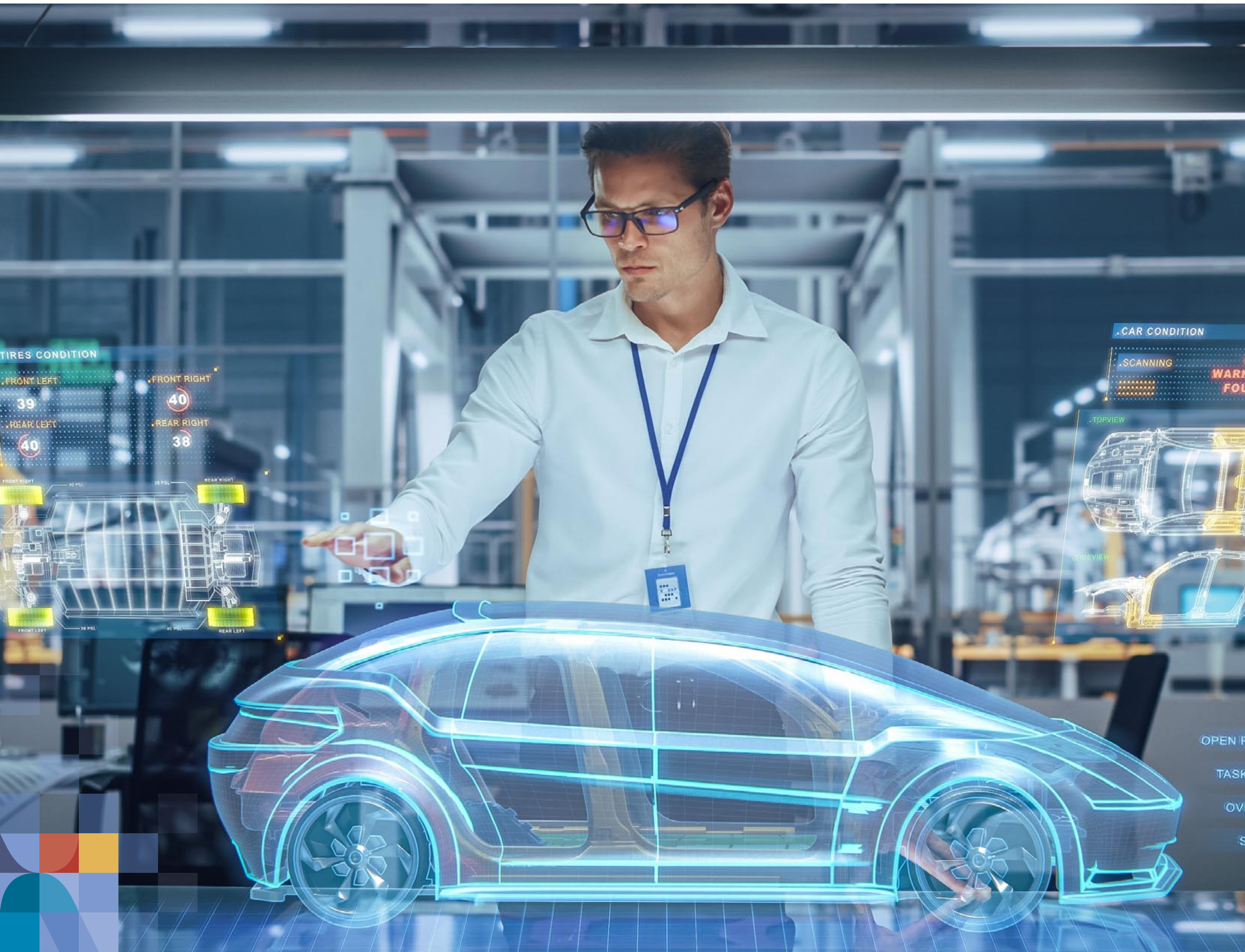
Trusted Global Innovator



WHITEPAPER / AUTOMOTIVE

How to build Automotive Solutions with SAFe

Content

Follow us:
NTT DATA Automotive

Visit us at:
de.nttdata.com/Industrien/Automotive

Connect with our expert:
Oliver Wörl, NTT DATA

Abstract

»Any organization that designs a system will produce a design whose structure is a copy of the organization's collaboration structure.«

Melvin E. Conway, Computer Scientist

Preamble
This white paper is written from an Agile and SAFe perspective. There might be other views/disciplines, like Systems Engineering, Product Development Process (PDC), or Product Life Cycle Management, who are based on different models.
»All models are wrong, but some are useful.«
George EP Box

Overview
The Automotive Industry runs through the deepest transformation process since the last 100 years. Within this disruptive environment, more and more OEMs adapt to the Scaled Agile Framework In dedicated parts of their organization.

An Automotive Solution is a Mobility Solution by a large OEM that is designed, developed and tested, e.g. autonomous driving taxis. Automotive Solutions consist of mechanics (Wheel, Chassis), hardware (e.g. ECUs) and software.

This White Paper provides you with a guideline, how to implement an Automotive Solution using SAFe (Scaled Agile Framework) in a major shorter product development cycle, using INCOSE Systems Engineering, aligned around a value driven virtual organization.

We start creating a **Future-state Architectural Runway** using **Model based Systems Engineering (MBSE)**.

According to Convey's Law, this allows us to create from the future-state Automotive System Architecture a very good design of an organizational/collaboration structure by building **Automotive Solution Trains**. We then **scale the Automotive Solution Train** up to the SAFe Portfolio Level and talk about the **Product Development Process (PDP) of Automotive Solutions**.

At the end MBSE, lead to a **robust, modular system architecture** that allows a continuous integration, which leads to **more FLOW (less waiting time, less bottlenecks) and value delivery**.

Details

Experiences with TOP 10 companies in the Automotive Industry, deep rooted existing process models (e.g. Product Development Process PDP), frameworks (INCOSE Systems Engineering and ISO 15288) and quality/compliance regulations are in place. This white paper deals with the following questions:

- How does the way, how system architecture is created, will change?
- How can the SAFe Large Solution Configuration be applied to the Automotive industry?
- How does MBSE (Model based Systems Engineer ing) and SAFe fit together?
- What kind of Lean Portfolio Management is needed for the Automotive Industry?
- How will the collaboration between OEM and its suppliers change?
- How to combine the Systems Engineering V-Model with SAFe?

This white paper article describes how to organize around value and architect for large automotive solutions that are implemented by thousands of people.

The implementation teams are aligned around value driven solution trains using a common vision, and roadmap without losing agility.

Automotive Systems Engineering

Systems Engineering (SE) is an approach to master the resulting complexity by looking at the whole "system of systems" before dividing it into sub-systems and finally into engineering artifacts such as mechanical parts or software code. This allows abstract modeling of the system architecture in terms of requirements, functions and logical system components including their interfaces – the RFLP approach (Requirements, Functional, Logical, and Physical).

What happens to systems engineering, if we architect for Innovation, Time to Market and for Fast Learning as described in the interview between Elon Musk (Tesla) and Jim Bridenstine (NASA)[5]. OEMs need to bring the Supplier knowledge and components back into their organization to be faster and more innovative (e.g. Tesla's Full Self-Driving Chip).

Many suppliers characterize the Automotive Industry, where Development Value Streams span over a network of companies with 50-100 ECUs brought in. Traditional OEMs try to build a bottom-up solution from the parts of their suppliers to optimize the costs. Innovation and rapid development over company boundaries is very difficult.

Besides Systems Engineering the following disciplines further evolve
- Safety Engineering
- Security Engineering
- Machine Learning & Artificial Intelligence

Systems Engineering has a high value in the development of complex systems. Experience shows that the use of SE reduces project schedule while improving product quality. Systems Engineering allow developing complex systems that meet the objectives in an efficient way:

- Minimize rework, by using a set-based approach and trade studies
- Catch errors early due to learning cycles and Shift-Left of Validation and Verification
- Effectively communicate

Systems Engineering is a process that we can use to develop a system that is too complex to design and build as a monolithic entity. When we develop Agile Solutions and Products in SAFe, Systems Engineering can help us to create a sophisticated architecture using the following practices:

- Loose Coupling, defined Interfaces
- Architect for testability
- Separate deploy and release
- Decouple release elements
- Architect for operations
- Architect for Security

Systems engineers plan, validate, simulate, configure, analyze, specify, design, and verify the system to ensure that it is logical, functional, and physical systems and their interfaces, as well as its performance, costs and other quality characteristics are balanced to meet the needs of the system stakeholders.

