



# Adopting Agile/DevOps ALM in Automotive & Safety-critical Development

Peter Haller - FAE

## Digital transformation

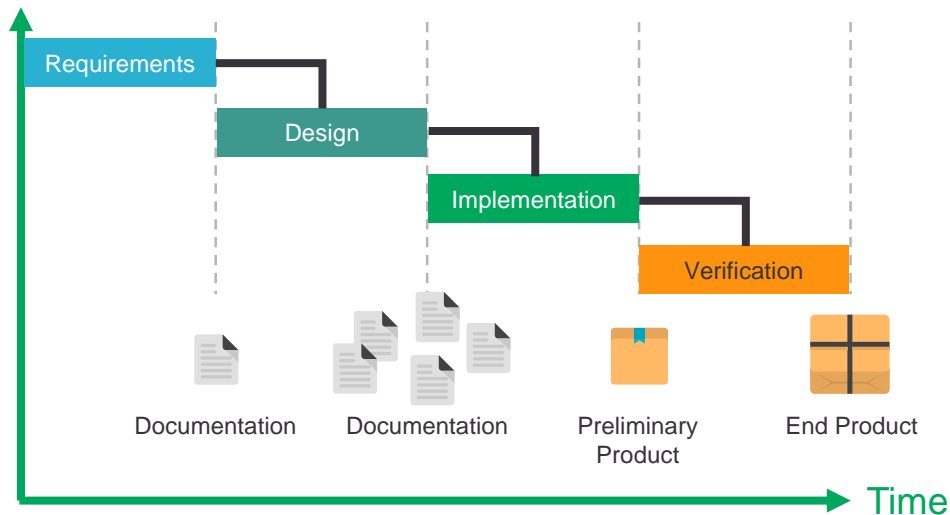
### Digitalization:

- Growing software content in engineered products
- Software development maturity a strategic advantage
- Market pressure to increase SW development velocity
- Agile, DevOps becoming the standard

