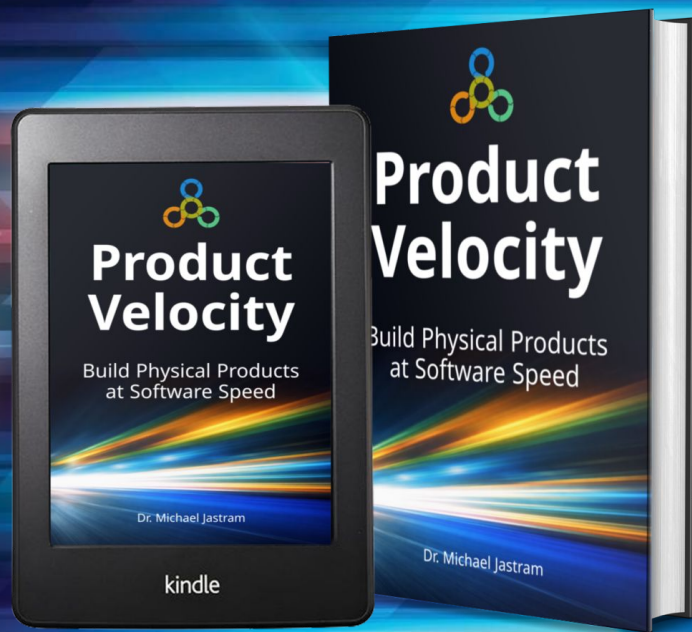


Product Velocity

Vier Prinzipien für radikal beschleunigte Produktentwicklung

Dr. Michael Jastram

29-Apr-2026 | **ReConf Vortrag**



Product Velocity weiter vertiefen

productvelocity.org bietet frei zugängliche Inhalte zur direkten Anwendung im Alltag.

Fallstudien zeigen konkrete Entscheidungen, deren Wirkung und messbare Ergebnisse.

Der Anti-Pattern Finder identifiziert typische Blockaden in wenigen Minuten.

Alle Inhalte lassen sich ohne Vorbereitung nutzen.

Jetzt starten

Viele Ressourcen sind Creative Commons Lizenziert

- productvelocity.org besuchen
- [Fallstudien](#) durchgehen und vergleichen
- [Anti-Pattern Finder](#) ausprobieren
- [Velocity Loop](#) nutzen

Dranbleiben

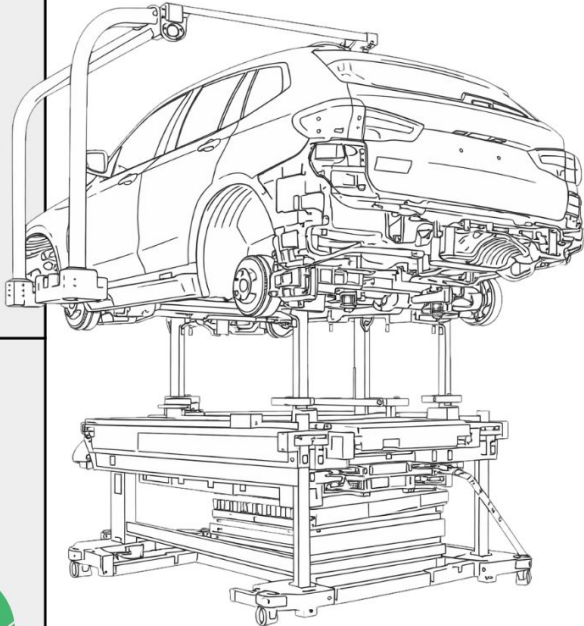
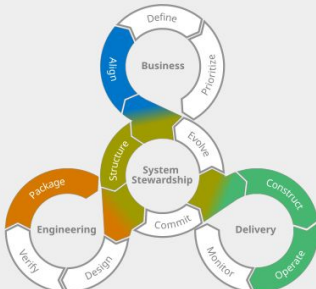
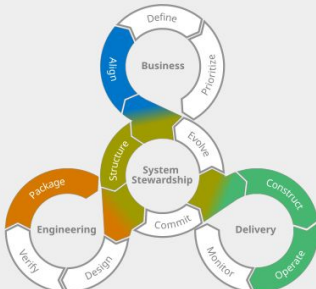
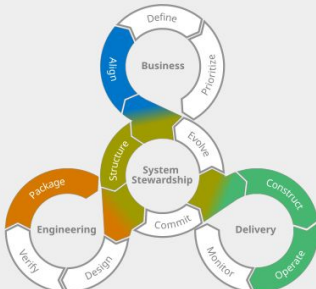
se-trends.de liefert regelmäßig neue Inhalte zu Systems Engineering und Product Velocity.

Der Newsletter fasst neue Artikel kompakt zusammen und bleibt fachlich.



Was ist Product Velocity

?

1		2		3		4		5																																															
productvelocity.org/case-studies/bmw/manufacturing																																																							
A		BMW: Hardware Modularity for Manufacturing								A																																													
Production lines allowed on demand acceleration of electric car production																																																							
B		How velocity is created: Applied in this case  Define & Align: Value Thinking  Structure & Scale: Architect for Flow  Build & Validate: Shift Left  Operate & Evolve: Accelerate  <tr><td colspan="2">C</td><td colspan="8"></td><td colspan="2">C</td></tr> <tr><td colspan="2">D</td><td colspan="8"> Where velocity emerges: Observed in this case  </td><td colspan="2">D</td></tr> <tr><td colspan="2">E</td><td colspan="8"> Product Velocity © 2026 Michael Jastram License: CC BY-NC 4.0 productvelocity.org </td><td colspan="2">E</td></tr> <tr><th colspan="2">1</th><th colspan="2">2</th><th colspan="2">3</th><th colspan="2">4</th><th colspan="2">5</th></tr>								C										C		D		Where velocity emerges: Observed in this case 								D		E		Product Velocity © 2026 Michael Jastram License: CC BY-NC 4.0 productvelocity.org								E		1		2		3		4		5	
C										C																																													
D		Where velocity emerges: Observed in this case 								D																																													
E		Product Velocity © 2026 Michael Jastram License: CC BY-NC 4.0 productvelocity.org								E																																													
1		2		3		4		5																																															



“Normally, the car industry will take four years to develop a new car. At BYD, our standard is 18 months.”

Stella Li,
Executive vice president of BYD and
CEO of BYD Americas

Not just cars

SpaceX Raptor

History Architecture

First Testrun

- 2016: Raptor 1
- 2021: Raptor 2
- 2023: Raptor 3

Design

2024: Every 2 days:

- Redesign
- Build
- Test



Payload Insights	Raptor 1	Raptor 2	Raptor 3
Thrust	185tf	230tf	280tf
Chamber pressure	250 bar	300 bar	350 bar
Specific impule	350s	347s	350s
Engine mass	2,080kg	1,630kg	1,525kg
Engine + side commodities mass	3,630kg	2,875kg	1,720kg



Not just Rocket Science:

