MAJOR MILESTONES B.V.

SAFe® Transformaties — Van Strategie naar Executie

Achter de Schermen van SAFe® Transformaties

Praktijklessen uit SAFe®-transformaties
2012 – 2026

Oktober 2026

Versie 1.0

ing. R.P. Peels

SAFe® Practice Consultant (SPC) | Authorised SAFe® Trainer

Certified SA, SPC, SSM, RTE, LPM, POPM, SDP, SP, SMA, RT, KMP1

Voorwoord

Dit boek is niet in één keer geschreven. Het is ontstaan zoals de beste inzichten ontstaan: in de praktijk, iteratie na iteratie, Planning Interval na Planning Interval.

Tussen 2012 en 2026 schreef ik ruim 110 artikelen en reflecties; de 64 beste en meest gedeelde staan in dit boek. Ze zijn geboren op de werkvloer van complexe organisaties, in vergaderzalen waar de grote vragen werden gesteld, en soms laat op de avond achter een laptop, wanneer een gedachte te goed was om niet op te schrijven.

De rode draad is SAFe, het Scaled Agile Framework van Scaled Agile, Inc. Maar dit boek gaat over meer dan een framework: over mensen die moeten veranderen, managers die hun rol opnieuw uitvinden, teams die leren samenwerken in een steeds sneller bewegende wereld, en over jou als Agile Professional, soms in onzekere tijden. En over de vragen die niemand hardop durfde te stellen, en de antwoorden die we samen vonden.

De publicaties zijn analytisch, provocerend en soms bijna filosofisch. Van *"wat is de oorsprong van SAFe?"* tot *"Beste werknemer van de maand"*, van de inrichting van een LACE Kanban board en de kosten van een Epic tot beschouwingen over leiderschap, cultuur en de evolutie van Agile. En ja, ook een antwoord op de vraag of Agile überhaupt nog leeft. Niets is vanuit een ivoren toren geschreven: elk artikel wortelt in echte vraagstukken, echte mensen en echte organisaties. Dat geeft de teksten hun ruwheid én hun waarde.

Dit boek is voor iedereen die midden in een SAFe-transformatie staat of er binnenkort mee te maken krijgt: de manager die zich afvraagt wat het voor hem of haar betekent, het teamlid dat voelt dat er iets verandert, de consultant die inzichten zoekt die niet in het officiële SAFe-materiaal staan, en de executive die wil begrijpen waarom een SAFe transformatie best ingewikkeld is.

Volg de hoofdroute uit de leeswijzer, of blader gericht naar de titel die je aanspreekt en laat je verrassen. Zet een regel je aan het denken, of ben je het ergens mee oneens, dan heeft het zijn werk gedaan. Eerlijk is eerlijk: ik weet ook niet alles. Ik ben me terdege bewust van mijn eigen *Dunning-Krugereffect*, waarbij je het meest zelfverzekerd bent op het moment dat je het minst weet. Ik probeer aan de goede kant van die curve te zitten, maar garanties geef ik niet.

Dit boek is dan ook nooit af. Ik leer elke dag bij, en inzichten die gisteren stevig leken, verschuiven soms vandaag. Daarom blijft het voorlopig een PDF die regelmatig wordt bijgewerkt; let op het versienummer, want er is mogelijk al een nieuwere versie. Complimenten, tips, taalfouten, zinnen die niet lopen of alternatieve denkrichtingen zijn welkom via safe@majormilestones.nl. Want daar draait het om: verbinding en elkaar laten groeien. SAFe noemt het zelf: "Growing others."

Roel Peels
Oktober 2026

Leeswijzer

De hoofdroute loopt van kennismaking met SAFe via theorie, transformatie en leiderschap naar teams, work items, governance, metrics en kosten. Het negende deel biedt aanvullende hoofdstukken over persoonlijk vakmanschap. De uitleg van SAFe-begrippen vind je in hoofdstuk 10, ‘Het SAFe-jargon ontcijferd: ~30 woordjes om presentaties en workshops beter te begrijpen’. Informatie over de auteur staat op de cover.

Hoe dit boek is georganiseerd

Binnen elk deel bouwen de hoofdstukken op van inleiding naar verdieping. De introductie begint met het DNA van SAFe als fundament en gaat daarna van herkenning en eenvoudig begrip naar de schaalvraag, achtergrond, verdieping en naslag. Je kunt de hoofdroute volgen of gericht een onderwerp opzoeken.

Leesregel voor SAFe-versies

Dit boek bundelt praktijkartikelen die in verschillende jaren en vanuit verschillende versies van SAFe zijn geschreven. Veel voorbeelden gebruiken de terminologie en inrichting van SAFe 6.0. Andere hoofdstukken bespreken latere ontwikkelingen, zoals de vijf disciplines en AI-Native SAFe. Een expliciet genoemd versienummer geeft de context van de betreffende uitleg of het praktijkvoorbeeld aan; het betekent niet dat die versie nog de meest actuele is.

Voor hetzelfde begrip kun je daardoor verschillende namen tegenkomen, zoals Program Increment en Planning Interval, beide afgekort tot PI. Een andere naam betekent niet automatisch dat ook de inhoud of werkwijze is veranderd. De hoofdstukken over het DNA en de bouwstenen van SAFe bieden aanvullende context bij de ontwikkeling van het framework. Gebruik bij een nieuwe implementatie de actuele officiële SAFe-guidance en beoordeel welke historische voorbeelden en eigen ontwerpkeuzes uit dit boek passen bij jouw organisatie.

Deel 01: Introductie, wat is SAFe? Hoofdstukken 1 tot 11 beginnen met Het DNA van SAFe als fundament voor de rest van het boek. Daarna volgen Waarom SAFe?, SAFe voor Dummies en de schaalvraag. Vervolgens komen het Spotify-model, de evolutie van Agile en de discussie over ‘Agile is dead’ aan bod. Daarna verdiepen we de oorsprong en het ecosysteem, en de geleende en eigen bouwstenen. Het jargonhoofdstuk en SAFe leren kennen: video’s en boeken sluiten deel 01 af als naslag en praktische oriëntatie.

Deel 02: Theoretische fundamenten Hoofdstukken 12 tot 19 behandelen frameworks en empirisme, het Agile Manifesto en MECE. Daarna volgen twee hoofdstukken over waarde: eerst wat waarde in SAFe is, met kosten, urgentie en Cost of Delay, en vervolgens waarom het begrip waarde zo lastig is en wie die waarde eigenlijk bepaalt. Samen bereiden ze het hoofdstuk over economische prioritering en WSJF voor. Dependencies en Flow Accelerators sluiten het deel af.

Deel 03: SAFe-transformatie Hoofdstukken 20 tot 29 lopen van visie en pitch via de impact op klanten, gebruikers, medewerkers en management naar terminologie en het Target Operating Model. Daarna volgen het beoordelingskader voor een écht goed veranderplan en accountability in SAFe. Lees bij de start ook de cultuur- en leiderschapshoofdstukken uit deel 04. Van PMO naar VMO staat in deel 07.

Deel 04: Leiderschap, cultuur en mindset Hoofdstukken 30 tot 35 gaan over de verandering in denken en leiden die SAFe vraagt. Dit deel begint bij het individu: wat maakt iemand een ware Agilist? Vervolgens behandelen we drie culturen in één organisatie, hoe je cultuur meetbaar maakt, leiderschap in hiërarchie en waardestroomnetwerk en de rolgrenzen tussen lijnmanager en collega. We sluiten af met de perceptie dat SAFe overhead creëert.

Deel 05: Teams en teamorganisatie Hoofdstukken 36 tot 41 beginnen bij de essentiële teamkenmerken en het topteam. High Performance Teams verdiept vertrouwen, focus en discipline. Daarna volgen roulerend voorzitterschap, meetings en effectieve Iteration Reviews. Het hoofdstuk over projecten en SAFe Epics staat in deel 06.

Deel 06: Work items, portfolio en waarde Hoofdstukken 42 tot 48 lopen van missie, visie, strategie, Strategic Themes, OKR's en kernwaarden naar work item governance. Daarna volgen projecten en SAFe Epics, Business en Enabler Features, de kernelementen van een Feature, User Story sizing en het onderscheid tussen risico's en impediments. De twee hoofdstukken over het begrip waarde staan in deel 02; professioneel adviseren staat in deel 09.

Deel 07: LACE, governance en enterprise Hoofdstukken 49 tot 55 behandelen de enterprise-laag van SAFe: het LACE, het Hub-and-Spoke Model en het LACE Kanban board. Daarna volgen de transformatie van PMO naar VMO, SAFe-governance, de vraag waarom je geen extra werkrollen op ART-niveau wilt en de Enterprise Cadence die de organisatie synchroniseert.

Deel 08: Metrics en kosten Hoofdstukken 56 tot 57 verbinden sturing op de juiste metrics binnen het operating model met inzicht in de kosten van de SAFe-fabriek: Agile Teams, ART's en Lean Portfolio Management.

Deel 09: Persoonlijk vakmanschap Hoofdstukken 58 tot 64 vormen het optionele bonusdeel: pragmatisme en theorie, professioneel adviseren, Inbox Zero, de e-mailonderwerpregelconventie, documentbeheer met een GMS, bestandsnamen volgens de BNC en individuele erkenning versus teamdynamiek. De e-mailconventie, het GMS en de BNC zijn drie afzonderlijke hoofdstukken. SAFe leren kennen: video's en boeken staat in deel 01.

Gebruik de inhoudsopgave voor de hoofdstuktitels en de paginanummers van de opgenomen delen. In de PDF zijn alle negen delen en alle hoofdstukken direct aanklikbaar. De hoofdstuknummers blijven doorlopen over de negen delen.

Inhoudsopgave

Voorwoord . . . 3

Leeswijzer . . . 4

Deel 1: Introductie: wat is SAFe? . . . 10

1. Het DNA van SAFe: oorsprong, opbouw en de kracht achter het framework . . . 10
2. Waarom SAFe? 19 structurele organisatieproblemen die het framework helpt oplossen . . . 18
3. SAFe voor Dummies: wendbaar werken in Jip-en-Janneke-taal . . . 21
4. SAFe is een Scaled Agile Framework, maar wat schaalt er eigenlijk? . . . 29
5. Waarom SAFe en niet het Spotify-model? Zeven argumenten voor enterprise-organisaties . . . 35
6. De evolutie van Agile: van Darwin tot SAFe . . . 40
7. People who say "Agile is dead" are dead . . . 45
8. SAFe ontleed: oorsprong en het ecosysteem achter 's werelds grootste Agile framework . . . 51
9. Het SAFe framework ontleed: wat het zelf bedacht en wat het van anderen leende . . . 54
10. Het SAFe-jargon ontcijferd: ~30 woordjes om presentaties en workshops beter te begrijpen . . . 65
11. SAFe leren kennen: video's en boeken . . . 69

Deel 2: Theoretische fundamenten . . . 71

12. Verschillen en overeenkomsten tussen frameworks, methodologieën en empirisme . . . 71
13. Het Agile Manifesto: de core van SAFe met een uitleg . . . 73
14. Professionals denken en doen MECE . . . 78
15. Wat is waarde in SAFe? Over kosten, urgentie en Cost of Delay (deel 1) . . . 83
16. Waarom het begrip 'waarde' voor velen zo lastig is (deel 2) . . . 86
17. Economische prioritering in SAFe: waarde, Cost of Delay en WSJF, oefen zelf met prioriteren . . . 93
18. Van dependency management naar dependency orchestration in SAFe . . . 97
19. Hoe SAFe experts met slimme ingrepen 'Time-to-field' verkorten en Value Stream efficiency verhogen . . . 104

Deel 3: SAFe-transformatie . . . 109

20. Visie op de Agile transformatie: waar organisaties mee stoppen en mee beginnen . . . 109

21. Van pitch naar praktijk: de kunst van het verkopen van je SAFe-transformatie . . . 114

22. Transformeren naar SAFe: wat betekent dit voor klanten, gebruikers en toekomstige Business Owners? . . . 121

23. Transformeren naar SAFe: wat verandert er voor werknemers? . . . 127

24. Transformeren naar SAFe: wat betekent dit voor zittend management? (deel 1) . . . 135

25. Transformeren naar SAFe: wat betekent dit voor zittend management? (deel 2) . . . 139

26. De impact van het aanpassen van SAFe-terminologie: voordelen en nadelen voor je organisatie . . . 145

27. Een Target Operating Model (TOM): het fundament onder elke complexe en grootschalige SAFe-transformatie . . . 148

28. Beoordelingskader: wanneer vinden wij dit een écht goed veranderplan? . . . 153

29. Transformeren naar SAFe: wie is waarvoor verantwoordelijk? (deel 3) . . . 156

Deel 4: Leiderschap, cultuur en mindset . . . 159

30. De essentiële kenmerken van een ware Agilist . . . 159

31. Eén organisatie, drie culturen: het echte vraagstuk achter SAFe-transformaties . . . 163

32. Hoe maak je SAFe-cultuur, houding en gedrag meetbaar? . . . 166

33. Leiderschap in SAFe: schakelen tussen hiërarchie en waardestroomnetwerk . . . 171

34. Lijnmanager én gelijkwaardige collega? Waarom SAFe duidelijke rolgrenzen vraagt . . . 174

35. SAFe-overhead: perception or reality? De onmisbare schakel naar efficiëntie . . . 176

Deel 5: Teams en teamorganisatie . . . 178

36. Dit zijn de essentiële kenmerken van een SAFe Agile Team . . . 178

37. Wanneer is een Agile Team écht een topteam? . . . 184

38. High Performance Teams: waar vertrouwen, focus en discipline samenkomen . . . 187

39. Waarom roulerend voorzitterschap goed past in een SAFe-omgeving . . . 194

40. Meer meetings door SAFe? Wat er werkelijk verandert . . . 196

41. SAFe Iteration Review: handleiding voor een effectieve Iteration Review . . . 198

Deel 6: Work items, portfolio en waarde . . . 201

42. Missie, visie, strategie, Strategic Themes, OKR's en kernwaarden bij Defensie . . . 201

43. SAFe Work Items: het governance model dat je goed in je tooling moet inrichten . . . 208

44. Projecten en SAFe Epics: overeenkomsten en verschillen . . . 215

45. Business of Enabler? Waarom het verschil begrijpen essentieel is . . . 218

46. De kernelementen van een Feature: theorie versus praktijk . . . 221

47. User Story sizing in SAFe: hoe groot mag een User Story zijn? . . . 226

48. SAFe-risico's en impediments: wat is het verschil? . . . 229

Deel 7: LACE, governance en enterprise . . . 231

49. Het LACE in SAFe: van bottom-up implementaties naar enterprise agile governance . . . 231

50. SAFe transformaties met het SAFe LACE Hub-and-Spoke model . . . 235

51. Ontwerp van het LACE Kanban board . . . 241

52. Van PMO naar VMO: de transformatie naar waardegedreven governance in SAFe . . . 247

53. SAFe-governance: structuur in een wendbare organisatie . . . 251

54. Waarom we géén werkrollen op ART-niveau willen, en welke alternatieven SAFe biedt . . . 254

55. De enterprise cadence: ontwerpcriteria voor een werkend SAFe-ritme . . . 257

Deel 8: Metrics en kosten . . . 264

56. Data Driven Improvements: sturen op de juiste metrics binnen je totale operating model . . . 264

57. Een rekenmodel voor de kosten van de SAFe fabriek . . . 275

Deel 9: Persoonlijk vakmanschap . . . 280

58. Goed pragmatisme zonder solide theoretische basis is gokken! . . . 280

59. Wie ongevraagd wil adviseren moet goed weten wat hij doet . . . 284

60. Waarom mijn zakelijke inbox elke dag leeg is en ik nooit iets vergeet of mis: ondanks 100+ mails per dag! . . . 292

61. E-mailonderwerpregelconventie: grip op je inbox tijdens een SAFe-transformatie 297

62. Documentchaos in je Agile transformatie? Een slim ontworpen GMS lost het op (deel 1) 299

63. Documentchaos in je Agile transformatie? Een slim ontworpen BNC lost het op (deel 2) 316

64. Werknemer van de maand. Klinkt leuk. Maar is het ook slim? 320

Deel 1: Introductie: wat is SAFe?

1. Het DNA van SAFe: oorsprong, opbouw en de kracht achter het framework

Waarom dit hoofdstuk er moet zijn

Als je dit boek openslaat, is de kans groot dat je organisatie midden in een SAFe-transformatie zit, of er een overweegt. In de hoofdstukken die volgen duiken we diep in de praktijk: work-itemmodellen, governance-inrichting, Jira Align-configuratie, portfolio management en de tientallen ontwerpbeslissingen die je moet nemen voordat je ook maar één tool aanraakt.

Maar al die hoofdstukken veronderstellen dat je begrijpt **wat SAFe® is**. Niet op slidedeck-niveau, maar écht: wat het framework drijft, hoe het is opgebouwd, en waarom het werkt zoals het werkt. Zonder die basis lees je de rest van dit boek als een handleiding zonder context: je ziet de stappen, maar niet het waarom.

Dit eerste hoofdstuk legt dat belangrijke fundament. We beginnen bij de oorsprong: waarom Agile ontstond, wat Scrum en Lean eraan toevoegden, en hoe SAFe die stromingen samenbrengt tot één schaalbaar model. Daarna nemen we je mee door de opbouw van het framework (van team tot portfolio) en sluiten we af met een verzameling korte video's waarmee je het in je eigen tempo kunt verkennen.

Beschouw dit hoofdstuk als je kompas. Hoe beter je dit begrijpt, hoe scherper je de keuzes in de rest van het boek kunt beoordelen.

Wat is SAFe®?

“SAFe® is the world’s leading, customizable framework for Enterprise Business Agility, now complemented by AI-Native SAFe. SAFe integrates a vast array of proven best practices from a wide range of disciplines, including Lean, Agile, DevOps, Design Thinking, Lean Startup, systems thinking, leadership, finance, and many others.

SAFe’s distinctive contribution is an integrated operating model connecting strategy to execution, organizing around value, and aligning and synchronizing teams across the enterprise.

It combines these practices with SAFe-specific constructs, including ARTs, PI Planning, RTEs, Lean Portfolio Management and Inspect & Adapt, among others.”

Dat is de formele definitie. Maar wat betekent dat in de praktijk?

SAFe®, ontstaan in het jaar 2011, is een bewezen kennisbank van principes, praktijken en competenties waarmee grote organisaties wendbaarheid bereiken op enterprise-schaal. Het is geen abstract model dat in een la verdwijnt, het is een operationeel raamwerk, een business operating model, dat bepaalt hoe je strategie vertaalt naar taken op teamniveau en tot een goed werkend product, hoe je honderden mensen laat samenwerken, en hoe je dat meetbaar en bestuurbaar houdt.

De kern in één zin

SAFe brengt **strategie en uitvoering** samen: van de visie van de directie tot het werk dat teams dagelijks oppakken, in één doorlopende, transparante keten.

Vier pijlers, één framework

SAFe brengt inzichten en praktijken uit Lean, Agile, DevOps en Systems Thinking samen in één geïntegreerd raamwerk. Hieronder licht ik deze vier pijlers toe:

• **Agile**: iteratief en incrementeel ontwikkelen in kleine, zelforganiserende teams. Denk aan het Agile Manifesto als absolute basis met Scrum en Kanban als empirisch toegepaste procesmodellen; korte cycli, snelle feedback, continu leren.

• **Lean**: afkomstig uit het Toyota Production System. Focus op waardestroom, eliminatie van verspilling, en respect voor de mensen die het werk doen.

• **Systems Thinking**: begrijp het geheel voordat je de delen optimaliseert. Voorkom suboptimalisatie, herken feedback loops, en ontwerp oplossingen die het hele systeem versterken, niet alleen één team of afdeling.

• **DevOps (inclusief DevSecOps)**: de brug tussen ontwikkeling, security en operations. SAFe® behandelt DevOps en DevSecOps als synoniemen; security is geen aparte laag, maar zit verweven in elke stap van de delivery pipeline. Continue integratie, continue levering, en een cultuur van gedeelde verantwoordelijkheid voor het eindresultaat.

Elk van deze pijlers is krachtig op zichzelf. SAFe® combineert ze tot een samenhangend model dat werkt voor tientallen of zelfs honderden teams tegelijk. De beloftes van SAFe (sneller leveren, hogere kwaliteit, hogere betrouwbaarheid en hogere werknemerstevredenheid) zijn geen marketingpraat, maar de meetbare uitkomsten wanneer je deze vier pijlers goed laat samenwerken op schaal.

SAFe®: integratie én innovatie

SAFe® wordt soms afgedaan als 'niets nieuws': een framework dat alleen bestaande ideeën bij elkaar harkt tot 1 model. Dat is een klein misverstand. De waarheid is genuanceerder.

Want, ja, SAFe heeft enorm veel overgenomen van de markt. Op teamniveau herken je Scrum, Kanban, XP en Lean. Reinertsen's Product Development Flow en Kotter's verandermanagement zitten diep verweven in SAFe's DNA. Het overzicht aan het eind van dit hoofdstuk toont meer dan vijftig best practices die SAFe heeft geïntegreerd, stuk voor stuk bewezen methodieken met hun eigen grondleggers. In die zin is SAFe inderdaad een greatest hits album: de nummers zijn niet nieuw, de compilatie wel. En wie de originele artiesten kent, leert SAFe in een fractie van de tijd.

Maar SAFe heeft óók zelf fundamentele concepten ontwikkeld die nergens anders bestonden en waar de eer volledig aan SAFe zelf toekomt, zoals o.a.:

• **De Agile Release Train (ART)**: het idee om 50 tot 125 mensen (conform Dunbar's numbers) te organiseren als een langlevend 'team of teams' dat op vaste cadence samenwerkt en levert. Dit concept werd door Dean Leffingwell geïntroduceerd in zijn boek *Agile Software Requirements* (2011) en vormde de kern van SAFe vanaf versie 1.0.

• **PI Planning**: het grote, gezamenlijke planningsevenement waarin alle teams van een ART face-to-face hun Planning Interval plannen. Geïnspireerd door Lean's Obeya ('big room'), maar door SAFe geformaliseerd tot een gestructureerd, herhaalbaar evenement met een specifiek format, rollen en artefacten.

• **De Release Train Engineer (RTE)**: een volledig nieuwe rol, de 'uber-ScrumMaster' , de Master der Scrum Masters, die op programmaniveau de ART faciliteert, impediments escaleert en de cadence bewaakt en de CICD pipeline mede *engineert*. Deze rol bestond niet vóór SAFe.

• **Het Planning Interval (PI)**: de timebox van 8 tot 12 weken als ritmisch hart van het framework, met de Innovation & Planning Iteration (laatste in een PI) als ingebouwde buffer voor innovatie en verbetering tussen PI's.

• **Solution Trains**: het schalen boven ART-niveau; meerdere ART's en leveranciers coördineren rondom één grote oplossing, zoals bijvoorbeeld een grootschalige Defensie-oplossing.

• **Lean Portfolio Management**: de complete portfoliolaag (ook LPM genoemd) die strategie, lean budgeting, guardrails en epic-governance koppelt aan de uitvoering. SAFe was het eerste framework dat dit zo expliciet uitwerkte.

• **De gelaagde architectuur**: de doorlopende keten van Team → ART → Solution Train → Portfolio → Enterprise als één geïntegreerd besturingsmodel.

De kracht van SAFe zit dus niet alleen in integratie, maar ook in de originele schaalconcepten die het heeft toegevoegd. SAFe nam de beste ideeën van de markt, voegde daar eigen innovaties aan toe voor het schalen, en smeedde het samen tot een samenhangend geheel met één gemeenschappelijke taal. Dat is geen kopie, dat is systeemontwerp.

Een volledig overzicht van alle geïntegreerde best practices vind je aan het eind van dit hoofdstuk.

Hoe is SAFe® opgebouwd?

SAFe® organiseert werk op meerdere niveaus, zodat zowel teams als het management grip houden:

• **Team level**: hier werken agile teams in korte iteraties aan concrete Stories en Tasks. SAFe kent twee teammodellen: **SAFe Scrum** en **SAFe Team Kanban**, aangepaste varianten van Scrum en Kanban die zijn geoptimaliseerd voor samenwerking binnen het bredere SAFe-framework.

• **ART level (Agile Release Train)**: meerdere teams die samen aan één waardestroom werken, gesynchroniseerd via Planning Intervals (PI's) van doorgaans 8 tot 12 weken.

• **Solution Train level**: voor grotere oplossingen die meerdere ART's omvatten en solution-integratie vereisen.

• **Portfolio level**: hier verbindt Lean Portfolio Management (LPM) de strategie aan de uitvoering. Budgetten, Epics en guardrails worden hier beheerd.

• **Enterprise level**: hier worden de visie, de missie en de strategie van een onderneming geformuleerd, en indien nodig, ook Enterprise level Epics.

Vijf disciplines

Om business agility te bereiken, organiseert SAFe het geheel aan kennis, vaardigheden en praktijken in vijf disciplines. Elke discipline richt zich op een specifiek aandachtsgebied en bevat meerdere competenties die elk een concreet bedrijfsprobleem helpen oplossen.

Eerdere versies van SAFe werkten met zeven kerncompetenties (zoals Lean-Agile Leadership, Organizational Agility en Continuous Learning Culture). Sinds 2025 zijn deze geherstructureerd naar de huidige vijf disciplines: een verschuiving van abstracte competentiegebieden naar een modulaire, probleemgerichte aanpak.

1. **Team and Technical Agility**: het fundament; hoog-performante agile teams met sterke technische praktijken die snel en betrouwbaar waarde leveren. Denk aan het opzetten van effectieve agile teams, het ontwikkelen van kwaliteitssoftware, en het continu leveren van waarde via DevOps-principes.

2. **Product Development Flow**: de stroomlijning van productontwikkeling, van strategie tot klantwaarde. Teams en ART's organiseren rondom waarde, versnellen de productflow, benutten klantfeedback, en creëren responsieve roadmaps met geïntegreerd productdesign.

3. **Large Solution Integration and Delivery**: voor organisaties die grote, complexe systemen bouwen met meerdere ART's, leveranciers en specialisten. Coördinatie van grote oplossingen, lean systems engineering, en roadmapping op solution-niveau.

4. **Lean Portfolio Management**: de verbinding tussen strategie en uitvoering op portfolioniveau. Organisatie van portfolio's, validatie van investeringskansen, enterprise-architectuur, gebalanceerd portfoliomanagement, en de transitie naar value stream funding.

5. **Leadership and Culture**: de menselijke kant; leiders die Lean-Agile principes belichamen, een cultuur van continu leren en verbeteren stimuleren, en de organisatie in staat stellen wendbaar te opereren. Hier komen de vroegere competenties Lean-Agile Leadership, Organizational Agility en Continuous Learning Culture samen.

Elke discipline bevat concrete competenties die steeds starten vanuit een herkenbaar bedrijfsprobleem, niet vanuit theorie, maar vanuit de dagelijkse realiteit van de organisatie. Die probleemgerichte benadering maakt het framework toegankelijker en direct toepasbaar.

Dual Operating System

Een belangrijk concept binnen SAFe is het **Dual Operating System**, geïnspireerd op het werk van John Kotter (*Accelerate / XLR8*). De organisatie kent twee 'besturingssystemen' die naast elkaar bestaan:

• **De lijn**: de bestaande hiërarchische structuur die zorgt voor stabiliteit, compliance en dagelijkse operatie. Ontwikkelprocessen verdwijnen hier.

• **Het Value Stream Network**: een dynamisch netwerk van ART's en Solution Trains dat zorgt voor snelheid, innovatie en waardecreatie. Dit netwerk voert feitelijk alle ontwikkelmanagementprocessen uit die voorheen door de lijn werden uitgevoerd.

Beide systemen zijn nodig. De lijn biedt structuur; het netwerk biedt wendbaarheid. SAFe helpt ze naast elkaar te laten functioneren zonder dat ze elkaar in de weg zitten.

Ontdek zelf SAFe in 7 korte video's

De volgende video's introduceren Agile en SAFe en laten zien hoe het framework samenhangt:

1. Waarom Agile?

Waarom stappen organisaties wereldwijd over op agile manieren van werken? Deze korte video legt het uit.

Bekijk: Why Agile?

2. Wat is Agile?

Een heldere uitleg van de kernprincipes achter agile werken, en waarom incrementeel ontwikkelen zo krachtig is.

Bekijk: Wat is Agile?

3. Wat is Scrum?

Let op: Scrum is geen *methode*, maar een framework, een empirisch procesmodel. Het verschil is belangrijk.

Bekijk: Wat is Scrum? of bekijk deze video

4. Is Scrum hetzelfde als Agile?

Scrum en Agile worden vaak door elkaar gehaald, maar het is niet hetzelfde. Deze video legt het verschil helder uit.

Bekijk: Is Scrum Agile?

5. SAFe integreert vier pijlers

SAFe brengt inzichten en praktijken uit Lean, Agile, DevOps en Systems Thinking samen in één geïntegreerd raamwerk. Deze video legt uit hoe Lean en Agile zich tot elkaar verhouden, en waar ze elkaar versterken.

Bekijk: Lean vs Agile

6. Het Dual Operating System

SAFe kent twee besturingssystemen die naast elkaar bestaan: de traditionele lijnorganisatie en het Value Stream Network. Dit concept is afgeleid van John Kotter's boek Accelerate (XLR8).

Bekijk: Dual Operating System

7. Algemene introductie van SAFe

Een bredere introductievideo die het hele framework in context plaatst.

Bekijk: SAFe Introductie

Tot slot: Overzicht best practices achter SAFe®

SAFe® integreert tientallen bewezen methodieken, modellen en denkwijzen. Hieronder een overzicht, gegroepeerd per domein, inclusief de grondleggers. Herken je er een paar? Dan ben je al goed op weg. Herken je ze allemaal? Dan weet je al het meeste van SAFe®, je moet het alleen nog leren toepassen op schaal.

Agile & teamwerkmethoden

Best Practice	Grondlegger(s)
Scrum (in SAFe: SAFe Scrum)	Jeff Sutherland & Ken Schwaber
Kanban (in SAFe: SAFe Team Kanban)	Taiichi Ohno (Toyota), David J. Anderson (software)
Extreme Programming (XP)	Kent Beck
Test-Driven Development (TDD)	Kent Beck
Agile Manifesto (4 waarden, 12 principes)	17 ondertekenaars (2001)
Tuckman's teamfasen	Bruce Tuckman
Lencioni's vijf dysfuncties	Patrick Lencioni
Communities of Practice	Etienne Wenger

Lean & productiefundamenten

Best Practice	Grondlegger(s)
Toyota Production System	Taiichi Ohno & Shigeo Shingo
Lean Manufacturing	James Womack & Daniel Jones
Value Stream Mapping	Mike Rother & John Shook
Hoshin Kanri (strategische planning)	Yoji Akao
Gemba ("go and see")	Taiichi Ohno
Plan-Do-Check-Act (PDCA)	W. Edwards Deming
Theory of Constraints	Eliyahu Goldratt
Little's Law (wachtrijtheorie)	John Little
Lean Software Development	Mary & Tom Poppendieck

Productflow & economisch denken

Best Practice	Grondlegger(s)
Product Development Flow	Donald G. Reinertsen
Cost of Delay	Donald G. Reinertsen
Weighted Shortest Job First (WSJF)	Donald G. Reinertsen / SAFe
Lean Startup / Build-Measure-Learn	Eric Ries
Minimum Viable Product (MVP)	Eric Ries / Frank Robinson
Innovation Accounting	Eric Ries

Leiderschap & verandermanagement

Best Practice	Grondlegger(s)
Servant Leadership	Robert K. Greenleaf
Kotter's 8-stappenmodel	John Kotter
Dual Operating System / Accelerate (XLR8)	John Kotter
Leiderschapsstijlen (situationeel)	Hersey & Blanchard
Motivatietheorieën	Maslow, Herzberg, Deci & Ryan

Design & klantgerichtheid

Best Practice	Grondlegger(s)
Design Thinking	IDEO / Stanford d.school
Lean UX	Jeff Gothelf & Josh Seiden
Customer Journeys	Diverse (service design)
Empathy Mapping	Dave Gray (XPLANE)
Jobs To Be Done	Clayton Christensen
Business Model Canvas	Alexander Osterwalder
Net Promoter Score (NPS)	Fred Reichheld
Design Sprints	Jake Knapp (Google Ventures)

DevOps & technische praktijken

Best Practice	Grondlegger(s)
Continuous Integration	Martin Fowler & Kent Beck
Continuous Delivery	Jez Humble & Dave Farley
Infrastructure as Code	DevOps-beweging
Feature Toggles	Martin Fowler
DevSecOps	DevOps/Security-community
ITIL / CMDB-concepten	UK Office of Government Commerce

Systeemdenken & probleemanalyse

Best Practice	Grondlegger(s)
Systems Thinking / The Fifth Discipline	Peter Senge
Ishikawa / Visgraatdiagram	Kaoru Ishikawa
Root Cause Analysis (5× Waarom)	Sakichi Toyoda / Toyota
OODA Loop	John Boyd
SWOT / TOWS-analyse	Albert Humphrey

Strategie & portfoliomanagement

Best Practice	Grondlegger(s)
OKR's	Andy Grove (Intel), John Doerr
Set-Based Design (SBD)	Allen Ward (Toyota)
Model-Based Systems Engineering (MBSE)	INCOSE
Portfolio Management	Diverse (PMI, MoP, P3O)

Kwaliteit & governance

Best Practice	Grondlegger(s)
Kwaliteitsmanagementsystemen (QMS)	ISO / Diverse
Agile contractvormen	Diverse (Agile community)
PMO-principes	PMI / Diverse
Compliance & Regulatory frameworks	ISO, CMMI, COBIT

Dit overzicht maakt één ding duidelijk: SAFe staat op de schouders van tientallen jaren aan kennis en ervaring. Elk onderdeel van het framework heeft wortels in bewezen methodieken, van Toyota's productielijnen tot de agile manifesto-ondertekenaars in Snowbird. Wie deze bouwstenen herkent, begrijpt niet alleen *wat* SAFe voorschrijft, maar ook *waarom*. En dat is precies het fundament dat je nodig hebt voor de rest van dit boek. In de volgende hoofdstukken vertalen we dit DNA naar concrete ontwerpbeslissingen, governance-modellen en tooling, van theorie naar de werkvloer.

2. Waarom SAFe? 19 structurele organisatieproblemen die het framework helpt oplossen

Voordat organisaties overstappen op het Scaled Agile Framework (SAFe®), lopen ze doorgaans tegen een reeks structurele knelpunten aan. Die knelpunten variëren per sector en organisatiegrootte, maar zijn opvallend consistent bij organisaties die werken met traditionele projectmethodieken zoals MSP en PRINCE2, kortweg: waterval.

De problemen in deze lijst zijn algemeen: we komen ze tegen in vrijwel elke traditioneel ingerichte organisatie. Daarnaast heeft elke organisatie haar eigen, specifieke knelpunten, zoals een gebrek aan psychologische veiligheid op de werkvloer. Die vragen om een eigen analyse en vallen buiten deze lijst.

De meest voorkomende algemene problemen op een rij:

1. Rigide methodieken

MSP en PRINCE2 volgen een strak, lineair stramien. SAFe® brengt agile- en lean-principes in, waardoor organisaties veel beter kunnen inspelen op verandering.

2. Gebrek aan prioritering en projectoverload

Zonder een objectieve methode voor prioritering worden projecten vaak gekozen op basis van wie het hardst roept. SAFe®'s WSJF-methode (Weighted Shortest Job First) maakt economisch gefundeerde keuzes mogelijk. Het ontbreken hiervan leidt tot overvolle portfoliolijsten, resourceconflicten en uiteindelijk burn-out.

3. Inefficiënte resource-allocatie

In waterval worden mensen naar het werk gebracht. In SAFe® wordt het werk naar de mensen gebracht: naar langdurige, stabiele teams en ARTs, georganiseerd rond value streams. Dit voorkomt dat teams continu worden opgebroken en herverdeeld, en vermindert de opstarttijd en het kwaliteitsverlies die daarbij horen.

4. Lange time-to-market

Het duurt te lang om producten en nieuwe initiatieven te ontwikkelen en te leveren. Het gevolg: gemiste marktkansen en potentieel omzetverlies. SAFe® versnelt de doorlooptijd via kortere planningscycli en continue levering.

5. Gebrek aan alignment

Traditionele aanpakken leiden vaak tot verkokering: teams, afdelingen en projecten werken langs elkaar heen. SAFe® bevordert organisatiebrede afstemming zodat iedereen werkt aan dezelfde strategische doelen.

6. Beperkte flexibiliteit

Trage en omslachtige change request-procedures, change boards en besluitvormingsprocessen remmen de organisatie. SAFe® introduceert wendbaarheid en het vermogen om snel te reageren op marktveranderingen.

7. Moeizaam op- en afschalen

In grotere organisaties is schalen complex binnen traditionele hiërarchische structuren. SAFe® biedt concrete handvatten voor het managen van meerdere agile teams en programma's op grote schaal. Schalen gaat relatief eenvoudig, zowel horizontaal (meer ARTs, meer agile teams) als verticaal (meer SAFe levels).

8. Governance en compliance

De balans tussen governance, compliance en wendbaarheid is vaak moeilijk te vinden. SAFe® biedt governancemechanismen die structuur geven zonder de bewegingsvrijheid te beknotten.

9. Lage medewerkersbetrokkenheid

Traditionele managementstijlen zijn vaak directief van aard: command & control. SAFe® benadrukt medewerkersbetrokkenheid en empowerment, wat leidt tot hogere werktevredenheid en betere resultaten.

10. Kwaliteitsproblemen

Watervalmethoden produceren regelmatig kwaliteitsgebreken die pas laat in het proces zichtbaar worden. SAFe®'s agile en lean werkwijzen zijn gericht op het vroegtijdig inbouwen van kwaliteit: minder defects, minder rework, conform de ‘Built-in Quality’-principes.

11. Verkokerde organisaties en Conway's Law

Afdelingen en teams die in silo's werken, resulteren in systemen die de organisatiestructuur spiegelen (ref: Conway's Law). SAFe® stimuleert cross-functionele samenwerking en breekt deze silo's actief af.

12. Ineffectieve communicatie

Communicatiestoornissen op alle niveaus (van strategische visie tot teamoverleg) zijn een hardnekkig probleem. SAFe® werkt met een vooraf vastgestelde en doordachte cadans, zodat alle betrokkenen altijd op één lijn zitten.

13. Push versus pull

Traditionele organisaties werken primair push-gestuurd: werk wordt opgelegd, wat leidt tot stress, burn-out en kwaliteitsverlies. SAFe® is pull-gestuurd. Scrum Masters/Team Coaches en RTE's bewaken actief dat teams zich niet overcommitteren.

14. Gebrek aan continu verbeteren

SAFe® legt veel meer nadruk op een cultuur van continue verbetering dan waterval. Een concreet voorbeeld is de IP-iteratie (Innovation & Planning), speciaal ingericht voor reflectie, innovatie en verbetering.

15. Lage productiviteit

Silo's zijn niet zo productief als ze zouden kunnen zijn, wat leidt tot inefficiëntie, gemiste deadlines en hogere kosten. Een veelgehoorde klacht is ‘over de schutting gegooid’ of ‘te laat betrokken’. SAFe® optimaliseert de capaciteits- en werklastverdeling tijdens meerdere planningsevenementen.

16. Trage besluitvorming

Stuurgroepen die eens per maand bijeenkomen. Alle beslissingen die bij het management liggen. In watervalorganisaties is dit de norm, en het remt de doorstroom van waarde ernstig. SAFe® bevordert gedecentraliseerde besluitvorming, wat de flow van waarde direct ten goede komt.

17. Overmatige papierwinkel

Projectplannen van veertig pagina's, uitgebreide documentatieprocessen: ze kosten tijd, geld en zorgen regelmatig voor fouten. SAFe® werkt met een Epic Hypothesis Statement van één pagina en een Lean Business Case van maximaal twee pagina's.

18. Gebrek aan transparantie

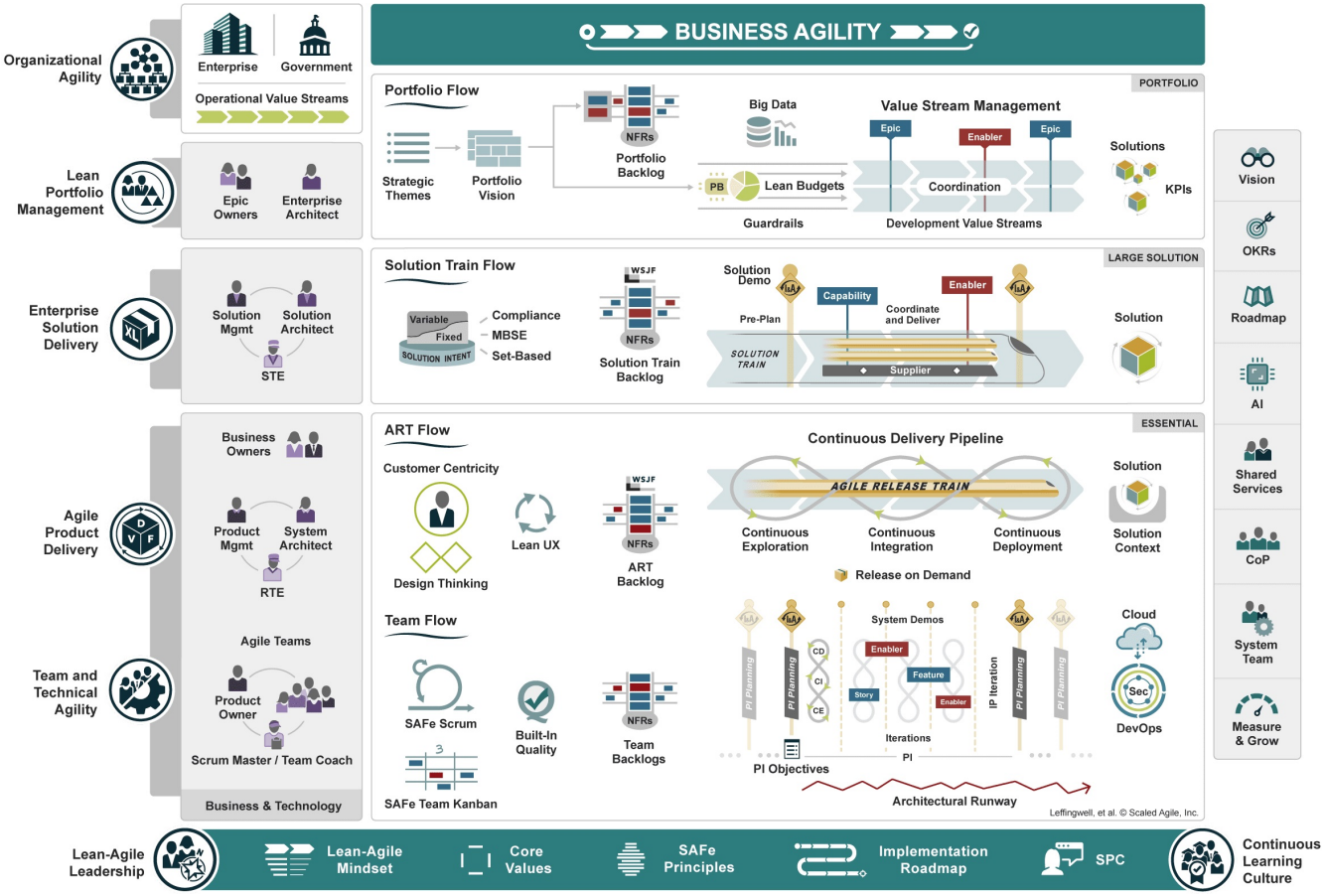
PRINCE2 en MSP hebben van oudsher een formeel, documentatiegedreven karakter dat problemen kan verbergen en heldere communicatie belemmert. SAFe® pakt dit aan met transparantieprincipes zoals Big Visual Information Radiators (BVIR), vrij toegankelijk voor iedereen in de organisatie.

19. Strategie in PowerPoint en Excel, uitvoering in Jira

In veel organisaties stuurt het management nog met PowerPoint en Excel: de strategie wordt primair in die tools vastgelegd en bijgehouden. De agile teams werken daarentegen in Jira. Daardoor ontbreekt een oprolfunctionaliteit: welk team werkt waaraan, en hoe draagt dat bij aan de strategische doelen en strategische thema's? Jira-output moet handmatig worden doorvertaald naar voortgangsrapportages op managementniveau, wat voor teams vaak dubbele invoer betekent. SAFe® verbindt strategie en uitvoering in één doorlopende hiërarchie: van Strategic Themes via Epics en Features naar Stories. In portfoliotooling zoals Jira Align wordt die voortgang automatisch opgerold, zodat op elk niveau realtime zichtbaar is wat er gebeurt, zonder handmatige rapportages of dubbele invoer.

3. SAFe voor Dummies: wendbaar werken in Jip-en-Janneke-taal

SAFe voor Dummies: wendbaar werken in Jip-en-Janneke-taal



Je ziet en hoort het woord *SAFe* steeds vaker in gesprekken, rapporten, beleidsstukken en trainingen. Toch vragen veel mensen zich af: wat ís dat nou eigenlijk, “SAFe”? Het klinkt ingewikkeld, zit vol Engelse termen, en doet denken aan ICT'ers met managementjargon.

Speciaal voor iedereen die daar (nog) niet goed mee bekend is, is deze uitleg geschreven: zo helder mogelijk, in Jip-en-Janneke-taal, met toelichting op de belangrijkste SAFe-termen. Zodat je na het lezen (een beetje) begrijpt waar het over gaat, waarom het bestaat, en vooral: wat jij of jouw organisatie eraan heeft. Dit hoofdstuk geeft antwoord op de volgende meest gestelde vragen:

1. Waarom is iets als SAFe® nodig? Op welk probleem is SAFe het antwoord?

2. Waar komt SAFe vandaan, van wie is SAFe? Uit welke vakgebieden is SAFe opgebouwd, wat zijn de onderliggende principes of markt best practices?

3. Is SAFe alleen maar voor Software? Is SAFe alleen maar voor ontwikkeling of ook voor beheer?

4. Wat is het verschil tussen Scrum en SAFe?

5. Hoe leg je SAFe uit in max 3 zinnen?

6. Hoe implementeer je SAFe? Richt je net als bij projecten een ‘tijdelijke projectorganisatie’ in? Wat gebeurt er met de lijnorganisatie in SAFe?

7. Hoe stroomt waarde door SAFe? Hoe kom je van een missie naar concrete producten en diensten in SAFe?

8. Projectmanagement kent de duivelsdriehoek: hoe werkt dat in SAFe? Kan je in SAFe ook ‘uitlopen’ net als bij projecten?

9. Wat is na het lezen van bovenstaande antwoorden, het exacte verschil tussen projectmanagement en SAFe?

10. Wat zijn de allerbelangrijkste woordjes in SAFe, bijvoorbeeld welke rolnamen, welke begrippen?

Ik heb voor je gemak het hoofdstuk ook voorzien van een paar handige URL's, voor wat meer achtergrondinformatie en uitleg.

Rest me je heel veel leesplezier toe te wensen!

Allereerst: waarom iets als SAFe® nodig is!

Laten we beginnen met de vraag op welk probleem SAFe een antwoord biedt.

In grote organisaties wordt er vaak gewerkt volgens de watervalmethode, met marktbekende methodieken als PPM, PRINCE2 en MSP. Daarbij wordt alles van tevoren uitgedacht, gepland en daarna uitgevoerd: stap voor stap. Dat werkt goed bij voorspelbare projecten, zoals het bouwen van een brug of iets wat je al 10x eerder hebt gedaan en veel routine in hebt. Je weet *dan wat je weet* en *wat je niet weet.* Maar deze aanpak werkt niet meer bij complexe, grootschalige IT-transformaties zoals grootschalige IT-programma's of de OV-chipkaart. Want dan weet je namelijk niet wat je niet weet, het werken naar de oplossing wordt een 'zoektocht' en vergt maximale wendbaarheid ("Agile")

Dr. Ir. Niels Groen onderzocht waarom zulke waterval projecten vaak mislukken en beschreef dat in zijn boek 'De Bodemloze Put: Escalerend Commitment in Publieke IT-projecten'. De belangrijkste oorzaken:

• De wereld verandert sneller dan het projectplan.

• Gebruikerswensen worden onvoldoende opgehaald en meegenomen.

• Politieke beloften leiden tot druk om door te gaan, ook als het project niet werkt.

• Er is al zoveel geïnvesteerd dat stoppen 'geen optie meer is'.

• Onduidelijke besluitvorming en verantwoordelijkheden.

• De kloof tussen beheer en ontwikkeling leidt vaak tot problemen en vertragingen ('over muur gooien')

• Een te groot geloof in voorspelbaarheid en maakbaarheid (plannen maken tot 2035).

Kortom, de wereld is grillig en vooral onvoorspelbaar bij complexere zaken. Grote organisaties moeten snel kunnen schakelen, juist nu door de huidige geopolitieke realiteit. Dat lukt helaas niet met ouderwetse, trage, bureaucratische aanpakken of systemen die 20 jaar geleden wellicht nog werkten. Of zoals John P. Kotter zegt:

"The world is now changing at a rate at which the basic systems, structures, and cultures built over the past century cannot keep up with the demands being placed on them."

SAFe: een raamwerk voor het wendbaar maken van hele grote organisaties

Laten we nu wat verder inzoomen op SAFe®, met 3x een hoofdletter en kleine letter e. SAFe staat voor **Scaled Agile Framework voor lean enterprises**. Het framework is in Amerika ontwikkeld door Dean Leffingwell. De organisatie achter het framework is Scaled Agile, Inc. Scaled Agile helpt met hun framework SAFe grote organisaties in het geheel wendbaar(der) te worden, zonder grip te verliezen op kwaliteit, richting en structuur. Het is geen losse volledig nieuw bedachte methode, maar een verzameling van in de markt bewezen praktijken en inzichten, samengebracht in één geïntegreerd raamwerk. Als je deze praktijken en inzichten al beheerst, is SAFe® vrij snel te doorgronden:

SAFe combineert bijvoorbeeld denkwijzen, principes en methoden als:

• **Agile mindset** (2001: ontstaan uit softwareontwikkeling)

• **Lean** (1950: ontwikkeld bij Toyota)

• **DevOps** (vanaf 2009): verbindt ontwikkeling en beheer, met security geïntegreerd in de werkwijze (DevSecOps).

• **Bedrijfskundige inzichten** (zoals Design Thinking, Lean-UX, OKR's, TDD, SWOT en Ishikawa Root Cause Analysis, Dual OS van Kotter)

Dean Leffingwell, de bedenker van SAFe, bracht dit alles samen in die poster boven aan het hoofdstuk in het jaar 2011. Zijn visie toentertijd: Scrum werkt voor kleine teams, maar niet voor grote uitdagingen waarvoor je honderden mensen nodig hebt. Zoals bijvoorbeeld de bouw van een ASML machine.

SAFe: Scaled Scrum

Daarom introduceerde hij 3 nieuwe concepten als aanvulling op Scrum als:

• **Agile Release Trains (ART's)**: Dit is een *Team van Agile Teams* dat werkt als één groot *Scrumteam* (bijv. in 1 ART zitten 10 Scrumteams. Elk team heeft 10 members, dan heeft de ART totaal 100 members). De ART wordt beheerd door 3 kernrollen: de RTE, de PM en de SA (zie bijlage voor uitleg)

• **Solution Trains (ST's)**: meerdere ART's samen die werken aan één geïntegreerde oplossing (bijv. 10 ART's van 100 FTE= ca. 1.000 FTE in 1 ST). Solution Train is de laag boven de Agile Release Trains. Een Solution Train kent drie kernrollen: de Solution Train Engineer (STE), Solution Management en de Solution Architect.

• **Lean Portfolio Management (LPM)** - Een Portfolio aanpak om wendbaar met vraag en aanbod om te gaan en de wendbaarheid te maximaliseren. Lean Portfolio Management is de laag boven de Solution Trains. De LPM-laag wordt gefaciliteerd door: Epic Owners, Enterprise Architecten, Business Owners en Executive Management.

Dean heeft vervolgens alle voorgaande elementen in 1 grote poster gegoten (zie boven). Deze plaat is te vinden op scaledagileframework.com. Zoek bijvoorbeeld eens de woorden *Enterprise, Visie, Lean Mindset, Agile Mindset, Agile Release Train, Solution Train, Dev(Sec)Ops en Design Thinking*. Je ziet dat de hele poster redelijk logisch is gegroepeerd. Elk icoontje op de plaat is clickable met een uitleg van het icoon. Je kunt dus op het icoontje RTE klikken en lezen hoe SAFe de RTE rol heeft bedacht (helaas, sinds kort, alleen als je een betaald account hebt).

SAFe anno 2025: allang niet meer alleen voor software of alleen ontwikkeling

SAFe is ooit begonnen als framework voor grootschalige softwareontwikkeling, geïnspireerd door het Agile Manifesto van 2001. Inmiddels is het uitgegroeid tot een breed toepasbaar raamwerk voor complexe organisatieveranderingen, ver buiten de IT. SAFe helpt organisaties bij het organiseren van multidisciplinaire samenwerking: van IT tot inkoop, van hardware tot cyber-physical systems, van operatie tot beleid.

Een belangrijk kenmerk is dat SAFe het volledige traject van ontwikkeling tot beheer dekt, dankzij de integratie van DevOps. Elk Agile Team is in feite een DevOps-team dat verantwoordelijk is voor zowel bouwen als beheren (Build & Run). Klassieke frameworks zoals BiSL en ASL2 zijn hierdoor vaak overbodig geworden. SAFe kan samenwerken met ITIL-processen zoals Incident Management of Configuration Management, maar veel van deze praktijken zijn al impliciet opgenomen in SAFe. Het uitgangspunt blijft: als iets niet direct bijdraagt aan waarde, flow of snelle feedback, dan moet je het durven herzien.

Formele definitie van SAFe

Tijdens de SAFe-lessen leer je het volgende antwoord op de vraag 'wat is SAFe?':

"SAFe® 6.0 for Lean Enterprises is 's werelds leidende opensource scaled agile framework voor het behalen van Enterprise Business Agility. SAFe brengt inzichten en praktijken uit Lean, Agile, DevOps en Systems Thinking samen in één geïntegreerd raamwerk. SAFe helpt organisaties innovatieve producten & diensten (1) sneller, (2) met een hogere kwaliteit, (3) hogere betrouwbaarheid en (4) hogere werknemerstevredenheid op de markt te brengen"

Maar op de vraag wat is SAFe, is ook een goed en bondig antwoord:

Met toepassing van het SAFe Framework kunnen we de hele organisatie wendbaarder maken, beter reageren op veranderingen in de wereld en levertijden drastisch inkorten.

Of heel kort…..

SAFe is de geschaalde versie van Scrum

SAFe is een tweede besturingssysteem

Waar zit SAFe ten opzichte van de bestaande hiërarchische organisatie? Dat leg ik je hieronder uit. SAFe bouwt geen tijdelijke projectstructuren naast de bestaande hiërarchie. Nee, SAFe introduceert een **tweede besturingssysteem** dat over alle afdelingen heen werkt ('Cross-Functional'). Dat kan door en over de hele organisatie zijn (meest ideale situatie), of alleen over één of meerdere IT-afdelingen. In de wandelgangen noemen we dat 2de besturingssysteem vaak de ‘SAFe Fabriek’. Daarmee ontstaat samenwerking zónder te verdwalen in bureaucratie en afdelingssilo’s. SAFe verandert dus niets aan de bestaande afdelingen, de bestaande hiërarchie. Dat is heel belangrijk te weten.

Kenmerken van dit tweede systeem en dus kenmerken van SAFe:

• **Er zijn vaste, door SAFe gedefinieerde rollen** als Product Owner, Scrum Master, RTE, Architect.

• **De fundatie van SAFe zijn multidisciplinaire, cross-functionele agile teams** die zichzelf organiseren.

• **Agile teams werken allemaal, tegelijk in een vaste cadans**: zoals iteraties van 2 weken voor agile teams.

• **De ART’s werken ook allemaal tegelijk in een vaste cadans**, met iteraties van twee weken en Planning Intervals (PI’s) van circa tien weken.

• **Vaste door SAFe gedefinieerde vergaderingen die volledig op elkaar zijn afgestemd**: De output van event 1 is input voor event 2. Deze heten bijvoorbeeld de PI Planning, Inspect & Adapt, System Demo, Product Owner Sync, Portfolio Sync, etc.

• **Vaste ontwikkelstraat voor de ontwikkeling van producten en diensten. Met vier standaardfasen die je continu doorloopt**: Exploration, Integration, Deployment en Release. Wat per fase gebeurt, is highlevel door SAFe voorgeschreven en moet echt op maat worden gemaakt per organisatie.

• **SAFe experts prioriteren economisch**: Ze rekenen uit wat de *relatieve* belangrijkheid of costs-of-not-ownership (dat heet 'Cost of Delay') van wens 1 ten opzichte van 2 ten opzichte van 3 is (bijvoorbeeld mbv Fibonacci’s reeks) en bepalen dan, wederom *relatief*, hoeveel werk ze hebben aan wens 1, 2 en 3. Dat delen ze en de hoogste doen ze als eerste, want dat wat de meeste waarde oplevert en je heel snel kunt opleveren, doe je economisch als eerste.

SAFe’s Flow of Value: van missie strategie naar uitvoering in korte cycli

SAFe maakt de reis van idee naar uitvoering overzichtelijk en is door SAFe ontwikkeld en doordacht.

Het werk in SAFe stroomt van:

• De Lean Portfolio Management ('LPM') laag boven aan de SAFe poster

• van Enterprise die een missie naar visie hebben en ‘iets willen’

• naar een organisatie strategie,

• naar strategische thema's met Objectives & Key results. (OKR’s),

• naar portfolios,

• naar Epics die economisch worden geprioriteerd (dat heet ‘WSJF’)

• Naar de Agile Release Trains, midden op de poster

• naar Features (op ART-niveau),

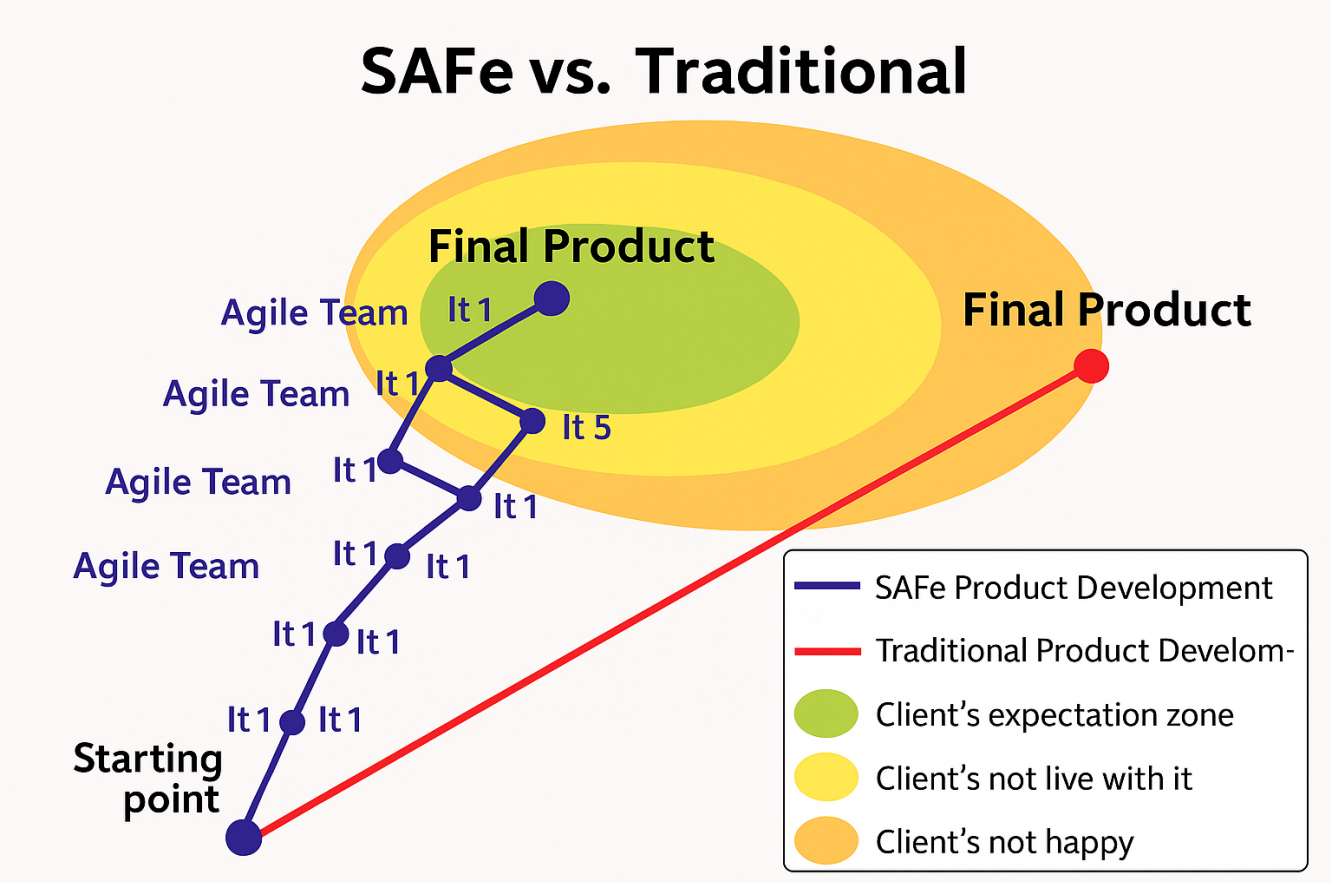
• Naar de Agile Teams onder aan de poster

• naar User Stories (op teamniveau).

• En weer terug als alle User Stories, Features en Epics zijn ontwikkeld *en* geaccepteerd.

Al deze requirements worden visueel weergegeven en georganiseerd op met elkaar verbonden Kanbanborden. ‘Kanban’ is een Japans woord dat letterlijk ‘visueel bord’ betekent en komt uit de Lean-filosofie. In SAFe is dit een van de krachtigste hulpmiddelen: elk niveau: van team tot portfolio: maakt gebruik van een eigen Kanbanbord om werk zichtbaar te maken, te organiseren en te beheren. Doordat deze borden met elkaar gekoppeld zijn, ontstaat een continu en transparant overzicht van de stroom van werk (de flow) door de organisatie zowel werk wat moet worden gedaan en werk wat is afgerond en gaat worden opgeleverd. Hierdoor kunnen teams beter prioriteren, knelpunten tijdig signaleren en sneller waarde leveren. SAFe experts hebben tools en middelen om te meten hoe snel die ‘waarde’ door alle Kanbanborden loopt. Dat heet “Flow”.

In SAFe zijn tijd en geld, kwaliteit fixed: functionaliteit fluctueert.



In tegenstelling tot projecten, staan in SAFe Tijd, geld en kwaliteit vast: Tijd van 1 iteratie van 10 dagen + een vast team van mensen tegen vaste kosten met vaste kwaliteit (dat noemen agilisten Definition of Ready en Definition of Done). Alleen wat er haalbaar is om te bouwen in die 2 weken is flexibel, dus de hoeveelheid functionaliteit die het team gaat opleveren. Klanten kunnen en mogen in SAFe elke 2 weken hun wensen aanpassen. Daar kan SAFe uitstekend mee omgaan, *afwijkingsrapportages* kent SAFe niet.

Bij traditionele projecten is T,G,K anders, waardoor projecten kunnen ‘uitlopen’. Een project krijgt fixed geld, fixed scope, fixed functionaliteit en dat gaat het Project Team proberen te bouwen. In de praktijk levert dat vaak ellende op: na twee jaar krijg je iets waar niemand om heeft gevraagd.

SAFe implementeren = leren

SAFe is geen kant-en-klare werkaanpak die je uitpakt en direct begint. Het is veelal toch maatwerk en vooral een leerproces voor iedereen in de SAFe Fabriek. Je begint meestal klein met één ART en groeit dan. Als je SAFe implementeert verandert er een en ander voor zittend management en zeker ook voor de gebruikers en voor de werknemers, alhoewel die verandering in de praktijk voor deze laatste groep minder groot is dan de meesten denken. Immers, software ontwikkelaars ontwikkelen software en blijven dat ook doen. Het enige verschil nu is dat er een paar standaard ceremonies zijn, waar die software ontwikkelaar dan wordt verwacht.

Resumé: hoe SAFe verschilt van PRINCE2 en MSP

Onderdeel	Waterval / PRINCE2 / MSP	SAFe
Resourcing	Tijdelijke projectteams	Vaste Agile Teams met stabiele samenstelling
Werkwijze	Alles vooraf plannen	Iteratief plannen en steeds bijstellen
Gebruikers	Begin & einde	Continu betrokken
Leiderschap	Top-down, hiërarchisch	Faciliterend, dienend leiderschap
Flexibiliteit	Lage aanpasbaarheid	Hoge wendbaarheid
Resultaten	Grote opleveringen in één keer	Kleine werkende resultaten elke 2 weken
Focus	Focus op tijdelijke projecten die ‘iets opleveren’	Focus op Producten die onderhouden worden door ‘long lived agile teams’. Er zijn geen start stops meer en geen constante wisselingen

SAFe heeft eigen termen: maar die leer je snel

SAFe gebruikt termen die in het begin wat ‘Amerikaans’ aanvoelen en waarbij je mogelijk denkt *‘dat de SAFe expert in een parallel universum leeft'*. Datzelfde gevoel hebben de SAFe experts overigens als ze voor ASML, de Politie of andere grote organisaties gaan werken. Vaak duurt het een jaar voordat je de ‘taal’ machtig bent en iedereen snapt.

Maar goed nieuws voor iedereen: met enkele tientallen woorden en begrippen kun je al prima en als echte SAFe expert meepraten! Dus wees niet bang, in de praktijk wennen de nieuwe woordjes erg snel.

Conclusie: waarom grote organisaties niet zonder SAFe kunnen

SAFe helpt grote organisaties om wendbaar te worden zonder grip en structuur te verliezen. In een wereld die steeds sneller verandert, zijn klassieke projectmethodes zoals PRINCE2 en MSP te traag en inflexibel. SAFe biedt vaste rollen, teams en ritmes, maar laat tegelijk ruimte voor continue bijsturing. En zo implementeer ik Target Operating Models ook vaak bij organisatie eenheden: we weten nu al dat dat allesbehalve in een rechte lijn gaat lopen. Daarom kijken we elke 2 weken (= iteratie) hoe alles loopt en passen dat aan waar nodig. En kijken dan de opvolgende iteratie of een en ander dan wel goed uitpakt.

Daardoor kunnen SAFe organisaties sneller reageren op veranderingen, en vaker waarde leveren aan hun klanten of burgers. SAFe doorbreekt silo’s en maakt samenwerking over afdelingen heen mogelijk. Het is geen hype, maar een bewezen aanpak om grote organisaties toekomstbestendig te maken.

Bijlage: belangrijkste begrippen in SAFe

Hieronder de belangrijkste begrippen van SAFe, versie 6.0-compliant.

SAFe Lagen

• **Portfolio laag (LPM)**: kiest de juiste Epics en verdeelt budget

• **Large Solution laag (ST)**: coördineert meerdere ART's

• **Essential Laag** - bevat ART's en Agile teams (minimale configuratie van SAFe)

Belangrijkste rollen

• **De portfolio function -** de functie die Epics selecteert en strategisch stuurt

• **Epic Owner**: eigenaar van een Epic

• **Business Owner (BO)**: verantwoordelijke business stakeholder

• **Solution Train**: meerdere ART's samen

• **ART**: Agile Release Train (team van teams)

• **RTE**: Release Train Engineer: eigenlijk beetje de Scrum Master van de ART. De Master of the Scrum Masters.

• **SA**: System Architect: technisch geweten van de ART

• **PM**: Product Manager: eigenaar van de Features en de Super Product Owner.

• **PO**: Product Owner: eigenaar van de Stories en werkt nauw samen met de PM

• **SM**: Scrum Master/Teamcoach en werkt nauw samen met de RTE

• **Shared Services**: specialisten die meerdere teams ondersteunen, bv. security, legal

Belangrijkste events

• **PI Planning**: het belangrijkste event: teams plannen samen het werk voor de komende PI

• **Inspect & Adapt (I&A)**: evaluatie en verbetersessie aan het einde van een PI

• **System Demo**: gezamenlijke demo van werkende software

• **Innovation & Planning Iteration (IP-Iteratie)**: ruimte voor innovatie en verbetering

Processen en concepten

• **PI**: Planning Interval (ca. 10 weken)

• **Iteration**: 2 weken werken, daarna demo

• **Value Stream**: stroom van waarde voor de klant

• **Agile Product Delivery**: snel én duurzaam waarde leveren

• **DevOps**: samenwerking tussen Dev & Ops

• **Continuous Delivery Pipeline**:

• Continuous Exploration

• Continuous Integration

• Continuous Deployment

• Release on Demand

• **Design Thinking**: ontwerpen vanuit de klant

• **Customer Centricity**: klant staat centraal

• **Relentless Improvement**: een SAFe kernwaarde: continu verbeteren

Belangrijkste artifacten

• **LPM Kanban** - bord op portfolio niveau

• **ART Kanban**: bord voor ART-niveau

• **Team Kanban**: bord voor werk op teamniveau

• **Epic**: groot initiatief met business waarde waar SAFe mee start.