When working in an agile environment, you will have cross-functional teams, who work together on Firmware, Hardware, and Software for an individual feature.

However, Systems Engineering is crucial done by system and solutions architects, as well as cross-functional system teams.

Systems Engineering need to work on a repository implemented by different tools, like Rhapsody, CAMEO, and others. In SAFe, we call this the solution intent.

When developing using Artificial Intelligence and Machine Learning Algorithms, how does this fit with SAFe and Agile Development? The product and applications are using AI to further learn about their environment. How does this learning correlate with the architecture and the learning cycles in the Continuous Delivery Pipeline in SAFe? More information can be found in the new advanced article "Succeeding with AI in SAFe"[21].

Model based systems engineering (MBSE)

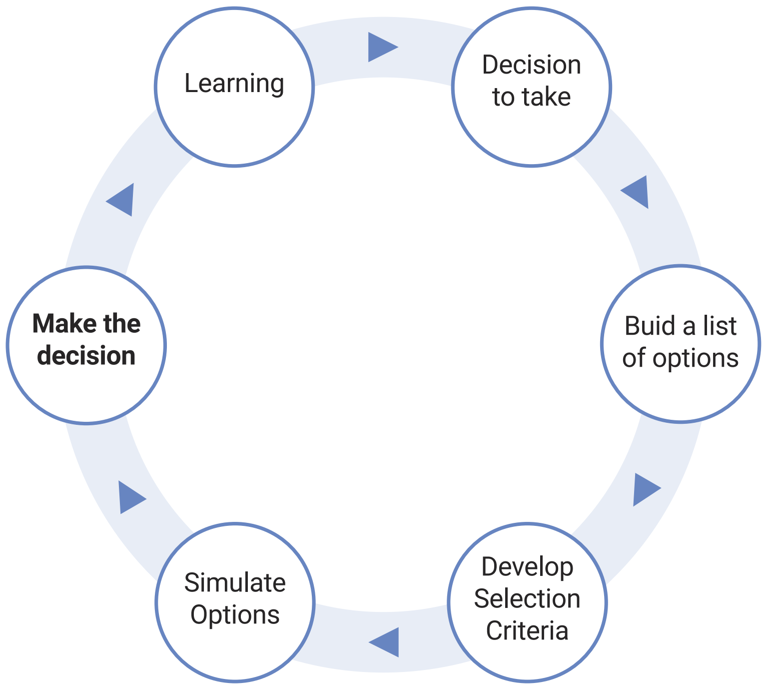
Model-Based Systems Engineering (MBSE) is the practice of developing a set of related system models that help define, design, and document a system under development. The SAFe has adopted MBSE in its Framework [10].

MBSE is an extension of SE with focus on formal models instead of free-text documents. These models are usually described using a defined notation and special tools for modeling, simulation, model validation etc. – e.g. the SysML language. The semantic relationships between the models are also defined in order to allow automation. MBSE promises better management of complexity (e.g. when dealing with ten-thousands of requirements), automation and better quality by avoiding misinterpretation of ambiguous text. But it also requires a high organizational maturity and integrated processes and IT systems.

Model Based systems engineering in the context of SAFe has been well described in [16] by 321Gang. The intent here is to give you an overview of how to apply MBSE in the SAFe context.Model based systems engineering (MBSE) uses models to make better (sub-)system level decisions. This fosters SAFe principle #3 Set-based approach.

Systems Engineering (SE) is an approach to master the resulting complexity by looking at the whole "system of systems" before dividing it into sub-systems and finally into engineering artifacts such as mechanical parts or software code. This allows abstract modeling of the system architecture in terms of requirements, functions and logical system components including their interfaces – the RFLP approach (Requirements, Functional, Logical, Physical).

Figure 1 MBSE Decision Making Process



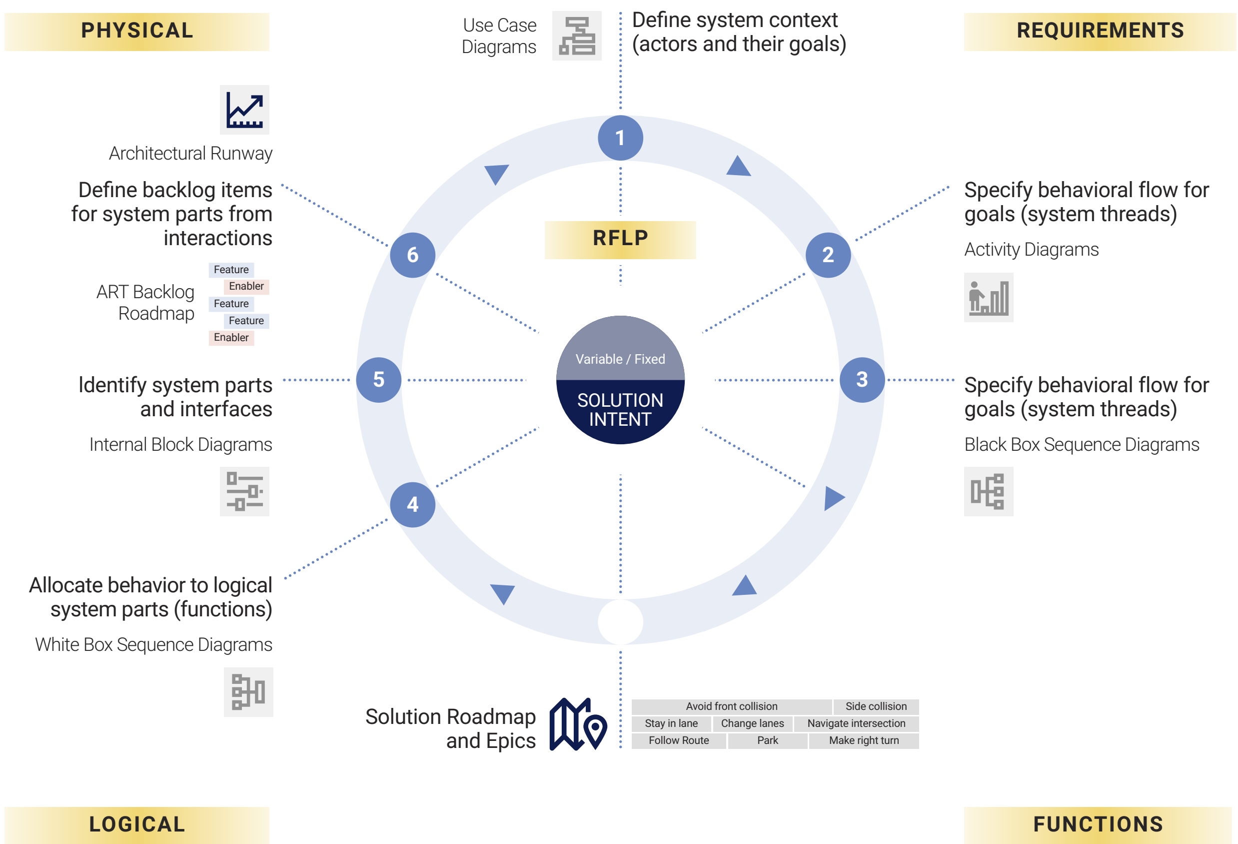
When you follow the RFLP approach, the MBSE decision making process will be applied to all layers, like Requirements, Functional, Logical, and Physical Layer. For a set of requirements, e.g. you develop a physical simulation model, and as a result adapt the architecture if necessary.

A further topic in SAFe is to build the SAFe Architectural Runway. The MBSE approach can be used in an iterative way to create JUST ENOUGH modeling knowledge to implement the next systems part. The MBSE modeling knowledge leads to a perfect Architectural Runway for building cyber-physical systems

The following picture shows the six steps to develop the systems engineering specifications. All the model diagrams are using SysML and are included in the SAFe solution intent. The SAFe Artifacts Solution Roadmap and Epics, Architectural Runway and an initial ART Backlog and Roadmap will be created through the specification process.