# Waterfall vs. Agile Development

## Waterfall Model

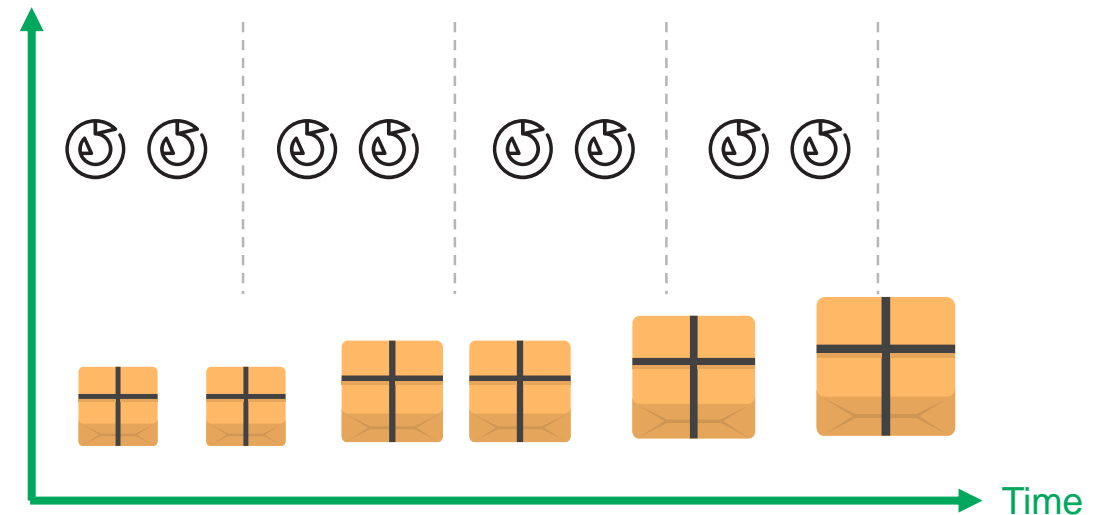
Sequential product development



- Linear
- Documentation based
- Deterministic

## Adaptive Methods

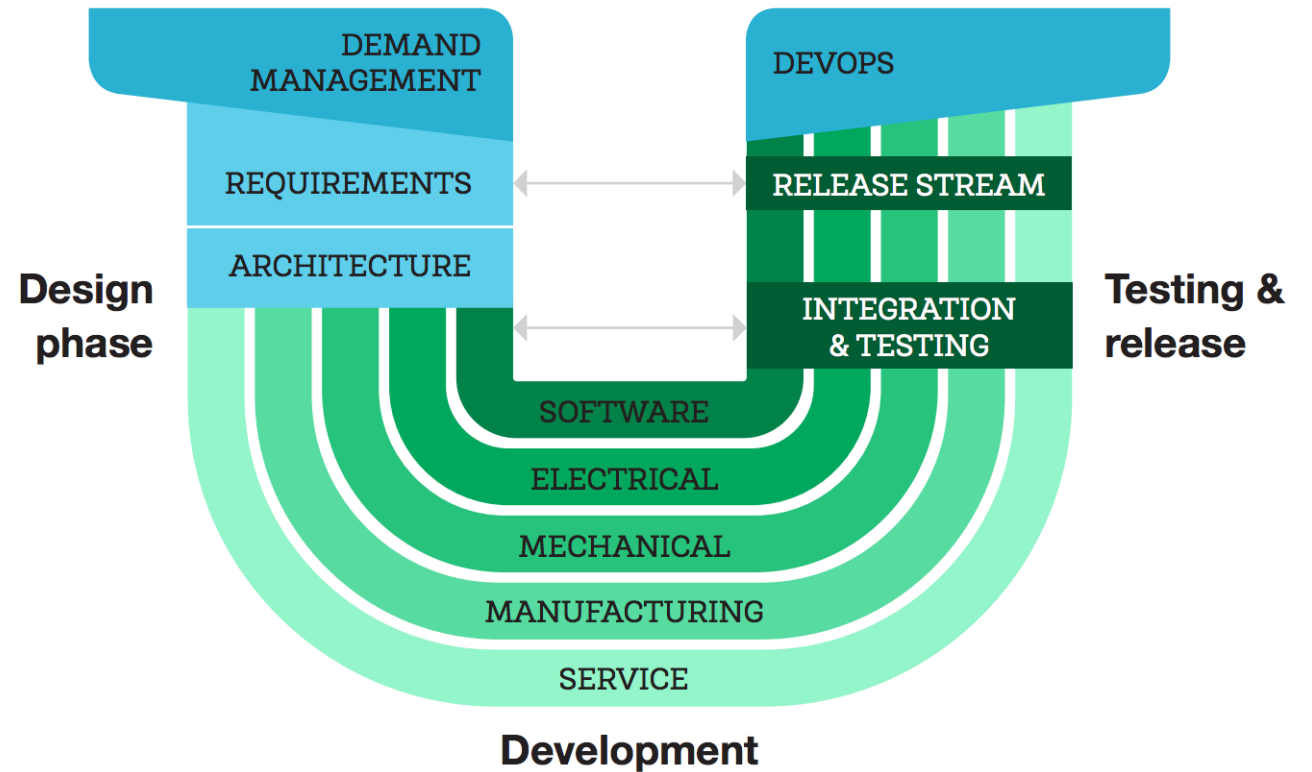
Incremental and iterative product development



- Adaptivity
- Learning
- Focus on max. customer value

# Requirements of regulatory standards

- Adequate planning (product design, engineering, development, and production)
- Traceability across the lifecycle
- Lifecycle-wide process control and risk management
- Documentation and validation of quality-focused procedures



## Adopting Agile and Continuous compliance

### Goals:

- Process control
- Automation
- Documentation

### Benefits:

- Reduce risks
- Reduce the time to market
- Increased efficiency and product quality
- Gapless traceability