• **Feature**: onderdeel van een Epic, realiseerbaar binnen één PI

• **User Story**: klein werkpakket voor het team

• **Enabler**: werk dat technische voorbereiding of compliance vereist

• **Capacity Allocation**: verdeling van tijd tussen Run en Change (DevOps)

• **Built-in Quality**: kwaliteit is ingebouwd in elk onderdeel

• **Lean Budgeting**: wendbaar budgetteren met financiering op waarde

4. SAFe is een Scaled Agile Framework, maar wat schaalt er eigenlijk?

Introductie

In een steeds complexer wordende wereld moeten organisaties steeds wendbaarder worden. Marktontwikkelingen volgen elkaar snel op: COVID, geopolitieke spanningen, de energietransitie, AI en digitalisering veranderen complete sectoren in hoog tempo.

Voor kleine organisaties is het vaak relatief eenvoudig om zich aan te passen. Maar naarmate organisaties groter worden, wordt verandering complexer. Meer teams, meer systemen en meer afhankelijkheden maken het moeilijker om snel te reageren.

Daarom hebben grote organisaties een model nodig waarmee zij Agile kunnen opschalen naar enterprise-niveau. Het Scaled Agile Framework (SAFe) biedt hiervoor een bewezen structuur. SAFe schaalt niet één ding, maar **een volledig operating model**. Maar dat roept meteen een interessante vraag op: SAFe is een **Scaled Agile Framework**, maar wat schaalt er dan precies van wat naar waar?

Het antwoord is: veel meer dan alleen teams, wat de meesten wel kennen en begrijpen. In SAFe schalen meerdere dimensies tegelijk zoals o.a.: organisatiestructuren, rollen, planning, ceremonies, backlogs, waarde-levering en zelfs de Definition of Done. In dit hoofdstuk leg ik uit wat er exact schaalt.

1. Organisatiestructuur schaalt

De meest zichtbare vorm van scaling in SAFe is de organisatiestructuur.

Waar Agile vaak begint met één team, schaalt SAFe dit op naar meerdere organisatielagen:

Agile Team → Agile Release Train (ART) → Solution Train → Lean Portfolio Management (LPM)

• Agile Teams leveren werkende software of systemen in korte iteraties.

• Meerdere teams vormen samen een Agile Release Train, een virtuele organisatie van teams die gezamenlijk waarde leveren.

• Wanneer meerdere Agile Release Trains samenwerken aan één groot systeem of product ontstaat een Solution Train.

• Op het hoogste niveau zorgt Lean Portfolio Management (LPM) voor strategische sturing en investeringsbeslissingen.

Hierdoor kan een organisatie groeien van enkele teams naar honderden teams, terwijl de Agile principes behouden blijven.

2. Rollen schalen

Met de organisatiestructuur schalen ook de rollen mee. In SAFe ontstaan rollen op meerdere niveaus rond drie hoofddomeinen: process leadership, product management en architecture.

Process

Scrum Master → Release Train Engineer → Solution Train Engineer → LPM / Executive leadership

Content

Product Owner → Product Manager → Solution Manager → Epic Owner

Technology

System Architect → Solution Architect → Enterprise Architect

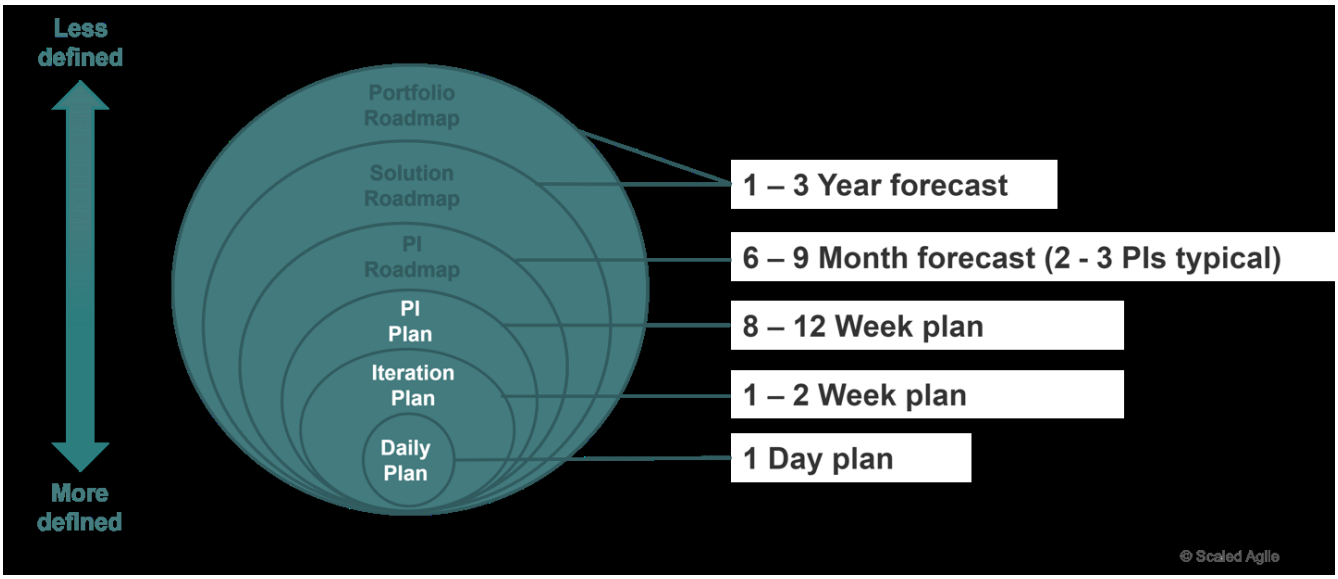
Elke rol heeft een vergelijkbare verantwoordelijkheid, maar dan op een groter schaalniveau.

3. Roadmaps schalen

Planning schaalt in SAFe van strategisch naar operationeel.

Op **portfolio-niveau** worden langetermijnroadmaps opgesteld die vaak 1 tot 3 jaar vooruitkijken.
Op **ART-niveau** wordt de roadmap concreter gemaakt en vertaald naar Planning Intervals (PI's).
Tijdens **PI Planning** plannen teams het werk voor de komende PI.
Binnen iteraties maken teams hun **iteration plans**, en dagelijks wordt dit verfijnd in de **Team Sync** (Daily Scrum).

Zo ontstaat een planningstructuur die loopt van **strategie tot dagelijkse uitvoering**.



4. Ceremonies schalen en verbinden de organisatie

SAFe schaalt niet alleen structuren, maar ook de Agile events. Deze events zorgen voor **synchronisatie tussen teams en organisatieniveaus**.

Voorbeelden van opgeschaalde ceremonies:

Teamniveau

• Team Sync

• Iteration Review

• Iteration Retrospective

ART-niveau

• ART Sync (PO Sync + Coach Sync)

• System Demo

• Inspect & Adapt

De **Team Sync** synchroniseert het team, terwijl de **ART Sync** teams op ART-niveau synchroniseert.
De **Iteration Review** wordt opgeschaald naar de **System Demo**, waarin de geïntegreerde oplossing van alle teams wordt getoond.

Events functioneren daarmee als het **ritme van de organisatie**.

5. Backlogs schalen

In SAFe bestaan meerdere backlogs die met elkaar verbonden zijn.

Portfolio Backlog
Bevat strategische initiatieven in de vorm van **Epics**.

Solution Backlog
Bevat **Capabilities** voor grote systemen.

ART Backlog
Bevat **Features** die door ART's worden gerealiseerd.

Team Backlog
Bevat **User Stories** en **Enabler Stories** die door Agile Teams worden ontwikkeld.

Elke backlog vertegenwoordigt een verfijning van werk van **strategie naar uitvoering**.

6. Het Requirements Model schaalt

Het SAFe Requirements Model ondersteunt deze opschaling.

Werk wordt stap voor stap gedecomposeerd:

Epic → Capability → Feature → Story → Task / Sub-task (in geval van SAFe's full configuratie)

Hierdoor blijft werk beheersbaar, terwijl tegelijkertijd **traceability** ontstaat van strategische initiatieven tot concrete teamactiviteiten.

7. Value delivery schaalt

Een van de belangrijkste principes in SAFe is dat waarde incrementeel wordt geleverd.

• Agile Teams leveren in elke iteratie een **Team Increment**.

• Binnen een ART worden deze geïntegreerd tot een **System Increment**.

• Wanneer meerdere ART's samenwerken ontstaat een **Solution Increment**.

• Op portfolioniveau vertaalt dit zich uiteindelijk naar de realisatie van Epics en strategische business outcomes en dus indirect naar de visie en missie van de organisatie.

Met andere woorden: waarde wordt op elk niveau geïntegreerd en opgeschaald.

8. Definition of Done schaalt

In SAFe wordt kwaliteit niet alleen op teamniveau geborgd, maar opgeschaald over meerdere niveaus.

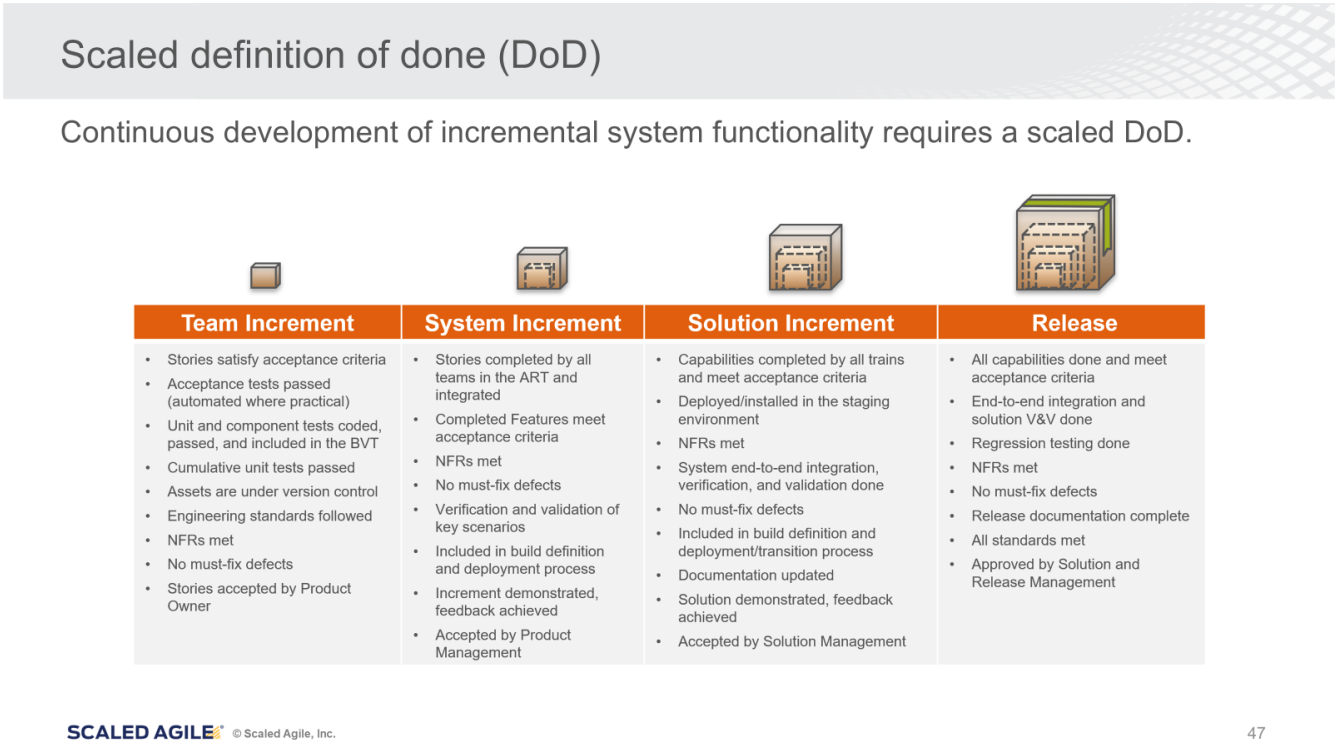
• Een User Story is "Done" wanneer deze voldoet aan de team-DoD en resulteert in een **Team Increment**.

• Wanneer alle teams binnen een ART hun werk integreren ontstaat een **System Increment**, waarbij Features en systeemkwaliteit worden gevalideerd.

• In grotere omgevingen worden meerdere ART-resultaten samengebracht tot een **Solution Increment**.

• Pas wanneer integratie, testen, documentatie en governance volledig zijn afgerond kan een **Release** plaatsvinden.

Deze gelaagde Definition of Done voorkomt dat integratie- en kwaliteitsproblemen pas laat in het proces zichtbaar worden.



9. De LACE schaalt

Tot slot schaalt ook de transformatieorganisatie zelf.

De **Lean-Agile Center of Excellence (LACE)** ondersteunt de implementatie en adoptie van SAFe.

In grotere organisaties wordt vaak een **Hub-and-Spoke model** toegepast, zie plaatje onder: Centrale LACE → meerdere Local LACE's. Ook wel Enterprise LACEs versus Operational LACEs genoemd.

Hiermee kan de Lean-Agile transformatie gecoördineerd en consistent worden opgeschaald over meerdere onderdelen van de organisatie (zie hoofdstuk 50, 'SAFe transformaties met het SAFe LACE Hub-and-Spoke model').



10. Het VMO schaalt

In grotere SAFe-implementaties ontstaat vaak ook een Value Management Office (VMO). Hoewel dit geen formele schaling-structuur is, wordt dit in de praktijk wel regelmatig als ook een Hub-and-Spoke model toegepast, vergelijkbaar met een LACE. Hierbij bestaat een centrale Enterprise VMO (E-VMO) die portfolio governance en value management verzorgt, terwijl meerdere Operational VMO's (O-VMO's) dichter bij de Agile Release Trains opereren en verantwoordelijk zijn voor value tracking, flow metrics en operationele portfolio alignment.

Conclusie

Wanneer we spreken over "scaling" in SAFe, gaat het dus niet alleen om het opschalen van teams.

SAFe schaalt tegelijkertijd:

1. Organisatiestructuur

2. Rollen

3. Planning horizons

4. Ceremonies

5. Backlogs

6. Requirements

7. Value delivery

8. Definition of Done

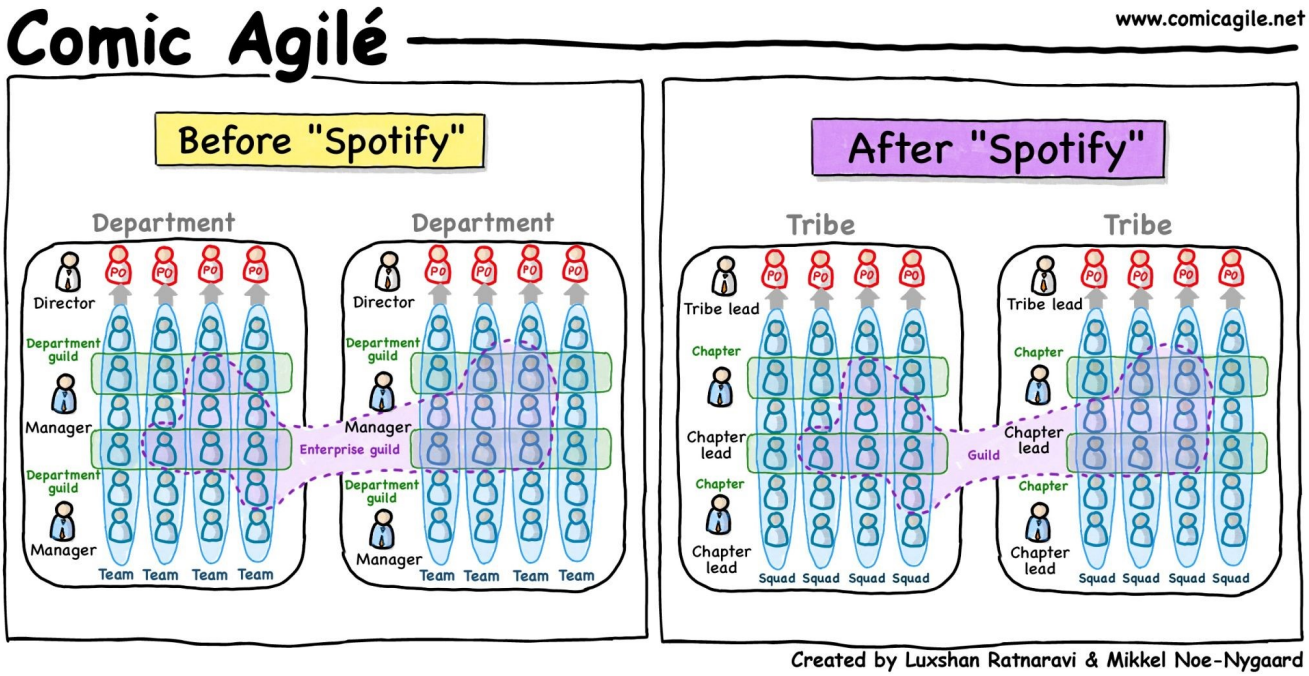
9. LACE

10. VMO

SAFe schaalt een compleet operating model: structuur, rollen, planning, synchronisatie, requirements, value delivery en governance.

Door deze elementen consistent met elkaar te laten schalen, ontstaat een organisatie die met tientallen tot honderden teams tegelijk kan werken aan dezelfde strategische doelen, zonder de samenhang tussen strategie, uitvoering en waardecreatie te verliezen.

5. Waarom SAFe en niet het Spotify-model? Zeven argumenten voor enterprise-organisaties



Inleiding

Als SAFe-expert kom ik regelmatig bij bedrijven over de vloer. Vaak hoor ik dan dat ze “het Spotify-model” hebben toegepast of willen toepassen. Op dat moment stel ik mezelf direct drie vragen: hoe is dat besluit tot stand gekomen, wie heeft dat besluit genomen, en is die keuze bewust afgezet tegen een alternatief zoals SAFe®? Daar ben ik altijd benieuwd naar.

In vrijwel elke enterprise-transformatie komt de vraag op tafel: "Waarom SAFe® en niet het Spotify-model?" Het is een begrijpelijke vraag. Het Spotify-model oogt aantrekkelijk: autonome teams, hippe namen als squads en tribes en de belofte van startup-snelheid in een grote organisatie. Maar schijn bedriegt. In dit hoofdstuk zet ik zeven concrete argumenten uiteen waarom SAFe® voor enterprise-organisaties de betere keuze is, en wat de consequenties zijn als je tóch voor het Spotify-model kiest.

Dit hoofdstuk richt zich op enterprise-organisaties: grote, complexe organisaties met honderden tot duizenden medewerkers, meerdere afdelingen, gereguleerde omgevingen en de noodzaak tot strategische sturing op portfolio-niveau. Denk aan defensieorganisaties, overheidsinstanties, banken, verzekeraars en grote industriële bedrijven. Voor een startup van twintig man kunnen de overwegingen anders liggen. Maar voor wie schaal, continuïteit en governance essentieel zijn, zijn de zeven argumenten hieronder doorslaggevend.

Context: wat is het Spotify-model eigenlijk?

Laten we beginnen met een ongemakkelijke waarheid: het Spotify-model is geen framework. Het is een beschrijving, een momentopname, van hoe Spotify op een bepaald moment georganiseerd was. In 2012 publiceerden Henrik Kniberg en Anders Ivarsson een whitepaper en twee video's over hun werkwijze met squads, tribes, chapters en guilds. Dat is het. Geen handleiding, geen implementatiegids, geen doorontwikkeling.

Joakim Sundén, agile coach bij Spotify van 2011 tot 2017, zei hierover: *"Even at the time we wrote it, we weren't doing it. It was part ambition, part approximation. People have really struggled to copy something that didn't really exist."*

Jeremiah Lee, voormalig product manager bij Spotify, schreef in zijn veelgeciteerde artikel *"Failed #SquadGoals"*: het squad-model heeft gefaald bij Spotify zelf, en het zal ook bij jouw bedrijf falen. Spotify is inmiddels grotendeels overgestapt naar meer traditionele managementstructuren.

Dit alles roept bij mij als SAFe-expert en implementatieconsultant een fundamentele vraag op: waarom kiezen organisaties dit wetende dan tóch voor het Spotify-model? Hieronder zet ik zeven concrete argumenten uiteen waarom ik die keuze, zeker voor enterprise-organisaties, niet verstandig vind. Niet als dogma, maar vanuit de praktijk.

Argument 1: theoretische fundering: Body of Knowledge ↔ twee video's

SAFe® heeft een complete, doorontwikkelde body of knowledge, het Spotify-model heeft twee YouTube-video's en een whitepaper uit 2012. Dat is geen gelijkwaardig alternatief; dat is geen fundament voor een enterprise-transformatie.

SAFe® is doorontwikkeld naar versie 6.0, aangescherpt op basis van duizenden real-world implementaties en wetenschappelijke inzichten uit Lean, Agile en Systems Thinking. Elke rol, elk artifact, elke ceremony staat gedocumenteerd op scaledagileframework.com. Er is altijd een autoritatieve bron om op terug te vallen.

Het Spotify-model heeft geen van dat alles. Geen website, geen versiehistorie, geen doorontwikkeling, geen community. De co-auteurs zeggen het zelf: kopieer het niet. Een enterprise die haar transformatie bouwt op dit fundament, bouwt op drijfzand.

Argument 2: Certificerings-ecosysteem: aantoonbare kennis ↔ onverifieerbare claims

SAFe® heeft wereldwijd honderdduizenden gecertificeerde professionals:

• **SPC's** (SAFe Practice Consultants) die bewezen hebben dat ze een transformatie kunnen leiden

• **RTE's** (Release Train Engineers) die aantoonbaar weten hoe ze een ART moeten runnen

• **SPCT's** (SAFe Practice Consultant Trainers) die zowel kunnen trainen als coachen op het hoogste niveau

• Daarnaast tienduizenden gecertificeerde SA's, PO/PM's, Scrum Masters en meer

Bij het Spotify-model: Hoe bewijs je dat iemand het "beheerst"? Er is geen certificering, geen assessment, geen body of knowledge om tegen te toetsen. Bij een intake heb je geen enkele objectieve maatstaf om vakkennis te beoordelen.

De consequentie: Je kunt niet objectief vaststellen of de expert die je inhuurt daadwerkelijk over de juiste kennis beschikt. Dat is een risico dat je als enterprise niet wilt lopen.

Argument 3: Beschikbaarheid van experts: grote internationale markt ↔ nauwelijks vindbaar

SAFe® is het meest geadopteerde scaling framework ter wereld. Dat betekent dat je als enterprise relatief eenvoudig ervaren mensen kunt vinden, SPC's, RTE's, Product Managers, System Architects, die eerder SAFe-transformaties hebben gedaan bij vergelijkbare organisaties. De markt is groot, divers en internationaal.

Bij het Spotify-model: Probeer maar eens een "Spotify-expert" te vinden met een bewezen track record bij meerdere enterprise-implementaties. Die markt is minuscuul. En als je er al eentje vindt, wat heeft diegene dan precies geleerd? Van twee video's en een whitepaper?

Argument 4: Continuïteitsrisico: vervangbaar ↔ single point of failure

Als je als grote organisatie je transformatie bouwt op een model waarvoor nauwelijks experts beschikbaar zijn, dan creëer je een single point of failure. Vertrekt je ene "Spotify-guru," dan zit je met een probleem. Wie neemt het over? Waar vind je vervanging? En spreekt die vervanger dezelfde taal, hanteert die dezelfde interpretatie?

Bij SAFe® kun je vervangen, opschalen en bijscholen, omdat er een gestandaardiseerd kennisplatform is waar iedereen uit dezelfde bron leert. Een nieuwe RTE kan direct aan de slag omdat de rolomschrijving, de ceremonies en de werkwijze gedocumenteerd zijn.

De consequentie voor grote, gereguleerde organisaties: Continuïteit van de transformatie is essentieel. Je kunt het je niet veroorloven dat je hele verandertraject stil komt te liggen omdat één persoon vertrekt. SAFe® biedt die continuïteit; het Spotify-model niet.

Argument 5: Volledigheid: van team tot portfolio ↔ alleen team/ART-level

Dit is misschien wel het sterkste argument. SAFe® dekt het hele spectrum van enterprise-schaal: van Team level, via ART (programma) level en Solution level tot aan Portfolio level. Lean Portfolio Management, Solution Trains, Strategic Themes, Participatory Budgeting, Lean Business Cases, Value Stream identification, het is allemaal uitgedacht, gedocumenteerd en beproefd.

Het Spotify-model geeft je in feite alleen een teamstructuur (squads, tribes) en een stukje coördinatie. Vertaald naar SAFe-termen: je hebt iets op team- en ART-level, en daarna houdt het op.

En dan? Dan moet een enterprise met duizenden medewerkers *zelf* gaan uitvinden:

• Hoe je portfolio management doet

• Hoe je budgettering aanpakt

• Hoe je strategische prioriteiten vertaalt naar uitvoering

• Hoe je Solution Trains inricht als je meerdere ARTs hebt

• Hoe je waardestroom-identificatie doet

• Hoe je Lean governance inricht

Argument 6: kosten en risico: customization van bestaand framework ↔ invention op enterprise-schaal

Het voorgaande argument heeft directe financiële consequenties. Bij SAFe® is het implementatiemodel helder: je neemt het framework, je past het aan op je context, en je gaat aan de slag. De blauwdruk is er. De best practices zijn er. De valkuilen zijn gedocumenteerd.

Bij het Spotify-model begin je met een leeg vel papier voor alles boven team-level. Elke keuze die je maakt moet je zelf valideren. Elke fout die je maakt kost tijd en geld. En in een enterprise-context praten we dan niet over weken, maar over maanden en jaren.

Het bekendste Nederlandse voorbeeld is ING, dat in 2015 een grote transformatie doormaakte geïnspireerd door het Spotify-model, begeleid door McKinsey. Na zeven jaar spraken ING-techleiders over "Death of the Spotify Model" als presentatietitel. ING verbeterde uiteindelijk zijn aanpak door alsnog te kiezen voor een volwaardig framework: LeSS. Ze ontdekten dat ze een nieuw type silo hadden gecreëerd rondom teams met relatief rigide missies. Precies het probleem dat SAFe® met zijn uitgewerkte portfolio- en governancestructuur voorkomt.

Argument 7: Tooling-ecosysteem: enterprise tools gebouwd voor SAFe® ↔ zelf aanpassen of zonder zitten

De enterprise agile planning-markt is gebouwd rondom SAFe®. Jira Align, de enterprise agile planning-tool van Atlassian, is letterlijk ontwikkeld als SAFe-hulptool. Planning Intervals, ARTs, Features, Enablers, Strategic Themes, Lean Portfolio Management: het zit er allemaal in, in dezelfde taal als het framework. Als je SAFe® kent, kun je Jira Align relatief snel inrichten en gebruiken.

En dat geldt niet alleen voor Jira Align: Digital.ai, Planview, Targetprocess en ServiceNow zijn eveneens voorbeelden. Al deze enterprise agile planning-tools zijn gebouwd of geoptimaliseerd voor SAFe®, omdat dat het marktleidende framework is. Softwareleveranciers investeren hun ontwikkelbudget in wat de meeste klanten vragen.

Bij het Spotify-model heb je twee opties, en allebei zijn ze slecht:

1. **Optie 1:** Je koopt enterprise tooling die gebouwd is voor SAFe® en gaat die aanpassen aan je Spotify-structuur. Dat kost customization, geld, onderhoud, en bij elke tool-update loop je het risico dat je aanpassingen breken.

2. **Optie 2:** Je bouwt je eigen tooling of werkt met spreadsheets en losse Jira-boards. Dan heb je op enterprise-schaal geen geïntegreerd zicht op je portfolio, je waardestroom en je dependencies.

De consequentie: Met het Spotify-model val je buiten het tooling-ecosysteem. Je betaalt voor software die je niet optimaal kunt gebruiken, of je mist essentiële enterprise-functionaliteit.

De zeven argumenten kort samengevat:

#	Argument	SAFe®	Spotify-model
1	Theoretische fundering	Complete body of knowledge, continu doorontwikkeld	Twee video's en een whitepaper uit 2012
2	Certificerings-ecosysteem	Honderdduizenden gecertificeerde professionals	Geen certificering, geen assessment
3	Beschikbaarheid van experts	Grote, internationale markt	Nauwelijks vindbaar
4	Continuïteitsrisico	Vervangbaar door gestandaardiseerde kennis	Single point of failure
5	Volledigheid	Van team tot portfolio level	Alleen team/ART-level
6	Kosten en risico	Customization van bestaand framework	Invention op enterprise-schaal
7	Tooling-ecosysteem	Enterprise tools gebouwd voor SAFe®	Zelf aanpassen of zonder zitten

Conclusie

Het Spotify-model is geen framework, het is een momentopname van hoe één tech-bedrijf op één moment georganiseerd was, die zelfs door dat bedrijf zelf niet meer wordt gebruikt. Voor enterprise-organisaties, zeker in gereguleerde sectoren zoals defensie, de financiële sector en de overheid, biedt het onvoldoende basis voor een duurzame transformatie.

SAFe® is een body of knowledge. Het biedt een complete, doorontwikkelde, bewezen aanpak van team tot portfolio level, ondersteund door een wereldwijd ecosysteem van gecertificeerde experts, uitgebreide tooling en een actieve community. SAFe® geeft je het fundament, jij bepaalt hoe je het toepast. Bij het Spotify-model moet je alles boven team-level zelf uitvinden, met alle kosten, risico's en vertragingen van dien.

De keuze zou helder moeten zijn in mijn ogen.

Een eerlijke noot tot slot

Ik realiseer me dat dit hoofdstuk een stammenstrijd kan uitlokken en ik ken de tegenargumenten. SAFe® is commercieel want Scaled Agile Inc. verdient aan elke certificering en training. SAFe® is groot en complex: tientallen rollen, ceremonies en lagen die voor sommigen meer weghebben van bureaucratie dan van wendbaarheid. SAFe® kan autonomie ondermijnen: waar het Spotify-model vertrouwen en eigenaarschap uitstraalt, voelt SAFe® voor teams soms als iets wat van bovenaf wordt opgelegd. En ja, er zijn mensen die zeggen dat het framework expres ingewikkeld is gehouden zodat je altijd consultants nodig hebt.

Die kritiek neem ik serieus. Maar er is een verschil tussen een framework met nadelen en de afwezigheid van een framework. Mijn punt is niet dat SAFe® perfect is. Mijn punt is dat je als enterprise een fundament nodig hebt, en dat het Spotify-model dat fundament simpelweg niet biedt.

Als jij goede ervaringen hebt met het Spotify-model op enterprise-schaal: ik hoor het graag. Elke case die het tegendeel bewijst, maakt mij als practitioner scherper.

Bronvermelding

• Lee, J. (2020). *"Failed #SquadGoals: Spotify doesn't use the Spotify model and neither should you."* jeremiahlee.com

• Kniberg, H. & Ivarsson, A. (2012). *"Scaling Agile @ Spotify."* Whitepaper.

• Ramos, C. (2019). *"ING improves on the Spotify Model using LeSS."* LeSS Conference keynote.

• Scrum.org (z.d.). *"Spotify Model, 10 lessons in transplantology."*

• McKinsey & Company (2017). *"ING's agile transformation."* Interview met Peter Jacobs en Bart Schlatmann.

• Martyoo (2024). *"Stop SpotiSAFe."* Medium.

• Scaled Agile, Inc. (2024). *SAFe® 6.0 Framework.* scaledagileframework.com

6. De evolutie van Agile: van Darwin tot SAFe

Elders in dit boek heb ik je verteld over wat het betekent om ‘a true Agilist’ te zijn en heb ik je ook nog een gedetailleerde uitleg gegeven over SAFe met tips naar waardevol zelfstudiemateriaal. Daarvoor verwijs ik je naar hoofdstuk 30, ‘De essentiële kenmerken van een ware Agilist’, en hoofdstuk 1, ‘Het DNA van SAFe’.

In dit hoofdstuk ga ik de volgorde van enkele belangrijke termen en concepten in de Agile wereld beschrijven, namelijk: Darwin, Lean, Scrum, Agile, DevOps en SAFe. Ik zal uitleggen hoe deze begrippen met elkaar verbonden zijn, wat ze precies inhouden en welke input en output ze voor elkaar betekenen. Al deze termen zijn onderling verbonden, maar hoe? Wat kwam eerst, en wat volgde later? Kennis hiervan is belangrijk voor het vormen van je algemene Agile basis.

Ik begin dit hoofdstuk met een chronologische opsomming van de belangrijkste stromingen en hun oprichters, gevolgd door een beschrijving van hun onderlinge samenhang.

Daarnaast geef ik je in een paar gevallen boekentips voor meer informatie over elk onderwerp. Rest me je tot slot veel leesplezier toe te wensen!

Volgorde van de verschillende stromingen

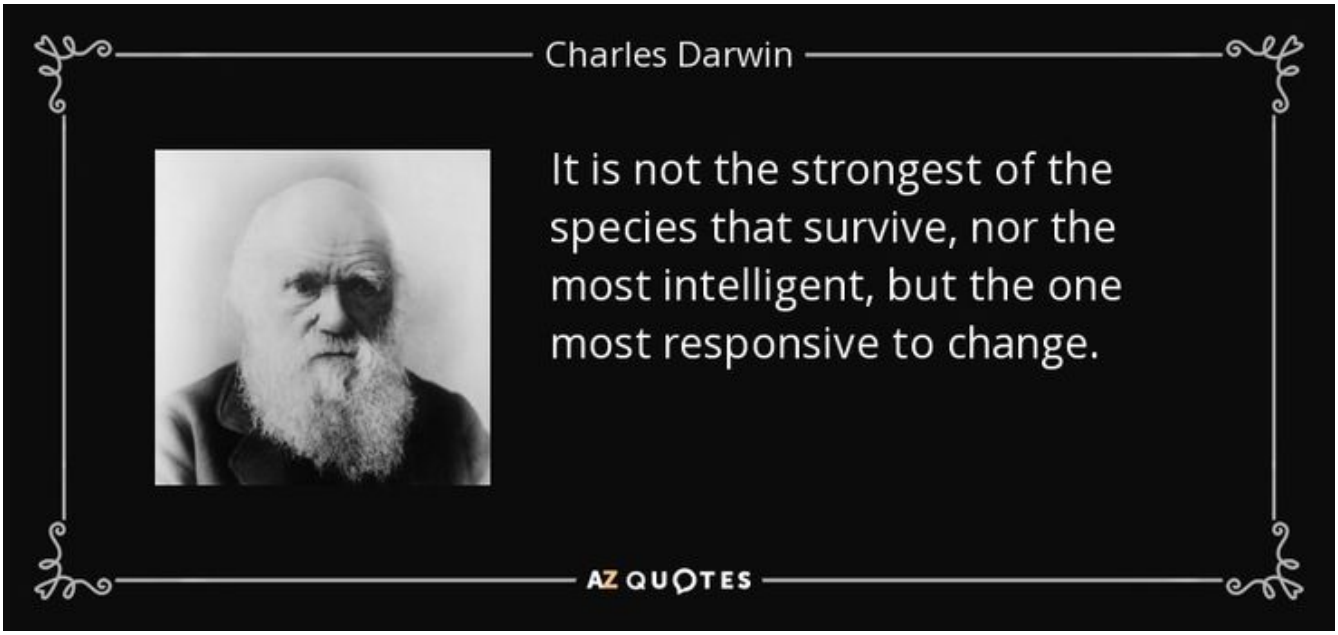
Dit zijn de belangrijkste stromingen die ik in dit hoofdstuk met je wil bespreken, wie de grondleggers zijn, wanneer de stromingen ontstonden en wat ze precies inhouden.

Stroming	Jaar	Founding Fathers	Wat is het ?
Evolutietheorie	1859	Charles Darwin	Mindset en filosofie
Lean	1940s-1950s	Taiichi Ohno, Shigeo Shingo (Toyota Production System, afgekort TPS)	Filosofie en Methodologie
Scrum	1990’s	Jeff Sutherland en Ken Schwaber	Empirical Process model, een agile werkwijze
Agile	2001	17 software-experts, incl. Kent Beck, Ward Cunningham, Jeff Sutherland	Behavior, Principles en Mindset
DevOps	2009	Patrick Debois, Gene Kim, John Allspaw	“A mindset, culture of shared responsibilities and best practices” volgens SAFe.
SAFe	2011	Dean Leffingwell	Een Framework voor het bereiken van “Enterprise Business Agility"
Dev*Sec*Ops	2012-2014	Shannon Lietz, John Willis, e.a.	Zie boven, maar nu incl security

Beschrijving samenhang

Darwin’s evolutietheorie (1859)

Charles Darwin introduceerde halverwege de 19e eeuw het principe van "natuurlijke selectie" (later ook bekend als "survival of the fittest", een term van Herbert Spencer die Darwin overnam), dat de basis legt voor de noodzaak van adaptief vermogen in elke stroming die volgde. De agile basis begint feitelijk bij Darwin: “Respond to Change”



Lean gebaseerd op adaptatie en efficiëntie (1940s-1950s)

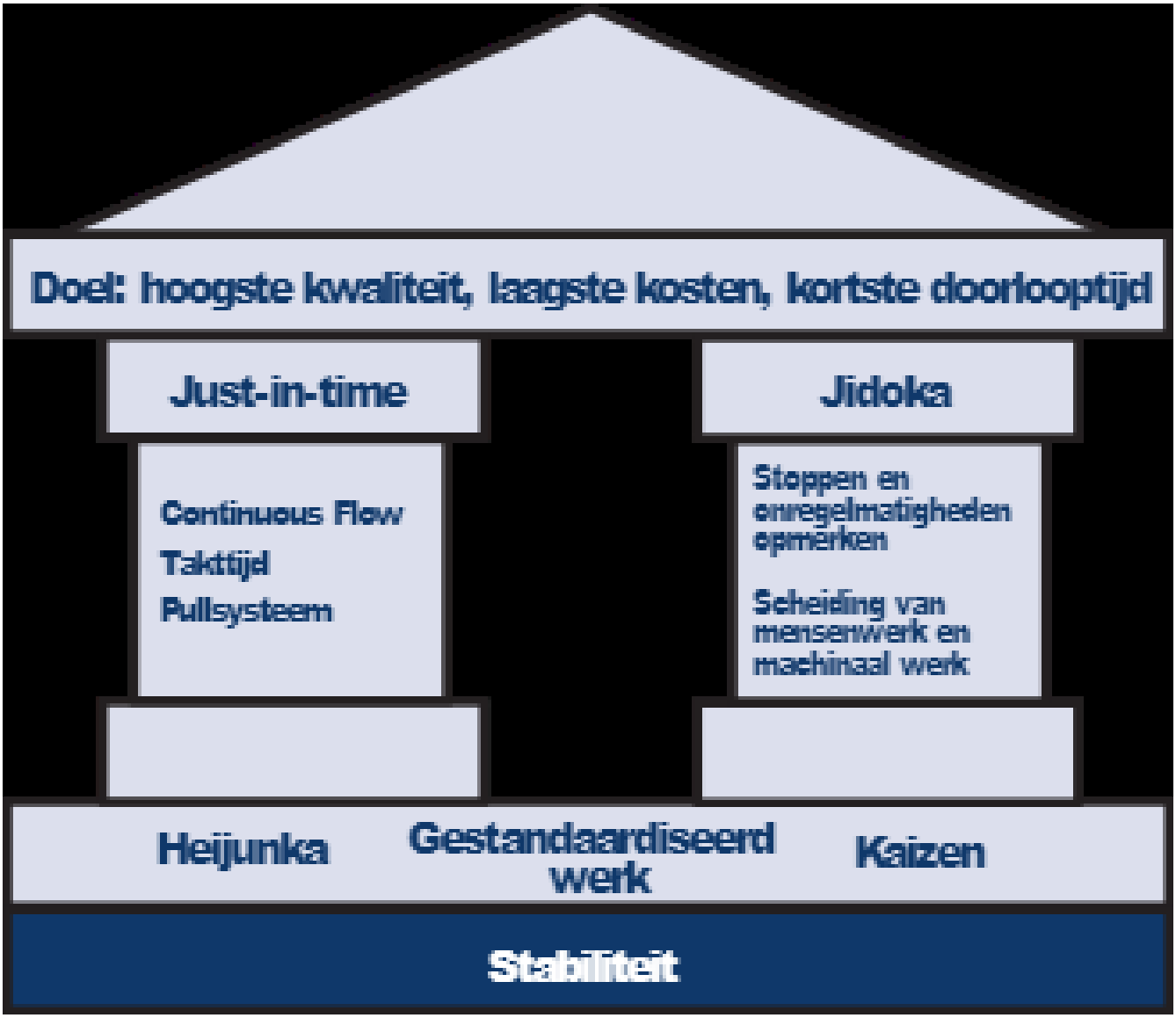
Lean is een filosofie en methodologie die zich richt op het maximaliseren van klantwaarde door verspilling te minimaliseren. Het is ontstaan binnen het Toyota Production System (TPS) in de jaren 1940 en 1950, ontwikkeld door Taiichi Ohno en Shigeo Shingo. TPS legt de nadruk op het verbeteren van de efficiëntie en het stroomlijnen van processen, waardoor de productiviteit toeneemt en de kwaliteit verbetert. Belangrijke concepten binnen Lean zijn Kaizen (continue verbetering) en Value Stream Mapping (het visualiseren van processtappen en elimineren van Waste). Lean heeft invloed gehad op andere methodologieën zoals Agile en DevOps.

Mijn Lean-boekentips:

• "The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses" van Eric Ries.

• “Lean Thinking: Banish Waste and Create Wealth in Your Corporation" van James P. Womack en Daniel T. Jones.

• “The Toyota Way” met de 14 fundamentele veranderprincipes van Jeffrey Liker



Scrum, o.a. gebaseerd op Lean (1990s)

Scrum is een van de meest populaire Agile-werkwijzen die o.a. Lean-principes toepast (simplicity - the art of…) door teams in staat te stellen hun werk te organiseren in iteraties (sprints) en waarde snel te leveren. Scrum van 1990 is niet hetzelfde als het huidige Scrum, veel van de elementen en accenten zijn door de jaren heen ontwikkeld en aangescherpt. SAFe heeft Scrum rond 2011 opgepakt, ietsje aangepast, en noemt dat sinds versie 6.0 “SAFe Scrum”. Scrum zit dus in SAFe, maar beetje gefinetuned naar het SAFe-gedachtegoed.

Mijn Scrum-boekentips en leestips:

• De formele Scrum Guide Versie 2020 (De online versie, hier te vinden: https://scrumguides.org/scrum-guide.html)

• Scrum voor Dummies, Michael Franken

• De kracht van Scrum, Rini van Solingen en Eelco Rustenburg

• https://scaledagileframework.com/safe-scrum/

Agile Manifesto, o.a. gebaseerd op Lean en Scrum principes (2001)

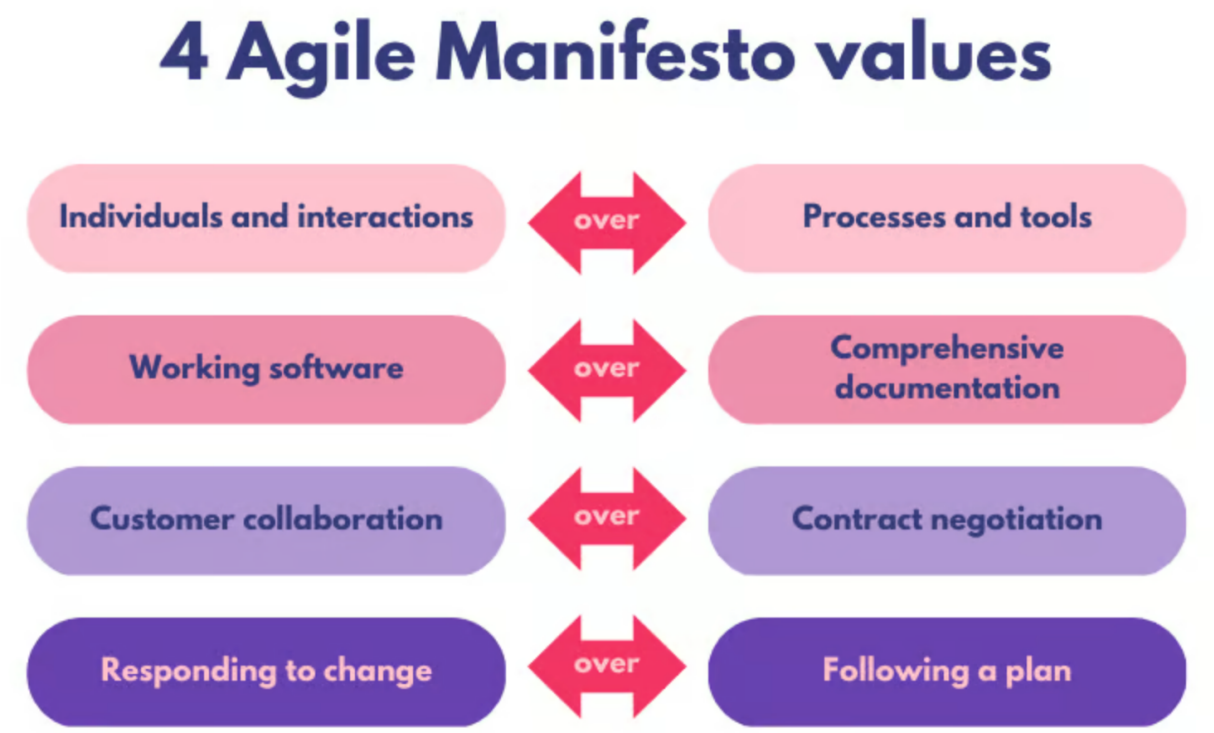
Het Agile Manifesto werd in 2001 opgesteld door 17 softwareontwikkelaars tijdens een bijeenkomst in Snowbird, Utah. Bekende namen zoals Jeff Sutherland, Ken Schwaber en Martin Fowler creëerden het manifest om een flexibele, collaboratieve aanpak, mindset en filosofie te bevorderen in softwareontwikkeling. Het manifesto bestaat uit vier kernwaarden en twaalf principes, gericht op klantgerichtheid, samenwerking en aanpassing. Agile is sterk beïnvloed door Lean-principes (zoals verspilling verminderen, agile principe #10) en een gedeelde filosofie met Scrum.

Mijn agile boek en lees tips:

• “Agile Managen” van Mike Hoogveld

• “Agile” van Rini van Solingen

• Hoofdstuk 13, ‘Het Agile Manifesto: de core van SAFe met een uitleg’



DevOps gebaseerd op Lean, Agile, en adaptatie (2009)

DevOps breidde Agile’s iteratieve aanpak uit naar samenwerking met operations, wat continu aanpassingsvermogen in de volledige productlevenscyclus mogelijk maakt.

Mijn devops boek tips

• “The DevOps Handbook”, How to Create World-Class Agility, Reliability, & Security in Technology Organizations van Gene Kim, Jez Humble, Patrick Debois, John Willis, Nicole Forsgren.

• “DevOps For Dummies” door Emily Freeman.

SAFe gebaseerd op Lean, Agile, DevOps en Systems Thinking

SAFe brengt inzichten en praktijken uit Lean, Agile, DevOps en Systems Thinking samen in één geïntegreerd raamwerk. De focus ligt op aanpassingsvermogen op *grote schaal*, met Built-in Quality en continue verbetering. SAFe bevat Scrum als onderste werklaag en heeft vervolgens daar scaling op toegepast.

Mijn boek en lees tips:

• Bekijk en lees SAFe online na op https://scaledagileframework.com/

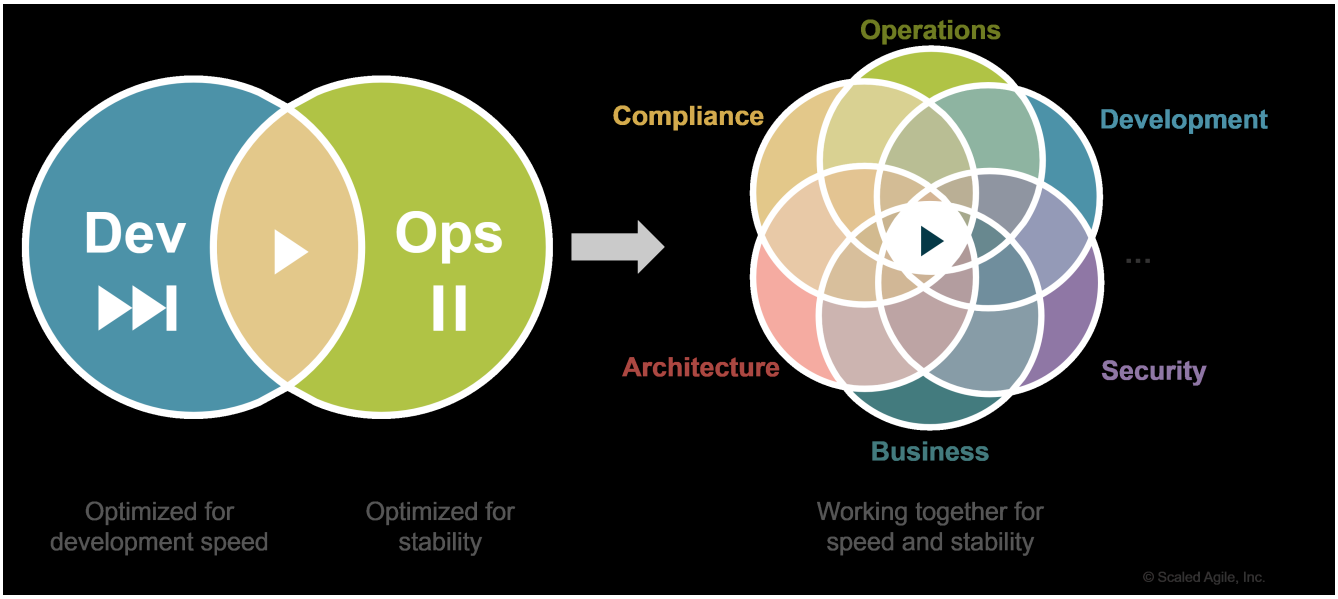
• Niet de laatste versie van SAFe, maar wel in boekvorm naleesbaar: "SAFe 5.0 Distilled: Applying the Scaled Agile Framework for Your Enterprise" door Richard Knaster en Dean Leffingwell

DevSecOps gebaseerd op DevOps en veiligheid (2012-2014)

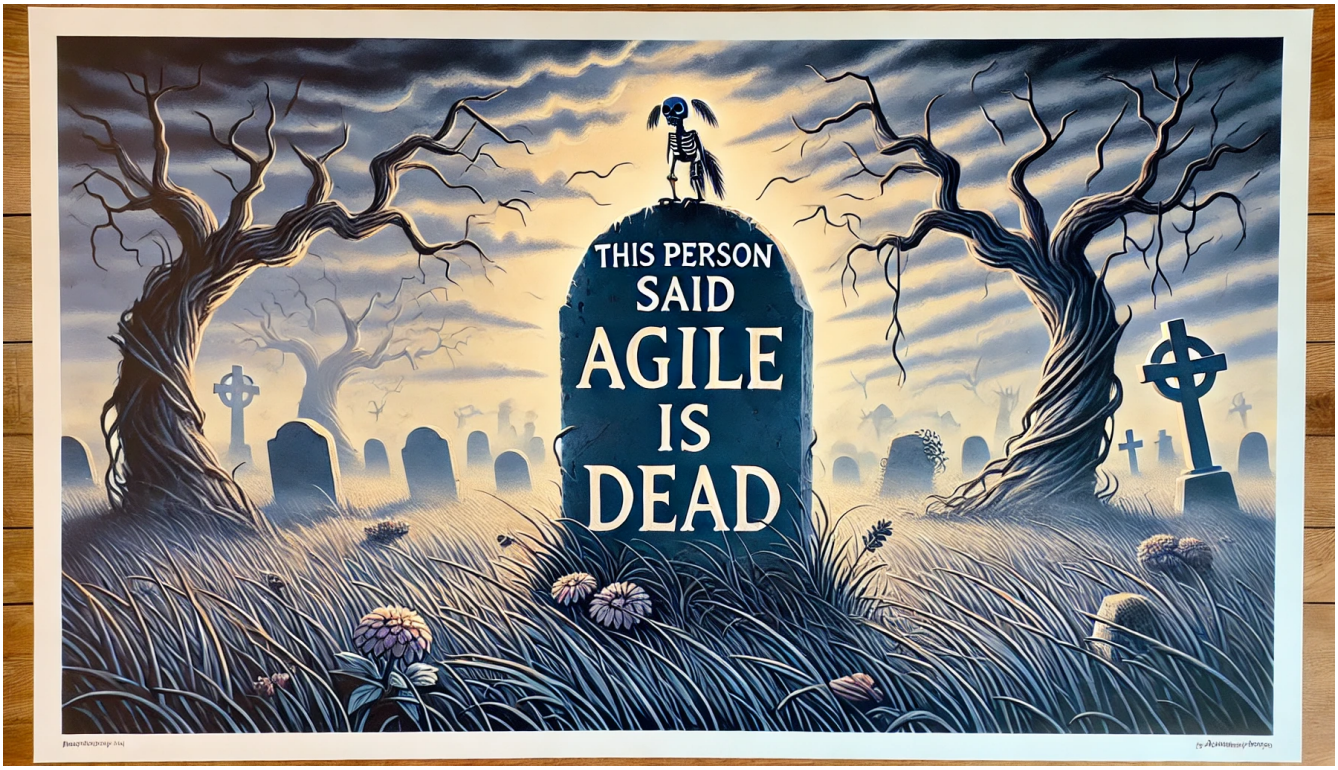
DevSecOps breidde DevOps uit door beveiliging centraal te stellen in elke fase van de ontwikkeling. In SAFe's Versie 5.0 werd DevSecOps opgenomen om in een veranderend landschap voortdurend veilige software te kunnen leveren.

Mijn DevSecOps boek tip:

• https://scaledagileframework.com/devops/



7. People who say "Agile is dead" are dead



Inleiding

Is Agile écht dood? Of zijn we simpelweg slecht in het *toepassen* ervan? Want, je hebt ze vast voorbij zien komen: LinkedIn-experts met grote claims en fancy plaatjes. Laten we naar die posts eens kritisch kijken.

Agile is allesbehalve dood, de revolutie is voorbij en e.e.a. raakt nu overal eigenlijk ingeburgerd. Nog sterker, door de toegenomen oorlogsdreiging is wendbaarheid belangrijker dan ooit. Niet voor niets wordt binnen Defensie op alle fronten aan een Agile Transformatie gewerkt en staat die transformatie op de to-dolijst van het topmanagement.

Dus weet je wat daarom wél dood is? De zogenaamde experts die dit roepen. Het lijkt soms alsof ze tegelijkertijd beweren dat gezond verstand ook niet meer bestaat.

Agile begon ooit als een mindset die gebaseerd is op kernwaarden en -principes, zoals zelforganiserende teams, klantgerichte samenwerking en aanpassingsvermogen. Maar door de jaren heen heeft de massale adoptie geleid tot o.a. misbruik en overmatige commercialisering, waardoor de oorspronkelijke essentie vaak verloren gaat. Ik weet bijna zeker dat geen enkel bedrijf weer op iets nieuws zit te wachten, na jaren van Waterfall en nu Agile. Wat is het alternatief dan en werkt dat dan wel?

In dit hoofdstuk wil ik dieper ingaan op deze critici en verkennen wat volgens mij de oorzaken zijn van deze aandachtsroep om de ‘dood van Agile’. Alsmede de oorzaken van falende SAFe-implementaties en valkuilen waar implementatie-consultants goed op moeten letten.

Waarom zeggen ze dat?

Waarom roepen die LinkedIn-gebruikers dat? Is er iets beters dan? Terug naar PRINCE2? Waterfall? Lees dan vooral het boek ‘de bodemloze put’ om te leren wat er dan in hele grote phase gate projecten gebeurt. Die lessen hebben we nu wel geleerd, toch?

Ik observeer een aantal niet-limitatieve mogelijke redenen:

1. **Zelfpromotie en nieuw heldendom:**
“Koop mijn diensten in plaats van die van Agile-consultants, want ik heb dé oplossing met Buzzword x, y en z.” Dan vaak gevolgd door: Reply with ‘*Oplossing*’ in de comments.

2. **Zijkantschreeuwers:**
Mensen die zelf nooit een SAFe-implementatie van enige omvang (>10 ART’s) hebben uitgevoerd, of er hooguit beperkte invloed op hadden.

3. **Slechte ervaringen:**
Deze critici hebben mogelijk te maken gehad met ‘blinde framework-implementators’ of ‘Agile theologen’. Ze hebben het zelf nooit goed zien werken, maar hebben er zelf ook geen 100% dominante invloed op Agile succes gehad. Waarom zeg je anders dat je eigen geesteskind dood is? Zegt Elon Musk ook dat elektrisch rijden dood is?

4. **Alles voor de views:**
Controversiële statements leveren vaak meer interactie en dus views op.

Agile is niet dood, mits je deze valkuilen vermijdt

Agile is niet dood. Wat wel dood is, zijn slechte Agile-implementaties geleid door slecht opgeleide, niet geroutineerde Agilisten. Waar gaat het dan fout? Dit zijn de veelvoorkomende valkuilen die ik zie, tegenkom en mezelf voor behoed:

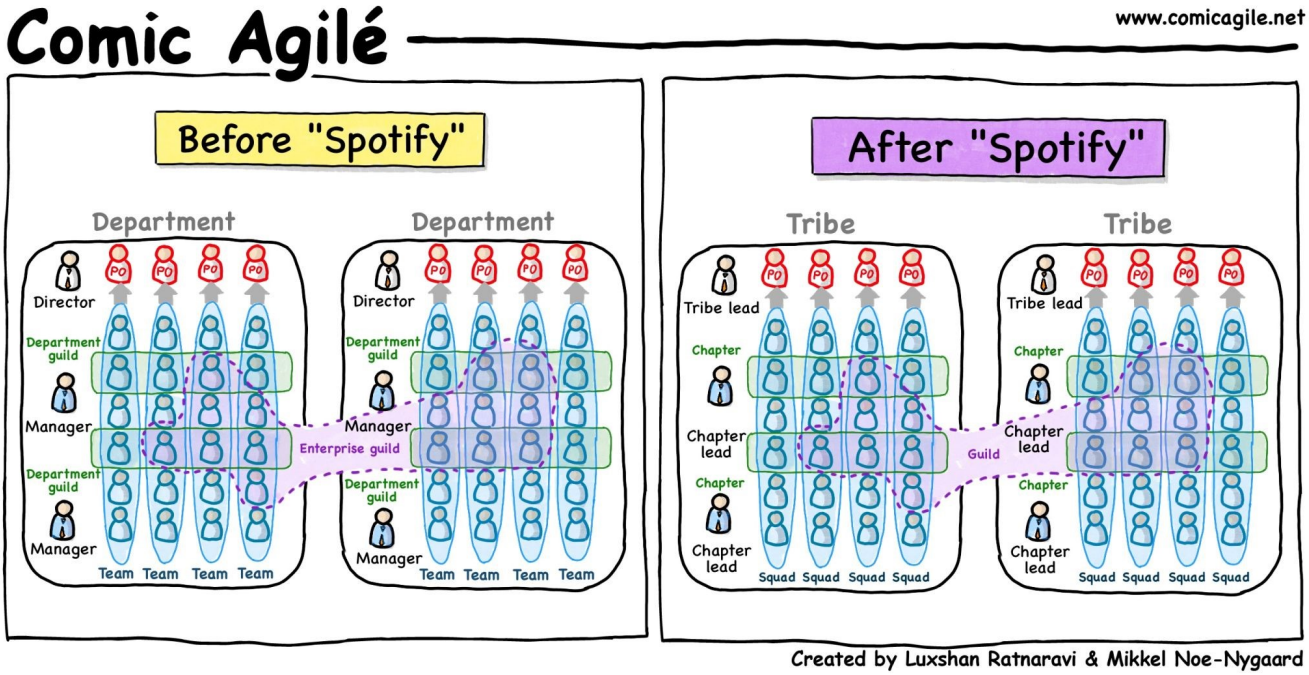
• **Geen duidelijk “Tipping Point”:**
Het gaat al mis bij stap 1: de sense of urgency bepalen. Wat is de pijn? Wat zijn de root causes? Welke feiten onderbouwen de fundamentele problemen? En hoe gaat SAFe deze oplossen? Zonder rigoureuze probleemidentificatie, datagedreven planning en een strategische aanpak is elke implementatie gedoemd te mislukken. Het is onduidelijk voor welk probleem SAFe de oplossing is.

• **Agile als werkwoord in plaats van mindset:**
Veel organisaties claimen “Agile te doen” maar begrijpen de achterliggende kernprincipes en waarden niet. Ze zien Agile als een set tools of ceremonies, in plaats van een mindset.

• **Het ‘SAFe-politiemodel’:**
SAFe wordt zo aangepast dat het zijn werking verliest en de hele essentie verloren gaat, waarna diezelfde mensen roepen: “Agile werkt niet.”

• **Mode- en buzzwords, vercommercialiseerd:**
Agile is zo wijdverbreid dat de term leeg aanvoelt. Veel organisaties profileren zich als “Agile” zonder echt de onderliggende principes toe te passen. Vooral (IT) leveranciers willen Agile nog wel eens commercieel inzetten, ruim op hun website vermelden en zich zo profileren. Zonder zelf echt Agile te zijn. Want Agile ‘verkoopt lekker’.

• **Rebranden & SINO (SAFe In Name Only):** Simpelweg “rebranden” van de functies en afdelingsnamen en verder alles bij het oude laten. Geen flauw benul van *Conway’s Law* of *VSAIW’s* (Value Stream and ART Identification Workshops). “Monkey see, Monkey do” implementaties. Het misplaatste geloof dat hernoemen of oppervlakkig herstructureren diepe uitdagingen kan oplossen, ondermijnt het transformatieve potentieel van frameworks als SAFe.



• **Slechte rituelen:**
SAFe-events worden oppervlakkig uitgevoerd zonder begrip van de achterliggende principes of eventdoelen. Het is niet de eerste keer dat ik een developer hoor roepen ‘dat PI events zonde van de tijd zijn!’. Dan heb je noch PI-events, noch SAFe goed begrepen.

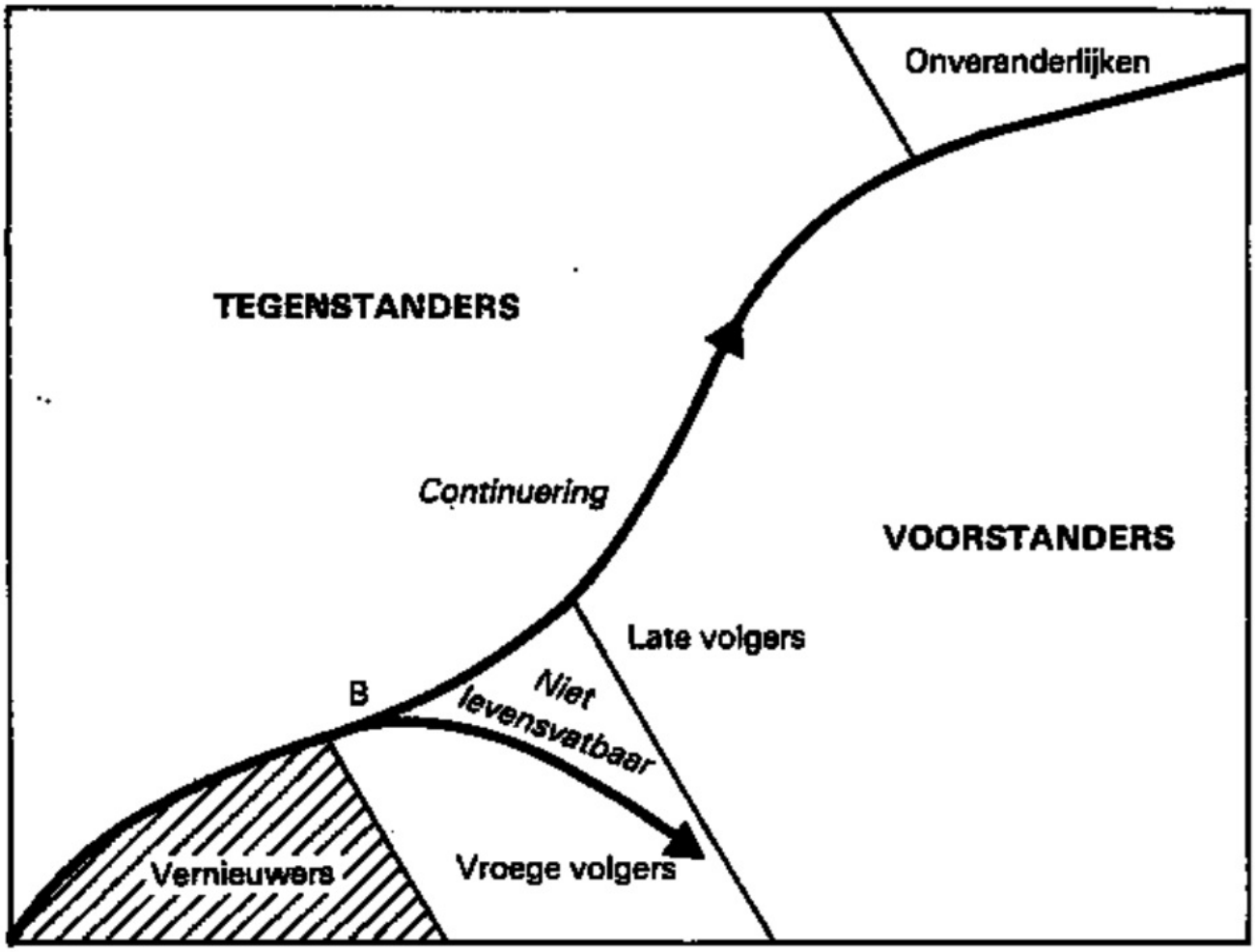
• **Geen shift van project- naar productmanagement:**
Veel organisaties slagen er niet in om de stap te maken van project- naar productmanagement.

• **Hard zonder zacht:**
Tools zoals Jira en processen, events worden geïmplementeerd zonder aandacht voor cultuurverandering, gedrag en mindset.

• **Maak ook de CI/CD-pipeline SAFe-compliant:** Wanneer SAFe wordt geïmplementeerd, moet ook de CI/CD-pipeline worden aangepast aan de SAFe-principes, woorden, processtappen etc. Als dit niet gebeurt, ontstaat er een kloof tussen wat in trainingen wordt geleerd en wat in de praktijk wordt ervaren. Medewerkers zullen moeite hebben om het trainingsmateriaal te herkennen in hun dagelijkse werk. Zorg daarom ervoor dat de tools, taal, processen en werkwijzen consistent zijn met de SAFe-manier van werken.

• **Trainen, trainen en trainen:**
Gebrek, geen geld, geen tijd voor training en certificering: Managers en werknemers moeten worden opgeleid in de formele SAFe-opleidingen. SAFe is gewoon te complex en soms te nieuw, denk onder andere aan alle nieuwe begrippen en constructs. Iedereen, dus op alle niveaus, moet worden getraind in de kernprincipes van Agile en het gebruik van SAFe-instrumenten en de agile tooling. Dit vergroot het begrip en de betrokkenheid en helpt mensen elkaar te verstaan. Zie ook Agile Coaches.

• **De onveranderlijken:**
Elke organisatie heeft een hele kleine groep onveranderlijken, maar soms wel met heel veel lawaai: werknemers die weerstand blijven bieden en transformaties bewust saboteren. Als organisaties deze personen daarmee weg laten komen, vermindert dit de kans op succesvolle implementatie.



• **De verandering minimaliseren:**
De RTE Projectmanager blijven noemen, PI-events 'MT-meetings' noemen, Epics projecten en vooral zoveel mogelijk alles bij het oude laten. SAFe-begrippen consequent aanpassen aan eigen woordjes, om 'de verandering' voor de werknemers, denken ze, zo klein als mogelijk te houden. Met gevolg dat na 1 jaar niemand weet wat er nu exact is veranderd.

• **Verkeerde prioriteiten in de backlogs**
Het gebruiken van een backlog als een simpele Todo-lijst zonder deze te koppelen aan strategische thema's, OKR's, milestones of klantwaarde, leidt tot een gebrek aan focus en inefficiëntie.

• **Wagile:**
Organisaties claimen Agile te zijn, maar blijven in werkelijkheid waterfalletjes uitvoeren, maar dan met sprints.

• **Hiërarchische besluitvorming handhaven**
Wanneer besluitvorming binnen een hiërarchische top-downstructuur blijft bestaan, kunnen teams niet de autonomie ontwikkelen die nodig is om echt Agile te werken en vooral het tempo van waardecreatie verhogen middels SAFe-principes zoals 'decentralize decision-making' (principe #9). Dit leidt tot frustratie en een gebrek aan eigenaarschap.

• **Onervaren, ongeschoolde consultants die SAFe gaan implementeren:**
Meerdere keren meegemaakt dat (junior) consultants/zzp-ers, *zonder enige SAFe-opleiding* of doorleefde kennis en ervaring 'SAFe gingen implementeren'. SAFe is gewoon te complex om zonder gedegen opleiding en training te implementeren. Dit soort implementatie-consultants missen routine, de grote verbanden, samenhang der dingen en essentiële inzichten om Agile op een doeltreffende manier te implementeren. Weten zij bijvoorbeeld dat SAFe Epics kent op verschillende niveaus: Enterprise, Portfolio, Solution Train en ART? Enterprise Epics ondersteunen initiatieven die meerdere portfolio's raken. En kennen zij überhaupt bijvoorbeeld het ART Epic Kanban board om je LPM kanban te offloaden? Ze zien het grote geheel niet, kennen de mogelijkheden niet,

weten niet exact hoe waarde door het hele SAFe Framework stroomt en weten hoogstwaarschijnlijk ook niet wat een LACE of VMO is. Ze implementeren SAFe ‘op gevoel’ en ‘van zien en horen zeggen’

• **Scheid hiërarchie van het value stream-netwerk:**
Veel organisaties claimen de SAFe-principes te omarmen, maar zodra het nieuwe Target Operating Model ook maar enigszins een verschuiving in het mandaat van of tussen lijnmanagers inhoudt, ontstaat er weerstand. Want de lijnmacht neemt af en verschuift naar 2de Operating System (OS). Het is daarom essentieel om een strikte scheiding aan te brengen tussen de hiërarchische structuur en het value stream-netwerk, met heldere nieuwe verantwoordelijkheden en taken.

