April, 2018

Nambi G Ganesh
Global Software Chief Engineer

Addressing Complexity in Automotive Software using Model-Based Design

APTIV

About Aptiv

Aptiv Provides End-To-End Solutions That Allow Us To Commercialize New Mobility

Aptiv Brief Intro

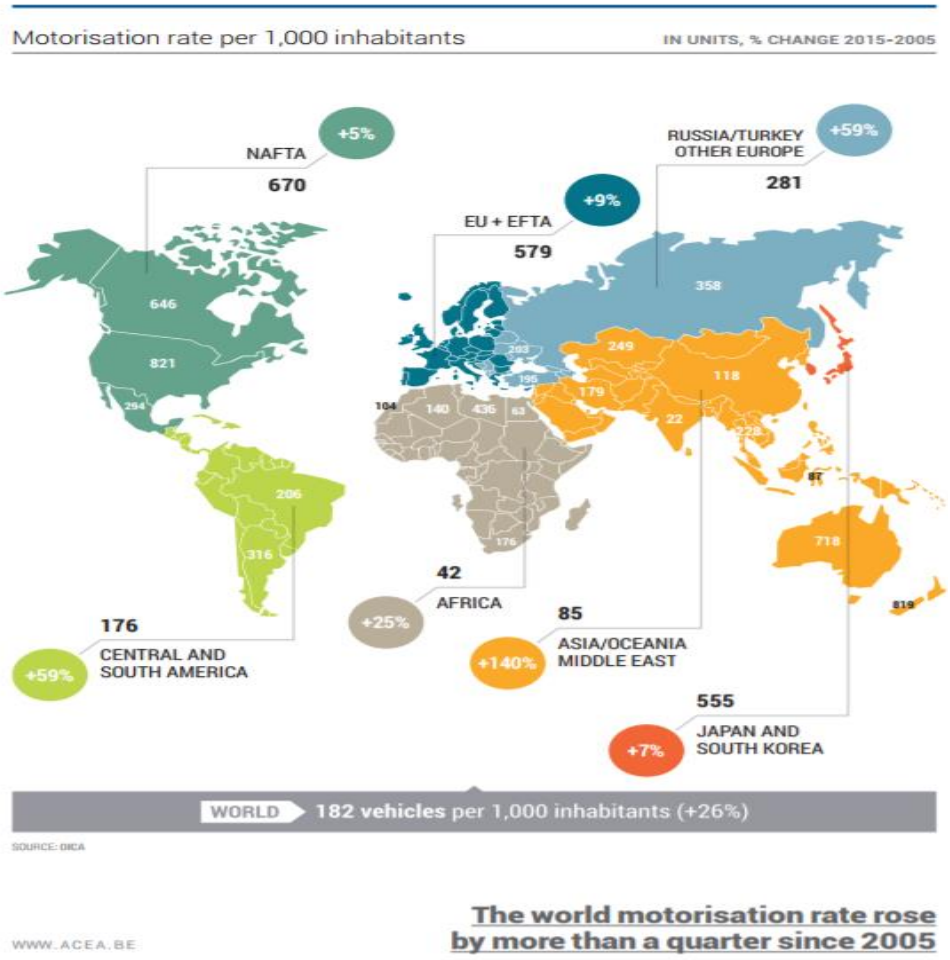
Aptiv PLC to Transform Future Mobility

Formerly known as Delphi Automotive, Aptiv emerges from the completion of Delphi's spin-off of its Powertrain segment. Aptiv brings unparalleled capabilities in solving the complex challenges associated with safer, greener and more connected transportation. At the core of this capability is the software and vehicle architecture expertise that enables the advanced safety, automated driving, user experience, and connected services that are making the future of mobility work.

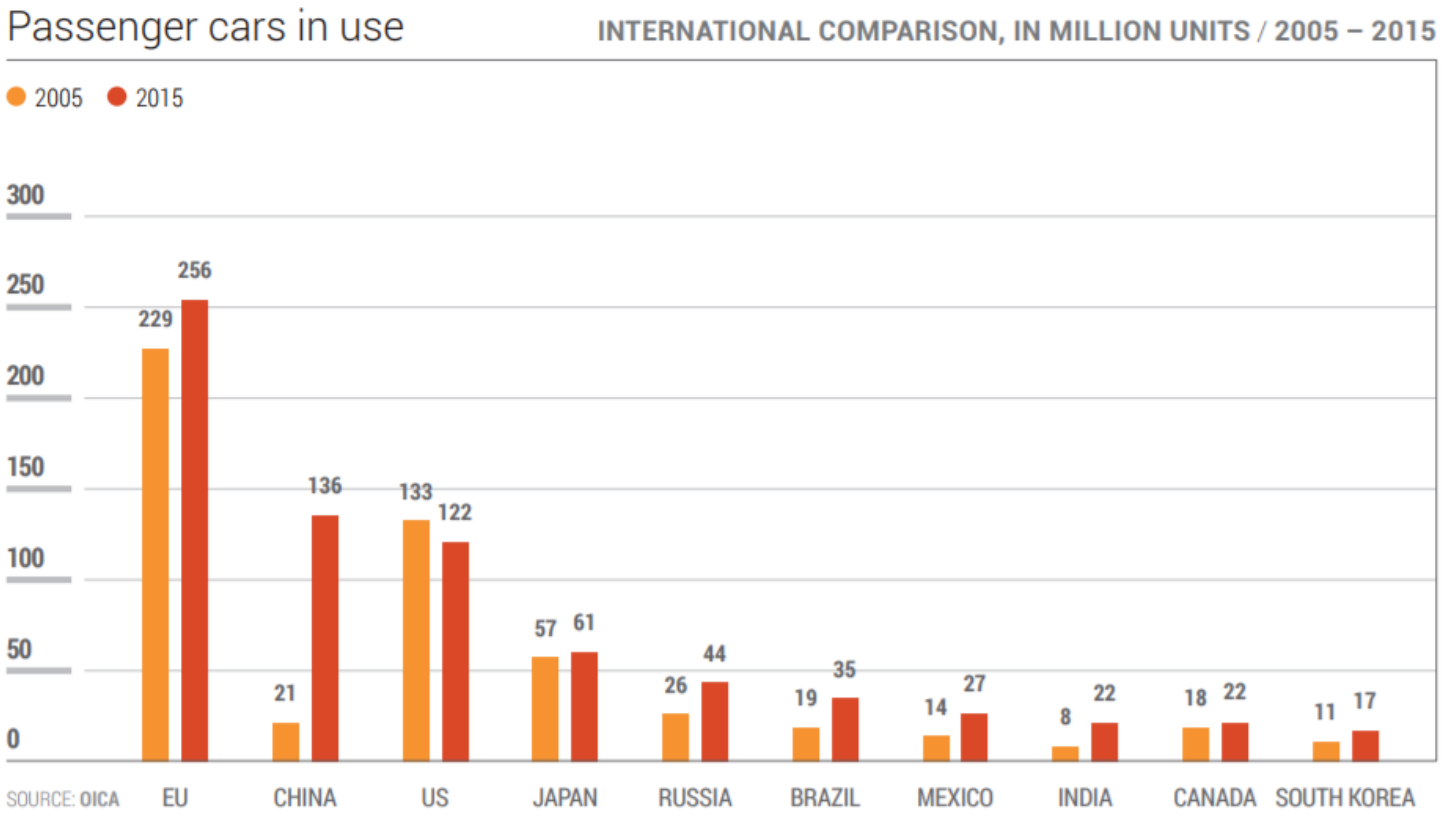
Automotive Trend and Software Complexity