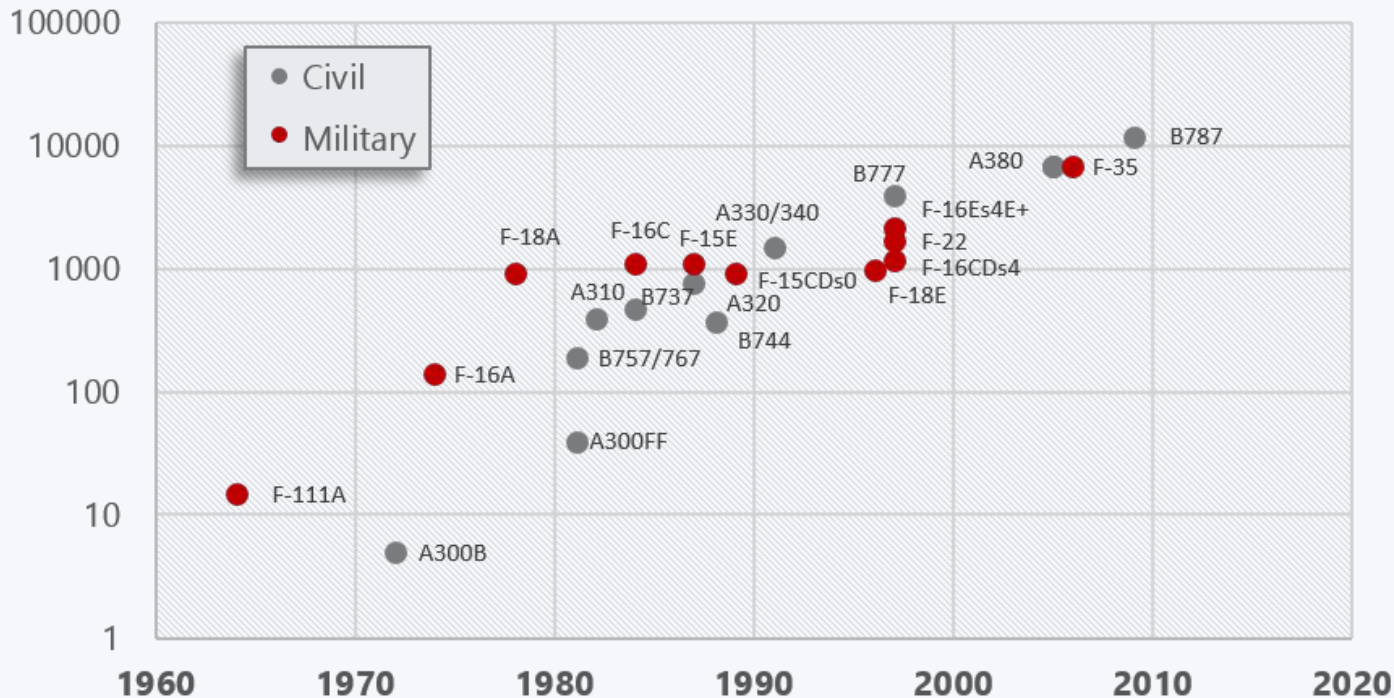
Opal Nugget Ice Maker

GE: In 4 Monaten von
Konzept bis Verkauf



Die Software-Welt hatte ähnliche Probleme

*Exhibit 1: Aircraft Software Volume Growth
(airborne system software, K SLOCs)*

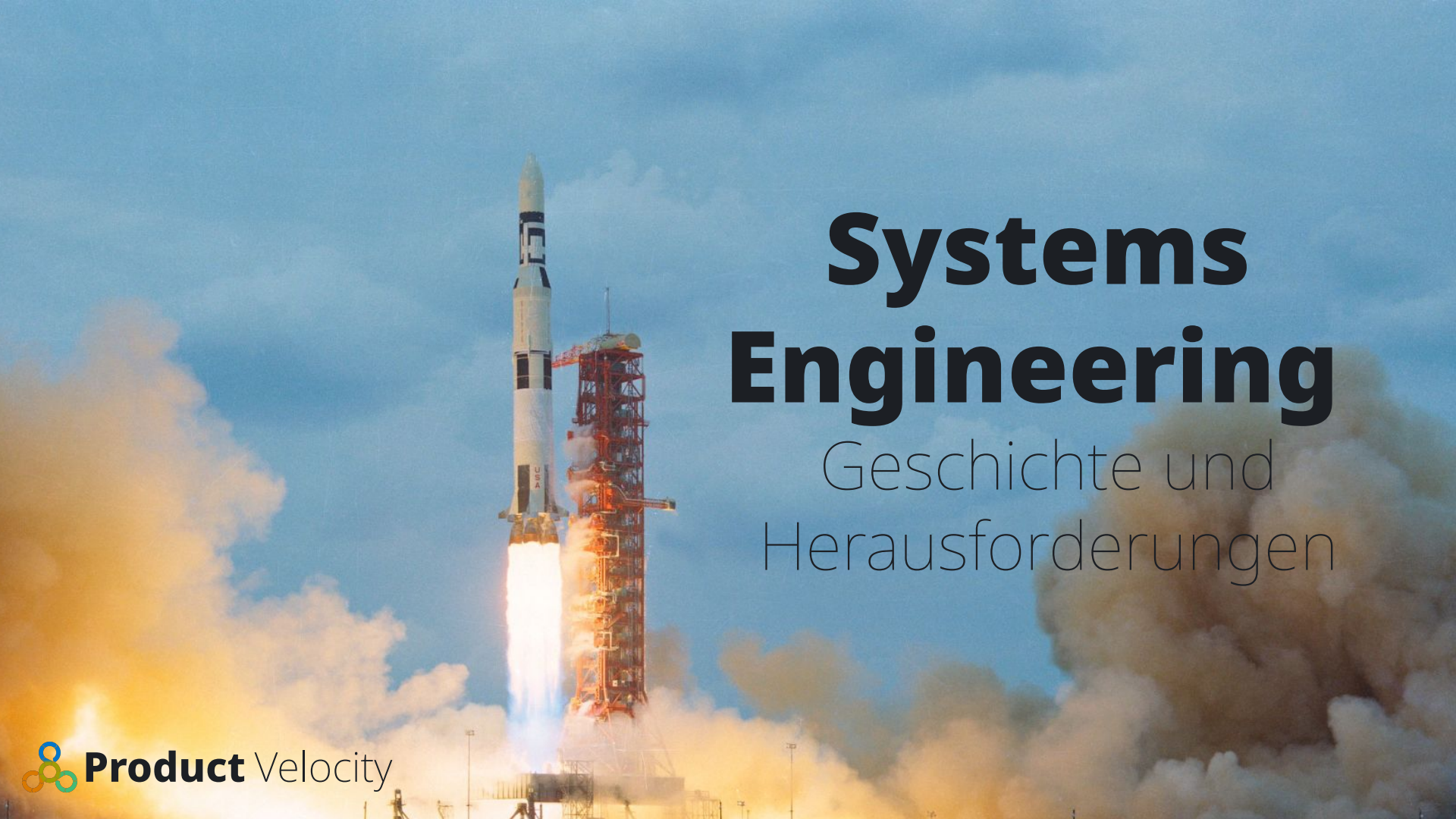


Product Velocity in der Softwareentwicklung

- Agile Development
- DevOps
- CI/CD
- Container
- Openness

Wiederholt
sich das für
physische
Produkte?



A photograph of a Space Shuttle Columbia launching from the launch pad. The shuttle is white with black and blue markings, including the NASA logo and 'USA'. It is surrounded by a large plume of white smoke and orange fire from the engines. The launch pad structure is visible to the right of the shuttle. The sky is blue with some clouds.

Systems Engineering

Geschichte und
Herausforderungen

Die Herausforderung

1970 hierarchisch dekomponiert

Komponenten dienen der Lenkfunktion und diese kann ausschließlich durch diese Komponenten gebildet werden.

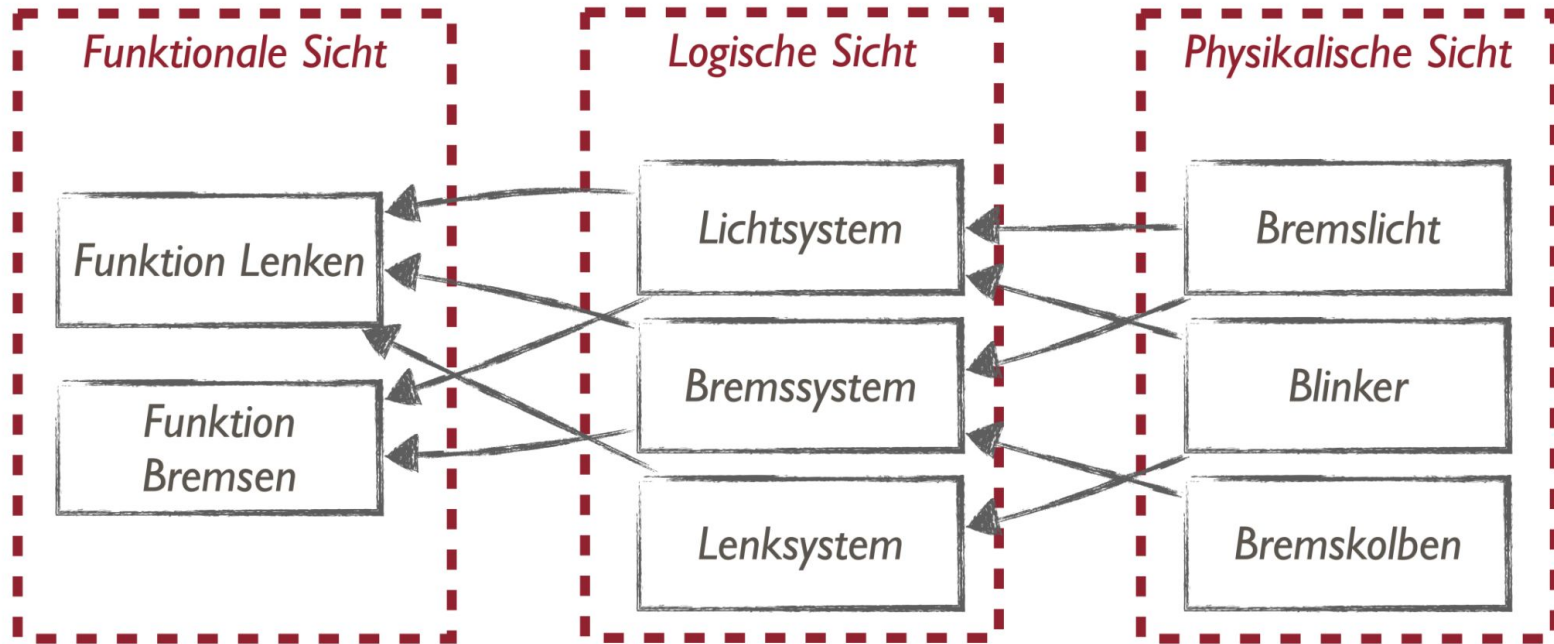


2020 orthogonal vernetzt

Komponenten dienen mehreren Funktionen und diese zieht sich durch verschiedenste Komponenten.

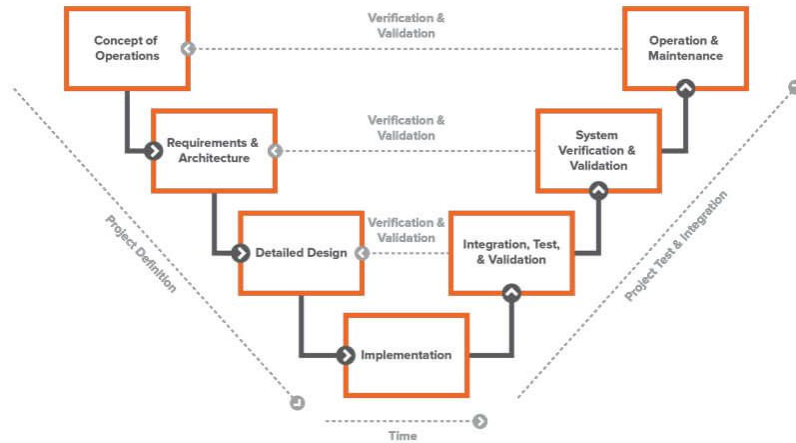
Lenksystem	Lenkwinkel an Vorderrädern einstellen
Bremssystem	Innenräder verzögern
Lichtsystem	Kurvenlicht
Sitz	Sitzwangen ausfahren (Fahrer stabilisieren)
Fahrwerk	Fahrwerk seitlich straffen (Neigung reduzieren)
Hinterachse	Hinterachslenkung
...	