• **Inconsistent leiderschap**
Leiders die Agile van hun teams verwachten, maar zelf blijven handelen volgens de principes zoals ze altijd al werkten, zorgen voor frustratie en ondermijnen de verandering.

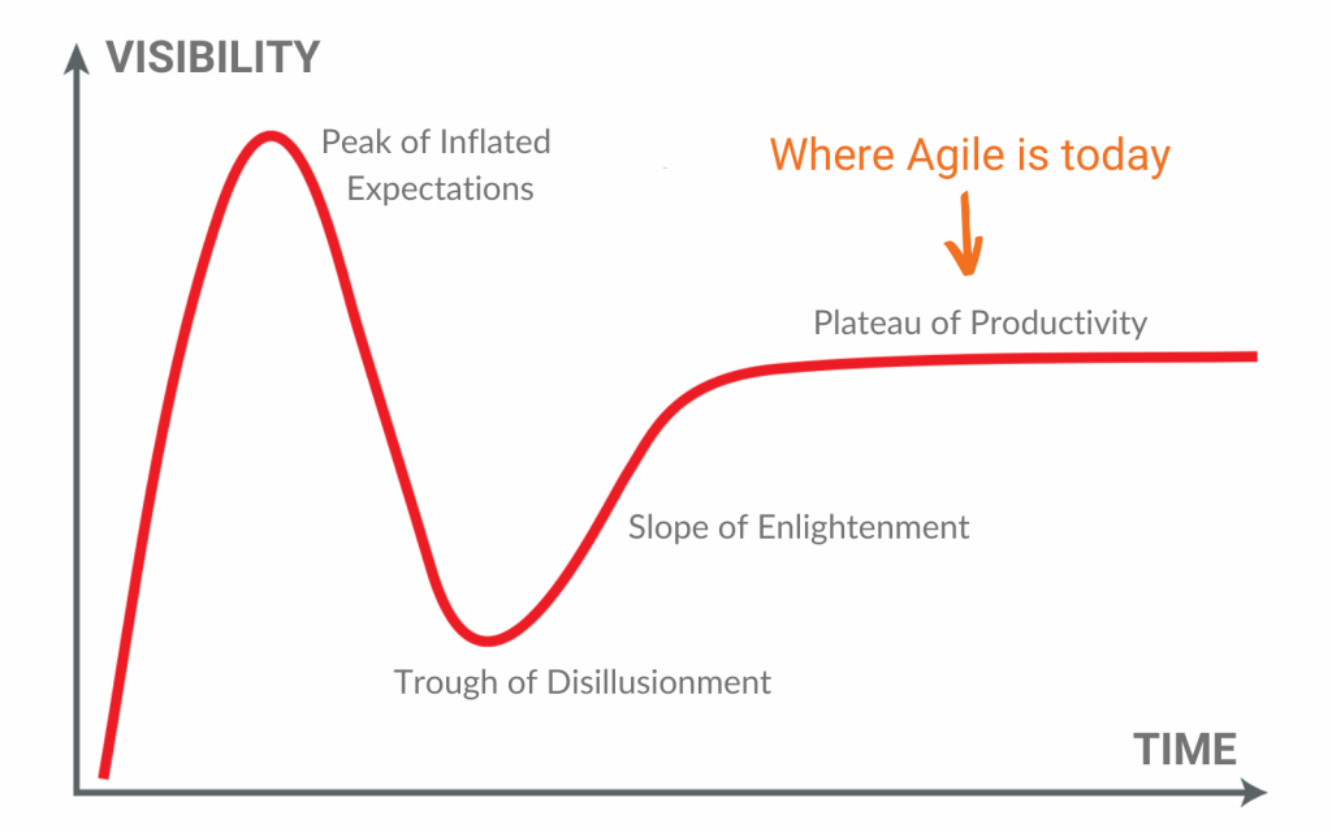
• **Focus op balans**: Hoe combineren we structuur met flexibiliteit? Hoe zorgen we voor kwaliteit terwijl we autonomie vergroten?

• **Dogmatisch Agile. Praten zonder executiekracht:**
Sommige consultants passen Agile dogmatisch toe en doen dingen omdat ‘ze die moeten doen’, zonder principe #10 van het Agile Manifesto (“Simplicity: the art of maximizing the amount of work not done”) te kennen of te volgen. Je moet niet over Agile of SAFe praten, maar over het oplossen van de pijn (zie bullet #1). En uitleggen dat herorganiseren in waardestromen en het elimineren van silo’s ons daarbij helpt.

• **Onbekwame of geen SAFe-coaches**
De uiteindelijke SAFe-implementatie, vooral ‘from scratch’, is in hoge mate afhankelijk van de (in te vliegen) Agile SAFe-coaches, hun vakbekwaamheid en hun effectieve bijdrage aan het lopend krijgen van SAFe. De regel zegt 1 agile coach per 5 tot 8 agile teams in opstartfase.

• **Onduidelijk ‘wanneer Agile dan niet dood is’: Vaststelling van de succes metrics**
Het ontbreken van meetbare doelen (OKR’s) maakt het lastig om de voortgang van een Agile-transformatie te evalueren en daarop bij te sturen indien nodig. Deze organisaties blijven hangen in een transformatie en zien nauwelijks succes…

Conclusie



Agile is niet dood, het leeft: mits we de moed hebben om het écht toe te passen en de genoemde valkuilen te mitigeren.

Agile evolueert en past zich aan. Succesvolle transformatie draait om het beheersen van de elementen die een organisatie vooruit stuwen: gedeeld doel, vertrouwen en aanpassingsvermogen. Leiders moeten met visie en wendbaarheid navigeren door complexe omstandigheden. Zie boven de “Gartner Hype Cycle”.

Ik zal Agile voortaan *Generative Agile* en SAFe voortaan *Generative SAFe* noemen om de term weer wat nieuw leven in te blazen en de critici op het verkeerde been te zetten. En het woord Generative bekt en verkoopt lekker (-:

8. SAFe ontleed: oorsprong en het ecosysteem achter 's werelds grootste Agile framework

SAFe ontleed: oorsprong en het ecosysteem achter 's werelds grootste Agile framework

Wie met SAFe aan de slag gaat, doet er goed aan te begrijpen wie er achter dit framework zit. SAFe, de afkorting van Scaled Agile Framework, werd ontwikkeld door Dean Leffingwell, een Amerikaanse softwareondernemer, auteur en denker op het gebied van grootschalige softwareontwikkeling. De wortels van het framework liggen in zijn boek uit 2007 'Scaling Software Agility: Best Practices for Large Enterprises', waarin hij de basisprincipes beschreef voor het schalen van agile werkwijzen in grote organisaties. Op basis daarvan lanceerde hij in 2011 de eerste officiële versie van SAFe, en sindsdien is het framework uitgegroeid tot de mondiale standaard voor bedrijfswendbaarheid.

Het framework is eigendom van Scaled Agile, Inc., een privaat bedrijf gevestigd in Boulder, Colorado, in de Verenigde Staten. Het bedrijf richt zich op de verdere ontwikkeling, het beheer en de verspreiding van SAFe, ondersteund door een uitgebreid programma van trainingen en certificeringen. In november 2021 verwierf het investeringsbedrijf Eurazeo, afkomstig uit Frankrijk, een meerderheidsbelang in Scaled Agile, Inc. met een investering van circa 300 - 350 miljoen dollar. Die stap bevestigde niet alleen de financiële slagkracht van het bedrijf, maar ook de brede erkenning van SAFe als strategisch waardevol raamwerk voor organisaties wereldwijd.

Gartner erkent SAFe als het meest gehanteerde en overwogen framework voor het schalen van agile, met een marktaandeel van ~30% (ref: **State of Agile Report**) en nog steeds groeiend.

(ref: https://pages.eml.atlassian.com/rs/594-ATC-127/images/Gartner-Critical-Capabilities-for-Enterprise-Agile-Planning-Tools_en_366957.pdf)

SAFe's evolutie

Het SAFe-framework heeft sinds zijn introductie een duidelijke evolutie doorgemaakt, zoals in de tabel onder te zien is. Sinds de eerste release in 2011 zijn er meerdere grote versies verschenen die het framework stap voor stap hebben uitgebreid. Versie 3.0 uit 2014 introduceerde het portfolioniveau, waardoor strategische afstemming voor het eerst expliciet onderdeel werd van het model. Met versie 4.0 in 2016 werd de structuur rond value streams verder uitgewerkt en kreeg het framework een duidelijkere architectuur voor grootschalige implementaties. In versie 4.5 uit 2017 werden DevOps en de Continuous Delivery Pipeline nadrukkelijk geïntegreerd en werd het framework gepresenteerd in vier herkenbare configuraties: Essential, Large Solution, Portfolio en Full SAFe. Met de introductie van versie 5.0 in 2020 verschoof de nadruk naar Business Agility, ondersteund door een set van totaal 7 kerncompetenties die organisaties helpen wendbaarder te worden. In maart 2023 verscheen SAFe 6.0, met een sterke nadruk op flow-optimalisatie, betere samenwerking in value streams en verdere ondersteuning van moderne software- en productontwikkeling. In 2025 volgde een herstructurering van de zeven kerncompetenties naar vijf disciplines (zie hoofdstuk 1, ‘Het DNA van SAFe’). Samen laten deze versies zien hoe het framework zich blijft ontwikkelen in lijn met de behoeften van organisaties die agile werkwijzen op schaal toepassen.

SAFe is meer dan alleen voor software

Veel mensen associëren SAFe nog steeds uitsluitend met softwareontwikkeling. Dat beeld komt deels voort uit het Agile Manifesto, waarin softwareontwikkeling centraal staat. In de praktijk is het framework echter al lang breder toepasbaar. SAFe is door de jaren heen uitgebreid met concepten en praktijken die geschikt zijn voor complexe systeemontwikkeling, waarin software, firmware, hardware en cybersecurity samenkomen. Vooral in sectoren zoals defensie, automotive, aerospace en industriële systemen werken organisaties immers niet alleen aan software, maar aan geïntegreerde systemen. Om die reden heeft Scaled Agile het framework verder ontwikkeld met aanvullende guidance en trainingen, waaronder het recente **SAFe for Hardware-programma**, waarin specifieke principes, cadans en ontwikkelpraktijken voor hardwareontwikkeling worden behandeld. Daarmee positioneert SAFe zich nadrukkelijk als een framework voor **end-to-end systeemontwikkeling**, en niet uitsluitend voor software.

Gecertificeerde professionals en actieve expertise

Een van de sterkste indicatoren van het succes van SAFe is het aantal mensen dat wereldwijd met het framework werkt. Meer dan een miljoen professionals zijn getraind via het SAFe-certificeringsprogramma, dat is opgebouwd rondom verschillende rollen en niveaus binnen een agile organisatie. De Leading SAFe-certificering, ook wel aangeduid als SA, is daarbinnen een van de meest gevraagde en meest afgenomen trajecten, al worden de exacte aantallen per certificeringsniveau niet openbaar gemaakt.

Achter die miljoenengroep staat een robuust netwerk van praktijkexperts. SAFe Practice Consultants, de zogenoemde SPC's, vormen de ruggengraat van dat netwerk. Zij zijn gecertificeerd om organisaties te begeleiden bij SAFe-transformaties en om andere professionals te trainen en te certificeren. Naast de SPC's zijn er Release Train Engineers (RTE's) en tal van andere gecertificeerde specialisten die dagelijks actief zijn in SAFe-omgevingen bij bedrijven over de hele wereld. Samen zorgen zij ervoor dat de kennis en het gebruik van het framework levend en actueel blijven, ook buiten de directe invloedssfeer van Scaled Agile, Inc. zelf. Het allerhoogste haalbare certificeringsniveau is dat van **SAFe Practice Consultant Trainer (SPCT)**. SPCT's vormen een kleine, wereldwijd geselecteerde groep experts die bevoegd zijn om **SAFe Practice Consultants (SPC's)** op te leiden en te certificeren. Zij spelen een belangrijke rol in de kwaliteitsborging en verdere ontwikkeling van het SAFe-framework. Let op, wereldwijd zijn 'er maar' circa 100 SPCT's, waarvan ongeveer 5 in Nederland. Dit betekent dat het SPCT certificeringsniveau tot een van de meest exclusieve certificeringen binnen het SAFe-ecosysteem behoort.

Het SAFe-ecosysteem

SAFe is allang meer dan een framework op papier: het is een levend ecosysteem van mensen, bedrijven en kennis. Wereldwijd zijn meer dan 500 gecertificeerde partner- en transformatiebedrijven actief als deel van het officiële SAFe-netwerk, aanwezig in meer dan 110 landen. Zij ondersteunen organisaties van elke omvang en sector bij de implementatie van het framework, van de eerste oriëntatie tot volwassen transformaties over meerdere Agile Release Trains. Meer dan 20.000 organisaties wereldwijd werken inmiddels met SAFe: van kleine, wendbare bedrijven tot grote multinationals in sectoren als financiën, gezondheidszorg, overheid en technologie.

De bedrijfswaarde van Scaled Agile, Inc.

De investering van Eurazeo in november 2021 gaf een eerste concrete indicatie van wat Scaled Agile, Inc. waard is. Met een investering, volgens analisten, van circa 300 - 350 miljoen dollar voor een meerderheidsbelang werd de totale bedrijfswaarde destijds geschat op 400 tot 500 miljoen dollar. Dat was echter nog maar een beginpunt. Sindsdien is de adoptie van SAFe verder gegroeid, en in 2023 nam Scaled Agile ook Rentouch over, een overname die het aanbod van het bedrijf verder verbreedde. Wanneer je die groei en marktdynamiek meeneemt, is een huidige bedrijfswaarde van 600 tot 800 miljoen dollar een reële schatting, hoewel het exacte getal afhankelijk blijft van bredere marktfactoren en toekomstige ontwikkelingen.

De huidige waarde van Scaled Agile, Inc. valt dan ook niet los te zien van de groeiende wereldwijde vraag naar Lean-Agile-transformaties. Naarmate meer organisaties op zoek gaan naar bewezen methodes om complexe veranderingen in goede banen te leiden, blijft SAFe aan relevantie winnen. Een bedrijfswaarde van 600 tot 800 miljoen dollar is in dat licht een plausibele en conservatieve schatting, een getal dat de strategische positie van Scaled Agile, Inc. weerspiegelt in een markt die nog lang niet zijn plafond heeft bereikt.

Historisch overzicht van SAFe-versies en releasedata

De onderstaande evolutie van SAFe laat zien dat het framework zich voortdurend aanpast aan nieuwe technologische en organisatorische realiteiten. Wat begon als een model voor het schalen van agile softwareontwikkeling is uitgegroeid tot een breed toepasbaar raamwerk voor complexe organisaties die wendbaarheid op grote schaal willen realiseren.

SAFe versie	Releasedatum	Opmerking
SAFe 1.0	2011	Eerste publieke versie van het framework
SAFe 2.0	2013	Eerste grotere structurering van het model
SAFe 3.0	2014	Introductie van Value Streams en Portfolio niveau
SAFe 3.5	2015	ART stabilisatie, architectuur, scaling patterns
SAFe 4.0	Jan 2016	Grote update met nieuwe configuraties
SAFe 4.5	Apr 2017	Introductie van Lean Portfolio Management en DevOps.
SAFe 5.0	Jan 2020	Focus op Business Agility en Customer Centricity
SAFe 5.1	Feb 2021	Kleine update met verbeteringen en verduidelijkingen
SAFe 6.0	Mar 2023	Flow Accelerators, AI-ondersteuning, uitbreidingen in Measure & Grow en een sterkere focus op Value Streams.
SAFe 6.0.1	Jul 2023	Kleine maintenance update
SAFe 6.0.2	Nov 2023	Kleine update / correcties
Herstructurering	2025	Zeven kerncompetenties geherstructureerd naar vijf disciplines (zie hoofdstuk 1, ‘Het DNA van SAFe’)

9. Het SAFe framework ontleed: wat het zelf bedacht en wat het van anderen leende

Dit hoofdstuk sluit aan bij de SAFe-microtrainingen die ik verzorg. Microtrainingen zijn korte quick start-trainingen met één doel: deelnemers leren wat SAFe is, zodat ze zelf antwoord kunnen geven op de vraag ‘Wat is SAFe?’. De nadruk ligt op bewustwording en basiskennis; vanwege de beperkte tijd blijft dat noodzakelijk op hoofdlijnen. Wie de details wil leren, kan terecht bij de training Leading SAFe. Dit hoofdstuk vult de microtraining aan: het geeft cursisten en andere geïnteresseerden in SAFe inzage in hoe SAFe is opgebouwd en welke marktbekende theorieën, denkwijzen en stromingen in de loop der jaren in het model zijn verwerkt.

SAFe heeft het wiel niet opnieuw uitgevonden. Het framework brengt tientallen bestaande, beproefde principes en best practices samen en vult die aan met eigen constructies voor samenwerking op grote schaal. Dit hoofdstuk probeert ongeveer 90% van de relevante concepten op hoofdlijnen te behandelen: een redactionele ambitie, geen gemeten SAFe-dekkingspercentage.

In juni 2026 introduceerde Scaled Agile bovendien AI-Native SAFe: een vernieuwd operating model waarin AI een kerncapaciteit van de organisatie is in plaats van een losse tool. De eigen AI-concepten van SAFe staan in paragraaf 3; de AI-bouwstenen uit de markt staan in paragraaf 4.6. Het hoofdstuk begint met de kern van SAFe en de vraag of je mag cherry picken. Paragraaf 3 laat daarna zien wat SAFe zelf heeft ontwikkeld: de eigen SAFe-constructies en SAFe-uitwerkingen. Paragraaf 4 laat zien wat SAFe uit de markt heeft overgenomen, verdeeld over zes thema's: Lean, Agile en teams; product en klantwaarde; cultuur, psychologie en leiderschap; portfolio, strategie en economie; architectuur, kwaliteit en operations; en AI en data. Per bouwsteen zie je de bron of grondlegger, een korte omschrijving en in de laatste kolom het SAFe-artikel waarin het begrip wordt genoemd of uitgewerkt. Paragraaf 5 sluit af met een slotsom.

1. Wat is de kern van SAFe?

De herkenbare kern is de integratie van Agile Release Trains, Planning Intervals, PI Planning, rollen als de RTE, System Demo, Inspect & Adapt en de verbinding tussen strategie en uitvoering. Plannen, leren en samenwerken bestonden eerder; de samenhangende SAFe-uitwerking daarvan is kenmerkend.

2. Mag je cherry picken?

Ja, maar bewust. Een framework is geen verplichte boodschappenlijst. Begin bij het probleem, kies samenhangende bouwstenen, begrijp hun afhankelijkheden en meet of de organisatie daardoor sneller en beter waarde levert. Een ART zonder gezamenlijke prioriteiten of integratie is niet ineens Agile. SAFe moedigt dat zelf ook aan: in het artikel Customizing SAFe beschrijft Scaled Agile wanneer en hoe je het framework aan je eigen context aanpast, met vier guardrails en een stappenplan in vier stappen.

3. Eigen SAFe-constructies en SAFe-uitwerkingen

Alle bouwstenen in deze paragraaf zijn type S: constructies die SAFe zelf heeft ontwikkeld en die je in deze vorm alleen in SAFe tegenkomt. Bij Confidence Vote, Value Management Office, Business Owners in SAFe, Product Management, Product Owner (PO), System Architect, Operational Value Streams, Development Value Streams, Epic, Feature, Benefit Hypothesis, ART Kanban, SAFe Flow Metrics en Eight Flow Accelerators gaat het om een SAFe-uitwerking van een idee dat ook buiten SAFe bestaat.

Bouwsteen	Wat is het?	Meer lezen in SAFe
Agile Release Train (ART)	Langdurig ‘team of agile teams’ van 50 tot 125 mensen dat samen waarde levert	Agile Release Train
Release Train Engineer (RTE)	Coach en facilitator van de ART	Release Train Engineer
Planning Interval (PI)	Gezamenlijke periode van doorgaans 8 tot 12 weken	Planning Interval

Bouwsteen	Wat is het?	Meer lezen in SAFe
PI Planning	Gezamenlijke planning van teams, afhankelijkheden en doelen	PI Planning
PI Objectives	Business- en technische doelen voor het Planning Interval	PI Objectives
ART Planning Board	Features, mijlpalen en afhankelijkheden zichtbaar maken	PI Planning
Confidence Vote	Vertrouwen in het gezamenlijke PI-plan expliciet toetsen	PI Planning
Innovation & Planning Iteration	Tijd voor innovatie, leren en nieuwe planning. Laatste iteratie	Innovation and Planning Iteration
Inspect & Adapt	Gezamenlijk ART-resultaten analyseren en verbeteren	Inspect and Adapt
System Demo	Geïntegreerd resultaat van de ART demonstreren	System Demo
ART Sync	Afstemming tussen belangrijke ART- en teamrollen over voortgang	ART Sync
Solution Train	Meerdere ART's voor één complexe oplossing verbinden	Solution Train
Solution Train Engineer	Samenwerking over meerdere ART's faciliteren	Solution Train Engineer
Lean-Agile Center of Excellence	De transformatie coachen en ondersteunen	Implementation Roadmap
Value Management Office	Praktische Lean Portfolio Management-ondersteuning	Value Management Office
SAFe Implementation Roadmap	Gestructureerd transformatie- en implementatiepad	Implementation Roadmap
SAFe Core Values	Vier kernwaarden voor gedrag en beslissingen	SAFe Core Values
Ten Lean-Agile Principles	Tien richtinggevende economische en organisatorische principes	SAFe Lean-Agile Principles
SAFe Configurations	Essential, Large Solution, Portfolio en Full SAFe	SAFe Configurations
Measure and Grow	Business Agility meten en gericht verbeteren	Measure and Grow
CALMR (~DevOps)	Culture, Automation, Lean Flow, Measurement, Recovery	CALMR
Business Owners in SAFe	Stakeholders met verantwoordelijkheid voor businessresultaten	Business Owners
Portfolio Sync	Strategie en uitvoering op portfolioniveau afstemmen	Lean Portfolio Management
Lean Budget Guardrails	Financiële autonomie met governance combineren	Guardrails
Solution Intent	Eisen, ontwerpen en verificatie van een oplossing samenbrengen	Solution Intent

Bouwsteen	Wat is het?	Meer lezen in SAFe
Architectural Runway	Technische basis voor toekomstige functionaliteit	Architectural Runway
Product Management	Verantwoordelijk voor wat de ART bouwt: visie, roadmap en ART Backlog	Product Management
Product Owner (PO)	Vertegenwoordigt klant en business in het team en beheert de Team Backlog	Product Owner
System Architect	Bewaakt de technische visie en architectuur van de ART	System Architect
Epic Owner	Begeleidt een epic door de Portfolio Kanban, van idee tot uitvoering	Epic Owner
System Team	Ondersteunt de ART met integratie, testomgevingen en de pipeline	System Team
Shared Services	Specialisten die meerdere ART's ondersteunen zonder er vast deel van uit te maken	Shared Services
Operational Value Streams	De stappen waarmee de organisatie waarde levert aan de klant	Operational Value Streams
Development Value Streams	De stappen waarmee de organisatie de oplossingen bouwt die waarde mogelijk maken	Development Value Streams
Value Stream and ART Identification Workshop (VSAIW)	Waardestromen in kaart brengen om ART's rond waarde te organiseren	Development Value Streams
SAFe Requirements Model	Samenhangende hiërarchie van Epic, Capability, Feature en Story	SAFe Requirements Model
Epic	Groot initiatief dat een Lean Business Case en portfoliobesluit vraagt	Epic
Capability	Grote oplossingsfunctionaliteit die meerdere ART's overspant	Features and Capabilities
Feature	Dienst of functionaliteit die één ART binnen een PI levert	Features and Capabilities
Enabler	Werk aan architectuur, infrastructuur, exploratie of compliance dat toekomstige waarde mogelijk maakt	Enablers
Epic Hypothesis Statement	Gestructureerd format om de aannames en verwachte uitkomsten van een epic vast te leggen	Glossary: Epic Hypothesis Statement
Lean Business Case	Beknopte onderbouwing van een epic voor het go/no-go-besluit	Epic
Benefit Hypothesis	Verwacht meetbaar resultaat van een feature of epic, te toetsen na levering	Features and Capabilities
ART Backlog	Geprioriteerde lijst van features en enablers voor de ART	ART and Solution Train Backlogs
ART Kanban	Visualiseert en begrenst de flow van features door de ART	Applying Kanban in SAFe

Bouwsteen	Wat is het?	Meer lezen in SAFe
Capacity Allocation	Capaciteit vooraf verdelen over nieuwe functionaliteit, enablers en onderhoud	ART and Solution Train Backlogs
SAFe Flow Metrics	Zes metrieken: Flow Distribution, Velocity, Time, Load, Efficiency en Predictability	Measure and Grow
ART Predictability Measure	Vergelijkt behaalde met geplande business value van PI Objectives	Inspect and Adapt
Eight Flow Accelerators	Acht hefbomen om flow te versnellen, zoals WIP begrenzen en batches verkleinen	Principle #6: Make Value Flow without Interruptions
Release on Demand	Waarde vrijgeven aan klanten wanneer de business dat wil, los van deployment	Release on Demand
Continuous Exploration	Markt- en klantbehoeften verkennen en vertalen naar features	Continuous Exploration
PI Roadmap	Planning van verwachte resultaten voor het lopende en de komende PI's	Roadmap
Pre-Plan	Voorbereiding van PI Planning over meerdere ART's in een Solution Train	Pre-Plan
Solution Demo	Geïntegreerd resultaat van een Solution Train demonstreren	Solution Demo
PO Sync	Afstemming tussen PO's en Product Management over voortgang en scope	Coordinate and Deliver
Coach Sync	Afstemming tussen RTE en Scrum Masters/Team Coaches over impediments	Coordinate and Deliver
Spanning Palette	Elementen die in elke SAFe-configuratie kunnen worden ingezet	SAFe Configurations
AI-Native SAFe	Operating model (juni 2026) waarin AI een kerncapaciteit van de organisatie is in plaats van een losse tool	AI-Native SAFe
AI-Empowered Agility	Richtlijnen in de SAFe Big Picture om AI in bestaande SAFe-implementaties op te nemen	AI-Empowered Agility
Four Critical Shifts	Vier verschuivingen: Focus on Outcomes & Intent, Iterative Learning & Rapid Experimentation, Development & Innovation at Scale en Cross-Functional, AI-Augmented Teams	AI-Empowered Agility

4. Bouwstenen uit de markt

SAFe bouwt voort op tientallen theorieën, methoden en praktijken die elders zijn ontstaan. Deze paragraaf groepeert die bouwstenen in zes thema's: Lean, Agile en teams; product en klantwaarde; cultuur, psychologie en leiderschap; portfolio, strategie en economie; architectuur, kwaliteit en operations; en AI en data. Per paragraaf staat aangegeven welke bouwstenen SAFe overneemt (type B), welke SAFe een eigen uitwerking geeft (type S) en welke je aanvullend naast SAFe kunt toepassen (type A).

4.1. Lean, Agile en teams

De bouwstenen in deze paragraaf zijn type B: bestaande praktijken waarop SAFe voortbouwt. Uitzonderingen: type S (SAFe-uitwerking) is Continuous Delivery Pipeline; type A (aanvullend, naast SAFe toe te passen) is Scrumban.

Bouwsteen	Bron / grondlegger	Wat is het?	Meer lezen in SAFe
Lean Thinking	Taiichi Ohno; Womack & Jones	Waarde leveren met zo min mogelijk verspilling	Lean-Agile Mindset
Toyota Production System	Taiichi Ohno; Shigeo Shingo	Flow, kwaliteit en continu verbeteren als productiesysteem	Lean-Agile Mindset
Agile Manifesto	17 auteurs, 2001	Vier waarden en twaalf principes voor wendbaar werken	Lean-Agile Mindset
Scrum	Ken Schwaber; Jeff Sutherland	Productontwikkeling in korte Sprints met expliciete verantwoordelijkheden	SAFe Scrum
Kanban	Toyota; David J. Anderson	Werk visualiseren en WIP begrenzen voor betere flow	SAFe Team Kanban
Extreme Programming	Kent Beck	Technische kwaliteit met TDD, refactoring en pair programming	Built-in Quality
DevOps	Patrick Debois; Gene Kim e.a.	Ontwikkeling en operations integreren	DevOps
DevSecOps	DevOps-securitygemeenschap	Security continu in ontwikkeling en beheer meenemen	DevSecOps and the Software Factory
Scrumban	Corey Ladas	Scrum en Kanban in een hybride teamaanpak	SAFe Team Kanban
Kaizen	Toyota; Masaaki Imai	Voortdurend kleine verbeteringen doorvoeren	Inspect and Adapt
PDCA/PDSA	Shewhart; Deming	Plannen, uitvoeren, bestuderen en verbeteren	Iterations (PDCA)
Little's Law	John D. C. Little	Wiskundige relatie tussen WIP, doorvoer en doorlooptijd	Principle #6: Make Value Flow without Interruptions
Theory of Constraints	Eliyahu Goldratt	De belangrijkste beperking van een systeem verbeteren	Principle #6: Make Value Flow without Interruptions
Test-Driven Development	Kent Beck	Tests schrijven vóór implementatie	Test-Driven Development
Behavior-Driven Development	Dan North	Gewenst gedrag specificeren in begrijpelijke voorbeelden	Behavior-Driven Development
Pair Programming	XP-gemeenschap	Twee ontwikkelaars werken gelijktijdig aan dezelfde code	Developing Quality Software
Continuous Integration	XP-gemeenschap; Martin Fowler e.a.	Wijzigingen vaak integreren en automatisch testen	Continuous Integration
Continuous Delivery	Jez Humble; David Farley	Software steeds betrouwbaar uitrolbaar houden	Continuous Delivery Pipeline

Bouwsteen	Bron / grondlegger	Wat is het?	Meer lezen in SAFe
Continuous Delivery Pipeline	Jez Humble; David Farley; SAFe-uitwerking	Exploration, Integration, Deployment en Release verbinden	Continuous Delivery Pipeline
Trunk-Based Development	Software-engineeringgemeenschap	Frequent integreren in één centrale hoofdbranch	Continuous Integration
Built-in Quality	Lean; Agile Engineering	Kwaliteit vanaf het begin in het werk inbouwen	Built-in Quality
Value Stream Mapping (VSM)	Lean; Toyota	Stappen naar klantwaarde en verspilling zichtbaar maken	Value Stream Management
WIP Limits	Kanban	Gelijktijdig werk bewust begrenzen	SAFe Team Kanban
Batch Size Reduction	Don Reinertsen	Werk in kleinere batches opknippen voor snellere flow en feedback	Principle #6: Make Value Flow without Interruptions
Develop on Cadence	Don Reinertsen	Vaste ritmes en synchronisatie om onzekerheid beheersbaar te maken	Principle #7: Apply Cadence, Synchronize with Cross-Domain Planning
Gemba	Toyota; Lean	Gaan kijken waar het werk werkelijk gebeurt	Glossary: Gemba
Spike	Extreme Programming	Tijdgeboxt onderzoek om onzekerheid of risico te verkleinen	Enablers
Relative Estimation / Story Points	Agile-gemeenschap; XP	Werk ten opzichte van elkaar schatten in plaats van in uren	Story
Planning Poker	James Grenning; Mike Cohn	Teamschatting via gelijktijdig getoonde kaarten	Story
Feature Toggles / Dark Launch	Continuous Delivery-praktijk	Deployen loskoppelen van releasen door functionaliteit verborgen uit te rollen	Continuous Deployment

4.2. Product en klantwaarde

De bouwstenen in deze paragraaf zijn type B: bestaande praktijken waarop SAFe voortbouwt. Uitzonderingen: type S (SAFe-uitwerking) is Customer Centricity; type A (aanvullend, naast SAFe toe te passen) zijn Business Model Canvas, Value Proposition Canvas, Jobs to Be Done, A/B Testing en Design for Manufacturability.

Bouwsteen	Bron / grondlegger	Wat is het?	Meer lezen in SAFe
Design Thinking	Ontwerpdiscipline; IDEO	Klantproblemen onderzoeken en ideeën prototypen en testen	Design Thinking
Customer Journey Mapping	Service Design; meerdere voorlopers	Contactmomenten en klantbeleving over de hele klantreis visualiseren	Glossary: Customer Journey Map
Empathy Mapping	Dave Gray; XPLANE	Wat klanten denken, voelen, zeggen en doen in kaart brengen	Glossary: Empathy Map
Lean UX	Jeff Gothelf; Josh Seiden	Snel ontwerpen en aannames bij gebruikers toetsen	Lean UX

Bouwsteen	Bron / grondlegger	Wat is het?	Meer lezen in SAFe
User Experience Design	Don Norman e.a.	Producten begrijpelijk, bruikbaar en prettig maken	Lean UX
Human-Centered Design	Ontwerpdiscipline; ISO 9241-210	Ontwerpen vanuit echte gebruikersbehoeften	Using Human-Centered Design with SAFe
Lean Startup	Eric Ries	Ideeën testen via bouwen, meten en leren	Epic
Minimum Viable Product	Frank Robinson; Eric Ries	Kleinste product om een cruciale aanname te toetsen	Epic
Minimum Marketable Feature	Mark Denne; Jane Cleland-Huang	Kleinste stuk functionaliteit dat zelfstandig waarde levert voor de klant	Lean UX
Business Model Canvas	Alexander Osterwalder; Yves Pigneur	Negen onderdelen van een bedrijfsmodel zichtbaar maken	Portfolio Vision (Portfolio Canvas)
Value Proposition Canvas	Osterwalder e.a.	Productaanbod verbinden met klantenbehoeften	Niet expliciet in SAFe
Jobs to Be Done	Clayton Christensen; Tony Ulwick e.a.	Onderzoeken welke taak een klant wil vervullen	Customer Centricity
User Story Mapping	Jeff Patton	Stories langs de gebruikersreis structureren	Glossary: Story Map
User Stories	Kent Beck; Ron Jeffries e.a.	Kleine behoeften vanuit gebruikersperspectief beschrijven	Story
Hypothesis-Driven Development	Lean Startup	Veronderstellingen met experimenten valideren	Epic
Set-Based Design	Toyota Product Development	Meerdere oplossingen parallel openhouden en vergelijken	Set-Based Design
Product Discovery	Productmanagement; Marty Cagan e.a.	Onderzoeken welke problemen en oplossingen waardevol zijn	Continuous Exploration
A/B Testing	Statistiek	Alternatieve oplossingen gecontroleerd vergelijken	Continuous Deployment
Design for Manufacturability	Industriële engineering	Producten ontwerpen voor efficiënte fabricage	Applying SAFe to Hardware Development
Customer Centricity	Klantgerichtheid; SAFe-uitwerking	Mindset en werkwijze die beslissingen vanuit de klant laat vertrekken	Customer Centricity
Personas	Alan Cooper	Archetypische gebruikers om ontwerpkeuzes op te richten	Design Thinking
Leading Indicators	Lean Startup; innovation accounting	Vroege signalen of een investering het verwachte resultaat gaat opleveren	Validating Investment Opportunities

4.3. Cultuur, psychologie en leiderschap

De bouwstenen in deze paragraaf zijn type B: bestaande praktijken waarop SAFe voortbouwt. Uitzonderingen: type A (aanvullend, naast SAFe toe te passen) zijn Transformational Leadership, Team Development en Emotional Intelligence.

Bouwsteen	Bron / grondlegger	Wat is het?	Meer lezen in SAFe
Servant Leadership	Robert Greenleaf	Leiders dienen teams en verwijderen obstakels	Scrum Master/Team Coach
Psychological Safety	Amy Edmondson	Fouten en vragen zonder angst kunnen bespreken	Lean-Agile Leadership
Generative Culture	Ron Westrum	Vrije informatiestroom en gezamenlijke probleemoplossing	Lean-Agile Leadership
Growth Mindset	Carol Dweck	Vaardigheden zijn te ontwikkelen door leren	Lean-Agile Mindset
Self-Determination Theory	Edward Deci; Richard Ryan	Autonomie, competentie en verbondenheid voeden motivatie	Principle #8: Unlock Intrinsic Motivation
Motivation 3.0	Daniel Pink	Autonomie, meesterschap en zingeving stimuleren	Principle #8: Unlock Intrinsic Motivation
Transformational Leadership	Burns; Bass	Inspireren door visie en ontwikkeling	Niet expliciet in SAFe
Systems Thinking	Peter Senge e.a.	Samenhang en systeemgevolgen begrijpen	Principle #2: Apply Systems Thinking
Learning Organization	Peter Senge e.a.	Systematisch leren en kennis delen	Continuous Learning Culture
Change Management	John Kotter e.a.	Verandering mobiliseren en in gedrag verankeren	Leading Change
Dual Operating System	John Kotter	Hiërarchie combineren met flexibel verandernetwerk	Balancing the Dual Operating System
Communities of Practice	Jean Lave; Etienne Wenger	Vakgenoten leren over teamgrenzen heen	Communities of Practice
Organizational Culture	Edgar Schein	Gedeelde aannames sturen gedrag	Leadership and Culture
Team Development	Bruce Tuckman	Fasen van teamvorming herkennen	Working Successfully with Remote Team Members
Emotional Intelligence	Salovey; Mayer; Goleman	Emoties begrijpen in samenwerking en leidinggeven	Niet expliciet in SAFe

4.4. Portfolio, strategie en economie

De bouwstenen in deze paragraaf zijn type B: bestaande praktijken waarop SAFe voortbouwt. Uitzonderingen: type S (SAFe-uitwerking) zijn Lean Portfolio Management, WSJF in SAFe, Strategic Investment Planning, Portfolio Kanban, Strategic Themes, Value Stream Funding en Lean Governance; type A (aanvullend, naast SAFe toe te passen) zijn SWOT en TOWS-analyse.

Bouwsteen	Bron / grondlegger	Wat is het?	Meer lezen in SAFe
Portfolio Management	Managementdiscipline	Strategische initiatieven en investeringen besturen	Portfolio
Lean Portfolio Management	Lean; SAFe-uitwerking	Strategie, budgetten en uitvoering via waardestromen verbinden	Lean Portfolio Management

Bouwsteen	Bron / grondlegger	Wat is het?	Meer lezen in SAFe
OKR	Andy Grove; John Doerr	Doelen en meetbare sleutelresultaten koppelen	OKRs
Management by Objectives	Peter Drucker	Sturen op gezamenlijk bepaalde doelen	Niet expliciet in SAFe
Cost of Delay	Economische productontwikkeling; Don Reinertsen	Economische schade van uitstel zichtbaar maken	WSJF
WSJF in SAFe	Don Reinertsen; SAFe-uitwerking	Cost of Delay delen door job size voor prioritering	WSJF
Lean Accounting	Lean Accounting-gemeenschap	Financiële besturing meer waardestroomgericht inrichten	Lean Budgets
Beyond Budgeting	Jeremy Hope; Robin Fraser	Flexibel sturen buiten starre jaarbudgetten	Whitepaper Lean-Agile Financial Planning
Agile Contracts	Agile Contracting-gemeenschap	Contracten laten meebewegen met voortschrijdend inzicht	Agile Contracts
Investment Horizons	Innovatie- en portfoliomanagement	Investeringen verdelen naar tijdshorizon	Glossary: Investment Horizons
SWOT	Strategisch management	Sterktes, zwaktes, kansen en bedreigingen analyseren	Portfolio Vision
Business Case	Bedrijfskunde	Baten, kosten en risico's van investeringen onderbouwen	Epic (Lean Business Case)
Risk Management	Risicomanagement; ISO 31000	Risico's identificeren, beoordelen en behandelen	Agile Risk Management
Real Options	Financiële economie	Keuzevrijheid bij onzekerheid waarderen	Improving Portfolio Outcomes with Real Options
Strategic Investment Planning (voorheen Participatory Budgeting)	Publieke budgettering; SAFe-uitwerking	Betrokkenen laten meebeslissen over besteding	Strategic Investment Planning
Portfolio Kanban	Kanban; SAFe-uitwerking	Portfolio-epics via zichtbare stadia beheren	Portfolio Flow
Strategic Themes	Strategisch management; SAFe-uitwerking	Strategie richtinggevend maken voor de uitvoering	Strategic Themes
Value Stream Funding	Lean Budgeting; SAFe-uitwerking	Waardestromen in plaats van losse projecten financieren	Transitioning to Value Stream Funding
Flow Framework	Mik Kersten	Waardestromen meten met flow-metrieken; bron van de SAFe Flow Metrics	Value Stream Management
Lean Governance	Governance; SAFe-uitwerking	Uitgaven, compliance en voortgang van het portfolio lichtgewicht besturen	Lean Governance
CapEx en OpEx	Financiële verslaglegging	Agile werk correct verdelen over investerings- en exploitatiekosten	CapEx and OpEx
TOWS-analyse	Heinz Weihrich	SWOT-uitkomsten vertalen naar strategische opties	Glossary: TOWS Analysis

4.5. Architectuur, kwaliteit en operations

De bouwstenen in deze paragraaf zijn type B: bestaande praktijken waarop SAFe voortbouwt. Uitzonderingen: type S (SAFe-uitwerking) zijn Intentional Architecture en Lean Quality Management System; type A (aanvullend, naast SAFe toe te passen) zijn Evolutionary Architecture, Site Reliability Engineering en Definition of Ready.

Bouwsteen	Bron / grondlegger	Wat is het?	Meer lezen in SAFe
Systems Engineering	Engineeringdiscipline; INCOSE	Complexe systemen integraal ontwerpen en verifiëren	Lean Systems Engineering
Model-Based Systems Engineering	Systems Engineering-gemeenschap	Eisen en ontwerpen in samenhangende modellen beheren	Model-Based Systems Engineering
Emergent Architecture	Agile Architecture	Architectuur op basis van nieuwe inzichten laten groeien	Agile Architecture
Intentional Architecture	Architectuurdiscipline; SAFe-uitwerking	Bewuste architectuurbesluiten voor samenhang	Agile Architecture
Evolutionary Architecture	Neal Ford; Rebecca Parsons; Patrick Kua	Architectuur gecontroleerd laten evolueren	Agile Architecture
Nonfunctional Requirements	Requirements Engineering	Eisen aan prestaties, veiligheid en betrouwbaarheid	Nonfunctional Requirements
Lean Quality Management System (Lean QMS)	ISO 9001; Lean; SAFe-uitwerking	Kwaliteit en compliance borgen zonder de flow te blokkeren	Compliance
Root Cause Analysis	Kwaliteitsmanagement	Onderliggende probleemoorzaken vinden	Inspect and Adapt (Problem-Solving Workshop)
Ishikawa Diagram	Kaoru Ishikawa	Visgraatdiagram voor oorzakenanalyse	Inspect and Adapt (Problem-Solving Workshop)
Five Whys	Toyota-traditie	Met herhaald waarom doorvragen naar oorzaken	Inspect and Adapt (Problem-Solving Workshop)
Site Reliability Engineering	Google	Software engineering inzetten voor betrouwbaarheid	DevOps Practice Domains
Continuous Monitoring	DevOps; Operations	Voortdurend systemen meten en bewaken	DevOps Practice Domains
Fix Forward	DevOps; Continuous Deployment-praktijk	Productiefouten herstellen met een snelle nieuwe release in plaats van terugdraaien	Agile Product Delivery
Infrastructure as Code	DevOps; Cloud	Infrastructuur reproduceerbaar beheren via code	DevOps Practice Domains
Security by Design	Security Engineering	Beveiliging al in het ontwerp verwerken	DevOps Practice Domains
Technical Debt	Ward Cunningham	Toekomstige nadelen van technische compromissen zichtbaar maken	Refactoring
Definition of Done	Scrum; Agile	Gezamenlijke kwaliteitsgrens voor voltooid werk	Built-in Quality
Definition of Ready	Scrum-praktijk (niet in de Scrum Guide)	Gezamenlijke afspraken wanneer een story of feature klaar is om aan te beginnen	Applying SAFe to Hardware Development

4.6. AI en data

De bouwstenen in deze paragraaf zijn type B: bestaande praktijken waarop AI-Native SAFe voortbouwt. Uitzonderingen: type A (aanvullend, naast SAFe toe te passen) zijn Prompt Engineering, MLOps en Data Governance.

Bouwsteen	Bron / grondlegger	Wat is het?	Meer lezen in SAFe
Generative AI / Large Language Models	AI-onderzoek; Vaswani e.a. (Transformer, 2017)	Modellen die tekst, code en analyses genereren op basis van grote hoeveelheden data	AI-Native SAFe
AI Agents	AI-onderzoek; Russell & Norvig e.a.	AI die zelfstandig een doel opknipt in taken, tools gebruikt en resultaat oplevert	AI-Native SAFe
Human-in-the-Loop	Mens-machine-interactie	Mensen beoordelen, sturen en corrigeren wat AI oplevert	AI-Native SAFe
Responsible AI / AI Governance	EU AI Act; NIST AI RMF; ISO/IEC 42001	AI transparant, veilig en binnen wet- en regelgeving inzetten	AI-Native SAFe
Prompt Engineering	Generative AI-praktijk	Opdrachten aan AI-modellen zo formuleren dat ze bruikbaar resultaat geven	Niet expliciet in SAFe
MLOps	Google e.a.; DevOps-gemeenschap	DevOps-principes toepassen op het bouwen, uitrollen en bewaken van AI-modellen	Niet expliciet in SAFe
Data Governance	DAMA International (DAMA-DMBOK)	Kwaliteit, eigenaarschap en beveiliging van data borgen	Niet expliciet in SAFe

5. Tot slot

Veel van deze kennis is te bestuderen bij de oorspronkelijke grondleggers. Voor de specifieke definities van de SAFe-constructies is de officiële SAFe-kennisbank de primaire bron. Begin met het probleem, niet met de poster.

Mocht je tijdens het lezen merken dat iets niet klopt, of heb je tips voor aanvullingen: laat het me gerust weten!

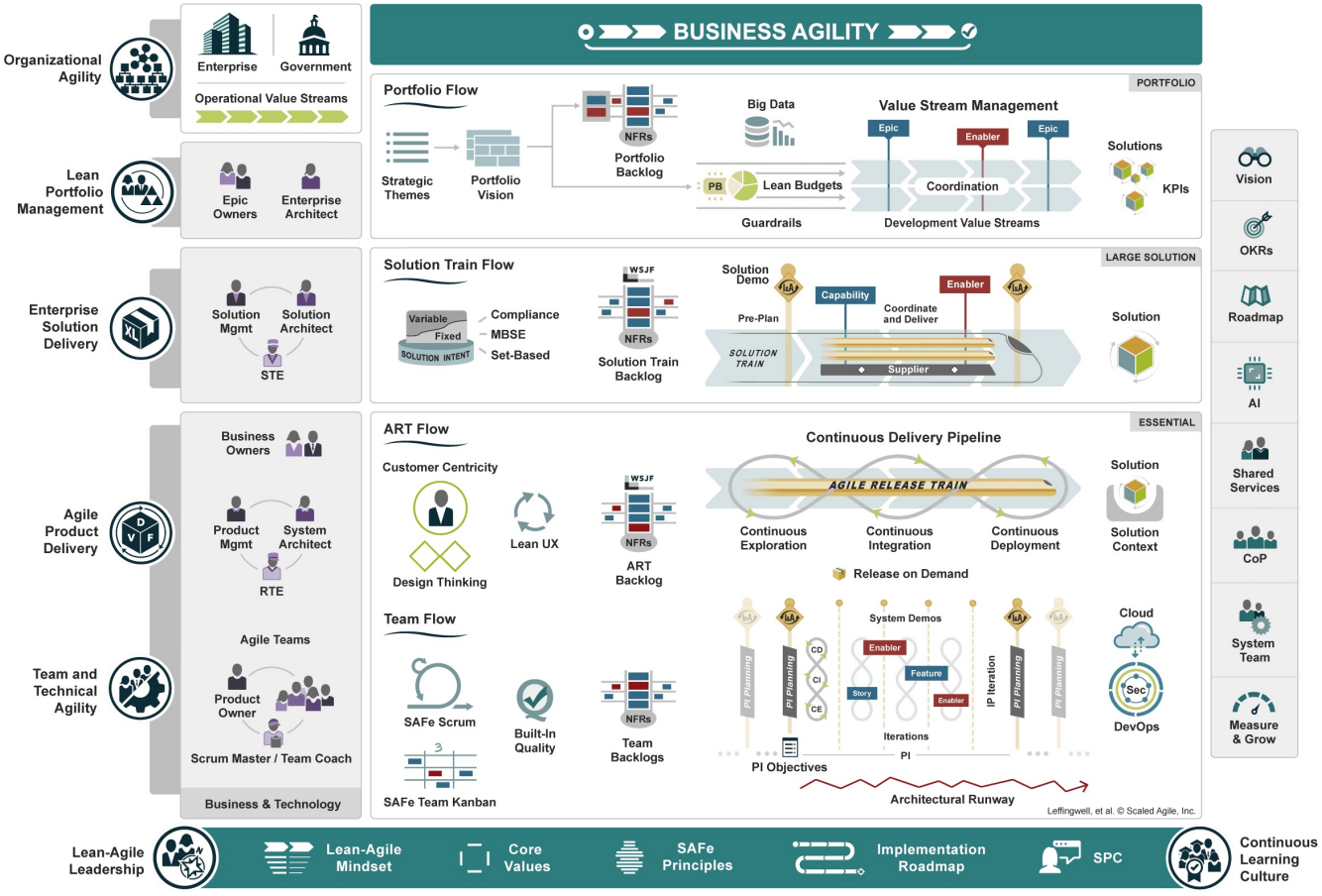
10. Het SAFe-jargon ontcijferd: ~30 woordjes om presentaties en workshops beter te begrijpen

Veel mensen vinden SAFe ingewikkeld. Dat komt vaak omdat SAFe, net als de grote bedrijven als ASML, Belastingdienst, UWV, Vodafone, NS ook een eigen jargon heeft. Wie dat jargon niet beheerst, kan dan de gesprekspartners ‘niet goed verstaan’. Bekend?

Goed nieuws: je hoeft niet álle SAFe woorden te kennen. Als je deze ruim 30 belangrijkste woorden kent, begrijp je 80% van wat er in een SAFe-presentatie of gesprek gezegd wordt.

In dit hoofdstuk geef ik een overzicht van de SAFe versie 6.0 **basiswoorden en de afkortingen** die je bij de SAFe implementatie snel moet leren kennen. Dit is namelijk het formele SAFe-jargon dat SAFe experts 1 op 1 gebruiken.

Dit is SAFe met de beroemde SAFe poster.



Alle woorden kun je ook terugvinden op deze SAFe poster, versie V6.0

Eerst wat algemene basisbegrippen die je moet kennen

Lean = Lean is een managementfilosofie en methode die gericht is op het maximaliseren van waarde voor de klant door het minimaliseren van verspilling in processen (Lean Six Sigma, BlackBelt, etc).

Agile Manifesto = het korte document uit 2001 waarin de 17 grondleggers van Agile hebben vastgelegd wat het betekent om Agilist te zijn: 4 Values met 12 principles.

Agile = Agile is een mindset: wendbaar werken in korte stappen met feedback, gebaseerd op de 4 waarden en 12 principes van het Agile Manifesto (2001). Een mindset is iets wat je *bent*, niet iets wat je *doet*. Je kunt regels

naleven zonder de mindset te hebben (cargo cult Agile), en je kunt de mindset hebben zonder elk principe strikt te volgen

Scrum = een manier van werken waarbij kleine teams in korte sprints van 2 tot 4 weken steeds een stukje werkende software of product opleveren, met vaste rollen (Scrum Master, Product Owner, Team) en events (dagstart, sprintplanning, review, retrospective).

Dev(Sec)Ops = bouwen en beheren samen: *you build it, you run it.* Ingebouwde security vanaf dag 1. Teams onderhouden en beheren bij voorkeur zelf de oplossingen die zij bouwen.

SAFe = SAFe wordt ingezet om grote enterprises agile te maken ('Enterprise Business Agility'). SAFe staat voor Scaled Agile Framework. Het is een raamwerk (‘Framework’) dat wordt gebruikt om agile werkwijzen op te schalen binnen grotere organisaties, waarbij meerdere teams samenwerken aan complexe geïntegreerde oplossingen. Het helpt bedrijven om flexibel en efficiënt te werken, terwijl ze de uitdagingen van schaal en complexiteit aangaan. SAFe brengt inzichten en praktijken uit Lean, Agile, DevOps en Systems Thinking samen in één geïntegreerd raamwerk.

Portfolio SAFe: strategie & besturing

Lean Portfolio Management (LPM): het hoogste orgaan en level in SAFe, zie poster, het niveau dat beslist waar de organisatie tijd en geld in steekt. Hier worden de prioriteiten bepaald en ‘de Epics gedefinieerd’

Lean Portfolio Leadership - de leiders, beslissers op hoogste SAFe level die strategie koppelen aan uitvoering (Business Owners, EA’s, Epic Owners).

Epic: een groot initiatief (meerdere kwartalen werk, meerdere teams). Alhoewel een Epic *geen* Project is in SAFe, mag je dat wel zo even onthouden. Met een Minimum Viable Product (MVP) toets je de verwachte businessresultaten. Daarna besluit je door te gaan, bij te sturen of te stoppen. Een Epic wordt afgesloten wanneer is besloten te stoppen of wanneer verdere sturing op portfolioniveau niet meer nodig is. Niet alle ooit voorziene Features hoeven daarvoor gerealiseerd te zijn.

Portfolio Backlog: de lijst met alle Epics.

Business Owner (BO): vertegenwoordiger van de Business, die de waardecreatie bewaakt.

Epic Owner (EO) - De Epic Owner begeleidt een epic door de voorbereidende fases van Epic en bij implementatie door de organisatie.

Enterprise Architect (EA) - eigenaar van de **technische koers** (hoe zorgen we dat alles past in het grotere geheel en duurzaam blijft).

LACE (Lean-Agile Center of Excellence) - team dat de SAFe-transformatie drijft (trainen, coachen, cultuur). Strategisch.

VMO (Value Management Office) - Coördineert ART’s, haalt Metrics (dit zijn meetbare waarden die aangeven hoe goed een organisatie, team of proces presteert ten opzichte van een bepaald doel) op, etc. Alle mensen die nodig zijn om de implementatie van LPM te ondersteunen. Tactisch operationeel.

Agile Release Train (ART)

Agile Release Train (ART): 50 tot 125 mensen die samen hetzelfde product maken. Een ART is ‘a team of agile teams’

Release Train Engineer (RTE): de conducteur van de trein; zorgt dat alles loopt. De RTE wordt ook wel ‘de master of Scrum Masters' genoemd. De RTE implementeert de ontwerpen van de LACE en begeleidt en faciliteert zijn Agile Release Train (ART) en is verantwoordelijk voor de embedding van de Agile/SAFe Way of working binnen zijn ART (dus incl de Agile teams). De RTE stimuleert continue verbetering bij Teams, de ART en de totale organisatie en rapporteert deze terug aan de LACE via Metrics.

PM (Product Manager) - eigenaar van de ART Backlog en stuurt op Features en waarde. De PM is ook wel de Master of the PO's. De PM zorgt ervoor dat de gewenste functionaliteiten worden gebouwd en ontwikkelt goede productvisies en productroadmaps.

SA (System Architect/Engineer) bepaalt de technische richting en architectuur van de ART. Hij zorgt ervoor dat de non-functionele requirements (NFR's) worden gehaald.

System Team - ondersteunt de ART met integratie, testen en CI/CD.

Feature - een middelgrote behoefte of functionaliteit die waarde levert voor de gebruiker of business. Een Feature is klaar als alle user stories in die feature klaar zijn.

ART Backlog: Een overzicht van alle Features die moeten worden ontwikkeld.

Agile Team (AT)

Agile Team: ~10 mensen die met elkaar in korte sprints (ook wel genoemd iteraties) waarde leveren. Tegelijk ontwikkeling en beheer doen, dus DevOps.

Scrum Master (SM): coach voor het team. Daarvan is er altijd maar 1, maar die mag in deeltijd werken.

Product Owner (PO): bepaalt wat het team bouwt en maximaliseert de op te leveren waarde per iteratie. Daarvan is er altijd maar 1.

Dev-team member - lid van een Agile Team (ontwikkelaar, tester, UX'er, analist, etc.) die samen met andere dev team members verantwoordelijk is voor ontwerpen, bouwen, testen en opleveren van werkende software of systemen.

User Story: klein stukje werk dat in een Iteration past.

Tot slot

Iteration (Bij Scrum heet dat een Sprint): In theorie een periode van 2 weken.

PI (Planning Interval): periode van 8 tot 12 weken waarin de ART plant en levert. In 1 PI zitten 4 tot 6 iteraties, meestal 5 (4 ontwikkeliteraties + 1 IP Iteration).

PI Planning: gezamenlijke tweedaagse planningsbijeenkomst van alle teams binnen 1 ART. De teams werken samen met stakeholders om het volgende increment te plannen. Wanneer je 10 ART's hebt, zijn er 10 PI Planningen die allemaal parallel lopen.

Inspect & Adapt (I&A) - Aan het einde van elke Planning Interval (PI) is er een groot 'inspect' event om samen terug te kijken op de afgelopen PI, problemen te analyseren en verbeteringen af te spreken. In dit event worden de metrics besproken, de performance van de ART.

IP Iteration (Innovation & Planning Iteration): de laatste iteratie in een PI (meestal de 5e), gereserveerd voor innovatie, verkenning en de voorbereiding op de volgende PI Planning. Er worden geen reguliere Features opgeleverd. Teams gebruiken deze iteratie voor hackathons, het wegwerken van technische schuld, leren en het PI Planning event zelf.

CI/CD (Continuous Integration / Continuous Delivery): twee samengevoegde DevOps-praktijken. CI: ontwikkelaars voegen hun code continu samen en testen dit automatisch. CD: de software is altijd klaar om uit te rollen naar productie. Samen zorgen CI/CD voor sneller, betrouwbaarder en risicoarm leveren.

CDP (Continuous Delivery Pipeline): de SAFe-term voor de volledig geautomatiseerde keten van idee tot productie. De CDP bestaat uit 4 fasen: Continuous Exploration (wat bouwen we?), Continuous Integration (bouwen en testen), Continuous Deployment (automatisch uitrollen) en Release on Demand (wanneer en hoe leveren we aan de klant). De CDP maakt het mogelijk om snel en veilig waarde te leveren.

Value Stream - de volledige keten van stappen die nodig zijn om waarde te leveren aan de klant: van idee tot werkend product in gebruik. In SAFe onderscheid je twee typen: een Operational Value Stream (het dagelijkse bedrijfsproces) en een Development Value Stream (het bouwen van de oplossing). Een ART is georganiseerd rondom een Development Value Stream.

✓ Samenvatting

Ken je deze woorden, dan kun je de meeste SAFe gesprekken en presentaties bij je eigen organisatie volgen. Je zult merken dat het jargon snel vertrouwd gaat voelen.

Dus wees niet ongerust, alles komt goed met je gedurende een complexe SAFe transformatie !

11. SAFe leren kennen: video’s en boeken

Nieuw met SAFe, of wil je je kennis even opfrissen? In dit hoofdstuk staat een selectie van video's en boeken die ik zelf gebruik in trainingen en transformaties. Begin met de korte video's over Agile en Scrum, ga daarna door naar de SAFe-intro's en gebruik de boeken voor verdieping.

Videotips

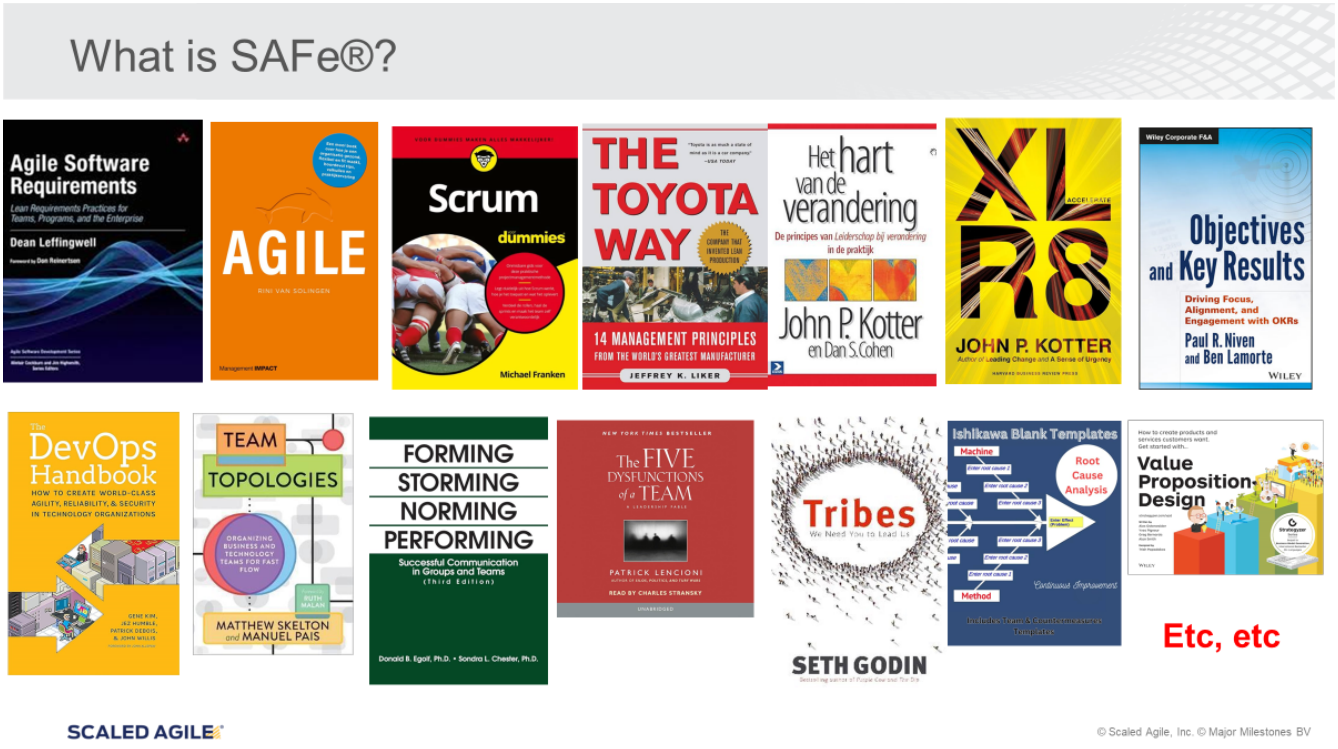
Agile en Scrum: de basis

Video	Waar gaat het over?	Link
Why Agile?	Korte video over waarom organisaties Agile gaan werken.	Bekijk video
What is Agile?	Minivideo over de kern van Agile.	Bekijk video
Incrementeel ontwikkelen	Waarom je in kleine stappen waarde oplevert.	Bekijk video
What is Scrum?	Scrum is géén methodologie, maar een framework: een empirisch procesmodel.	Bekijk video
Is Scrum Agile? The same or different?	De overeenkomsten en verschillen tussen Scrum en Agile.	Bekijk video
Trust versus Performance (Simon Sinek)	Simon Sinek over hoe de Navy SEALs teamleden kiezen: vertrouwen weegt zwaarder dan prestatie.	Bekijk video

SAFe

Video	Waar gaat het over?	Link
Algemene intro van SAFe (5 min)	Een algemene introductie op het framework.	Bekijk video
Lean versus Agile	SAFe brengt inzichten en praktijken uit Lean, Agile, DevOps en Systems Thinking samen in één geïntegreerd raamwerk. Deze video legt het verschil tussen Lean en Agile helder uit.	Bekijk video
Dual Operating System	SAFe kent twee operating systems: de Lijn en het Value Stream Network, afgeleid van het boek Accelerate (XLR8) van John Kotter.	Bekijk video
Hoe bereik je Enterprise Business Agility?	Een organisatie heeft SAFe volledig geïmplementeerd als alle core competenties succesvol draaien. Deze video behandelt de zeven core competenties van SAFe 6.0; sinds 2025 zijn die gebundeld in vijf disciplines (zie hoofdstuk 1).	Bekijk video

Boekentips



SAFe in cijfers



Deel 2: Theoretische fundamenten

12. Verschillen en overeenkomsten tussen frameworks, methodologieën en empirisme

Inleiding

Tijdens de SAFe Trainingen komen altijd een paar termen voorbij, zoals **Frameworks**, **methodologieën**, en **empirisme**. Hun betekenis en toepassing verschillen echter aanzienlijk. Dit hoofdstuk onderzoekt deze woorden, legt hun definities uit en bespreekt de verschillen en overeenkomsten tussen hen, met specifieke aandacht voor SAFe, Scrum, PRINCE2 en PPM.

Wat is een methodologie?

Een **methodologie** is een prescriptieve set van richtlijnen, processen en regels die gevolgd moeten worden om een project of programma succesvol uit te voeren. Methodologieën zoals **PRINCE2** en **PPM** zijn ontworpen om consistentie en voorspelbaarheid te waarborgen in project- en programmamanagement. Ze zijn sterk prescriptief, wat betekent dat ze een vaste reeks stappen, rollen, en procedures voorschrijven die nauwgezet gevolgd moeten worden.

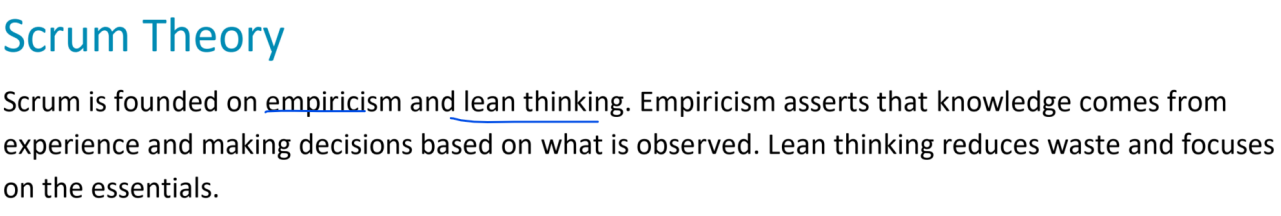
• **PRINCE2**: gericht op projectmanagement, biedt een gedetailleerde structuur voor het beheersen en uitvoeren van projecten door ze in beheersbare stadia te verdelen.

• **PPM**: gericht op Project & Portfolio Management, helpt bij het coördineren van meerdere gerelateerde projecten om strategische bedrijfsdoelen te realiseren.

Wat is empirisme?

Empirisme is een filosofische benadering die stelt dat kennis voortkomt uit ervaring en observatie. Francis Bacon staat bekend als de grondlegger van de empiristische stroming. Tegenover het empirisme staat het rationalisme, de stroming die beweert dat kennis enkel gebaseerd kan zijn op het denken of de rede.

• **SAFe Scrum**: Een Agile aanpak in het SAFe Framework die gebruik maakt van empirisme om teams te laten leren van ervaring. Scrum werkt in korte iteraties, genaamd sprints in Scrum en iteraties in SAFe, waarin feedback wordt gebruikt om continu te verbeteren.



Screenshot uit de Scrum Gids editie 2020

Wat zijn frameworks?

Een **framework** biedt een gestructureerde context, ofwel een raamwerk, voor het uitvoeren van taken, maar is minder rigide dan een methodologie. Frameworks bevatten ook richtlijnen en principes die gevolgd moeten worden (SAFe zonder PI kan niet, SAFe zonder Agile Teams ook niet), maar bieden meer flexibiliteit om aan specifieke situaties en behoeften aan te passen. (WSJF kan, hoeft niet, Solution Trains mogen, maar hoeven niet)

• **SAFe** (Scaled Agile Framework): SAFe is een aanpasbaar framework (en dus geen methodologie) dat organisaties helpt om Agile-principes, zoals die in Scrum en Kanban worden gebruikt, op grotere schaal,

geschaald, toe te passen. Een aangepaste versie van Scrum, genaamd "SAFe Scrum" zit in SAFe, dus ook SAFe is gebaseerd op het Empirisme.

Prescriptief versus flexibel

Prescriptief betekent dat een methode of framework specifieke stappen en regels voorschrijft die gevolgd moeten worden. Methodologieën zoals PRINCE2 en PPM zijn sterk prescriptief en bieden weinig ruimte voor afwijking. Frameworks zoals SAFe zijn daarentegen minder strikt en bieden meer flexibiliteit. Ze geven een raamwerk waarbinnen teams kunnen werken, maar laten ruimte voor aanpassing op basis van de context.

Conclusie

Methodologieën, frameworks en empirisme hebben elk hun plaats in projectmanagement en softwareontwikkeling. **Methodologieën** zoals PRINCE2 en PPM zijn prescriptief en bieden gedetailleerde processen die strikt gevolgd moeten worden. **Frameworks** zoals SAFe bieden een gestructureerd, maar flexibel, raamwerk dat aanpasbaar is aan specifieke omstandigheden.

Empirisme, zoals toegepast in Scrum (en dus indirect ook SAFe), legt de nadruk op leren door ervaring en continue verbetering. Door deze benaderingen te begrijpen en te kiezen welke het beste past bij de behoeften van een project of organisatie, kunnen teams effectiever werken en betere resultaten behalen.

13. Het Agile Manifesto: de core van SAFe met een uitleg

Inleiding: wat is het Agile Manifesto?