Automotive Trends

- Vehicle fleet continues to grow



The world motorisation rate rose by more than a quarter since 2005



In the Next 10 years …..

In the next 10 years Globally:

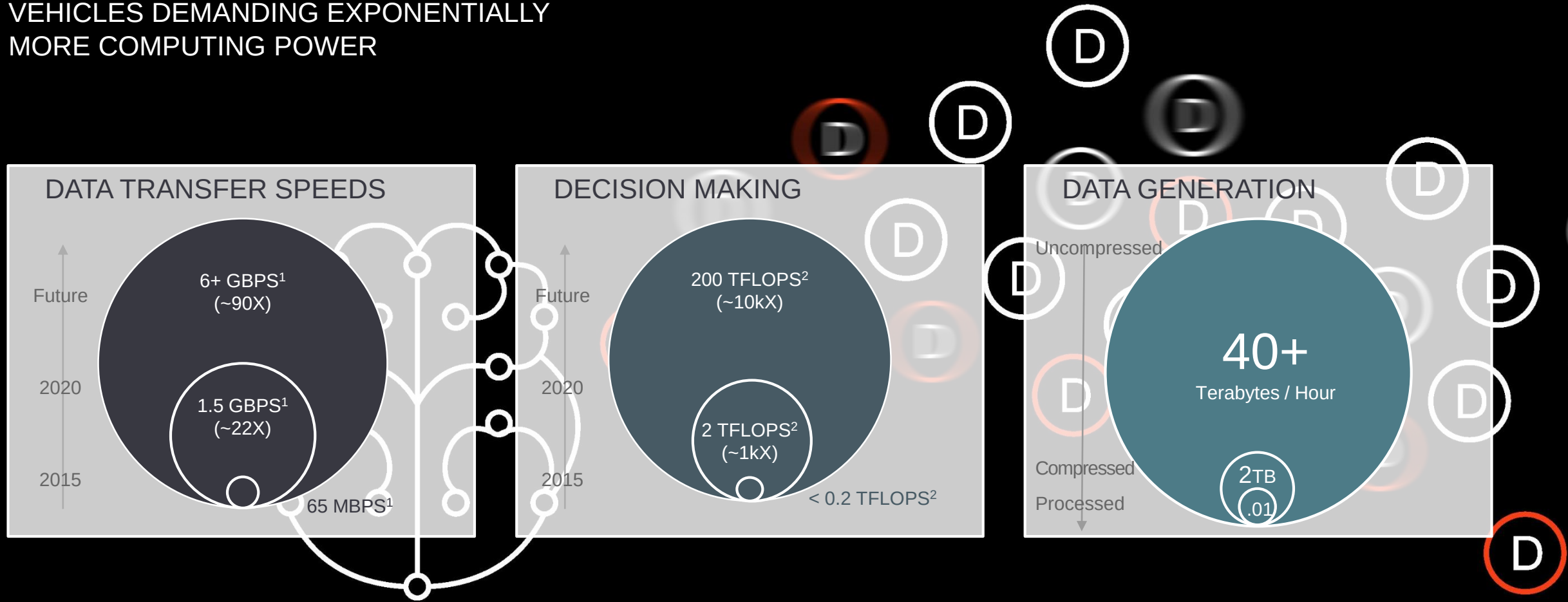
- 50% more vehicles on the road
- Stricter fuel economy regulations
- Automated driving reality

In the next 10 years India

- India to be #3 in global passenger vehicle
- India's share to touch 8 per cent from current 4
- Automotive Hub for Small Vehicle

