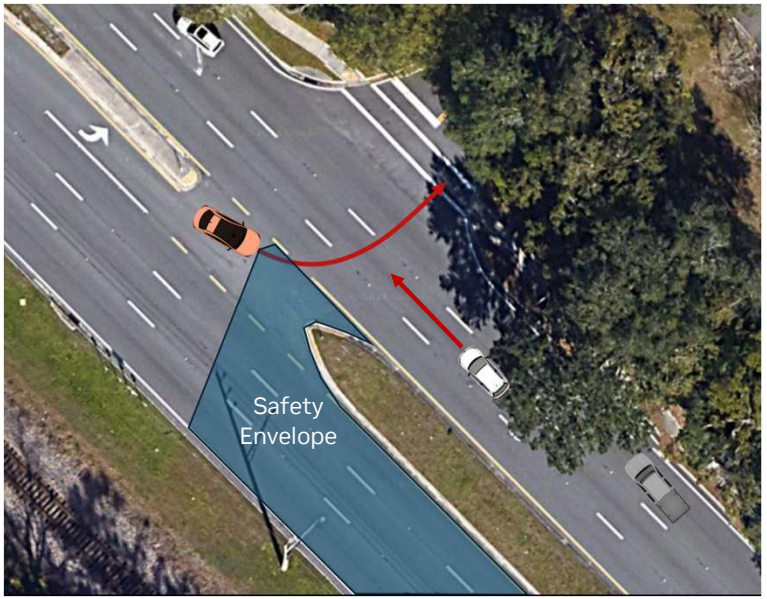
Verified Safe:

- Decompose safety requirements between modular and E2E stacks
- Subsystem testing improves test coverage

Safety Architecture for Fused E2E

Principles for Monitoring & Arbitration

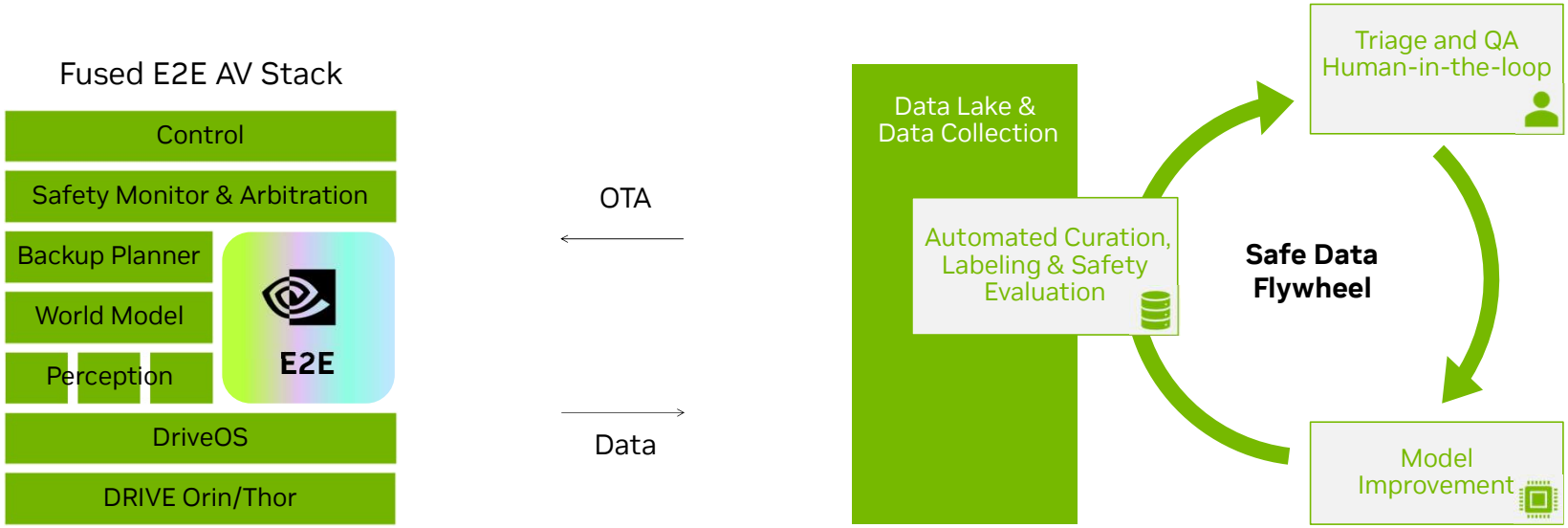




- E2E controls the vehicle within safety envelope
- Safety monitor
  - Checks E2E trajectory and prevents severe mistakes
  - Use diverse World Model as input
- More capable/safer E2E model ==> Larger safety envelope

NVIDIA Halos Two Pillars for Safe AV design

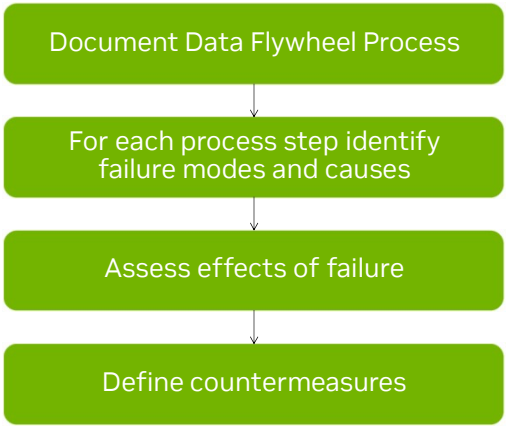
Safe Design is Broader Than the E2E AV stack

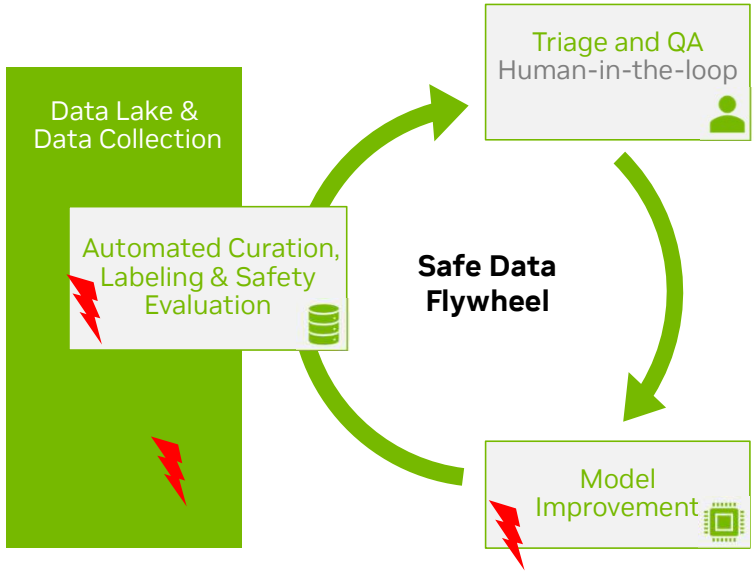


How Do We Build a Safe Data Flywheel?