MBSE supports the creation of main SAFe artifacts. The Architectural Runway consists of Behavior Models, Logical and Physical Architectures. Experts for a certain architectural domain are coming together. An example is an electric vehicle with Cockpit, Battery Management, Braking System and Drive System.

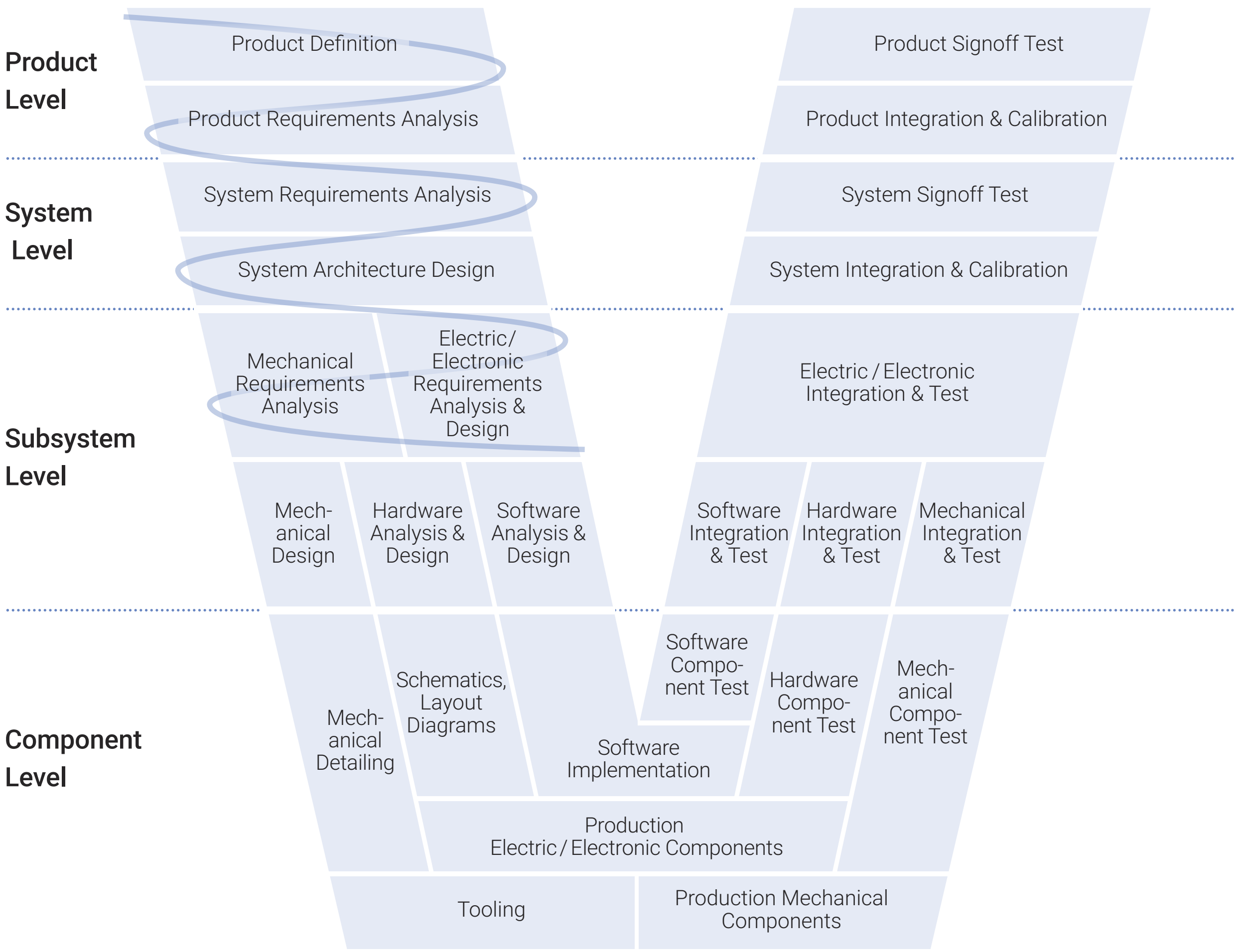
Figure 2 6 Steps to develop systems engineering specifications in SAFe (Product Roadmap and ART Roadmap)



MBSE in the V Model

When implementing the V-Model following the MBSE approach, you are diving down in a spiral on the left wing from top to bottom. You model your product, run simulations and iterate until you have a clear picture on the product requirements. And then you rinse and repeat on the system and subsystem level. Your model or digital twin allows you to learn about your system in iterations and PIs.

Figure 3 MBSE Zigzag-Pattern



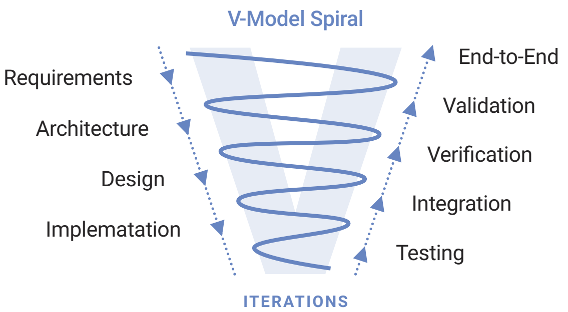
For a long time OEMs have used several techniques like Hardware in the Loop (HiL), Model in the Loop (MiL), and Software in the Loop (SiL), which already implements an agile, iterative approach. The higher we can simulate or test the product or system in the V-Model, the more we can identify architecture or design flaws from the beginning.

A customer can have the following Systems Engineering levels:

1. **Product**
2. **System**
3. **Subsystem**
4. **Component**

The next level of product development will be to implement an iterative, spiral approach over the whole V-Model. E.g. the requirements will be modeled and simulated, so that they can be tested. The architecture is created and validated, as CARIAD did this with it's Platform MVP E3 2.0, and Design and Implementation is continuously integrated and tested within iteration.[19]

Figure 4 Iterations over the whole V-Model



The next level of product development will be to implement an iterative, spiral approach over the whole V-Model. E.g. the requirements will be modeled and simulated, so that they can be tested. The architecture is created and validated, as CARIAD did this with its Platform MVP E3 2.0, and Design and Implementation is continuously integrated and tested within iteration.[19]

SAFe approach to systems engineering disciplines

When we review the systems engineering disciplines, we can see that most of the different disciplines are addressed in SAFe using different wordings and concepts:

1. Requirement Management
Since SAFe has its roots in the book "Agile Software Requirements"[1], this discipline is fully embraced by SAFe.

To name a few obvious mapping:

Systems engineering	Scaled Agile Framework
Analyzing Stakeholder Needs	Design Thinking
Behavior Models	User Journey Maps
Requirement Hierarchy	Epic, Capability, Feature, Story
Constraints	NFRs
Optimize Design through trade studies	Set-based approach, Simulations, Exploration

In addition, SE requirement management defines further artifacts, where the mapping is defined in more detail in "01 Achieving Regulatory and Industry Standards Compliance with SAFe 5.0"[7]:

- Requirement Specification
- Baselining
- Impact analysis
- ASIL classification
- Release assignment
- Traceability
- Function assignment
- Verification criteria
- Conflict resolution
- Cost estimation,
- Compliance specifications
- Supplier assignment (component specification), exchange of specifications, needs validation, requirements review,
- Feasibility analysis

2. Integration, Verification and Validation (IVV)
Vehicles underlie compliance regulations e.g. regarding functional safety (ISO 26262), software process reference and assessment models (Automotive SPICE), cyber security and software update management (UNECE WP.29) and CO2 emissions (WLTP, RDE). These regulations impose requirements on the development process or the product itself. Regulations that affect the development process can be implemented with processes, methods, tools, and organizational measures. Regulations that address the product must be fed into the development process as requirements and systematically implemented, verified, and validated in order to obtain certifications/type approvals.[18] Traceability between the artifacts must be maintained over the product life cycle so that evidence for meeting the regulations can be presented e.g. during an assessment or audit.