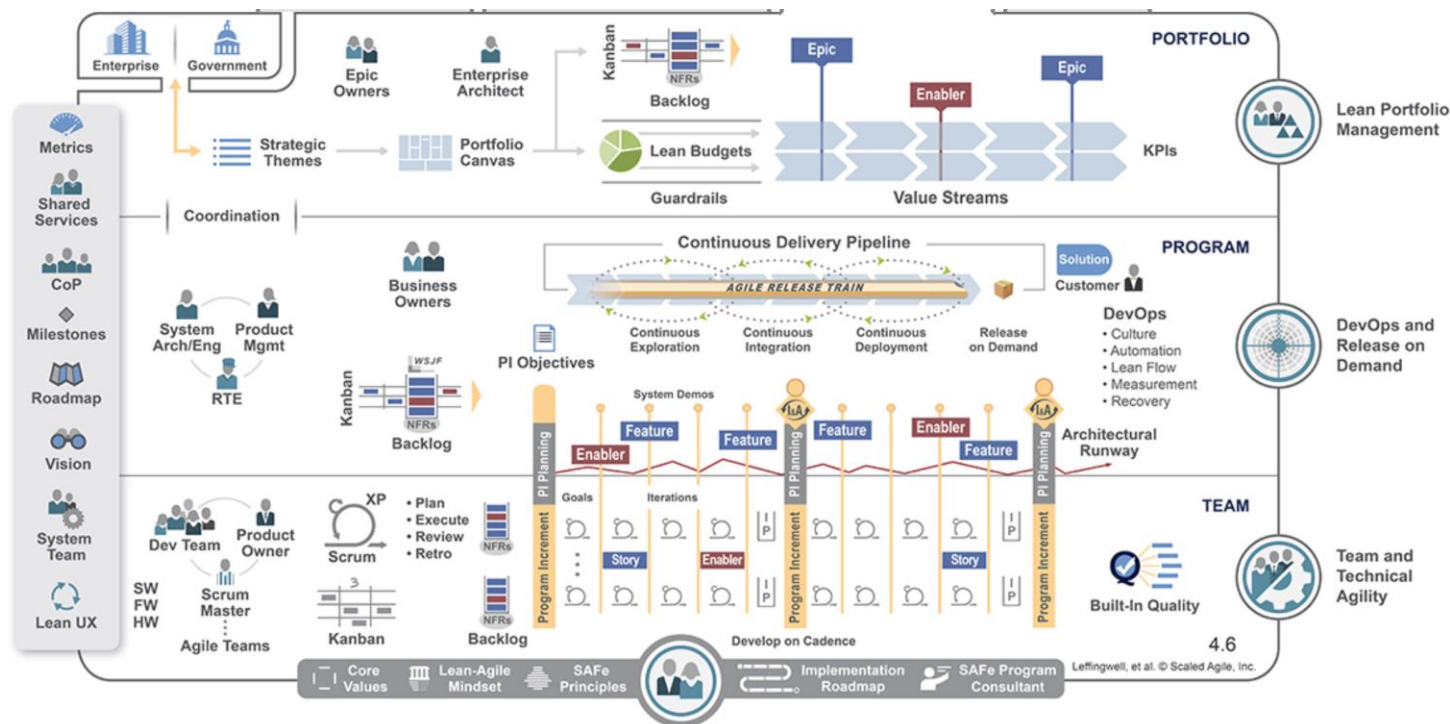
## Agile in regulated industries

- Challenge: manage risk at speed, accelerate dev while maintaining compliance
- Cornerstones of Agile/DevOps maturity in safety-critical development:
  - Map workflows to regulatory requirements
  - Automation to enforce defined processes, avoid manual error
  - Automated documentation for continuous auditability & compliance
- Agile/DevOps maturity supported by:
  - Agile scaling frameworks (DAD, SAFe®, and LeSS)
  - Integrated lifecycle management tooling (for process control, documentation, automation)