Het Agile Manifesto, opgesteld in 2001 door zeventien softwareontwikkelaars, waaronder 1 Nederlander alsmede beide bedenkers van Scrum, is de basis van de Agile-werkwijze. Het manifesto werd ontworpen om softwareontwikkeling **flexibeler**, **effectiever** en **klantgerichter** te maken. Hoewel het oorspronkelijk bedoeld was voor softwareontwikkeling, is het inmiddels breed toegepast in andere sectoren.

Het Agile Manifesto bestaat uit **vier kernwaarden** en **twaalf principes** die organisaties en teams begeleiden bij het leveren van waarde in een veranderende wereld. Het manifesto benadrukt samenwerking, flexibiliteit en continue verbetering, waardoor teams sneller kunnen inspelen op veranderende eisen en klantbehoeften.

Als je zegt dat jij een ‘true agilist’ bent, verwacht ik dat je de onderliggende mindset, principes en filosofie nastreeft en zo goed als mogelijk naleeft en als daily mindset hebt.

De vier kernwaarden van het Agile Manifesto

De kernwaarden van het Agile Manifesto bieden een filosofische leidraad. Ze helpen teams focussen op wat belangrijk is voor het leveren van waarde.

1. Individuen en interacties boven processen en tools

De mens staat centraal in Agile. Effectieve communicatie en samenwerking binnen en tussen teams zijn belangrijker dan het strikt volgen van processen of het gebruiken van tools. Tools en processen ondersteunen het werk, maar mogen nooit de interactie tussen teamleden belemmeren.

Wat betekent dit in de praktijk?

• **Wel:** Creëer een cultuur van vertrouwen en openheid waarin teamleden gemakkelijk informatie delen.

• **Niet:** Vertrouwen op tools om problemen op te lossen zonder aandacht voor menselijke interactie.

Veelvoorkomende valkuilen:

• Teams die te afhankelijk zijn van tools, zoals Jira of Trello, zonder aandacht voor samenwerking.

• Overmatig procesgericht werken dat innovatie en creativiteit belemmert.

2. Werkende software boven uitgebreide documentatie

Het doel is om waarde te leveren aan de klant door functionerende software. Dit betekent dat documentatie ondersteunend moet zijn aan het opleveren van werkende oplossingen, zonder een onnodige belasting te vormen. Het gaat om zowel **projectdocumentatie als productdocumentatie.** In Agile gaat het erom dat beide typen documentatie worden beperkt tot wat nodig is, zonder overbodige complexiteit of tijdsverlies.

Wat je moet doen:

• Prioriteer het opleveren van kleine, werkende software-iteraties.

• Zorg dat projectdocumentatie gericht is op communicatie en het ondersteunen van teams.

• Maak productdocumentatie actueel en relevant voor gebruikers en ontwikkelaars.

Wat je niet moet doen:

• Uren besteden aan documentatie die niet wordt gelezen of gebruikt.

• Documentatie belangrijker maken dan het daadwerkelijk opleveren van software.

• Geen documentatie maken, wat overdracht en onderhoud kan bemoeilijken.

Veelvoorkomende fouten:

• Overdocumentatie, zoals uitgebreide technische handleidingen voordat er werkende software is.

• Teams die helemaal geen documentatie maken, wat problemen veroorzaakt bij overdracht en onderhoud.

• Verouderde of irrelevante documentatie die niet aansluit bij het product.

• 1,5 jaar lang PV, PK's en PID gaan schrijven zonder ook maar 1 regel code te hebben opgeleverd met een grote delay in 'the time to market'

3. Samenwerking met de klant boven contractonderhandelingen

Klantbetrokkenheid is essentieel in Agile. Klanten worden gezien als partners die regelmatig feedback geven om ervoor te zorgen dat het eindproduct aan hun behoeften voldoet.

Wat betekent dit in de praktijk?

• **Wel:** Betrek klanten frequent via demo's, feedbacksessies en prioritering. Dit is standaard embedded in SAFe.

• **Niet:** een "wij versus zij"-mentaliteit ontwikkelen door alleen naar contractafspraken te kijken.

Veelvoorkomende valkuilen:

• Klanten pas betrekken aan het einde van het project, wat vaak leidt tot teleurstelling.

• Star vasthouden aan contracten en eisen, zelfs als de behoeften veranderen.

4. Reageren op verandering boven het volgen van een plan

De wereld verandert continu, en Agile teams passen zich aan deze veranderingen aan. Flexibiliteit is belangrijker dan rigide vastklampen aan vooraf gemaakte plannen. Zie ook principe #2.

Wat betekent dit in de praktijk?

• **Wel:** Werk in korte iteraties (zoals sprints) en herzie prioriteiten regelmatig. Per iteratie en per PI.

• **Niet:** Star vasthouden aan oorspronkelijke plannen, zelfs als duidelijk is dat deze niet langer relevant zijn.

Veelvoorkomende valkuilen:

• Projecten/Epics die nog steeds worden uitgevoerd volgens een "waterfall"-aanpak, met vaste plannen van begin tot eind.

• Angst om aanpassingen door te voeren, zelfs wanneer deze nodig zijn.

• Reageren op veranderingen is prima, maar het is ook niet de bedoeling dat de klant 'alle kanten opgaat'. Goede inrichting van de SAFe CE fase is belangrijk.

De twaalf principes van het Agile Manifesto

De twaalf principes van het Agile Manifesto geven verdere diepgang aan de kernwaarden en bieden praktische richtlijnen voor teams die Agile willen werken. Hieronder volgt een uitgebreide uitleg van elk principe:

1. Onze hoogste prioriteit is het tevredenstellen van de klant door het vroegtijdig en continu leveren van waardevolle software.

Klanttevredenheid staat centraal in Agile. Dit wordt bereikt door regelmatig waarde te leveren (Cost of Delay), zodat klanten al vroeg in het proces voordelen ervaren. Door kortcyclisch te werken, kan feedback snel worden verwerkt en kunnen producten beter aansluiten op klantbehoeften. Continu opleveren voorkomt lange wachttijden en minimaliseert risico's. Teams moeten prioriteit geven aan de functies die de meeste waarde toevoegen, in SAFe conform WSJF. Dit principe vereist een cultuur van voortdurende verbetering en samenwerking met de klant. Het stimuleert ook transparantie, omdat klanten voortdurend inzicht hebben in de voortgang. Klantbetrokkenheid is hierbij essentieel, zodat verwachtingen goed gemanaged worden.

2. Verwelkom veranderende eisen, zelfs laat in het ontwikkelingsproces. Agile processen benutten verandering om de concurrentiepositie van de klant te vergroten.

In een dynamische omgeving veranderen klantbehoeften vaak. Agile-teams zien dit niet als een obstakel, maar als een kans *om meer waarde* te leveren. Flexibele planningen en iteratief werken maken het mogelijk om snel in te spelen op veranderingen. Door iteratieve oplevering blijven aanpassingen beheersbaar en voorkomen ze grote verstoringen. Teams moeten voorbereid zijn om plannen te herzien zonder terughoudendheid. Het vereist een open mindset en de bereidheid om te experimenteren. Dit principe stimuleert innovatie en concurrentievoordeel. Klanten profiteren van producten die beter aansluiten bij hun veranderende behoeften.

3. Lever regelmatig werkende software op, van een paar weken tot een paar maanden, met de voorkeur voor kortere tijdsperiodes.

Frequent opleveren verkleint risico's en verhoogt de transparantie. Klanten zien sneller resultaten, *waardoor ze vertrouwen krijgen in het proces*. Korte oplevercycli, zoals sprints, zorgen voor een ritme en maken voortgang beter meetbaar. Regelmatige releases stimuleren samenwerking tussen teams en stakeholders. Dit principe vereist een focus op werk dat direct waarde toevoegt. Door kleine iteraties te gebruiken, blijven Epics wendbaar en beter controleerbaar. Problemen worden sneller zichtbaar en opgelost. Dit draagt bij aan een hoger eindresultaat en grotere klanttevredenheid.

4. Businessmensen en ontwikkelaars moeten dagelijks gedurende het hele project samenwerken.

Agile-teams doorbreken de traditionele kloof tussen business en IT. Door dagelijks samen te werken, ontstaan betere oplossingen en meer begrip voor elkaars perspectief. Dit versterkt de samenwerking en voorkomt miscommunicatie. Teams moeten een gedeelde verantwoordelijkheid voelen voor het succes van het project. Face-to-face interacties (of virtuele equivalenten) bevorderen vertrouwen en efficiëntie. Samenwerking maakt het eenvoudiger om prioriteiten te stellen en knelpunten snel op te lossen. Dit principe vereist een cultuur waarin communicatie en samenwerking worden gewaardeerd. Het helpt om het uiteindelijke product beter af te stemmen op de zakelijke behoeften.

5. Bouw projecten rond gemotiveerde individuen. Geef hen de omgeving en ondersteuning die ze nodig hebben, en vertrouw erop dat ze hun werk doen.

Gemotiveerde medewerkers leveren de beste resultaten conform SAFe's Lean-Agile Principle #8. Teams hebben autonomie nodig om beslissingen te nemen en verantwoordelijkheid te nemen voor hun werk. Leiderschap moet zich richten op het *wegnemen van obstakels* en het *bieden van de juiste middelen.* Vertrouwen is hierbij cruciaal; *micromanagement ondermijnt motivatie*. Dit principe moedigt een cultuur van eigenaarschap aan. Teams die zich gesteund voelen, zijn creatiever en productiever. Het vereist ook aandacht voor een gezonde werkbalans om motivatie op lange termijn te behouden. Door medewerkers de ruimte te geven, bloeit innovatie en betrokkenheid op.

6. De meest efficiënte en effectieve manier om informatie over te brengen binnen een ontwikkelteam is een face-to-face gesprek.

Communicatie is effectiever wanneer deze direct plaatsvindt. Face-to-face gesprekken met zowel verbale als non-verbale feedback, verminderen misverstanden en versnellen besluitvorming. Het bevordert ook een persoonlijke connectie tussen teamleden, wat vertrouwen versterkt. Tools zoals videovergaderingen zijn nuttig, maar directe menselijke interactie blijft het krachtigst. Dit principe benadrukt het belang van een open en transparante communicatiecultuur. Het vereist een omgeving waarin teamleden zich vrij voelen om ideeën en zorgen te delen. Door barrières in communicatie weg te nemen, verbeteren snelheid en kwaliteit. Dit principe bevordert ook samenwerking en gedeeld begrip binnen het team. SAFe bevordert dit met onder andere de Team Sync, big room planning meetings etc.

7. Werkende software is de primaire maatstaf voor voortgang.

Agile legt de focus op het opleveren van werkende oplossingen in plaats van op documentatie of plannen. Werkende software toont tastbare vooruitgang en bewijst dat het team waarde levert. Dit sluit aan bij SAFe's Lean-Agile principe #5. Beide principes helpen om focus te houden op wat belangrijk is: klantwaarde. Door iteratief te werken, kunnen teams sneller zien of ze op de juiste weg zijn. Het vereenvoudigt ook de communicatie met stakeholders, omdat resultaten direct zichtbaar zijn. Dit principe stimuleert teams om prioriteit te geven aan werk dat het product verbetert. Het vereist een cultuur van continue levering en snelle feedbackloops. Dit verhoogt de transparantie en het vertrouwen binnen en buiten het team.

8. Agile processen bevorderen duurzame ontwikkeling. Sponsors, ontwikkelaars en gebruikers moeten een constant tempo kunnen volhouden.

Een duurzaam (sustainable) werktempo is cruciaal voor langdurig succes. *Overbelasting* leidt tot burn-out en kwaliteitsverlies. Agile-teams streven naar een evenwichtige workload die consistent kan worden volgehouden. Dit principe moedigt planning aan die realistisch en haalbaar is. Het vereist ook dat teams leren van eerdere iteraties om hun tempo te optimaliseren. Door een gezonde werkbalans te behouden, blijven teams productief en gemotiveerd. Het versterkt de kwaliteit van het werk, omdat er ruimte is voor reflectie en verbetering. Dit principe bevordert een cultuur van continu leren en verbeteren.

9. Blijf voortdurend aandacht besteden aan technische excellentie en goed ontwerp om wendbaarheid te verhogen.

Technische excellentie is de *ruggengraat van Agile en een ware agilist.* Goede ontwerpen en hoogwaardige code maken het eenvoudiger om veranderingen door te voeren. Dit principe moedigt teams aan om te investeren in onderhoudbaarheid en schaalbaarheid. Het vereist een cultuur waarin kwaliteit prioriteit heeft boven snelheid. Door te focussen op technische uitmuntendheid, worden problemen in een vroeg stadium opgelost (Shift Learning Left). Het versterkt ook het vertrouwen van klanten en stakeholders. Dit principe vraagt om discipline en toewijding van het team. Het stimuleert innovatie en een duurzame aanpak van productontwikkeling.

10. Houd het simpel: maximaliseer de hoeveelheid werk die niet wordt gedaan.

Eenvoud ('Lean') is een kernprincipe in Agile. Door onnodige complexiteit te vermijden, kunnen teams zich focussen op wat echt waarde toevoegt. Dit principe moedigt teams aan om regelmatig te evalueren welke taken overbodig zijn en welke middelen bijdragen aan welke doelen. Het helpt ook om de kosten en doorlooptijd te verminderen. Simpliciteit vereist echter een scherp oog voor prioriteiten en een duidelijke visie. Dit principe bevordert wendbaarheid, omdat minder werk betekent dat teams sneller kunnen reageren. Het vraagt om discipline om alleen *het essentiële te doen.* Dit verhoogt de effectiviteit en verlaagt de kans op fouten.

11. De beste architecturen, eisen en ontwerpen komen voort uit zelforganiserende teams.

Zelforganiserende teams hebben de vrijheid en verantwoordelijkheid om beslissingen te nemen. Dit leidt tot innovatie en eigenaarschap. Teams die hun eigen werkwijze bepalen, presteren beter en leveren hogere kwaliteit. Dit principe moedigt leiders aan om vertrouwen te hebben in de expertise van hun teams. Zelforganisatie vereist een cultuur waarin fouten worden gezien als leermogelijkheden. Het versterkt samenwerking en gedeeld begrip binnen het team. Dit principe helpt *om silo's te doorbreken* (Conway's law) en wendbaarheid te vergroten. Het vraagt om leiderschap dat inspireert in plaats van controleert.

12. Evalueer regelmatig hoe je effectiever kunt worden en pas je gedrag dienovereenkomstig aan.

Continue verbetering is essentieel in Agile. Teams moeten regelmatig reflecteren op hun prestaties (People, Process, Technology) en leren van ervaringen. Retrospectives zijn een belangrijk hulpmiddel om verbeterpunten te identificeren. Dit principe vereist een open cultuur waarin feedback wordt gewaardeerd. Het stimuleert teams om experimenten uit te voeren en aanpassingen te doen. Door regelmatig te evalueren, blijven teams groeien en presteren ze beter. Dit verhoogt ook de betrokkenheid en motivatie van teamleden. Het principe moedigt een cultuur van innovatie en aanpassing aan die essentieel is voor succes.

Conclusie

Het Agile Manifesto vormt de basis voor wendbaar werken door flexibiliteit, samenwerking en klantgerichtheid te benadrukken en is de foundation van SAFe. Excellente kennis van het manifesto is extreem belangrijk voor onze Agile Transformatie, onze cultuurverandering en mindset.

De principes van SAFe bouwen hierop voort en maken het mogelijk om Agile toe te passen in complexe schaalbare grote organisaties. Samen bieden deze kaders een krachtige aanpak om waarde te leveren, aan te passen aan veranderingen en teams optimaal te laten presteren.

14. Professionals denken en doen MECE

Een groot gedeelte van ons leven houden we ons bezig met het maken van lijstjes, rijtjes en vakjes. We sorteren, categoriseren en ordenen. Denk hierbij in de privésfeer aan het indelen of opruimen van de kledingkast. Of in het bedrijfsleven aan het inrichten van een mappenstructuur. Of aan het maken van een opsomming met de top vijf problemen na een uitgevoerde organisatiediagnose.

Toch blijkt ordenen in de praktijk lastig. Dikwijls tref ik op afdelingsschijven een brij van informatie aan. Dit belemmert de toegankelijkheid. Gestelde organisatiediagnoses zijn vaak niet meer dan een wazige opsomming van onsamenhangende punten. Daardoor dringen deze vaak niet door tot de echte kern van het probleem. Dit alles heeft tot gevolg dat je als klant, toehoorder of gebruiker je doel mist: op de meest efficiënte en effectieve wijze toegang verkrijgen tot het aanbod.

De oplossing is even slim als handig: leer MECE te denken en doen.

Ordenen wordt als lastig ervaren, met alle gevolgen van dien

Hieronder staan enkele voorbeelden:

Toegang tot de juiste deur

Een aannemer moet een nieuw overheidscomplex opleveren. Uiteraard ontbreekt daarbij de noodzakelijke openbare sanitaire voorziening niet. Enthousiast richt hij de sanitairblokken in, maar plaatst de volgende borden op de deuren:



Eigen illustratie: Major Milestones.

• Is hier volkomen duidelijk welke deur te kiezen?

• Welke kiest een jongetje? Een puber? Een volwassene?

• Hoe bepaal je welke de juiste voor jou is?

Toegang tot de kern van de zaak

Een organisatieadviseur heeft onderzoek verricht naar eventuele problemen bij het invoeren van performance management binnen een organisatie. Hij presenteert zijn voorstel als volgt bij de directie:

De conclusie van ons onderzoek luidt:

1. Invoering van performance management heeft vergaande effecten op de organisatie.

2. Invoering van performance management heeft een ongewenste impact op de flexibiliteit van de medewerkers.

3. Individuele performance contracten vergroten de spanningen tussen de betrokkenen door tegengestelde belangen.

4. Invoering van performance management gaat ervan uit dat de organisatiedoelen voorlopig niet veranderen.

5. Performance management is erop gebaseerd dat alle doelstellingen meetbaar zijn.

• Komt deze opsomming direct tot de kern van het probleem?

• Wat is een mogelijke relatie tussen punt 1, 2 en 3? Zijn 2 en 3 in feite ook geen effecten te noemen?

Toegang tot het elektronisch archief

Een projectmanager heeft een offerte voor een klant opgesteld als onderdeel van zijn werkzaamheden voor project X. Bij het archiveren ziet hij twee opties:

C:\klanten

C:\projecten

Is hier voor een nieuwe projectmanager volkomen duidelijk waar de offerte moet worden opgeslagen?

Toegang tot het gepresenteerde aanbod van diensten

Een interim-managementbureau adverteert met de volgende zinsnede:

"Onze organisatie levert interim management-, projectmanagement- en consultancydiensten"

• Betekent dit dat een interim manager en een projectmanager verschillende diensten leveren?

• En dat een interim manager noch adviseert, noch projectmanagement bedrijft, maar blijkbaar iets anders doet?

Al deze voorbeelden hebben één grote gemene deler: een brij van meer of minder uitgebreide informatie moet zo efficiënt mogelijk worden opgedeeld. Zodanig dat eenieder op logische en eenduidige wijze toegang kan verkrijgen tot de gekozen onderdelen: tot de juiste map, het wezenlijke probleem of de juiste deur naar het gewenste toilettenblok.

Logisch ordenen wordt als lastig ervaren

In bovenstaande voorbeelden blijkt dat logisch ordenen een vak apart is. Veel mensen vinden het lastig. Hiervoor zijn drie hoofdredenen aan te wijzen:

1. **Te weinig plezier**: niet iedereen krijgt energie van het analyseren van een wiskundige en/of taalkundige puzzel. Hierdoor nemen sommigen te weinig tijd om na te denken. Later weten ze dan zelf niet meer waarom ze voor een bepaalde ordening kozen. Immers: *"wazige gedachten leveren wazige ordeningen op; logische gedachten logische ordeningen"*. Logische ordeningen zijn beter overdraagbaar.

2. **Te weinig talent**: niet iedereen is geboren met het talent dit soort werkzaamheden goed te verrichten. Zo vergt het bijvoorbeeld enig wiskundig inzicht.

3. **Te weinig kennis**: gebrek aan specifieke deskundigheid. Er is onvoldoende kennis van de kernkarakteristieken (wat doet een interim manager precies?). Dan is het moeilijk het verband te zien met andere objecten uit een min of meer gelijke klasse (wat doet een projectmanager precies?). Om goed te kunnen ordenen moet je het ordeningsprobleem kunnen doorzien. Bovendien moet je verstand hebben van de karakteristieken

van de objecten zelf. Deze moeten vervolgens goed en eenduidig worden gedefinieerd.

De oplossing heet MECE en is even effectief als simpel

MECE is een lekker in de mond liggend acroniem en is daardoor goed te onthouden. Het doel van MECE is de meest logische ordening van mappenstructuren, documenten, presentaties en probleemanalyses te vinden. Met als uiteindelijk effect: maximale kracht door bondigheid, helderheid en eenduidigheid.

Een ordeningsresultaat is MECE of niet MECE. MECE-uitkomsten vormen een overzichtelijk, logisch en compact geheel. Zonder gaten, zonder overlap, en hierdoor goed benaderbaar voor derden.

MECE (spreek uit als *"me-se"*) is oorspronkelijk ontstaan bij McKinsey, 's werelds meest prestigieuze bureau voor strategische managementadviezen. MECE staat voor **Mutually Exclusive, Collectively Exhaustive** (wederzijds uitsluitend en gezamenlijk uitputtend). McKinsey vindt kennis van MECE dermate belangrijk dat het van al haar medewerkers vereist dat zij MECE tot in de puntjes beheersen. McKinsey past MECE voornamelijk toe bij het formuleren van organisatiediagnoses en het opstellen van adviesdocumenten. Het is echter ook bijzonder nuttig voor interim managers, directieleden, publicisten en sprekers.

Alleen logische ordeningen leveren MECE-uitkomsten op

Verzamelingenleer

MECE is nauw verwant met de wiskundige logica en de verzamelingenleer. De laatste vormt sinds het begin van de twintigste eeuw een van de grondslagen van de wiskunde. Haar drie belangrijke basisbegrippen zijn:

• deelverzameling

• doorsnede en vereniging van twee verzamelingen

• lege verzameling

Met de uit de verzamelingenleer afkomstige venn-diagrammen kun je visualiseren hoe de eigenschappen van verschillende zaken met elkaar samenvallen en hoe iets overzichtelijk op eigenschappen is te ordenen. Doel bij MECE is het uitsluiten van overlap, zodanig dat je geen lege verzamelingen overhoudt (= lege mappen). Pas dan zijn al je deelverzamelingen binnen je hoofdverzameling MECE.

Ordenen

MECE is ook nauw verwant met ordeningsprincipes. Er zijn veel manieren van ordenen. Je kunt je kledingkast ordenen naar expliciete karakteristieken: hemden bij hemden, broeken bij broeken. Of: leer bij leer, wol bij wol. Dit is het meest voor de hand liggend en het eenvoudigst.

Een andere en soms betere vorm van ordenen is **naar doel**. Je legt dan alle kledingstukken bij elkaar die te maken hebben met een bepaalde activiteit: golfkleding, motorkleding, etc. Bij een organisatie betekent dit dat je alle werknemers bij elkaar voegt die als doel hebben het leveren van een bepaalde dienst. Gezamenlijke doeleinden stimuleren samenwerking. Ordenen naar functiekarakteristieken (techneuten bij techneuten, sales bij sales) is minder effectief.

Ook kun je ordenen naar **impliciete karakteristieken**. Deze ordening ligt minder voor de hand en vraagt meer kennis van de objecten zelf. Het risico is dat voor iemand die niet bekend is met de achtergronden de ordening als onlogisch wordt ervaren.

Sorteren

Als laatste is MECE verwant met sorteerprincipes. Sorteren is het bepalen van de juiste rangorde tussen de uiteindelijke ordeningen. Van groot naar klein, links naar rechts, onbelangrijk naar belangrijk, ouder naar nieuwer, numeriek oplopend, etc.

MECE toegepast in de praktijk

Het streven naar MECE begint meestal bij een onsamenhangende brij van informatie, verdeeld in diverse objecten. We borduren voort op twee eerder genoemde voorbeelden:

• In het voorbeeld van de aannemer is sprake van een grote verzameling mensen. Deze moet zodanig worden opgedeeld dat eenduidige, gescheiden toegang tot de toiletblokken ontstaat.

• In het voorbeeld van het interim-managementbureau zien we een onsamenhangende brij van diensten. Deze moet zodanig worden gepresenteerd dat de dienstverlening volkomen helder is en elke dienst uniek is.

Stap 1: decompositie naar een of meer aggregatieniveaus

Allereerst moet je de brij van informatie opdelen naar een hoger of lager niveau van aggregatie. Je aggregeert naar iets: de gemeenschappelijke deler over de deelverzameling heen. Hier moet je naar zoeken.

Ingeval van de toiletten: de gemeenschappelijke deler heeft te maken met de karakteristieken van het object zelf. Een aggregatievraag: zijn er binnen de verzameling mensen meerdere soorten te onderscheiden? Jazeker: mannen en vrouwen. En dit is het goede antwoord, want de gekozen verzameling moet "wederzijds uitsluitend en gezamenlijk uitputtend" zijn.

Ingeval van het interim-managementbureau: de vraag luidt hier wat de relatie is tussen de werkzaamheden van een interim manager en die van een adviseur. Jacques Reijniers stelt in zijn boek *Interim management: het vak* dat interim managers zich kenmerken door zowel een operationele management- als een strategische verandermanagementrol. Dit vergt drie specifieke vaardigheden: 1) management, 2) verander(project)management en 3) consultancy. Met andere woorden: de twee andere aangeboden diensten overlappen qua vaardigheden die van de eerste, de interim manager. Dit wordt echter alleen zichtbaar als voor iedere betrokkene duidelijk is wat interim management precies omvat.

Stap 2: de ordening op iets sorteren

Na decompositie kun je de geordende deelverzamelingen vervolgens sorteren. Bijvoorbeeld op numerieke waarde, op prioriteit, op volgorde van binnenkomst, etc. De mannendeuren links, de vrouwendeuren rechts. De rollen van de interim manager kun je opsommen op volgorde van oplopende complexiteit, of eenvoudig willekeurig sorteren.

Stap 3: de eindcontrole

Nu de eindcontrole: alle deelverzamelingen vertegenwoordigen tezamen het grotere geheel (*Collectively Exhaustive*). Ze sluiten daarbij niets uit, terwijl elke deelverzameling op zichzelf uniek is en niet overlapt met een andere verzameling (*Mutually Exclusive*). Een MECE-resultaat kent geen lege verzamelingen.

Stap 4: presentatie van het eindresultaat

Alle voorbeelden zijn nu strikt logisch naar een gekozen principe geordend en vervolgens gesorteerd. De eindresultaten zijn MECE.

Wees MECE en win aan professionaliteit

Goede MECE-toepassingen leveren veel op:

• **Organisatiestructuren**: voorkomen conflicten door het uitsluiten van overlap in werkzaamheden en functiegebieden. Structuren geordend naar doel bieden betere serviceverlening en zijn motiverend, omdat gezamenlijke doeleinden samenwerking stimuleren.

• **Dienstendefinities**: MECE-gedefinieerde diensten maken klanten duidelijk wat de organisatie levert. Een volledige MECE-dienstenopsomming komt door zijn bondigheid tot de kern. U laat zien dat u weet waarover u spreekt.

• **Mappenstructuren**: goede MECE-mappenstructuren bevorderen de toegankelijkheid, voorkomen dubbele opslag, comprimeren de mappenboom en vergemakkelijken het zoeken naar het juiste document.

Met andere woorden: voor alle adviseurs, consultants, directieleden en presentatoren wordt het eindresultaat compacter, leesbaarder en inhoudstechnisch beter. Er is zichtbaar over nagedacht.

Epiloog

Als ik vanavond thuiskom, stel ik misschien de voor iedereen bekende vraag: *"Schat, wat eten we vanavond?"* Een mogelijk antwoord zou kunnen zijn: *"Speklapjes, gebakken aardappeltjes met wat groenten en sla."*

Na het lezen van dit hoofdstuk weet elke MECE-professional dat ik dit antwoord niet zonder leuke discussie ga laten passeren.

15. Wat is waarde in SAFe? Over kosten, urgentie en Cost of Delay (deel 1)

Eerst een anekdote

Een grote fabriek had een machine die plotseling stopte met werken, wat leidde tot een complete stillegging van de productie. Geen van de technici in de fabriek kon de fout vinden of de machine repareren. Uiteindelijk besloten ze een ervaren monteur in te huren.

De monteur liep rond de machine, luisterde aandachtig, en bekeek alle onderdelen zorgvuldig. Na een paar minuten pakte hij een kleine hamer, gaf de machine een tik, en tot ieders verbazing begon de machine weer perfect te werken.

Een paar dagen later ontving de fabriek een factuur van de monteur voor €10.000. Verbaasd over de hoogte van het bedrag vroeg de fabriek om een gedetailleerde specificatie van de kosten.

De monteur stuurde de volgende specificatie:

• *Tik met hamer: €1*

• *Weten waar te tikken: €9.999*

Kosten versus waarde

Het bovenstaande verhaal illustreert perfect het verschil tussen de kosten van een actie en de waarde van in dit geval, de expertise. Het tikken met de hamer zelf was simpel en iets wat elke technicus zou kunnen doen. Wat de monteur echter onderscheidde, was zijn kennis en ervaring om precies te weten waar hij moest tikken. Dat is de toegevoegde waarde van zijn expertise, waarvoor hij mogelijk vele euro's heeft geïnvesteerd in opleidingen en certificeringen. In de anekdote stond er een machine stil en kon er niet worden geproduceerd, wat resulteerde in een hard kwantificeerbaar omzetverlies.

En zo komen we op een van de belangrijkste SAFe-begrippen, namelijk waarde, of in SAFe termen “Value”. Product Owners (PO’s), Product Managers (PM’s), Solution Managers (SM’s) en Epic Owners creëren allemaal waarde. Om die taak te benadrukken, zouden alle SAFe organisaties al hun PO’s eigenlijk moeten aanspreken als 'Value Maximiser'. Hun taak is om, met behulp van economische prioriteringsmechanismen, de maximale waarde uit een iteratie of increment te halen.

Maar wat is waarde exact?

Voor agilisten is het belangrijk eens in hun leven goed over het woord *Waarde* te hebben nagedacht. Wat is waarde? Hoeveel mag waarde kosten? Hoeveel levert een bepaalde handeling op?

Voordat je verder leest, wil ik je met een paar vragen aan het denken zetten:

1. Wat is de waarde van het plakken van de band van je fiets?

2. Wat is de waarde van je nieuwe auto?

3. Wat is de waarde van 1 SAFe training voor jou?

4. Wat is de waarde van een Rolling Stones kaartje 6 mnd voor aanvang concert? 30 min voor aanvang concert? En 1 maand na het concert?

5. Wat is de waarde van een advies van een gerenommeerde BTW-expert, die met 2 uur advies jou mogelijk 10 miljoen aan BTW-afdracht bespaart?

6. Wat is de waarde van een PI event?

Waarde is contextueel bepaald

Het issue is dat Waarde niet fixed is, maar altijd *contextueel* wordt bepaald. Waarde wordt namelijk bepaald door de volgende elementen:

Context en Urgentie: Als jouw fietsband lek is vlak voordat je met vakantie wilt en jij bijna geen andere opties hebt (zie onder), kan de **urgentie** enorm zijn. In dat geval is de waarde van jouw dienst hoger, omdat het oplossen van het probleem voor die klant relatief een veel hogere prioriteit heeft. In principe zou de fietsmonteur zijn tarief met een factor 2 kunnen verhogen, ware het niet dat het commercieel slim is dat niet te doen en te zorgen dat hij met deze actie zijn Reputatie verhoogt, of bijvoorbeeld een zeer hoge Google Review score erbij krijgt.

Alternatieven & unieke expertise: De prijs van leveranciers wordt mede bepaald door schaarste en uniciteit van de gevraagde kennis. Veel trainers kunnen een basistraining SAFe geven, maar als het gaat om Agile Software Delivery, Lean Portfolio Management (LPM) of DevOps, dunt het aanbod snel uit. De beschikbaarheid van alternatieven speelt dus een rol. Als er veel goedkopere en toegankelijke opties zijn om je band te laten repareren, is het moeilijk om een hoge prijs te rechtvaardigen. Dit geldt ook voor de monteur met die hoge factuur in de anekdote.

Perceptie van Waarde: Waarde is vaak *subjectief* en wordt beïnvloed door wat de klant denkt dat iets waard is. Als een klant de reparatie van een band als iets routineus en goedkoop beschouwt, zal die klant niet bereid zijn om een buitensporig bedrag te betalen. Dit werk is ook te simpel. En toch is de relatie tussen kosten en waarde in vooral het luxueuze segment soms volledig zoek. Denk aan handtasjes die voor X,- worden gemaakt en voor 50 x X,- worden verkocht. Dat komt omdat die bedrijven in staat zijn geweest hun product of dienst te verbinden aan een emotionele waarde, waarbij klanten niet alleen betalen voor de dienst, maar vooral ook voor status, merkervaring, of gemak.

Functionele versus Economische Waarde: Er zit in de perceptie van ‘Waarde’ een groot verschil tussen *Functionele* en *Economische* waarde. Mijn geplakte band is mij best iets waard, omdat ik op vakantie wil fietsen. Maar ik verdien er niks mee, en ik verlies ook geen geld als mijn band niet is geplakt. Met andere woorden, door het gemis van dit aspect, is de functionele waarde, financieel gezien, beperkt.

De Economische waarde verwijst naar de financiële impact van een oplossing of product. Het draait hier om hoeveel geld een product of dienst oplevert/bespaart voor de klant. In het geval van de Anekdote lag de hele fabriek stil. Stel die fabriek ligt al 6 uur stil met 100.000 euro omzetverlies per uur. Stel dat de monteur direct is gebeld en komt na 6 uur aan. En geeft die ene tik van 1 seconde. Dan zijn de Cost of Not Ownership, het niet hebben van een werkende fabriek, ook wel Cost of Delay genoemd, €600.000,- geweest. Ik geef je op een briefje dat niemand in die fabriek 1 vraag stelt over dat factuurtje van 10k (-:

Waarde in relatie tot SAFe’s Cost of Delay (“CoD”)

SAFe experts hoor je vaak over Cost of Delay. Dit is de economische waarde die verloren gaat door uitstel van een deliverable, op te leveren product of dienst. Het concept komt uit Lean-manufacturing en Agile-principes en is overgenomen in het SAFe Framework.

CoD maakt inzichtelijk hoeveel waarde je verliest door uitstel en dwingt Agile teams om na te denken over de impact van vertragingen. Dit helpt teams om gefundeerde beslissingen te nemen over welke user stories of features prioriteit moeten krijgen. Door de CoD te kwantificeren, kun je beter prioriteren en de focus leggen op werkitems met de grootste economische waarde. Dit bevordert de snelle levering van de meest waardevolle functionaliteit.

Denkexercitie: wat denk jij dat de CoD’s van de volgende LACE User Stories zijn?

1. **As** the LACE responsible for a successful SAFe implementation, **we want** all Agile teams to adhere to the formal SAFe vocabulary and naming conventions (“common language”) **so** that we maintain consistency and alignment across the organization and simplify communication with all our suppliers.

2. **As** the LACE, **we want** all ARTs to run a SAFe-compliant Planning Interval (PI) Planning event **so** that we can ensure alignment on a common goal, foster collaboration and make explicit what we promise to deliver the next 10 weeks.

3. **As** the LACE, **we want** to deliver a comprehensive SAFe implementation plan to the organization's executives, **so** that we can ensure a structured and well-organised Agile transformation across the organisation, driving successful adoption of SAFe practices and convincing the MT we are on the right track and have thought it through well.

4. **As** the LACE, **we want** to implement SAFe within our organisation, **in order** to (1) reduce time to market (2) increase productivity, (3) increase the built-in quality and (4) increase employee satisfaction.

16. Waarom het begrip ‘waarde’ voor velen zo lastig is (deel 2)

Inleiding

Elke professional gebruikt in zijn werk het woord *waarde*, maar bijna niemand definieert precies wat waarde is of wat ermee wordt bedoeld. In Scrum spreken we over waarde leveren voor de klant, in SAFe staat de Product Owner bekend als de ‘*value maximizer*’, en in MSP van het OGC draait alles om het Business B*enefits Realisation process.* Toch blijft het begrip ‘waarde’ in de praktijk verwarrend en vaak abstract.

In grote organisaties speelt dit nog sterker. Rollen veranderen, mindset verschuift, verantwoordelijkheden worden anders ingevuld, en teams worden aangespoord om “meer in waarde” te denken, maar hoe doe je dat als het niet duidelijk is wat waarde precies betekent? De impact op waarde? Wie de waarde levert of wie de waarde bepaalt?

Het doel van dit hoofdstuk is daarom helderheid brengen: **wat is waarde, wie bepaalt waarde, wie genereert waarde, en hoe gaan SAFe en MSP met waardebegrip om?** Daarbij leg ik expliciet de link naar alle SAFe lagen: welke waarde levert het portfolio feitelijk en voor wie? Welke waarde leveren de ART’s? En nog fundamenteler: *leveren ze überhaupt waarde, of leveren ze vooral producten of diensten zonder effect?*

Om deze vragen te beantwoorden, starten we dit hoofdstuk met een verkenning van het fundament: het begrip waarde zelf. Want voordat we kunnen praten over portfolio-flow, ART-doelen of strategische thema’s, moeten we eerst begrijpen **wie eigenlijk de waarde bepaalt**.

Dit hoofdstuk is opgebouwd in drie delen. Eerst verkennen we het begrip waarde zelf: fluïde, contextueel en altijd bepaald door de gebruiker. Daarna kijken we hoe SAFe en MSP dit vertalen naar concrete rollen en verantwoordelijkheden. Ten slotte behandelen we hoe je als organisatie meet of de beoogde waarde ook daadwerkelijk is gerealiseerd.

Wie bepaalt de waarde? Eén product: meerdere gebruikers: meerdere waarden

Waardetoekenning wordt **nooit** bepaald door een bouwer, leverancier, projectmanager, LPM of zelfs een ART die iets oplevert. Alleen de **gebruiker** bepaalt de voor hem/haar feitelijke waarde. Waarde ontstaat pas wanneer iets daadwerkelijk gebruikt wordt; zonder gebruik bestaat er geen waarde, alleen output, activiteit, product of dienst.

Een eenvoudig voorbeeld maakt dit zichtbaar. Mercedes levert de S-klasse, het topmodel uit hun collectie. De prijs sluit aan bij het comfort, de technologie en de kwaliteit die erin is verwerkt. Maar ondanks dat Mercedes één en hetzelfde product levert, verschilt de waarde per type gebruiker:

• Voor de ene koper vertegenwoordigt de S-klasse **statuswaarde**: een symbool van succes of maatschappelijke positie.

• Voor de ander draait het juist om **veiligheidswaarde**, omdat de S-klasse tot de veiligste auto’s ter wereld behoort.

• Een derde koper hecht vooral aan **comfortwaarde**: rust, stilte en ultiem rijcomfort tijdens lange ritten.

• Weer een ander koopt de auto vanwege **technologische waarde**, omdat hij elke nieuwe sensor, assistentiemodus of innovatie wil ervaren.

• Ten slotte kan de auto een sterke **emotionele waarde** hebben: het vervullen van een persoonlijke droom of herinnering.

Met andere woorden: één product kan meerdere waardeproposities vertegenwoordigen, gebaseerd op de kenmerken van het product (de samenstelling). Dit kan door verschillende typen gebruikers verschillend worden

gewaardeerd. Dus: Eén product: meerdere gebruikers: meerdere waarden.

De fabrikant levert mogelijkheden, en marketing kan potentiële waarden identificeren en benoemen, maar alleen de gebruiker **bepaalt welke van die waarden** daadwerkelijk betekenis hebben in zijn of haar gebruik en of marketing gelijk had. Dit heet ‘fit for purpose tegen de juiste acceptabele prijs’.

Dit principe maakt WSJF soms complex: Business Owners (BO) schatten de relatieve waarde in op een Fibonacci-reeks, maar doen dat vaak vanuit hun eigen **waarde-perspectief**. Terwijl gebruikers onderling verschillende waarden afnemen en dus een heel ander beeld kunnen hebben van wat werkelijk belangrijk is. Dus is het heel belangrijk te begrijpen hoe een BO de Waarde scoort in de WSJF analyse en bijvoorbeeld op basis van welke productkenmerken. (veiligheid, comfort, uitstraling, gemak, etc). Dit verschilt uiteraard per product of dienst.

Vanuit dit theoretische fundament starten we de paragrafen die nu volgen. We verkennen wat waarde werkelijk is, waarom waarde zo fluïde is, en welke vier elementen bepalen hoe waarde in een specifieke situatie wordt ervaren.

Waarde is fluïde en altijd contextueel bepaald

Waarde lijkt vaak een objectieve eigenschap, maar in werkelijkheid is waarde altijd fluïde en volledig afhankelijk van tijd, context, mens (zoals boven beschreven) en situatie. Wat vandaag hoog op de prioriteitenlijst staat kan morgen ineens irrelevant zijn. Daarom verandert waardebepaling zodra de omstandigheden veranderen.

Het eerste waarde-element is **Context & urgentie**. Urgentie kan waarde fors verhogen, zoals bij een systeem dat morgen nodig is voor een deadline of kritische taak. Als de tijdsdruk wegvalt, daalt de waarde net zo snel. Urgentie is daarmee een directe versterker of verzwakker van waardebeleving. Daarom rekent SAFe ook niet met ‘waarde’ maar rekent in de WSJF met ‘CoD’.

Het tweede waarde-element is **Alternatieven & unieke expertise**. Wanneer er schaarste is, bijvoorbeeld een team met unieke domeinkennis, stijgt de marktwaarde automatisch. Als meerdere alternatieven bestaan, daalt de waarde omdat vervangbaarheid groter wordt. Expertise vormt daarmee een waarde-multiplicator in complexe organisaties.

Het derde element is de **Perceptie van waarde**. Een product kan objectief eenvoudig of goedkoop te maken zijn maar toch een extreem hoge marktwaarde krijgen door hoe mensen het beleven. Een mooi voorbeeld is een Louis Vuitton-tas: de **kostprijs** van materialen en productie is relatief laag (geschat op €80,-), maar de **marktwaarde** is vele malen hoger (zelfde tas VKP a €4000,-). Dat verschil ontstaat niet door functionaliteit: vrijwel elke tas kan spullen dragen: maar door **status**, **emotionele verbondenheid**, **merkbeleving** en **slimme marketing**. Dat laat zien hoe sterk perceptie waarde stuurt: niet het ding zelf bepaalt de waarde, maar de **betekenis** die de gebruiker eraan geeft.

Het vierde element is **Functionele vs. economische waarde**. Functionele waarde draait om het directe nut en de bijdrage aan uitvoering, kwaliteit of beschikbaarheid. Economische waarde draait om besparing, efficiëntie, minder verstoring en lagere kosten. Beide waardesoorten kunnen elkaar versterken of juist compenseren.

Wanneer we deze vier elementen combineren, zien we dat waarde nooit absoluut is, maar voortdurend verschuift. Dit sluit aan bij de MSP-benadering waarin waarde niet ontstaat op het moment dat een project een output oplevert, maar pas later: wanneer de business het gebruikt, routines aanpast en outcomes worden gerealiseerd. MSP laat zien dat de echte waardestijging vaak pas in de *post-transition* fase zichtbaar wordt. Pas wanneer de nieuwe werkwijze wordt gestabiliseerd als normaal gedrag (*Sustained Business Operations*), kunnen mensen het nieuwe systeem optimaal benutten en worden benefits daadwerkelijk gerealiseerd. Daarom loopt de MSP benefit-curve pas maanden na oplevering omhoog: gebruikers hebben tijd nodig om de verandering te internaliseren, oude routines los te laten en de voordelen te ervaren.

Juist daarom moet waardebepaling regelmatig worden herijkt. Waarde ontstaat immers niet in de fabriek, niet in het project en niet in de ART, maar in het gebruik, en gebruik verandert over tijd. Alleen door deze dynamiek expliciet te erkennen, kunnen teams de juiste keuzes maken en middelen inzetten waar ze op dat moment de grootste impact bieden.

SAFe en Lean Thinking als ons theoretisch kompas

De stelling dat waarde niet ‘ingebakken’ zit in een product, maar pas ontstaat in de ogen van de gebruiker, is geen willekeurige mening. Het is fundamenteel verankerd in SAFe’s *Lean Thinking* van Womack & Jones. Het allereerste principe van SAFe’s Lean Thinking is immers: *“Precisely specify Value by product (by the end user).”*

In de traditionele ‘fabrieksmentaliteit’ werd vaak gedacht in termen van **intrinsieke waarde**: “Wij hebben hard gewerkt aan dit complexe systeem, het bevat dure materialen en geavanceerde technologie, *dus* het heeft waarde.” Lean breekt radicaal met die gedachte. Het maakt niet uit hoeveel budget, tijd of zweet er in een solution zit; als de gebruiker het niet kan toepassen om zijn probleem op te lossen, is de waarde nul. Een belangrijke nuancering geldt echter: in sommige contexten kan een capability waarde genereren zonder dat deze actief wordt gebruikt. De loutere beschikbaarheid en geloofwaardigheid van een oplossing kan strategische of preventieve waarde hebben, zelfs als de eindgebruiker er zelden of nooit feitelijk gebruik van maakt. In dat geval is de definitie van 'gebruiker' breder dan alleen de directe eindgebruiker: ook stakeholders die hun gedrag aanpassen op basis van de veronderstelde capability dragen bij aan de waarderealisatie. Dit is een nuancering op de gebruikersgerichte hoofdregel die het waard is expliciet te benoemen.

Sterker nog, Lean classificeert alles wat geen directe waarde toevoegt voor de gebruiker als **verspilling**. Stel dat een ART een communicatiesysteem bouwt met vijftig geavanceerde opties. Vanuit een technisch perspectief (output) is dit een ‘rijk’ product. Maar als de eindgebruiker in de praktijk onder hoge druk slechts drie van die opties gebruikt, zijn die andere 47 opties geen ‘extra waarde’. Ze zijn ballast. Ze maken het systeem complexer, de opleiding zwaarder en de kans op fouten groter. In Lean-termen hebben we dan geen waarde toegevoegd, maar *waste* gecreëerd.

Dit onderstreept waarom het onderscheid tussen de maker (ART) en de gebruiker (Business Owner) in SAFe zo cruciaal is. Een ART kan slechts **potentiële waarde** leveren (inventory). Net zoals voorraden in een magazijn pas geld opleveren als ze verkocht worden, levert onze SAFe fabriek pas waarde op als deze **geoperationaliseerd** wordt. Tot dat moment is het product slechts een investering: en in sommige gevallen zelfs een liability: maar nog geen waarde.

MSP als tweede lens: Output vs Outcome vs Business Benefits

Het MSP-raamwerk laat heel duidelijk zien dat waarde **niet** ontstaat op het moment dat een project als onderdeel van een programma een product oplevert, maar pas wanneer de organisatie dat product daadwerkelijk middels een organisatietransformatie gaat gebruiken. In onderstaand MSP diagram wordt dat onderscheid strak zichtbaar gemaakt. Onderaan, onder de stippellijn, bevindt zich het domein van projectmanagement (ART’s): hier worden systemen, processen, tools en andere vormen van capability opgeleverd. Dat zijn de outputs. Ze zijn technisch af en correct, maar ze vertegenwoordigen nog geen waarde. Ze zijn slechts de enabler voor de MSP Business Benefits Manager(s) om waarde te genereren.

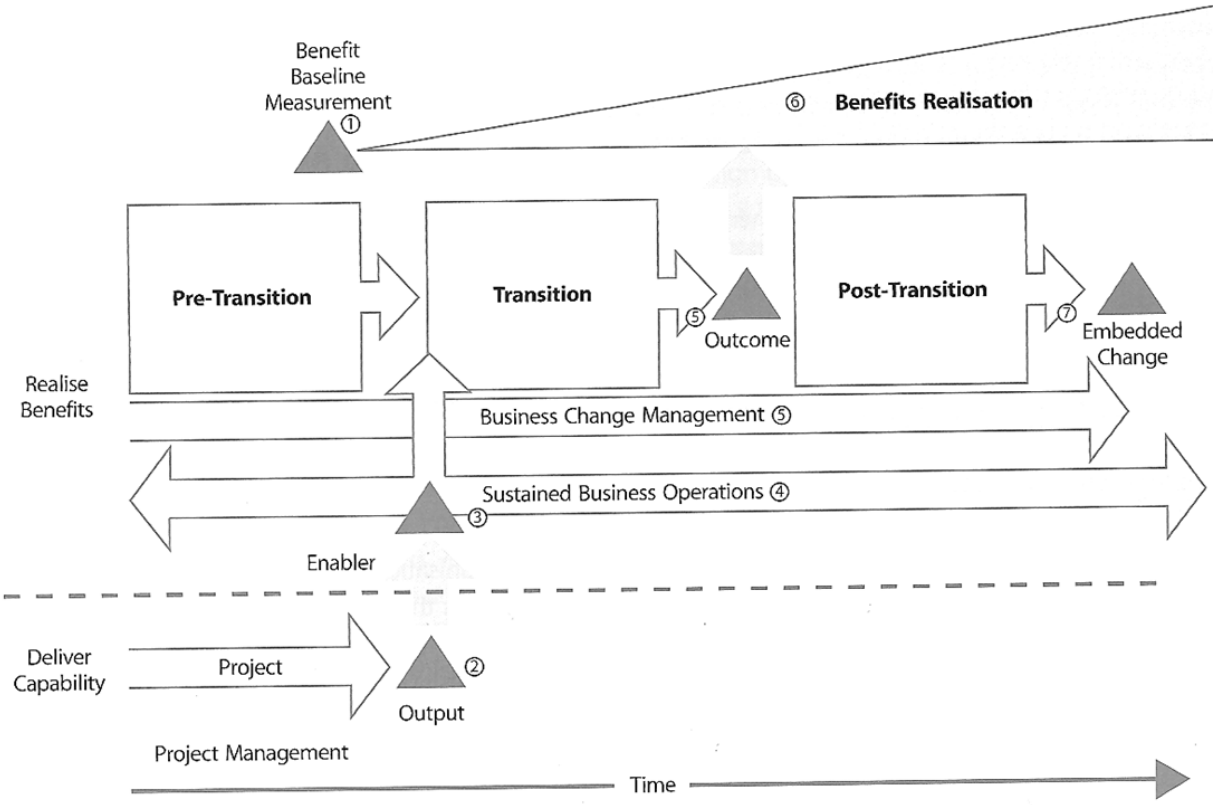


Figure 9.3 Simple example showing outputs, transition management and benefits realisation

Pas wanneer deze output wordt overgedragen aan de business, begint het echte werk: de overgang van output naar outcome. In de pre-transition en transition-fases organiseert de Business Change Manager ('BCM') de verandering die nodig is om het nieuwe product in de organisatie te laten landen. Processen moeten worden aangepast, mensen moeten worden opgeleid, gedrag moet veranderen, transitie moet worden begeleid en bestaande routines moeten worden doorbroken (met onder andere decommissioning). Dit is de fase waarin **benefits** ontstaan en ook groeien in de loop der tijd: de nieuwe manier van werken die het product daadwerkelijk bruikbaar maakt.

Daarna volgt de post-transition fase, waarin de organisatie de verandering vasthoudt in de dagelijkse operatie. MSP noemt dit *Sustained Business Operations*: het moment waarop de nieuwe werkwijze als normaal gedrag wordt ervaren. Pas in deze fase begint de echte waardestijging zichtbaar te worden. De curve van benefits realisation loopt vaak pas maanden na de oplevering omhoog, omdat gebruikers tijd nodig hebben om het nieuwe product volledig te benutten en de voordelen ervan te ervaren.

Helemaal aan het einde staat Embedded Change: het moment waarop de verandering volledig geïntegreerd is in de organisatie en de beoogde benefits structureel worden gerealiseerd. Dat is het punt waarop waarde niet alleen is ontstaan, maar ook duurzaam is geworden en bijvoorbeeld de VMO het CDP-Measure-proces kan uitvoeren (wat natuurlijk ook tussentijds al kan en mag).

De essentie van de MSP-lens is dat er een duidelijke scheiding is tussen degene die iets maakt en degene die waarde realiseert. **De projectmanager levert de output. De Business Change Manager zorgt dat de organisatie zich aanpast**. Maar de lijnorganisatie zelf, de gebruikers die het product dagelijks gebruiken, realiseert de benefits.

MSP bevestigt daarmee exact dezelfde logica die ik eerder heb beschreven: waarde ontstaat nooit in onze SAFe fabriek, niet in onze projecten, niet in onze ART's, maar in het daadwerkelijke day-to-day gebruik. Output is pas waarde wanneer mensen ermee werken, hun gedrag aanpassen en de organisatie er daadwerkelijk voordeel van ondervindt. De MSP post-transitionfase is een waarheid die vaak vergeten wordt.

Het SAFe-equivalent: ART’s, gebruikers en LPM

ART’s

ART’s zijn in SAFe verantwoordelijk voor het bouwen, integreren, testen en opleveren van solutions: producten, diensten, applicaties, capabilities en systeemketens. Zij leveren wat in MSP-termen de outputs zijn van het programma. Deze solutions hebben intrinsieke waarde: ze zijn technisch werkend, veilig, getest, compliant en voldoen aan de afgesproken specificaties. Maar: technisch werkend betekent nog geen echte waarde.

Wanneer een Solution nog niet is ingebed in de processen, het gedrag, de uitvoering en besluitvorming van de lijnorganisatie, is die Solution nog niet volledig geoperationaliseerd. Net als bij MSP geldt: de Program Manager levert de producten (outputs), maar dat is nog geen outcome. ART’s leveren dus **Potentiële waarde**. Zonder onze ART’s is er niets om te gebruiken. Wij werken daarom in een soort estafette: Een ART levert het stokje (het product), maar de wedstrijd is pas gewonnen als de Business Owner ermee over de finish rent (het gebruikt).

LPM

Lean Portfolio Management (LPM) levert zelf geen producten noch diensten, maar creëert waarde voor de gebruikers, LPM business owners en executives door richting te geven aan de organisatie. LPM bepaalt welke strategische doelen prioriteit krijgen en zorgt dat investeringen worden gedaan waar de grootste strategische en operationele impact ligt. Daarmee voorkomt LPM dat middelen worden verspild aan werk dat geen bijdrage levert aan de doelstellingen. LPM bewaakt de Guardrails.

Daarnaast levert LPM economische waarde door budgetten slim te verdelen, WSJF toe te passen en initiatieven stop te zetten die onvoldoende resultaat opleveren. LPM verbetert ook de portfolio flow door blokkades te verwijderen, afhankelijkheden te verminderen en governance te vereenvoudigen, zodat ART’s sneller en voorspelbaarder kunnen leveren.

LPM levert **alignment-waarde** door strategie, portfolio, ART’s en teams op één lijn te brengen. Zo weten alle lagen van de organisatie precies waar ze aan werken en waarom.

Tot slot identificeert LPM in nauwe samenwerking met de business de **gewenste business outcomes in de vorm van OKR’s** in de Epic Hypothesis Statement-sectie ‘outcomes’. Zodat duidelijk is welke operationele waarde de Business Owner met een bepaalde Epic wil realiseren.

LPM bouwt of levert dus niet zelf **Waarde**, maar zorgt dat de organisatie wél de juiste waarde kan leveren en helpt met kennis en kunde om de Waardeproposities scherp te stellen.

Business Owners als gebruikersvertegenwoordigers

De Business Owners vanuit de lijnorganisatie vervullen in SAFe feitelijk dezelfde rol als de Business Change Manager (BCM) in MSP. Zij zijn de enigen die met de geleverde solutions waarde kunnen genereren door de MSP organisatietransformatie toe te passen. De Business Owners zijn daarmee de verbindende schakel tussen de solutions die ART’s opleveren, de outcomes die de lijnorganisatie moet behalen en de waarde die alleen in echt gebruik kan ontstaan.

Zie onder andere deze SAFe BO-video uitleg: https://share.vidyard.com/watch/KLTKaZux9MoWH7Nmk8h1n6

Want, pas wanneer de lijnorganisatie een Solution in gebruik neemt, toepast en benut, ontstaat er echte waarde. Net als in MSP neemt de lijnorganisatie het product over, maakt er een outcome van en realiseert de uiteindelijke waarde(s).

tip: Alleen BO’s van echte stream aligned ART’s kunnen de rol van de MSP BCM volledig vervullen. Want BO’s van complicated sub system teams leveren halffabricaten aan de stream aligned ART’s. Maw, in dat geval is het werk van een BO iets anders en met iets minder impact.

De cruciale feedback-loop: van aanname naar bewijs via ‘Measure’

Een fundamentele vraag blijft vaak onbeantwoord: Hoe meet OS #2, de SAFe fabriek, de daadwerkelijke waardegeneratie? Hoe valideren we of de beoogde Outcomes uit de *Epic Hypothesis Statement* (EHS) succesvol zijn gerealiseerd?

Binnen SAFe en het Target Operating Model (TOM) is ‘Release on Demand’ niet het eindstation, maar slechts een tussenstap in de *Continuous Delivery Pipeline* (CDP). Na de release volgt de belangrijkste stap om te bepalen of we werkelijk waarde hebben geleverd: de **Measure-fase**.

Meten is weten: De CDP is geen eenrichtingsverkeer voor code en hardware richting productie; het moet data terugsturen naar de ART’s. Dit proces, ondersteund door *Application Telemetry* en *Innovation Accounting*, levert de harde feiten. In een commerciële omgeving meet men clicks en conversie. We kijken breder naar *leading indicators*: Wordt de Solution daadwerkelijk gebruikt? Op welke functies wordt geklikt en welke worden genegeerd? Neemt de operationele bruikbaarheid toe of juist af door bugs? Hoeveel bugs waren er na Release on Demand?

Van waterfall naar Agile: de pivot-or-persevere-beslissing: Deze data vormt de cruciale input voor de *Check* en *Act* stappen van de PDCA-cyclus. Zonder deze feedbacklus is er geen sprake van Agile werken, maar van ‘Waterfall in sprints’. De feedback dwingt LPM en Product Management tot een expliciet besluit:

• **Pivot:** De EHS-hypothese is ontkracht. De gebruikers gebruiken de Solution niet of de verwachte waarde blijft uit. We stoppen direct met investeren in deze oplossingsrichting en passen de strategie aan. Dit voorkomt dat we belastinggeld blijven pompen in systemen die technisch werken (output), maar operationeel niet helpen (geen outcome).

• **Persevere:** De hypothese is bevestigd. De telemetry toont aan dat de waarde wordt gerealiseerd. We investeren verder (via LPM investment horizons) om de oplossing uit te breiden.

Als we accepteren dat waarde pas ontstaat bij gebruik, volgen daaruit vier onvermijdelijke conclusies:

1. **Gedeelde verantwoordelijkheid:** Het totale succes wordt bepaald door de kwaliteit van de ‘SAFe-fabriek’ (de ART) én de kwaliteit van de Business Owners (BO’s). De BO moet bij de lijnorganisatie de waarde ‘oogsten’ middels een actieve hoogkwalitatieve organisatietransformatie.

2. **Begin van het bewijs:** een ART release is nooit het einde van het werk, maar het begin van de bewijsvoering. Pas na de release weten we of de fabriek het juiste heeft gemaakt (fit for purpose)

3. **Het "Operatie geslaagd, patiënt overleden"-risico:** Dit medische adagium geldt hier onverminderd. De SAFe-fabriek kan technisch perfecte producten leveren (Operatie geslaagd), maar als de BO’s niet in staat zijn deze goed te implementeren en te borgen in de organisatie, blijft het effect uit (Patiënt overleden).

4. **Borging in de TOM:** er moet in onze WoW, de werkwijze, een expliciet business-benefits- measure-proces zijn dat deze metingen verricht en LPM voedt met de Pivot- of Persevere-beslissing. Dit proces is geïdentificeerd in het TOM en moet in de fabriek worden ingericht en nageleefd om waardevernietiging, of blind sturen, te voorkomen

Note: Het assignen van Business Value tijdens I&A meet de output en de betrouwbaarheid van het planningsproces door een subjectieve score (1-10) te assignen aan elk bereikt PI Objective. De BO controleert of de ART de belofte die tijdens de PI Planning is gemaakt, daadwerkelijk heeft waargemaakt. Het meet de outputkwaliteit van de ART.

Conclusie

Dit hoofdstuk toont aan dat het begrip ‘Waarde’ binnen SAFe en MSP veel complexer is dan het simpelweg opleveren van een product. We hebben vastgesteld dat waarde geen statisch bezit is dat ‘in’ een applicatie of voertuig zit. Waarde is fluïde, contextueel en bestaat uitsluitend in de ogen van de gebruiker. Zoals Lean ons leert: alles wat de gebruiker niet direct helpt in zijn werk, is geen waarde maar verspilling (waste), ongeacht hoe geavanceerd de technologie is.

Van potentie naar realisatie De kern van een moderne, waardegedreven werkwijze ligt in de erkenning dat onze ‘SAFe-fabriek’ (de ART’s) slechts **potentiële waarde** levert. De ART is de leverancier van bouwstenen, maar het is de Business Owner (de lijnorganisatie) die het doel moet raken. De MSP-lens leert ons dat deze ‘schotprecisie’ niet vanzelf komt; het vereist een actieve organisatietransformatie door de Business Owners om van output (het systeem) naar outcome (het effect) te komen. Zonder deze adoptie en gedragsverandering in de lijnorganisatie blijft zelfs de beste software een ‘dood product’. Het is gevaarlijk om dan te roepen "dat SAFe heeft gefaald"; het kan immers ook zijn dat de implementatie in de lijn is gefaald, wetende dat de BO’s zowel in de ART’s als in LPM vertegenwoordigd zijn, en daarmee dus ook een impliciet onderdeel van de SAFe fabriek.

Sturen met open ogen Misschien wel het belangrijkste inzicht is dat we niet langer blind kunnen varen op aannames. De introductie van de **CDP-Measure-fase** en **Innovation Accounting** principes in de Continuous Delivery Pipeline is geen optie, maar een noodzaak voor onze Fabriek. Alleen door harde data terug te halen uit de praktijk (middels application telemetry) kunnen we valideren of onze LPM EHS hypothesen klopten. Deze data stelt LPM in staat om de cruciale beslissing te nemen: *Pivot* (stoppen en bijsturen) of *Persevere* (doorgaan en opschalen). Zonder deze feedbackloop is Agile slechts een cosmetische laag over een oude waterval-werkwijze en blijven we budget investeren zonder garantie op daadwerkelijk effect.

Gezamenlijk leveren Uiteindelijk dwingt dit waardebegrip ons tot een nieuwe vorm van samenwerking tussen de teams en overige stakeholders. Het succes van de digitale transformatie wordt niet bepaald door wie het hardst werkt in de fabriek, maar door de synergie tussen de makers (ART) en de gebruikers (via alle BO’s). Waarde ontstaat nooit in het project, nooit in de ART en nooit in de directiekamer. Waarde ontstaat alleen daar waar het werk wordt uitgevoerd.

Het is aan ons om het systeem zo in te richten dat we die waarde niet alleen beloven, maar ook daadwerkelijk meten, oogsten en borgen!

17. Economische prioritering in SAFe: waarde, Cost of Delay en WSJF, oefen zelf met prioriteren

Na het lezen van dit hoofdstuk weet je hoe je werk economisch prioriteert met Business Value, Cost of Delay en WSJF, en heb je dat zelf geoefend aan de hand van echte LACE deliverables. Dit hoofdstuk volgt drie stappen: (1) schat Business Waarde in op gevoel, (2) verfijn je inschatting met inzicht in Cost of Delay, en (3) bereken de WSJF-score voor een volledige economische prioritering.

Gefeliciteerd! Je bent nu bevorderd tot Generaal bij Defensie en je mag beslissen wat het LACE de komende PI gaat opleveren en waaraan het gaat werken (=Planning Interval) voor de komende 10 weken. We kennen ‘gebruikers/klanten’ allemaal vrij goed: die willen alles, nu en bij voorkeur gratis. Herkenbaar? Helaas werkt dat niet zo, want elke organisatie staat voor de uitdaging om keuzes te maken door een overvloed aan vraag en een beperkt aanbod. Net zoals bij jouw organisatie wellicht. Daarom moeten we slim beslissen: wat nu, wat straks, wat later, en wat nooit.

In dit hoofdstuk gaan we samen oefenen, spelen, en wat ‘waarde inschattings- en prioriteringsgevoel’ opbouwen. Hoe gaat je (interne) IT supplier haar afnemers/gebruikers/BO’s helpen al het werk slim te prioriteren? En vooral: waarom en met welke onderbouwing?

Rest mij je veel leesplezier toe te wensen. Ik hoop dat ik je wat waarde- & prioriteringskennis en inzichten heb weten over te brengen!

Oefening #1

Je krijgt in je nieuwe functie de taak om prioriteiten te stellen voor een aantal (*echte*) LACE deliverables. Dit doe je door de ‘Business waarde (BW)’ voor Defensie toe te kennen, met een cijfer tussen de 1 en 10. Je mag niet alles een 10 of een 1 geven. En je moet de scores *relatief inschatten en evenwichtig verdelen.*

Tip: de eerste rij is ingevuld als voorbeeld, inclusief een toelichting op de redenering. Vul de overige rijen zelf in op dezelfde manier.

Deliverables LACE	Business waarde
Een SAFe Project Voorstel (PV)	7 (hoog): een PV is nodig om budget en mandaat te krijgen; zonder PV start niets.
Een maturity scan op huidige Agile Teams binnen Defensie	[score]
Het nieuwe Target Operating model met alle nieuwe ART’s en Agile teams	[score]
Een volledig operationeel recurring PI event, conform SAFe	[score]
Een geïmplementeerde Agile Cadence	[score]
De Why, klip en klaar, gepubliceerd in onze nieuwsbrief	[score]
Publicatie van een SAFe-artikel in de nieuwsbrief	[score]
Een SAFe opleiding voor de LACE members	[score]
Aantrekken van een Senior Agile Coach	[score]
Een teambuildingsessie voor alle LACE members	[score]

Dank voor je input! We hebben nu inzicht in jouw prioritering voor de komende PI. Maar de vraag is: op basis waarvan heb je net de waarde bepaald? Waarom krijgt de ene deliverable 10 punten en de andere slechts 1? Wat is waarde precies en hoe estimate je die? Welke elementen van de bovenstaande deliverables bepaalden voor jou of iets veel of weinig waarde had? Was bijvoorbeeld de kostprijs of doorlooptijd van een deliverable van invloed op je beslissing?

Wat mag waarde kosten?

Voor executives, kolonels, business owners, key users en uiteraard ook de Agilisten is het belangrijk om goed na te denken over het begrip ‘waarde’. Wat is waarde precies? En hoeveel mag waarde kosten? Hoeveel levert een bepaalde deliverable voor jou op of hoeveel euro’s verlies je per dag zolang je de deliverable niet hebt?

In SAFe draait alles om waarde, in het Engels ‘Value’ genoemd. Echte SAFe-experts denken alleen maar in termen van waarde en waardestromen (Value Streams). Niet voor niets wordt de rol van Product Owner ook wel ‘*Value Maximizer’* genoemd. Om te benadrukken wat die rol als allerbelangrijkste taak behelst. *Echte* SAFe-experts zijn daarom nooit dogmatisch en volgen ook niet per se strikt de officiële richtlijnen, rollen, processen en terminologie van het framework, zonder aanpassingen te maken voor de lokale omstandigheden om die waarde te maximaliseren. Echte SAFe-experts doen ook niets dat *geen waarde oplevert* of dat niet leidt tot (meer) tevreden gebruikers. Echte SAFe experts zijn verslaafd aan het tiende principe van het Agile Manifesto, wat luidt: *“Simplicity, the art of maximizing the amount of work not done, is essential.”*

SAFe is hun slimme grote gereedschapskist, maar hun primaire doel is altijd om de *Soll Metrics* van hun opdrachtgever en de gebruikers te realiseren. SAFe is het middel, geen doel.

Waarde is niet absoluut, maar contextueel bepaald

Laten we in dit hoofdstuk even samen dieper inzoomen op dat begrip waarde. Waarde is namelijk geen vaststaand gegeven; het wordt *contextueel bepaald, wat een en ander complexer maakt.* Het kan goed zijn, dat als wij morgen alle waardes in de oefening weer opnieuw bepalen, er een andere volgorde uitkomt.

Waarde op een bepaald moment wordt beïnvloed door de volgende 4 waarde-elementen:

1. Context en Urgentie: Stel je voor dat je fietsband lek is vlak voordat je op vakantie gaat en er geen andere opties of alternatieven zijn. In dat geval is de waarde van de dienst van de fietsenmaker hoog. De fietsenmaker zou zijn prijs zelfs kunnen verdubbelen, veel mensen zouden dat accepteren. Met andere woorden, urgentie heeft impact op de waarde beoordeling.

1. Alternatieven & unieke expertise: De prijs die een leverancier vraagt of kan vragen, wordt mede bepaald door schaarste en de unieke kennis die vereist is. Er zijn veel trainers die een basistraining SAFe kunnen geven, maar bij complexe onderwerpen zoals Agile Software Delivery, Lean Portfolio Management (LPM) of DevSecOps, neemt de beschikbaarheid van trainers snel af, simpelweg omdat de complexiteit en de moeite voor de docent toeneemt. Als er dus veel goedkopere alternatieven zijn, is het moeilijker om een hoge prijs te rechtvaardigen, de klant gaat dan ‘shoppen’

1. Perceptie van waarde: Waarde is vaak *subjectief* en hangt af van wat de klant denkt dat iets waard is. Er zijn drie belangrijke varianten van deze perceptie:

1. **Routine en eenvoud:** Wanneer een klant iets als eenvoudig of routineus beschouwt, zoals de reparatie van een fietsband, is de bereidheid om veel te betalen vaak laag. Het werk lijkt te simpel om een hoge prijs te rechtvaardigen. Maar in andere gevallen, zoals in het luxesegment, kan de relatie tussen kosten en waarde volledig zoek zijn. Denk aan een handtas die voor X,- wordt geproduceerd en vervolgens voor 50 x X,- wordt verkocht.