Methodical Approach to Ensure and Monitor Safety

Data Flywheel Safety Analysis

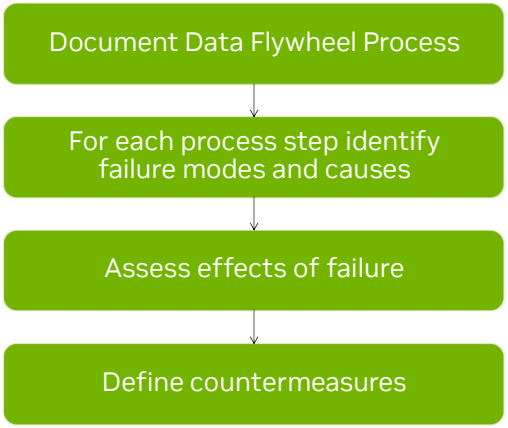


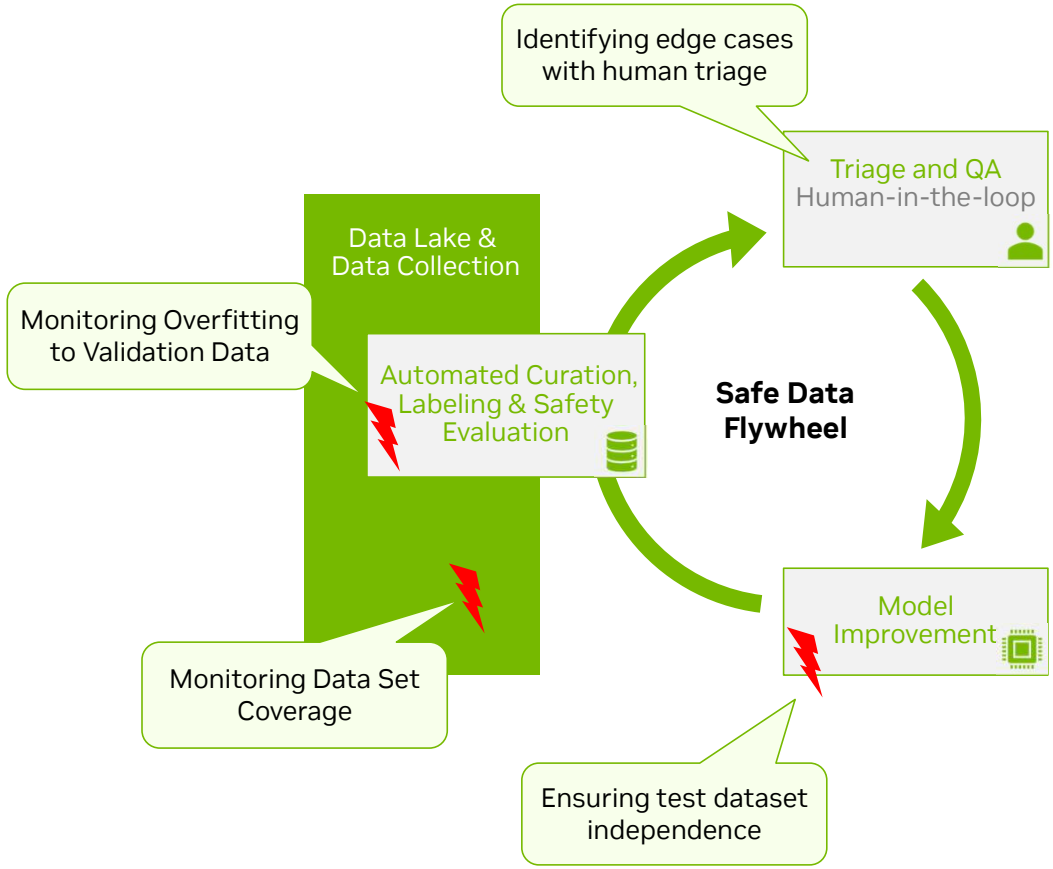


Countermeasure Highlights in the Safe Data Flywheel

Key is to Monitor Safety Continuously

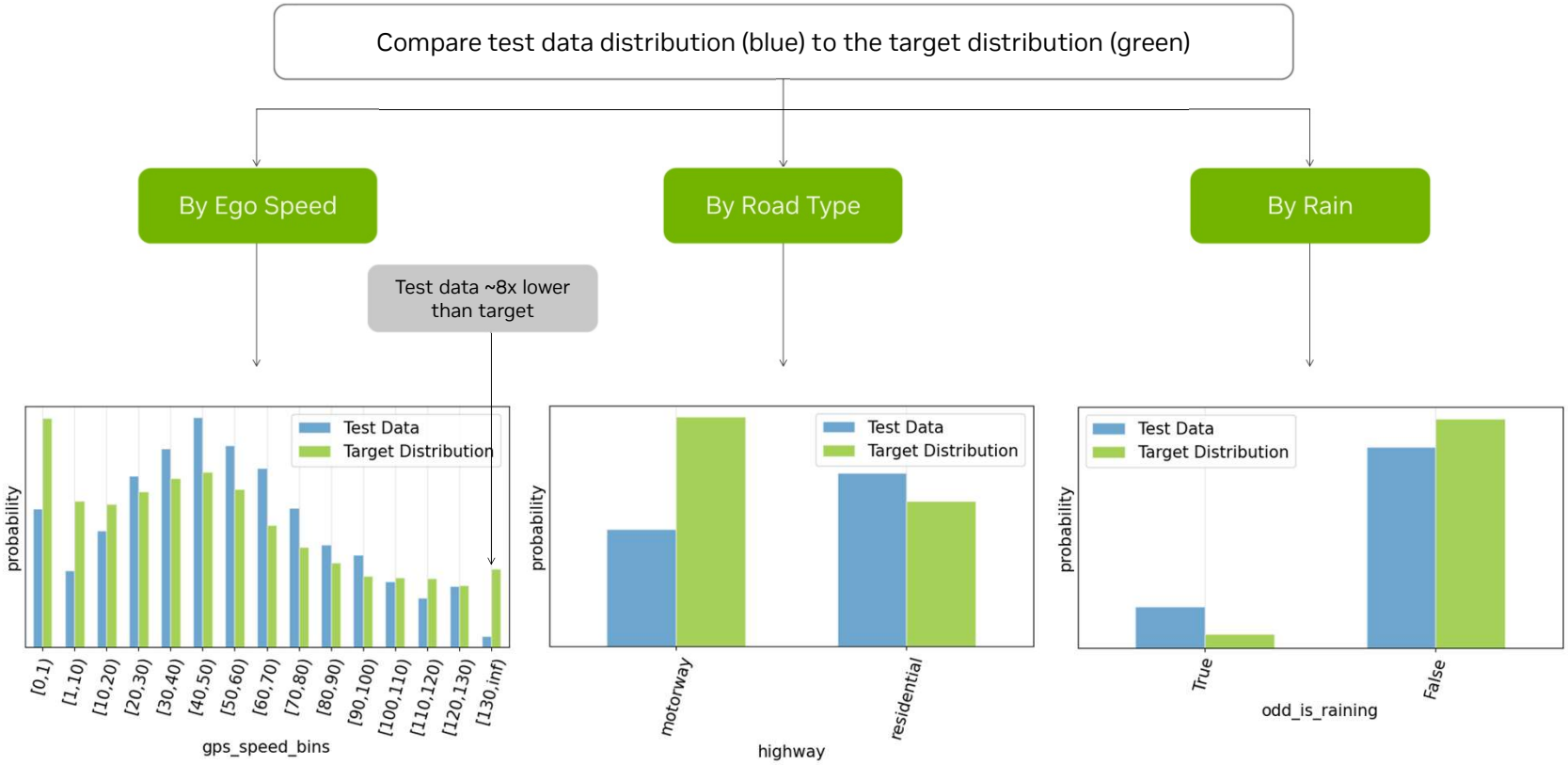
Data Flywheel Safety Analysis





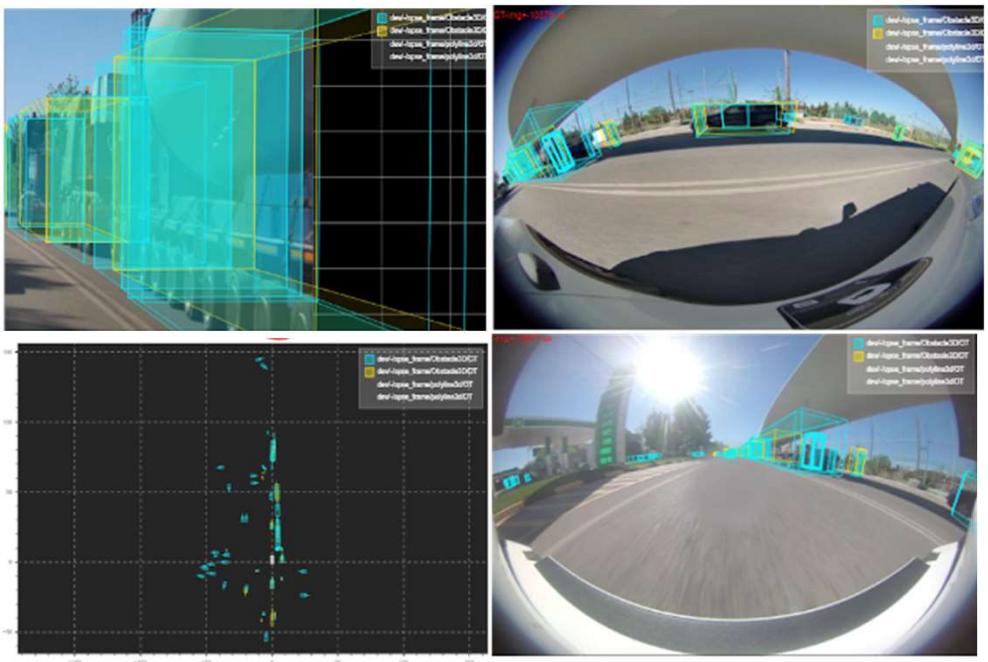
Highlight 1: Monitoring Data Set Coverage

Ensuring Test Data Distribution is Representative to Prevent Bias



Highlight 2: Identifying Edge Cases with Human Triage

By Analyzing “What-if” Scenarios From Automated Safety KPIs



- Truck transporting wind turbine blade
- Automated perception KPI flagged issue
- Human triage identified that area under blade not marked as hazardous to drive into

Very difficult to find by only looking at E2E behavior!