# SAFe® vs LeSS vs DAD

SAFe is prescriptive and comes with a lot of guidance

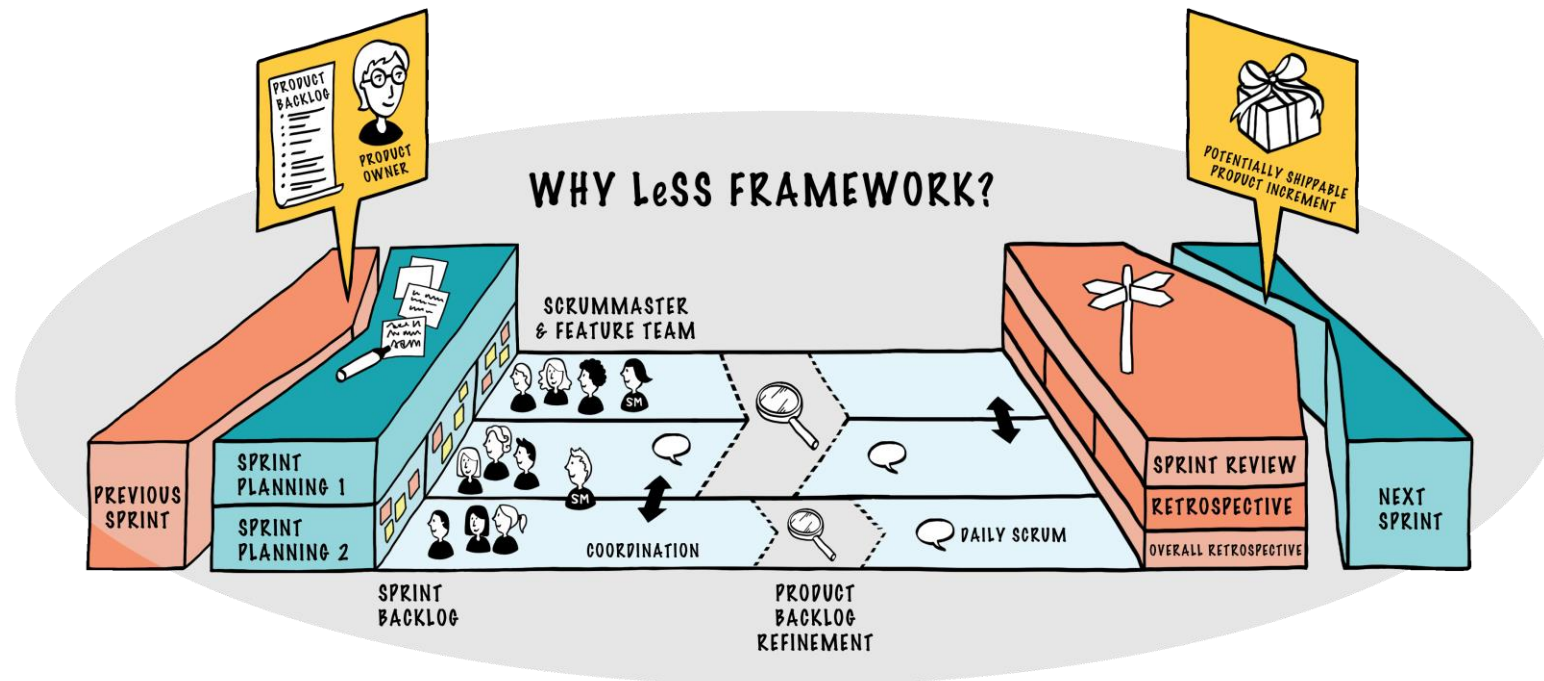
- Portfolio level – use lean principles to optimize value streams and prioritize epics
- Program level – works on Release Trains composed of 5 sprints and one for innovation
- Team level – techniques outlines are those used in Scrum, but even supports teams working with different methodologies



# SAFe® vs LeSS vs DAD

LeSS is not prescriptive, an elegant extension of Scrum

- Lean thinking, minimal waste, focus on what really needs to be done
- Feature-oriented, customer-centric teams
- One Product Backlog, one Product Owner, one potentially shippable product increment, one Sprint
- Transparency based on tangible done items