Sensing & Computing Capabilities

VEHICLES DEMANDING EXPONENTIALLY
MORE COMPUTING POWER

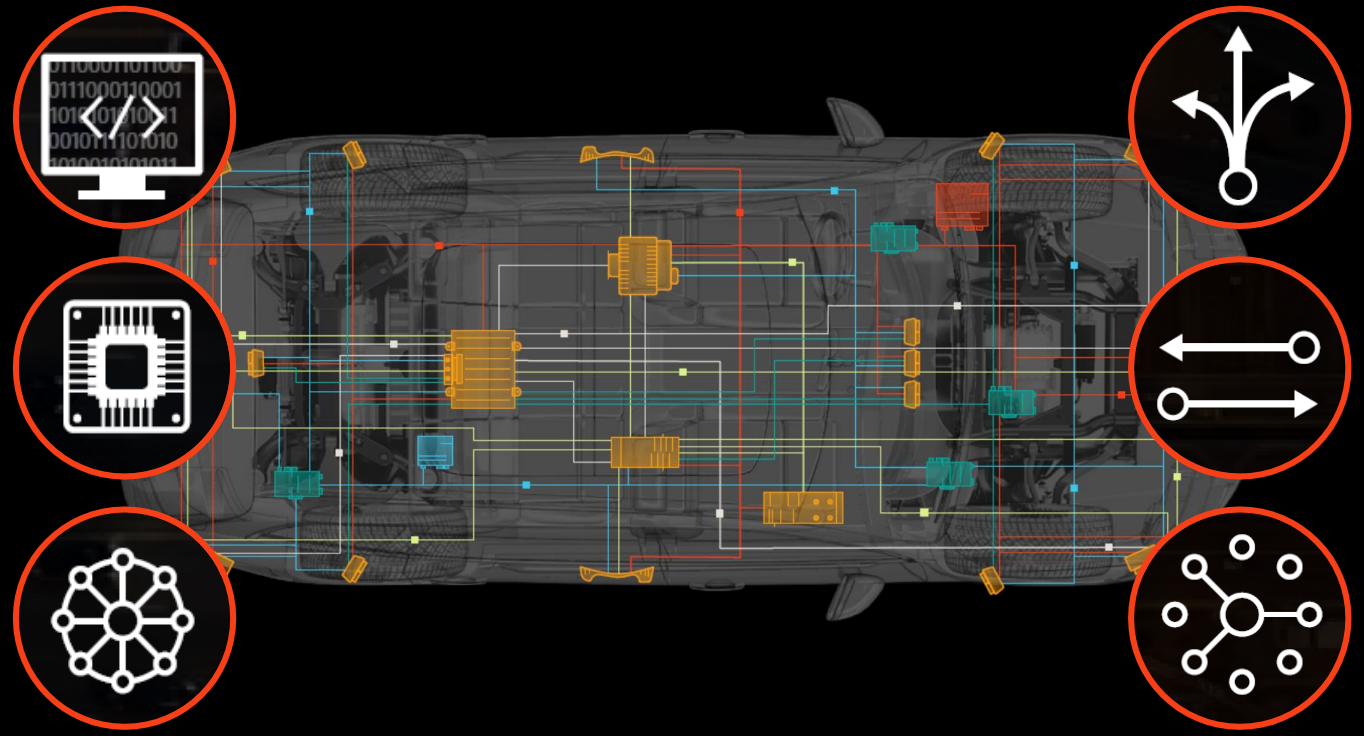


Smart Vehicle Architecture - Key Enabler

SOFTWARE DESIGN

ADVANCED HARDWARE

EFFICIENT SIGNAL AND POWER DISTRIBUTION



FLEXIBILITY
Software framework that supports dynamic feature sets and compute needs

LIFECYCLE
Decoupling hardware & software with fully abstracted approach allows independent lifecycles

RESILIENCE
Addresses multilayer system fault tolerances; meets redundancy requirements