SAFe provides concepts and processes that continuously foster verification and validation, while making the V&V activities part of a regular flow:

- Iterations verification using automated story testing
- Iteration validation using bi-weekly system demos
- PI validation using quarterly PI system and solution demos
- Release validations

Compliance Example:
Within the UNECE, the CSMS (CyberSecurity Management System) and the SUMS (Software Update Management System) are two important regulations the automotive industry must fulfill. The aim is to specify a structured process for the CSMS/SUMS at car manufacturers and for cybersecurity and software update in the vehicle, which reduces the success rate of hacker attacks and establishes a standard against cyber threats in the automotive industry. Using the SAFe V&V activities, those regulations can be implemented.[18] The PEDCO Applied SAFe Platform[9] is a process model on the standard process platform stages that maps difference compliance frameworks like ASPICE, or Safety to the SAFe Framework. OEMs can extend Applied SAFe with their own process models and therefore prove the process conformity. This reduces the implementation time of SAFe processes models.

3. Configuration Management
The SAFe Concept to implement versioning, baselining, delta analysis, etc. is the "solution Intent", which contains all artifacts (specifications, code, tests, etc.) that are subject of configuration management. The "solution intent" also contains hardware and physical models, and a list of experiments by registering rooms and locations. To show the complexity an OEM might implement the "solution intent" using an IT landscape with over 100 of applications.

When integrating the system continuously, configuration management may become a bottleneck, because the system will be continuously changed. Automation and Tooling of the Configuration Management is key.

4. Release Management (not part of ISO 15288)
One major Systems Engineering discipline when building a vehicle or other safety relevant products is Release Management. Who takes the responsibility for releasing products? In the SAFe "Agile Product Delivery" Competency, the concept "Develop on Cadence and Release on Demand" differentiates between a technical release and a business/contractual release:

a. Technical Release Management
The complete solution train (450 people) especially the teams with their Build-In Quality core value are responsible to check the technical compliance by building automated compliance checks.
The technical "release" happens on the sprint and PI cadence, where development increments are continuously being developed and accepted by PO, PM and SM with the support of DoD and test results.

b. Business Release Management

For an Automotive Solution you have one or more business owners who take the responsibility to sign and give fast feedback to the teams. This release management function has the authority, knowledge, and capacity to foster and approve releases.

5. Function and System Architecture Creation

In SAFe Architecture descriptions, Functional Architecture, Logical Architecture, and Physical Architecture are created using the "Agile Architecture" approach. This approach is not part of Systems Engineering.

Agile Architecture is a set of values, practices, and collaborations that support the active, evolutionary design and architecture of a system. It is a collaboration and balance between intentional architecture driven by the system/solution architects and emergent design developed by the teams. It is important that architecture will be developed iteratively. Architecture evolves supported by MBSE where models evolve more easily than documents.

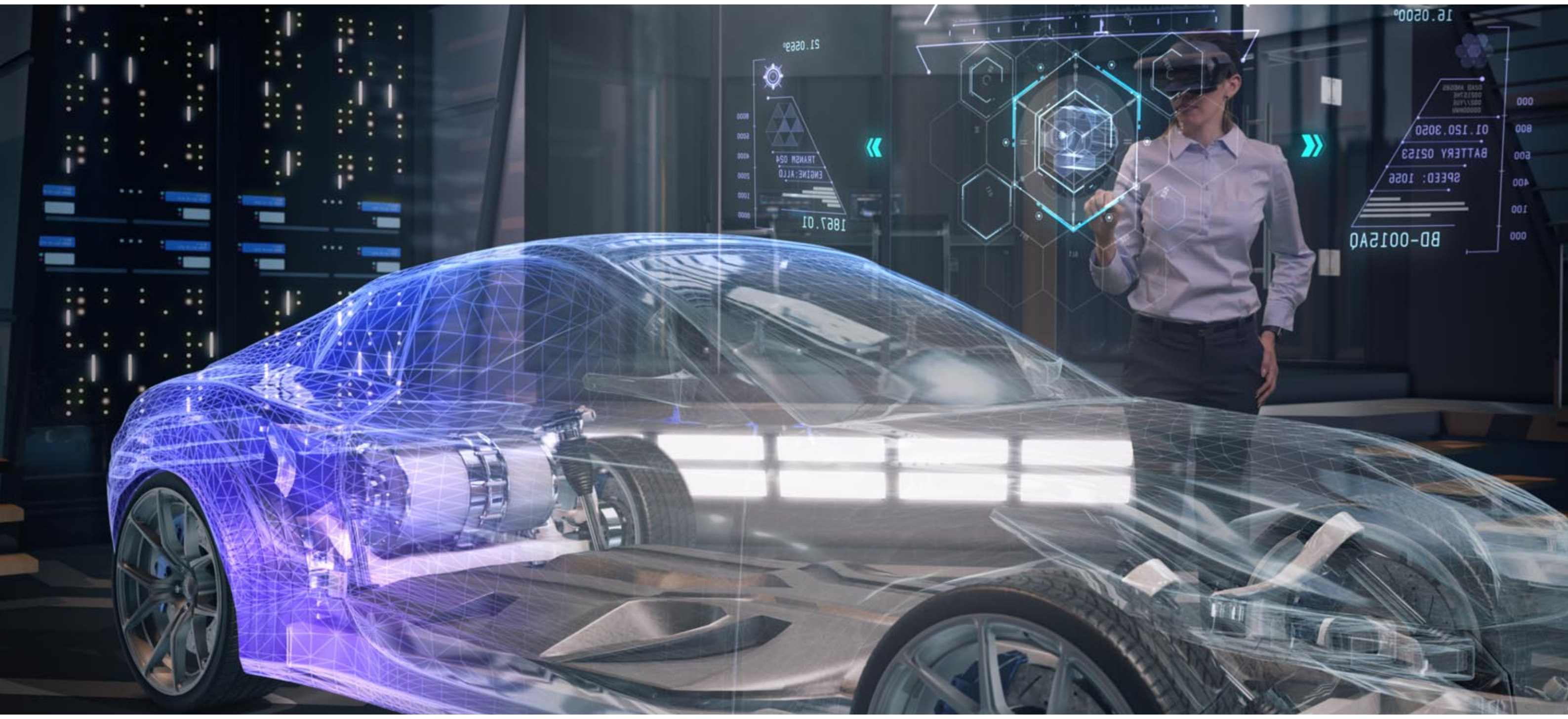
6. Quality Management

For applying system quality in cyber physical systems the Built-In Quality core value in SAFe ensure that each solution element, at every increment, meets the appropriate quality standards throughout implementation:

- Increase Quality by behavior-driven and test-driven implementation
- Apply pair work and collective ownership
- Balance the testing portfolio with many fast, automated tests
- Build a DevOps pipeline
- Architecture and design to ease Testing for cyber-physical systems
- Continuous integration (CI) for cyber-physical systems

7. Project Management

Most of the projects where the Systems Engineering approach is used, manage the product lifecycle time using a phase gate approach, which is based on the traditional waterfall project management approach. SAFe has transformed this approach into a scaled, lean-agile, value stream and team based framework for long-lived products. Project Management becomes Product and Solution Management as described in.[20]



The Automotive Large Solution Model

The Enterprise Solution Delivery Competency Article in SAFe describes how to apply Lean-Agile principles and practices to the specification, development, deployment, operation, and evolution of the world's largest, connected cyber-physical systems.

In order to build complex systems, we need the cooperation of many people. The goal is to create communication and collaboration structures between experts to minimize waste and maximize value creation.