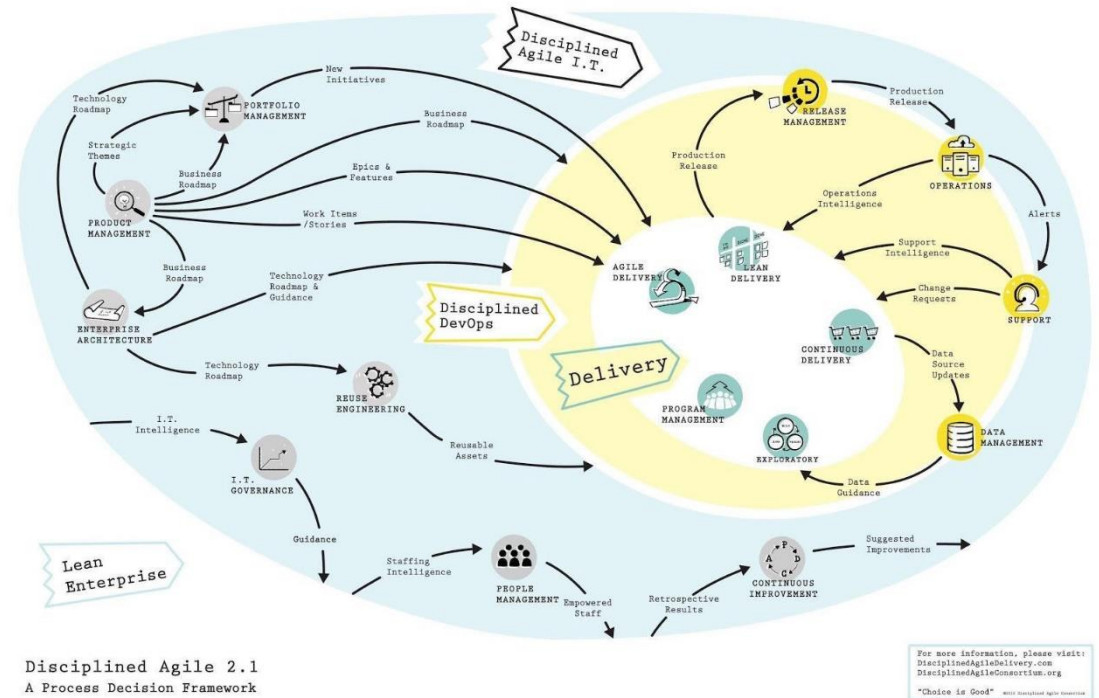
<http://less.works>



# SAFe® vs LeSS vs DAD

DAD is not prescriptive. It forces the teams to makes choices.

- Solution focused that include SW & HW
- Phases:
  - Inception – How do we start?
  - Construction – How do we produce a solution?
  - Transition – How do we deploy?
- Enterprise-aware – Teams follow common conventions