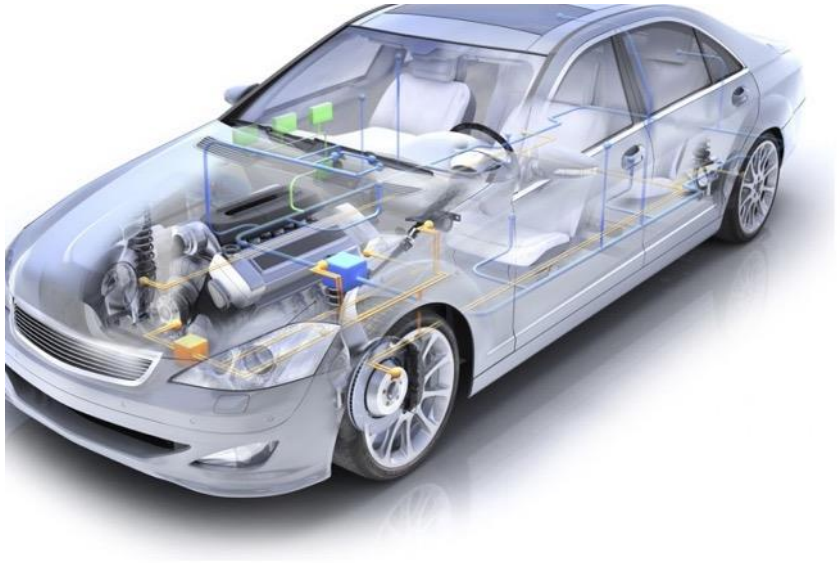
Order of Magnitude and More Software

From traditional to connected to autonomous

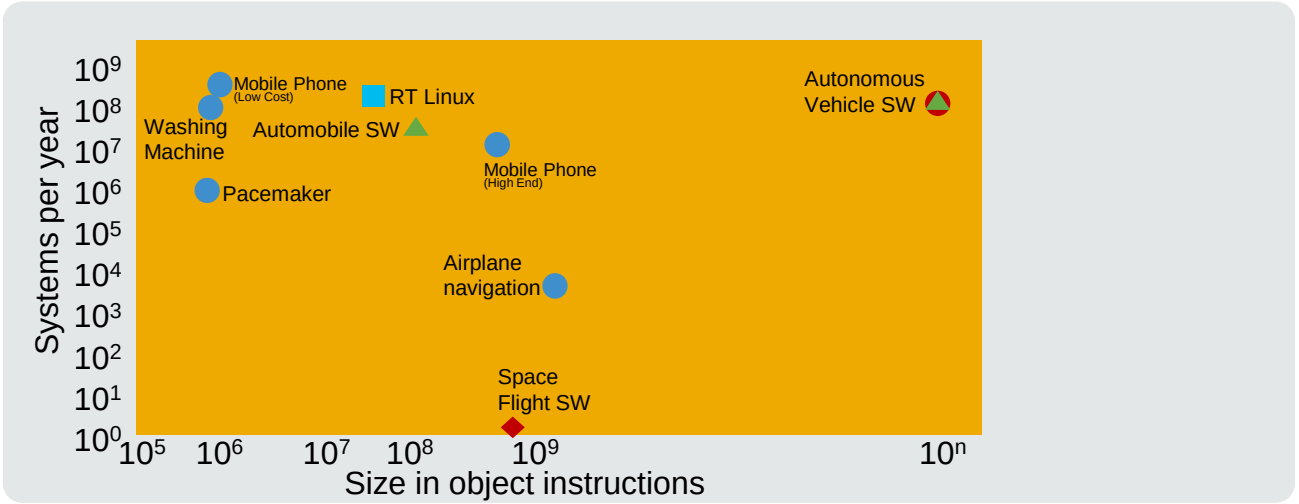
Cybersecurity

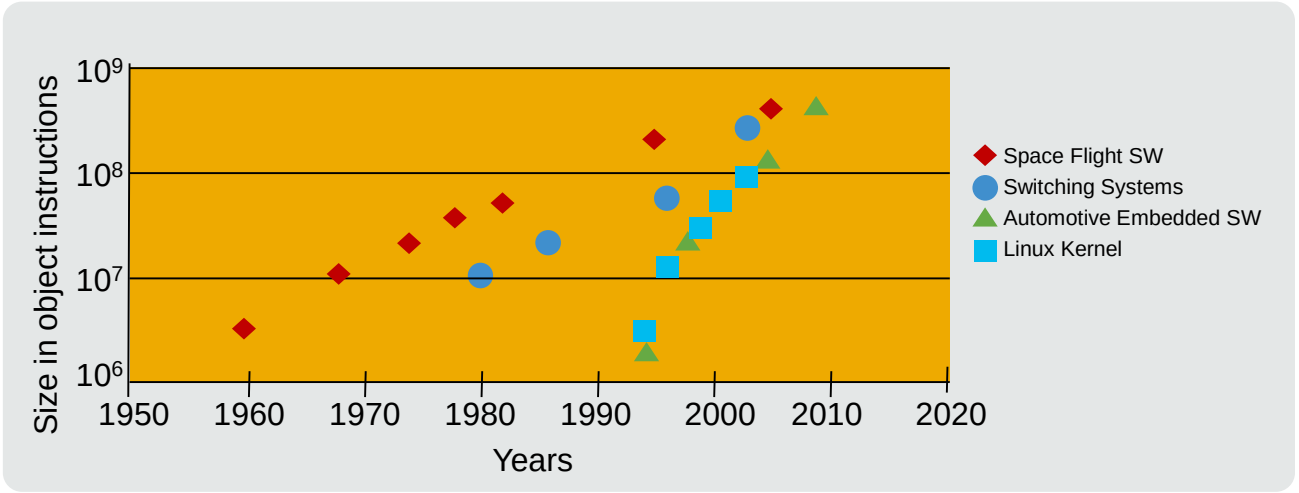
Safety Critical (Functional Safety)

Highly Integrated, Internal and External



Today 100 ECUs and 100 MLOC





Source: Ebert IEEE Computing

Safety: SAE Automation Levels

SOCIETY OF AUTOMOTIVE ENGINEERS (SAE) AUTOMATION LEVELS

Full Automation

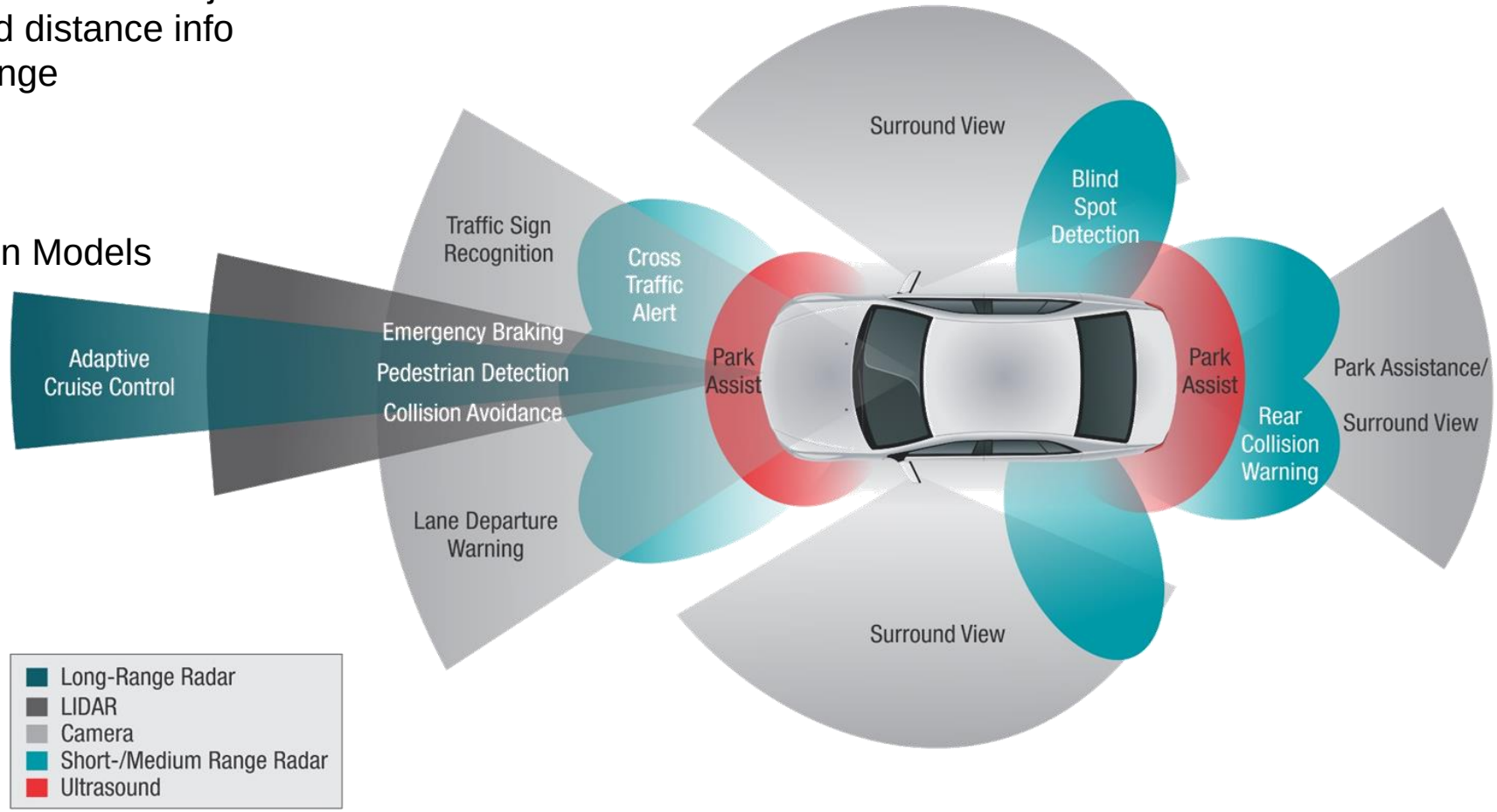
0	1	2	3	4	5
No Automation	**Driver Assistance**	**Partial Automation**	**Conditional Automation**	**High Automation**	**Full Automation**
Zero autonomy; the driver performs all driving tasks.	Vehicle is controlled by the driver, but some driving assist features may be included in the vehicle design.	Vehicle has combined automated functions, like acceleration and steering, but the driver must remain engaged with the driving task and monitor the environment at all times.	Driver is a necessity, but is not required to monitor the environment. The driver must be ready to take control of the vehicle at all times with notice.	The vehicle is capable of performing all driving functions under certain conditions. The driver may have the option to control the vehicle.	The vehicle is capable of performing all driving functions under all conditions. The driver may have the option to control the vehicle.