Building Automotive Solution Trains

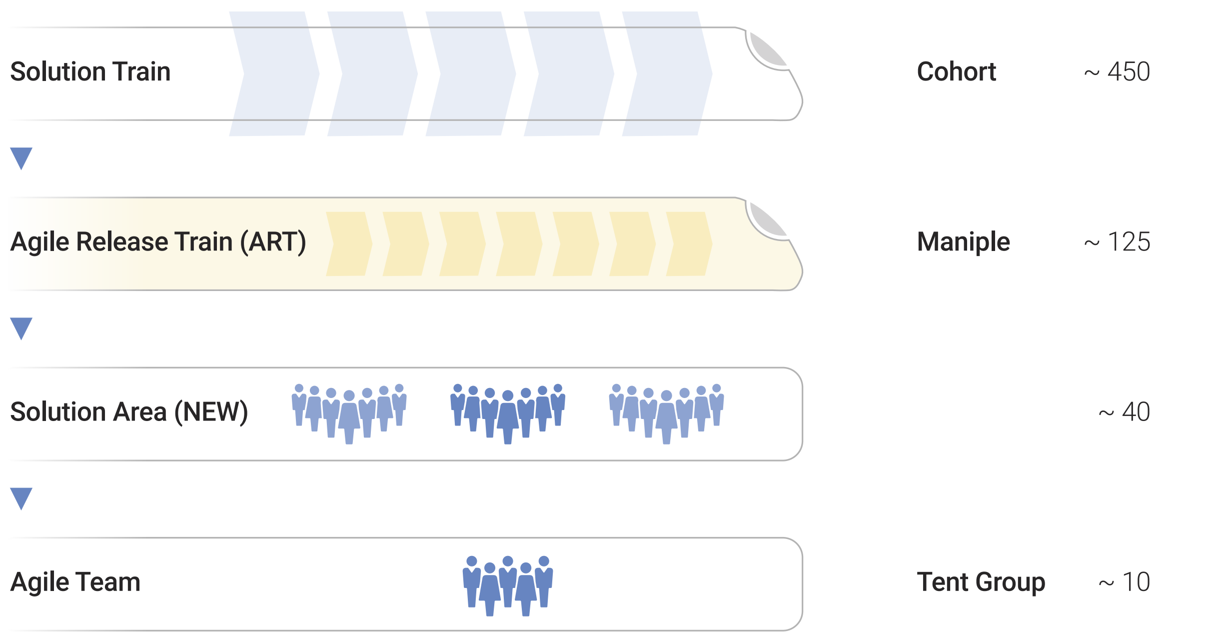
Wolfgang Brandhuber [12] has extended the SAFe Large Solution Model by Solution Areas, which perfectly fits to the Product Development Process (PDP) Organization of an Automotive Mobility Solution.

His thoughts have been influenced by the way the Romans have organized their troops. In a Roman Imperial Legion the basic structure of the army is as follows:

- **Contubernium (tent group):** consisted of 8 men.
- **Maniple:** consists of 10-12 contubernium
- **Centuria (century):** was made up of 10 contubernium with a total of 80 men commanded by a centurion.
- **Cohorts (cohort):** included 6 centuriae or a total of 480 fighting men, not including officers. In addition, the first cohort was double strength but with only 5 centuriae instead of the normal 6.
- **Legio (legion):** consisted of 10 cohorts.

The Roman structures have been proven to align an army to a common vision, roadmap and plan with regards to the social networking restrictions of human beings, like Dunbar's Number.

Figure 5 SAFe Organizational Units that implement „System of Systems“



In Mobility Solutions often thousands of people need to work together. In that Solution Area (NEW) case SAFe develonment value stream chains are needed.

Wolfgang Brandhuber[12] introduced solution areas, which is a collaboration unit of 3 +/- 1 agile teams to work in a common problem space within a Solution Train. Solution Areas provide the following benefits:

- **Decentralized Dependency Management**
  With the size of the solution, complexity and dependencies increase. We don't care about the dependencies within an agile team. The same way we do not care about the dependencies in Solution Areas either. Solution Areas care for their dependency management themselves.

- **Strengthening the Feature-Team Idea**
  We can only get a grip on more complex problems through the collaboration of an increasing number of experts. When we organize solution areas around features we can get a completely different class of complexity under control.

- **Flexible Solution Areas**
  are not fixed organizational units, but can change from PI to PI as the collaboration between the agile teams is necessary. The agile teams within a solution area decide which kind of common backlogs, ceremonies, or roles should be used in a solution area. Solution Areas use the "Collaboration Interaction Mode" from Team Topologies[13].

Automotive Solution Trains consist of

- **Agile Release Trains**

- **Solution Areas, which can be also cross-ART**

- **Agile Teams, ideally created in self-selection workshops**

When you create an Automotive solution you should orient to the Team and ART Topologies in SAFe[13]. In the Team interaction mode "Collaboration" you will have changing communication structures between the teams depending on the problem the teams are currently solving. A problem can be a step in the development value stream or a feature where the teams focus on a PI. Therefore it makes sense that the right persons with the right skills come together and 2-4 teams form a solution area to collaborate for a certain PI. Especially, when you work on certain interfaces.

Please keep in mind, the solution areas build no extra layer in SAFe. They have no Product Owner, and are flexible in defining what they need in a PI. 4 collaboration pillars are discussed:

1. **Events:**
   What kind of Solution Area Events are needed?
2. **Artifacts:**
   E.g., do we need a common backlog or Definition of Done?
3. **Roles:**
   Do we need temporary a Scrum Master or Product Owner for the Solution Area
4. **Collaboration Practices:**
   What kind of agile practices are we using? E.g. Pair Work, KANBAN,

The collaboration patterns and feature refinement are first developed in the Feature Refinement Meetings and then finalized in the PI Planning. This increases heavily the T-Shaped Skills. T-shaped skills is a metaphor used in job recruitment to describe the abilities of persons in the workforce. The vertical bar on the letter T represents the depth of related skills and expertise in a single field, whereas the horizontal bar is the ability to collaborate across disciplines with experts in other areas and to apply knowledge in areas of expertise other than one's own.

For dependency management the ART Program Board is enhanced by a solution board with dependencies between the solution areas and a solution area board with dependencies between the teams.

Only dependencies to the outside world are shown in the Solution Board. This has the advantage that we have only dependencies to the solution areas and not to every team.

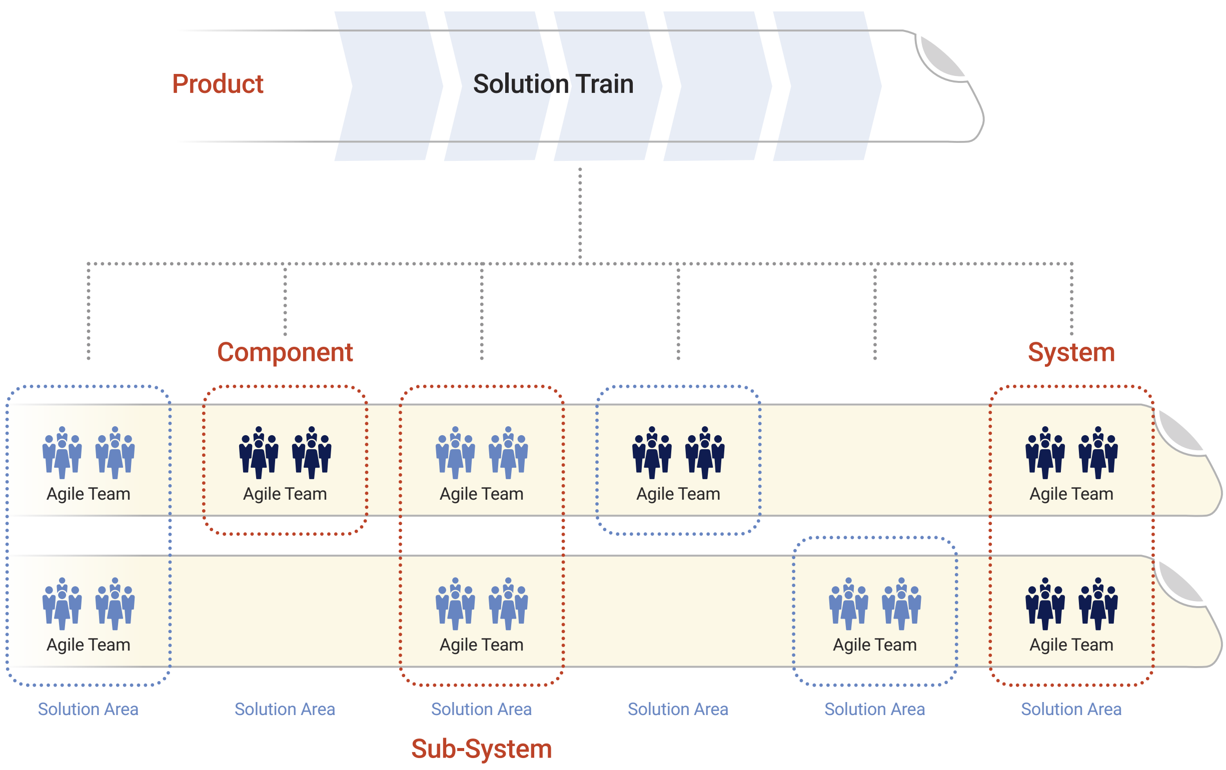
In the middle of every PI a solution planning conference with representatives from the solutions areas will be conducted to (re)-align to a common solution with a common vision and roadmap, while the PI Plannings with their Solution Pre- and Post Plannings remain. For the solution planning conference make sure you keep the 1:5 factor for representative selection, which means in a solution train of 400 people, make sure you have at least 80 people in the solution planning conference.