Highlight 3: Ensuring Test Dataset Independence

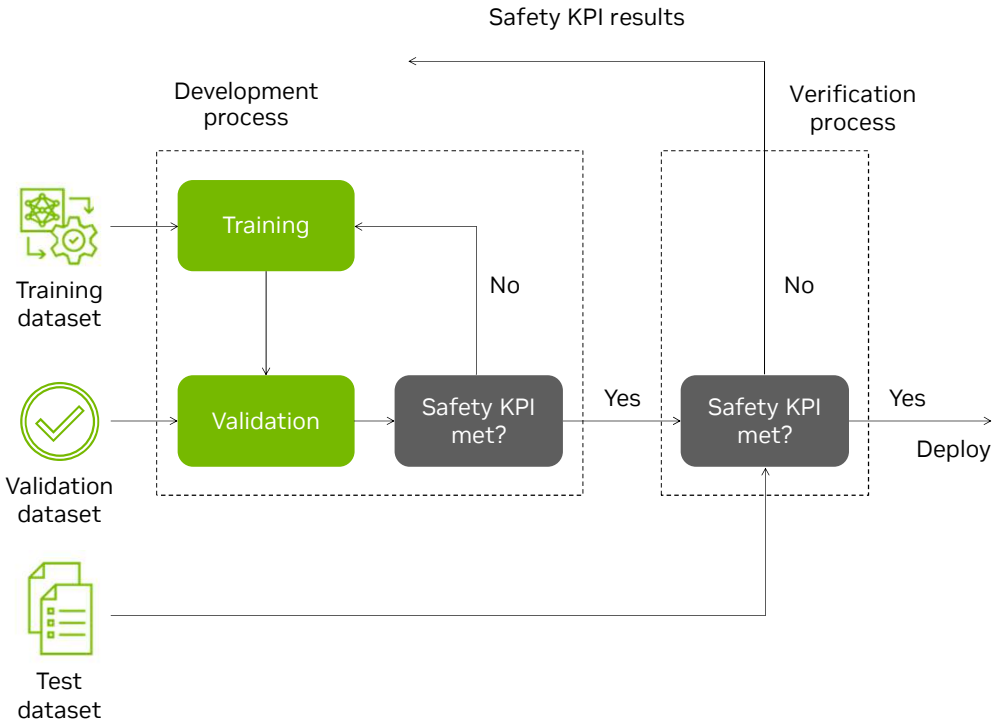
To Prevent Overfitting

Challenge: Overfitting to test data

- Weight overfitting: Training DNN models directly on test dataset.
- Hyperparameter overfitting: Repeatedly testing small hyperparameter changes against the test dataset.
- Dataset overfitting: Looking for challenging scenarios in test dataset, collecting similar scenarios in training datasets.

Countermeasures: Independent test data

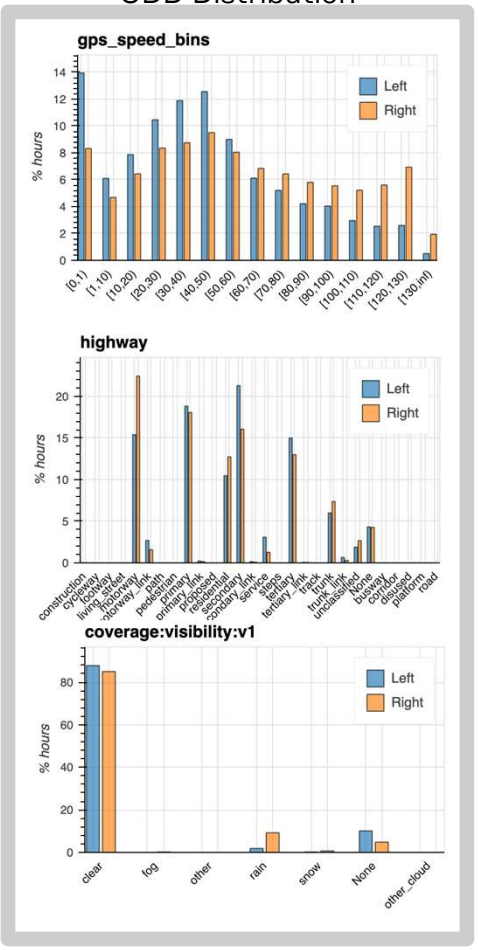
- Blind Safety test data: Raw data not viewable for SW development engineers (but KPIs shared)
- Continuous monitoring of overfitting



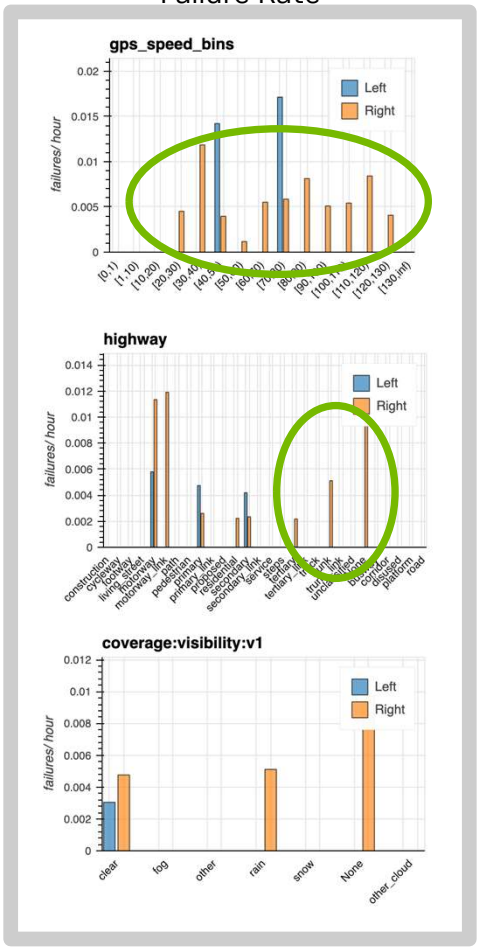
Highlight 4: Monitoring Overfitting to Validation Data