Safety: Sensors and Features

- Sensors
  - Multiple Radars supporting different Ranges that isn't impacted by weather
  - Multiple Vision (cameras) that classifies objects
  - LiDAR that provides range and distance info
  - Ultrasond sensors for short range

- Processing
  - Tracker, Fusion and Perception Models
  - Greater computing power
  - High-speed data transmission

- Standardized development
  - AUTOSAR
  - ISO 26262 Functional Safety



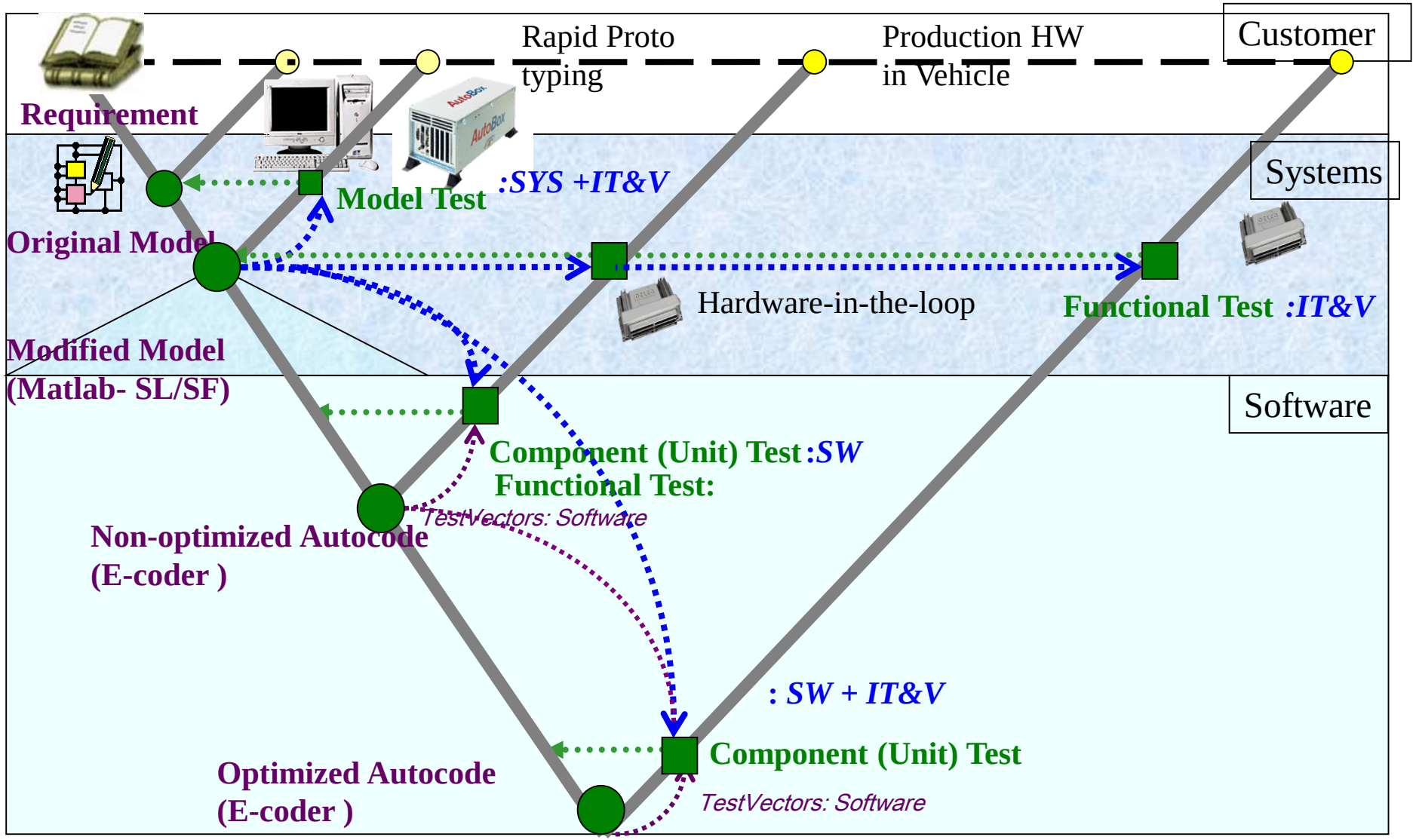
APTIV

How do we address it?