Die Lösung: Vernetzte Schichten

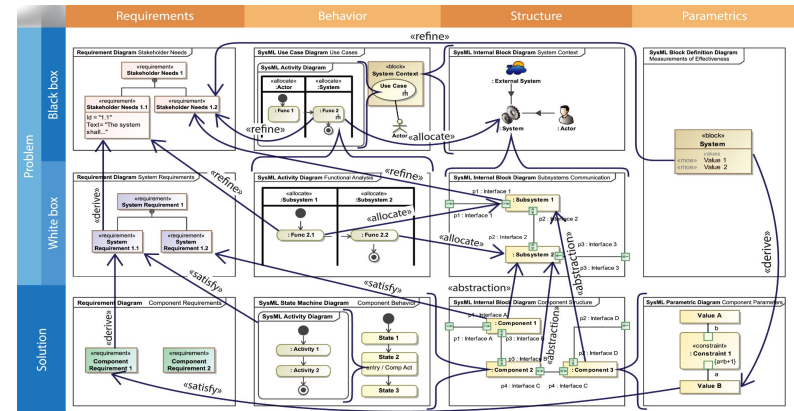


Systems Engineering: V-Modell & MBSE

Systems Engineering



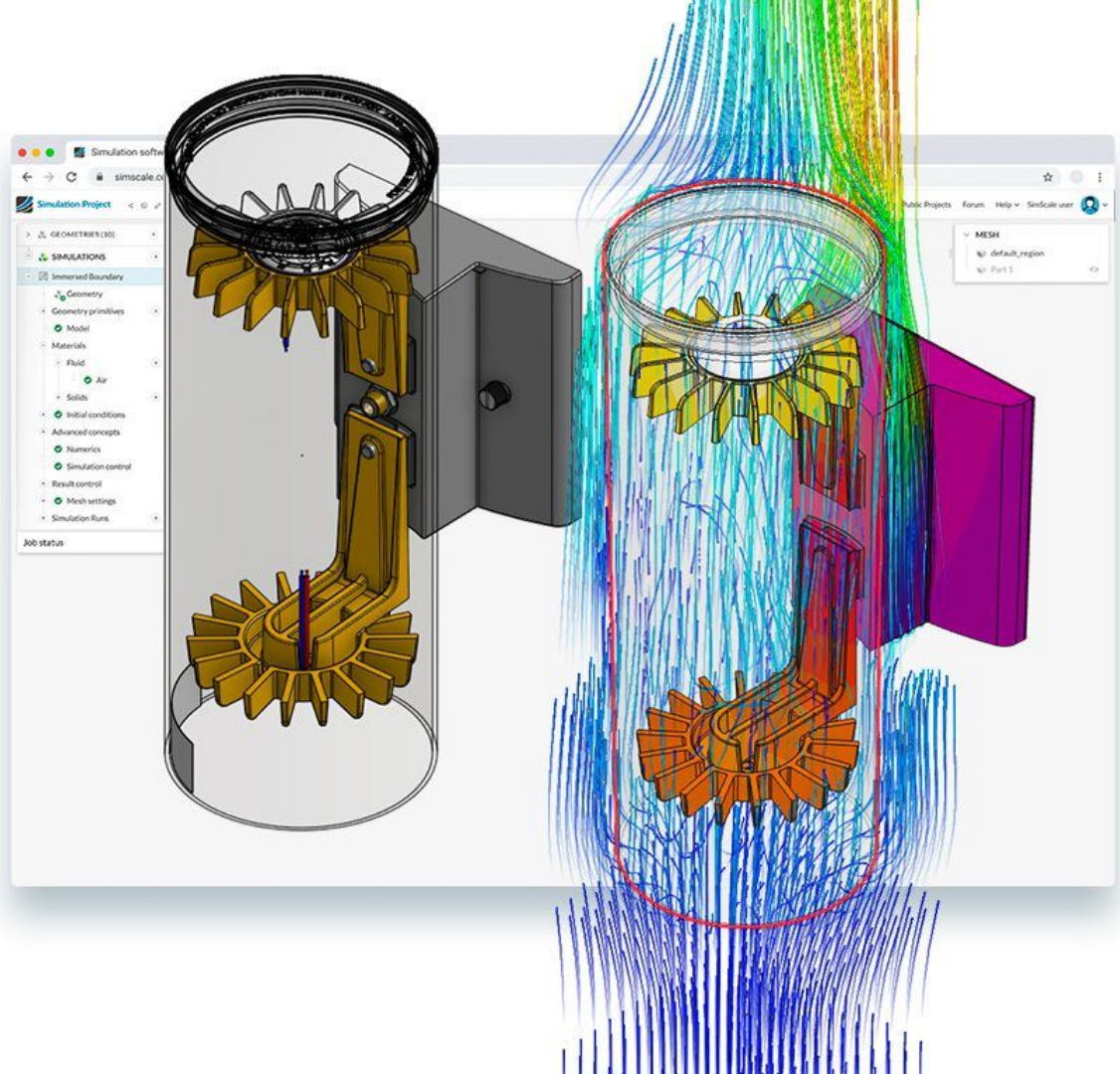
Model Based Systems Engineering (MBSE)



Dassault MagicGrid

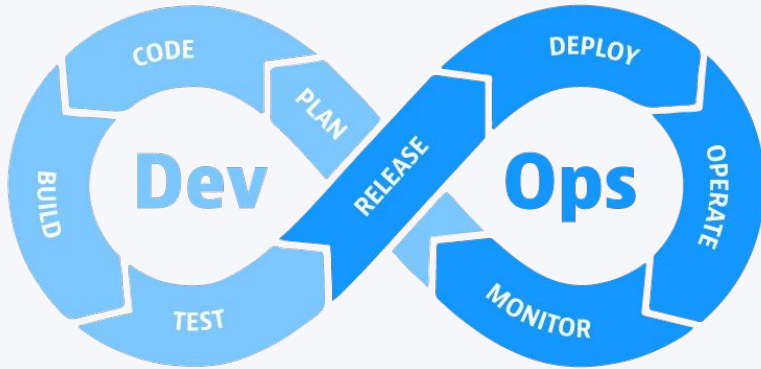
Modelle

Enabler für den
"Digital Twin"