Open validation data vs Blind safety test data

ODD Distribution



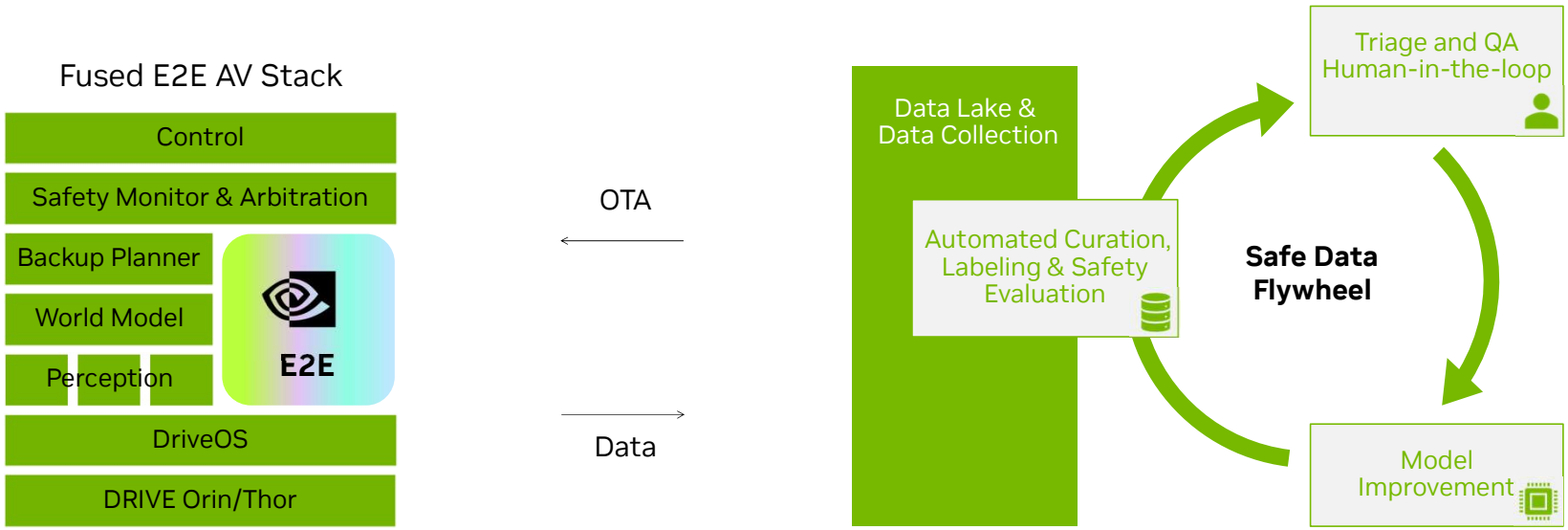
Failure Rate



- Data acquired per ODD category is similar
- Significantly higher failure rates in the Safety test dataset
- This Indicates:
  - Performance not fully generalizing to test dataset
  - Overfitting to the validation dataset

Key Takeaways Safe AV design

NVIDIA Halos Two Pillars for Safe AV design

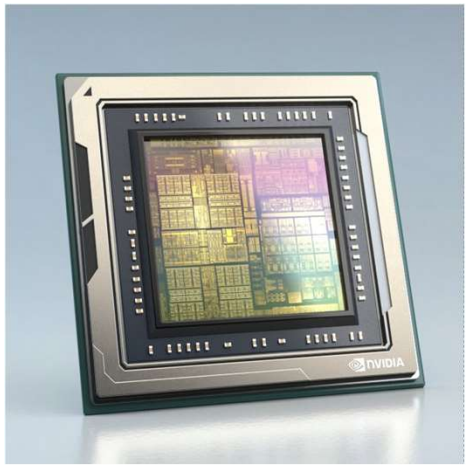


Chapter 2b: Design-Time Safety

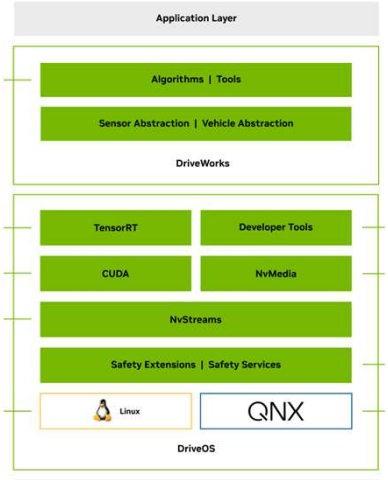
AV Platform Safety

Karl Greb, Sr. Director of Safety Engineering, NVIDIA

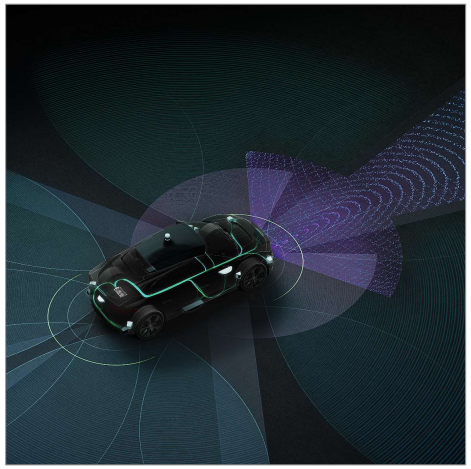
Safety Platform for E2E AV



HW Safety
SoC and Reference Board



SW Safety

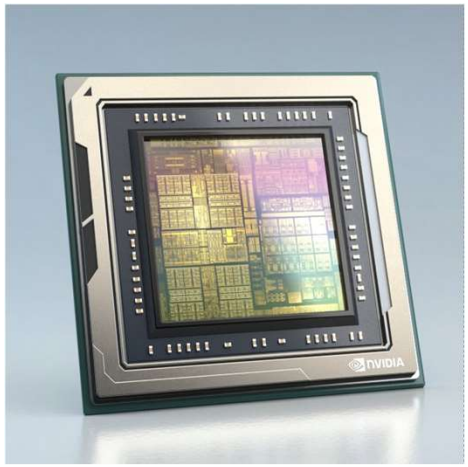


Platform Safety

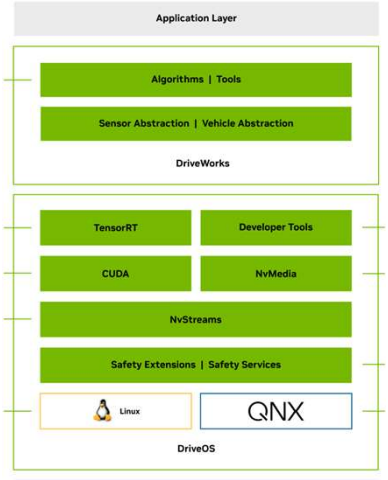
Main Challenges

HW Safety	SW Safety	Platform Safety
Complexity: billions of transistors Freedom from Interference Integrating safety IP from dozens of suppliers	Real-time scheduling AI support	Integrating multiple sensors Integration of complex HW and SW

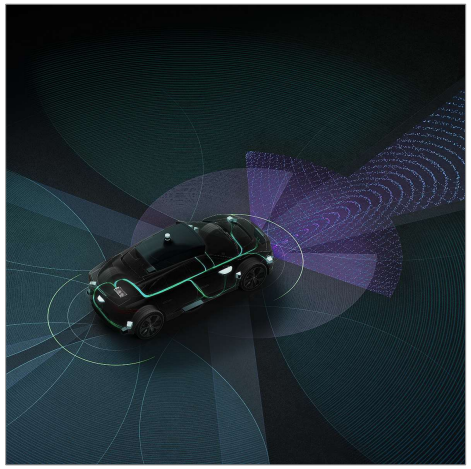
Key Pillars of NVIDIA Halos' Safety Platform