The teams remain organized in Agile Release Trains, which might change from PI to PI.

The concept of an Automotive solution train is explained in detail in [12].

When we look at the V Model, we can map the V-Model levels to the SAFe Framework Levels of Large Solution, Agile Release Trains, Solution Areas and Agile Teams.

Figure 6 Mapping of the V-Model to SAFe



Scaling Automotive Solution Trains

1. Super Solution Trains

When a solution train (limited to ~ 450 people) is not enough, some companies, like Boeing, are creating a so-called "Super Solution Train ". A Super Solution Train consists of several Solution Trains. This happens when the architecture is so monolithic that most of the people in a Super Solution Train need to align.

However, the theory of social systems shows us that a Super Solution Train creates an organization which is so large that a common alignment (common vision, common roadmap) is no longer possible. Alignment means that if someone changes a system the connected systems get indirectly informed by the others.

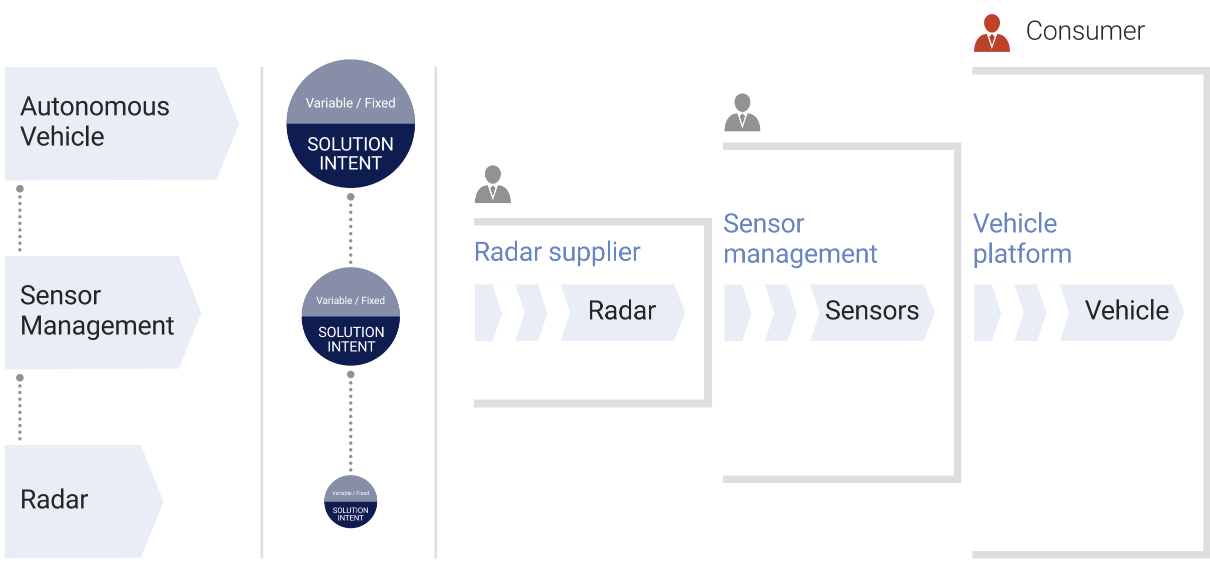
To share an alignment with others is limited to an organization which has the size of about 450 people which is the size of a Roman troops cohort. In addition a Super Solution Train creates an inflexible hierarchical structure, which cuts down the agility. A better flexible organization is to coordinate the solution train value streams via portfolio-like events, e.g. Portfolio Strategic Review and Portfolio Sync Meeting.

In the ARTs PI Planning, the people have guardrails to plan by themselves, but if you create Super Solution Trains you don't have this freedom any more. The spirit of the agile way is gone. In (Super) Solution Trains People make conservative decisions and are handing down work packages to the teams. With (Super) Solution Trains we end up in the classical work break-down structure mode.

2. Value Stream Chains

If the product will be created by more than 450 people, or if the solution train spans multiple enterprises, like OEM and further suppliers, then V Model Levels are mapped to a Development Value Stream Chain, where each value stream can be a Product, System, Sub-System or Component.

Figure 7 SAFe Development Value Stream Chain



A System in one context (vehicle to consumer) is a Customer in another (sensor management to vehicle platform). The solution intent (from Radar to Sensor Management to Autonomous vehicle) is aggregated to support integration and compliance. If we realize that certain Systems, Sub-Systems or Components have the same CDP (Continuous Delivery Pipeline) we could aggregate those on a same development value stream, implemented by a solution or agile release train.

If you have a good architecture, where interfaces are clearly defined using Systems Engineering you no longer need events from a solution train or a super solution train. The way out is to decompose your architecture in loosely coupled parts (using thin interfaces), which allows you to run separate value streams. The value stream coordination is well described in the Value Stream Coordination Article. At the beginning, we identify dependencies that are more technical. The focus moves then to product driven and finally business driven dependencies. Some teams from each value stream need to coordinate, but not all.

3. Portfolios of Value Streams

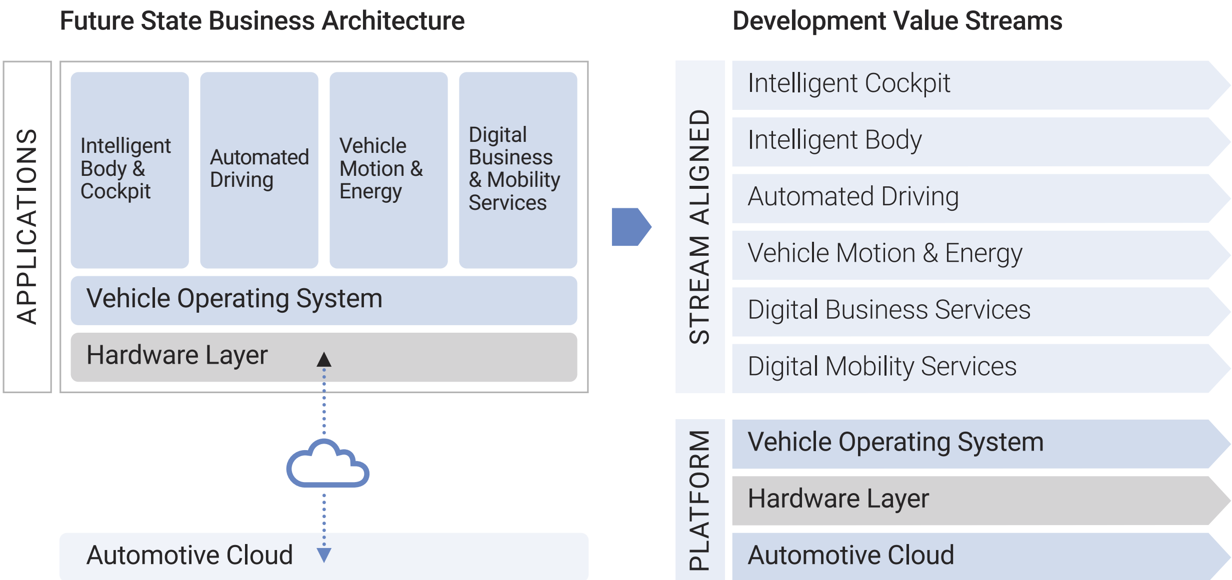
The best way to preserve flexibility is to create a portfolio level which embraces the value streams of the related solutions trains so that you can connect strategy to execution within a Lean Portfolio Management (LPM)

OEMs in the Automotive Industry have thousands, and even tens of thousands, of IT, system, application, and solution development practitioners. In this case an OEM has multiple SAFe portfolios connected to the enterprise strategy and enterprise budget. For more information please review the Enterprise Article in SAFe.