1. **Emotionele waarde:** Sommige bedrijven slagen erin om hun product of dienst te koppelen aan een emotionele waarde. Klanten betalen dan niet alleen voor het product, maar ook voor de status, merkervaring of het gemak dat ermee gepaard gaat. Deze emotionele verbinding kan een veel hogere prijs rechtvaardigen, zelfs als de productiekosten relatief laag zijn.

1. **Onbegrip van waarde:** Soms begrijpen management en gebruikers de werkelijke waarde van een deliverable niet volledig, omdat het niet duidelijk is wat het oplevert of welke voordelen eraan verbonden zijn. Dit kan leiden tot een onjuiste inschatting van de waarde. Denk bijvoorbeeld aan het PI Event: welke waarde heb jij hier als Generaal in de oefening aan toegekend? Ik ben benieuwd hoe jij het PI Event percipieert, want dat vertelt veel over jouw waarde beleving van deze SAFe ceremonie. Nog sterker, iemand van ons in een van de SAFe sessies

zei de PI ‘zonde van de tijd’ te vinden, want dan ‘werd er niet gewerkt’.

1. Functionele versus Economische waarde: Functionele waarde verwijst naar het directe nut van een product of dienst, zoals het kunnen fietsen na een reparatie. Je band is weer geplakt, de functie ‘fiets’ doet het weer. Economische waarde heeft veel meer te maken met de financiële impact: hoeveel bespaart of verdient iemand met een bepaald product of dienst? Als ik morgen mijn fiets niet nodig heb voor vakantie maar om naar mijn werk te gaan, wordt de functionele waarde plots economisch. Want mogelijk moet ik kosten gaan maken voor de aanschaf van brandstof en verlies ik geld.

Oefening #2

Nu je meer weet over waarde, verandert dit je score van de vorige oefening? Welke deliverable is voor jou het meest urgent? Welke deliverable zou economisch verlies veroorzaken als die niet op tijd geleverd wordt? Welke deliverable heeft functionele waarde? Welke deliverable heeft de hoogste emotionele waarde?

Denken in Cost of Delay en doorlooptijden

Heb je ooit gehoord van de Lean-term ‘Cost of Delay' (CoD)? Dit verwijst naar de economische waarde die verloren gaat door het uitstellen van een deliverable, product of dienst. Oftewel 'cost of not ownership’. Dit concept komt uit Lean-manufacturing en Agile-principes en is een belangrijk instrument uit het SAFe-framework. Het helpt inzicht te geven in de impact van uitstel.

Cost of Delay laat zien hoeveel waarde verloren gaat door vertraging door je leverancier (in ons voorbeeld de LACE als transformatieorgaan) en dwingt je na te denken over de gevolgen daarvan voor Defensie, jouw gebruikers en jouw stakeholders.

Het berekenen van de CoD helpt de business om gefundeerde beslissingen te nemen over welke PI-features de hoogste prioriteit moeten krijgen tijdens het PI Event. Door de CoD te berekenen, kun je de prioriteiten scherper stellen en focussen op deliverables die de grootste economische waarde opleveren. Dit bevordert een snelle levering van de meest *waardevolle* functionaliteiten.

Voor een volledige economische prioritering missen we echter nog één belangrijke factor: de doorlooptijd, oftewel 'Lead Time/job size/job duration'. Dit wordt berekend door de verwerkingstijden en alle wachttijden bij elkaar op te tellen. Hoe lang duurt het voordat de LACE de deliverable van begin tot eind heeft ontwikkeld? Deze inschatting kun je niet zelf maken als Opdrachtgever; dat moeten de LACE-leden zelf doen, door het werk in te schatten en jou te informeren. In SAFe wordt werk altijd ingeschat door het team dat het werk uitvoert en niet door de RTE, de PM, of iemand anders. We zijn er nu bijna: we hebben de Cost of Delay en we hebben een Job Size score.

Oefening #3: bepaal nu jouw economische prioritering

Vul nu onderstaande tabel aan met de Job Size score die de LACE members je hebben gegeven. Deel de Cost of Delay vervolgens door de Job Size. Dat getal heet WSJF oftewel de *Weighted Shortest Job First* Score.

Deliverables LACE	**Cost of Delay (Business waarde + urgentie + Risk Reductie/Opportunity Enablement)**	**Job Size (maat voor doorlooptijd)**	**de WSJF Score**
Een SAFe Project Voorstel (PV)	[score]	[score]	CoD/JS=
Een maturity scan op huidige Agile Teams binnen Defensie	[score]	[score]	CoD/JS=
Het nieuwe Target Operating model met alle nieuwe ART’s en Agile teams	[score]	[score]	CoD/JS=

Een operationeel recurring volwassen PI event	[score]	[score]	CoD/JS=
Een geïmplementeerde Agile Cadence	[score]	[score]	CoD/JS=
De Why, klip en klaar, gepubliceerd in onze nieuwsbrief	[score]	[score]	CoD/JS=
Publicatie van een SAFe-artikel in de nieuwsbrief	[score]	[score]	CoD/JS=
Een SAFe opleiding voor de LACE members	[score]	[score]	CoD/JS=
Aantrekken van een Senior Agile Coach	[score]	[score]	CoD/JS=
Een teambuildingsessie voor alle LACE members	[score]	[score]	CoD/JS=

Pas op: Job Size bij Defensie-epics omvat ALLE kosten, ook hardware

Bij Defensie worden voor epics soms enorme bedragen aan militaire hardware ingekocht (denk aan tanks, voertuigen of wapensystemen ter waarde van honderden miljoenen euro's), waarbij de arbeidskosten slechts een klein deel van het totaal uitmaken.

De vraag die dan opkomt: neem je voor de WSJF job size schatting alleen de arbeidskosten mee, of ook de hardwarekosten? SAFe is daarin vrij onduidelijk

Antwoord: alle kosten meenemen, inclusief hardware. (ref: Supportdesk SAFe, 22 okt 2025)

Je prioriteert immers significante initiatieven op portfolioniveau. Als je alleen naar arbeidskosten kijkt, krijg je een vertekend beeld van de werkelijke investering en neem je economisch gezien verkeerde prioriteringsbeslissingen. Een epic waarbij voor €400 miljoen aan tanks wordt ingekocht maar slechts 500 manuur aan arbeid zit, krijgt daardoor een veel grotere Job Size, en dus een lagere WSJF-score, dan zijn arbeidscomponent doet vermoeden.

Wat heb je na het lezen geleerd?

Als je oefening 1 vergelijkt met oefening 3, zie je een verschil: in oefening 1 prioriteerde je puur op gevoel, terwijl oefening 3 je dwingt om economisch te onderbouwen. Dat is precies de kern van dit hoofdstuk. Waarde in SAFe is contextueel: het wordt bepaald door urgentie, alternatieven, perceptie en functie. Cost of Delay laat zien wat het kost als je een deliverable uitstelt. En WSJF combineert die twee: de deliverable met de hoogste CoD gedeeld door de kleinste Job Size gaat als eerste. Door deze drie stappen te doorlopen (gevoel, verdieping en berekening) bouw je het economische prioriteringsgevoel dat Business Owners, kolonels en agilisten nodig hebben om elke PI de juiste keuzes te maken.

18. Van dependency management naar dependency orchestration in SAFe

1.1. Introductie

Vanuit mijn ervaringen bij Defensie en met name het grote en complexe ASML kijk ik bij dependencies vooral naar de samenwerking tussen ART's en interne en externe leveranciers. Wie heeft wat van wie nodig, en wanneer? Die vragen lijken eenvoudig. Toch blijkt een afhankelijkheid op een planningboard vaak duidelijker dan de afspraak erachter.

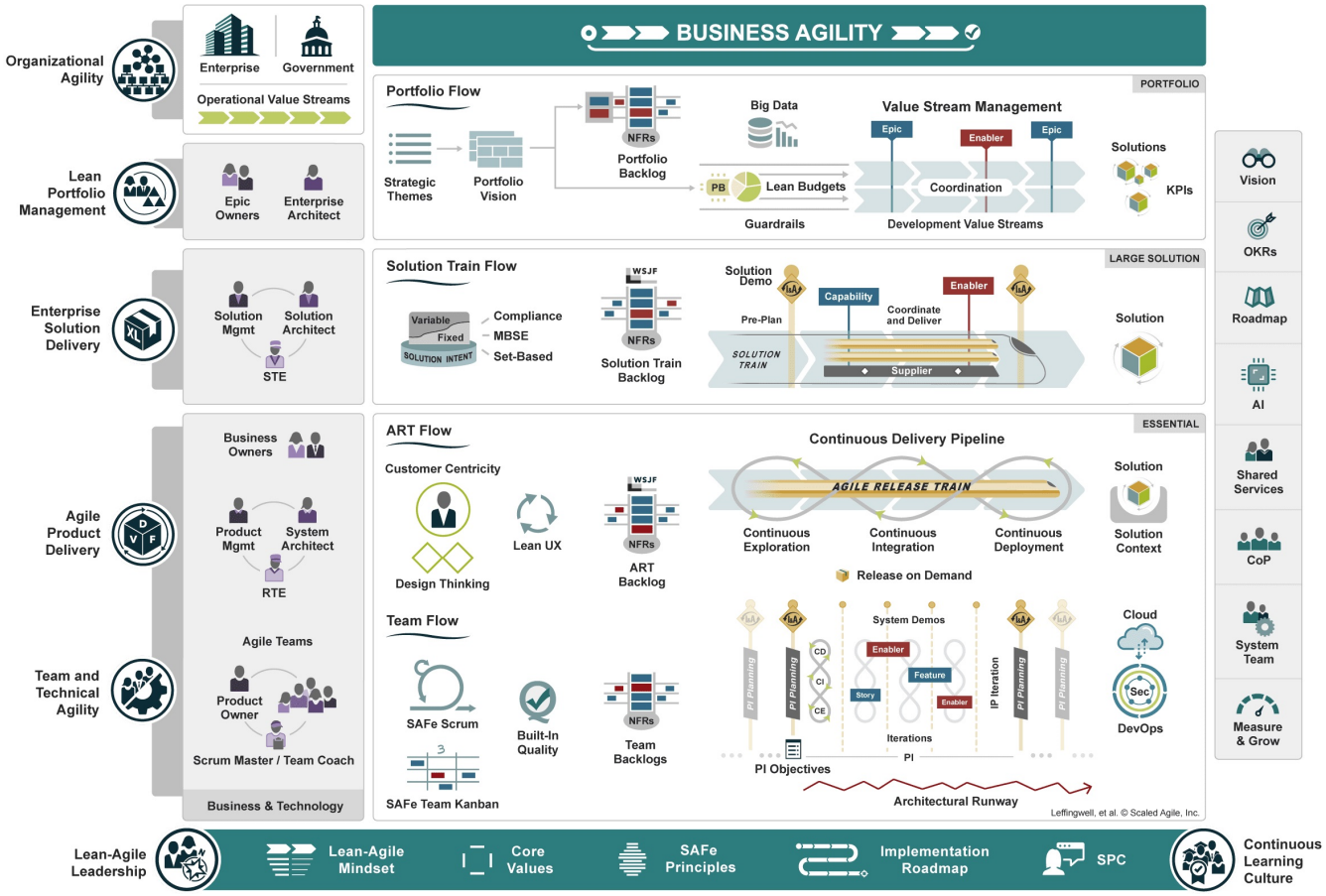
Een agile team als onderdeel van een ART heeft op een bepaald moment een koppeling, systeem, besluit of testomgeving nodig. Er komt een geeltje op het ART Planning Board en iemand trekt een rode lijn conform de SAFe-opleiding. Een paar iteraties later blijkt dat de andere partij een andere datum verwachtte, of het werk helemaal niet had ingepland. Het agile team rapporteert in de Iteration Review als resultaat van een bepaalde feature 'not completed'.

Daar zit voor mij de kern. Dependencies vastleggen kan bijna iedereen. Vervolgens moet je samen met alle betrokkenen zorgen dat de benodigde bijdragen op elkaar aansluiten. Dat samenspel noem ik dependency orchestration. In dit hoofdstuk ga ik je met concrete tips en adviezen uitleggen hoe dat in de praktijk werkt.

1.2. Wat is een dependency?

Een dependency is een afhankelijkheid op een van de niveaus van SAFe, bijvoorbeeld tussen Epics op portfolioniveau, Features op ART-niveau of user stories van agile teams. De afhankelijkheid kan intern zijn (binnen de ART) of extern (buiten de ART of zelfs buiten de organisatie) en beïnvloedt de levering van werk en waarde. Technische afhankelijkheden gaan bijvoorbeeld over interfaces. Functionele afhankelijkheden ontstaan wanneer onderdelen samen nodig zijn voor een gebruikersproces. Organisatorische afhankelijkheden betreffen onder meer besluiten, specialistische capaciteit of toegang tot een omgeving.

Die afhankelijkheden spelen op verschillende SAFe-niveaus. Binnen een ART raken ze teams en hun Stories of Features. Tussen Solution Trains kunnen Capabilities en gezamenlijke integratiemomenten meespelen. Op portfolioniveau raken Epics en investeringsbeslissingen elkaar. Maak daarom zichtbaar welk werk op welk niveau van een andere bijdrage afhankelijk is.



Figuur 1. De samenhang tussen teams, ART's, Large Solution en Portfolio.

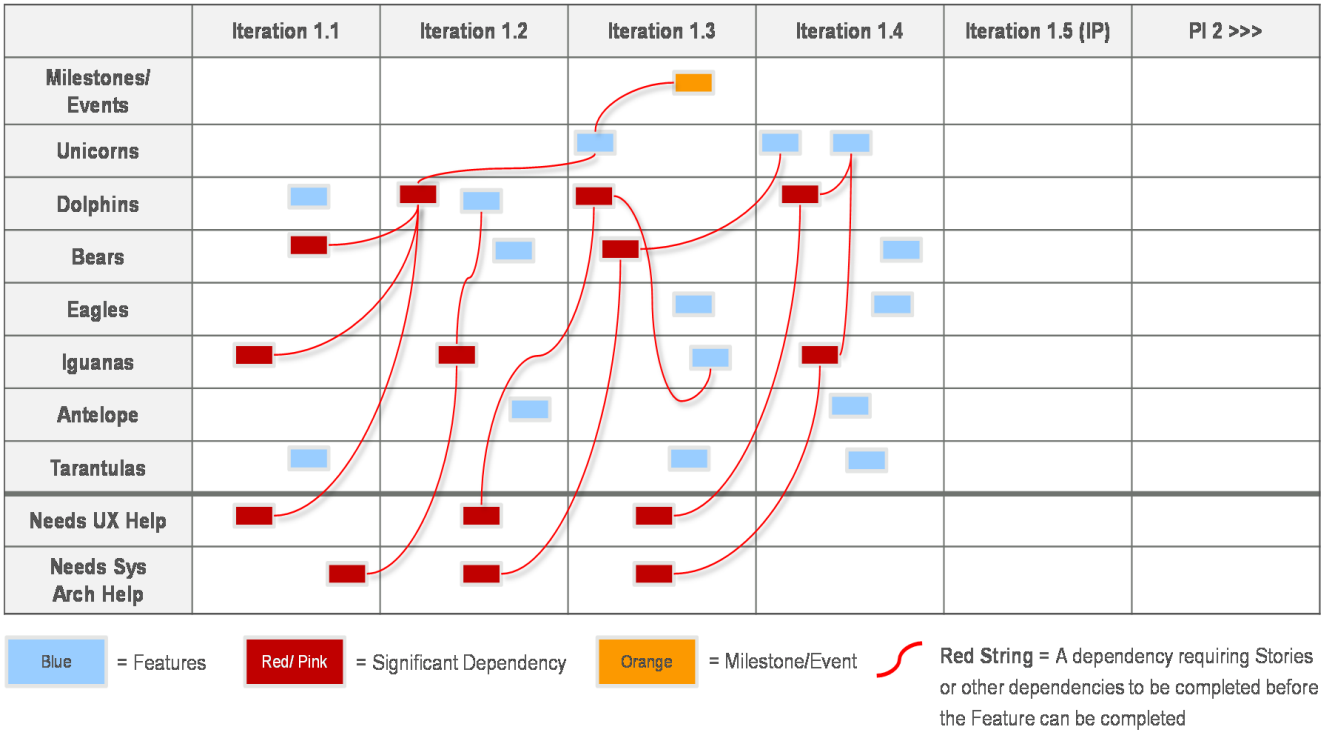
1.3. PI Planning omvat meer dan dependencies verzamelen

Ik kijk bij PI Planning altijd verder dan de lijst met afhankelijkheden. Kunnen de teams het afgesproken werk ook uitvoeren? Past de gevraagde bijdrage in de capaciteit van de leverancier van de benodigde bijdrage? En weten beide partijen wat ze van elkaar verwachten? Is de dependency wederzijds *refined*?

PI Planning brengt alle agile teams en belanghebbenden samen rond een gedeelde missie, visie en doelen voor het komende *interval*. De agile teams maken hun capaciteit en belasting per iteratie zichtbaar, werken Stories uit en formuleren PI Objectives. Ze bespreken risico's en stemmen hun plannen af. Business Owners kennen businesswaarde toe aan de PI Objectives.

Het ART Planning Board toont de verwachte Feature-oplevering, mijlpalen en met name ook de afhankelijkheden. Met geschikte, professionele agile tooling zijn die afhankelijkheden goed zichtbaar. Samen met de teamplannen maakt het duidelijk of het geheel uitvoerbaar is. In het onderstaande voorbeeld staan vier ontwikkeliteraties en een vijfde Innovation and Planning (IP) Iteration. Die IP Iteration is primair bedoeld voor Innovation & Planning.

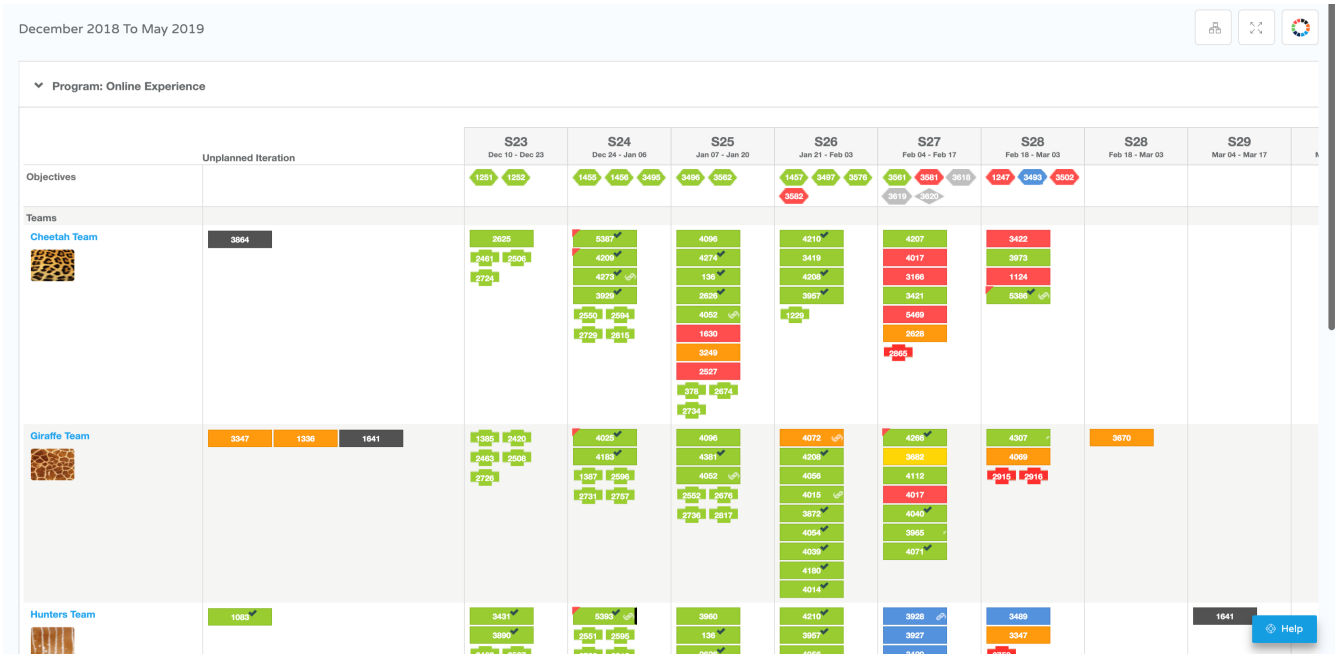
1.4. Het ART Planning Board als hoofdgespreksonderwerp



Figuur 2. Het ART Planning Board met Features, afhankelijkheden en mijlpalen per iteratie.

De teams staan onder elkaar, de iteraties naast elkaar. Blauw staat voor Features, rood voor belangrijke afhankelijkheden en oranje voor mijlpalen of events. De rode lijnen verbinden benodigde bijdragen met het afhankelijke werk. Ook ondersteuning vanuit UX en System Architecture is zichtbaar.

Voor mij als Agile Coach begint het nuttige gesprek waar plannen botsen. Een team heeft iets in iteratie 1.2 nodig, terwijl de leverancier pas in 1.4 ruimte heeft. Dan moet je de volgorde, scope, oplossing of prioriteit aanpassen. De rode lijn helpt het probleem te zien, maar lost het niet op.



Figuur 3. Het ART Planning Board van Jira Align.

Een professioneel digitaal board in de juiste agile tooling helpt om deze relaties tijdens het PI bij te houden. Kleuren en symbolen hangen af van de inrichting van de software. De betrokken teams moeten blijven

controleren of hun afspraken en plannen aansluiten.

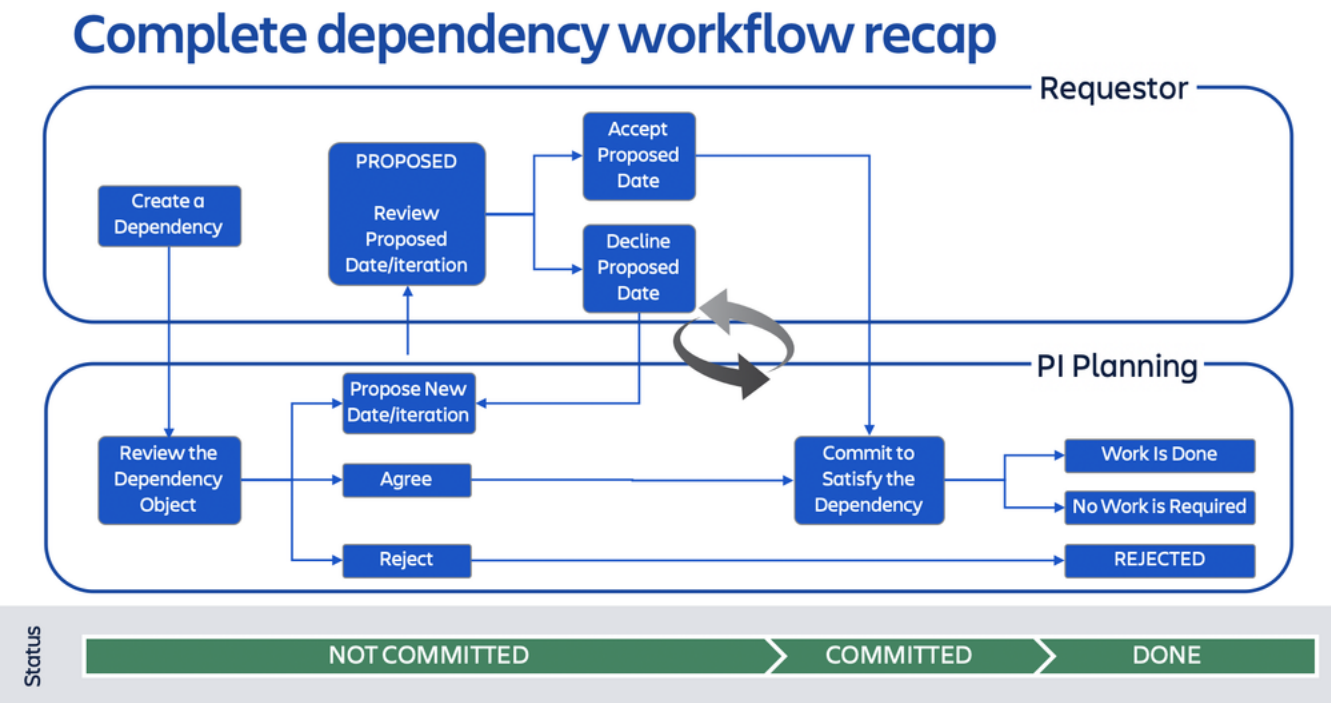
1.5. Een dependency moet concreet genoeg zijn om op te plannen

Met ‘wij zijn afhankelijk van team B’ *kan de ander weinig*. Welk resultaat is nodig, voor welk werkitem en op welk moment? Wanneer is de bijdrage bruikbaar? Ik wil dat beide partijen daar hetzelfde antwoord op geven door de dependency bij voorkeur tijdens PI Planning goed te verfijnen.

De Scrum Master/Team Coach kan dit gesprek organiseren, direct tijdens PI Planning of tijdens de uitvoering van een iteratie (afhankelijk van de materiekennis van de Scrum Master). Voor de echte inhoud zijn daarom vaak de Product Owner, Product Manager, System Architect of een Subject Matter Expert (SME) nodig. Vooral de beschrijving en acceptatiecriteria vragen hun aandacht. Maak die criteria SMART: specifiek, meetbaar, acceptabel, realistisch en tijdgebonden. Bespreek ook hoe je de bijdrage samen gaat toetsen.

Onderdeel	Wat leg je vast?
Niveau, ID en relatie	Het niveau, een uniek dependency-ID en de gekoppelde werkitems, inclusief de richting van de afhankelijkheid.
Titel en beschrijving	Een herkenbare titel en een duidelijke omschrijving van de benodigde bijdrage.
Acceptatiecriteria	Toetsbare voorwaarden waarmee beide partijen vaststellen of de bijdrage bruikbaar is.
Betrokken partijen	Wie aanvraagt, waarvoor de bijdrage nodig is en wie haar beoordeelt en levert.
Needed by en leverafspraak	Het benodigde PI en de iteratie, plus de bevestigde oplevering. Houd wens en toezegging uit elkaar.
Status en blokkade	Niet toegezegd, toegezegd of afgerond. Registreer apart of het afhankelijke werk daadwerkelijk geblokkeerd is.
Gesprekshistorie	Afspraken, wijzigingen en besluiten, met hun aanleiding.

1.6. Van aanvraag naar commitment (of uncommitted)



Figuur 4. De dependency-workflow in Jira Align: aanvraag, afstemming, commitment en afronding.

De aanvrager beschrijft de behoefte. De ontvangende partij beoordeelt het verzoek en gaat akkoord, stelt een andere datum voor of wijst het af. Een gewijzigd voorstel gaat terug naar de aanvrager, die moet beoordelen of het nog bij zijn planning past.

Pas na overeenstemming ontstaat een *commitment*. Goede agile software legt dit ook digitaal vast. Ik wil dat vervolgens terugzien in het werk en de capaciteit van de leverende partij. Anders is de toezegging weinig waard.

Het schema gebruikt ‘Not committed’, ‘Committed’ en ‘Done’, met daarnaast uitkomsten zoals ‘Rejected’ en ‘No work is required’. Een afwijzing betekent dat de behoefte nog openstaat. Als geen werk nodig is, leg je vast waarom de behoefte al is vervuld of de afhankelijkheid vervalt.

Bij afronding controleren beide partijen de acceptatiecriteria. Kan de aanvrager werkelijk verder? Verandert onderweg de inhoud of datum, dan stemmen ze de gevolgen opnieuw af.

1.7. Afstemming tijdens het PI

Hier krijgt *dependency orchestration* betekenis. Levert team B de interface later, dan verschuift mogelijk ook het integratiewerk van team A. Een gezamenlijke test of PI Objective kan daardoor in gevaar komen. Je moet dus verder kijken dan de afzonderlijke afspraak.

Teams bespreken zulke gevolgen zodra ze zichtbaar worden en escaleren als een belangrijke mijlpaal in gevaar dreigt te komen. Product Owners en Product Management helpen bij scope en prioriteit. Architecten onderzoeken alternatieven. De RTE faciliteert afstemming en helpt impediments aan te pakken. Binnen een Solution Train ondersteunt de STE de afstemming tussen ART’s.

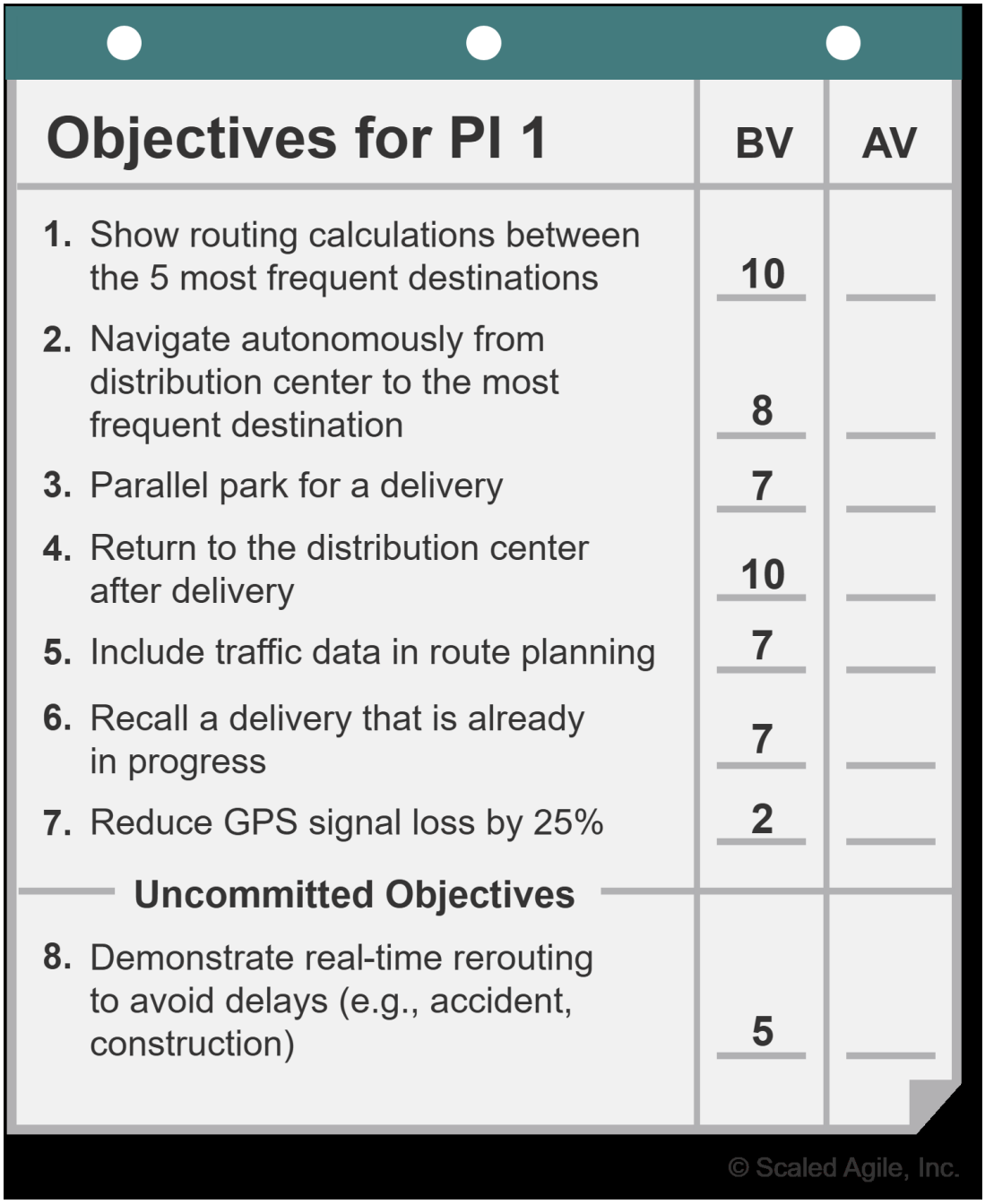
Gebruik daarvoor bestaande synchronisatiemomenten, zoals de ART Sync, en betrek de LACE bij organisatorische problemen. Zorg daarnaast dat teams elkaar rechtstreeks bereiken. Bij een actuele blokkade wil ik niet wachten op het volgende overleg.

Onderzoek ook of de afhankelijkheid kan worden verkleind door het werk op te splitsen (slicing). Misschien volstaat eerst een deel van de interface of kunnen teams eerder integreren. Terugkerende wachttijd is voor mij aanleiding om naar de werkverdeling en overdrachten te kijken.

1.8. Committed versus uncommitted PI objectives

Tijdens het PI Planning-event worden de initiële dependencies bepaald. Gedurende de iteraties kunnen er nog dependencies bijkomen of kunnen er nieuwe inzichten ontstaan.

In SAFe heb je twee soorten team PI objectives: committed en uncommitted. Vooral dependencies die niet in control zijn, leiden tot uncommitted objectives. Denk aan externe leveranciers die onbetrouwbaar leveren, interne suppliers, of ART’s die qua SAFe minder volwassen zijn. Met andere woorden: uncommitted objectives hangen sterk samen met slecht dependency management.



Het doel in termen van volwassenheid is dat ART's ervoor zorgen dat zoveel mogelijk uncommitted PI objectives committed worden. Dat kan bijvoorbeeld door concrete leverafspraken te maken en door externe suppliers boetes op te leggen bij te late levering.

1.9. Centrale kaders, decentrale uitvoering

Mijn advies voor een omgeving zoals Defensie en ASML begint met één overzicht van ART's, teams en raakvlakken. Wie levert wat en hoe bereik je die partij? Dat uitzoekwerk hoort niet pas tijdens een breakout te beginnen.

Een LACE kan SAFe-expertise en gezamenlijke werkwijzen coördineren. Denk aan begrippen, registratievelden, tooling en escalatiepaden. De teams en ART's beheren hun eigen dependencies. Zij kennen het werk en moeten de afspraken waarmaken.

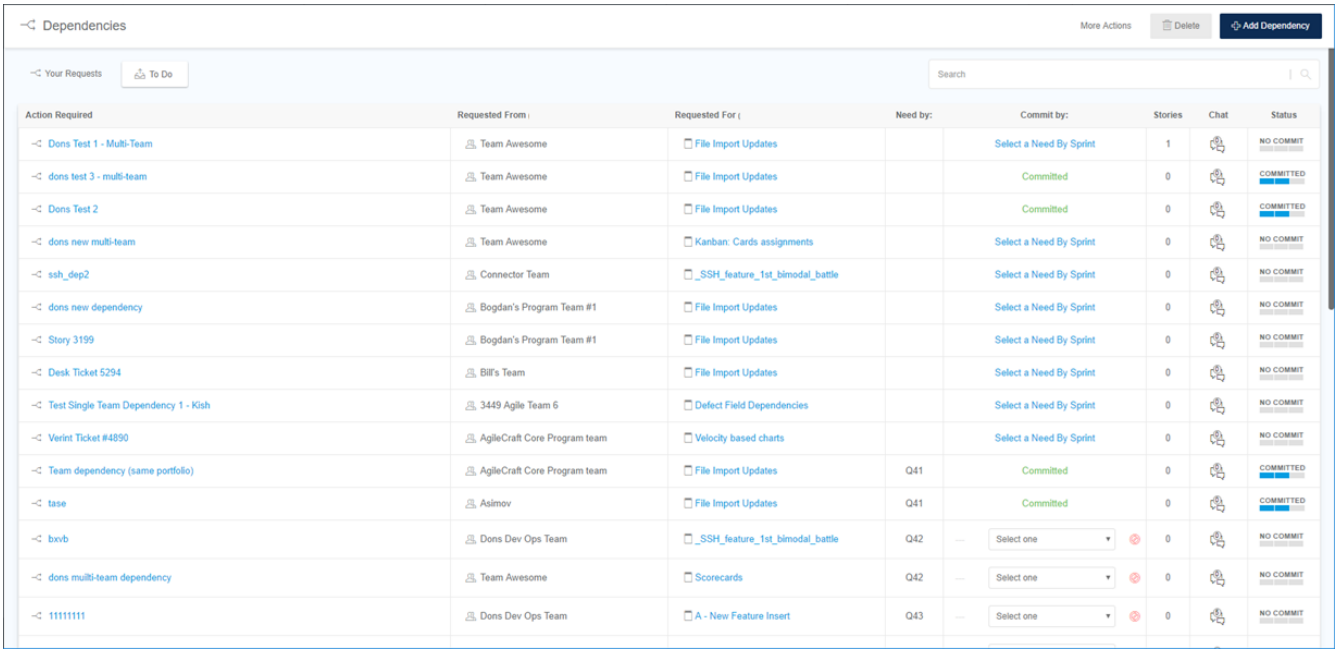
Een gedeelde PI-cadans helpt bij afstemming tussen ART's binnen en buiten de organisatie. Zorg dat contactpersonen tijdens PI Planning beschikbaar zijn. Ontdekt een team een externe dependency, dan organiseren de Scrum Master/Team Coach en Product Owner direct contact. Zo kan de andere partij het verzoek

nog tijdens het event beoordelen.

Dit veronderstelt een SAFe-omgeving. Bij samenwerking met partijen buiten SAFe maak je aanvullende afspraken over hun planningsritme, beslismomenten en levermogelijkheden.

In grote organisaties helpt het bovendien om het raakvlak te bespreken voordat de plannen vastliggen. Een leverancier kan meerdere ART's ondersteunen en dus meer aanvragen krijgen dan hij aankan. Dan moeten de betrokken Product Managers samen prioriteren. Als ieder team alleen zijn eigen aanvraag volgt, blijft dat capaciteitsprobleem gemakkelijk buiten beeld.

1.10. Tooling die de dependency orchestration maximaal ondersteunt



Figuur 5. Het digitale dependency-overzicht in Jira Align met aanvragen, betrokken partijen, levermomenten, commitment en gesprekshistorie.

Bij veel teams en ART's is een goede, gedeelde registratie van dependencies nodig. Beide partijen moeten aanvragen, toezeggingen en wijzigingen kunnen terugvinden. Organisatiebrede enterprise agile tooling helpt, mits iedereen dezelfde werkwijze volgt.

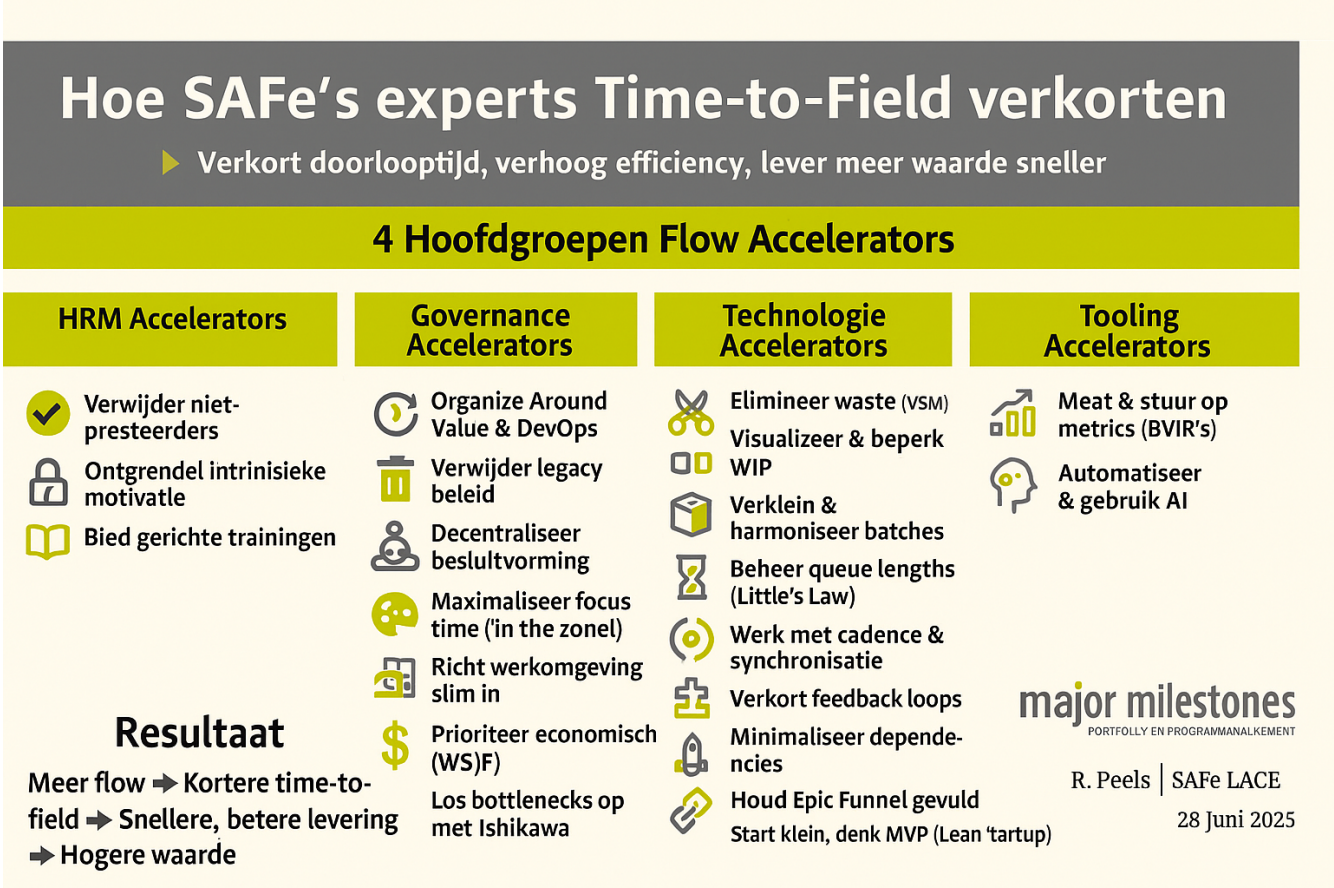
Daarom adviseer ik mijn klanten om ook af te spreken wie aanvragen beoordeelt, hoe wijzigingen worden gemeld en wanneer escalatie nodig is. Houd de registratie eenvoudig. Bespreek vooral onbevestigde aanvragen, naderende levermomenten en blokkades. Dan blijft er tijd over om het werk te doen.

1.11. Conclusie

Dependencies definiëren is één. Ze goed managen en erop sturen is twee. Zo verbeter je de voorspelbaarheid van je ART, zoals gemeten met de ART Predictability Measure. Voor mij begint dat bij goede, gedegen en uitvoerbare PI-plannen, goede stukken voor de Final Plan Review en concrete leverafspraken. Tijdens het PI blijven teams samen afstemmen wat veranderingen betekenen voor de gezamenlijke oplevering.

Daarmee krijgt dependency orchestration praktische betekenis: minder wachten op elkaar en eerder bijsturen wanneer plannen niet aansluiten.

19. Hoe SAFe experts met slimme ingrepen ‘Time-to-field’ verkorten en Value Stream efficiency verhogen



In dit hoofdstuk leer je hoe je met 22 bewezen ingrepen (van HRM tot Tooling) de time-to-field verkort, bottlenecks oplost en maximale flow creëert.

SAFe-experts beschikken over een breed arsenaal aan strategieën en interventies om de *time-to-field* (in algemene termen *Idea to Impact*) drastisch te verkorten en een continue *Flow of Value* door de hele organisatie te realiseren. Het doel: méér bereiken in minder tijd. Zeker in situaties met hoge urgentie, zoals de enorme *time-to-field*-druk die we momenteel bij Oekraïne zien, is dit essentieel.

In dit hoofdstuk ga ik uit van een klassieke situatie: een klant zonder SAFe, die traditionele (waterfall) processen uitvoert en die ontevreden is over de doorlooptijden en jou als SAFe-expert inschakelt (bijvoorbeeld RTE of SPC) om de *time-to-market* fors te verkorten en een nieuwe, optimaal draaiende ‘productiefabriek’ te bouwen. Wat doe je? Welke ingrepen pas je toe? Wat zit er in jouw SAFe-gereedschapskist? Welke interventies ga je plegen?

Hieronder vind je een overzicht van de belangrijkste praktijk *Flow Accelerators* (verdeeld in **HRM**, **Governance**, **Technology** en **Tooling**) en hoe deze bijdragen aan een kortere *time-to-market* en maximaal waarde leverende waardestroom.

Note: In sommige gevallen houd ik bewust de Engelse termen aan voor maximale theoretische herkenbaarheid. Er zit geen bewuste volgorde in de nummering.

Flow Accelerators in SAFe

HRM Accelerators

#1 Ontgrendel de intrinsieke motivatie van kenniswerkers

Oftewel: Unlock the intrinsic motivation of knowledge workers. Systemen ontwikkelen is kenniswerk: de mensen die het doen, weten zelf het beste hóe dat moet. Ze komen gemotiveerd binnen, willen bijdragen en verdienen vertrouwen en autonomie. Geef ze de ruimte, en je benut hun volledige potentieel. Goed gemotiveerde teamleden verhogen de productiviteit en helpen zo de Time-to-Market te verkorten.

Autonomie, mastery en purpose zijn hierin onmisbaar. Autonomie en vakmanschap vragen om vertrouwen en ontwikkelkansen. Purpose vraagt om een heldere visie (van strategische thema's tot productvisies) zodat iedereen weet waar hij of zij aan bijdraagt. Zo ontstaat het gevoel van 'samen sterk': niet alleen binnen teams, maar door de hele organisatie (=Value Stream Network) heen.

#2 Geef juiste en gerichte trainingen

Een goed draaiende SAFe-fabriek werkt alleen als de mensen het framework, de begrippen en de processen snappen. Training is de sleutel tot een optimaal functionerend systeem met een maximale flow of value. Geen kennis = geen goed begrip = geen maximale flow. Richt je dus op doorlopende training en ontwikkeling van iedereen in de fabriek.

#3 Zorg dat je de juiste mensen op de juiste plekken hebt. Neem daarna afscheid van niet-presterende teamleden en de onveranderlijken

Alles begint met mensen die willen groeien en openstaan voor verandering. Het is daarom cruciaal dat je, waar mogelijk, alleen de allerbeste mensen toelaat in Kotters Value Stream Network. Creëer eerst een omgeving waarin iedereen kan excelleren. Lukt dat, en blijven sommige medewerkers structureel onderpresteren of zelfs contraproductief werken? Dan coach je, begeleid je, geef je een kans, maar daarna houdt het op.
Niet-functionerende mensen vertragen de hele keten en halen energie weg bij goed presterende teams. Wacht dus niet te lang met ingrijpen: de vaart van je verandering hangt direct samen met de kwaliteit van je mensen.

Governance Accelerators

#1 Organize Around Value, pas DevOps toe en werk incrementeel

De belangrijkste structurele ingreep is *Organize Around Value* en het toepassen van Kotter's *Dual Operating System*. Werk stroomt moeizaam door silo's en afdelingen die rond competenties zijn georganiseerd. Identificeer je Value Streams en bouw Agile DevOps-teams en ART's die de waardestroom van begin tot eind bedienen. Voer het werk vervolgens iteratief, op cadans gebaseerd uit.

#2 Remediate Legacy Policies and Practices

Het aanpassen of verwijderen van verouderde beleidslijnen en praktijken verwijdert barrières die innovatie en snelle levering belemmeren, waardoor teams efficiënter en effectiever kunnen werken. Als je overgaat naar SAFe is het verstandig te stoppen met oude ceremonies en overleggen of minimaal de bestaande overleggen zoveel mogelijk te migreren en te transformeren naar de standaard SAFe-overleggen (Portfolio Sync, Coach Sync etc.) en te stoppen met oude gewoonten zoals het maken van lange PID's.

#3 Decentraliseer zoveel mogelijk de besluitvorming

Agilisten “decentraliseren besluitvorming” wat betekent dat beslissingen sneller kunnen worden genomen door mensen met de meeste context en expertise. SAFe heeft een governance die gericht is op het proces, bijvoorbeeld het verankeren van besluitvorming in de SAFe overleggen waar alle stakeholders vertegenwoordigd moeten zijn. Dit versnelt de voortgang omdat teams niet hoeven te wachten op goedkeuring van hoger management. SAFe kent zodoende geen centrale besluitvormingsorganen (zoals voorheen CAB’s en stuurgroepen).

#4 Optimize ‘Time in the Zone’

Ontwerp een slimme cadence, met alle overleggen op slimme vastgestelde tijdstippen, die ervoor zorgen dat teams zich maximaal op het werk kunnen richten, gefocust, geconcentreerd, zonder onderbrekingen en zonder hoge ‘task switching costs’. Professionals, knowledge workers hebben in het algemeen een hekel aan interrupties. Dit helpt teams om in een flow-toestand te komen waarin ze efficiënt en effectief kunnen werken.

#5 Richt juiste workspace en indeling in

Het Agile Manifesto benadrukt dat de meest efficiënte en effectieve manier om informatie over te brengen binnen een team *face-to-face* communicatie is. Dat betekent dat je de werkomgeving zo moet inrichten dat mensen direct, optimaal, snel met elkaar kunnen samenwerken, zonder onnodige muren of organisatorische schotten. Vermijd dat teams verspreid raken in aparte hokjes en eilanden: korte communicatielijnen en een open, gezamenlijke (stille en rustige) werkplek zorgen voor snellere afstemming, directe feedback en minder ruis. Een slim ingerichte werkruimte fungeert als accelerator voor productiviteit en flow.

#6 Prioriteer economisch voor maximale waardelevering

Bij vele organisaties is de vraag groter dan het aanbod met dus een executive push-strategie. Mijn advies: take an economic view in plaats van HSE modellen (Hardste Schreeuwer Eerst)! Gebruik hiervoor bijvoorbeeld WSJF (Weighted Shortest Job First) om met dezelfde mensen de meeste waarde te leveren. Denk in economische baten, niet in uren. Verminder en beperk ook de influx van alle initiatieven (Epics), prioriteer zo snel als mogelijk. Want al het uitwerken kost heel veel tijd, die je ook op de implementatie kan zetten.

#7 Los complexe pipeline bottlenecks op met Ishikawa

In elke waardestroom ontstaan onvermijdelijk bottlenecks, en sommige zijn hardnekkig en blijven steeds terugkomen. Om zulke knelpunten echt op te lossen, moet je de kern van het probleem vinden en aanpakken. Met een **Ishikawa-diagram** (visgraatdiagram) breng je alle mogelijke oorzaken overzichtelijk in kaart. Zo vind je de échte *root cause*, los je die structureel op en verbeter je de *flow of value* blijvend. Gebruik een visgraatdiagram bijvoorbeeld voor het oplossen van een fout in je ontwerp die elke PI lijkt terug te komen. Zo voorkom je dat dezelfde blokkade telkens opnieuw vertraagt met langere Time-to-Market.

#8 Voer ‘SAFe Accelerating Flow Workshops’ uit

Tot slot: SAFe kent zelf standaard 8 van deze Flow Accelerators, die diepgaand worden uitgewerkt in de SAFe Accelerating Flow Workshop en de bijbehorende toolkit. Door deze accelerators diepgaand uit te werken werk je systematisch en consistent aan flowverbetering. Kortom: maak slim gebruik van wat er is, en benut de kracht van bewezen tools.

Technologie Accelerators

#1 Eliminate Waste: Eliminate Non-Value-Added Activities & Handovers (VSM workshops).

Voer een Value Stream Mapping-workshop uit en elimineer alle waste: door het elimineren van activiteiten die geen waarde toevoegen, alles zo simpel als mogelijk te houden, overdrachten te minimaliseren door cross-functional teams in te richten, kunnen teams zich richten op taken die direct bijdragen aan de levering van waarde aan de klant. Dit vermindert verspilling en verbetert de efficiëntie.

#2 Visualize and Limit Work in Process (WIP) & Pull

Het visualiseren van werk in *Big Visible Information Radiators* (BVIR's) en het beperken van het werk in uitvoering (WIP limits) helpt teams om zich te concentreren op het voltooien van taken voordat ze aan nieuwe beginnen. Zo heeft Jira bijvoorbeeld WIP limits ingebouwd in hun software. Dit voorkomt veelvuldig wisselen tussen taken, overbelasting en versnelt de doorstroom van werk door het systeem.

#3 Verklein Batch Sizes en maak ze gelijk

Het verkleinen van de batchgrootte, dus kleine User Stories in Jira/ADO, betekent dat werk in kleinere, beter beheersbare delen wordt opgesplitst. Idealiter is 1 user story tussen de 2 en 4 dagen. Dit maakt snellere feedback mogelijk, vermindert risico's en verkort de time-to-field doordat kleine, waardevolle incrementen sneller kunnen worden opgeleverd. Kleine batch sizes versnellen de flow of value. Zorg parallel ervoor dat batches ongeveer even groot zijn.

#4 Manage Queue Lengths (Little's Law)

Het beheren van de lengte van wachtrijen helpt bij het verminderen van de doorlooptijd en het verbeteren van de voorspelbaarheid van leveringen. Korte wachtrijen zorgen ervoor dat werk sneller door het systeem stroomt met minder variaties.

#5 Enhance Flow with Cadence and Synchronization

Het werken met een regelmatige cadans en het synchroniseren van alle teams zorgt voor voorspelbaarheid en maakt het eenvoudiger om voortgang en risico's te beheren.

#6 Get Faster Feedback

Door feedbackloops te verkorten en te versnellen, kunnen teams sneller reageren op veranderingen en problemen vroegtijdig oplossen wat ten goede komt aan een korte Time-to-Market. Dit wordt bereikt door frequente integratie, testen en demonstraties van incrementele voortgang.

#7 Minimize Dependencies

Het minimaliseren van overdrachten en afhankelijkheden tussen teams vermindert vertragingen en fouten die voortkomen uit overdrachten, en verhoogt de snelheid en kwaliteit van leveringen.

#8 Zorg voor continue Epic Funnel Flow

Epic Owners werken samen met de ART om Epics op tijd op te knippen in behapbare Features en Enablers, ruim vóór het Planning Interval (PI) waarin het werk daadwerkelijk start. Dit opknippen gebeurt vaak tijdens backlog refinement-sessies. Als dit proces niet goed verloopt, stokt de aanvoer naar de 'SAFe-fabriek' en functioneert deze nooit optimaal. Een voorspelbare en stabiele flow begint bij een goed gevulde funnel: het begint allemaal bij de start!

#9 Denk in MVP's: Deliver early, learn fast, adapt faster.

Een krachtige manier om de *time-to-market* te verkorten is niet te mikken op de perfecte eindoplossing, maar klein te starten volgens het Lean Startup-principe van Eric Ries. Bouw een Minimum Viable Product (MVP) van de Epics met de allerhoogste WSJF's: een eerste werkbare versie die waarde levert en snel feedback oplevert. Zo test je aannames vroeg, leer je van gebruikers en kun je slim en sneller bijsturen. Dit voorkomt dure verspilling achteraf en versnelt de flow van idee naar echte waarde.

Tooling Accelerators

#1 Stuur op metrics

Gissen is missen, gokken is dokken, meten is weten: een waarheid die ook hier geldt. Zonder duidelijke metrics en dashboards (BVIR's) heb je geen zicht op hoe jouw 'fabriek' echt presteert. Wat je niet meet, kun je immers ook niet verbeteren. Meten maakt prestaties zichtbaar én stuurbaar. Zo bouw je gericht aan een steeds betere flow en hogere waarde.

#2 Automate & AI

Automatisering (DevOps) van bijvoorbeeld je Source Code Quality verkleint de kans op menselijke fouten en bespaart tijd op handmatig werk. Werk wordt hierdoor herhaalbaar met dezelfde kwaliteit uitgevoerd. Met de hedendaagse AI tooling en inzet daarvan voeg je daar nog een versnelling aan toe!

Kortom

Door deze Flow Accelerators te implementeren, richt SAFe zich op het optimaliseren van elke stap in het ontwikkelingsproces. Dit verhoogt de efficiëntie, voorspelbaarheid en productiviteit van teams, en zorgt voor snellere feedback en besluitvorming. Teams kunnen zich beter concentreren op waardetoevoegende activiteiten en sneller reageren op veranderingen en feedback, wat leidt tot een meer responsieve en wendbare organisatie. Hierdoor kunnen organisaties hun workflows optimaliseren, verspilling minimaliseren en de snelheid en kwaliteit van hun leveringen verbeteren, wat uiteindelijk leidt tot een kortere time-to-field voor onze gebruikers en ons land, waarvoor we het uiteindelijk allemaal doen.

Deel 3: SAFe-transformatie

20. Visie op de Agile transformatie: waar organisaties mee stoppen en mee beginnen

Van silo's en projecten naar waardestromen, producten en continue levering

Inleiding

Een agile transformatie bij een grote organisatie is geen project met een begin- en einddatum. Het is een fundamentele verschuiving in de manier waarop een organisatie denkt, werkt en zichzelf organiseert, en dat raakt tegelijkertijd aan strategie, structuur, cultuur, technologie en governance.

Wie als LACE (Lean-Agile Center of Excellence, het interne transformatieteam) of lid van het executive management het woord 'SAFe' laat vallen, riskeert een bekende reactie: ogen die ten hemel slaan, gefronste wenkbrauwen en de onvermijdelijke opmerking vanuit de zaal, "daar hebben we er weer een." Die scepsis is begrijpelijk. Veel organisaties hebben al eerder grootschalige verandertrajecten doorgemaakt, denk aan PRINCE2, MSP, ITIL, Lean etc met wisselend succes en achterblijvende resultaten. Het label 'Framework' roept daarom vaak reflexmatig weerstand op.

Een tweede, minstens even hardnekkige reactie is: "nieuwe wijn in oude zakken!". Dat is een van de grote gevaren van een slecht ontworpen transformatie: wie de taal niet aanpast en oude woordjes gebruikt voor nieuwe concepten zoals PI-event, krijgt precies die reactie. Precies daarom is nauwe samenwerking met changemanagers en communicatie-experts geen bijzaak, maar een voorwaarde. Zij helpen de organisatie te zien wat er werkelijk anders is: dus ook niet een nieuw label op bestaand gedrag en werkwijze, maar een fundamenteel andere manier van prioriteren, produceren, beslissen en samenwerken. Die boodschap vraagt om heldere taal, consistente verhaallijnen en zichtbare gedragsverandering op alle niveaus: van de werkvloer tot de boardroom.

"Een transformatie zonder zichtbare verandering is geen transformatie"

Daarom is de centrale boodschap van dit hoofdstuk bewust anders: **we implementeren geen SAFe**. We introduceren een betere manier van organiseren en samenwerken gebaseerd op bewezen principes en bewezen best practices, zonder het label "SAFe" centraal te stellen. We pakken de huidige problemen aan door een andere manier van opereren, omdat onze veranderende organisatiecontext dat vraagt. SAFe® is daarbij het kompas dat we gebruiken, niet het verhaal dat we verkopen.

Dit hoofdstuk geeft concrete antwoorden op de vraag die altijd volgt: 'Maar wat gaat er dan precies veranderen?' Er is een context, er zijn interne problemen maar wat is dan jouw visie op de aanpak, en wat ga jij als Transformatie Consultant concreet veranderen? Op die vragen geeft dit hoofdstuk in detail antwoord.

Ideeën en tips voor het opstellen van je verandervisie

Onderstaande tabel laat zien wat organisaties **moeten stoppen** en waar zij **mee moeten beginnen,** dus wat er verandert. Ik deel deze lijst op in 2 delen. Ten eerste absolute essentie van de verandering van de werkwijze en ten tweede overige zaken. De eerste lijst kun je dan gemakkelijk gebruiken in je transformatiecommunicatie.

De essentie

✕ VAN *Oude situatie*		✓ NAAR *Nieuwe situatie*
Alles door elkaar		**Een maximaal lean ingerichte organisatie waar *alle* Solutions in OS #2 worden gebouwd, alleen aangestuurd via LPM, makkelijk aan derden uitlegbaar, robuust en schaalbaar.**
Silo Based, werken in afdelingen De organisatie is georganiseerd rondom hiërarchische afdelingen	→	**We gaan organiseren rondom waarde** We gaan werken in samenwerkingsverbanden van teams, End-to-End, aan integrale oplossingen met dedicated capaciteit. Mitigeer Conway's law.
Waterfall opleveringen We leveren alles in één keer, bij einde project op en dragen over aan 'instandhouding, beheers organisatie'		**incrementeel waarde leveren** We gaan incrementeel geïntegreerde en uitgeteste waarde leveren: elke iteratie, elke PI.
Silo Based: Gescheiden Dev en Ops Development en Operations werken los van elkaar vaak in twee verschillende afdelingen; integratie en deployment zijn pijnpunten, 'over de muur gooien', 'we worden niet vroegtijdig betrokken' etc.	→	**Dev, Ops en Sec werken samen in Cross-functional teams** Dev, Ops en Security zijn geïntegreerd in cross-functionele teams met geautomatiseerde pipelines. Agile Teams bouwen, onderhouden, refactoren en lossen bugs op.
Tijdelijke projectteams met grote resourceplanningen Teams worden per project gevormd en daarna ontbonden. Kennis verdwijnt, context gaat verloren, teamleden zitten weer op de bank. Het is lastig een high performance team te worden.	→	**Stabiele, langdurige agile devops teams (long-lived).** Agile teams blijven samen rondom een waardestroom. Ze bouwen diepgaande domeinkennis op en leveren voorspelbaar. Ze blijven (theoretisch) bij elkaar tot de Solution niet meer bestaat.
Mensen verplaatsen naar het werk (projectmatig) Mensen worden aan projecten toegewezen en verhuizen van context naar context. Onmogelijk High Performance Teams te bouwen	→	**Werk toewijzen aan Stabiele Agile Teams** Teams zijn stabiel; het werk (user stories, features) komt naar de teams, niet andersom. Daardoor kunnen HPT's ontstaan, omdat agilisten langdurig met elkaar samenwerken.
Complexe, hiërarchische organisatiestructuur Veel managementlagen, overdrachten tussen afdelingen en lange lijnen tussen klant en uitvoering vertragen de levering.	→	**Simplificeren middels Lean-Agile structuren** Slanke structuren rondom waardestromen en ARTs, met korte feedbackloops en zo min mogelijk overdrachten.
Geen organisatie cadans Er is geen organisatie brede heart beat, met vaste alignment en integratie momenten.	→	**Develop on Cadence, Release on Demand** Vaste events, vaste tijden, vaste agenda, zelfde mensen. Rust en regelmaat.
Van Push Elk initiatief wordt gestart, ongeacht capaciteit. Resultaat: files, vertragingen en lage kwaliteit, stress en burnouts	→	**Naar Pull: Vraag en capaciteit op elkaar afstemmen** Teams trekken werk naar zich toe op basis van beschikbare capaciteit. PI Planning borgt realistische commitments.

✕ VAN *Oude situatie*		✓ NAAR *Nieuwe situatie*
Vaak zeer complexe, lastig te begrijpen governance Zoals John Kotter zo mooi zei, worden om de lijn heen stuurgroepen geformeerd, zelfgevormde werkgroepjes, overleg groepjes, tal van coördinatoren om de train op de rails te houden en zonder goede horizontale en verticale afstemming.	→	**Out of the box governance: Simplicity en consistentie.** SAFe bevat een uitgewerkt governance-model met duidelijke rollen, beslismomenten en alignment-events, en legt besluiten zo laag mogelijk neer (Lean-Agile Principle #9: decentraliseer besluitvorming). Hierdoor worden veel ad-hoc governance en tussenoplossingen overbodig.
Verschillende talen, begrippen, rollen etc. Elk team en elke afdeling heeft eigen terminologie voor de governance (“zeepkist sessie”, etc). Communicatie kost energie, misverstanden zijn structureel. Velen hebben bij hetzelfde woord een ander beeld zoals alleen al het woord ‘Solution’.	→	**Gemeenschappelijke SAFe® taal & terminologie** Eén gemeenschappelijk vocabulaire voor rollen (Product Owner, RTE, Scrum Master/Team Coach), events (PI Planning, Iteration Review) en alle artefacten. Iedereen verstaat elkaar, er komt een markt conform Uniform Begrippen Kader (UBK).

Overige veranderingen

✕ VAN *Oude situatie*		✓ NAAR *Nieuwe situatie*
We bouwen wat intern bedacht is Geen klantvalidatie.	→	**Continue klantvalidatie via Design Thinking, Customer Centricity en hypothesis-driven development** De stem van de klant stuurt prioriteiten
Kwaliteit is een QA-probleem achteraf Testen na oplevering	→	**We gaan naar Built-in Quality** Kwaliteit wordt ingebouwd door continuous testing, shift left, testautomatisering en Definition of Done per team en ART.
Projectgebaseerde en fluctuerende financiering Budget wordt per project toegewezen en ingetrokken, complexe financiering. Kans op budgetproblemen bij uitloop project.	→	**Lean Budgets per waardestroom** Financiering wordt per ART/waardestroom toegewezen voor een langere periode. Stabiele capaciteit, minder herplanningsoverhead. Eenvoudiger.
Eilandjes architectuur; geen alignment met werkvloer Architectuur wordt los van de teams bedacht en opgelegd, zonder verbinding met de dagelijkse praktijk.	→	**Intentional Architecture + Emergent Design** Richt architectuur in op de huidige technologie en modulaire opbouw. Architecten en teams werken samen. System Architects zitten in de ARTs en coachen teams richting een gedeelde technische visie.
Variatie en maatwerk per team of afdeling Ieder team of iedere afdeling werkt op zijn eigen manier: eigen processen, eigen tools en een eigen definitie van ‘klaar’. Resultaten zijn niet vergelijkbaar en samenwerking over teamgrenzen heen loopt stroef.	→	**Eén gestandaardiseerde werkwijze (SAFe®)** Eén gemeenschappelijk framework voor alle teams: zelfde cadence, zelfde events, een gedeelde Definition of Done en geïntegreerde tooling.
Management bij Excel / Powerpoint Besluitvorming en rapportages op basis van (eindeloze) handmatige spreadsheets en PowerPoints die per email worden gedistribueerd.	→	**Enterprise Agile software suite** Realtime enterprise managementinformatie via geïntegreerde corporate Agile tooling (bijv. Jira / Jira Align) in combinatie met bijvoorbeeld Confluence. Significante email reductie en ms office opmaak tijd.
Alles van tevoren tot op de puntkomma uitdenken Uitgebreide Big Designs Upfront en specificaties die verouderd zijn voordat het werk begint. (BDUF’s)	→	**Incrementeel plannen en ontdekken** Planning op meerdere niveaus: Visie en Roadmap op lange termijn, PI Planning per 8-12 weken, Sprint per 2 weken.
Prioriteren op basis van HSE (Hardste schreeuwers eerst) Prioriteiten worden bepaald op basis van politiek, de luidste stem of gevoel of meeste sterren op de borst.	→	**Economisch Beslissen op basis van data & objectieve criteria** Prioritering via CoD, WSJF (Weighted Shortest Job First), OKR's en meetbare businesswaarde.

Mijn afsluitende advies voor een paar belangrijke spelregels

Naast de bovenstaande veranderingen hanteren we een set concrete spelregels die borgen dat de transformatie consistent en professioneel verloopt. Deze spelregels zijn bij voorkeur in een transformatie niet onderhandelbaar, ze vormen het fundament waarop we bouwen. Vanaf de start.

1. SAFe® als leidend framework - SAFe tenzij…

We houden ons aan het SAFe®-framework. We volgen de laatste versie van het framework, best-practices en Agile mindset, met een maximale vertraging van 1 jaar na de nieuwste release. Aanpassingen, uitbreidingen op het Framework ('customisaties') worden gedocumenteerd, inclusief redenen, impact en gevolgen zodat nieuwe werknemers, consultants heel snel weten hoe SAFe is geïmplementeerd en wat een organisatie miste of welke begrippen zijn aangepast (bijvoorbeeld: de PM op ART-niveau noemen wij ‘Chief PO’).

2. 1 taal en 1 set van begrippen

We vermijden het mixen van termen uit andere frameworks. Geen 'tribes', 'squads', 'guilds' of 'replenishing meetings'. Vertalingen van manifesto-regels, rollen en events naar andere talen worden niet toegepast. Consistentie in taal = consistentie in begrip.

3. Alle agilisten zijn getraind en gecertificeerd

Alle deelnemers zijn minimaal getraind in SAFe® V6.0. Kernrollen zijn specifiek en op maat getraind en gecertificeerd. Maatwerk betekent een mix tussen SAFe® training en toepassing in de specifieke organisatiecontext. Dit borgt kwaliteit en voorkomt 'SAFe in naam alleen' (SINO).

4. Taalflexibiliteit

Teamleden kunnen kiezen tussen Engels (US spelling) of Nederlands, met oog voor samenwerking met internationale partners. Of, alles wat we doen is NL of alles wat we doen is UK. Stel vast.

Conclusie: transformatie is een reis, geen bestemming

Een agile transformatie bij een grote organisatie is complex, maar niet onmogelijk. De sleutel ligt in het combineren van bewezen frameworks met pragmatisch vakmanschap. SAFe® biedt het kompas, de leidende principes bieden richting, en onze spelregels borgen consistentie.

Wat we veranderen is niet alleen de manier van werken, maar de manier van denken. Van output naar outcome. Van controle naar vertrouwen. Van plannen naar aanpassen. En dat alles met één doel: sneller, voorspelbaarder en met hogere kwaliteit waarde leveren aan de eindgebruiker.