How do we handle Such Massive Scenarios?

- **Simulation, Automated Development and Testing Tools are becoming more Necessary then ever before**
- **Agile and Continuous Integration – To facilitate Predictable Development**
- **Autosar Based Development - Layered SW Architecture provides Scalability and Flexibility**
- **Compliance with FS and CS (Safety and Security Standards)**
- **Tools and Test Systems**
  - **Demand increasing for Tools and Test Systems to Simulate, Test, Log / Record and Replay time synchronized data**
    - Control system / algorithm in a Micro Controllers
    - Multi Core Microcontrollers in a single ECU
    - Multiple Micro Controllers (With Multi Core Architecture) in a Single ECU

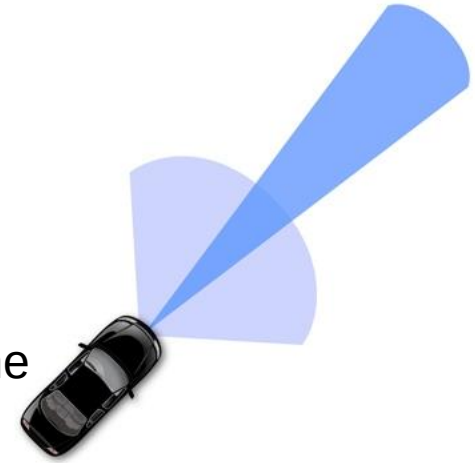
Software Development and Tools



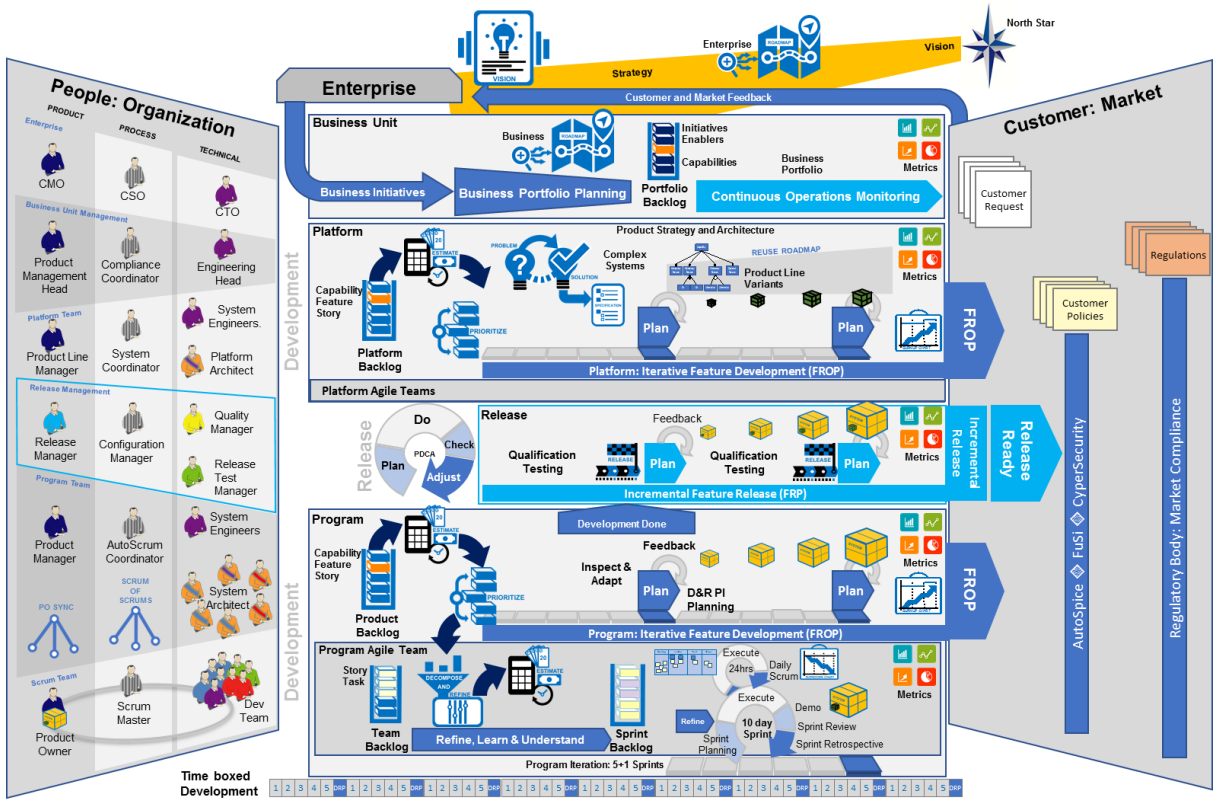
Simulation, Automated Development & Testing Tools are becoming more Necessary then ever before.

Example: Algo Development in Matlab and C-Code Generation

- Automotive radar is an enabling technology for collision mitigation, blind-spot alerts, adaptive cruise control, and many other active safety features.
- Aptiv used Matlab® and Matlab Coder™ to accelerate the design, simulation, and implementation of a production radar sensor alignment algorithm.
- Matlab was used to **analyze** recorded sensor data from road testing a real vehicle.
- Powerful Matlab built-in functions used to **realize** and simulate with huge amounts of vehicle data to **verify** the accuracy of the sensor misalignment angle calculated by the algorithm.
- Used Matlab Coder to **generate** production C code.
  - Generated C code was efficient and also easy to integrate.
- **Verified** C code (in PC environment) by calling a MEX function within Matlab and comparing the results of the generated code with the results of the original Matlab algorithm.
  - Reduced development
  - Reduced turn around time. Algorithm changes easily verified and coded quickly.