The question is, how to design the different SAFe Portfolios within an OEM?

The following picture shows the different Development Value Streams that implement a tech stack of a modern autonomous-driving, electric vehicle. Good Practices are to combine the value streams into a SAFe Portfolio, or even to create a Portfolio for a large value stream, like Vehicle Operating System.

Figure 8 Sample Business Architecture and Development Value Streams at an Automotive Tech-Stack Vendor



In an OEM enterprise, you have of course further value stream patterns[15], where the collection of development value streams in each pattern could be a seperate SAFe Portfolio:

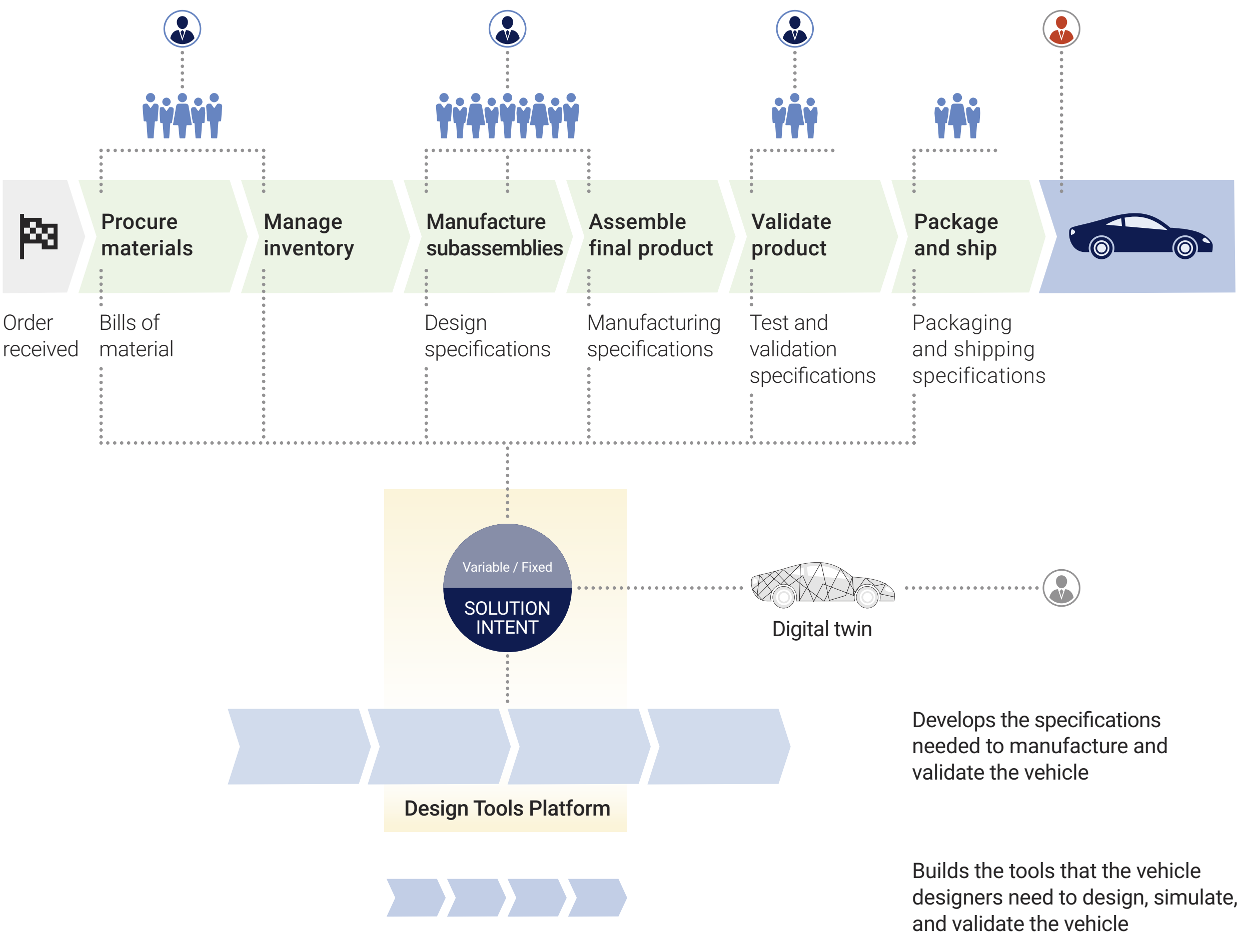
- **Manufacturing Value Stream Pattern**
  The Manufacturing Value Stream Pattern consists of an operational value stream (green) that manufactures the vehicle on a production plant. This is supported by a development value stream (blue) that develops the Processes, Methods and build the Tools (PMT solutions) needed to manufacture and validate the vehicle.

- **Product Development Process (PDP)**
  From the point of view of SAFe the PDP is a development value streams that develops the specifications needed to manufacture and validate the vehicle.

- **Supporting Value Stream Patterns**
  Development Value Stream (e.g. ERP System) that supports internal operational value streams like Sales, Procurement, Financial Auditing

Another way could be to create a portfolio of development value streams that support the creation of a new car model from designing over manufacturing to servicing.

Figure 9



PDP by PMT (Process, Method, Tools)

When you develop PMT-Solutions for designing and building vehicles, some customers have a process owner for each of the 500+ processes describing the life cycle management processes of a vehicle. However, there is no process owner defined in SAFe. How do we cope with this? The PDP (product development process) for cyber-physical systems is a blueprint of processes with a clearly defined control.

However, in Software the situation is more complex. The fixed, defined PDP process (Concept, Requirements, Implementation, Validation, Launch, Support) struggles with software, which can be integrated and release every day. Agile Frameworks support the short release cycles.
When we want to transfer the Agile Framework model to cyber-physical products, we have to have a more flexible Product Development Process.

Instead of having a fixed, defined PDP process, you create a product development process in a Lean Agile Way, even for systems. In addition, we need to make sure that system architects define clear interfaces between the systems.

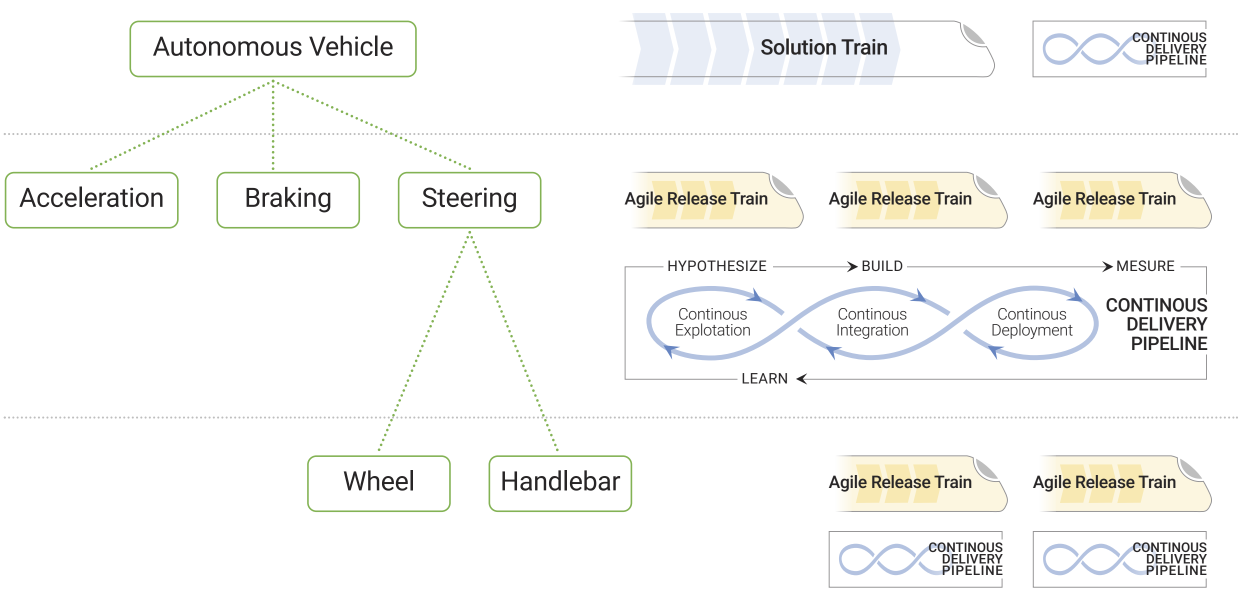
The question is how to align and create a flexible PDP process and a common understanding in a system-of-systems (SoS) environment within an Automotive Solution Train, where about thousands of people work on? From our experiences, the following measures help:

- **Enhance the ability to overlook a system of systems.**
  By practicing a CDP (Continuous Delivery Pipeline) approach for SoS, we get a large picture of the whole system of systems constellation and gives the Automotive Solution Train means to control a broader range of systems
- **Have a network with enough system architects:**
  One system architect can oversee about the work of 25 agile team members. Therefore, we need about 20 system architects in an Automotive Solution Train with 450 people. In vehicle, projects within an OEM thousands of people are involved, so that a development value stream chain is created and implemented by multiple solution trains.