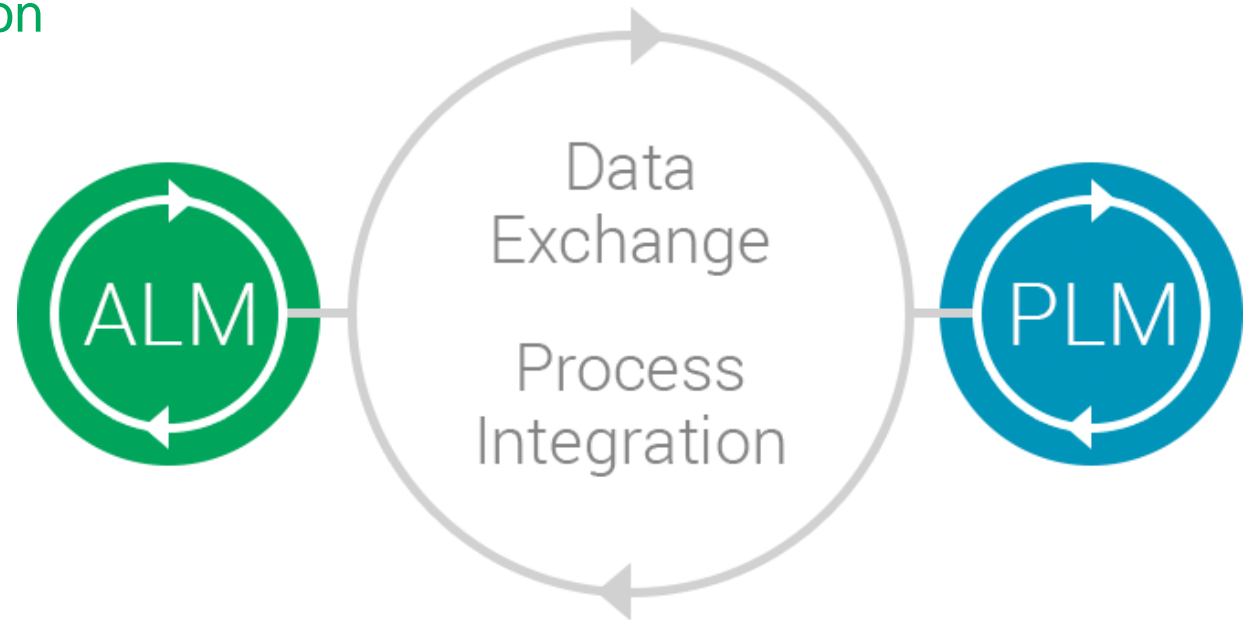


# Integrated Lifecycle Management

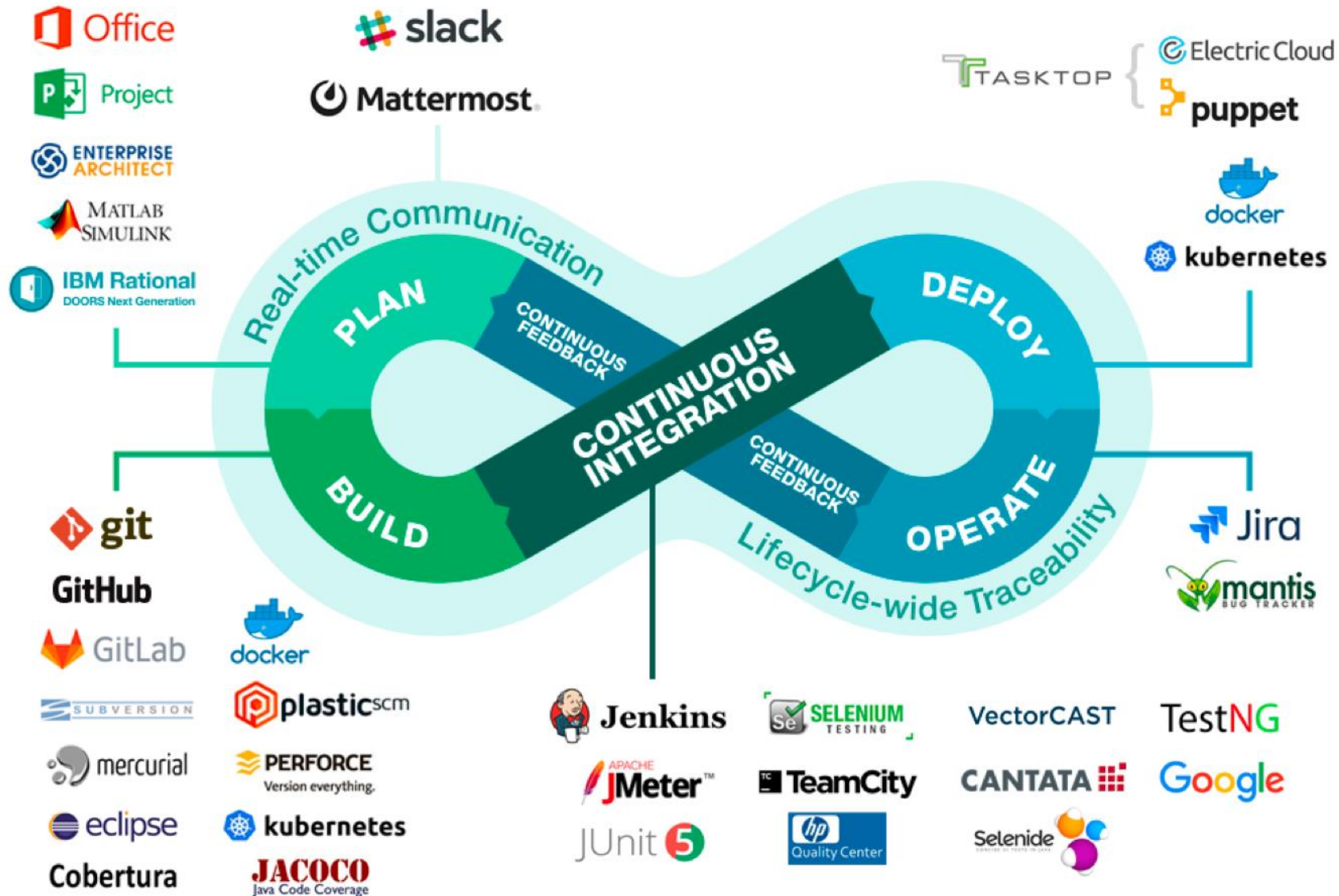
- Non-integrated approach: increased costs and risks
- PLM+ALM = integration of 1) processes 2) data
- Business Process Management (BPM) to connect processes & data