21. Van pitch naar praktijk: de kunst van het verkopen van je SAFe-transformatie

De kunst van het goed pitchen van je businessidee

Je kent het vast: een gedreven ondernemer/founder staat voor een rij kritische investeerders in het bekende tv-programma *Dragons' Den en* heeft slechts een paar minuten om hen te overtuigen. Geen overbodige woorden, geen lange uitwijdingen. Gewoon een messcherp verhaal: wat is het probleem, wat is de oplossing, waarom werkt het, en waarom moet je nú instappen? 15 min zendtijd.

Het is spannend om naar te kijken. Maar wie goed oplet, ziet dat de beste pitches niet winnen op het idee alleen, ze winnen op een overtuigend, goed gestructureerd en tot in den treure geoefend verhaal. De structuur is vrijwel altijd hetzelfde: begin met pijn, eindig met perspectief.

En er is nog iets dat elke goede pitcher begrijpt: als je je verhaal goed vertelt, verandert de energie in de zaal. Mensen gaan naar voren leunen, vragen worden nieuwsgierig in plaats van sceptisch. Dat is het moment waarop de ondernemer zijn A*sk* doet: de vraag om een investering die zijn idee werkelijkheid kan maken.

Van startup pitch naar het pitchen van je organisatietransformatie

Opvallend genoeg lijkt dit sterk op wat er gebeurt tijdens grote organisatieveranderingen, zoals een SAFe-implementatie.

Misschien herken je dat: jij staat als transformatieleider voor een grote groep medewerkers en moet uitleggen wat er gaat veranderen, waarom het gebeurt en wat het gaat opleveren. Het publiek is alleen anders. Geen investeerders met een open portemonnee, maar nieuwsgierige medewerkers, agilisten, business stakeholders en managers met een gezonde dosis scepsis. *"Na ITIL, ASL2, BISL, Lean, MSP & PRINCE2 weer een framework…!"*

Toch lijkt de structuur van een transformatieverhaal opvallend veel op een startup pitch. Alleen vraagt de transformatieleider aan het einde niet om een financiële investering in ruil voor aandelen. Hij vraagt om iets dat misschien nog waardevoller is: **buy-in voor de transformatie**.

De bereidheid van mensen om mee te gaan in de verandering, weerstand tegen de verandering te verminderen. En idealiter zelfs meer dan dat: mensen die zó enthousiast worden dat ze ambassadeurs van de verandering worden en zelf in de wandelgangen de SAFe-transformatie gaan verkopen en verdedigen.

Want een SAFe-transformatie die alleen van bovenaf wordt gedragen, strandt vaak. Een transformatie die gedragen wordt door mensen **die erin geloven, krijgt momentum.**

Wanneer je een Agile of SAFe-transformatie uitvoert, speelt communicatie daarom een cruciale rol. Die communicatie vindt meestal op meerdere manieren plaats. In veel organisaties worden bijvoorbeeld grote plenaire sessies, workshops, open dagen georganiseerd waarin medewerkers worden meegenomen in het verhaal van de transformatie. Tegelijkertijd worden vaak brochures, artikelen of interne informatiedocumenten (PowerPoint-slides) gemaakt waarin wordt uitgelegd wat er gaat veranderen en waarom.

Interessant genoeg kunnen al deze vormen van communicatie gebruikmaken van dezelfde structuur als een goede pitch. Of je nu op een podium staat voor honderden medewerkers, of een brochure schrijft die de organisatie uitlegt wat er gaat gebeuren, de kracht zit in een helder, logisch opgebouwd verhaal met een zeer sterke verhaallijn.

In dit hoofdstuk bekijken we eerst hoe startups hun verhaal pitchen aan investeerders. Daarna vertalen we deze structuur naar het pitchen van een SAFe-transformatie voor een grote groep van mensen.

Deel 1: Hoe startups hun verhaal pitchen aan investeerders

In de startupwereld is in de loop der jaren een vrij vaste structuur ontstaan voor succesvolle pitches. Vrijwel elke investeerderspitch bevat dezelfde bouwstenen. Een van de bekendste experts op dit gebied is Guy Kawasaki. Kawasaki was een van de eerste medewerkers van Apple in de jaren '80, waar hij als "Chief Evangelist" verantwoordelijk was voor het promoten van de Macintosh. Later werd hij durfkapitalist, auteur en een van de meest geciteerde stemmen in de startupwereld.

Zijn bekendste bijdrage aan pitchpresentaties is de **10-20-30 regel**. Volgens Kawasaki bestaat een effectieve pitch uit maximaal tien slides, duurt de presentatie niet langer dan twintig minuten en gebruikt elke slide een minimale lettergrootte van dertig punten.

Het idee erachter is simpel: **minder is meer**. Slides horen ondersteunend te zijn aan het verhaal en niet het verhaal zelf te worden. Idealiter bevat een slide slechts een paar kernpunten, bij voorkeur drie - en echt *maximaal* vijf bullets zodat het publiek naar de spreker blijft luisteren in plaats van de slides te lezen.

Volgens Kawasaki bevat een sterke pitch de volgende tien onderdelen:

1. **Het probleem**: Wat is er mis in de wereld en waarom doet dat ertoe?
2. **De oplossing**: Hoe lost jouw product of dienst dat probleem op?
3. **Businessmodel**: Hoe wordt er geld verdiend?
4. **De onderliggende magie**: Wat maakt jouw oplossing uniek?
5. **Marketing & sales**: Hoe bereik je je klanten?
6. **Concurrentie**: Wie zijn de alternatieven en waarom win jij?
7. **Het team**: Waarom is dit het juiste team?
8. **Financiële prognoses**: Wat zijn de verwachte resultaten?
9. **Huidige stand van zaken**: Wat is al bereikt?
10. **The Ask**: Wat vraag je van de investeerder?

Bron: https://guykawasaki.com/the-only-10-slides-you-need-in-your-pitch/

Deel 2: De pitchstructuur toegepast op een SAFe-transformatie

De pitchstructuur van Kawasaki is commercieel bedoeld, maar verrassend goed te vertalen naar een SAFe-implementatie. Ook een transformatieleider moet immers een verhaal vertellen dat mensen overtuigt.

We gaan hier uit van een organisatie die al heeft gekozen voor SAFe als framework en waar mogelijk al enkele Agile Release Trains (ARTs) actief zijn. De manier waarop je het verhaal presenteert, zal uiteraard altijd afhangen van de fase van de transformatie en de specifieke context van de organisatie.

Hieronder volgt een voorstel voor de structuur voor jouw topnotch SAFe-pitchdeck. Deze structuur is niet alleen bedoeld voor een presentatie tijdens een plenaire sessie of een kick-off met medewerkers. Dezelfde verhaallijn kan namelijk ook worden gebruikt voor andere communicatiemiddelen binnen de transformatie, zoals een brochure, een intern artikel, onboardingsdocument of een PowerPoint-deck zoals al eerder aangegeven. Zorg dat je hele communicatie consistent van aard is.

Belangrijk daarbij is dat deze communicatie nooit een doel op zich is. Je organiseert geen infodagen om het houden van infodagen, en je maakt geen brochure om het maken van een brochure.

Elke vorm van communicatie heeft een doel: **The Ask**. Wat wil je dat er gebeurt nadat mensen jouw verhaal hebben gehoord of gelezen? Wil je dat teams zich aanmelden voor een pilot-ART, dat managers nieuwe rollen gaan ondersteunen, of dat medewerkers actief gaan deelnemen aan communities of practice? Die gewenste actie moet dus al duidelijk zijn voordat je je pitchdeck, brochure of presentatie gaat ontwerpen.

Mijn voorgestelde indeling voor je verhaallijn ('storyline')

• **Cover**: Pakkende titel die de beweging van de organisatie samenvat

• **De context**: Externe druk en waarom niet veranderen geen optie meer is (sense of urgency)

• **Het probleem**: Interne belemmeringen en zichtbare effecten (lange time-to-market, afhankelijkheden, etc.)

• **De visie**: Toekomstbeeld op basis van principes, niet het framework zelf

• **De oplossing**: Target Operating Model en SAFe-ontwerp (ARTs, teams, governance, rollen)

• **Hoe het werkt**: Flow van idee naar waarde, samenwerking binnen ART, CI/CD, financiering, kwaliteit

• **Voordelen -** Wat zijn de voordelen van deze oplossing?

• **Wat verandert er voor wie / impact**: Vertaling naar dagelijkse praktijk per rol, en wat blijft hetzelfde

• **Bereikte resultaten / bewijzen**: Quick wins, eerste successen, traction (pilot-ARTs, kortere doorlooptijden, etc.)

• **Implementatie roadmap / ambitie**: Fasen, mijlpalen, trainingen, tooling, uitbreiding scope

• **Communicatie en betrokkenheid**: Hoe medewerkers worden meegenomen en continu geïnformeerd

• **Kritieke succesfactoren / risico's / lessons learned**: Randvoorwaarden, valkuilen, wat al geleerd is

• **Het team**: LACE, transformatieleider, Agile coaches, executive sponsors

• **Investering / Business Case** *(optioneel)*: Kosten, tooling, businesscase

• **The Ask**: Duidelijke oproep tot actie; wat verwacht je dat mensen nu gaan doen?

• **Contact / eindslide**: Waar mensen terecht kunnen, hoe ze vervolg geven aan The Ask

Deel 3: Alle hoofdstukken toegelicht

1. Cover

Begin met een pakkende titel en een duidelijke visuele cover die meteen de kern van de transformatie samenvat. De titel moet energie geven en nieuwsgierig maken. Goede titels beschrijven vaak de beweging die de organisatie wil maken, bijvoorbeeld dat de organisatie zich voortaan rondom waarde organiseert, samen wendbaarder wil worden, of de stap maakt van projectmatig werken naar een continue flow van waarde.

2. De context

Leg vervolgens de context uit waarin de organisatie zich bevindt. Waarom is verandering nodig? Wat speelt er in de wereld rondom de organisatie waardoor de huidige manier van werken onder druk staat? Denk aan toenemende complexiteit, sneller veranderende technologie, hogere verwachtingen van gebruikers of langere doorlooptijden dan wenselijk.

Deze slide is misschien wel de belangrijkste slide van de hele pitch. Hier moet het publiek voelen waarom niet veranderen geen optie meer is. Idealiter knikken de aanwezigen op dit moment instemmend: *ja, dit herkennen we*. Als mensen de pijn van de huidige situatie niet herkennen of niet voelen, wordt het vrijwel onmogelijk om draagvlak te creëren voor de transformatie. De sense of urgency moet hier dus echt landen bij het publiek.

3. Het probleem

De vorige slide beschreef de externe druk; deze slide gaat over onze interne belemmeringen: maak het interne probleem concreet. Welke structurele belemmeringen zitten in de huidige manier van werken? Wat blokkeert ons om sneller te veranderen of beter te leveren? Zijn we wel goed georganiseerd? Gaan de integratietests vaak fout? Te hoog ziekteverzuim? Veel afhankelijkheden tussen teams of onduidelijk eigenaarschap over producten en diensten?

Benoem vervolgens ook de **zichtbare effecten**. Denk aan een lange time-to-market, projectmatig werken waardoor kennis telkens opnieuw moet worden opgebouwd, ontevreden klanten, de enterprise organisatie doelen worden niet gerealiseerd.

De gevolgen zijn vaak duidelijk: gefrustreerde gebruikers, oplopende doorlooptijden, minder voorspelbare leveringen en een organisatie die aan wendbaarheid verliest. Door deze effecten expliciet te benoemen wordt duidelijk waarom de **huidige organisatie inrichting en besturing niet langer goed werkt.**

4. De visie

Na het probleem volgt de visie. Als de huidige situatie niet werkt, moet je als pitcher uitleggen hoe jij denkt dat de organisatie wél zou moeten werken. Dit is het moment waarop je het toekomstbeeld schetst en laat zien welke richting je op wilt.

Belangrijk daarbij is dat je niet blijft hangen in het noemen van het SAFe framework zelf als de oplossing. Je pitcht **niet het framework zelf**, maar je pitcht de achterliggende principes **en best practices**.

Beschrijf bijvoorbeeld dat de organisatie zich (1) **gaat organiseren rondom waarde in plaats van rondom projecten en afdelings silo's**, (2) dat er **stabiele en long-lived Agile DevOps-teams** komen, dat de beweging wordt gemaakt van (3) **projects naar products** en dat bijvoorbeeld (4) **feedbackloops met gebruikers korter worden**. Dit zijn enkele voorbeelden. Zie ook hoofdstuk 20, 'Visie op de Agile transformatie: waar organisaties mee stoppen en mee beginnen'.

In deze slide presenteer je dus de visie op hoe het probleem wordt opgelost en hoe de organisatie er in de toekomst gaat werken, niet hoe die eruit gaat zien, dat is namelijk al 'de oplossing'

5. De oplossing

Nu presenteer je de gekozen oplossing. In een SAFe-context is dat meestal het Target Operating Model en het gekozen SAFe-ontwerp. Beschrijf hoe de organisatie wordt ingericht, welke Agile Release Trains er zijn, welke teams, welke governance (LPM laag) daarbij hoort en hoe rollen, verantwoordelijkheden en besluitvorming georganiseerd worden. Kan 1 slide zijn, maar kunnen ook 3 slides zijn. Zorg wel voor goede en logische verhaallijn volgorde.

6. Hoe het werkt

Leg daarna uit hoe het nieuwe systeem, de oplossing, in de praktijk functioneert. Hoe ontstaat er flow van idee naar waarde? Je kunt hier bijvoorbeeld kort uitleggen hoe epics, capabilities en features door het systeem bewegen, hoe teams samenwerken binnen een ART en hoe de CI/CD-pipeline helpt om sneller en betrouwbaarder software of diensten te leveren. Hoe gaan de gebruikers 'projecten aanvragen', hoe verloopt in je oplossing de resourcing? Financiering? Hoe wordt kwaliteit geborgd? Mijn advies is dat je hier een paar goede slides uit je TOM pakt.

7. Voordelen (optioneel)

Als de vorige slides de oplossing en de werking hebben uitgelegd, kun je op deze slide de concrete voordelen van de SAFe-transformatie samenvatten: snellere time-to-market, betere samenwerking tussen business en IT, hogere voorspelbaarheid van deliveries, meer medewerkerstevredenheid en een organisatie die continu leert en verbetert. Je kunt de voordelen ook mergen met ‘bereikte resultaten’ en de pitch daarmee korter houden.

8. Wat verandert er voor wie / impact

Een belangrijk onderdeel van de pitch is het vertalen van de verandering naar de dagelijkse praktijk. Wat betekent deze manier van werken voor verschillende rollen in de organisatie? Wat verandert er voor ontwikkelaars, product owners, managers of business owners? En minstens zo belangrijk: wat blijft hetzelfde? Dit helpt mensen zichzelf in de nieuwe situatie te herkennen.

9. Bereikte resultaten / bewijzen

In startup-pitches heet dit *traction*. In een transformatiecontext laat je hier zien wat er al bereikt is. In de verandermanagementliteratuur, bijvoorbeeld bij John Kotter, wordt dit ook wel het genereren van *quick wins* genoemd. Bijvoorbeeld een eerste succesvol draaiende Agile Release Train, kortere doorlooptijden, succesvolle POC’s, aantoonbaar betere samenwerking tussen teams of positieve feedback van gebruikers en stakeholders. Dit geeft vertrouwen dat de aanpak werkt.

10. De implementatie roadmap / ambitie

Na het bewijsmateriaal van wat al bereikt is, hoe de oplossing er YTD uitziet, is de volgende logische vraag: hoe ziet de weg vooruit eruit? Next steps? De implementatieroadmap geeft een helder overzicht van de komende fasen van de SAFe-transformatie. Per fase wordt duidelijk gemaakt welke mijlpalen worden nagestreefd, of er nog VSAIWs worden geïnitieerd, welke Agile Release Trains worden opgestart, of er nog SAFe levels bijkomen, welke trainingen en enablement-activiteiten plaatsvinden, of er nog nieuwe tooling komt, uitbreiding van de scope, en wat de daarbij verwachte doorlooptijden zijn.

De roadmap is daarbij geen blauwdruk in steen gebeiteld, maar een levend document dat meebeweegt met de realiteit van de organisatie. Inspectie en aanpassing zijn een onderdeel van de aanpak. Toch biedt de roadmap houvast: medewerkers en stakeholders weten wat er wanneer staat te gebeuren en wat er van hen verwacht wordt. Dat schept helderheid en vertrouwen in het verdere verloop van de transformatie. Die roadmap moet worden gecommuniceerd.

11. Communicatie en betrokkenheid

Hierna beschrijf je hoe medewerkers en stakeholders worden meegenomen in de transformatie. Hoe worden mensen geïnformeerd, betrokken en ondersteund? Denk aan informatiesessies, workshops, PI Plannings, communities of practice en andere vormen van samenwerking en kennisdeling.

Dan is het belangrijk uit te leggen hoe mensen gedurende de transformatie **op de hoogte worden gehouden** van de voortgang. Dat kan bijvoorbeeld via periodieke updates, interne nieuwsbrieven, dashboards, demo’s van opgeleverde resultaten, communicatiekalenders, townhalls of intranetpagina’s waar de laatste informatie over de transformatie wordt gedeeld. Zo blijft de organisatie niet alleen betrokken bij de start van de verandering, maar ook continu geïnformeerd over wat er gebeurt en welke resultaten worden geboekt.

12. Kritieke succesfactoren/risks/lessons learned

Een geloofwaardige transformatiepitch erkent niet alleen de kansen, maar ook de uitdagingen. Afhankelijk van de context kun je kiezen welke elementen je opneemt en of je ze combineert op één slide of uitwerkt over meerdere slides.

Kritieke succesfactoren beschrijven de randvoorwaarden voor een succesvolle transformatie. Denk aan zichtbaar leiderschap, voldoende mandaat voor het transformatieteam en de bereidheid van de organisatie om echt te veranderen, ook op managementniveau.

Risico's laten zien dat je de transformatie serieus neemt. Benoem bekende valkuilen zoals weerstand bij middenmanagement of het gevaar van SAFe implementeren als compliance-oefening in plaats van een echte cultuurverandering, en geef aan hoe je deze adresseert.

Lessons learned zijn waardevol als de organisatie al ervaring heeft opgedaan, bijvoorbeeld vanuit een pilot-ART. Wat werkte wel, wat niet? Wat kunnen andere divisies leren van jouw SAFe implementatie? Do’s en don'ts. Dit vergroot de geloofwaardigheid en laat zien dat de organisatie leert.

Niet elke pitch vraagt mogelijk om alle drie bovengenoemde aspecten, de spreker/auteur bepaalt zelf wat relevant is en hoeveel slides hij hiervoor gebruikt.

13. Het team (optioneel)

Introduceer vervolgens het transformatieteam dat verantwoordelijk is voor de verandering, bijvoorbeeld het LACE, de transformatieleider, Agile coaches en de executive sponsors. Leg kort uit waarom juist dit team de juiste ervaring, kennis en mandaat heeft om de transformatie te begeleiden. Het doel van deze slide is vertrouwen op te wekken: laat zien dat de verandering in goede handen is en dat er een deskundig team klaarstaat om de organisatie door deze transformatie te leiden.

14. Investering / financieel/ Business Case (optioneel)

Indien relevant kun je hier ook de totale investering benoemen die nodig is voor de totale transformatie tot aan de nieuwe stabiele sustainable Future State. Dat kan gaan over implementatie kosten, coaching, zaalhuur, trainingen, tooling zoals Jira Align of andere ondersteunende systemen.

Soms is hier ook een businesscase beschikbaar die de verwachte waarde van de transformatie beschrijft. Weet dat SAFe al een voorbeeld Business Case heeft gemaakt.

15. The Ask

Elke goede pitch eindigt met een duidelijke oproep tot actie. In een startup is dat de vraag om een funding, Seed Capital, Series A, Series B, etc. In een SAFe-transformatie is dat meestal een oproep aan de organisatie om actief mee te doen, ergens aan deel te nemen, feedback te geven.

The Ask kan verschillende vormen aannemen. Denk bijvoorbeeld aan het stellen van vragen, deelnemen aan workshops of informatiesessies, ideeën aandragen voor verbeteringen of actief betrokken raken bij de volgende stappen van de transformatie.

Het doel van deze slide is dat het publiek niet **alleen luistert naar het verhaal,** maar ook weet wat er nu van hen verwacht wordt en hoe zij concreet kunnen bijdragen aan het succes van de transformatie, in beweging komen. Anders vertel je een goed verhaal dat iedereen na 10 min weer is vergeten. Ze moeten ‘iets met je verhaal gaan doen’.

16. Contact / eindslide

Sluit af met een duidelijke slide waarop staat waar mensen terecht kunnen voor meer informatie én hoe zij vervolg kunnen geven aan *The Ask*. Vermeld contactgegevens, interne community-pagina's, documentatie of andere bronnen waar medewerkers en stakeholders zich verder kunnen verdiepen in de transformatie. Verwijs naar je Transformatie Brochure als je die al hebt.

Deze slide kan bijvoorbeeld tevens aangeven hoe mensen zich kunnen aanmelden voor een rol, kunnen deelnemen aan een pilot-ART, zich kunnen inschrijven voor workshops of zich kunnen aansluiten bij communities of practice. Zo wordt het niet alleen een informatieslide, maar ook een praktische vervolgstap voor

iedereen die actief wil bijdragen aan de transformatie.

Conclusie

Een succesvolle SAFe-transformatie vraagt niet alleen om een goed ontworpen operating model. Het vraagt ook om een goed verteld verhaal. Net zoals startups investeerders moeten overtuigen van hun idee, moeten transformatieleiders en executive management medewerkers overtuigen van de waarde van verandering.

Wie het verhaal helder, eenvoudig en inspirerend kan vertellen, vergroot de kans dat mensen niet alleen begrijpen wat er gebeurt, maar er ook actief onderdeel van willen worden.

En precies daar begint een succesvolle transformatie: bij een verhaal dat mensen niet alleen begrijpen, maar dat hen ook aanspreekt en in beweging zet.

22. Transformeren naar SAFe: wat betekent dit voor klanten, gebruikers en toekomstige Business Owners?

Wanneer een organisatie overstapt van projectmatig werken naar SAFe®, verandert voor klanten en gebruikers hoe oplossingen worden ontwikkeld, hoe zij daarbij betrokken worden en hoe resultaten worden opgeleverd. SAFe® experts kunnen SAFe® niet implementeren zonder klanten daarin mee te nemen en actief op te nemen in bijvoorbeeld het nieuwe Target Operating Model.

Dit hoofdstuk geeft antwoord op de veel aan mij gestelde vraag: we gaan SAFe® implementeren, wat betekent dit voor onze huidige gebruikers?

In het kort

• Gebruikers zien elke iteratie werkende resultaten en geven direct feedback.

• Met actuele, toegankelijke dashboards kunnen klanten en gebruikers op ieder moment inzicht krijgen in status en voortgang.

• Er zijn vaste, frequente contactmomenten met de ART en LPM.

• Verantwoordelijkheden die eerder bij een projectmanager lagen, worden binnen SAFe® over meerdere rollen verdeeld.

• Sommige vertegenwoordigers van de business of klantorganisatie krijgen de formele rol van Business Owner. Die rol vraagt verantwoordelijkheid en voldoende beslissingsbevoegdheid. Een eventuele vertegenwoordiger moet daarvoor expliciet mandaat hebben.

• De Business Owner is verantwoordelijk voor Business Benefits Realisatie: het omzetten van geleverde output in daadwerkelijke outcomes.

1.1. De impact voor de huidige klanten en gebruikers

1.1.1. Betrokkenheid van de gebruikers

Gebruikersbetrokkenheid ('Customer Engagement') is geïntegreerd in het SAFe®-framework. UX/UI-experts passen ontwerpmethodieken toe zoals Design Thinking, Empathy Mapping, Story Mapping, Customer Journeys, Gemba Walks en Rapid Prototyping. Zo zorgen ze ervoor dat elke ART levert wat het beste aansluit bij de wensen van de gebruikers. UX/UI-experts zijn daarom belangrijke teamleden met directe invloed op de gebruikerservaring en daarmee op de gebruikerstevredenheid (NPS score).

Wat betekent dit voor gebruikers? Gebruikers worden actief betrokken bij het ontwerp. Ze worden uitgenodigd voor interviews, observaties op de werkvloer (Gemba Walks) en het testen van prototypes. Hun ervaring in de praktijk bepaalt mede wat er gebouwd wordt, in plaats van alleen een vooraf vastgelegd eisenpakket. Dat vraagt wel tijd en beschikbaarheid van gebruikers.

1.1.2. Werkende resultaten beoordelen en samen bepalen wanneer ze in gebruik gaan

In SAFe® wordt in korte iteraties gewerkt, die bij veel organisaties twee weken duren. Tijdens reviews en demo’s bekijken gebruikers werkende resultaten en geven zij feedback. Dat helpt vroeg te ontdekken of de oplossing aansluit bij hun behoeften. Een demo is echter nog geen ingebruikname: nieuwe functionaliteit komt niet automatisch na iedere iteratie in productie. Product Management en Business Owners stemmen het releasemoment af op de klantbehoefte, de gereedheid van de oplossing en de gebruikersorganisatie.

Wat betekent dit voor gebruikers? Gebruikers hoeven niet maanden of jaren te wachten op één eindoplevering. Tijdens reviews en demo’s is er iets concreets te zien en te beoordelen. Feedback wordt beoordeeld en waar nodig geprioriteerd voor volgende iteraties. Gebruikers weten zo eerder wat eraan komt en

kunnen samen met hun vertegenwoordigers afspreken wanneer nieuwe functionaliteit daadwerkelijk in gebruik gaat.

1.1.3. Minder papier, meer focus op werkende oplossingen

Agilisten richten zich op werkende oplossingen (*Working Solutions*), conform waarde 2 van het Agile Manifesto en SAFe® Principle #5. Documentatie is in principe Lean: denk aan een Lean Business Case van twee A4'tjes of een Epic Hypothesis Statement van één A4. Klanten van wie de leverancier met SAFe® werkt, krijgen zodoende veel minder papier.

Wat betekent dit voor gebruikers? Gebruikers hoeven minder lijvige documenten te lezen, te reviewen en te accorderen. In plaats daarvan beoordelen ze het werkende resultaat zelf of lean documentatie zoals een Epic Hypothese Statement. Hun tijd gaat dus naar het product, niet naar papier.

1.1.4. Ook laat in het traject wijzigingen kunnen doorvoeren

Agilisten verwelkomen veranderingen, zelfs laat in de ontwikkeling, conform principe 2 van het Agile Manifesto. Nieuwe of gewijzigde wensen worden via de backlog beoordeeld en geprioriteerd. Dat is iets anders dan toestemming voor productie-invoering: afhankelijk van risico's, regelgeving en organisatieafspraken kunnen daarvoor nog goedkeuringen nodig zijn, bijvoorbeeld via een Change Advisory Board (CAB). Richt die besluitvorming zo in dat noodzakelijke beheersing en snelle levering samengaan. Verandering wordt gezien als normaal en zelfs als wenselijk. Agile teams passen zich snel aan nieuwe eisen en feedback aan, waardoor het product beter aansluit bij de behoeften van de klant.

Wat betekent dit voor gebruikers? Gebruikers kunnen nieuwe of gewijzigde wensen op elk moment inbrengen bij de Product Owner of Product Manager. De wens komt op de backlog en wordt naar waarde geprioriteerd. Wel geldt: iets nieuws erbij betekent vaak dat iets anders later komt. Die afweging wordt samen en transparant gemaakt. Met elkaar bepalen ze het optimum, want 100 wijzigingen per dag is niet werkbaar.

1.1.5. Agilisten werken met metrics

SAFe® meet op drie domeinen (Measure and Grow):

1. **Competency**: hoe volwassen is de organisatie in het toepassen van SAFe®?

2. **Outcomes**: worden de beoogde resultaten behaald, bijvoorbeeld vastgelegd in OKR's?

3. **Flow**: hoe efficiënt stroomt waarde door het systeem? Met de Flow Metrics Distribution, Velocity, Time, Load, Efficiency en Predictability.

Daarnaast gebruiken teams vaak burndown charts en value-stream-metingen zoals %C&A en Activity Ratio. Bij PRINCE2 is werken met dit soort metrics veel minder gebruikelijk.

Wat betekent dit voor gebruikers? Gebruikers kunnen op basis van feiten zien hoe het ervoor staat: hoe lang het duurt voordat een wens is opgeleverd (Flow Time), hoe betrouwbaar toezeggingen worden nagekomen (Flow Predictability) en of de beoogde resultaten daadwerkelijk worden gehaald (Outcomes). Gesprekken over voortgang gaan daardoor over cijfers in plaats van over gevoel.

1.1.6. Andere rapportage: actueel inzicht in de status

Door werk en voortgang in de Agile tooling vast te leggen, kan de organisatie rapportages en dashboards beschikbaar maken. Veel tools ondersteunen dit met standaardrapportages. BI-dashboards kunnen informatie uit meerdere tools samenbrengen. Hoe actueel en volledig het inzicht is, hangt af van de vastlegging, koppelingen, verversingsfrequentie en toegangsrechten. Continu toegankelijk inzicht is daarmee een inrichtingskeuze, geen automatische garantie van SAFe®.

Wat betekent dit voor gebruikers? Gebruikers hoeven niet te wachten op een periodieke (Maandelijkse?) stuurgroep voortgangsrapportage. Mits goed ingericht, kijken ze zelf in een dashboard waar hun wens staat en

wanneer die naar verwachting wordt opgeleverd.

1.1.7. Verhoogde transparantie

In mijn ervaring bij vele Prince2 en MSP programma's werd op MT-niveau vaak alles groen gerapporteerd via 'stoplichtrapportages', terwijl het bij de engineers in de kelders van die organisaties zaken allang op rood stonden. SAFe® kent daarentegen een hoge mate van transparantie en eerlijkheid. Gebruikers en andere stakeholders kunnen de voortgang van de ART volgen via burndown charts, backlogs, de Team Sync en de andere events. De organisatie kan gebruikers toegang geven tot relevante backlogs, Kanban-borden en dashboards. Daarbij worden afspraken gemaakt over toegankelijkheid, vertrouwelijkheid en de toelichting die nodig is om de informatie goed te begrijpen.

Wat betekent dit voor gebruikers? Gebruikers krijgen een eerlijk beeld, ook als het tegenzit. Problemen komen eerder op tafel, zodat zij tijdig kunnen meedenken over keuzes en hun eigen planning kunnen bijstellen.

1.1.8. Verantwoordelijkheden van de projectmanager worden over meerdere rollen verdeeld

Gebruikers spreken nu vaak met de projectmanager over verschillende aspecten van het project. Die projectmanager is een alleskunner! Die snapt de inhoud, snapt de toegepaste methodologie en is ook nog expert op gebied van architectuur. Immers, in een stuurgroep zit vaak de PM als enige aan tafel. Binnen SAFe® worden verantwoordelijkheden die eerder bij een projectmanager lagen, over meerdere rollen verdeeld. Dat geeft meer rust en bevordert de opbouw van kennis en expertise. Op ART-niveau zijn Product Management, System Architect en Release Train Engineer (RTE) belangrijke aanspreekpunten.

De onderstaande tabel beschrijft hun aandachtsgebieden:

Rol	Weet alles van	Belangrijk voor gebruikers
Product Manager(s)	De requirements	Hoofdcontactpersoon voor gebruikers, met als doel *'fit for purpose'*
System Architect(en)	De toegepaste technologie	Bewaakt dat de oplossing technisch klopt en toekomstvast is
Release Train Engineer (RTE)	Het SAFe®-proces, de events, metrics, continue verbeteren, verhogen van de flow of value	Per ART is er altijd één RTE

Wat betekent dit voor gebruikers? Gebruikers hebben niet langer één aanspreekpunt voor alles. Voor wensen en prioriteiten gaan ze naar de Product Manager (of op teamniveau de Product Owner), voor technische vragen naar de System Architect en voor proces, planning en events naar de RTE. Het helpt om bij de start een overzicht van deze contactpersonen met gebruikers te delen.

1.1.9. Taal en begrippen veranderen

Net zoals bijvoorbeeld ASML en Defensiemedewerkers hun eigen taal en afkortingen hebben (en geloof me: die kunnen er ook wat van !!) die je als buitenstaander moet leren, hebben agilisten ook hun eigen vocabulaire. Omdat gebruikers maar met een deel van SAFe® in aanraking komen, leert de praktijk dat ze die taal snel oppikken. Zie ook hoofdstuk 10, 'Het SAFe-jargon ontcijferd: ~30 woordjes om presentaties en workshops beter te begrijpen'.

Wat betekent dit voor gebruikers? Gebruikers hoeven niet het hele SAFe®-vocabulaire te beheersen. Een handvol begrippen, zoals iteratie, PI event, backlog, I&A, Feature en System Demo, is voldoende om goed mee te kunnen praten. Een korte micro training, begrippenlijst of introductiesessie bij de start versnelt dit.

1.1.10. Meer en vaste contactmomenten

Gebruikers hebben veel mogelijkheden om regelmatig met de ART te spreken en hun wensen door te geven. Hieronder staan de events waarin zij participeren in volgorde, elk met een korte toelichting.

Event	Frequentie	Wat betekent het voor gebruikers?
PI Planning	1x per PI (meestal 8-12 weken)	Elk Planning Interval (PI) start met een Big Room Planning. Daar wordt het werk voor de komende PI gepland. Gebruikers of hun vertegenwoordigers doen mee, helpen functionaliteiten te prioriteren, geven feedback op de plannen en keuren ze plenair goed.
Backlog Refinement	Regelmatig, bijvoorbeeld wekelijks; volgens de afspraken van de ART en teams	Gebruikers, Product Managers, key users of SME's die alle belangen vertegenwoordigen, stellen prioriteiten, verduidelijken eisen en zorgen dat de backlog aansluit op de klantbehoeften.
Bilaterale gesprekken met content experts/SME's	Op verzoek	Gebruikers kunnen altijd met de Product Owners, de Product Managers of de LPM Epic Owners spreken om hun wensen kenbaar te maken.
Team Sync	Dagelijks	Zorgt dat de teams efficiënt werken en snel inspelen op veranderingen of problemen die de klant raken. Een organisatie kan met het team afspreken dat gebruikers als toehoorder aansluiten, zonder de afstemming tussen teamleden te verstoren. Toegang en deelname worden vooraf afgestemd; transparantie betekent niet dat ieder overleg automatisch voor iedereen openstaat.
Iteration Review	1x per iteratie	Gebruikers bekijken de werkende resultaten en geven direct feedback. Door dit elke iteratie te doen, ontstaat telkens een nieuw moment om afwijkingen tussen de behoefte en het resultaat te signaleren en bij te sturen. Bij actieve deelname en tijdige opvolging hoeft een misverstand zo niet langer dan één iteratie onbesproken te blijven. Dit is geen garantie dat iedere afwijking binnen één iteratie wordt opgelost.
System Demo	1x per iteratie	Gebruikers zien *het volledig geïntegreerde product*, geven feedback en evalueren de voortgang van de hele ART.
Inspect & Adapt	1x per PI	Gebruikers delen hun kijk op de voortgang en het resultaat en helpen verbeterpunten te vinden. Ook hier geldt eerlijkheid en transparantie: rood is rood en groen is groen.

Wat betekent dit voor gebruikers? Gebruikers hebben op vaste, voorspelbare momenten invloed en continue alignment, een van de kernwaarden van SAFe®. Ze weten vooraf wanneer ze wensen kunnen inbrengen (PI Planning, Backlog Refinement) en wanneer ze resultaten zien (Iteration Review, System Demo). Dat vraagt wel dat zij of hun vertegenwoordigers deze momenten structureel in hun agenda reserveren. Oude legacy ceremonies dienen te worden gestopt: stuurgroepen, SPA-dagen en afdelingsoverleggen bestaan in SAFe® niet meer.

1.1.11. De gebruikersorganisatie moet veranderingen kunnen opnemen

Vaker werkende resultaten leveren vraagt ook voorbereiding aan de kant van de gebruikersorganisatie. Denk aan opleiding, aangepaste werkinstructies, communicatie, ondersteuning en beheer. Soms is een gefaseerde invoering nodig, bijvoorbeeld omdat niet alle locaties of gebruikersgroepen tegelijk kunnen overstappen. Ontwikkelcapaciteit alleen is dus niet voldoende: ook de capaciteit om veranderingen in gebruik te nemen moet worden georganiseerd.

Wat betekent dit voor gebruikers? Maak tijd vrij om nieuwe functionaliteit te leren gebruiken en spreek af wie instructies, opleiding en ondersteuning verzorgt. Betrek key users en beheer vroeg en stem de invoering af op de dagelijkse operatie. Zo wordt een technisch gereed resultaat ook daadwerkelijk bruikbaar in de praktijk.

1.2. De impact voor toekomstige Business Owners

Sommige vertegenwoordigers van de business of klantorganisatie krijgen de formele rol van **Business Owner**. Die rol vraagt verantwoordelijkheid en voldoende beslissingsbevoegdheid als representant van een grotere groep. Een eventuele vertegenwoordiger moet expliciet mandaat hebben om de bijbehorende besluiten te nemen. Business Owners zijn een kleine groep stakeholders die eindverantwoordelijk is voor de bedrijfswaarde die een ART levert.

Voor wie die rol gaat vervullen, verandert het volgende:

• **Een actieve rol in de PI Planning.** Business Owners lichten de bedrijfscontext toe, bespreken de conceptplannen met de teams en kennen aan elk PI Objective een business value toe (1-10).

• **Eigenaarschap van resultaten.** Ze bewaken of de ART de afgesproken PI Objectives haalt en beoordelen aan het einde van de PI de werkelijk geleverde waarde. Dat vormt de basis voor de predictability-meting.

• **Aanwezig bij System Demo en Inspect & Adapt.** Ze geven feedback op het geïntegreerde resultaat en denken mee over verbeteringen.

• **Meesturen op prioriteiten en knopen doorhakken.** Samen met Product Management en Lean Portfolio Management bepalen ze mee waar capaciteit en budget naartoe gaan. Wanneer belangen, wensen en beschikbare capaciteit botsen, helpen Business Owners keuzes te maken over waarde, risico’s en haalbaarheid. Ze nemen binnen hun mandaat besluiten of zorgen dat een besluit tijdig op het juiste niveau wordt genomen. Hun verantwoordelijkheid gaat daarmee verder dan deelnemen en feedback geven.

• **Een andere houding.** Van opdrachtgever die aan het eind accepteert naar agile teamlid die elke iteratie meestuurt. Dat vraagt tijd en commitment: plan de PI-events daarom structureel in de agenda.

• **SAFe kennis en begripsvorming:** dit alles vraagt tijd en energie om het SAFe Framework te begrijpen, de taal te leren en de details, taken en bevoegdheden van die rol.

Wat betekent dit voor Business Owners? Business Owners krijgen directe zeggenschap over wat de ART bouwt en worden aangesproken op de waarde die dat oplevert. Daar staat tegenover dat zij tijd en mandaat moeten vrijmaken: ze zijn structureel aanwezig bij PI Planning, System Demo en Inspect & Adapt, en nemen daar namens hun achterban besluiten.

1.3. Business Benefits Realisatie: van output naar outcome

Business Owners zijn degenen die baat hebben bij het eindresultaat. LPM en de ART leveren alleen de output: werkende oplossingen, Epics, Features en releases. Output is echter geen doel op zich. Waarde ontstaat pas wanneer de business die output in de eigen operatie gaat gebruiken en implementeren. Pas dan worden de outcomes gerealiseerd die in het Epic Hypothesis Statement zijn vastgelegd. Dat omzetten van output in outcome heet Business Benefits Realisatie, en dat is de verantwoordelijkheid van de Business Owner.

De SAFe®-fabriek kan haar werk nog zo goed doen: lukt het de Business Owner niet om de output om te zetten in outcome, dan geldt *operatie geslaagd, patiënt overleden*.

Wat Business Benefits Realisatie concreet van de Business Owner vraagt:

• **Outcomes vertalen naar de eigen operatie.** Welke werkprocessen, rollen en afspraken veranderen er, en per wanneer?

• **Ingebruikname organiseren.** Zorg voor opleiding, werkinstructies, key users, ondersteuning en beheer (zie 1.1.11). Een technisch gereed resultaat levert pas waarde op als het ook echt wordt gebruikt.

• **De oude werkwijze afbouwen.** Zolang oude systemen, procedures en overleggen blijven bestaan, ontstaat dubbel werk en blijft de benefit uit.

• **Leading indicators meten en terugkoppelen.** De Business Owner levert de feiten aan die Epic Owners en LPM nodig hebben om te bepalen of de hypothese klopt (persevere) of dat bijsturen nodig is (pivot).

• **Benefits blijven volgen na oplevering.** Waarde ontstaat vaak pas maanden na ingebruikname. Benefits realisatie stopt dus niet bij de System Demo.

Wat betekent dit voor Business Owners? Business Owners worden niet afgerekend op wat er gebouwd is, maar op wat het oplevert. Zij zijn de schakel tussen de SAFe®-fabriek en de operatie. Zonder hun actieve inzet blijft zelfs de beste output onbenut.

1.4. Kortom

SAFe® biedt eindgebruikers en Business Owners aanzienlijke voordelen door de overstap naar Agile werkwijzen, met een focus op continue verbetering en gebruikersgerichtheid.

Ze profiteren van:

1. een kortere time-to-field;

2. minder fouten;

3. een hogere leverbetrouwbaarheid;

4. betere afstemming;

5. meer transparantie en meer feedbackmomenten;

6. beter inzicht in voortgang en resultaten, mits de metrics actueel, betrouwbaar en toegankelijk zijn.

Meer weten over de rol van Business Owner? Voor (toekomstige) Business Owners zijn er speciale eendaagse trainingen. Daarnaast heb ik een uitgebreide, theoretisch correcte rolbeschrijving van de Business Owner beschikbaar. Wil je die ontvangen? Stuur me dan een mail via safe@majormilestones.nl of een bericht.

Vragen na het lezen van dit hoofdstuk? Neem gerust contact met me op. Ik ga graag met je in gesprek, of het nu gaat om een toelichting, overleg of een stevige discussie.

23. Transformeren naar SAFe: wat verandert er voor werknemers?

Stel: jouw organisatie stapt over van een traditionele, projectgestuurde (PRINCE2, MSP, PPM) werkwijze naar SAFe® 6.0 (Scaled Agile Framework). Wat betekent dat voor jou als werknemer en toekomstig lid van een Agile Team?

Het antwoord hangt sterk af van het startpunt. Werkt je organisatie al een beetje agile en worden de ceremonies al min of meer gevolgd, dan valt de verandering mee. Begin je vanaf nul, dan verandert er flink wat.

SAFe® is overigens veel meer dan een andere werkwijze voor teams. Ook de manier waarop de organisatie wordt ingericht, bestuurd, gefinancierd en aangestuurd verandert; dat ga jij deels ook merken.

In dit hoofdstuk zet ik de belangrijkste en meest voorkomende veranderingen voor jou als toekomstig agilist op een rij, zoals die in de meeste SAFe®-transformaties naar voren komen en ik in mijn eigen transformatiepraktijk heb meegemaakt.

De impact voor werknemers

1. De werkwijze wordt iteratief/incrementeel

In plaats van te werken aan één groot project met een begin- en einddatum, zullen teams in SAFe® werken in korte iteraties (meestal van twee weken), in SAFe® 'Iterations' genoemd.

Elke iteratie is gericht op het opleveren van werkende, geteste onderdelen van een product of oplossing. Het doel is om regelmatig waarde op te leveren en snel feedback te krijgen, in plaats van pas aan het einde van een langdurig project te ontdekken of het resultaat voldoet.

Voor werknemers betekent dit dat je werk in kleinere stukken opdeelt, sneller resultaat ziet en elke twee weken feedback krijgt. Je werkt minder lang naar één verre deadline toe, maar levert wel vaker iets op dat écht af moet zijn.

2. Rust en regelmaat, alles in cadence. Minder ad hoc

Agilisten werken in een Cadence, volgens een bepaalde Heartbeat, een werkritme: elke ochtend een Team Sync van ongeveer 15 minuten, aan het einde van elke iteratie de Iteration Review en de Iteration Retrospective, waarin het team kritisch naar de eigen werkwijze kijkt en concrete verbeteracties afspreekt.

Om energie te besparen en ad-hocgedrag te vermijden, zijn er (bijna) altijd (1) dezelfde mensen, (2) op dezelfde locatie of via hetzelfde communicatiekanaal, (3) op dezelfde tijden en (4) met dezelfde agenda.

SAFe® heeft goed gekeken welke vaste meetings nodig zijn om samenwerking, planning en verbetering te ondersteunen. Aanvullend overleg kan uiteraard nodig blijven, maar het uitgangspunt is om onnodige en ad-hocmeetings zo veel mogelijk te vermijden. Veel terugkerende meetings kunnen daarom voor het hele jaar worden ingepland. Deze aanpak biedt rust, regelmaat en voorspelbaarheid.

Dit heeft wel impact op bestaande overleggen: die moeten kritisch worden bekeken en waar mogelijk worden gestopt, samengevoegd of getransformeerd. Ook de deelnemers aan die overleggen worden kritisch bekeken en mogelijk gaan veranderen.

Voor werknemers betekent dit een voorspelbare agenda: je weet ruim vooruit welke vaste meetings er zijn en wanneer. Tegelijk verdwijnen mogelijk overleggen waar je aan gewend was, en word je niet meer automatisch voor ieder overleg uitgenodigd.

3. Maximaliseren 'Time in the Zone' voor alle teamleden

Team members zijn 'knowledge workers' en in het algemeen wars van onnodige meetings. Daarom is het maximaliseren van de 'Time in the Zone' een belangrijk aandachtspunt bij een Lean-Agile transformatie. Dit verwijst naar de tijd waarin professionals ononderbroken en geconcentreerd kunnen werken aan hun taken, zonder afleidingen of onderbrekingen. Dit aandachtspunt benadrukt het belang van gefocuste, productieve werkperiodes om de kwaliteit en snelheid van leveringen te verbeteren. Iets wat vrijwel elke professional fijn vindt en waardeert! Dit betekent onder andere dat bestaande overleggen mogelijk worden verplaatst naar andere begintijden of worden gecombineerd, zodat grotere aaneengesloten blokken voor geconcentreerd werken ontstaan.

In veel traditionele organisaties hebben werknemers enorm last van interrupties uit alle hoeken en gaten. Manager X wil nú Y hebben en verwacht dat je alles uit je handen laat vallen. In SAFe® werkt dat anders: teams worden zoveel mogelijk afgeschermd van die 'schreeuwende buitenwereld'. Werk komt niet via de zijkant binnen, maar gereguleerd via de backlogs. Grotere initiatieven lopen via Lean Portfolio Management (LPM) en de ART, van Portfolio Backlog via ART Backlog naar Team Backlog. Daarnaast ontstaat werk rechtstreeks vanuit klanten, gebruikers, onderhoud en operationele incidenten. Ook dat komt via de Team Backlog of de daarvoor afgesproken capaciteit binnen, niet via de achterdeur. De Product Owner en de Scrum Master/Team Coach bewaken dit. Is er écht iets urgents, dan loopt dat via de Product Owner, die het afweegt tegen het lopende werk en het Iteration Goal. Iets nieuws erin betekent dan bewust iets anders eruit.

Voor werknemers betekent dit meer ruimte voor geconcentreerd werk en minder versnipperde dagen. Het vraagt ook iets van jou: zelf kritisch zijn op welke overleggen echt nodig zijn en je focusblokken bewaken.

4. Meer autonomie en verbeterde, doordachte samenwerking

• **Autonomie:** In SAFe® krijgen teams meer ruimte voor zelforganisatie. Ze beslissen binnen afgesproken kaders zelf hoe ze hun werk aanpakken en verdelen, in plaats van voortdurend gedetailleerde opdrachten te ontvangen van een projectmanager.

• **Werknemers schatten zelf in:** In traditionele projecten worden doorlooptijden soms al vastgesteld voordat de uitvoerende professionals betrokken zijn. In SAFe® spelen de teamleden zelf een centrale rol bij het inschatten en plannen van hun werkzaamheden. Het team maakt gezamenlijk afspraken over wat haalbaar is.

• **Samenwerking met andere teams:** Teams werken minder in isolatie en hebben gestructureerde afstemming met andere teams. Dit wordt georganiseerd binnen Agile Release Trains (ART's), waarbij meerdere teams samenwerken aan een gedeelde waardestroom en tijdens PI Planning afhankelijkheden identificeren en bespreken.

5. In SAFe® is groen = groen en rood = rood. En dat kan iedereen zien.

• **Dagelijkse Team Sync:** Elke dag houdt het team een korte bijeenkomst waarin voortgang, afstemming en obstakels centraal staan. Andere stakeholders kunnen waar passend meekijken, zonder de zelforganisatie van het team te verstoren.

• **Meer zichtbaarheid van opleveringen:** In tegenstelling tot traditionele projecten, waar resultaten soms pas laat zichtbaar worden, stimuleert SAFe® het regelmatig demonstreren van werkende oplossingen. Hierdoor worden problemen eerder zichtbaar en kunnen ze sneller worden aangepakt. SAFe® legt daarbij de nadruk op vroegtijdige validatie met klanten en gebruikers, bijvoorbeeld via MVP's (Minimum Viable Products) en het incrementeel opleveren van waarde.

Voor werknemers betekent dit dat voortgang en problemen voor iedereen zichtbaar zijn. Een probleem 'groen rapporteren' werkt niet meer; je wordt juist gestimuleerd obstakels vroeg te melden, zodat het team ze samen kan oplossen.

6. SAFe®-experts prioriteren economisch. Eerlijk en transparant.

Werk wordt in SAFe® waar passend economisch geprioriteerd met behulp van WSJF (Weighted Shortest Job First), onder andere voor Features, Capabilities en Epics. Daarbij worden de relatieve kosten van uitstel afgezet tegen de omvang van het werk. Zo ontstaat een onderbouwde volgorde, in plaats van HSE: Hardste Schreeuwers Eerst. Niet alle prioritering gebeurt door dezelfde persoon.

Product Management is verantwoordelijk voor de prioritering van de ART Backlog en de Product Owner voor de Team Backlog, binnen de kaders van de gezamenlijke doelen en prioriteiten.

Voor werknemers betekent dit dat veel duidelijker wordt waarom bepaald werk voorrang krijgt boven ander werk. Prioriteiten kunnen veranderen wanneer nieuwe inzichten ontstaan, maar wijzigingen vragen om goede afstemming.

7. Verdeling van capaciteit: Capacity Allocation (DevOps)

SAFe® kent het mechanisme Capacity Allocation. Daarmee kan bewust capaciteit worden gereserveerd voor verschillende soorten werk, zodat niet alle beschikbare capaciteit automatisch naar nieuwe functionaliteit gaat.

Een organisatie kan bijvoorbeeld onderscheid maken tussen drie soorten werkzaamheden:

1. Ontwikkeling van nieuwe functies en diensten.

2. Beheer en onderhoud.

3. Continu verbeteren.

Dit is een mogelijke praktische inrichting; SAFe® schrijft deze drie categorieën niet verplicht voor.

Reguliere lijnactiviteiten, zoals bilaterale overleggen, werkoverleg en HR-gesprekken, vallen hier niet onder. Die tijd gaat eerst af van de beschikbare capaciteit; Capacity Allocation verdeelt vervolgens wat er overblijft voor het teamwerk.

Tijdens de planning wordt rekening gehouden met de beschikbare capaciteit en de afgesproken verdeling. Teams kunnen daarbij gebruikmaken van Story Points en historische gegevens over hun gerealiseerde werk. Zo wordt voorkomen dat beheer, onderhoud en verbetering voortdurend worden verdrongen door nieuwe verzoeken.

Voor werknemers betekent dit dat er structureel tijd is voor beheer, onderhoud en verbetering, en niet alleen voor nieuwe wensen. Ook je lijntaken tellen mee in de beschikbare capaciteit, waardoor duidelijker wordt hoeveel tijd je echt overhoudt voor teamwerk.

8. Andere rollen en verantwoordelijkheden

Verandering in rollen: Teamleden kunnen een verschuiving in hun verantwoordelijkheden en de reikwijdte van hun werkzaamheden ervaren.

SAFe® onderscheidt specifieke rollen, waaronder Scrum Master/Team Coach (SM/TC), Release Train Engineer (RTE), Product Owner (PO), Product Management en verschillende rollen binnen Lean Portfolio Management (LPM). Ook kunnen bij grotere oplossingen Solution Train Engineers (STE's) worden ingezet. De rollen hebben eigen verantwoordelijkheden en onderlinge samenwerkingsafspraken. Dit betekent niet dat iedere organisatie automatisch alle mogelijke SAFe®-rollen nodig heeft.

Voor werknemers betekent dit dat je taakinvulling kan veranderen. Misschien krijg je een nieuwe rol, of werk je voortaan samen met rollen die je nog niet kende, zoals een Scrum Master/Team Coach of Product Owner. Bespreek tijdig met je leidinggevende wat dit voor jou betekent.

9. 'Responding to Change' over 'Following a Plan'

In plaats van uitsluitend te werken volgens een vooraf volledig vastgelegd plan, moeten teamleden voortdurend rekening houden met nieuwe inzichten, klantfeedback en veranderende omstandigheden. SAFe® moedigt aan om plannen aan te passen wanneer dat economisch of inhoudelijk verstandiger is.

Dit betekent overigens niet dat de Product Owner morgen zomaar eenzijdig de gehele Iteration Backlog kan veranderen. Het team en de PO stemmen wijzigingen samen af, met respect voor het Iteration Goal en de gemaakte afspraken.

Voor werknemers betekent dit dat veranderingen normaal worden, maar niet willekeurig of ongecontroleerd plaatsvinden.

10. SAFe® helpt overbelasting en burn-out voorkomen

In traditionele projectorganisaties wordt werk regelmatig van bovenaf opgedragen. Management kan deadlines en werklast opleggen, waardoor teams soms meer werk krijgen dan ze feitelijk aankunnen.

In SAFe® wordt nadrukkelijk rekening gehouden met de werkelijke capaciteit van teams. Een belangrijk uitgangspunt is dat teams niet onbeperkt werk moeten aannemen. Door capaciteit inzichtelijk te maken, werk te prioriteren en gebruik te maken van pull-principes ontstaat een betere balans tussen vraag en capaciteit. Teams kunnen bijvoorbeeld bewust een buffer aanhouden voor onvoorziene omstandigheden.

SAFe® garandeert uiteraard niet dat medewerkers nooit overbelast raken, maar biedt mechanismen om structurele overbelasting zichtbaar te maken en te verminderen.

Voor werknemers betekent dit dat je meer invloed krijgt op hoeveel werk je team oppakt. Je mag, en moet, aangeven wanneer de capaciteit op is, in plaats van stilzwijgend meer werk aan te nemen.

11. SAFe® helpt informatie-isolatie voorkomen

Alignment, of afstemming, is een essentieel principe binnen SAFe®. Door een sterke, gestructureerde focus op communicatie en het delen van visie, roadmaps, strategische doelstellingen en werkvoorraden wordt het voor medewerkers gemakkelijker om te begrijpen waar de organisatie naartoe wil en waaraan andere teams werken.

SAFe® ondersteunt ook het gebruik van OKR's (Objectives and Key Results), waarmee strategische doelen worden vertaald naar concrete en meetbare resultaten. Werknemers krijgen daardoor beter inzicht in de bijdrage van hun eigen werkzaamheden aan de grotere organisatiedoelen. Het risico op informatie-isolatie wordt hiermee kleiner, mits de organisatie deze transparantie ook daadwerkelijk toepast.

12. Transparantie en informatiedeling: andere tooling, minder MS Office

SAFe® bevordert een gemeenschappelijk begrip door relevante informatie zichtbaar en toegankelijk te maken. Veel organisaties gebruiken hiervoor Jira, Azure DevOps of vergelijkbare agile tooling. Medewerkers en managers kunnen, binnen de geldende autorisaties, inzicht krijgen in backlogs, prioriteiten, afhankelijkheden en voortgang.

Dat betekent een belangrijke verandering voor organisaties waarin informatie traditioneel verspreid staat over Word-documenten, Excel-bestanden, PowerPoint-presentaties en e-mails. Het streven is een zo veel mogelijk geïntegreerde en transparante informatievoorziening: een 'Single Source of Truth'.

Voor werknemers betekent dit minder zoeken naar de laatste versie van een document en meer mogelijkheden om zelfstandig informatie te vinden.

13. Datagedreven werken: metrics en statistieken

Een van de belangrijke onderdelen van SAFe® is het gebruik van meetgegevens om prestaties, kwaliteit en voortgang te beoordelen. SAFe® onderscheidt onder andere Flow Metrics, zoals Flow Distribution, Flow Efficiency, Flow Load, Flow Time, Flow Velocity en Flow Predictability. Flow Distribution laat zien hoe het werk is verdeeld over verschillende soorten werk, zoals nieuwe functionaliteit en enablers. Daarnaast worden onder meer resultaten, gerealiseerde waarde, kwaliteit en verbetervermogen gemeten.

Met deze gegevens kunnen teams en ART's knelpunten identificeren en structureel verbeteren. Belangrijk is dat metrics primair worden gebruikt om het systeem te verbeteren en niet om individuele medewerkers met elkaar te vergelijken of af te rekenen. SAFe®-experts meten dus hoe efficiënt en effectief de 'SAFe®-fabriek' functioneert.

Voor werknemers betekent dit dat je werk zichtbaar en meetbaar wordt, bijvoorbeeld in doorlooptijd en voorspelbaarheid. Het doel is niet jou af te rekenen, maar samen te zien waar het systeem hapert en hoe dat beter kan.

14. Opleiding, leren en continu verbeteren

De transitie naar SAFe® betekent dat medewerkers zich moeten inwerken in een nieuwe manier van werken. Om effectief te kunnen functioneren in een SAFe®-omgeving moeten medewerkers voldoende kennis hebben van de principes, rollen, werkwijzen en afspraken die voor hun werkzaamheden relevant zijn.

Niet iedereen hoeft SAFe® volledig te beheersen. Voor Agile Teamleden zijn andere vaardigheden en kennis nodig dan voor bijvoorbeeld Product Owners, Scrum Masters, RTE's of leidinggevenden. In de praktijk ondersteunen Scrum Masters/Team Coaches, RTE's, SPC's en Agile Coaches de betrokken medewerkers bij het leren en toepassen van de nieuwe werkwijze. Dat kost tijd en energie, maar voorkomt veel misverstanden en onnodige frustraties.

Continu leren en verbeteren: Het leren stopt niet na de eerste training. Er komt meer nadruk op voortdurend verbeteren, experimenteren en het ontwikkelen van vaardigheden.

SAFe® kent hiervoor onder andere de Innovation and Planning (IP) Iteration aan het einde van ieder Planning Interval. Deze biedt ruimte voor innovatie, leren, voorbereiding en verbetering, naast andere noodzakelijke werkzaamheden.

Voor werknemers betekent dit dat leren en verbeteren een vast onderdeel van je werk worden, met daar ook echt tijd voor in de planning, in plaats van iets wat je 'erbij' doet als het even rustig is.

15. Centralisatie van werk: samenhangende backlogs in plaats van losse to-do-lijsten

In een SAFe®-omgeving wordt werk zichtbaar gemaakt en beheerd via samenhangende backlogs. SAFe® kent daarbij verschillende niveaus, zoals Portfolio Backlogs, ART Backlogs en Team Backlogs. Het gaat dus niet letterlijk om één backlog voor de hele onderneming, maar om een geïntegreerde structuur waarin duidelijk is welk werk waar thuishoort en hoe het samenhangt.

Het streven is om losse to-do-lijsten, ongecontroleerde spreadsheets en parallelle administraties zoveel mogelijk te voorkomen. Hierdoor worden werkvoorraden, prioriteiten en voortgang overzichtelijker en kan er veel onnodig mailverkeer verdwijnen.

Voor werknemers betekent dit dat werk niet langer via losse mailtjes, verzoeken op de gang of eigen lijstjes binnenkomt, maar via de backlog. Werk dat niet op de backlog staat, wordt in principe ook niet opgepakt.

16. Verandering in mindset / cultuur

SAFe® vereist een verschuiving van een fixed mindset naar een growth mindset, waarin medewerkers openstaan voor leren, samenwerking, experimenteren en voortdurende verbetering. Ook vertrouwen, transparantie, eigenaarschap en het bespreekbaar maken van problemen worden belangrijker.

Regelmatige afstemming via bijvoorbeeld PI Planning, Coach Sync en PO Sync ondersteunt samenwerking en alignment. Maar een andere cultuur ontstaat niet automatisch door nieuwe meetings of functietitels. Leidinggevenden en medewerkers moeten hun gedrag daadwerkelijk veranderen. Dit is vaak een van de lastigste onderdelen van een SAFe®-transformatie.

Voor werknemers betekent dit dat er meer van je wordt verwacht dan het volgen van nieuwe processen: openstaan voor feedback, problemen bespreekbaar maken, initiatief nemen en eigenaarschap tonen. Dat biedt ruimte, maar kan in het begin ook onwennig voelen.

17. Nieuwe teamindeling: Team Topologies, stabiele Agile Teams en ART's

In de traditionele projectwereld worden mensen regelmatig opnieuw op projecten gezet: een nieuw team, een nieuwe projectleider en na afloop gaat iedereen weer uit elkaar. In SAFe® wordt dit principe zoveel mogelijk omgedraaid: het werk komt naar het team, in plaats van dat mensen steeds opnieuw naar het werk gaan.

Werknemers werken in relatief stabiele, multidisciplinaire Agile Teams die samen met andere teams een Agile Release Train (ART) vormen en zijn georganiseerd rondom waardestromen.

SAFe® 6.0 gebruikt daarnaast inzichten uit Team Topologies en onderscheidt vier belangrijke teamtypen:

• **Stream-aligned teams:** leveren direct waarde binnen een waardestroom.

• **Platform teams:** ontwikkelen diensten en voorzieningen waarmee andere teams sneller kunnen leveren.

• **Enabling teams:** helpen andere teams tijdelijk om nieuwe kennis en vaardigheden te ontwikkelen.

• **Complicated-subsystem teams:** werken aan specialistische onderdelen die bijzondere expertise vereisen.

Voor werknemers kan dit betekenen dat de vertrouwde afdelingsindeling verandert en dat zij onderdeel worden van een vast team dat langdurig samenwerkt aan een product, dienst of waardestroom. Het team leert elkaar steeds beter kennen en kan daardoor voorspelbaarder en effectiever worden. Voor veel werknemers is dit een van de meest fundamentele veranderingen.

18. Kwaliteit is van iedereen: Built-in Quality en Definition of Done

Kwaliteit is niet langer iets wat uitsluitend achteraf door een aparte testafdeling wordt gecontroleerd. In SAFe® neemt het team gezamenlijk verantwoordelijkheid voor kwaliteit, testen, integratie en oplevering.

De Definition of Done (DoD) beschrijft wanneer werk daadwerkelijk als afgerond mag worden beschouwd. Dat kan bijvoorbeeld betekenen dat iets niet alleen gebouwd is, maar ook getest, gereviewd, geïntegreerd en voldoende gedocumenteerd. Werk dat niet aan de afgesproken Definition of Done voldoet, wordt niet als Done beschouwd.

Voor werknemers betekent dit dat kwaliteit veel nadrukkelijker onderdeel wordt van het dagelijkse werk en dat problemen niet worden doorgeschoven naar een latere projectfase.

19. PI Planning: samen plannen voor een heel PI

Doorgaans iedere acht tot twaalf weken komen de teams van een ART samen voor PI Planning, waarin zij gezamenlijk het komende Planning Interval plannen. Een standaard PI Planning duurt twee dagen, al kan de praktische inrichting variëren.

Werknemers zitten daarbij zelf aan tafel. Ze vertalen Features naar concreet teamwerk, stemmen afhankelijkheden af met andere teams, bespreken risico's en formuleren hun eigen Team PI Objectives. Je plant dus niet meer alleen voor een ander, maar vooral samen met de mensen die het werk daadwerkelijk uitvoeren.

20. Van functionele silo's naar T-shaped professionals

Werknemers blijven specialisten, maar ontwikkelen daarnaast bredere vaardigheden. Dit wordt vaak het T-shaped-profiel genoemd: veel diepgang in het eigen vakgebied, gecombineerd met bredere kennis en vaardigheden daarbuiten.

Een ontwikkelaar die ook kan testen, of een analist die kan ondersteunen bij het ontwerp, maakt het team minder afhankelijk van afzonderlijke specialisten. Het betekent niet dat iedere professional alles moet kunnen. Het uitgangspunt is dat teamleden elkaar waar mogelijk helpen, zodat werk minder lang blijft liggen en het team als geheel effectiever wordt.

Voor werknemers ontstaan hierdoor nieuwe mogelijkheden om kennis op te doen en zich professioneel te ontwikkelen.

21. Direct contact met klanten en gebruikers

In SAFe® staat de klant centraal: Customer Centricity. Werknemers krijgen meer inzicht in wat klanten en gebruikers daadwerkelijk nodig hebben.

Ze kunnen bijvoorbeeld zelf resultaten demonstreren tijdens een Iteration Review of bijdragen aan een System Demo. Hierdoor ontstaat sneller feedback over de bruikbaarheid en kwaliteit van oplossingen. In plaats van maandenlang uitsluitend te werken vanuit een vooraf opgesteld eisenpakket, ontstaat een kortere feedbackcyclus met de mensen voor wie het product of de dienst bedoeld is.

Voor werknemers kan dit het werk betekenisvoller maken: je ziet beter wie je werk gebruikt en welke waarde je daadwerkelijk levert.

22. Leidinggevende en werknemer: een andere relatie

Ook leidinggevenden krijgen een andere rol. In plaats van opdrachten uitdelen en (bij sommige bedrijven) controleren, ligt het accent op faciliteren, coachen en het wegnemen van belemmeringen. SAFe® noemt dit Lean-Agile Leadership. Leidinggevenden verdwijnen dus niet, maar hun verantwoordelijkheden verschuiven.

In de praktijk ontstaat een heldere taakverdeling. Wat het team oppakt en in welke volgorde, bepaalt de Product Owner samen met het team via de backlog; je leidinggevende verdeelt geen dagelijkse taken meer en zit ook geen kernoverleggen meer voor. Loop je vast, dan helpt de Scrum Master/Team Coach je verder of daarna de RTE. Je leidinggevende blijft je aanspreekpunt voor ontwikkeling, loopbaan, welzijn en HR-zaken en voor escalaties die niet door de RTE kunnen worden opgelost. Productvisies worden ontwikkeld door de ART PM, ook niet meer door managers. Het gesprek over het werk zelf voer je dus primair binnen je Agile Team en op ART-niveau.

Voor werknemers betekent dit dat je beter weet bij wie je waarvoor terechtkunt, en dat het gesprek over je functioneren meer gaat over je bijdrage aan het team dan over individuele output. Dat vraagt van jou dat je zelf het gesprek over je ontwikkeling aangaat.

23. Tot slot: je kunt maximaal twee weken je 'snor drukken'

Deze titel is natuurlijk met een knipoog bedoeld: SAFe® is er niet om werknemers te controleren. Maar er zit wel een kern van waarheid in.

In SAFe® werken teamleden intensief samen door werk vanuit de Team Backlog naar de Iteration Backlog te halen. Het team bepaalt op basis van de beschikbare capaciteit hoeveel werk realistisch kan worden uitgevoerd. Tijdens de dagelijkse Team Sync wordt de voortgang besproken en worden belemmeringen zichtbaar gemaakt. Omdat de meeste iteraties ongeveer twee weken duren, wordt regelmatig duidelijk welk werk wel en niet is afgerond.

Daarbij ontstaat een veel grotere mate van transparantie dan in organisaties waar medewerkers soms maandenlang individueel aan werkzaamheden werken zonder tussentijdse demonstraties of gezamenlijke

evaluaties. Het uitgangspunt is niet om individuele werknemers voortdurend te controleren, maar om gezamenlijke verantwoordelijkheid en voorspelbaarheid te bevorderen. Teams kunnen bijvoorbeeld streven naar een hoge mate van voorspelbaarheid in hun Iteration Planning, maar percentages zoals 85% voorspelbaarheid of 90 tot 95% capaciteitsbenutting zijn geen universele SAFe®-normen.

Ziekte, onverwachte problemen en veranderende omstandigheden blijven uiteraard mogelijk. Maar doordat teams hun werk, voortgang en belemmeringen transparant maken, wordt het aanzienlijk lastiger om langdurig onzichtbaar te blijven of problemen voor collega's verborgen te houden. Dat vraagt van iedereen verantwoordelijkheid voor het gezamenlijke resultaat.

Samenvattend

De overstap naar SAFe® brengt voor professionals een verandering met zich mee van een overwegend traditionele, projectgestuurde aanpak naar een meer flexibele, collaboratieve en iteratieve manier van werken. Medewerkers krijgen meer invloed op hun planning, meer inzicht in de organisatiedoelen, intensievere samenwerking met collega's en meer verantwoordelijkheid voor de kwaliteit en resultaten van hun werk.

Tegelijkertijd brengt een SAFe®-transformatie niet alleen voordelen met zich mee. Het vraagt ook om het loslaten van vertrouwde werkwijzen, rollen, verantwoordelijkheden en soms bestaande vormen van autonomie. Dat kan in het begin weerstand en ongemak veroorzaken. Succesvol transformeren vraagt daarom niet alleen om nieuwe processen, rollen en tooling, maar vooral om vertrouwen, transparantie, communicatie, leiderschap en een andere mindset.