HW Safety
SoC and Reference Board



SW Safety



Platform Safety

Solutions

21B
Transistors Safety Assessed

ASIL D Systematics
>ASIL B Random Fault Metrics
Hardware Degradation Effects Detection

7M
Lines of Safety Assessed Code

TüV Safety Certified DRIVE OS

22K+
Platform Safety Monitors

TüV Safety Assessed base platform
Sensor Drivers & E2E Authentication

HW Safety - SoC

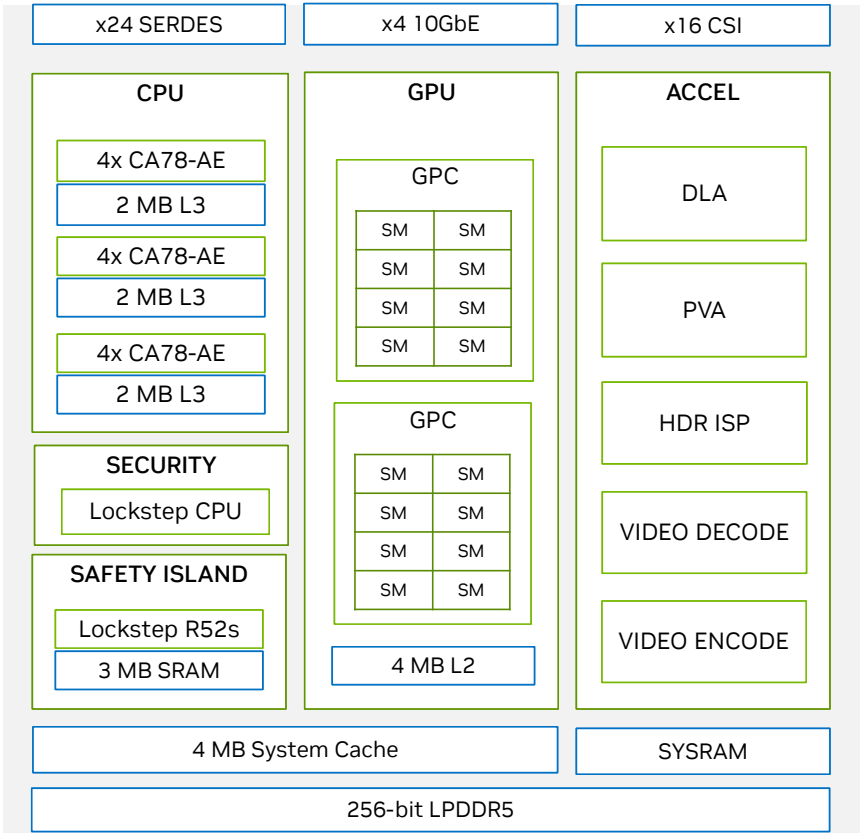
Designed to Ensure ASIL D for Systematics and ASIL B for Hardware Random Failures

Third-party Assessed
SoC process certified since 2018, multiple SoC SKUs passed product safety assessment by TUV SUD

ASIL D Systematics
All IPs supported for safety usage are developed with ASIL D capability

High Diagnostic Coverage
>97% overall coverage with >87% coverage on worst case IP provided by >22k safety mechanisms

Diversity & Redundancy
Multiple engines and interfaces can be paired for L3/L4 ASIL decomposition (e.g., GPU/CPU, GPU/DLA, GPU/PVA, CCPLEX CPU/FSI CPU, ...)



DRIVE AGX Orin

FFI and DFI Support
Rich features for coexistence of functionality and dependent failure initiator detection:
SMMU in CCPLEX and GPU, GFX execution watchdog and hardware Context Switch in GPU, NOC firewalls, clock/voltage/thermal monitors

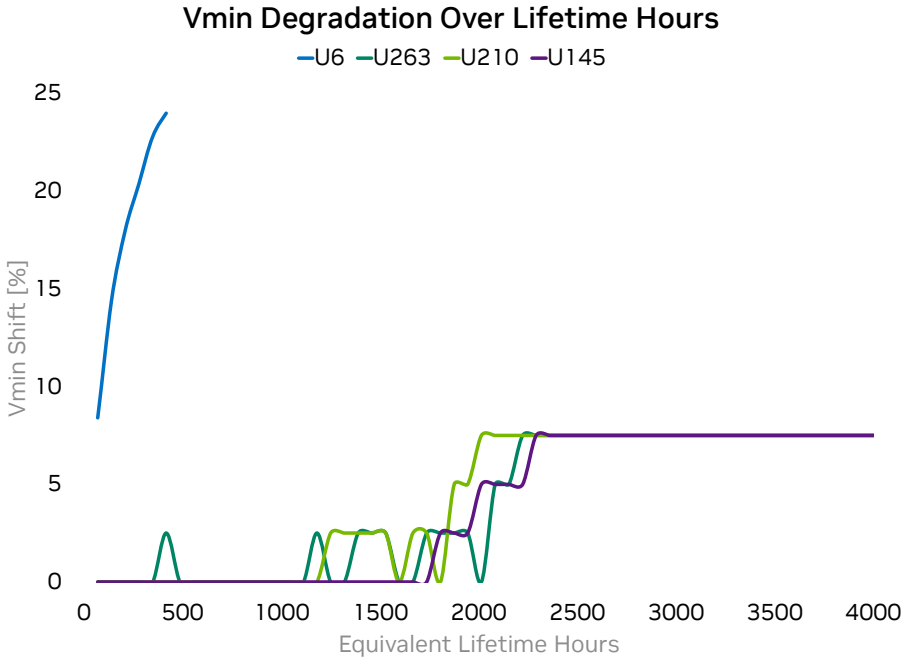
NVIDIA IST
Logic and Memory BIST on whole SoC for latent faults coverage
CCPLEX runtime IST
Predictive maintenance roadmap

Designed For AI Safety
Dedicated HW safety mechanisms for AI inferencing safety

ASIL D Safety Island
ASIL D Functional Safety Island with up to 10K DMIPs
Dedicated I/O, power, clocks