## Key requirements of ALM-PLM integration

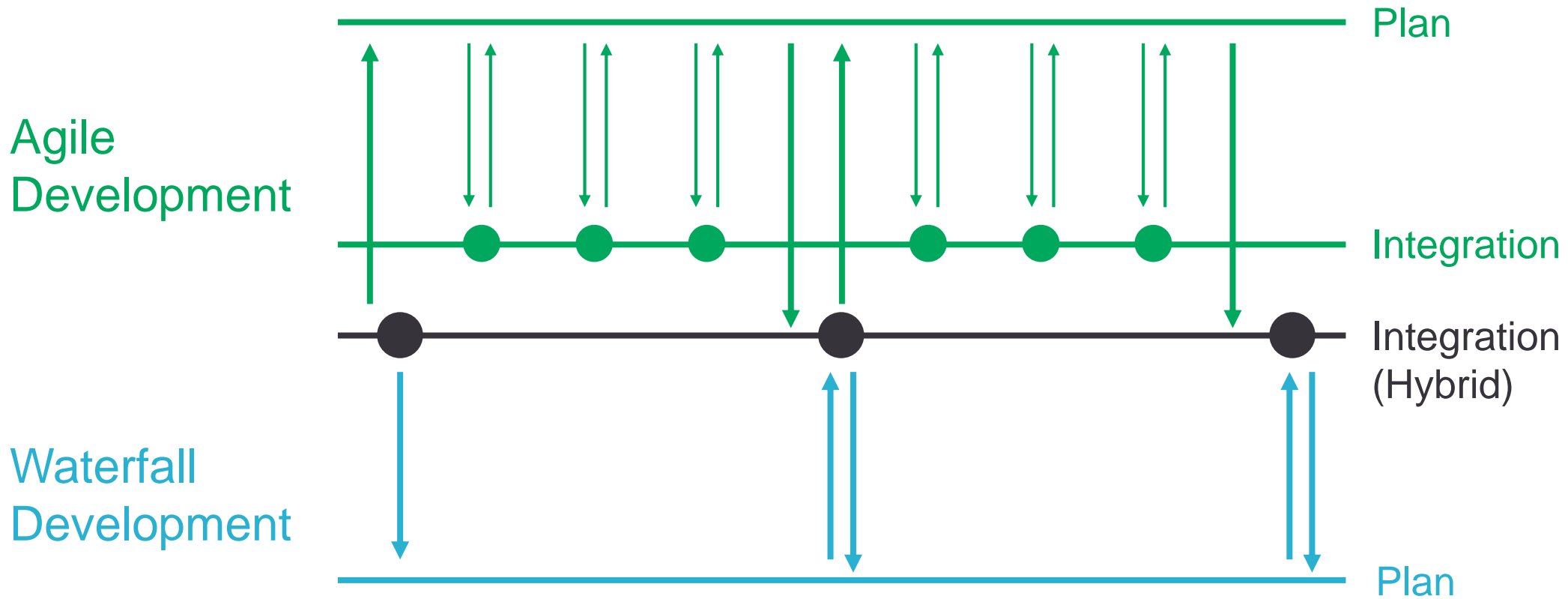
- Process & data integration
- Integrated hardware and software requirements (RM + BOM)
- Consistent defect and change management
- Tool-independent integration



# Integrated ALM from a DevOps perspective



# Integrated Lifecycle Management



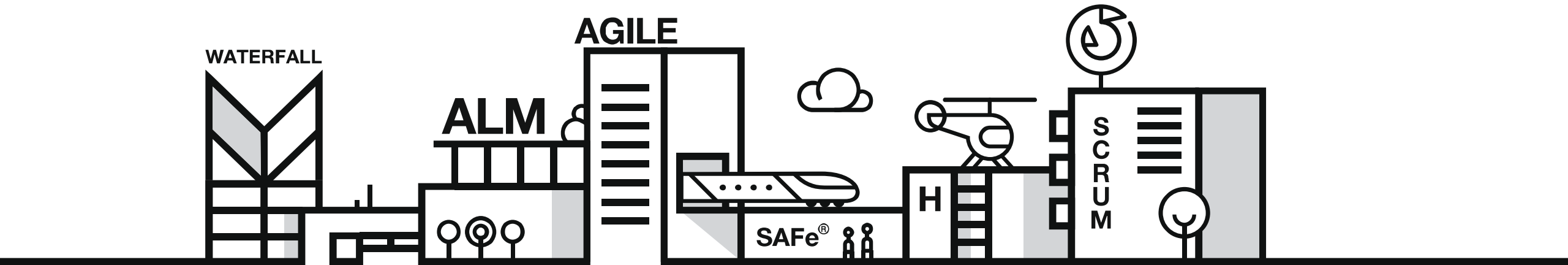
- Different mindset of Waterfall and Agile teams
- Software has to be integrated in to the product

## Key ALM Capabilities for Agile/DevOps maturity

Maturity checklist – Integrated ALM capabilities for Agile/DevOps:

- Integrated ALM and DevOps release management
- Variation, parameter, and configuration management
- Generate ISO-compatible reports
- Auditing
- Support Agile methods
- End-to-end traceability
- Multiple views: requirements, tests & QA, builds, deployments
- Supports single backlog across multiple teams
- Kanban workflow

Source: Ovum



Thanks for your attention!  
Any questions?