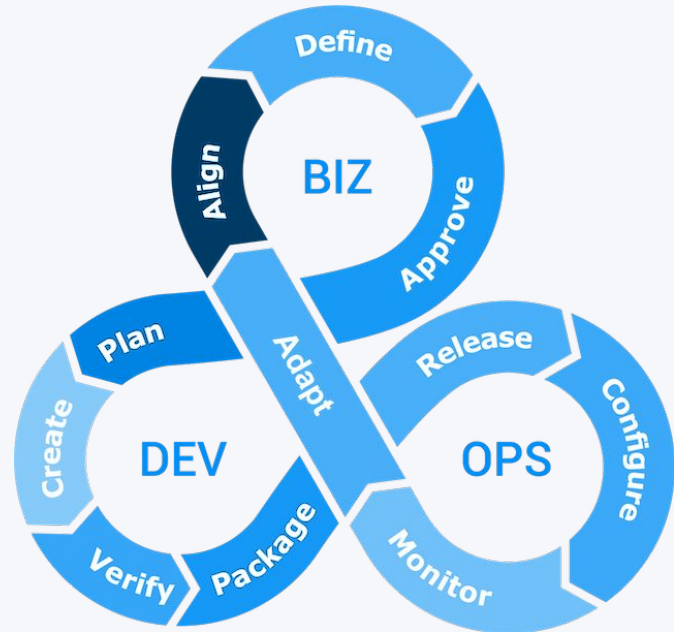


Betriebsmodell der Softwareentwicklung

DevOps Loop

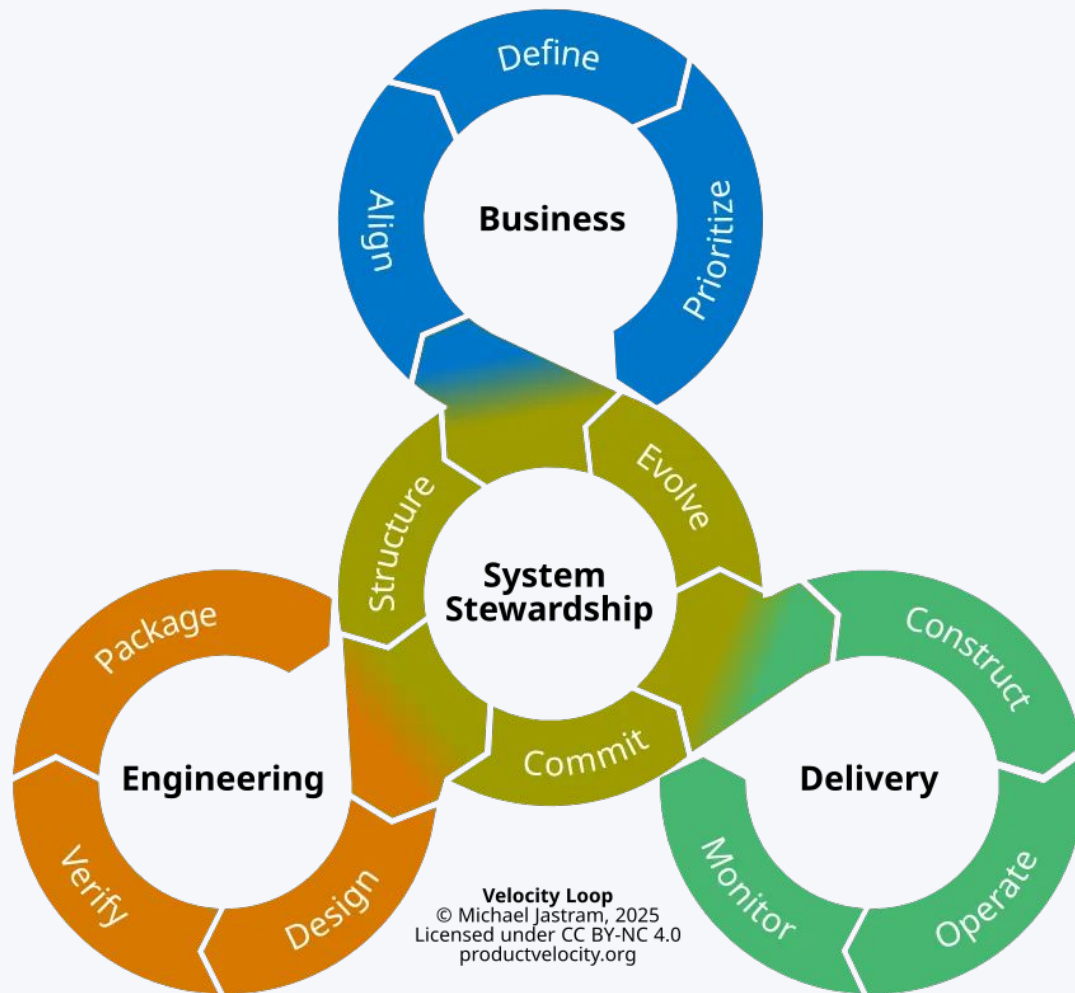


BizDevOps Loop



Velocity Loop

DevOps Loop für
cyberphysikalische
Systeme



Etwas Theorie: Minimierung des wirtschaftlichen Risikos

Minimizing Economic Exposure

$$\text{Exposure} = \int U(t) dt$$

U(t)

Uncertainty over time

$$V = -\frac{dU}{dt}$$

V

Velocity at which U decreases

$$V \sim \frac{\Delta U_r (1 - \lambda)}{L \cdot C}$$

L

End-to-End Latency

C

Complexity (de Weck)

ΔU_r

Relevant U removed

λ

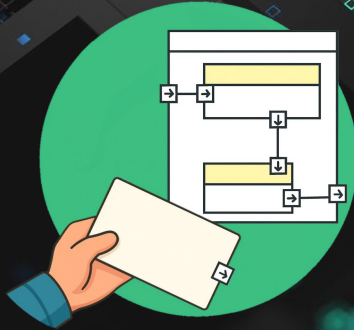
Leakage

4 Principles for Product Velocity

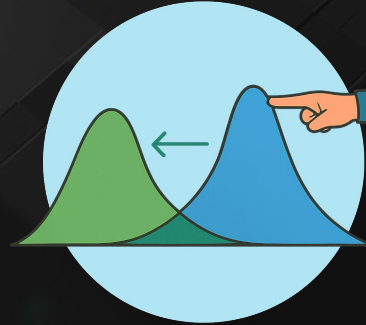
Define & Align:
Value Thinking



Design & Build:
Architect for Flow



Integrate & Validate:
Shift Left



Operate and Evolve:
Accelerate





1. Prinzip:

Value Thinking

1. Prinzip

Value Thinking



Praktiken

1. Schaffe eine Kultur, die Value Flow ermöglicht

Kulturwandel

- Schaffe eine gemeinsame Sprache
- Schaffe eine Lernkultur
- Formuliere die gewünschte Zukunft
- Schaffe (psychologische) Sicherheit
- Führungskräfte müssen authentisch sein
- Führungskräfte prägen die Kultur
- Verhaltensänderungen führen zu Kulturveränderungen
- Erstellen einen Fahrplan für Veränderung