HW Safety – Addressing Degrading Faults

- NVIDIA found degrading failures in FinFET based silicon
  - We identified the root causes
  - We created Predictive In System Test as a mitigation
- Similar issues possible in any state-of-the-art technology – not a unique NVIDIA problem but no industry consensus on handling
- NVIDIA proposed and led development of ISO TR 9839 to drive an industry consensus
- Will be integrated and enhanced in ISO 26262 3rd edition
- For more information, refer to the GTC 2023 DRIVE Developer Day presentation “Enhancing Autonomous Vehicle Safety Using NVIDIA In System Test”

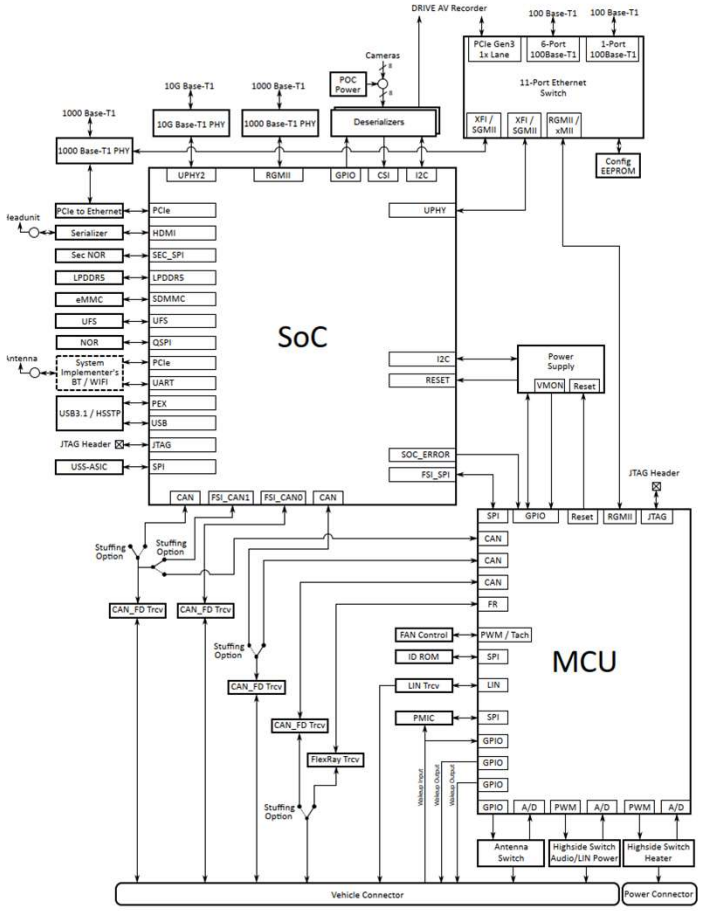


Abnormal and Normal Degradation
an NVIDIA experiment

HW Safety – Reference Board

Designed to Ensure ASIL D for Systematics and Hardware Random Failures

- Development process certified since 2019
- Assessed ISO 26262 ASIL D capable
- Delivered as a reference architecture and design for customer productization
- Customers can modify and extend the design as part of their productization and product safety assessment
- NVIDIA works with third parties to find the best supporting components and prequalifies alternative and second source devices



SW Safety - OS

DriveOS is safety certified up to ASIL D

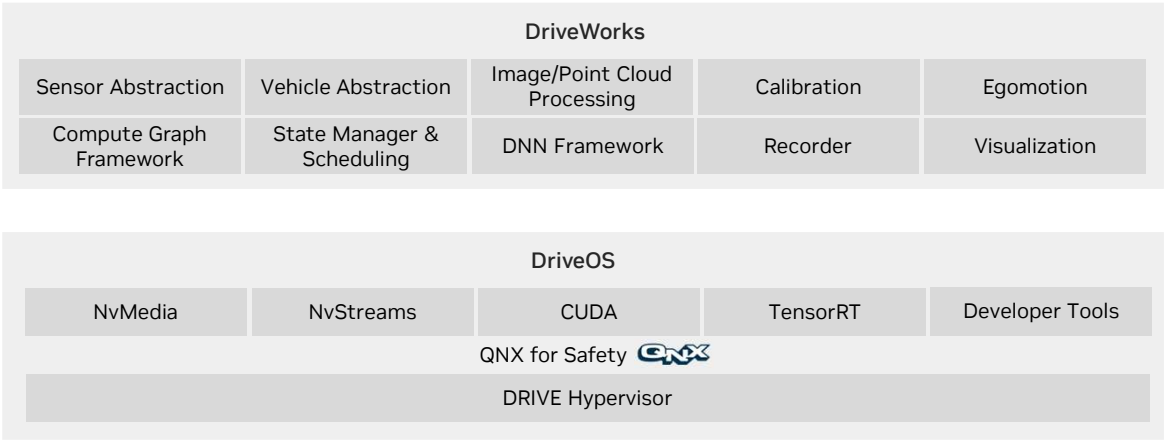
AI Ready SDK
Support for SOTA DNN models out of the box

Safety & Security Certified
ISO 26262 & ISO 21434

Rich Suite of Tools
Accelerating development

Programmability
Widely adopted APIs with large developer ecosystem

Cloud to Car
Same APIs and Frameworks work across the platforms



- Comprehensive **safety assessed acceleration libraries**
- Rich suite of **developer tools** & frameworks
- **Deterministic runtime scheduler** orchestrates execution of graphs across hardware engines

- **Safety certified OS** up to ASIL D
- Type 1 **hypervisor** guarantee safe partition
- Debug overlay environment
- Include required configuration (network, device tree) and tools for data collection and debugging