Daar staat tegenover dat organisaties en medewerkers beter in staat worden gesteld om sneller waarde te leveren, voortdurend te verbeteren en effectiever op veranderingen in te spelen. Precies de reden waarom veel organisaties voor SAFe® kiezen!

24. Transformeren naar SAFe: wat betekent dit voor zittend management? (deel 1)

De geopolitieke situatie is onrustiger dan ik in mijn hele loopbaan heb meegemaakt. Als het tegenzit, staan we aan de vooravond van concrete vijandigheid. Dat is geen abstract scenario meer. Ik merk het dagelijks in mijn opdracht bij Defensie.

Oekraïne laat zien wat er nodig is. Daar wordt in weken aangepast waar wij traditioneel jaren over doen. Denk je dat zij eerst een projectvoorstel schrijven, twee maanden later een stuurgroepbesluit nemen en daarna zes maanden aan een PID werken, weer gevolgd door een stuurgroepbesluit? Of dat een CAB één keer per maand vergadert over welke changes wel en niet worden doorgevoerd? Natuurlijk niet.

Neem het dronespeelveld: dat verandert in een tempo dat met klassieke projectmatige planning simpelweg niet bij te houden is. Wat vandaag werkt, is morgen achterhaald. Wie dan wacht op het volgende goedgekeurde plan, loopt per definitie achter.

Daarom ben ik ervan overtuigd dat innovatie in deze barre tijden geen luxe is, maar een noodzaak. Organisaties moeten direct kunnen inspelen op veranderingen, snel kunnen leren en die lessen meteen kunnen omzetten in werkende oplossingen. Precies daarvoor is SAFe® (Scaled Agile Framework) ontworpen. En precies daarom zet ik me er met zoveel overtuiging voor in.

Het implementeren van SAFe® binnen organisaties is een grote stap in die richting. Maar wat betekent dit voor jou als **lijnmanager** of als iemand die in de lijn werkt?

In dit hoofdstuk, deel 1, neem ik je stap voor stap mee. Ik begin met het kader waarbinnen deze verandering plaatsvindt: het **Dual Operating System van John Kotter**, waarop SAFe® is gebaseerd. SAFe® wordt naast de lijn geplaatst en vervangt de lijn dus niet, net zoals je vroeger “zelfstandige tijdelijke projectorganisaties” naast de lijn had. Met één belangrijk verschil: de SAFe®-organisatie is niet tijdelijk, maar blijvend. Daarna beschrijf ik kort wat er verandert voor de lijnorganisatie en het zittend management, wat hetzelfde blijft, hoe SAFe® jou als lijnmanager helpt en wat je nu al kunt doen. Ik sluit af met een conclusie.

In het kort

• SAFe® vervangt de lijn niet, maar komt ernaast te staan: het Dual Operating System van Kotter.

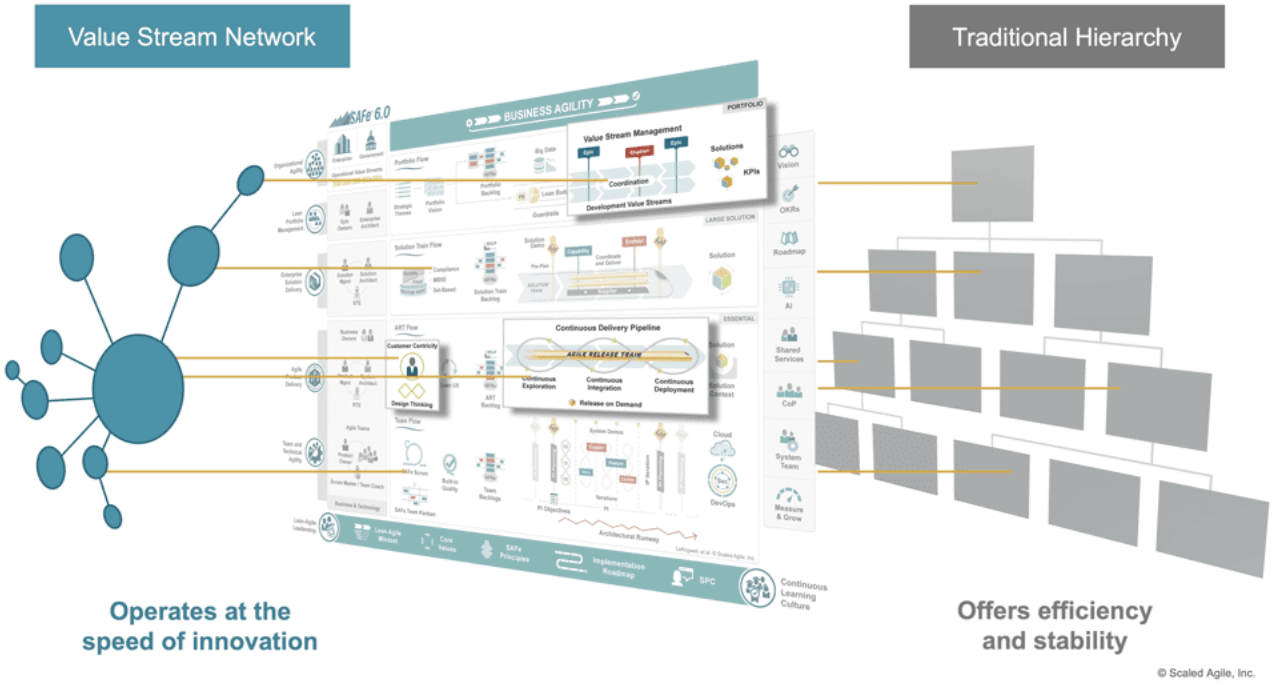
• Het primaire voortbrengingswerk, het ontwikkelen van oplossingen, verhuist naar de SAFe®-fabriek. Je medewerkers werken daar zo’n 97% van hun tijd.

• Je rol als lijnmanager verschuift van sturen naar ondersteunen, je eigen mensen coachen en grote obstakels wegnemen.

• Personeelsbeheer, compliance en faciliteiten blijven bij de lijn. Strategie en financiering bepalen de lijn en SAFe® samen.

1. Wat is een Dual Operating System?

Het **Dual Operating System**, zoals beschreven door John Kotter, is een model waarbij een traditionele hiërarchische structuur wordt *aangevuld* met een meer wendbare netwerkstructuur. Het bestaat dus uit twee “operating systems” (OS):



• **OS #1: De lijn (het harkje).** Kotter noemt dit “de hiërarchie”. De lijn blijft verantwoordelijk voor stabiliteit, efficiëntie en het uitvoeren van dagelijkse operationele taken. Denk aan personeelsbeheer, budgettering en compliance.

• **OS #2: SAFe®.** Kotter noemt dit “het netwerk”; in SAFe® heet het het *Development Value Stream Network*. In dit hoofdstuk noem ik het kortweg de SAFe®-fabriek. SAFe® richt zich op snelheid, innovatie en aanpassingsvermogen. SAFe® bestaat uit (heel veel) agile teams die samenwerken in waardestromen om waarde te leveren aan klanten en gebruikers.

Weet dat SAFe® “de fabriek is waar al het ontwikkelwerk wordt gedaan”. Daar zitten alleen mensen die Solutions bouwen en testen, UX-ontwerpers, architecten, SAFe®-experts, enzovoort. In de fabriek zijn de Scrum Masters/Team Coaches en Release Train Engineers straks degenen die zeggen: “ik heb vijf man extra nodig”. Zij hebben (als zij hun vak verstaan) ook inzicht in welke expertise LPM of een bepaalde ART wanneer nodig heeft. De financiering bepalen lijn en SAFe® samen: via Lean Portfolio Management (LPM) krijgen de waardestromen een eigen budget, in plaats van geld per project. De SAFe®-LPM-experts weten welke Epics (initiatieven die je vroeger als project zou starten) we kunnen oppakken, hoeveel tegelijkertijd en wanneer we verwachten dat ze klaar zijn.

In de SAFe®-fabriek zit geen bedrijfsbureau, geen HRM, geen schoonmakers, geen SAP-hulp voor het urenschrijfgedoe, geen portiers, geen facilitaire ondersteuning en geen lijnmanagers in hun lijnrol. Lijnmanagers die als Business Owner optreden, doen wel mee aan onder meer de PI Planning.

In deze opzet werken de hiërarchie en SAFe® dus naast elkaar en versterken ze elkaar. Het model van Kotter maakt het mogelijk om zowel flexibel (SAFe®) als betrouwbaar (de lijn) te zijn. Dat is cruciaal in een complexe en veranderlijke omgeving, zoals bij veel van mijn klanten.

2. Wat verandert er voor de lijnorganisatie en het zittend management?

Met de introductie van SAFe® introduceren we een tweede besturingssysteem binnen de bestaande organisatie: het waardestroomnetwerk, ook wel de SAFe-fabriek genoemd. Deze verandering heeft directe gevolgen voor jou als manager, afdelingshoofd of leidinggevende.

De SAFe®-fabriek gaat namelijk het ontwikkelwerk doen. De primaire voortbrengingsprocessen voor oplossingen worden binnen SAFe® georganiseerd: requirements opstellen, de klantvraag uithoren, architectuur, security, privacy en software development. Onderhoud en doorontwikkeling van oplossingen worden bij voorkeur door de Agile Teams uitgevoerd volgens DevOps. Operationele dienstverlening, zoals First Line Support, kan in de lijn blijven. De precieze taakverdeling hangt af van het organisatieontwerp. Ik hoor je denken: “Maar wat gaat de lijn dan nog doen?” Goeie vraag!

Hieronder beschrijf ik, op basis van mijn ervaringen uit diverse SAFe®-transformaties, wat ik verwacht.

Let op: SAFe® is altijd maatwerk. Per organisatie moet goed worden gekeken wat wel en niet past, en wat wel en niet werkt.

Nieuwe rollen en verantwoordelijkheden van lijnmanagers

• **Medewerkers zitten straks in SAFe®, OS #2.** Al je eigen mensen die in het primaire businessproces zitten en die nu mogelijk naast je in hetzelfde pand zitten, werken straks de hele dag in SAFe®, voor ongeveer 97% van hun tijd. Daar werken ze *intensief* samen met de Product Owners, Scrum Masters/Team Coaches, Release Train Engineers en alle andere SAFe®-rollen.

• **Focus op ondersteuning.** Als lijnmanager verschuift je rol van direct sturen naar ondersteunen en faciliteren. Jouw expertise en ervaring zijn essentieel om de echt grote obstakels voor de SAFe® agile teams weg te nemen. Kleine obstakels lossen de Scrum Masters/Team Coaches en Release Train Engineers zelf op. Het aantal problemen dat bij lijnmanagers terechtkomt, neemt daardoor flink af.

• **Talentontwikkeling.** Je wordt een mentor voor je eigen medewerkers. Je helpt hen groeien in hun rol en ondersteunt hen bij het omarmen van agile werken.

• **Samenwerking met de SAFe®-mensen.** Waar beslissingen, zoals make-or-buy-vraagstukken, voorheen vaak top-down door een afdelingshoofd werden genomen, worden ze nu vrijwel allemaal in SAFe® genomen. Dat vraagt samenwerking tussen de hiërarchie en de strategische SAFe®-rollen, zoals Product Management, Product Owners (PO’s) en Release Train Engineers (RTE’s). SAFe® neemt besluiten namelijk aanzienlijk sneller dan reguliere en traditionele lijnprocessen. Zo is SAFe® ingericht.

Minder directe controle, meer vertrouwen

SAFe® stimuleert zelforganiserende teams die in korte iteraties werken. Als manager heb je daardoor minder directe *controle*, maar geef je meer vertrouwen aan de teams. Dat vraagt een mindsetverandering: van sturen naar *faciliteren*. Je leidt door het goede voorbeeld te geven en ruimte te maken, niet door het werk te verdelen.

Verandering in tijdsbesteding

Je zult merken dat je als lijnmanager je tijd straks meer verdeelt over:

• strategisch werken aan langetermijnontwikkelingen;

• grote obstakels voor de agile teams wegnemen;

• samenwerken met andere managers en het netwerk om waardestromen te optimaliseren.

3. Wat blijft hetzelfde?

Hoewel SAFe® nieuwe verantwoordelijkheden en manieren van samenwerken met zich meebrengt, blijven sommige aspecten van jouw werk onveranderd:

• Je blijft verantwoordelijk voor compliance, personeelsbeheer, faciliteiten en de naleving van wet- en regelgeving.

• Je blijft betrokken bij strategie en financiering, maar nu samen met de SAFe®-rollen in LPM.

• Je blijft aanspreekpunt voor je medewerkers, maar in een ondersteunende rol: bij ziekte, bij persoonlijke problemen en als luisterend oor. Overdag zitten ze in SAFe®.

• Je expertise blijft van grote waarde bij het adviseren van teams en andere managers.

Let wel: de prioritering van al het ontwikkelwerk wordt volledig in SAFe® afgehandeld. Wat we wanneer gaan bouwen, komt dus uit SAFe®.

4. Voordelen van SAFe® voor jou als lijnmanager

SAFe® biedt niet alleen uitdagingen, maar ook kansen:

• **Minder ad-hocwerk.** Dankzij de focus op waardestromen en prioritering binnen SAFe® worden processen overzichtelijker en minder versnipperd.

• **Grotere impact.** Door te werken in een structuur die gericht is op klantwaarde, draag je directer bij aan het succes van de organisatie.

• **Ontwikkeling van je leiderschapsvaardigheden.** De rolverandering is een kans om nieuwe vaardigheden te ontwikkelen, zoals agile leiderschap en coaching.

• **Minder meetings.** De meeste meetings in SAFe® zijn volledig gestandaardiseerd; SAFe® heeft daar al lang over nagedacht. De SAFe®-mensen zitten in die meetings, waardoor de lijn meer tijd krijgt voor de zaken hierboven.

5. Wat kun jij nu doen?

• **Leer over SAFe®.** Volg trainingen en workshops om de basisprincipes van SAFe® en jouw rol binnen het netwerk te begrijpen.

• **Communiceer met je collega's.** Help hen te begrijpen wat deze verandering voor hen betekent en hoe zij zich kunnen voorbereiden.

• **Stel vragen.** De agile experts zijn er om je te ondersteunen. Schroom niet om vragen te stellen over je nieuwe rol of over de impact van SAFe® op je werk.

SAFe® is een reis die we samen maken. SAFe®-experts zijn zo agile als het maar kan, dus niets is in beton gegoten. Een agilist zet niets voort waarvan hij ziet dat het niet werkt. SAFe® is maatwerk, dus dit verhaal wordt helemaal op maat gemaakt voor jouw organisatie. En die transitie begeleiden we uiteraard tiptop, stap voor stap.

6. Conclusie

Het Dual Operating System geeft ons de structuur om stabiliteit en wendbaarheid te combineren. Als lijnmanager, maar ook als medewerker in een ondersteunende functie, speel jij een sleutelrol in het succes van deze transformatie. Samen bouwen we een toekomstbestendige organisatie die klaar is voor elke uitdaging. En samen werken we aan het doel van zeer tevreden gebruikers en klanten: op tijd leveren, met de hoogst mogelijke betrouwbaarheid.

Vragen of meer weten? Neem gerust contact met me op via safe@majormilestones.nl of stuur me een bericht.

25. Transformeren naar SAFe: wat betekent dit voor zittend management? (deel 2)

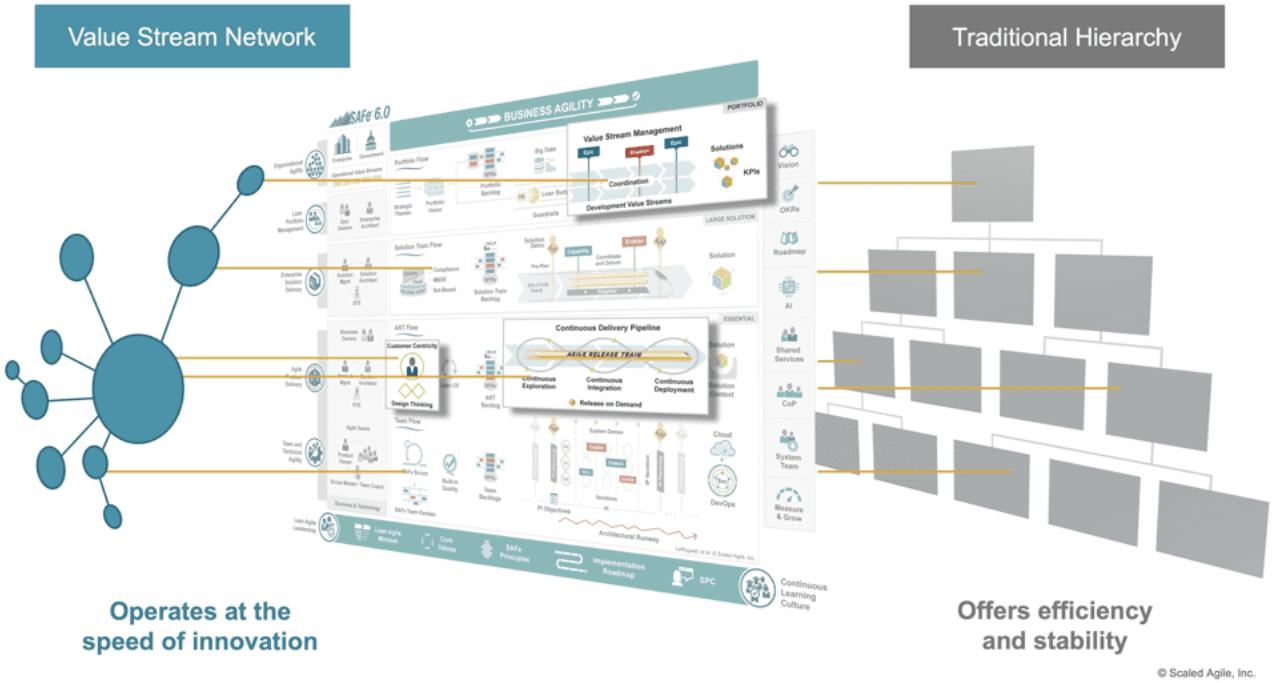
Wat doet een lijnmanager de hele dag als SAFe is geïmplementeerd en het dagelijkse ontwikkelwerk voortaan in Agile Teams en ART's plaatsvindt?

In twee eerdere hoofdstukken ben ik ingegaan op de impact op eindgebruikers en werknemers. In dit tweede deel kijk ik opnieuw naar de *impact op het management*, maar dan meer uitgebreid.

:check_mark: *Voor de eenvoud en scherpte heb ik in dit hoofdstuk een eventuele hybride omgeving (waterfall naast SAFe®) buiten beschouwing gelaten. Houd daar bij het lezen rekening mee.*

:check_mark: *Dit hoofdstuk gaat uit van een echte traditionele lijnorganisatie met een duidelijke hiërarchie, afdelingssilo's en traditionele lijnmanagers.*

1.1. Kotters dual operating system: de basis van SAFe®'s second operating system (OS)



Figuur 1: De LACE ontwerpt het linkerdeel; aan de structuur van het rechterdeel wordt niet getornd, aan de werkwijze wel.

In *Accelerate: Building Strategic Agility for a Faster-Moving World* (XLR8) introduceert John Kotter het concept van een 'dual operating system'. Dit idee combineert een traditionele hiërarchie met een flexibeler, netwerkachtig systeem dat gericht is op innovatie en snelle besluitvorming. Kotters concept biedt organisaties een manier om wendbaar te zijn zonder de stabiliteit van hun dagelijkse operatie te verliezen.

Kotter beschrijft in dat boek twee parallelle structuren:

1. **De hiërarchie**: deze structuur waarborgt stabiliteit en efficiëntie. SAFe®, en dus de LACE, herontwerpt de hiërarchie niet: operationele primaire processen (Defensie: 'Hoofdtaak 1, 2 en 3'), rapportagelijnen, functies en beloningsstructuren blijven bestaan. Dat betekent overigens niet dat de hiërarchie stilstaat (zie 1.1.3).

2. **Het Value Stream Network**: een aparte, flexibele structuur naast de hiërarchie, ontworpen door de LACE en gericht op het *uitvoeren van al het ontwikkelwerk*: het bedenken, bouwen, uitrollen en doorontwikkelen van oplossingen.

1.1.1. Waarom past SAFe® twee operating systems toe?

The solution is not to trash what we know and start over but instead to reintroduce a second system: one which would be familiar to most successful entrepreneurs.

You need a dual operating system.

- John P. Kotter, *XLR8*

SAFe® heeft Kotters dual operating system geïntegreerd als basis voor het Value Stream Network: een structuur die Agile-werkwijzen ondersteunt zonder de hiërarchie te verstoren. Dit model is om twee redenen overgenomen:

1. **Wendbaarheid verhogen**: het Value Stream Network maakt snelle aanpassing aan marktveranderingen mogelijk.

2. **Samenwerking over silo's heen**: het netwerk bevordert cross-functionele samenwerking, waardoor waarde efficiënter wordt geleverd met een kortere time-to-field.

Een belangrijk verschil met Kotter: bij Kotter is het netwerk een aanvullende, grotendeels vrijwillige structuur die strategische initiatieven versnelt. SAFe® maakt er een permanente, georganiseerde structuur van, ingericht rond Development Value Streams en Agile Release Trains. Het netwerk is in SAFe® dus geen bijzaak, maar de plek waar het ontwikkelwerk structureel gebeurt.

1.1.2. Twee systemen, één organisatie

Het netwerk en de hiërarchie zijn geen twee aparte organisaties, maar twee kanten van dezelfde medaille. Iedereen die in het netwerk werkt, heeft een thuisbasis in de hiërarchie. En iedereen in de hiërarchie verkoopt, ondersteunt en levert de oplossingen die het netwerk ontwikkelt. Beide systemen zijn bovendien voor iets anders geoptimaliseerd:

	Netwerk (OS #2)	**Hiërarchie (OS #1)**
Geoptimaliseerd voor	Snelheid van innovatie	Efficiëntie, betrouwbaarheid en continuïteit. :info: Dit is meer dan alleen een IT organisatie of afdelingshoofd!
Kerntaak	Oplossingen ontwikkelen, uitrollen en doorontwikkelen (Development Value Streams)	De organisatie laten draaien: financiën, juridisch, HR, inkoop, sales, productie en levering (Operational Value Streams)
Houding tegenover variatie	Experimenteren, snel leren, falen zonder schuld	Variatie vermijden, fouten analyseren en voorkomen
Structuur	Herschikt zich wanneer het werk verandert, zonder wijziging van functie, beloning of rapportagelijn	Stabiel kader voor loopbaan, beloning en verantwoording

Daarnaast zijn er **gedeelde verantwoordelijkheden** die beide systemen verbinden, zoals:

• **Strategie**: één strategie en visie, met afstemming over investeringen, portfolio-operaties en Lean Governance (de LPM-discipline).

• **Klantcontact**: de hiërarchie werft en behoudt klanten; het netwerk betrekt klanten vroeg bij de ontwikkeling.

• **Inkoop**: verwerving kan door beide OS-en worden gedaan.

• **Cultuur**: werkwijzen mogen per systeem verschillen, maar de organisatie heeft één verbindende cultuur.

• **Matchings proces:** OS #1 zorgt voor kwalitatief hoog opgeleide tevreden en gemotiveerde werknemers die een rol krijgen in OS #2 met dus een matchings-proces van vraag en aanbod.

Die verschillende optimalisaties leveren onvermijdelijk **spanning** op. Het netwerk wil snelheid en flexibiliteit; de hiërarchie wil voorspelbaarheid en ziet verandering soms zelfs als bedreiging. Dat is geen teken dat de transformatie mislukt, maar een gegeven dat actief gemanaged moet worden. Precies daar ligt een belangrijke rol voor het zittende management.

1.1.3. Ook de hiërarchie, OS #1, moet wendbaarder worden

Een veelgemaakte denkfout is dat het netwerk ‘agile’ is en de hiërarchie ‘stabiel’ mag blijven. SAFe® stelt juist dat Business Agility vraagt om groeiende wendbaarheid in *beide* systemen. De structuur van de hiërarchie blijft staan, maar de manier van werken verandert wel.

SAFe® geeft daar aanvullende richtlijnen voor, zoals:

• **Agile HR**: werving, beoordeling, beloning en loopbaanontwikkeling afstemmen op kenniswerkers en teamprestaties. Dat vraagt ook iets van HRM zelf: HRM-managers en HR-adviseurs moeten Agile HR leren kennen én zelf agile worden. Het klassieke HR-instrumentarium (vaste functieprofielen, een jaarlijkse beoordelingscyclus, individuele targets en bonussen) is ontworpen voor de hiërarchie en botst met teams en treinen in OS #2. Zelf agile worden betekent dat HR ook zo werkt: in korte cycli, met een eigen backlog, met kleine experimenten met nieuwe instrumenten en met de ART’s als interne klant.

• **Agile Contracts**: contractvormen tussen ‘fixed price’ en ‘time and materials’ die helpen om risico’s, samenwerking en de economische belangen van beide partijen beter op elkaar af te stemmen.

• **Agile Marketing**: marketing die in dezelfde cadans en met dezelfde wendbaarheid werkt als productontwikkeling.

• **Operational Value Streams** optimaliseren: Lean-denken toepassen op de primaire processen van levering en dienstverlening, die vaak door klanten en leveranciers heen lopen.

De discipline Leadership and Culture geldt daarom nadrukkelijk ook voor managers die in de hiërarchie blijven, met competenties als Leading Change.

1.2. De impact op zittend management

Als het ‘*operationele ontwikkelwerk (aka 'Development Value Streams’)* straks binnen het tweede operating system (OS #2) wordt uitgevoerd, wat doet het lijnmanagement dan de hele dag? Hieronder volgt een overzicht van de impact op het huidige lijnmanagement.

Werken in het eerste of het tweede OS: huidige lijnmanagers moeten kiezen of ze in 'de hiërarchie' blijven werken of overstappen naar het tweede OS, bijvoorbeeld in de rol van Release Train Engineer, Product Manager of een functie binnen SAFe®'s Lean Portfolio Management. Het woord *kiezen* is hier bewust gekozen: wie overstapt, laat de hiërarchische rol achter.

Geen hiërarchie in OS #2

SAFe® is een **framework dat op vertrouwen draait**: het werkt alleen als er **psychologische veiligheid** is. Teams en treinen organiseren zichzelf binnen afgesproken kaders, en mensen moeten problemen, fouten en risico's open kunnen benoemen zonder dat hun beoordelaar meekijkt.

Een lijnmanager die RTE, Scrum Master of Product Owner wordt van zijn eigen medewerkers, neemt de hiërarchie mee het netwerk in. Daarmee verdwijnt de veiligheid die SAFe® nodig heeft. Kies je voor een rol in OS #2, draag dan geen hiërarchische verantwoordelijkheid meer voor dezelfde mensen. Dat geldt het sterkst voor de Scrum Master/Team Coach en de RTE; bij Product Owner en Product Manager, die over het ‘wat’ gaan en niet over het ‘hoe’, is het risico kleiner. Waar een strikte scheiding (tijdelijk) niet haalbaar is, beleg de beoordeling dan bij een andere manager. Advies: Eén pet per relatie.

Lees meer in hoofdstuk 34, ‘Lijnmanager én gelijkwaardige collega? Waarom SAFe® duidelijke rolgrenzen vraagt’.

(Alle) professionals worden (bij voorkeur fulltime) gedetacheerd naar het tweede OS: Agile teams zijn cross-functioneel, wat betekent dat alle benodigde rollen voor het eindproduct in het team vertegenwoordigd moeten zijn. Dit houdt in dat de ontwikkelafdelingen en -sectoren overdag leeg zijn, omdat vrijwel iedereen daar in het tweede OS werkt. Voor afdelingen die de operatie draaiende houden (financiën, inkoop, logistiek) geldt dat niet: zij blijven het primaire proces in de hiërarchie uitvoeren. Bij beheer is onderscheid nodig: de operationele dienstverlening blijft bij veel bedrijven in de hiërarchie (OS #1), zoals bijvoorbeeld First Line Support. Terwijl onderhoud en doorontwikkeling van oplossingen door de Agile teams bij voorkeur op een DevOps wijze worden uitgevoerd. Waar die grens precies ligt, hangt af van de inrichting van de organisatie en het totale ontwerp.

Verschuiving van verantwoordelijkheden: in traditionele organisaties zijn lijnmanagers vaak verantwoordelijk voor het toewijzen van resources, het prioriteren van werk en het vaststellen van KPI's en deadlines. Hindernissen worden opgelost door het management in bijvoorbeeld een wekelijks afdelingsoverleg. In een SAFe®-omgeving worden deze taken echter uitgevoerd door Product Owners (PO's), Scrum Masters (SM's), Release Train Engineers (RTE's), Product Managers en ondersteunende architecten. Veel bevoegdheden verschuiven naar deze rollen. RTE's uit het tweede OS escaleren naar het lijnmanagement wanneer ze er zelf niet uitkomen, of wanneer een besluit buiten hun mandaat ligt.

Veranderend takenpakket, meer tijd voor HR: managers in een SAFe®-organisatie blijven vaak op hun positie in de hiërarchie, ook al voeren hun medewerkers hun dagelijkse werk voortaan uit in het tweede OS. Dat kan voor beide partijen een betere situatie opleveren. Medewerkers hebben nog steeds iemand nodig die hen helpt met hun loopbaanontwikkeling. Een manager blijft verwachtingen en beloning vaststellen en bewaken, en biedt de coaching die nodig is om individuele vaardigheden en loopbaandoelen te ontwikkelen. Ook moeten teamleden nog steeds worden aangenomen en HR-conflicten worden opgelost. Bovendien moeten Agile teams de ruimte krijgen om te experimenteren en te innoveren.

Beoordelen als je medewerker in een Agile Team werkt

De lijnmanager blijft verantwoordelijk voor zijn of haar mensen en hun ontwikkeling, ook als zij hun dagelijkse werk in een Agile Team uitvoeren. Je hoeft niet alles zelf te zien, maar je moet wel een onderbouwd beeld organiseren.

Spreek daarom vooraf af waarop je beoordeelt en verzamel gedurende het jaar concrete voorbeelden van de medewerker, directe collega’s en relevante inhoudelijke aanspreekpunten. Bespreek:

• mate van verhoogd vakmanschap;

• mate van commitment reliability

• mate van cultural fit en ‘growth mindset’

• samenwerking met de andere teamleden

• de bijdrage aan het totale teamresultaat.

• etc

De **dagelijkse werksturing** ligt bij het team zelf en de daarvoor aangewezen SAFe®-rollen. De **formele personeelsbeoordeling** blijft bij de daarvoor bevoegde beoordelaar. Die twee verantwoordelijkheden verbinden we met duidelijke afspraken over feedback.

⚙ **Eigenaar van de operationele waardestromen:** managers die verantwoordelijk zijn voor de operatie houden een volwaardige kerntaak. Zij zorgen dat producten en diensten betrouwbaar, rendabel en binnen wet- en regelgeving bij de klant komen. Nieuw is dat ook van hen wordt verwacht dat ze die operationele waardestromen met Lean-denken continu verbeteren.

Geen stuurgroepen en geen TGK-sturing, wel Lean governance: SAFe® kent geen stuurgroepen. Er wordt niet langer in de lijn gestuurd op Tijd, Geld en Kwaliteit (TGK) per project: kwaliteit, tijd en geld liggen in SAFe® vast via de vaste PI-cadans, stabiele teams en Lean Budgets, en de scope beweegt mee. De klassieke

projectsturing vervalt dus bij een native SAFe implementatie. De sturing van deliverables verloopt (afhankelijk van de gekozen SAFe configuratie) via de LPM-laag naar de Solution Train-laag, via ART's naar Agile teams en weer terug. Lijn governance verdwijnt daarmee niet, maar krijgt een andere vorm. De hiërarchie blijft bovendien verantwoordelijk voor de financiële en juridische governance van de organisatie.

Mede-eigenaar van strategie en financiering: de hiërarchie financiert zowel zichzelf als het netwerk. Managers dragen bij aan één gedeelde strategie en aan de verdeling van budgetten over de investeringshorizons. Horizon 1 (bestaande producten) vraagt om maximaal rendement; Horizon 2 (opkomende producten) en Horizon 3 (verkennende innovatie) leveren nu nog niets op maar zijn nodig voor de toekomst. Juist bij de overgang van Horizon 2 naar de reguliere operatie (sales, productie, forecasts) is het lijnmanagement onmisbaar: daar moet de hiërarchie capaciteit vrijmaken voor iets wat ze nog niet kent.

Verbinder tussen beide systemen: spanning tussen netwerk en hiërarchie lost zich niet vanzelf op. Managers die beide werelden kennen, zijn bij uitstek in staat om die spanning bespreekbaar te maken, belangen te verbinden en te voorkomen dat de twee systemen elkaar gaan tegenwerken.

Aanvullende rol: managers worden ook SAFe® Change Agents: veel managers nemen deel aan de implementatie van SAFe® door hun Lean-leiderschapsvaardigheden en hun aanpassingsvermogen aan de nieuwe werkwijze te tonen. Zij helpen het LACE actief om de cultuur te veranderen. Lijnmanagers kunnen desgewenst ook LACE-lid worden en krijgen dan de training en middelen die ze nodig hebben om anderen te trainen en te coachen bij het realiseren van de verandering.

Van silo's naar waardestromen: traditionele silo's maken plaats voor multidisciplinaire teams die in waardestromen werken. Managers zijn daardoor niet langer alleen verantwoordelijk voor hun eigen afdeling, maar moeten bijdragen aan het succes van de hele waardeketen. Dat vraagt om een bredere blik en een focus op klanttevredenheid.

Leiderschap herdefiniëren: met de komst van SAFe® moeten managers hun leiderschapsstijl aanpassen naar 'Lean-Agile Leadership'. Waar ze vroeger veelal gericht waren op het aansturen en controleren van afdelingen en sectoren, verschuift hun rol nu naar coaching en mentoring, oftewel servant leadership. Dit vraagt om een cultuur van continu leren en verbeteren, waarin experimenteren en itereren centraal staan.

Nieuwe prestatiemetingen op basis van SAFe®-metrics: succes wordt niet langer gemeten aan individuele of afdelingsprestaties, maar aan de bijdrage aan de waardestroom in het tweede OS. Managers moeten hun manier van meten en belonen dus herzien en zich richten op wat echt waarde toevoegt voor de klant.

Veranderende communicatiestijl: de communicatie verschuift van top-down, afdelingsinformatie sessies, zeepkistsessies, naar meer horizontaal.

⧗ **En tot slot: investeer in tijd en energie:** een SAFe®-transformatie vraagt extra tijd en energie van het management. Denk aan tijd voor opleidingen, tijd om stukken te lezen, voor deelname aan ontwerpworkshops en voor het informeren en meenemen van medewerkers. Plan die tijd bewust in naast het lopende werk; anders verliest de transformatie het van de waan van de dag.

1.3. Conclusie

De implementatie van een second operating system binnen een SAFe®-transformatie vraagt om een fundamentele heroverweging van de rol van het management en extra tijd en energie. Van controleren naar faciliteren, van silo's naar waardestromen en van formele macht naar invloed: deze transformatie is niet alleen voor teams, maar ook voor managers een grote stap. Om haar te laten slagen, is het essentieel dat managers actief betrokken zijn en ondersteuning krijgen om in hun (mogelijk nieuwe) rol te excelleren.

Tot slot: er is maar één organisatie. Netwerk en hiërarchie zijn twee kanten van hetzelfde bedrijf en zijn van elkaar afhankelijk; geen van beide kan slagen zonder de ander. De vraag is dus niet óf er werk overblijft voor het zittende management, maar of het management bereid is om ook de eigen kant van de organisatie wendbaarder te maken.

1.4. Bronnen

• Kotter, J.P. (2014). *Accelerate: Building Strategic Agility for a Faster-Moving World*. Harvard Business Review Press.

• Spence, I. (2023). *Balancing the Dual Operating System*. SAFe® Knowledge Base, Scaled Agile, Inc. https://framework.scaledagile.com/balancing-the-dual-operating-system

26. De impact van het aanpassen van SAFe-terminologie: voordelen en nadelen voor je organisatie

Kies het juiste examenantwoord!

Ik wil dit hoofdstuk graag starten met een examenvraag uit het formele Scrum Examen van Scrum.org. Wat denk jij dat het goede antwoord is?



Inleiding

Wanneer organisaties beginnen met het implementeren van SAFe (Scaled Agile Framework), komt vaak de vraag naar boven: moeten we de standaard SAFe-termen en -begrippen overnemen, of passen we deze aan aan onze eigen taal, cultuur, andere frameworks en bestaande begrippenkaders?

Hoewel het verleidelijk kan zijn om de SAFe terminologie aan te passen naar een volledig eigen vocabulaire, heeft deze keuze mogelijk voordelen, maar helaas meer nadelen. Het aanpassen van termen heeft bijvoorbeeld direct impact op alle inhuur en externe samenwerkingspartners en begripsvorming van het SAFe model. Het

juiste antwoord op de vraag boven is dan ook D).

In dit hoofdstuk neem ik je hierin mee en houd ik een pleidooi voor het zo veel mogelijk vasthouden aan de formele SAFe-terminologie, zodat je je niet laat verleiden tot een eigen, sterk afwijkende SAFe-variant.

Concrete praktijkvoorbeelden

• SAFe's Communities of Practice (CoP's) noemen we Gildes, overgenomen uit het Spotify model;

• De ART level PM noemen we Chief Product Owner, afgekort CPO;

• Het 'PI event' noemen we 'kwartaal meeting';

• De RTE noemen wij de ART Programma Manager;

• De ART level Coach-Sync Meeting noemen wij 'SM Sync' meeting;

• Agile Teams noemen wij 'innovation 'Squads'' of 'Project Sprint Teams'. Of Tribes.

• De LACE noemen wij het 'Transformatieteam'

Voordelen van het Aanpassen van Terminologie

• **Vermindering van weerstand:** Nieuwe taal en nieuwe woorden kunnen angst oproepen want *'onbekend maakt onbemind*'. Wat is een ART? Wat is een RTE? Wat is een PM? Wat is een Epic? Verandering kan daardoor weerstand oproepen, maar door dan vertrouwde termen te gebruiken, kan die weerstand verminderd worden. Mensen voelen zich meer comfortabel als de taal niet compleet nieuw is en een beetje vertrouwd blijft.

• **Verduidelijking naar eigen belevingswereld:** Eigen naampjes kunnen specifieke betekenissen toevoegen die aansluiten bij de werkprocessen van de organisatie, wat kan leiden tot beter begrip en minder verwarring.

• **Vermijden taaldubbelingen:** Ik heb meerdere keren meegemaakt dat de SAFe term PM, wat staat voor Product Manager op ART-niveau, al in gebruik was binnen marketing, de lijn etc en het te verwarrend voor de organisatie was. In die gevallen werd de rolnaam "PM" aangepast aan bijvoorbeeld cPO, *Chief Product Owner*. Die organisaties blijven zo dicht bij SAFe en ik snap nog steeds en direct, wat de naam qua functie inhoudt.

Nadelen van het Aanpassen van Terminologie

• **Verlies van consistentie:** wanneer je afwijkt van de standaard SAFe-terminologie, kan dit leiden tot verwarring, vooral wanneer er met andere teams, afdelingen of organisaties wordt samengewerkt.

• **Verwarring en onduidelijkheid:** Ik kom op intake en de opdrachtgever vraagt 'waar werk je precies?' en ik antwoord: "Ik ben de PO van het Transformatieteam". Is het dan direct voor iedereen duidelijk op welk niveau van SAFe ik zit, waar ik zit en wat ik doe?

• **Langzamere inwerkperiode voor externen met dito kosten:** Freelancers, extern ingehuurde medewerkers en samenwerkingspartners zijn vaak bekend met de standaard SAFe-termen en/of Certified SAFe. Door eigen taal te gebruiken, wordt hun inwerkperiode langer omdat ze de aangepaste terminologie niet kennen, wat de samenwerking en onboarding vertraagt.

• **Moeilijkheden bij training en ondersteuning:** SAFe is een gestandaardiseerd raamwerk met veel beschikbare trainingsmaterialen en ondersteuning. Als je de terminologie wijzigt, kunnen die bronnen minder relevant worden en trainingen lastiger te volgen. Je bent immers niet vertrouwd met de formele SAFe begrippen.

• **Beperkingen in benchmarking:** Door andere termen te gebruiken, wordt het moeilijker om je voortgang te vergelijken met andere bedrijven die SAFe implementeren, omdat je mogelijk appels met peren vergelijkt.

• **Uitdagingen bij schaling, toekomstige problemen:** als je organisatie besluit om SAFe op bredere, organisatiebrede schaal toe te passen, kunnen de aangepaste termen en begrippen voor extra complexiteit zorgen, vooral wanneer je terug moet naar de standaardterminologie.

• **Remmen van de change (de examenvraag):** Een belangrijk nadeel van het aanpassen van formele terminologie is dat dit het *gevoel van verandering* kan remmen. Wanneer je eigen termen gebruikt, kan het voor medewerkers lijken alsof er weinig verandert, waardoor de transformatie minder voelbaar is. Door de SAFe-terminologie te behouden, wordt de verandering juist expliciet gemaakt, wat helpt om de transformatie binnen de organisatie te benadrukken en te versterken.

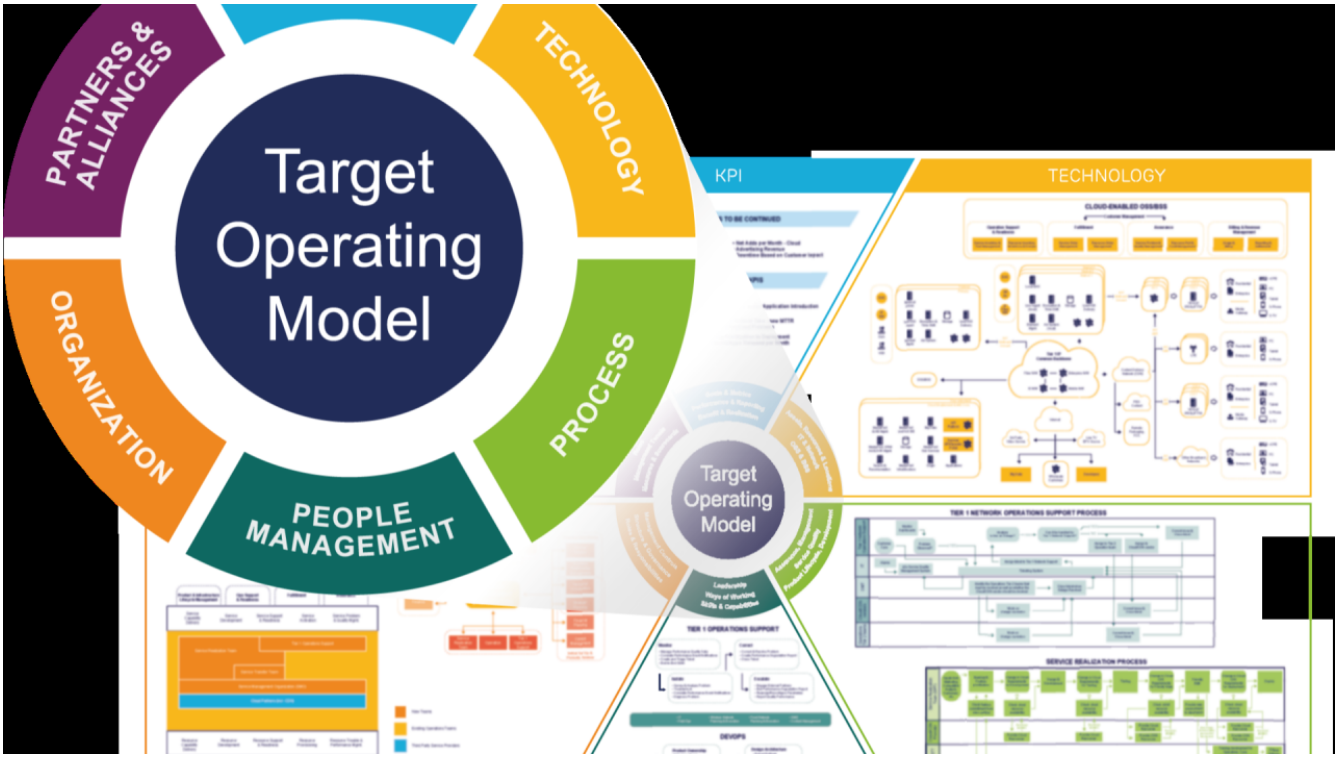
• **Gevaar van missen van de essentie van iets:** SAFe heeft heel goed nagedacht over de rolnaam Release Train Engineer. Waarom heet dit niet Release Train Manager? Om twee redenen: Managers zijn in SAFe servant leaders, ze willen niet teveel nadruk op het woord 'managen' leggen. Ten tweede moet de RTE de CICD pipeline door-engineeren om de LACE metrics te behalen. Dat engineeren, lean maken, is een van de belangrijkste RTE taken om Flow te genereren. Als je de naam aanpast, mis je deze essentie.

Samengevat: Liever niet doen!

Het aanpassen van SAFe-terminologie aan de organisatie(cultuur) kan voordelen bieden, zoals het verminderen van weerstand en extra organisatie specifieke verduidelijking. Echter, het brengt ook nadelen met zich mee, zoals verwarring bij samenwerking met externen en uitdagingen bij training en schaalbaarheid. Externe partijen, zoals freelancers, hebben meer tijd nodig om zich aan te passen aan de aangepaste taal, wat uit te drukken is in harde euro's. Ook voor jou persoonlijk, als professional, is het beter om de formele SAFe terminologie aan te houden.

Het is daarom belangrijk om een balans te vinden tussen maatwerk en standaardisatie om effectieve implementatie te waarborgen en taal (tribes, gildes, etc) van andere frameworks (LeSS, Spotify, etc) *zoveel als mogelijk* te vermijden, dit vertraagt de change en implementatie van SAFe. Laat de LACE alle taaluitzonderingen monitoren, identificeren, vragen naar een onderbouwing en bijhouden in een Vocabulary-uitzonderingen-log om de inwerktijd van externen te verkorten.

27. Een Target Operating Model (TOM): het fundament onder elke complexe en grootschalige SAFe-transformatie



De afgelopen weken kreeg ik in verband met de transformatie waaraan ik toen werkte steeds vaker de vraag:

Wat is een Target Operating Model (TOM) precies? Wat zit daarin?
Hoe is een transformatieplan exact opgebouwd?
Waar komen deze concepten vandaan?
En waarom is het zo'n cruciaal fundament voor een SAFe-transformatie?

In dit speciaal daarvoor gemaakte hoofdstuk geef ik alle vraagstellers, en uiteraard alle SAFe-geïnteresseerden, een compacte en toegankelijke uitleg, met bijzondere aandacht voor hoe een TOM organisaties helpt om van strategie naar uitvoering te komen. Niet als theoretisch model, maar als praktisch kompas dat richting geeft aan verandering. En ook papier produceren versus direct beginnen.

Wat is een TOM in een notendop?

Veel organisaties hebben een sterke strategie én goedlopende processen, maar worstelen om die twee werelden daadwerkelijk te verbinden. Daar biedt het **Target Operating Model (TOM)** uitkomst.

Een TOM is de brug tussen **strategie (waarom)**, **waardecreatie (wat)** en **operatie (hoe)**. Het beschrijft hoe de organisatie in de toekomst zou moeten werken om strategische thema's en OKR's te realiseren. Daarmee vertaalt een TOM visie naar een concreet werkend model: *wie doet wat, wanneer en met welke middelen*.

Een TOM kan de basis vormen voor een **SAFe-implementatie**, maar is net zo goed toepasbaar bij een ERP- of SAP-implementatie, of het ontwerp van een volledig nieuwe organisatie. In de management consulting-wereld wordt dit ook vaak de **Future State** genoemd.

Een TOM is *niet* alleen een **structuurdocument**, maar bovenal een **veranderverhaal, een Elevator Pitch** met economische onderbouwing en continue verankering van structuur, cultuur, governance, economie, adoptie.

Wat is een COM?

Een **Current Operating Model** (COM) geeft weer hoe de organisatie vandaag functioneert: processen, structuur en rollen, technologie, cultuur, etc. Het COM is het **nulmetingspunt** van iedere verandering. Een ander veelgebruikt woord voor COM is '**current state'.**

Binnen een SAFe-transformatie is het COM cruciaal. Het COM vormt de basis voor de uit te voeren **gap-analyse**: door de huidige situatie te vergelijken met het gewenste TOM wordt helder welke transformatie nodig is alsmede een inschatting van de omvang.

Belangrijk is dat SAFe-experts daarbij onderbouwde antwoorden geven op vragen als:

• Hoe gaat SAFe t.o.v. de COM bijvoorbeeld de productiviteit verhogen?

• Hoe draagt de TOM bij aan bijvoorbeeld foutreductie en hogere klanttevredenheid ten opzichte van de COM?

• Welke impact heeft SAFe op PRINCE2-rollen als PM of PMO?

• Welke impact heeft SAFe op executive management, op de werkvloer, op middenmanagement?

• Welke problemen gaan we niet oplossen met de TOM (andere leaseauto's etc)

• Etc, etc

Transparantie is essentieel: SAFe kan veel mogelijk maken, maar niet alles. Verwachtingen moeten door de SAFe experts realistisch worden gemanaged.

Waar komen deze concepten vandaan?

Het concept van een TOM vindt zijn oorsprong in de management consulting-wereld (McKinsey, EY, PwC, Deloitte) in de jaren '90 en 2000. Het werd ontwikkeld als instrument bij grote transformaties, zoals fusies, digitalisering en internationale expansies. Inmiddels wordt het breed toegepast, onder meer in overheid, defensie, ABN AMRO, financiële dienstverlening en energie. Het is ideaal voor grootschalige, high-impact, complexe SAFe transformaties.

De belangrijkste voordelen van een TOM

Hieronder een niet-limitatieve opsomming van de voordelen van het ontwikkelen van een TOM:

• **Leesbaarheid en helderheid:** de nieuwe manier van werken wordt concreet en begrijpelijk voor iedereen. Bij transformaties met 500+ agilisten moet je in grote plenaire sessies helder kunnen pitchen welke kant het opgaat.

• **Direct goed specifiek (SAFe) opleidingsmateriaal**: een TOM is besluitvormingsstuk, handleiding en trainingsdocument in één. Het helpt de LPM-laag, ART's en agile teams te trainen in de nieuwe werkwijze.

• **Just enough upfront thinking:** het dwingt de organisatie, SAFe experts, eerst goed na te denken en te ontwerpen, voordat men 'agile losgaat' met alle gevolgen van dien.

• **Besluiten goed vastgelegd:** vragen vanuit alle lagen van de organisatie kunnen steeds worden beantwoord met verwijzing naar het TOM.

• **Eén waarheid, één verhaal:** Iedereen praat vanuit hetzelfde toekomstbeeld. Dit voorkomt ruis, discussies en misverstanden tussen directie, management, teams en externe partners.

• **Continu leer- en verbeterinstrument:** Een goed onderhouden TOM groeit mee met de organisatie. Het fungeert als referentiekader bij inspecties, retrospectives en Inspect & Adapt-workshops.

• **En tot slot: concreet toetsingskader voor metrics:** Flow-, outcome- en financiële metrics krijgen betekenis omdat ze gekoppeld zijn aan het toekomstige ontwerp. Want hoe weet je als SAFe expert anders, ook naar je opdrachtgever toe, dat je meters hebt gemaakt en/of de Current State problemen hebt opgelost? Meten=Weten,

gissen=missen en gokken=dokken! Toch?

Een goed en volledig (SAFe) transitieplan kent doorgaans deze drie lagen

Laag #1: Het COM (huidige situatie / current state)

• Inventarisatie van zowel hard als zacht

• Organisatiestructuur, processen, tooling en cultuur

• Resultaten van audits, onderzoeken, enquêtes (current state assessment)

• Inventarisatie van applicaties, systemen en rollen

• Root-cause analyses van problemen (bij voorkeur MECE geordend)

• Huidige cultuur, houding en gedrag, hoe men ‘nu de dingen doet’

• En zeker nog een paar topics…

Laag #2: Het TOM (toekomstig ontwerp / future state)

• Ontwerp van zowel hard als zacht (dus 2 transities)

• Doelen en meetpunten

• Verandervisie (‘we gaan organiseren rondom waarde’, we gaan naar vaste long-lived agile teams etc')

• Leidende ontwerpprincipes (heel belangrijk!).

• SAFe icm eigen opgedane transitie best practices en lessons learned.

• Ontwerp van Kotters Dual OS en details hoe OS #1 en OS #2 gaan samenwerken en hoe hun verantwoordelijkheden worden afgebakend.

• Governance tot in detail uitgewerkt zodat duidelijk is wie waarover gaat, hoe besluiten worden genomen en splitsing tussen OS #1 en OS #2. Hoe SAFe Lean-Agile principles worden toegepast.

• De nieuwe ‘(SAFe) Organisatiestructuur’ (Gebruikers Org, LPM, ART’s, Agile teams, topologies en OS #2 rollen (RTE, SM, PO, PM, SA, LPM, etc.)

• Processen (CICD/CDP), Value Streams (OVS, DVS) met output van workshops als VSAIW, VSM.

• Aansluiting bij externe regelgeving, compliance, security of internationale (militaire) standaarden (ISO, ITIL, NIST, AVG, etc.)

• Tooling, data.

• Metrics (flow metrics, outcomes metrics, financiële metrics, OS #1 metrics / OS #2 metrics).

• Cultuur, waarden, innovatie, learning culture, een nieuwe leiderschapsstijl en gedrag.

• En zeker nog een paar topics…

Tip: Denk ingeval van SAFe transformaties om bijvoorbeeld de SAFe-competenties te ontwerpen (Agile Product Delivery, Team & Technical Agility, Continuous Learning Culture, Enterprise Solution Delivery, Lean Portfolio Management, etc) aangevuld met de benodigde artifacts per SAFe Snake Roadmap-stap zoals 'start een LACE'. Beschrijf deze competenties, dan ben je al een heel eind met een TOM. Voordeel is tevens dat iedereen naar een SAFe training kan en direct de TOM ‘kan lezen en begrijpen’

Laag #3: Implementatiestrategie

1. Hard en Zacht (hoe gaan de mensen de verandering omarmen) (meebeslissen of meedoen of meedenken of meeweten?)

2. Ontwerp en oprichting van een LACE (in verband met een VMO) (als al niet gedaan om de TOM te ontwerpen)

3. Een transitie backlog / Prepare for ART Launch etc

4. Opleidingen

5. Aanpak: Big bang of MVP. Evolutie, Revolutie

6. Veranderaanpak (voordoen, samen doen, zelf doen, mensen-meenemen, etc)

7. HRM plaatsingsplan OS #1 -> OS #2 en besluitvorming rondom plaatsing

8. Transitie Roadmap

9. Verwerking Lessons Learned en ophalen feedback

10. Hoe de TOM wordt beheerd, door wie en wanneer wordt geüpdatet. (anders verdwijnt deze wel in een la!)

11. Business case: kosten/Baten, economische onderbouwing

12. En zeker nog een paar topics…

Praktijkervaring

Bij bijvoorbeeld mijn opdracht bij Defensie hebben we met de LACE, ervaren SAFe-experts en tientallen Defensiemedewerkers een solide, naleesbaar en gedetailleerd TOM ontwikkeld. Het fungeerde als fundament voor de transformatie, gaf houvast en een gedeeld toekomstbeeld. Bovendien bleek het een krachtig opleidingsinstrument: uiteindelijk werden er vele Defensie-agilisten getraind en voorbereid op de nieuwe werkwijze. De volledige TOM bestaat in de praktijk voor organisaties met 500+ agilisten vaak uit circa 250 hoofdslides en 100 bijlageslides, aangevuld met een gedetailleerd CDP ontwerp (Systems Engineering, Testing, Ketenintegratie, suppliersstrategie en management), en nog aangevuld met ~200 ontworpen core SAFe Artifacts zoals ‘alle SAFe rolbeschrijvingen’, die bijvoorbeeld klaar moeten zijn *voor* je de ART launch. (En mogelijk ook nog langs de OR!). Voorts ontwerp van je eerste PI event, I&A, Draft en Final plan Review pitch opzet, Alle team events, capacity allocation rekenmodellen, indepth ontwerp van de Agile Cadence (LPM level, ART level en Teamlevel), je DoD, etc, etc.

Voordat je definitief van start gaat, moet dat er (eigenlijk en liefst) allemaal wel zijn om direct goed te kunnen starten. Maar learning by doing, opschrijven wat je hebt gedaan en merkt dat het werkt, mag en kan ook.

Een dilemma voor agilisten: papier produceren versus snel waarde leveren

Agilisten hanteren, als het goed is, allemaal het Agile Manifesto principe “*Working software over comprehensive documentation.”*. Dan krijg ik vaak de vraag: waarom zouden we dan tijd en energie steken in het ontwikkelen van een Target Operating Model met veel papier? Goede en terechte vraag!

Het antwoord is dat een TOM geen papieren tijger is en ook geen statisch rapport dat in een la verdwijnt. Een TOM is een **levend kompas** dat richting geeft aan de manier waarop de organisatie werkt en samen waarde creëert. Het biedt *net genoeg* structuur om duidelijkheid te scheppen, zonder de flexibiliteit en wendbaarheid van Agile te ondermijnen.

Een TOM is daarmee geen ballast, maar juist een hulpmiddel dat naadloos aansluit bij de Agile-principes. Het biedt een gedeeld vertrekpunt, maakt rollen en verantwoordelijkheden inzichtelijk en zorgt dat iedereen begrijpt hoe de organisatie rond waarde is ingericht. Zo voorkom je dat Agile verwordt tot rituelen, en maak je de stap naar échte resultaten.

En laten we eerlijk zijn: een SAFe-transformatie is een ingrijpend en zeer kostbaar traject (zomaar een paar miljoen euro). Zo’n high-impact verandering kun je simpelweg niet uitvoeren zonder iets op papier te hebben dat richting geeft, de samenhang bewaakt en ten grondslag ligt aan executive besluitvorming. Elke goede SAFe expert zal altijd het ‘just enough' principe nastreven, dus 'net zoveel papier als nodig’. En zo snel mogelijk

starten!

Tips & advies

Een SAFe-implementatie zonder TOM is mogelijk, maar brengt vooral in grote organisaties met een LPM laag en multiple ART’s aanzienlijke risico’s met zich mee. Zonder duidelijke keuzes kunnen OS #2 en OS #1 gaan zweven en alle kanten op bewegen. Met een solide en beschreven TOM leg je de brug tussen strategie en uitvoering en voorkom je dat SAFe verzandt in rituelen op teamniveau. In plaats daarvan borg je dat de organisatie daadwerkelijk waarde levert op alle niveaus en is de TOM voor iedereen, elke dag naleesbaar. En kan iedereen over elk detail (goede) vragen stellen en de SAFe experts scherp houden: is werkelijk aan alles gedacht?

Richt je TOM daarom altijd op het leveren van waarde en zorg dat de cruciale metrics vanaf dag één aanwezig zijn. Alleen dan kun je de transformatie van de huidige situatie (IST) naar de gewenste situatie (SOLL) onderbouwen en sturen.

28. Beoordelingskader: wanneer vinden wij dit een écht goed veranderplan?

Dit beoordelingskader heb ik opgesteld en toegepast bij klanten en organisaties waar ik opdrachten heb uitgevoerd. Het beschrijft wanneer wij, als opdrachtgever, LACE member of manager, een veranderplan een echt goed plan vinden.

Hieronder staan mijn acceptatiecriteria.

1. Klantspecifiek en contextgebonden

Het plan moet duidelijk geschreven zijn voor de context van de klant en niet generiek of ‘consultancy-achtig’ waar ik altijd bang voor ben bij de BIG5. *'Mooi duur plan! Maar wat moet ik morgen dan anders gaan doen, en doe ik dat niet, wat dan?'* Het moet onmiskenbaar toepasbaar en geschreven zijn op en voor de betreffende organisatie, haar onderdelen, OS1/OS2 en haar cultuur, niet iets dat ook bij een willekeurige andere organisatie kan worden neergelegd.

Een veranderkundige werkt tussen andere lopende trajecten en andere veranderaars in de organisatie. Het plan maakt duidelijk wie wat doet, hoe dat samenspel verloopt, wat de eigen rol is en die van anderen, en welke randvoorwaarden gelden.

2. Concreet, uitvoerbaar, geen theoretisch consultancy-document

We verwachten een *concreet* plan van aanpak, met duidelijke activiteiten, interventies, verantwoordelijkheden, deliverables en uitvoerbare stappen. Geen abstract kader of theoretisch model, maar een executable transformatieplan dat morgen in werking kan worden gezet waar ik kan zien ‘wie er wat wanneer hoe gaat doen’

3. Aannemelijk dat de doelstellingen worden gehaald

Na het lezen moet volledig helder zijn hoe het plan de gestelde doelstellingen realiseert en onder welke voorwaarden de opdracht van de veranderkundige succesvol zal zijn. Waar dit niet het geval is, moet expliciet zijn waarom niet, en welke risico’s of gaten bestaan. Ik wil KSF'en zien, ook weer het liefst met een wetenschappelijke onderbouwing op basis van onderzoek: Waarom falen cultuurverandertrajecten? Zijn die faalfactoren gemitigeerd?

4. SAFe 6.0-conform en Lean-Agile onderbouwd

Het plan moet aantoonbaar geïntegreerd zijn met SAFe 6.0 en aantoonbaar gebruikmaken van (cultuur, gedrag, mindset) concepten zoals:

• Lean-Agile Leadership

• Growth mindset

• Agile Manifesto

• SAFe principes (decentralized decision-making, Cadence & Synchronization, relentless improvement)

• Dual Operating System (OS1/OS2)

• ART, LACE, E-LACE, VMO, RTE, I&A, Cadence, flow
Het taalgebruik moet SAFe-correct zijn, zonder inconsistenties of terminologische fouten.

5. Gedrag per rol expliciet beschreven

Ik wil duidelijk zien welke rollen er zijn en welk gedrag gewenst, ongewenst en kritisch is. Denk aan: management (OS1), afdelingshoofden, sectiehoofden, teamleads, RTE/PO/SM, Agile teams. Dit moet concreet zijn: wat moeten zij *stoppen*, wat moeten zij *starten*? Zie ook de IST

6. Duidelijke IST-diagnose: hoe is de huidige situatie vastgesteld?

Het plan moet laten zien hoe de veranderkundige de IST heeft bepaald: interviews, surveys, observaties (PI, VSAIW, syncs), documentreviews, maturity-assessments. We willen weten: *wat, waar, hoe, wanneer* en hoe de veranderkundige “het net heeft opgehaald”. Een samenvatting van de IST zou mooi zijn.

7. Een tijdlijn / roadmap die realistisch is binnen de organisatie en de SAFe-cadence

Er moet een duidelijke planning zijn, gekoppeld aan PI’s, iteraties en SAFe-ceremonies. Niet als losse tijdlijn, maar geïntegreerd in onze Cadence.

8. Onderscheid tussen veranderplan, communicatieplan en trainingsplan

Het plan moet helder zijn over wat veranderinterventies zijn, wat communicatie is en wat trainingen zijn, en hoe deze elementen op elkaar aansluiten zonder te vervloeien.

9. Interventies op gedrag moeten expliciet en concreet zijn

Ik wil graag lezen en zien hoe en wie:

• gaat interveniëren bij oud gedrag

• wie intervenieert (de veranderkundige, LACE, teamleads, SM’s)

• hoe en wie op de werkvloer aanwezig is/zijn (VSAIW, PI, ART-events, Gemba-walks) en wat hij/zij daar dan exact als veranderkundige doet?

• hoe de veranderkundige weerstand zichtbaar maakt en doorbreekt

• Het moet duidelijk worden waar de veranderkundige *van is* en waar niet (scope).

10. Wetenschappelijke onderbouwing
Het plan moet aantonen op welke veranderkundige theorieën het is gebaseerd: denk aan stromingen als Senge, William Bridges, De Caluwé, Kotter, KPMG-transformatiemodellen. Deze theoretische basis geeft betrouwbaarheid en duiding. Kotter heeft bijvoorbeeld een duidelijk veranderstappenplan. Zie ik dit soort duidelijke stappen terug met ref naar marktgekende principes, frameworks, filosofieën?

11. Aansluiting op bestaande instrumenten van de organisatie
Het plan moet logisch passen binnen bestaande kaders zoals:

• P&C-cyclus

• HRM-systemen (Ik wil bijvoorbeeld Agile HRM concepten terugzien en lezen)

• opleidingen (leerhuizen, interne opleidingsinstituten)

• communicatieprocessen

• besluitvorming in OS1 (hoe helpen lijnmanagers, waar staat dat?)

• We willen geen plan dat buiten bestaande structuren om vliegt. Het moet ook specifiek geschreven zijn voor deze organisatie en onze situatie

12. SAFe-verbonden interventies, gedragingen en structuren
Alle interventies moeten herleidbaar zijn naar SAFe-waarden, SAFe-gedrag en het Lean-Agile gedachtengoed.

13. Correct gebruik van SAFe-terminologie en geen strijdig gedrag
Het plan mag geen termen of werkwijzen bevatten die in strijd zijn met SAFe (zoals command-and-control, upfront planning, projectmatige governance, statische structuren), tenzij het plan er juist naar verwijst als 'oud gedrag'.

14. Kritische doelgroepen helder en concreet beschreven
We willen zien waar de échte veranderuitdaging zit: niet bij de sponsors, die willen allemaal wel, niet bij de top

van de organisatie, maar bij de doelgroepen waar cultureel gedrag moet verschuiven. Waar zit de core doelgroep? Het gaat om concreetheid.

(let op stakeholdersmanagement comm mapping is wat anders dan veranderkundige mapping, ander doel)

15. Participatiemodellen en commitment-niveaus zijn zichtbaar

Ik verwacht participatie - en commitment woordjes als :

• voordoen, samen doen, zelf doen

• meebeslissen, meedenken, meeweten, meedoen.

• Het plan moet duidelijk beschrijven hoe participatie wordt vormgegeven en commitment wordt opgebouwd. Mijn plan= ons plan, mijn transformatie=onze gezamenlijke transformatie. Niet jullie/wij, maar wij samen

• ook zaken als ‘change the system and you will change the culture’, SAFe’s psychologische veiligheid, motivatieprincipes (autonomie), etc

16. Duidelijke gedragsstoplijst + startlijst
Het plan moet aangeven welk ongewenst gedrag binnen de organisatie moet stoppen en welk nieuw gedrag moet starten om OS2-werken mogelijk te maken.

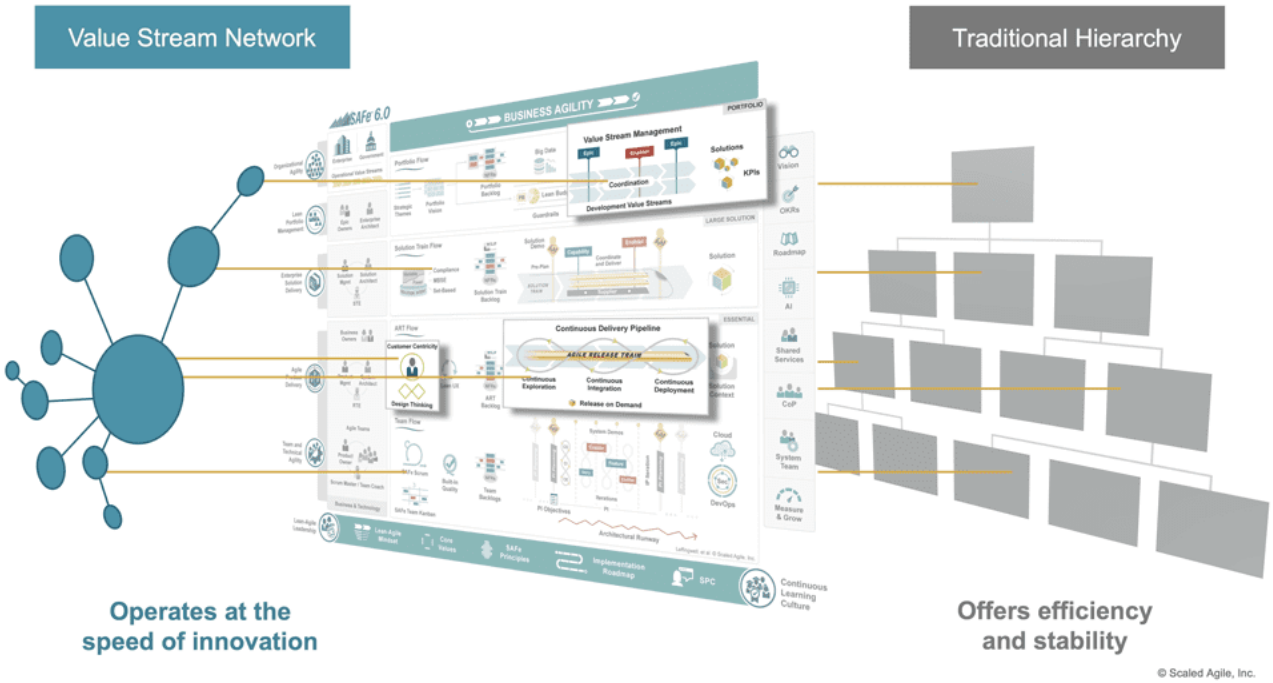
17. Meetbare indicatoren voor gedragsverandering

Het plan moet meetbare SMART-succescriteria bevatten die aangeven wanneer het plan succesvol is en de opdracht is afgerond. Wanneer is de cultuurverandering geslaagd? Concreet, OKR's, KPI's, metingen. Regulier meten, liefst in PI Cadence. Hoe, wat, waar?