1. Prinzip

Value Thinking

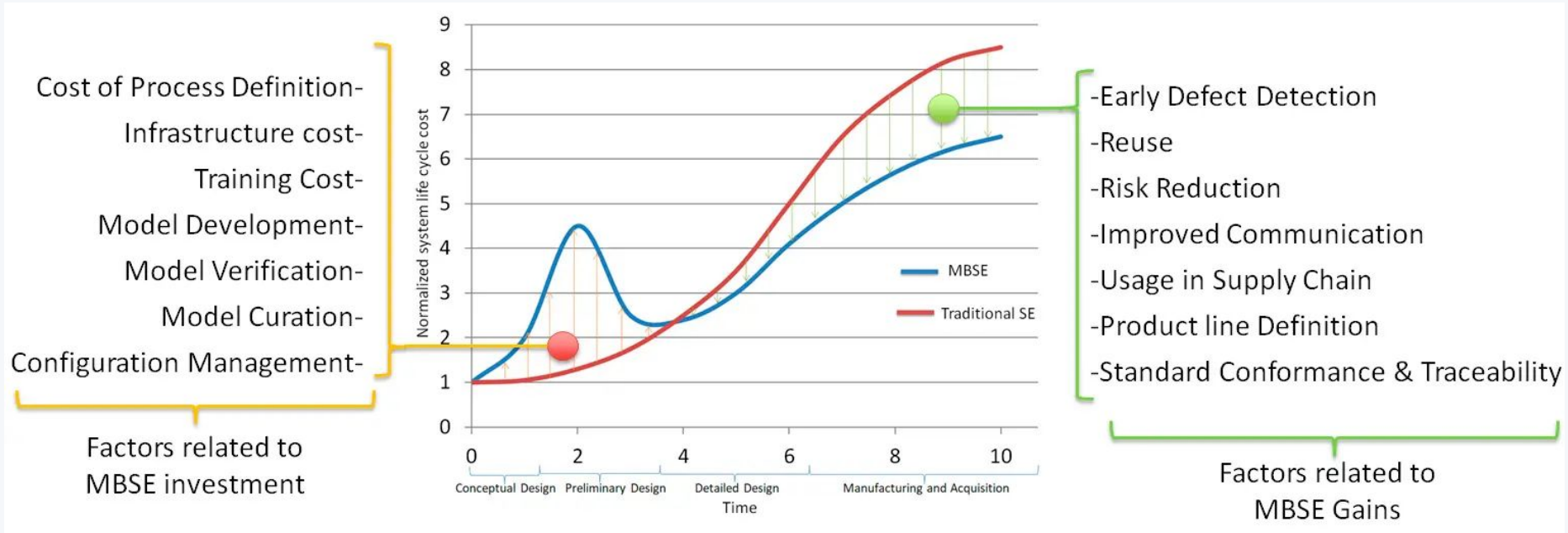


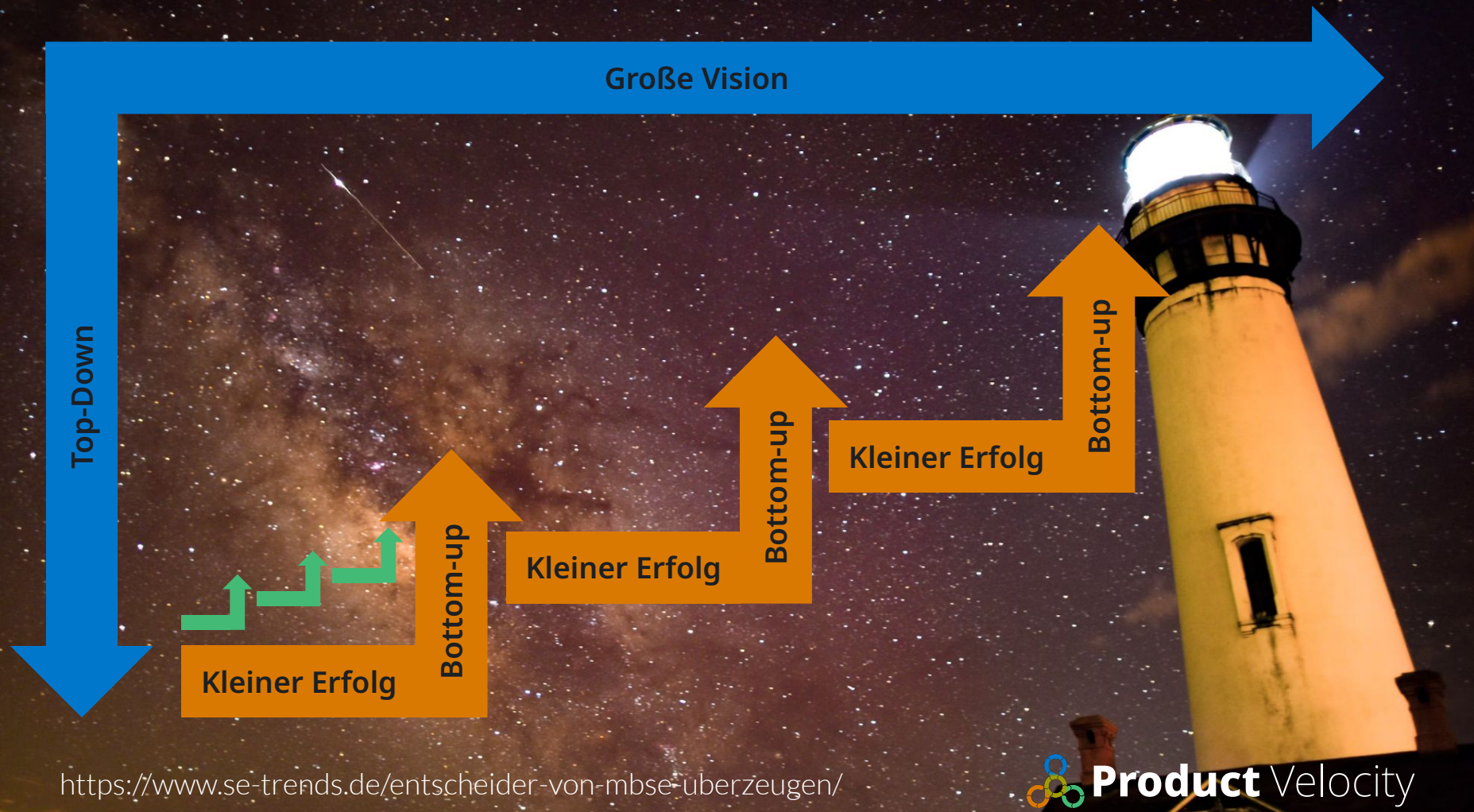
Praktiken

1. Schaffe eine Kultur, die Flow ermöglicht

2. **Zeige regelmäßig Erfolg**

MBSE-Dilemma

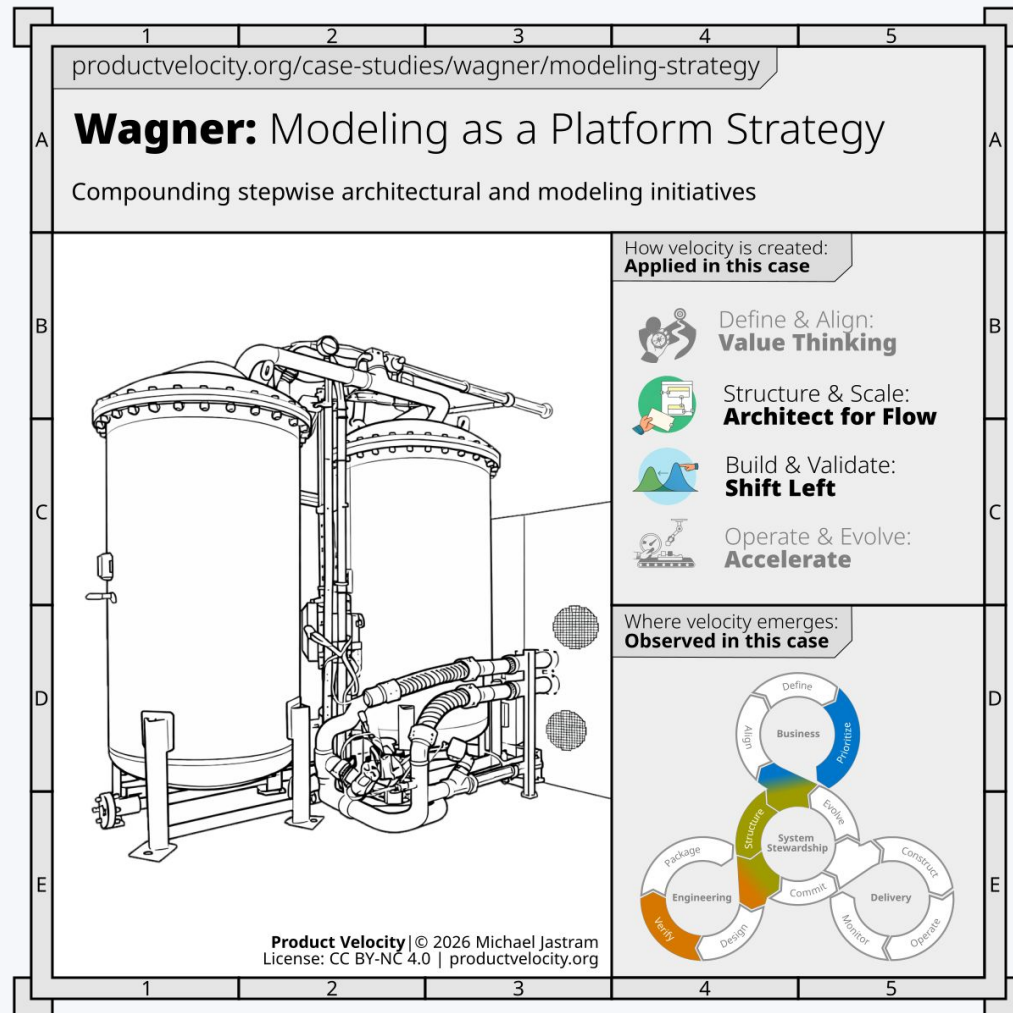




TÜV-Abnahme auf Knopfdruck

Kleine Schritte

- Spezifikation über MBSE
- Codegenerierung aus Modellen
- Zertifizierte Werkzeugkette
- Testraum
- Testraum-Zertifizierung



1. Prinzip

Value Thinking



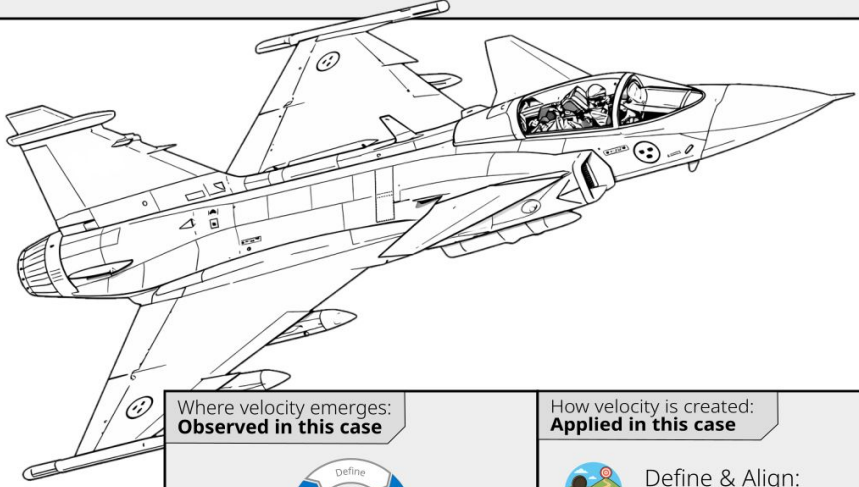
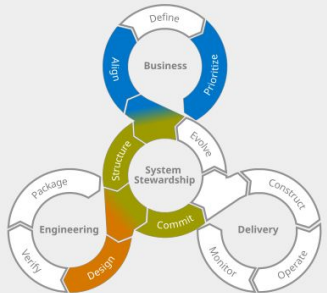
Praktiken

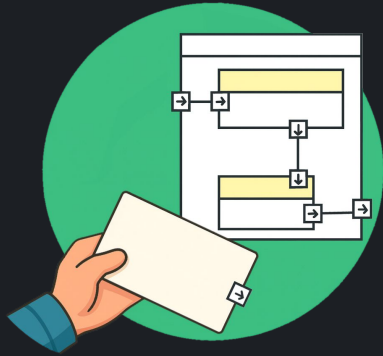
1. Schaffe eine Kultur, die Flow ermöglicht
2. Zeige regelmäßig Erfolg

3. **Optimiere Werteströme**

Blocker innerhalb von einem Tag bis auf Vorstandsebene eskalieren

- 7:30 am Daily Scrum at the individual contributor/team level
- 7:45 am Scrum of Scrum
- 7:45 am Scrum of Scrum of Scrum
- 8:15 am Scrum of Scrum of Scrum of Scrum
- 8:30 am Executive Action Team

	1	2	3	4	5
A	productvelocity.org/case-studies/saab/escalation-chain Saab AB: Escalating Blockers in 60 Minutes By institutionalizing daily escalation, Saab eliminated decision latency				
B					
C					
D	Where velocity emerges: Observed in this case		How velocity is created: Applied in this case		
E			 Define & Align: Value Thinking  Design & Build: Architect for Flow  Integrate & Validate: Shift Left  Operate & Evolve: Accelerate		
	1	2	3	4	5