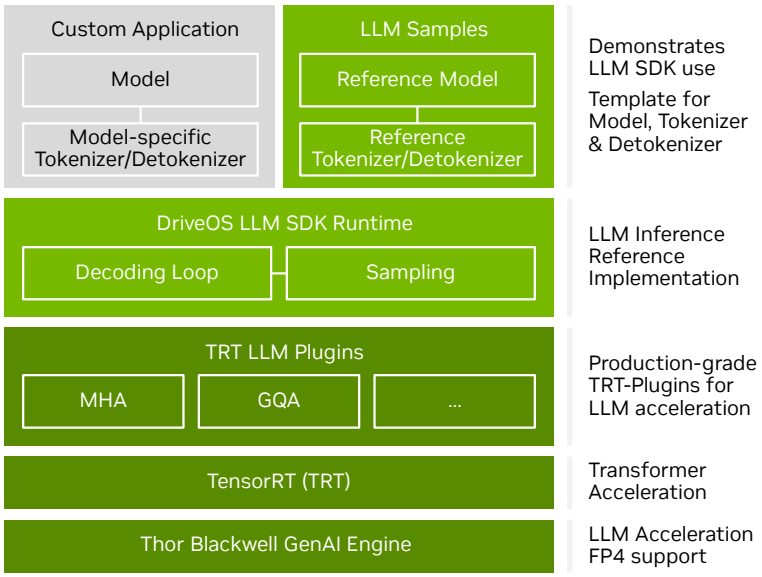
SW Safety - AI

LLM Acceleration on DRIVE AGX Thor

DRIVE AGX Thor platform will accelerate LLMs for production:

- Thor's GenAI Engine provides FP4 support with adaptive range for quantization and HW-level LLM acceleration
- TensorRT 10 comes with reduced memory consumption, faster build time, higher perf for transformers
- New DriveOS LLM SDK implements an optimized decoding loop with sampling for LLMs

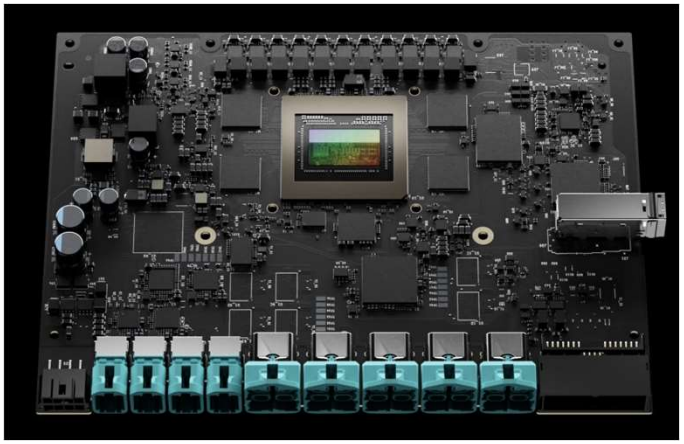
Developers can use the new DriveOS LLM Samples as a reference to get started with their own application



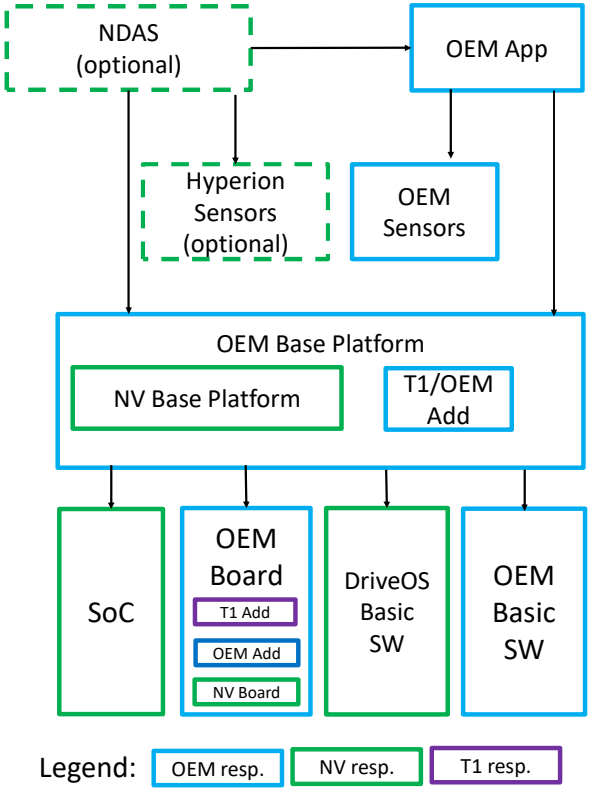
Platform Safety

Enabling End-to-End Safety Throughout the Product Stack

- The **Base Platform** delivers the foundational safe computer needed to enable safe systems, for all types of applications.



- **DRIVE Hyperion** delivers a library of AV sensors pre-qualified to work with NVIDIA's safety platforms.



Ecosystem Enablement and Customer Stories

Karl Greb

Halos Elements

Full-Stack System for Autonomous Vehicle Safety

HW/SW and Platform Safety	Algorithmic Safety	Ecosystem Safety	AI Systems Inspection Lab
• Safety assessed HW (SoC and reference board) • Safety certified DriveOS • Safety assessed base platform • NVIDIA DRIVE AGX Hyperion™ • DriveOS Linux for Safety *(future offering)*	• Libraries for safety data loading and accelerators • API for safety data creation, curation, reconstruction • NVIDIA Omniverse™ and Cosmos for AV Simulation Blueprint to train, test, and validate AVs • Diverse AV stack that combines a modular stack and E2E AI models	• Safety data with diverse, unbiased data • Continual improvements through a safety data flywheel	• Leadership in AV safety standardization and regulation • First of its kind to be accredited by ANAB, Inspects and verifies the integration of partners' products with Halos' safety elements

NVIDIA DRIVE AGX
NVIDIA DRIVE AV
NVIDIA OMNIVERSE

MERCEDES-BENZ

Every next-generation of Mercedes-Benz vehicle will include this first-of-its-kind software-defined computing architecture that includes the most powerful computer, system software and applications for consumers, marking the turning point of traditional vehicles becoming high-performance, updateable computing devices.

NVIDIA DRIVE AGX
NVIDIA DRIVE AV
NVIDIA OMNIVERSE

JAGUAR LAND ROVER

All-new Range Rover, Defender, Discovery, and Jaguar vehicles will be built on the NVIDIA DRIVE® AI-defined platform—from the cloud, built on NVIDIA data center solutions, to the car, powered by NVIDIA DRIVE Orin™. Range Rover, Defender, Discovery, and Jaguar vehicles will continuously improve through over-the-air updates, delighting customers with AI-based safety systems, automated driving capabilities, and digital services.

Thanks !

Scan the QR code below to visit NVIDIA Halos website and Contact Us to join the Halos community, or be informed on our future initiatives and get involved!

or contact:

mpavone@nvidia.com
rmariani@nvidia.com