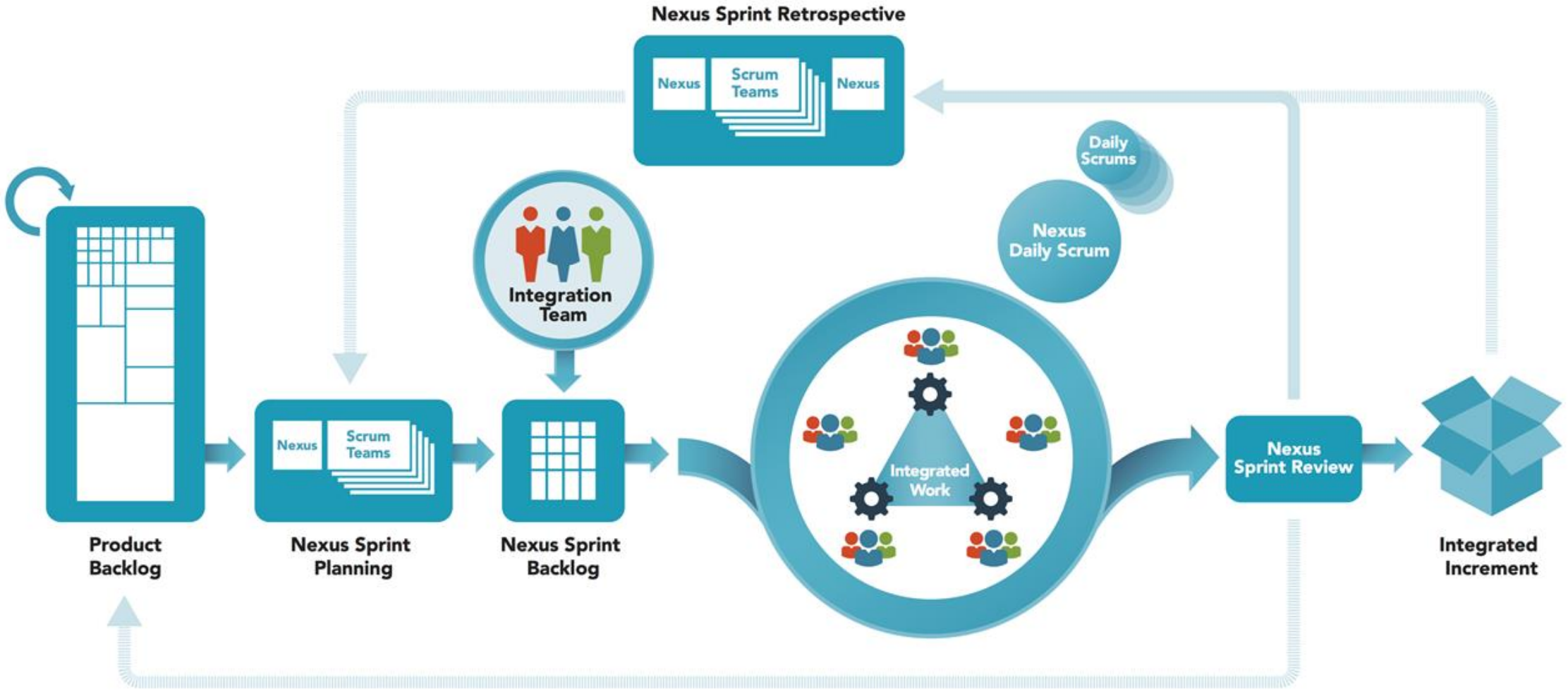
AutoScrum - *A Lean-Agile systems engineering framework*



AutoScrum facilitates best practices in terms of:

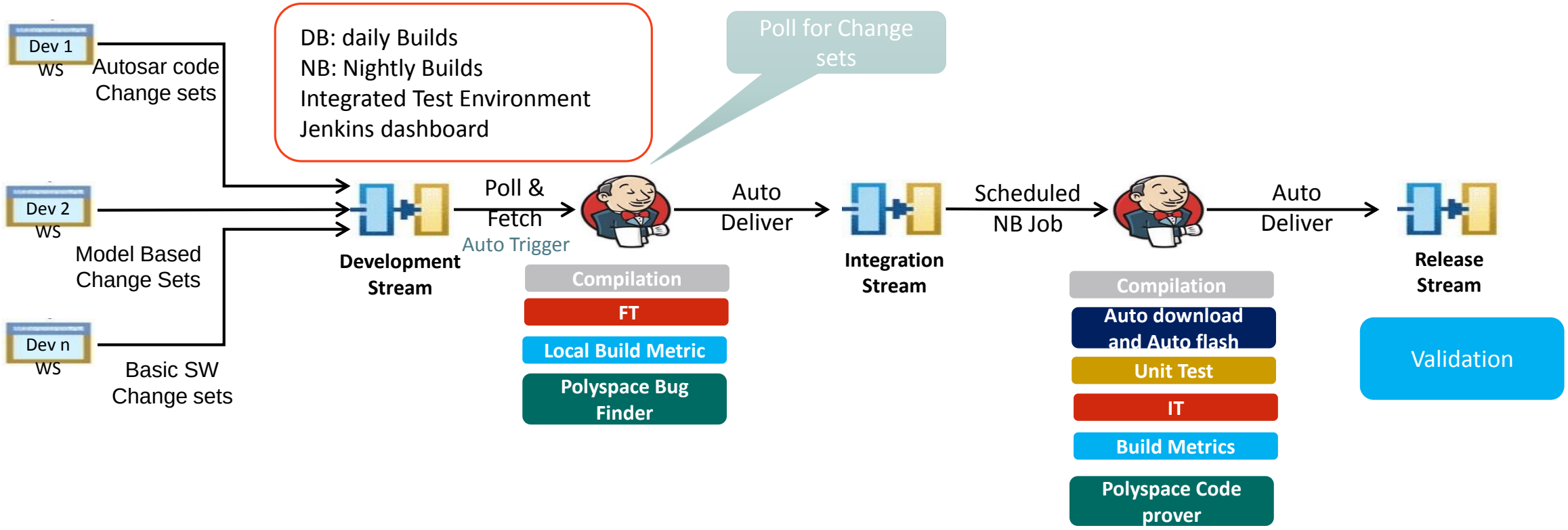
- System Feature Driven Development
- Fixed Cadence at Scale
- Systems Engineering Discipline
- Model Based Engineering
- Large Scale Team Collaboration
- Synchronized Cross Discipline Work
- Supports Platform/Domain Based Engineering
- Team/Backlog Inversions