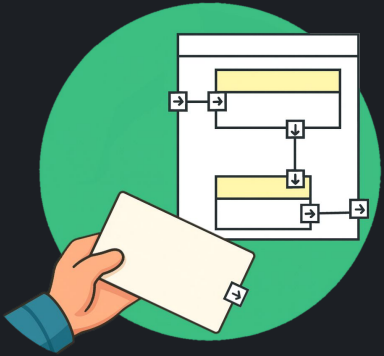


2. Prinzip:

Architect for Flow

2. Prinzip

Architect
for Flow

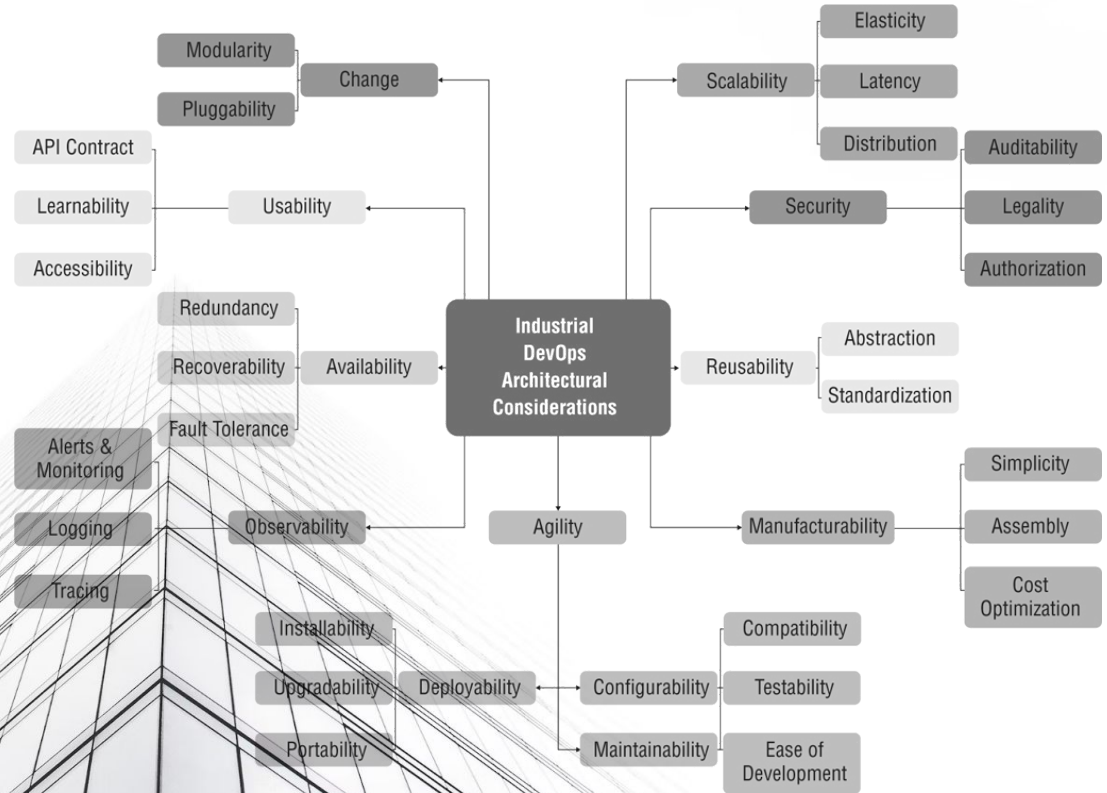


Praktiken

1. Berücksichtige alle relevanten Architekturen

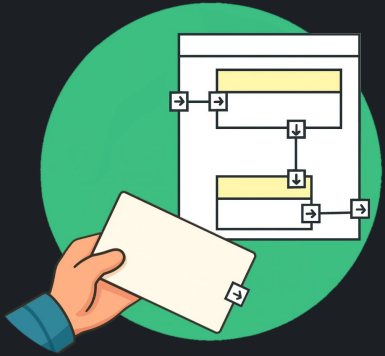
Was ist Architektur?

“Aufbau, Funktion und Management von Systemen”



2. Prinzip

Architect for Flow



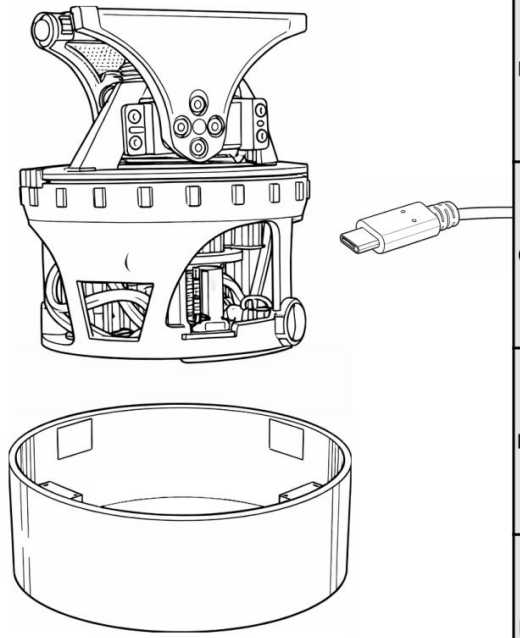
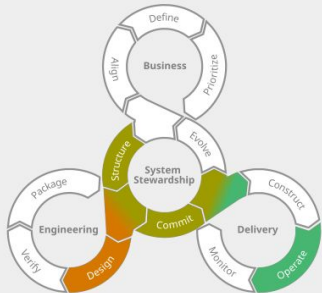
Praktiken

1. Berücksichtige alle relevanten Architekturen

2. Schnittstellen und klare Systemgrenzen

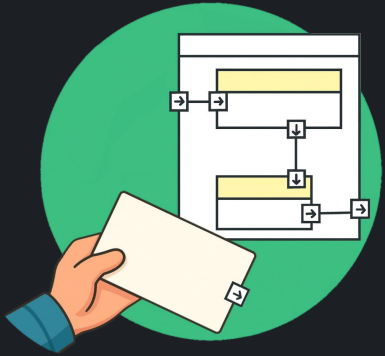
Universelle Schnittstelle:

Entwickelt für Zuverlässigkeit und Langlebigkeit

	1	2	3	4	5
	productvelocity.org/case-studies/ameru/universal-interface				
A	Ameru: Smart Bin Universal Interface Enabling tool-free field replacement through stable, future-proof interfaces				
B	How velocity is created: Applied in this case  Define & Align: Value Thinking  Design & Build: Architect for Flow  Integrate & Validate: Shift Left  Operate & Evolve: Accelerate				
C	Where velocity emerges: Observed in this case 				
D	Product Velocity © 2026 Michael Jastram License: CC BY-NC 4.0 productvelocity.org				
E					

2. Prinzip

Architect for Flow



Praktiken

1. Berücksichtige alle relevanten Architekturen
2. Schnittstellen und klare Systemgrenzen

3. Baue eine Plattform

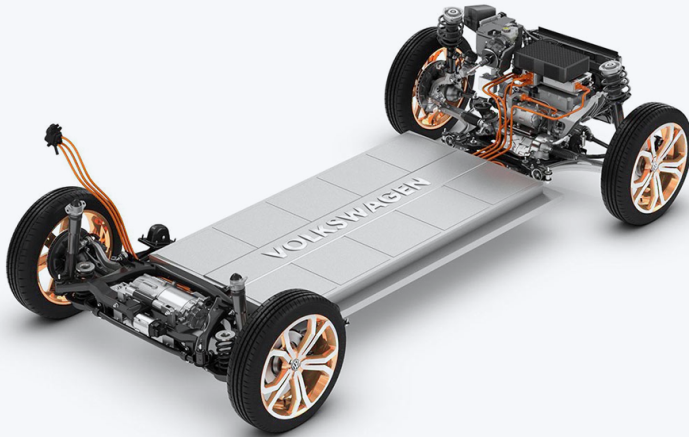
Plattform ermöglicht Veränderung und Geschwindigkeit

Smartphone-Betriebssysteme
entkoppeln Apps von der
Hardware über ein sicheres
Betriebssystem und stabile APIs.