Finally, we need a hierarchy of Lean Agile Delivery Pipelines practiced by DevOps.

Now, how do you build a CI/CD pipeline for a system of systems?

Figure 10 CI/CD Pipeline for a system of system constellation



A system of systems is usually implemented by a solution train, which consists of several Agile Release Trains. Each Agile Release Train might have a dedicated SAFe Delivery Pipeline (including CI/CD) or some Trains are using common Delivery Pipelines.

The Delivery Pipeline in SAFe is normally implemented by the belonging system team. Having a system of systems constellation, including hardware, and software, this might get quite complex.

Since the Online Remote Update Functionality, where Function on Demand (FoD) can be upload into the car, also a car has an operating phase.

Therefore we recommend the following approach:

1. **DevOps Architect Team (Scrum Team in Sprints)**
   This team works on all predictable and plannable tasks to build the DevOps Pipeline. You start building this team from the top system level. As you go you will have several DevOps Architect Teams each for every sub system pipeline.
   We recommend organizing all the DevOps architect teams for all systems pipes as an ART, with an own PI Planning, so you have enough cohesion for a seamless overall Pipeline. This can be an integration ART or infrastructure ART with 100% team members who have the following responsibility:
   a. Experts in the Infrastructure (Cloud, Vehicle Hardware, etc.)
   b. Build up the Infrastructure/Pipeline
   c. Members are willing to train the teams
2. **DevOps Developer (KANBAN Mode)**
   The DevOps Developers are organized in a system team per ART and have small tasks to keep the pipeline running. This is more an operation of the DevOps pipeline. DevOps Developers help with the day-to-day system integration tasks within an ART.

When applying the DevOps approach from the Software World to the space of cyber-physical systems and mechanics like a vehicle, we are at the level of manufacturing (building and integration) the vehicle, which is called *Industrial DevOps*.

Industrial DevOps[17] is a DevOps approach, methods and culture to the development and production of industrially manufactured products. The continuous principle of the approach is a process of operating, observing and developing the entire system. The goals are the gradual system integration, fast and flexible adaptability of the production as well as the improvement of the processes and the quality of services and products. At Tesla, e.g. the vehicle models are routed to flexible production lines.

For the Systems Engineering integration NTT DATA has created SENSEI (**Systems Engineering a**nd **S**calable **E**nterprise Integration)[18] in order to bring IT integration architecture best practices to the CI/CD pipeline for IT Infrastructure.

Summary

The traditional Product Development Cycle (PDC) processes for Cyber physical Systems takes too long in today's VUCA (volatility, uncertainty, complexity, and ambiguity) world, where software get more and more dominating factor.

When you combine System Engineering Best Practices with a Scalable Agile Framework, like SAFe you gain speed of value delivery as well as robust architecture and excellent quality managing the product life cycle of complex systems of tomorrow.

Finally, we need to come to a Lean Compliance and QMS approach that allows the OEMs to certify their mobility solutions in Safety, APSICE, UNECE, and further regulatory standards.

NTT DATA provides you with the relevant cross-domain consulting to transform your Product Development Cycle based on best practices of SAFe, System Engineering, Scalable Enterprise Integration and traceability of compliance regulations.

Learn More

1 Agile Software Requirements — Dean Leffingwell, 2011.

2 Automakers use virtual reality to cut the development time for vehicles like the Hummer EV. 2020. https://www.cnbc.com/2020/11/08/automakers-use-virtual-reality-to-cut-the-development-time-for-vehicles-like-the-hummer-ev.html

3 Why Tech Giants Like Amazon Are Designing Their Own Chips — And Who Benefits. 2018. https://www.thestreet.com/opinion/why-tech-giants-are-designing-their-own-chips-14807638

4 MHP - A Porsche Company: Scaled Agile @ systems engineering https://www.mhp.com/fileadmin/www.mhp.com/assets/pdf/MHPWhitePaper_Scaled-Agile-Systems-Engineering.pdf

5 Exclusive interview with Elon Musk and Jim Bridenstine about #DM2, SpaceX's first crewed launch! Everyday Astronaut, 2020. https://www.youtube.com/watch?v=p4ZLysa9Qqg&t=266s

6 Agile at Tesla with Joe Justice. The Agile Wires, 2021. https://www.theagilewire.com/recordings/agile-at-tesla-with-joe-justice

7 SAFe Toolkit: 01 Achieving Regulatory and Industry Standards Compliance with SAFe 5.0

8 SAFe Advanced Article: Agile Architecture in SAFe https://www.scaledagileframework.com/agile-architecture/

9 PEDCO - Applied SAFe https://www.pedco.eu/appliedsafe/?lang=de

10 Model Based System Enineering in SAFe https://www.scaledagileframework.com/model-based-systems-engineering/

11 Kainzbauer/Brandhuber: Designing Large Solutions: A structured approach to Value Stream Identification, SAFe Summit June 10-11th 2020.

12 Aligning ARTs within a Solution Train – Dr. Wolfgang Brandhuber, SAFe Summit May 7-10, 2019

13 Skelton, Matthew, and Manuel Pais. Team Topologies: Organizing Business and Technology Teams for Fast Flow. IT Revolution Press, 2019.

14 SAFe Framework: Enterprise Article https://www.scaledagileframework.com/enterprise/

15 SAFe Framework: Development Value Streams https://www.scaledagileframework.com/development-value-streams/

16 321 Gang: MBSE in the context of Scaled Agile Framework https://321gang.com/works/2018-10-05-model-based-system-engineering-mbse-in-the-context-of-scaled-agile-framework-safe/?wvideo=c0tdrfsasm

17 IT Revolution (DevOps Enterprise Forum) 2020: Applied Industrial DevOps 2.0

18 NTT DATA: SENSEI, Systems Engineering and Scalable Enterprise Integration, Jens Krüger https://de.nttdata.com/insights/whitepapers/sensei-systems-engineering-and-scalable-enterprise-integration

19 CARIAD: MVP E3 2.0: A huge step in our platform development https://cariad.technology/de/en/news/news/mvp-software-platform-development.html

20 Mik Kersten: From Project to Product – https://projecttoproduct.org/the-book/

21 SAFe Advanced Article – Succeeding with AI in SAFe https://www.scaledagileframework.com/succeeding-with-ai-in-safe/

Author

Oliver Wörl
SPCT Candidate, Chief Architect for Business Agility
NTT DATA

Let's get started

See what NTT DATA can do for you.

- Deep industry expertise and market-leading technologies
- Tailored capabilities with your objectives in mind
- Partnerships to help you build and realize your vision

Contact one of our authors, or visit **nttdata.com** to learn more.

Referencens – Pictures

Page 1: Superstar / Shutterstock
Page 2: Gorodenkoff / Depositphotos
Page 3: Superstar / Shutterstock
Page 4: Superstar / Shutterstock
Page 11: Superstar / Shutterstock
Page 20: Superstar / Shutterstock

About NTT DATA

NTT DATA – a part of NTT Group – is a trusted global innovator of IT and business services headquartered in Tokyo. We help clients transform through consulting, industry solutions, business process services, IT modernization and managed services. NTT DATA enables clients, as well as society, to move confidently into the digital future. We are committed to our clients' long-term success and combine global reach with local client attention to serve them in over 50 countries.

Visit us at nttdata.com

NTT DATA Deutschland GmbH
Hans-Döllgast-Straße 26
D-80807 München
Germany
Fon +49 89 9936 -0
www.nttdata.com/de