Global Team Integration - Nexus



Source: https://www.scrum.org/resources/scaling-scrum

Deploy Continuous Integration with Tool Chain



❖ Continuous Integration Objective

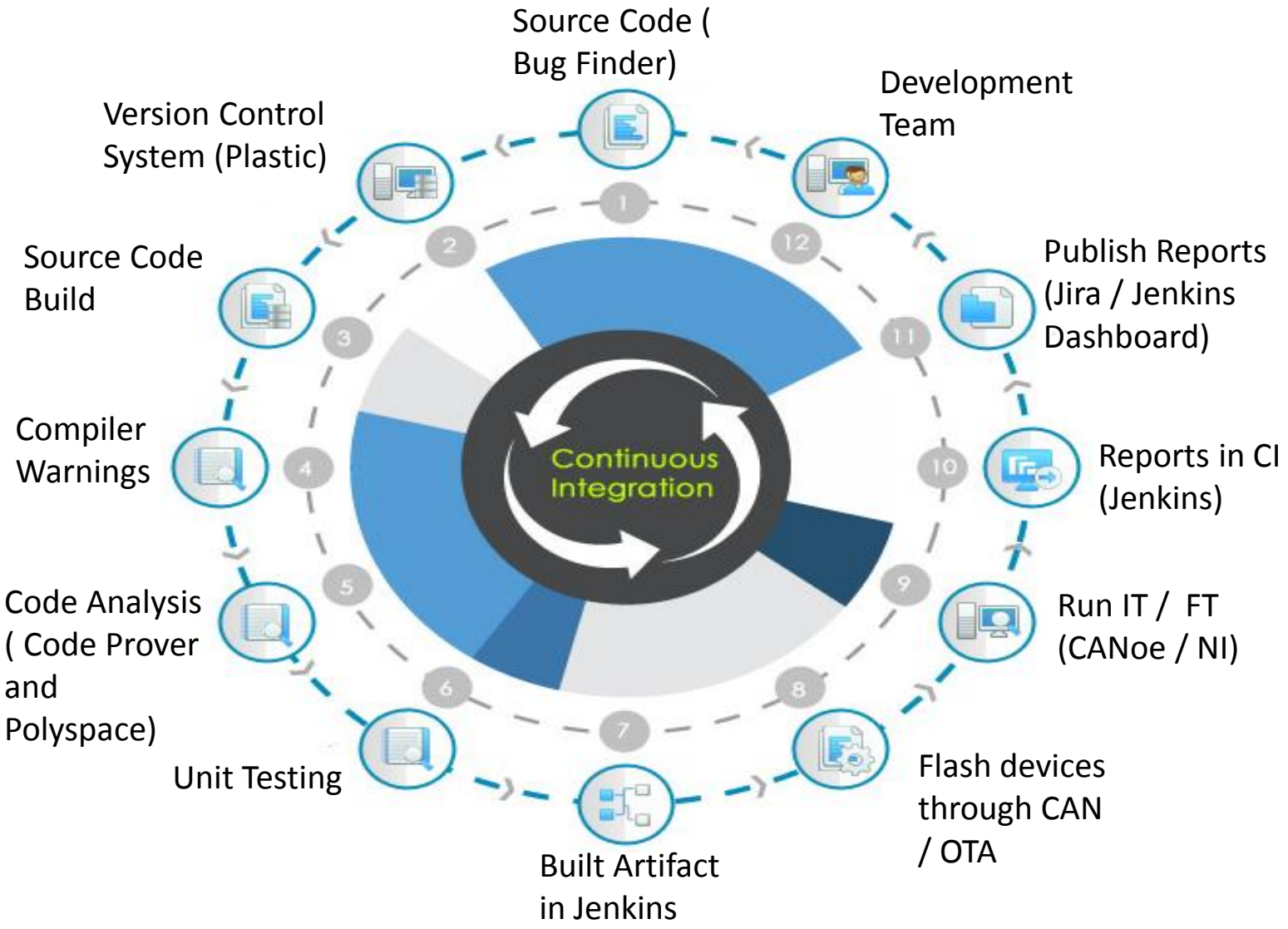
- Perform daily builds - Integrate, Build, Test and Release as per the build calendar
- Automated Test scripts for Unit, Integration and Functional Test
- Complier Warnings, Unit test, IT/ FT.
- Jenkins dashboard to host all relevant test data to monitor the health of the program delivery

❖ Derived Benefits

- Early engagement and early detection of SW defects from component to feature functional level
- Reduced cycle time in resolving SW defects
- Publish adequate errata for every SW release tested
- Accelerated defect verification / Fix – Identify, verify and close defects rapidly

APTIV

Scope of Continuous Integration – CI Flow



- **Build:** Configuration of Build scripts and scheduling automated builds through CI tool
- **Code** Analysis using Code Prover and Polyspace
- **UT:** Unit Testing
- **IT / FT:** Integration and Features Testing with Jenkins

APTIV

Summary

Model Based Design for Early Development and Validation

Agile – Faster, incremental and more predictable manner

Continuous Integration – Accelerated Growth and early solution

Thank You

APTIV