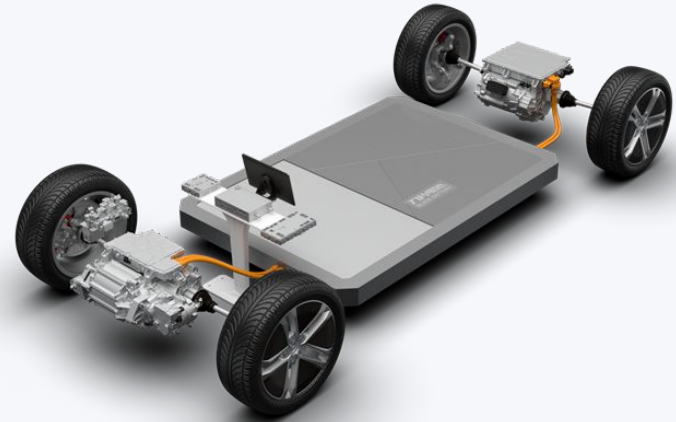


Der Trend geht zur Plattform-Architektur

VW MEB



BYD e-Platform 3





3. Prinzip:

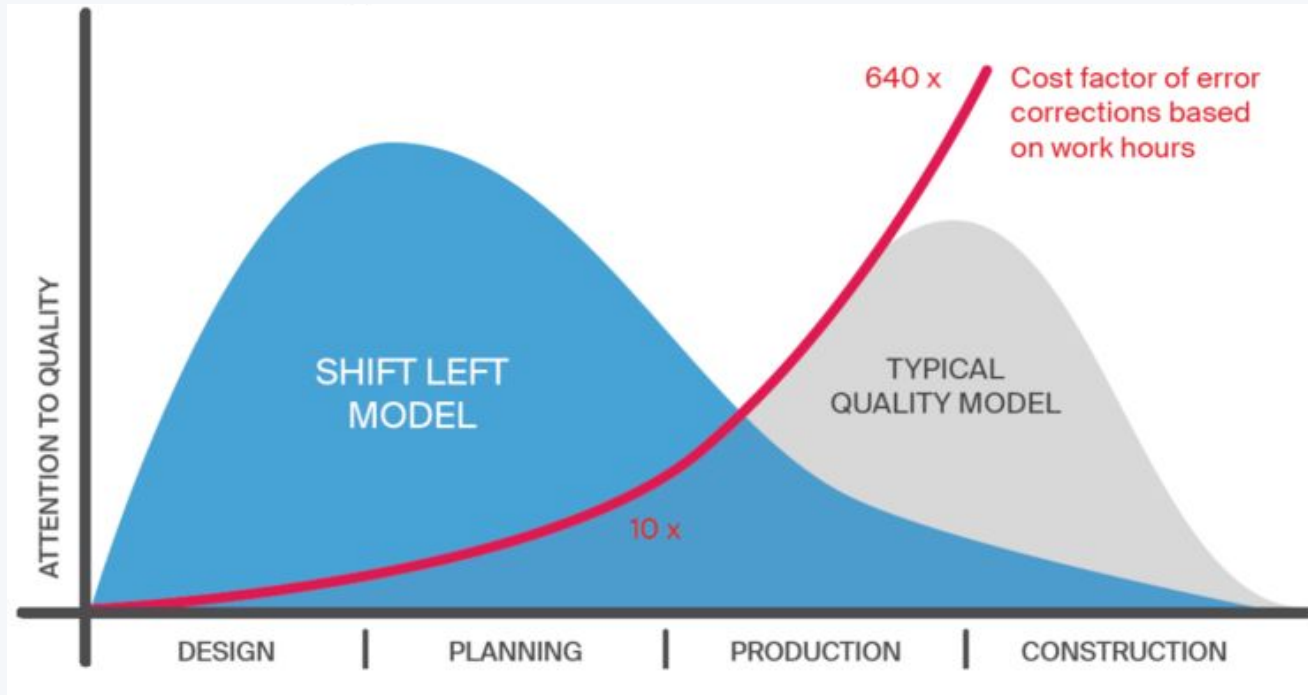
Shift Left

3. Prinzip

Shift Left



3. Prinzip: Shift Left



3. Prinzip

Shift Left



Praktiken

- 1. Mache Wissen früh sichtbar und prüfbar**



3. Prinzip: Daten sind das Fundament

Entwicklung selbstlandender Booster: 2013–2016



3. Prinzip

Shift Left



Praktiken

1. Mache Wissen früh sichtbar und prüfbar
- 2. Mache alles menschen- und maschinenlesbar**

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: example-deployment
  labels:
    app: example-app
spec:
  replicas: 3
  selector:
    matchLabels:
      app: example-app
  template:
    metadata:
      labels:
        app: example-app
    spec:
      containers:
        - name: example-container
          image: example-image:latest
          ports:
            - containerPort: 80
```

Softwareentwicklung: Infrastruktur ist interoperabel

- Quellcode
- Build-Skripte
- Konfigurationen
- Abhängigkeiten / Paketspezifikationen
- Infrastruktur-Beschreibungen
- CI/CD-Pipelines
- Testartefakte
- API-Spezifikationen
- Datenmodelle / Schemas
- Dokumentationsartefakte
- Tracing / Monitoring-Konfigurationen
- Container-Definitionen
- Security-Artefakte

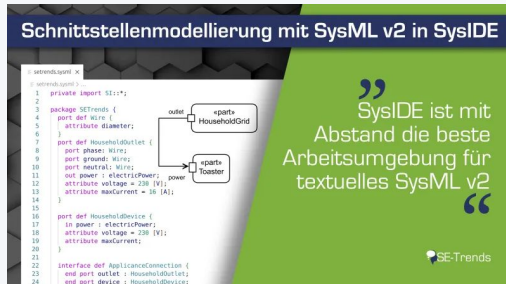
SysIDE: Model as Code

SysML v2 mit SysIDE:

- VS Code Integration
- Text based SysML

Sysand

- Package Manager
- Open Source



„SysIDE ist mit Abstand die beste Arbeitsumgebung für textuelles SysML v2“

SE-Trends

```
1 import syside
2
3
4 def main() -> None:
5     # Create a SysMLv2 document in memory.
6     model_document = syside.Document.create_st(
7         syside.DocumentOptions(
8             url=syside.Url("memory://example_model.sysml"), language="sysml"
9         )
10    )
11
12    with model_document.lock() as model:
13        # The document corresponds to a single root namespace.
14        root_namespace = model.root_node
15
16        membership, package = root_namespace.children.append(
```

```
1 package Requirements {
2     requirement def 'RQ-01' {
3         doc
4         /* The fridge shall provide adjustable shelves to accommodate a variety of food sizes and configurations. */
5     }
6     requirement def 'RQ-02' {
7         doc
8         /* The freezer compartment shall maintain a temp
9     }
10    requirement def 'RQ-03' {
11        doc
12        /* The fridge shall meet or exceed Energy Star e
13    }
14    requirement def 'RQ-04' {
15        doc
16        /* The fridge shall have an automatic defrosting
17    }
18    requirement def 'RQ-05' {
19        doc
20        /* The fridge doors shall have magnetic gaskets
21    }
22 }
```

```
1 import syside
2
3
4 def main() -> None:
5     model, diagnostics = syside.load_model(paths=["./example_model.sysml"])
6
7     assert not any(diagnostics.all(), f"Errors in model found: {list(diagnostics.all)}")
8
9     for doc in model.user_docs:
10         with doc.lock() as locked:
11             print(f"Model sexp {syside.decode_path(locked.url):}")
12             print(syside.sexp(locked.root_node))
13
14
15 if __name__ == "__main__":
16     main()
17
```

3. Prinzip

Shift Left



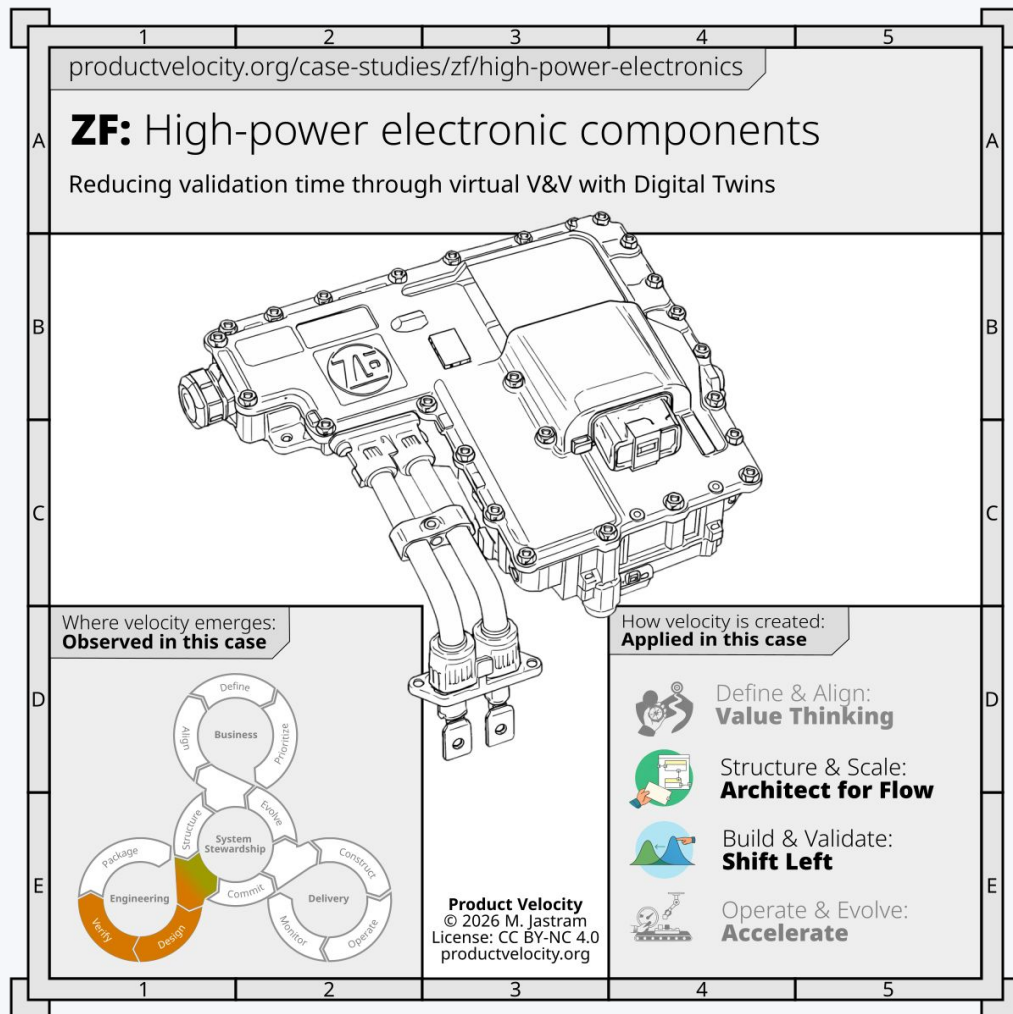
Praktiken

1. Mache Wissen früh sichtbar und prüfbar
2. Mache alles menschen- und maschinenlesbar

3. Automatisiere & Simuliere

Hochenergie-Halbleiter

Validierungszeit von 12 auf
2 Monate reduziert





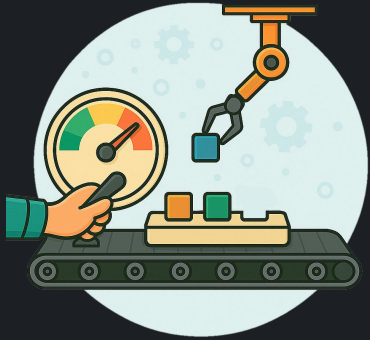
4. Prinzip:

Accelerate



4. Prinzip

Accelerate



Praktiken

1. Bleib lauffähig und integriere regelmäßig

SpaceX Starship

Testflug 10



ars TECHNICA

AI · BIZ & IT · CARS · CULTURE · GAMING · HEALTH · POLICY · SCIENCE · SECURITY · SPACE · TECH · **FORUM** · **SUBSCRIBE** · 🔍 · SIGN IN

FLIGHT 11

SpaceX's lesson from last Starship flight? "We need to seal the tiles."

SpaceX is targeting an orbital Starship flight with a next-gen vehicle next year.

STEPHEN CLARK · SEP 9, 2025 8:43 PM · 153

A smaller version of the SpaceX Starship rocket in flight, similar to the main image, but set against a blue sky with clouds and a dark ocean below.

→ SpaceX's 10th Starship test flight ended with an on-target splashdown in the Indian Ocean. Credit: SpaceX



It has been two weeks since SpaceX's last Starship test flight, and engineers have diagnosed issues with its heat shield, identified improvements, and developed a preliminary plan for the next time the ship heads into space.

Bill Gerstenmaier, a SpaceX executive in charge of build and flight reliability, presented the findings Monday at the American Astronautical Society's Glenn Space Technology Symposium in Cleveland.

The rocket lifted off on August 26 from SpaceX's launch pad in Starbase, Texas, just north of the US-Mexico border. It was the 10th full-scale test flight of SpaceX's Super Heavy booster and Starship upper stage, combining to form the world's largest rocket.

There were a couple of overarching objectives on the August 26 test flight. SpaceX needed to overcome problems with Starship's propulsion and propellant systems that plagued three previous test flights. Then, engineers were hungry for data on Starship's heat shield, an array of thousands of tiles covering the ship's belly as it streaks through the atmosphere during reentry.

"Things went extremely well," Gerstenmaier said.



Product Velocity from last starship flight we need to seal the tiles

Starship vs. Space Launch System (2025)

Capacity to LEO

Development Start

First mission planned

Reusable

Cost per Launch

Development cost to date (2025)

Number of (test) flights

100–150 t

mid 2010

2025/6

yes

\$ 10 m

\$ 7 b

10



SpaceX's Starship

95 t

2011

2026

no

\$ 4100 m

\$ 23 b

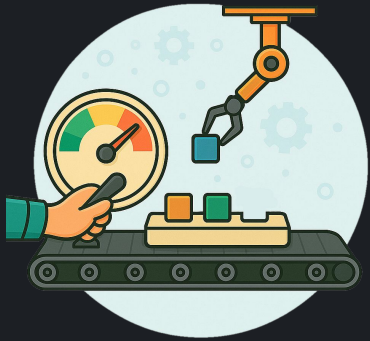
1



NASA's SLS

4. Prinzip

Accelerate



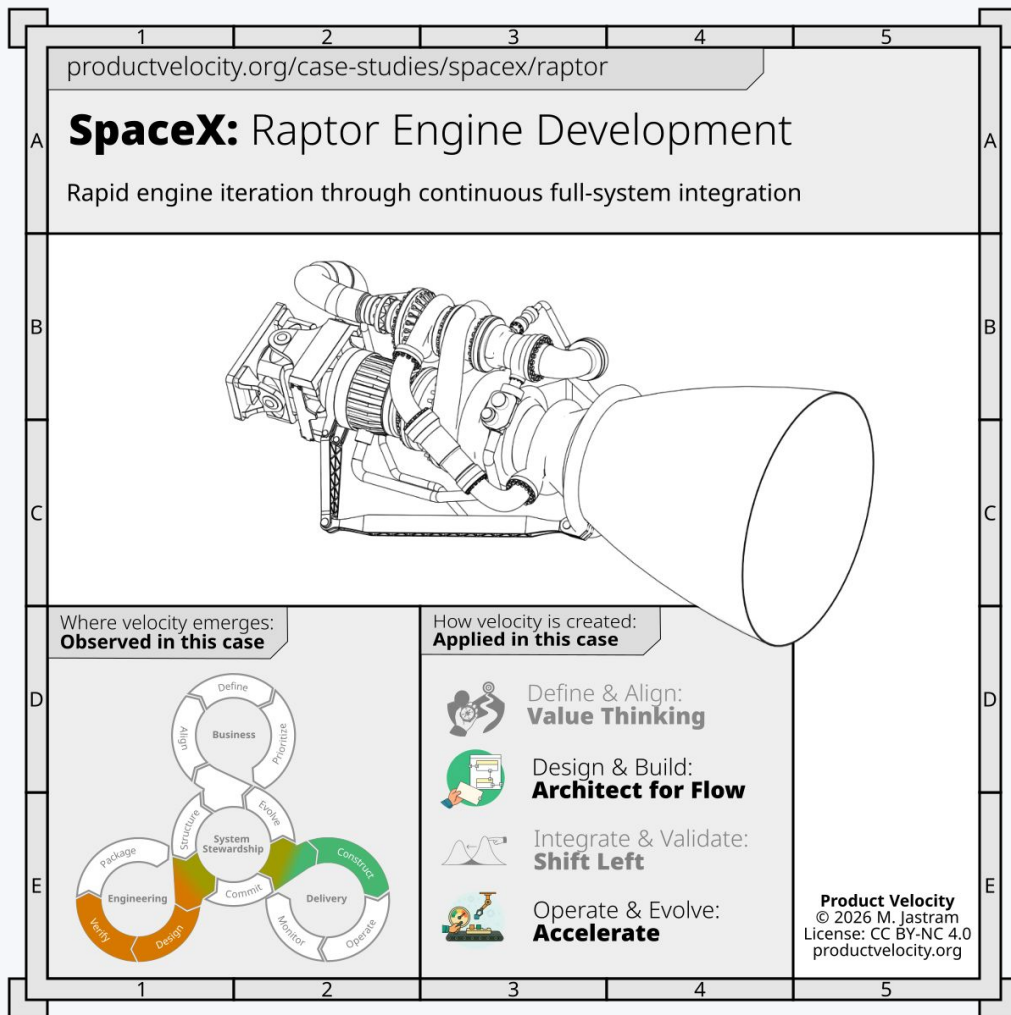
Praktiken

1. Bleib lauffähig und integriere regelmäßig
- 2. Etabliere Industrial DevOps**

SpaceX Raptor Engine

Vollständige Design-Iteration
alle 2 Tage:

- Redesign
- Bauen
- Testlauf



4. Prinzip

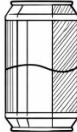
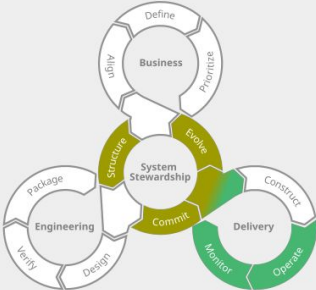
Accelerate



Praktiken

1. Bleib lauffähig und integriere regelmäßig
2. Etabliere Industrial DevOps
- 3. Nutze Betriebsdaten für schnelle Rückkopplung**



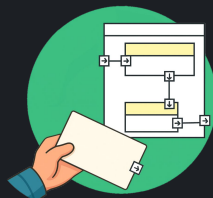
	1	2	3	4	5	
productvelocity.org/case-studies/ameru/mlops						
A	Ameru: Closing the Loop with MLOps Feedback on a cadence improves sorting models and keeps users engaged					A
B	METAL  CAN Recyclable WAS IT CORRECT? 					B
C						C
D	Where velocity emerges: Observed in this case 		How velocity is created: Applied in this case  Define & Align: Value Thinking  Structure & Scale: Architect for Flow  Build & Validate: Shift Left  Operate & Evolve: Accelerate			D
E	Product Velocity © 2026 M. Jastram License: CC BY-NC 4.0 productvelocity.org					E
	1	2	3	4	5	

4 Principles

Define & Align:
Value Thinking



Design & Build:
Architect for Flow



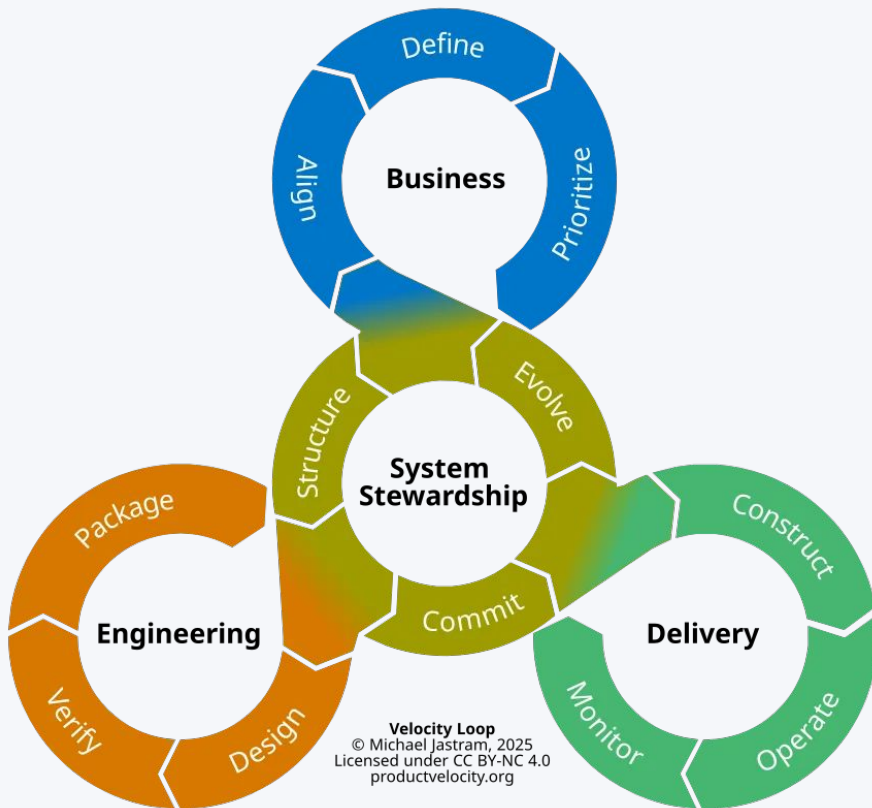
Integrate & Validate:
Shift Left

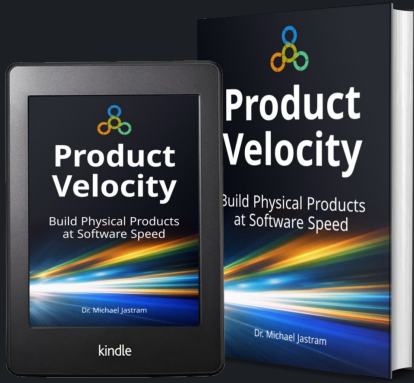


Operate and Evolve:
Accelerate



Velocity Loop





Buch ab
Sommer 2027

www.se-trends.de

www.productvelocity.org

 **Product** Velocity

Clarke's Gesetz, 1962

Der einzige Weg, die Grenzen des Möglichen zu entdecken, besteht darin, sich ein wenig über sie hinaus ins Unmögliche vorzuwagen.

Arthur C. Clarke

Vielen Dank!