29. Transformeren naar SAFe: wie is waarvoor verantwoordelijk? (deel 3)

Inleiding

In veel traditioneel georganiseerde organisaties ligt **accountability** meestal bij de lijnmanager of een project- of programmamanager. Die persoon heeft formeel mandaat, budget en bevoegdheid om beslissingen te nemen en resultaten af te dwingen. Hij of zij kan ingrijpen om doelen te behalen: *one line of control*. Dit past bij het verticale, top-down (militaire) besturingsmodel waarin verantwoordelijkheden via de lijn lopen.



Links OS #2 en rechts OS #1

OS #1: De makkelijke kant

In OS #1 wordt die controle en besluitwens nog steeds gehonoreerd. Managers, afdelingshoofden zijn hier hiërarchisch verantwoordelijk, hebben het mandaat, het budget en de positie om besluiten te nemen en zaken te regelen. Er is één aanspreekpunt, conform de herkenbare, klassieke structuur. Dit denkpatroon is volkomen logisch, maar het komt voort uit de “oude factuur- en rapportagemindset”: één baas, één beslisser, één lijn.

Die werkwijze heeft jarenlang goed gewerkt, maar in een SAFe-omgeving werkt ze niet meer optimaal. De overgang naar OS #2 vraagt een andere manier van denken: van controleren naar samen sturen, van eigendom bij één persoon naar gedeeld eigenaarschap.

Dan is de vraag dus: *naar wie loop je toe met betrekking tot accountability in SAFe, in OS #2?*

OS #2: De complexere kant

In SAFe werkt het nét anders. Zoals ik vaak tijdens SAFe workshops toelicht: SAFe gaat uit van **decentralized decision-making** (*Lean-Agile principe #9*).

Dat betekent dat beslissingen zoveel mogelijk worden genomen daar waar de kennis zit. De accountability verschuift dus van één centrale manager naar meerdere rollen met gedeelde verantwoordelijkheid. Dat vraagt

om vertrouwen, transparantie en duidelijke kaders, maar levert uiteindelijk meer wendbaarheid en eigenaarschap op.

SAFe kent 3 vormen van authority verdeeld over 3 rollen

SAFe kent geen programmamanagers of projectmanagers die zelf ‘accountable zijn’ (of is de stuurgroep dat?), omdat men ervan uitgaat dat geen enkel individu alles kan overzien. In OS #2, het value stream network, is accountability verdeeld over 3 rollen die samen waarde leveren:

• **Process authority**: bij de **Release Train Engineer (RTE)**: verantwoordelijk voor het faciliteren van de ART-events, het bewaken van de flow en het wegnemen van impediments. Loop naar hem/haar als de trein niet loopt: de flow stokt, events verlopen niet goed, of afhankelijkheden en risico’s blijven liggen.

• **Content authority**: bij de **Product Manager (PM)**: verantwoordelijk voor visie, prioriteiten en economische keuzes.

• **Technical authority**: bij de **System Architect (SA)**: verantwoordelijk voor technische richting en samenhang. Loop naar hem/haar als systemen niet schaalbaar zijn, responstijden te lang zijn of andere NFR’s niet worden gehaald. Of als de architectuur faalt.

Een ‘SAFe fabriek’ volgt de verkeerde koers: het Portfolio Leadership Team (LPM)

Op SAFe portfolioniveau ligt de **ultimate accountability** bij het **Portfolio Leadership Team** binnen **Lean Portfolio Management (LPM)**. Het LPM-leiderschap doet in essentie twee dingen:

1. **Bepalen van de strategische richting:** zij vertalen de organisatiedoelen naar portfoliostrategie, investeren in de juiste initiatieven en bewaken de alignment tussen alle value streams en de bredere ondernemers-strategie. Dit wordt onder meer besproken tijdens de Strategic Portfolio Review.

2. **Verantwoordelijkheid dragen voor de transformatie:** zij zijn via een speciaal Agile Team, het LACE, accountable voor het realiseren van die strategie, door de organisatie te helpen wendbaar te worden en effectief waarde te leveren.

Wie daarvan is dan ultimate accountable? Een bepaalde lijnmanager? Dat voelt niet zuiver: deze persoon is in bijna 90% van de gevallen niet de enige eigenaar van de bedrijfsstrategie, en ook niet van de Epics.

De ‘SAFe fabriek’ loopt niet: de rol van de LACE

Binnen SAFe maakt het Lean-Agile Center of Excellence (LACE) formeel onderdeel uit van het Lean Portfolio Management (LPM). De LACE fungeert als facilitator en uitvoeringsorgaan voor het Portfolio Leadership Team: de verbindende schakel tussen strategie en uitvoering. Kerntaken: Het faciliteren van de transformatie. Zoals het ervoor zorgen dat SAFe-principes, waarden en werkwijzen goed worden ingevoerd, geborgd en continu verbeterd. De LACE begeleidt teams, coacht leiders en verwijdert obstakels voor flow en wendbaarheid. Kiest tooling, stelt standaarden en richtlijnen op, bevordert Communities of Practice. Doet voor. Met andere woorden: het LPM bepaalt de koers, en de LACE zorgt dat de organisatie die koers daadwerkelijk kan varen.

Wie is dan degene die accountable is ingeval van het LACE? Formeel valt de LACE onder LPM, dus dan kom je uit bij de key decision makers op LPM niveau. Het eerlijke antwoord is dus: vaak is er niet één persoon. Op LPM-niveau is accountability in de meeste organisaties een gezamenlijke verantwoordelijkheid van een netwerk van gelijkwaardige leiders uit diverse business units, afdelingen, secties.

Conclusie

SAFe vraagt niet alleen een andere structuur, maar vooral **een andere mindset**. Waar OS #1 één aanspreekpunt kent met formeel mandaat, werkt OS #2 volgens gedeeld eigenaarschap en gedeelde verantwoordelijkheid. **De echte transformatie zit dus in het denken**: weg van de reflex om één verantwoordelijke aan te wijzen, naar het besef dat duurzame resultaten ontstaan door gezamenlijke accountability binnen een netwerk van professionals.

Het doel is niet om *minder verantwoordelijkheid* te creëren, maar om verantwoordelijkheid dichter bij de bron te leggen, daar waar de kennis, het eigenaarschap en de energie zitten.

Accountability in SAFe is dus geen “één persoon die de baas is”, maar een systeem van gedeelde verantwoordelijkheid dat zorgt voor betere beslissingen, meer eigenaarschap en snellere waardecreatie.

Elke organisatie moet zelf de SAFe theorie aanpassen aan haar eigen context.

Deel 4: Leiderschap, cultuur en mindset

30. De essentiële kenmerken van een ware Agilist

Ik hoor regelmatig binnen veel organisaties dat men ‘al Agile werkt.' Echter, bij het woord 'Agile' gaan bij echte Agilisten vaak diverse alarmbellen af. Want wat bedoelen collega’s eigenlijk als ze dat zeggen? Welke mindset hebben ze dan? Welke werkwijze? Wat doen ze overdag dan zoal? Als je tien mensen vraagt *“Ik wil een super-Agilist zijn, wat moet ik morgen doen?”* zul je waarschijnlijk tien verschillende antwoorden krijgen. Dit komt doordat het best een complexe vraag is, vooral voor degenen die niet diep in de SAFe-materie zitten, geen officiële diploma’s hebben gehaald of de absolute basis van SAFe nog onvoldoende hebben geleerd.

De vraag wordt nog ingewikkelder als je vraagt: 'Kan ik Scrummen zonder Agile te zijn en kan ik Agile zijn zonder te Scrummen? Het punt hierbij is dat Scrum vaak wordt geassocieerd met Agile vanwege de iteratieve aanpak, maar het is toch echt mogelijk om Scrum te implementeren zonder echt Agile te zijn. En andersom. De eerste vorm noem ik vaak "mechanische Scrum-implementaties".

Gezien de bovenstaande intro leek het me daarom een goed idee om jou vandaag mee te nemen in de vraag: 'Wanneer is iemand *echt* Agile?' Zodat jij straks die vraag goed kunt beantwoorden.

Rest me je heel veel leesplezier toe te wensen!

Reacties die geen goed antwoord zijn op de vraag

Hieronder een greep uit vaak gehoorde reacties, maar die geen echt antwoord geven op de vraag:

• Dat de persoon met Jira, Azure DevOps of met een Kanban-board werkt.

• Dat de persoon elke dag keurig een Daily Standup ('Team Sync') houdt.

• Dat er ScrumMasters en ProductOwners rondlopen.

• Dat ze in Sprints werken.

• Dat ze in ‘Agile Teams werken’.

• etc

Waar focus ik op bij het luisteren naar het antwoord

Als iemand antwoord probeert te geven, observeer ik meestal of de persoon verwijst en refereert naar de *onderliggende Agile waarden en principes* en deze ook daadwerkelijk in de dagelijkse praktijk omarmt en top of mind heeft. Zoals bijvoorbeeld het leveren: en maximaliseren van waarde, iteratief, flexibiliteit, cross-functionele samenwerking, zelfreflectie en aanpassingsvermogen.

De fundamenten van Agile *denken en werken* zijn vastgelegd in het **Agile Manifesto**. Dit Manifesto is van 11 februari tot 13 februari 2001 opgesteld in The Lodge van het Snowbird Ski Resort in de bergen van Utah, door een groep van 17 softwareontwikkelaars. Onder hen zijn mogelijk twee bekende namen voor de Scrummers: Jeff en Ken, de bedenkers van Scrum, zie onder foto. Zij hebben Scrum reeds in de vroege jaren van 1990 ontwikkeld nog voor het Agile Manifesto. Weet dat even, veel mensen denken namelijk andersom.



Het Agile Manifesto legt de kernwaarden & principes vast die de Agile benadering (destijds primair gericht op softwareontwikkeling) vormgeven. Tegenwoordig wordt Agile toegepast op software, hardware, firmware en diensten, zo is SAFe op dit moment ook ingericht. Dit kun je o.a. zien door de Built-In quality principes na te lezen in SAFe. Het Manifesto bestaat uit vier waardeprioriteiten en twaalf principes, die hieronder in posterformaat te vinden zijn.



Fundamentele antwoorden wanneer iemand Agile is

Agile werken is dus niet alleen maar een set van technieken, meetings of tooling zoals Jira, Scrum, Lean of Kanban's; het omvat een verandering in de cultuur en vooral de mindset en aanpak om tot iets binnen een organisatie te komen. Veranderaars, cultuurexperts en agile coaches worden geacht te helpen deze culturele fundamentele verschuiving te verankeren door de nadruk te leggen op alle aspecten van het Agile Manifesto en daarop te coachen en te trainen. Zij moeten het Manifesto tot op het bot beheersen:

Als dus iemand de vraag stelt 'Wanneer is iemand Agile?' zouden onderstaande (niet-limitatieve) reacties prima antwoorden zijn:

• Als die iemand focust op samenwerking en communicatie tussen mensen in plaats van strikt te vertrouwen op tools of het strikt volgen van (niet lopende) processen

• Als het werk verricht wordt door cross-functionele, self-managing Agile Teams waarin samenwerking centraal staat en deze teams zich ontwikkelen naar long-lived High Performance Teams.

• Als hij/zij het werk opdeelt in kleine, beheersbare iteraties van bij voorkeur iets tussen de 1 en 4 weken.

• Als per iteratie, het team (in meest ideale geval!) Design, Build, Test & Deploy binnen 1 iteratie doet en direct werkende software oplevert.

• Als het agile team er alles aan doet om de afnemers van het product of de dienst zo snel mogelijk concrete waarde te leveren, en wel elke sprint. En die waarde levering elke sprint maximaliseert door slim te kiezen welke set van work items de iteratie het allermeeste oplevert. (PO's zijn Value Maximisers)

• Als het team denkt in Minimum Viable Products (M.V.P.s) en Minimum Marketable Features (M.M.F.'s) om de time to market/field zo kort als mogelijk te houden. Echte agilisten hanteren het mantra: 'Fail often, fail fast en fail early'.

• Als het team frequent economisch (her) prioriteert en doorrekent welke requirements in welke optimale volgorde moeten worden gebouwd. Als het team dus een mindset heeft om plannen en prioriteiten direct aan te passen op basis van nieuwe informatie en feedback. Scope changes 'omarmen', daar ook uitstekend mee om kunnen gaan en daarop zijn ingericht. Adapt to Change.

"Agile is a mindset. Agile is behaviour"

• Als de omgeving, het agile team niet denkt in het produceren van dikke documentatie ("comprehensive documentation") voor het scheppen van een bepaalde zekerheid, maar zich focust op 'Working Solutions'. Papierwerk zo lean als mogelijk houden, of parallel doen.

• Als het Agile team regelmatig feedback van klanten, gebruikers en stakeholders vraagt om het product of de dienst te verbeteren en zo zeker weet dat de gebruiker dat opgeleverd krijgt wat deze had verwacht. Communicatie prefereren boven het continue naleven van elke regel in het contract.

• Als alle teamleden open communicatie en vooral transparantie hoog in het vaandel hebben, waarbij groen = groen en rood = rood. Dat iedereen eerlijk is als er iets fout is gegaan met uitleg hoe ze dat volgende keer gaan voorkomen.

• Als de teamleden zich bewust zijn dat directe, persoonlijke communicatie doorgaans de meest efficiënte en effectieve methode van communiceren is, en actief kiezen voor de best passende communicatievorm per situatie. Zij weten dat non-verbale communicatie van dominante invloed is op de totale communicatie, en dat dit bij mailen of bellen wegvalt met gevaar voor miscommunicatie.

• Als alle teamleden samen, maar ook de teamleden op zichzelf, regelmatig reflecteren en continu verbeteren op Domeinen (1) Process, (2) Technology & (3) People. Een mindset hebben om elke dag een betere versie van zichzelf te creëren. Niet alleen reflecteren en 100 retro items aanmaken in Miro, maar 'frequent' op basis van Concrete Data met concrete Metrics die ze verzamelen, kiezen voor nieuwe gewoonten en nieuwe benaderingen met echte progressie en doorgevoerde aanpassingen.

• Als ze streven naar eenvoud in ontwerpen en oplossingen en onnodige complexiteit vermijden conform " Doe minder, maar beter".

• Als ze continu aandacht hebben voor technische uitmuntendheid en goede ontwerpen.

• In geval ze Agile Tooling gebruiken om Agile werken te ondersteunen en bevorderen. Er een bewuste keuze is gemaakt om te werken met SAFe, Scrum, Kanban, mixes of iets anders.

• Tot slot verwacht ik proactief gedrag, empathie & begrip, en het vermogen om holistisch te denken.

Denkvraag: wat is je antwoord?

Ik hoop dat je nu iets meer achtergrond hebt over wat je moet gaan doen en denken om een ware 'Agilist' te worden. Nu kom je morgen met een collega in gesprek over '*een ware agilist zijn en de inhoud van dit hoofdstuk dat je net hebt gelezen'.* Waarop je collega jou aanvult en zegt: "Echte Agilisten passen de Agile waardes en principes ook Agile toe!".

Wat zeg jij dan? Wat zou jouw reactie daarop zijn?

31. Eén organisatie, drie culturen: het echte vraagstuk achter SAFe-transformaties

Veel organisaties denken dat een SAFe-transformatie vooral een procesverandering is.
In werkelijkheid ontstaat er iets anders: één organisatie met drie verschillende culturen.

Inleiding

LACE-teams (Lean-Agile Center of Excellence) zijn het primaire orgaan dat SAFe-transformaties begeleidt. Het implementeren van SAFe brengt vrijwel altijd een significante verandering in mindset, gedrag en werkwijze met zich mee. SAFe vraagt om meer wendbaarheid, transparantie, ownership, samenwerking, gebruikersgericht denken, minder documentatie en gedelegeerde besluitvorming. De inhoudelijke kant van deze veranderingen heeft SAFe al goed beschreven.

Organisaties die agile transformeren, hebben echter nog steeds te maken met een bestaande organisatiecultuur: de hiërarchie die onderdeel is van Kotter's dual operating system (OS #1), zie hoofdstuk 24 en 25. Dit OS is gericht op stabiliteit en controle, en bestaat straks naast het nieuwe SAFe-operating systeem (in Kotters termen het 2e OS) dat gericht is op snelle waardecreatie. Sommige organisaties werken bovendien in een hybride model en draaien naast SAFe ook nog een waterfall-based OS #3.

In dit hoofdstuk neem ik jullie mee in een aantal ideeën en nodig ik uit tot meedenken en bewustwording. Ik geef nog geen oplossingen of antwoorden.

Wat is een (organisatie)cultuur?

Voordat we verder gaan, is het belangrijk om scherp te hebben wat cultuur precies is. Cultuur is meer dan alleen 'mindset' of losse waarden zoals respect, openheid of wendbaarheid. Het is veel breder:

"Organisatiecultuur verwijst naar de (1) gedeelde waarden, (2) overtuigingen en (3) normen die het gedrag, de communicatie en de besluitvorming binnen een organisatie bepalen. Het omvat (4) dagelijkse gewoonten, (5) rituelen, (6) symbolen en (7) de taal die medewerkers gebruiken. Leiderschapsstijl speelt hierbij een belangrijke rol".

Of, simpeler en makkelijk te onthouden:

"*Cultuur is hoe wij hier de dingen doen."*

Als cultuur simpelweg betekent 'hoe wij hier de dingen doen', dan betekent de introductie van SAFe feitelijk dat er een tweede manier van werken naast de bestaande organisatiecultuur ontstaat.

We gaan de cultuur veranderen, maar welke cultuur precies, en van wat naar wat?

Met drie verschillende operating systems naast elkaar, OS #1 (hiërarchie), OS #2 (SAFe) en OS #3 (Waterfall), is het duidelijk dat we te maken hebben met meerdere culturen. Hieronder beschrijf ik de cultuur van elk in termen die als het goed is voor iedereen herkenbaar zijn.

Belangrijk om op te merken: ook binnen OS #1 bestaat er niet één cultuur. De hiërarchische lijnorganisatie kent meerdere lagen: een organisatiebrede cultuur, een subcultuur op het niveau van de business unit, en een verdere subcultuur binnen de specifieke IT-eenheid. SAFe komt daar als extra laag bij, wat de culturele complexiteit verder vergroot.

1. De lijnorganisatie (OS #1)

Waarden en overtuigingen: Stabiliteit, controle en voorspelbaarheid zijn de kernwaarden. Er is een sterk geloof in hiërarchie en autoriteit als sturingsmechanisme. De cultuur is gericht op het minimaliseren van risico's en het waarborgen van efficiëntie via duidelijk vastgelegde procedures en gezagsstructuren.

Gedrag en communicatie: Besluitvorming is top-down, met een duidelijke verdeling van autoriteit, tekenbevoegdheid en verantwoordelijkheid over de lagen van de organisatie. Communicatie verloopt formeel via vastgestelde kanalen: MT-overleggen, procedurele besluitvorming en planvorming. De gehanteerde taal is organisatiespecifiek, met bijbehorende afkortingen en jargon.

Leiderschapsstijl: Leiderschap is controlerend van aard. Het management stelt duidelijke richtlijnen en houdt toezicht op de uitvoering van taken volgens vastgestelde normen en procedures.

2. SAFe-agilisten (OS #2)

Waarden en overtuigingen: Flexibiliteit, aanpassingsvermogen, eindgebruikersgerichtheid en snelle waardecreatie vormen de kern. De overtuiging is dat agile teams autonoom moeten kunnen opereren om snel op veranderingen te reageren en voortdurend te verbeteren. Samenwerking en transparantie zijn fundamenteel. Er wordt één gedeelde taal gesproken: de SAFe-taal, aangevuld met organisatiespecifieke begrippen.

Gedrag en communicatie: De nadruk ligt op samenwerking tussen multidisciplinaire teams, met open en frequente communicatie. Besluitvorming is gedecentraliseerd (“decisions where the information is”), waardoor teams meer eigenaarschap krijgen en sneller kunnen inspelen op veranderende gebruikersbehoeften.

Leiderschapsstijl: Scrum Masters en RTE’s opereren faciliterend en ondersteunend, gericht op het empoweren van teams. Leiders fungeren als coaches die visie en richting aangeven, maar de uitvoering aan de teams overlaten. Agilisten verkiezen werkende software boven uitgebreide documentatie (Agile Manifesto). Leiderschapstaken zijn verdeeld over drie domeinen: Technologie (Solution Architect), Inhoud (Product Manager) en Proces (RTE). Traditionele projectstructuren zoals PID’s en klassieke projectmanagers maken in SAFe plaats voor een andere manier van sturen, gebaseerd op backlog-management, flow en value delivery.

3. Waterfall-mensen: projectmanagers en stuurgroepen (OS #3)

Waarden en overtuigingen: Plannen, zekerheid en controle staan centraal. De overtuiging is dat projecten het beste worden afgerond als alle stappen van tevoren zijn doordacht en vastgelegd. Langetermijnplanning en nauwgezette uitvoering (via projectkaarten, projectvoorstellen en PID’s) zijn essentieel.

Gedrag en communicatie: De focus ligt op formele besluitvorming via stuurgroepen en gestructureerde processen. Er is een sterke nadruk op uitgebreide documentatie, goedkeuringen en strakke procedures. Communicatie is gericht op voortgangsbewaking en het afvinken van vooraf bepaalde mijlpalen.

Leiderschapsstijl: Eén persoon bewaakt deadlines en budgetten en heeft een sterk sturend karakter. De waterfall-leider controleert de voortgang, bewaakt de planning en rapporteert bij afwijkingen. Hij of zij is het centrale aanspreekpunt voor inhoud, proces én technologie.

De subcultuurval: wanneer één eiland anders speelt dan de rest

SAFe wordt in de praktijk vrijwel nooit organisatiebreed gestart. De implementatie begint vrijwel altijd op het niveau van een business unit, subdivisie of IT-afdeling. Dat roept een fundamentele vraag op: wat betekent het als één groep een nieuwe (high performance) cultuur ontwikkelt, terwijl de rest van de organisatie nog volledig opereert vanuit OS #1? Het is vergelijkbaar met een golfteam waarbij een select groepje speelt op handicap 4, terwijl de rest van de club nog op handicap 33 staat. Ze spelen op hetzelfde parcours, maar volgens andere spelregels. Het ART werkt met gedelegeerde besluitvorming, een PI-ritme en faciliterend leiderschap, maar de omgeving eromheen verwacht escalatie, stuurgroepgoedkeuring en een klassieke projectmanager die rapporteert.

Dit levert drie concrete grensvlakken op die expliciete aandacht vragen:

1. Budget & governance: worden het ART en de teams nog gefinancierd via traditionele projecten en stuurgroepen, of is er al een beweging richting Lean Portfolio Management? Zolang de budgetcyclus OS #1-logica volgt, blijft het ART gevangen in een structuur die haaks staat op agile werken.

2. Leiderschap boven het ART: begrijpt en beschermt het middenmanagement boven het ART de agile werkwijze actief? Of verwacht men onbewust gewoon de oude rapportagestructuur, met de RTE in de rol van klassieke manager? Zonder bewust leiderschap op dit grensvlak wordt de subcultuurverandering van binnenuit ondermijnd.

3. HR & performance: worden mensen beoordeeld op agile gedrag (eigenaarschap, samenwerking, continu verbeteren) of nog op traditionele KPI's die passen bij OS #1? De cultuur die een organisatie beloont, is de cultuur die beklijft. Als deze drie grensvlakken niet meebewegen, blijft de subcultuurverandering structureel beperkt. Het ART speelt een ander spel, maar de scoreborden worden nog bijgehouden door mensen die denken dat ze een ander spel kijken.

Uitnodiging tot reflectie en open dialoog

SAFe implementeren verandert niet alleen aanpak en processen, maar raakt ook de cultuur van de organisatie. Het beïnvloedt mindset, besluitvormingsprocessen, leiderschapsstijlen en de onderliggende normen en waarden, kortom alle elementen van organisatiecultuur.

Als we tegelijkertijd SAFe, waterfall-projecten én hiërarchische structuren blijven hanteren, ontstaat een situatie waarin meerdere culturen naast elkaar bestaan. Maar is dat echt zo? Veel organisaties hebben immers al een overkoepelende enterprise-cultuur gedefinieerd, met gedeelde gedragingen en waarden. Is het dan mogelijk om één overkoepelende cultuur te formuleren die alle werkwijzen verbindt?

Stel dat een hoofd van een business unit actief is binnen de hiërarchie, vier dagen per week als RTE werkt in het Value Stream Network, én betrokken is bij drie stuurgroepen in een waterfall-context. Moet deze persoon voor elke rol een andere cultuur aannemen, steeds een andere pet op afhankelijk van de context? En wat als de cultuur die hij of zij heeft omarmd niet meer aansluit bij collega’s uit totaal andere organisatieonderdelen?

Of is het juist mogelijk om één overkoepelende cultuur te definiëren die aansluit bij alle werkwijzen? In dat geval zou een waarde als ‘werkende software boven uitgebreide documentatie’ breed worden toegepast, ook door waterfall-professionals, voor wie dit een fundamenteel andere manier van werken vraagt.

Een ander interessant gedachte-experiment: kun je de cultuur binnen één business unit veranderen, terwijl je tegelijkertijd volwaardig blijft acteren binnen de bredere enterprise met haar eigen, generieke cultuur?

De vraag is dus niet alleen hoe we SAFe implementeren, maar vooral hoe we omgaan met drie verschillende culturen die tegelijkertijd in dezelfde organisatie opereren.

Met bovenstaande observaties nodig ik alle lezers uit tot reflectie en een open dialoog over cultuurverandering en de veranderkundige aspecten van SAFe.

32. Hoe maak je SAFe-cultuur, houding en gedrag meetbaar?

Introductie

Een SAFe implementatie kent feitelijk 2 componenten: Een Harde en een Zachte component. Hard gaat primair over het ontwerp van het Target Operating Model, Zacht gaat primair over de gewenste cultuur, nieuwe manier van werken, bewustwording, team members meenemen, toekennen nieuwe rollen, etc.

Dit hoofdstuk bouwt voort op hoofdstuk 31, ‘Eén organisatie, drie culturen: het echte vraagstuk achter SAFe-transformaties’, en zoomt dieper in op juist de zachte kant van een Agile-transformatie: cultuurverandering, uitgewerkt met een concreet cultuurvoorbeeld. Wil je namelijk echt invloed uitoefenen op de cultuur binnen je business units of afdelingen, dan moeten cultuurveranderaars beginnen met het modelleren en meetbaar maken van de cultuurelementen die het management wil zien in een cultuur-definitie-model. Dit geheel moet worden ondersteund door een gedegen ingericht HRM-systeem.

Dit hoofdstuk vertrekt vanuit een Theorie Y-perspectief (McGregor): de aanname dat medewerkers intrinsiek gemotiveerd zijn, verantwoordelijkheid willen dragen en kunnen groeien, mits de organisatie de juiste randvoorwaarden schept. Cultuur meetbaar maken is vanuit dat perspectief geen controlemiddel, maar een groei-instrument: het geeft medewerkers en leidinggevenden een gedeeld kompas.

Nog even herhaling wat de definitie van cultuur is

Cultuur is niet alleen ‘je mindset’ of ‘waarden’ zoals respect, openheid of wendbaarheid. Cultuur is veel breder en wel:

Organisatiecultuur verwijst naar de (1) gedeelde waarden, (2) overtuigingen, en (3) normen die het gedrag, de communicatie en de besluitvorming binnen een organisatie bepalen. Het omvat (4) de dagelijkse gewoonten, (5) rituelen, (6) symbolen en (7) taal die medewerkers gebruiken. Leiderschapsstijl speelt ook een belangrijke rol in het vormgeven van de cultuur.

Of, nog veel simpeler en makkelijk te onthouden:

“Cultuur is hoe wij bij onze organisatie de dingen doen”

Parallelle culturen gevisualiseerd

Culturele coëxistentie	
Generieke organisatie cultuur	
PPM Waterfall cultuur	SAFe Lean-Agile cultuur
Business unit specifieke cultuur	
IT-afdeling specifieke cultuur	

Maak de nieuwe cultuur meetbaar voor de lijnmanagers

Om de cultuur van een organisatie daadwerkelijk te veranderen, moet cultuur niet alleen worden besproken in workshops, maar ook meetbaar worden gemaakt. Gedragingen, taal, aanpakken zijn immers de concrete uitingen van cultuur. Lijnmanagers hebben metrics nodig om te kunnen meten in welke mate de gewenste cultuur wordt nageleefd en waar men op een schaal van 1 tot 10 staat. Daarnaast is er training nodig voor lijnmanagers om dit goed te doen en te leren omgaan met de nieuwe metrics.

Building blocks van SAFe's cultuur

SAFe biedt een aantal bouwstenen die cultuurveranderaars kunnen gebruiken om cultuur te operationaliseren, meetbaar te maken, te trainen en te implementeren.

Enkele belangrijke elementen zijn:

• Het empirisme dat stelt dat kennis voornamelijk voortkomt uit ervaring en observatie;

• Agile Manifesto behaviors & principles: als 'Responding to change over following a plan, Working software over Comprehensive documentation'; The best way to convey information is Face to Face;

• De Scrum Values zoals Courage, Respect, Openness, Commitment en Focus;

• De vier SAFe Core Values: Alignment, Transparency, Respect for People en Relentless Improvement;

• Organisatie specifieke Values: wij werken samen, Wij zijn professioneel, Wij zijn betrouwbaar , Wij communiceren, Wij nemen verantwoordelijkheid en leggen verantwoording af;

• Business unit specifieke Values: Bijvoorbeeld Samenwerken, Professionaliteit, Openheid, Omgevingsgerichtheid en Resultaatgerichtheid;

• The House of Lean Pillars zoals Kaizen, pull in plaats van push, Pursue Perfection;

• SAFe's growth mindset als 'Failure is an opportunity to grow, challenges help me grow, etc';

• Lean Agile Leadership als 'Leading by Example, leading change, authenticity, growing others, decentralised decision making, non-hierarchical, etc.

Concreet voorbeeld: kernwaarde 'Professioneel'

Ik ga ervan uit dat alle medewerkers op zich best willen veranderen ten dienste van hun werkgever (McGregor Y-aanhanger). Maar als ik medewerker ben wil ik graag wel, concreet en meetbaar weten wat er van mij wordt verwacht en wanneer ik dan in de ogen van mijn werkgever, bijvoorbeeld 'professioneel ben'

Dit is een kernwaardedefinitie van een organisatie die ik ken:

"De medewerkers zijn in staat te innoveren en leveren hun bijdrage aan onze organisatie met hoogwaardige kennis en middelen. Onze organisatie is deskundig, zelfstandig, gemotiveerd en flexibel met personeel dat werkt vanuit een solide kennisbasis gericht op innovatie. Zij zijn integer, flexibel, leergierig en werken in het besef dat men bezig is voor de "mannen en vrouwen in het veld". De organisatie vertrouwt op de eigen verantwoordelijkheid van medewerkers en vakmanschap wordt gestimuleerd. Medewerkers krijgen de vrijheid van handelen door zelf binnen gegeven kaders tot beslissingen te komen. De organisatie zorgt goed voor het personeel met als doel een collegiaal, prettig en veilig werkklimaat"

Ik vind deze tekst niet alleen lastig te lezen en te begrijpen; hij is voor mij als medewerker ook onvoldoende vertaald in concrete meetbare gedrags- en werk elementen. Ook voor de lijnmanagers van die onderneming wordt het lastig met behulp van bovenstaande definitie mij 'te beoordelen' of ik wel professioneel genoeg ben.

Tegenvoorstel: dit is mijn definitie van het woord 'Professioneel':

Element 'Professioneel'	Eisen, normering en meting
Vakinhoudelijke expertise	Blijf up-to-date in je vakgebied en pas de juiste kennis toe. Meetbaar: Deelname aan trainingen en bijdragen aan kennisdeling.
Kwaliteit van werk	Lever consistent werk van hoge kwaliteit en vraag om feedback. Meetbaar: Max. 2 feedbackrondes per project, positieve evaluaties.

Element 'Professioneel'	Eisen, normering en meting
Betrouwbaarheid	Kom afspraken na, meld tijdig problemen, en neem verantwoordelijkheid. Meetbaar: 95% deadlines gehaald, proactieve probleemmelding.
Professionele houding	Toon respect, wees integer en houd je aan gedrags- en kledingcodes. Meetbaar: Tevredenheidsbeoordelingen, geen schendingen van gedragscodes
Communicatie	Communiceer duidelijk en respectvol, werk samen en geef constructieve feedback. Meetbaar: Positieve 360-graden feedback, actieve deelname in vergaderingen.
Zelfontwikkeling	Werk actief aan persoonlijke groei en vraag om coaching. Meetbaar: Minstens één ontwikkelingsgesprek per kwartaal, meetbare vooruitgang.

Voorbeeld modellering meetbare SAFe cultuur

Ik heb hierboven uit de definitie van Cultuur alleen het element *kernwaarden* eruit gehaald. Hieronder bouw ik een compact SAFe cultuurmodel op. Dit model kan vervolgens aangevuld worden met organisatiespecifieke kenmerken die behouden moeten blijven.

Even alleen ter ideevorming en als gedachte-experiment:

Cultuurelement	SAFe-Element	Meetbare indicatoren (incl normering)	Meting middels
Waarden en overtuigingen	Commitment (Scrum Value): Professionals nemen verantwoordelijkheid en committeren zich aan het behalen van hun doelen.	Percentage teams dat hun doelen binnen een PI haalt. Aantal retrospectieve acties om commitment te verbeteren.	Meten via PI doelen: Percentage behaalde doelen (bv. 80% of meer). Retrospectieve data.
	Working Software over Comprehensive Documentation (Agile Manifesto): De voorkeur gaat uit naar werkende oplossingen boven uitgebreide documentatie.	Aantal releases van werkende software binnen een PI. Kwaliteit van opgeleverde software (foutpercentage).	Aantal werkende software-opleveringen. Kwaliteitsbeoordelingen en klantfeedback.
	The Continuous Learning Culture (CLC)	Mate waarin medewerkers zijn gegroeid in kennis en competenties	aantal behaalde certificaten/jaar aantal uitgevoerde innovaties, hackathons
Normen	Autonomie, maar binnen kaders: Teams krijgen de vrijheid om zelf beslissingen te nemen binnen de afgesproken kaders van SAFe en de organisatie.	Mate waarin teams zelfstandig beslissingen nemen. Aantal escalaties naar hogere niveaus voor beslissingen.	Autonomiemeting via teamvragenlijsten. Escalatie-analyse per PI.
Gedrag en rituelen	We komen op tijd: Het naleven van de afgesproken tijdstippen voor meetings en ceremonies is essentieel voor discipline en respect binnen het team.	Aantal keren dat teamleden op tijd aanwezig zijn bij ceremonies. Aantal gemiste tijdsloten.	Survey over punctualiteit van teamleden. Analyse van tijdige start van ceremonies en meetings.
Symbolen en taal	SAFe-taal en concepten: SAFe wordt consistent	Consistentie in het gebruik van SAFe-termen in communicatie.	Interne audits of evaluaties van taalgebruik.

	toegepast met alle bijbehorende rollen en terminologie (RTE, PM, SA, ART, etc.).	Minder dan 5% gedocumenteerde uitzonderingen op de SAFe-begrippenlijst	Documentatie over afwijkingen van SAFe-taal.
	(*het kan bijna niet anders dan dat er ook PPM taal naast blijft)		
Besluitvormingsprocessen	SAFe Principle #9: Gedecentraliseerde besluitvorming: Beslissingen worden genomen op het niveau waar de meest relevante kennis aanwezig is, namelijk bij de teams zelf.	Aantal gedecentraliseerde beslissingen per PI. Tevredenheid van teams over de autonomie in beslissingen.	Evaluatie van besluitvormingsprocessen door teams. Aantal escalaties en overleg met management.
	Sturen op feiten en niet op roddels of vage observaties (heb horen zeggen dat…), want ‘Facts are Friendly’	Onderbouwde observaties, impediments, etc	Aantal onderbouwde/niet onderbouwde.
Leiderschap	Van ‘Deadlines en technische opties bespreken' naar 'Focus op businesswaarde’: Leiderschap verschuift van micromanagement naar het sturen op het leveren van klantwaarde.	Aantal keer dat leiders de focus op klantwaarde benoemen. Kwaliteit van communicatie over waarde in retrospect.	Feedback van teams op leiderschap tijdens PI en retrospectieven.
	Lead by Example: Leiders tonen het gewenste gedrag en dienen als rolmodellen door de waarden van SAFe uit te dragen. Walk the Talk.	Frequentie waarmee leiders SAFe-principes actief tonen. Observaties van teamleden over leiderschapsgedrag.	360-graden feedback van teams over leiderschap. Analyse van leiderschapsacties in lijn met SAFe-principes.

Toetsing van personeel op cultuurcompatibiliteit

Om de nieuwe cultuur effectief te implementeren, is het cruciaal om zowel bestaand personeel als nieuwe medewerkers te toetsen op hun fit met de gewenste cultuur. Dit kan worden gedaan door middel van een nulmeting en een evaluatieproces dat de culturele compatibiliteit van medewerkers in kaart brengt.

Nieuwe teamleden: Nieuwe medewerkers moeten bij voorkeur vanaf hun instap in de organisatie worden beoordeeld op hun match met de nieuwe cultuur. Dit kan al tijdens het wervingsproces plaatsvinden door vragen en scenario's te gebruiken die inzicht geven in hun culturele fit. Het doel is dat nieuwe medewerkers niet alleen passen binnen de organisatie, maar ook actief bijdragen aan de nieuwe cultuur en die bevorderen en zo direct vanaf binnenkomst al cultuurdrager zijn.

Uitnodiging tot reflectie en gesprek

Dit hoofdstuk sluit ik af met een aantal vragen die ik je meegeef: om zelf over na te denken, of om met een collega te bespreken. Er zijn geen goede of foute antwoorden, maar de vragen raken wel de kern van waar we als organisatie voor staan.

• **Hoe essentieel is cultuur voor een succesvolle SAFe-implementatie?** Kun je een Agile transformatie écht laten slagen zonder dat de cultuur mee verandert, of is cultuurverandering juist de doorslaggevende factor?

• **Eén cultuurmodel voor de hele organisatie: realistisch of naïef?** Is het überhaupt mogelijk om één gedeeld cultuurmodel te definiëren dat werkt voor alle business units en afdelingen? Of zijn subculturen onvermijdelijk, en misschien zelfs wenselijk?

• **Hoe ziet jouw ideale cultuurmodel eruit?** Welke elementen zouden daarin absoluut niet mogen ontbreken?

• **Het coëxistentieprobleem.** Stel: jouw team van 100 FTE ontwikkelt zich tot een topclub die op handicap 9,5 golft, terwijl de rest van de organisatie nog op GVB-niveau speelt. Hoe ga je dan om met die culturele kloof? Versnelt dat de transformatie, of creëert het juist wrijving?

• **Wat behoud je, wat laat je los?** Welke bestaande cultuurkenmerken zijn het waard om te bewaren, en welke moeten echt veranderen? En belangrijker: heeft de bredere organisatie daar ook de ruimte en het draagvlak voor?

33. Leiderschap in SAFe: schakelen tussen hiërarchie en waardestroomnetwerk

Goed Lean-Agile Leadership vormt het fundament van elke succesvolle Agile-transformatie. Het gaat niet primair om **management of controle**, maar om leiders die richting geven door voorbeeldgedrag, teams ondersteunen in hun ontwikkeling en een omgeving creëren waarin transparantie, samenwerking en continue verbetering centraal staan.

Wanneer organisaties starten met een SAFe-transformatie ontstaat er echter een nieuwe realiteit: de organisatie gaat werken met **twee parallelle besturingssystemen**. Deze twee systemen bestaan **niet als vervanging van elkaar**, maar **naast elkaar**. De hiërarchie creëert stabiliteit en randvoorwaarden; het waardestroomnetwerk creëert snelheid, flow en waardecreatie. Samen vormen zij het **Dual Operating System, dat je in een Target Operating Model (TOM) kunt uitwerken**.

Lean-Agile leiderschap betekent het bewust kunnen schakelen tussen hiërarchische sturing en het faciliteren van het value stream network, zodat stabiliteit en snelle waardecreatie gelijktijdig mogelijk worden.

Voor leiders betekent dit dat zij moeten leren **bewust te schakelen tussen beide systemen**. Soms vraagt de situatie om duidelijk hiërarchisch leiderschap en besluitvorming. Op andere momenten vraagt het waardestroomnetwerk om dienend leiderschap: faciliteren, obstakels wegnemen en teams helpen succesvol te zijn. De essentie van Lean-Agile Leadership ligt daarom niet alleen in **wat leiders doen**, maar vooral in **wanneer zij welk systeem toepassen**.

Twee besturingssystemen: hiërarchie versus Value Stream Network

Het **hiërarchische systeem (OS #1)** is gericht op stabiliteit, voorspelbaarheid en verantwoordelijkheid binnen traditionele lijnen. Dit systeem is essentieel voor compliance, budgetbeheer en formele besluitvorming. Mensen en middelen, facilitatie van OS #2.

Het **waardestroomnetwerk (OS #2)** daarentegen draait om snelheid, wendbaarheid en klantwaarde. Hier werken multidisciplinaire teams over afdelingsgrenzen heen samen om snel en effectief waarde te leveren. In dit netwerk is leiderschap vooral gericht op het faciliteren van samenwerking, het wegnemen van obstakels en het stimuleren van continue verbetering.

“Why Good Leaders must first become good servants”

Wat versta ik onder Lean-Agile Leadership?

Lean-Agile Leadership gaat over het *faciliteren (*dus niet *managen)* van een omgeving waarin *de professionals in de agile teams hun volledige potentieel* kunnen benutten met als eindresultaat *maximale waardeleveringen* en maximale *Flow of Value*. Dit betekent dat leiders hun traditionele, macht-georiënteerde, hiërarchische managementstijl loslaten en overstappen naar een *generatieve stijl* waarin eigenaarschap, vertrouwen, autonomie en betrokkenheid voorop staan. Binnen het hiërarchische systeem, OS #1, vraagt dit om duidelijke richting en besluitvaardigheid. Binnen het waardestroomnetwerk, OS #2, betekent het vooral dienend leiderschap: obstakels wegnemen, context bieden en teams helpen succesvol te zijn.

Wie zijn de Lean-Agile Leaders in SAFe?

Lean-Agile Leaders kunnen uit verschillende lagen van de OS-en komen, waaronder:

1. Change Agents zoals LACE (Lean-Agile Center of Excellence)

• Ondersteunen de organisatie bij de transformatie naar Lean-Agile werkmethoden.

• Begeleiden, coachen en monitoren de voortgang van de implementatie.

2. Executives en Business Owners

• Senior leiders die verantwoordelijk zijn voor de strategische richting en het financieren van de Value Streams en Agile Release Trains (ARTs).

• Zorgen voor alignment tussen bedrijfsstrategie en uitvoering.

3. Line Managers en People Managers

• Ondersteunen Agile Teams in hun ontwikkeling en prestaties.

• Helpen belemmeringen weg te nemen en stimuleren continue verbetering.

4. Release Train Engineers (RTE's)

• Servant leaders van de ART's die teams en ART-niveau events faciliteren.

• Bevorderen samenwerking tussen teams en stimuleren Lean-Agile praktijken.

5. Solution Train Engineers (STE's) *(indien van toepassing in grotere implementaties)*

• Ondersteunen meerdere ARTs binnen een Solution Train.

6. Product Managers en Product Owners

• Leiden de visie en prioriteiten voor producten en oplossingen.

• Zorgen ervoor dat teams waarde leveren die overeenkomt met de bedrijfsdoelen.

7. Lean Portfolio Management (LPM)

• Bestaat uit executives en andere leiders die investeringsbeslissingen en portfolio management uitvoeren.

Verwacht gedrag van leiders

Van leiders verwacht SAFe het volgende gedrag, afgestemd op beide systemen:

• **Luisteren en ondersteunen:** In het waardestroomnetwerk actief luisteren naar teamleden om hen te helpen problemen op te lossen. In de hiërarchie duidelijke kaders bieden.

• **Empathie tonen:** Begrip tonen voor de verschillende perspectieven vanuit de lijnorganisatie en de waardestromen.

• **Authenticiteit** - Vereist dat leiders het gewenste professionele en ethische gedrag modelleren. Walk the talk

• **Respect for People:** betekent dat leiders luisteren, vertrouwen opbouwen en medewerkers de ruimte geven om hun potentieel volledig te benutten in een cultuur van samenwerking en continue verbetering."

• **Moed** (**courage**)- Is essentieel voor leiders om hun organisaties te begeleiden door de snel veranderende dynamiek van het digitale tijdperk.

• **Persoonlijke ontwikkeling stimuleren (grow others):** Mensen helpen hun rol in beide systemen beter te begrijpen en zich daarin te ontwikkelen.

• **Overtuigen in plaats van bevelen:** in het waardestroomnetwerk inspireren en motiveren; in de hiërarchie helder communiceren over beslissingen en verantwoordelijkheden.

• **Vooruitdenken:** Anticiperen op hoe veranderingen in het ene systeem het andere beïnvloeden.

• **Hulp bieden zonder over te nemen (asking powerful questions):** Teams in het waardestroomnetwerk ondersteunen zonder hun eigenaarschap weg te nemen.

• **Openheid bevorderen:** Transparantie stimuleren, vooral bij het schakelen tussen beide systemen.

• **Unlock the Intrinsic Motivation of Knowledge Workers:** Agile leiders creëren een omgeving waarin professionals intrinsiek gemotiveerd worden door autonomie, doelgerichtheid en meesterschap te stimuleren.

• **Gedecentraliseerde besluitvorming** - Verplaatst de beslissingsbevoegdheid naar de plek waar de informatie zich bevindt.

The problem is not with our organizations realizing that they need to transform; the problem is that organizations are using managerial frameworks and infrastructure models from past revolutions to manage their businesses in this one.": Mik Kersten

Verwachte cultuur

De culturen binnen beide systemen moeten elkaar versterken in plaats van tegenwerken. Lean-Agile leaders moeten een omgeving scheppen met:

• **Vertrouwen en veiligheid (psychological safety):** Zowel in de hiërarchie als in het waardestroomnetwerk moeten mensen fouten kunnen maken en daarvan leren.

• **Samenwerking en transparantie:** het waardestroomnetwerk functioneert alleen goed als de hiërarchie deze samenwerking ondersteunt en niet belemmert.

• **Continue verbetering en leren:** Feedback integreren in beide systemen om als geheel wendbaar te blijven.

• **Eigenaarschap en autonomie:** Teams in het waardestroomnetwerk krijgen ruimte, terwijl leidinggevenden in de hiërarchie hen steunen zonder micromanagement.

• **Waarde-gedrevenheid:** Beide systemen richten zich op het leveren van maximale klantwaarde.

Verwachte acties van leiders

Om deze cultuur en het gewenste gedrag te bevorderen, verwacht SAFe van leiders de volgende acties:

• **Duidelijkheid scheppen:** maak helder wanneer je spreekt vanuit de hiërarchie en wanneer vanuit het waardestroomnetwerk.

• **Coachen met krachtige vragen:** Stimuleer kritisch denken, vooral in het waardestroomnetwerk.

• **Conflicten effectief oplossen:** Begrijp spanningen tussen beide systemen en help deze op te lossen.

• **Waarde leveren boven bureaucratie:** Vermijd onnodige processen in het waardestroomnetwerk zonder de noodzakelijke compliance vanuit de hiërarchie te negeren.

• **De lange termijn bewaken:** houd rekening met de stabiliteit van de hiërarchie én de wendbaarheid van het waardestroomnetwerk.

• **Obstakels proactief verwijderen:** Signaleer systemische problemen die samenwerking tussen de twee systemen belemmeren.

• **Voorbeeldgedrag vertonen:** Laat zien hoe je soepel kunt schakelen tussen beide besturingssystemen.

Door Lean-Agile Leadership actief te tonen en het juiste systeem op het juiste moment in te zetten, creëren we een organisatie waarin stabiliteit en wendbaarheid hand in hand gaan, teams excelleren en klantwaarde structureel wordt verhoogd.

34. Lijnmanager én gelijkwaardige collega? Waarom SAFe duidelijke rolgrenzen vraagt

Lijnmanager én gelijkwaardige collega? Waarom SAFe duidelijke rolgrenzen vraagt

In veel organisaties die transformeren naar SAFe® zie ik hetzelfde spanningsveld terug:
de wens om bestaande lijnmanagementstructuren (**OS #1**) te combineren met de nieuwe, wendbare netwerkstructuur (**OS #2**).

De redenering is vaak: *“Degene die straks beoordeelt, zit dan dicht bij degene die beoordeeld wordt.”* Dat klinkt aantrekkelijk, maar in de praktijk blijkt het zelden succesvol en vaak conflicterend met fundamentele SAFe-principes zoals **vertrouwen** (zie de SAFe Core Values) en **psychologische veiligheid** (zie Lean-Agile Leadership).

Belang van duidelijkheid in rollen

Scheiding van rollen
Het is cruciaal om de verantwoordelijkheden in OS #1 en OS #2 helder te scheiden.
Lijnmanagers behouden hun **hiërarchische verantwoordelijkheden in OS #1**, terwijl ze in OS #2 alleen kunnen deelnemen als **gelijkwaardige collega’s**.

Deze scheiding creëert een cultuur van **vertrouwen en psychologische veiligheid**, die onmisbaar is voor een succesvolle Agile-transformatie.

Psychologische veiligheid ontstaat wanneer leiders een omgeving scheppen waarin het nemen van risico’s wordt aangemoedigd, zonder angst voor negatieve gevolgen voor iemands zelfbeeld, status of loopbaan.

Vertrouwen en psychologische veiligheid

Openheid en innovatie
Wanneer teamleden hun leidinggevende in het netwerk als “collega” treffen, terwijl die hen in OS #1 ook beoordeelt, ontstaat er vaak terughoudendheid. Mensen voelen zich minder vrij om ideeën of zorgen te delen, uit angst dat dit gevolgen heeft voor hun beoordeling.

Dat ondermijnt precies de openheid, samenwerking en innovatie die SAFe wil stimuleren.
Leiders die de scheiding tussen OS #1 en OS #2 respecteren, creëren juist het vertrouwen waarin teams kunnen experimenteren, leren en groeien.

Actieve bijdrage aan de gezamenlijke backlog

Eenduidige pet
Iedereen die toetreedt tot het netwerk, bijvoorbeeld de LACE (Lean-Agile Center of Excellence), moet actief bijdragen aan de gezamenlijke backlog vanuit één duidelijke, in het **Target Operating Model (TOM)** gedefinieerde rol.

Zonder concrete user stories of verantwoordelijkheden is er geen rol. Zo voorkomen we dubbelrollen, verwarring en hiërarchische terugval naar oude patronen.

Nieuwe governancespelregels

Duidelijke governance
Een goed ingericht dual operating system (conform John Kotter) onderscheidt twee werelden met elk hun eigen spelregels:

• in **OS #1** liggen de hiërarchische verantwoordelijkheden;

• in **OS #2** werken we als gelijken, zonder hiërarchie, om openheid, respect en vertrouwen te bevorderen.

De **governance van het TOM** bepaalt wie waar acteert, met welke bevoegdheden, en onder welke condities deelname aan OS #2 plaatsvindt.

Duurzame transformatie

Cultuur van wendbaarheid
De kern van elke duurzame transformatie is helderheid:

“Iedereen mag meedoen, zolang dat gebeurt volgens de nieuwe, in het TOM vastgelegde governancespelregels.”

Zo bouwen we stap voor stap aan een organisatie die SAFe niet alleen **doet**, maar **is**: wendbaar, transparant en gebaseerd op vertrouwen.

Een en ander betekent dat lijnmanagers die in OS #2 een rol met eigen verantwoordelijkheid willen vervullen, een keuze moeten maken: óf HRM-activiteiten in OS #1, óf value stream-activiteiten in OS #2.

35. SAFe-overhead: perception or reality? De onmisbare schakel naar efficiëntie

Veel organisaties die het Scaled Agile Framework (SAFe) overwegen, krijgen vroeg of laat te maken met een veelvoorkomende perceptie: “SAFe creëert wel heel veel overhead!” De toevoeging van rollen zoals Scrum Masters (SM’s), Product Owners (PO’s) en Release Train Engineers (RTE’s) roept bij sommige stakeholders en teamleden het gevoel op dat de organisatie zich vult met extra rollen en processen die tijd en geld kosten, zonder directe toegevoegde waarde. Deze perceptie leeft vaker en daarom is het belangrijk om te erkennen dat dit gevoel bij teamleden en stakeholders kan ontstaan tijdens een transformatie.

Onderzoeken en praktijkvoorbeelden van toonaangevende instituten tonen aan dat, hoewel SAFe aanvankelijk lijkt te zorgen voor meer structuren en rollen, deze juist zijn ontworpen om verborgen inefficiënties in organisaties te verminderen.

Het is dus de vraag: *is de perceptie van overhead werkelijkheid of een tijdelijke indruk die verandert zodra de voordelen van SAFe zichtbaar worden?*

Dat wil ik in dit hoofdstuk nader onderzoeken en het woord ‘overhead’ gaan koppelen aan de woorden ‘Value Maximalisatie’, ‘Efficiency’ en ‘organisatiedoelen, Target Metrics’.

De werkelijke waarde van rollen in SAFe: verminderen van verborgen overhead

1. Rollen als sleutel tot productiviteitswinst

Een onderzoek van *McKinsey & Company* wijst uit dat het opschalen van Agile methoden zoals SAFe leidt tot productiviteitsverbeteringen van 20 tot 30% in organisaties. Rollen zoals SM’s, PO’s en RTE’s dragen hieraan bij door de communicatie te stroomlijnen, afstemmingsproblemen op te lossen en verspilling tegen te gaan. Deze rollen vervangen de verborgen kosten die vaak niet zichtbaar zijn, zoals herwerk door miscommunicatie en tijdverlies door onduidelijke verantwoordelijkheden (McKinsey & Company, *How to Select the Right Agile Framework*).

2. Minder coördinatiekosten en focus op waardecreatie

Onderzoek van *Scaled Agile Inc.* laat zien dat organisaties die SAFe adopteren gemiddeld een reductie van 30% in coördinatiekosten ervaren. De rollen zoals RTE’s en PO’s fungeren als verbindende schakels die afhankelijkheden beheren en samenwerking stimuleren, wat voorkomt dat teams constant tijd verliezen aan het oplossen van escalaties en misverstanden (Scaled Agile Inc., *2018 SAFe Case Study*).

3. Lockheed Martin: minder frictie en snellere levering

Lockheed Martin’s case is een goed voorbeeld van hoe SAFe de perceptie van overhead kan transformeren. Door SAFe in te voeren, slaagde het bedrijf erin om projectdoorlooptijden met 40% te verkorten en verspilling drastisch te verminderen. De toevoeging van gestructureerde rollen zorgde ervoor dat teams minder tijd kwijt waren aan interne coördinatie, wat uiteindelijk leidde tot betere focus en sneller resultaat. Dit toont aan dat rollen die in eerste instantie als ‘overhead’ worden gezien, juist helpen om efficiëntie te verhogen zodra het framework goed ingebed is (Scaled Agile Inc., *Lockheed Martin Case Study*).

4. Empirische data over Agile transformaties

Volgens *Forrester Consulting* leiden geschaalde Agile frameworks zoals SAFe tot 50% meer teamafstemming en 30% minder werk-in-uitvoering knelpunten. De rollen van SM’s en RTE’s spelen hier een sleutelrol door werkstromen actief te beheren en efficiëntie te maximaliseren, wat traditionele overhead door ingewikkelde overdrachtsprocessen vermindert. Dit benadrukt dat wat eerst aanvoelt als ‘extra’ rollen, in werkelijkheid noodzakelijk is voor het creëren van een soepele organisatie (Forrester, *The Total Economic Impact of Scaled Agile Framework (SAFe)*).

5. Lean-Agile praktijken: efficiëntie door duidelijke verantwoordelijkheden

Het *Lean Enterprise Institute* onderstreept dat de rollen in SAFe bijdragen aan een Lean benadering van projectmanagement, waarbij verspilling in processen wordt tegengegaan. Het doel van deze rollen is niet om de organisatie te belasten, maar om een cultuur te creëren waarin teams doelgericht kunnen werken, met minimale afhankelijkheden en maximale waardecreatie (Lean Enterprise Institute, *Lean Enterprise Case Studies*).

Conclusie: SAFe's rollen als strategische investering in efficiëntie

De perceptie dat SAFe extra overhead introduceert, is begrijpelijk, vooral in de vroege fases van een transformatie. Het kost tijd voordat teamleden en stakeholders de voordelen van deze nieuwe structuren ervaren. Echter, zodra de rollen goed zijn ingebed en hun waarde zichtbaar wordt, verandert de perceptie vaak in een begrip van hoe deze rollen bijdragen aan minder verspilling, verbeterde samenwerking en kortere doorlooptijden. Onderzoek en praktijkvoorbeelden tonen aan dat deze rollen een cruciale investering zijn in de wendbaarheid en kwaliteit van de organisatie.

Voor organisaties die streven naar een wendbare, productieve en kwalitatief hoogwaardige werkomgeving, biedt SAFe dus niet zomaar 'extra rollen', maar juist de basis voor een efficiëntere en meer waardevolle samenwerking.

Deel 5: Teams en teamorganisatie

36. Dit zijn de essentiële kenmerken van een SAFe Agile Team

Examenvraag: wat is het goede antwoord?

Question:
What is the foundation of SAFe?

A) Lean Portfolio Management
B) Solution Train
C) Agile Release Train
D) Agile Teams

Inleiding

In dit hoofdstuk wil ik dieper ingaan op wat het fundament is van SAFe (Scaled Agile Framework).

Hoewel de juiste keuze voor de vraag "What is the foundation of SAFe?" in het examen Lean Portfolio Management (A) *lijkt te zijn*, draait SAFe in de praktijk vooral om de Agile Teams en is (D) het juiste antwoord. Agile Teams vormen de kern van het operating system en zijn cruciaal voor het succesvol toepassen van SAFe.

Het belang van goed gemodelleerde Agile Teams binnen SAFe kan niet genoeg worden benadrukt. Zonder goed functionerende Agile Teams kunnen er geen *scaled* oplossingen worden gecreëerd, kan er geen waarde geleverd worden aan de klant en realiseren wij onze *Flow Metrics* straks niet.

Hieronder geef ik je inzicht in hoe ik zelf in mijn eigen praktijk nieuwe Agile Teams ontwerp, wat de ontwerpcriteria zijn, de randvoorwaarden en hoe ze bijdragen aan het grotere geheel.

Fundamentele kenmerken van de SAFe Agile Teams

Modelleerkenmerken

1. **(altijd) onderdeel van een Agile Release Train (ART):** Agile Teams opereren in SAFe binnen een ART, die de kaders biedt voor samenwerking en coördinatie. Individuele Agile Teams bestaan niet buiten de context van een ART, behalve in uitzonderlijke gevallen zoals bij gespecialiseerde teams (bijv. het LACE) of Enabler Teams. Dit is dus een fundamenteel verschil met Scrum Agile Teams, die geen verplichte synchronisatiestructuur kennen met andere teams: Scrum beschrijft één team, zonder voorgeschreven coördinatie op schaal.

2. **De wet van Dunbar:** De wet van Dunbar stelt dat er een cognitieve limiet is aan het aantal stabiele sociale relaties dat een individu kan onderhouden: circa 150, maar veel minder voor intieme nauwe relaties. Een Agile Release Train (ART) bestaat doorgaans uit 50 tot 125 mensen. Scrum en SAFe stelden vroeger de max size op 11, inmiddels zeggen beide dat de ideale Agile Team size, *5 tot 10 leden* is, inclusief de SM en de PO (SAFe: *'typically 10 or fewer people*'). Deze maximale omvang maakt communicatie en samenwerking makkelijker en stelt het team in staat om snel beslissingen te nemen. Hoe groter je team, hoe langer je Team-Sync event (max 15 min) en hoe meer de wet van de afnemende meeropbrengst van toepassing wordt.

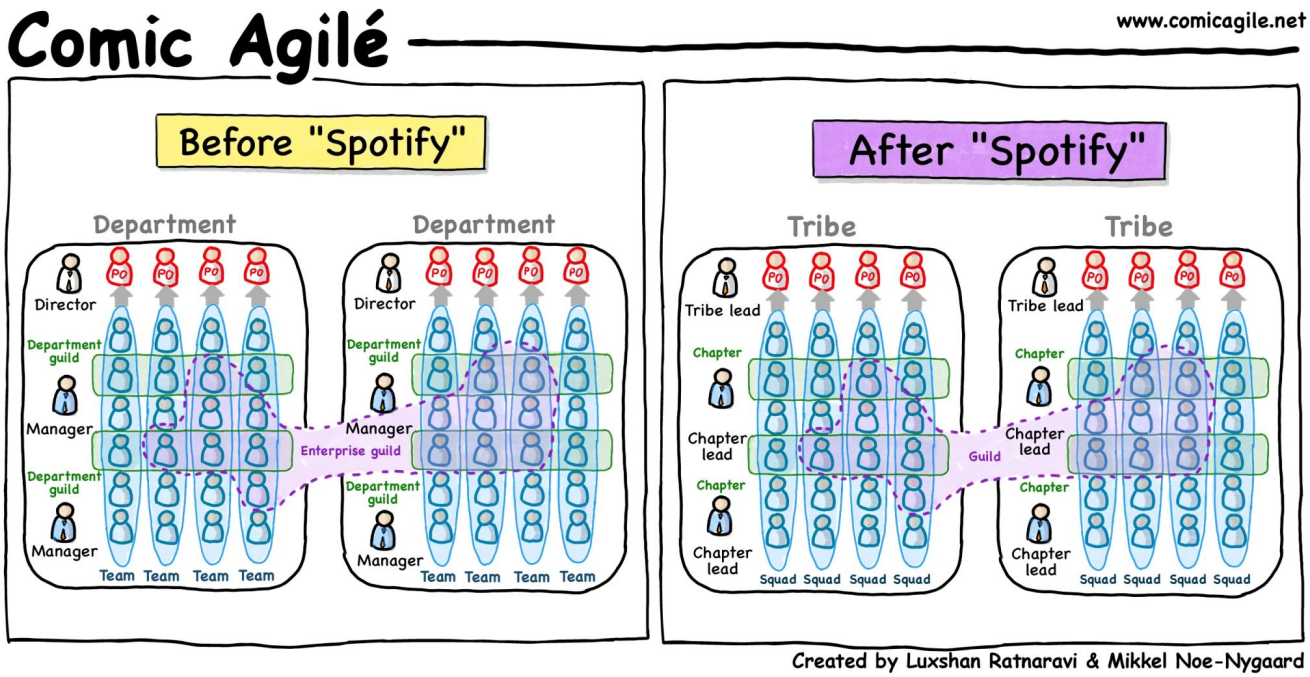
3. **Drie rollen:** In een Agile team zijn er drie essentiële rollen die gezamenlijk zorgen voor het leveren van waarde: (1) de Scrum Master (SM), (2) de Product Owner (PO) en (3) de ontwikkelteamleden (Dev-Team members). De Scrum Master faciliteert het team, verwijdert belemmeringen en zorgt dat het team zich kan focussen op zijn doelen. De Product Owner vertegenwoordigt de klant, stelt prioriteiten binnen de backlog en bepaalt welke User Stories de meeste waarde opleveren. De ontwikkelteamleden zijn verantwoordelijk voor het daadwerkelijk bouwen, testen en opleveren van 'het increment'. Samen vormen deze rollen de kern van een effectief en goed functionerend Agile team.

4. **Cross-functional (tenzij):** Dit betekent dat alle disciplines die nodig zijn om het *Potentially Shippable Product* te bouwen (UX/UI, Architect, Business Analist, developers, testers, maintenance, privacy, security, marketing, etc) in het agile team vertegenwoordigd zijn. Echter, tenzij….want, hoewel SAFe cross-functionele teams voorschrijft, kan Team Topologies helpen bij het analyseren van welke soorten teams er nodig zijn en hoe zij met elkaar moeten samenwerken. Zo is het bijvoorbeeld niet efficiënt om 1 AI-expert 1 dag per week in 1 agile team te laten werken. Deze 'functie' modelleer je bij voorkeur in een 'Complicated Subsystem Team' dat vervolgens 'AI componenten' aan alle andere Agile Teams levert. AI wordt dan niet cross-functional gemodelleerd.

5. **Self-organizing:** SAFe Agile Teams zijn zelforganiserend, wat betekent dat ze de verantwoordelijkheid dragen voor hoe ze hun werk aanpakken en uitvoeren. Ze bepalen zelf welk werk ze van de team backlog pullen, hoe ze het werk organiseren, hoe ze het product gaan bouwen, en onderling het werk verdelen om de PI doelstellingen te bereiken en het product te onderhouden. Maar, self-organizing binnen kaders (zie verderop)

6. **Long-lived:** Agile teams kennen geen start-stop, maar zijn Long-Lived en dus *uitermate stabiel.* In tegenstelling tot projecten, waar teamleden elke keer weer door de *forming-storming-norming-performing* fases heen moeten, en daardoor vrijwel nooit de performing-fase, en dus de status van High Performance Team (HPT), bereiken. De enige rimpeling bij Agile Teams kan een nieuw teamlid zijn, waardoor het agile team zich opnieuw gaat vormen.

7. **Minimaal aantal afhankelijkheden:** Agile Teams in SAFe moeten zo zelfstandig mogelijk werken met een minimaal aantal afhankelijkheden van andere Agile Teams binnen - en buiten hun eigen ART. Dit voor het bereiken van Maximale *Flow of Value*. Een van de meest belangrijke ontwerp-criteria, welke ik een paar keer fout heb zien gaan door gebrek aan modelleerkennis en ervaring. Dan krijg je een situatie als hieronder, waarin alle silo's nog 1 op 1 bestaan.



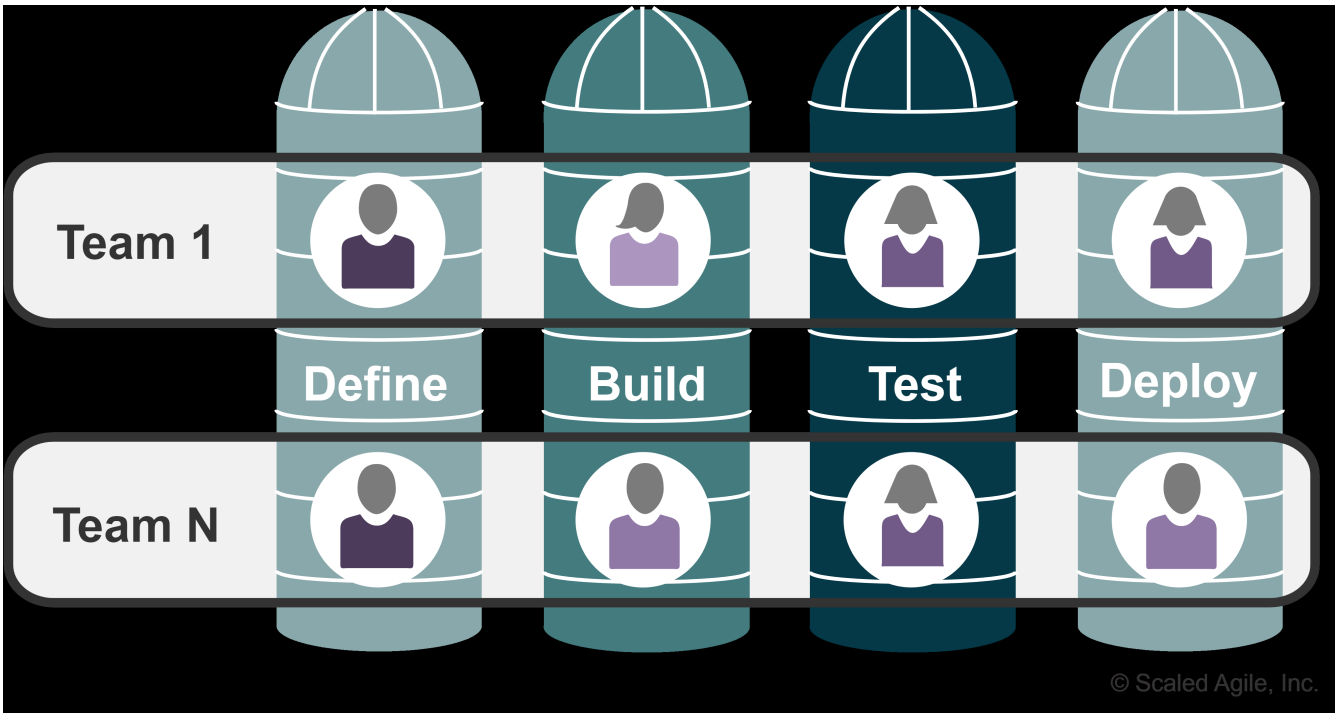
Werkkenmerken

1. **Werken in iteraties:** Agile Teams werken in korte iteraties van 1 tot max 4 weken, waarin ze waardevolle *Increments* leveren, weer een stukje van de grote puzzel. Deze werkwijze zorgt ervoor dat teams snel kunnen inspelen op feedback en veranderende omstandigheden. Je kunt elke iteratie gerust zien als een mini-projectje. Hoe korter de iteratie, hoe minder het risico voor Executive Management en de BO's.

2. **Synchronisatie & cadence:** In tegenstelling tot Scrum, maken SAFe Agile Teams deel uit van een Agile Release Train en leveren zij in onderlinge samenhang met alle andere teams (scaled) waarde binnen de context van een gezamenlijke oplossingsmissie. Dit betekent dat Agile teams regelmatig moeten synchroniseren met

andere teams. Dit kan alleen goed als alle Agile Teams *exact dezelfde* Cadence volgen.

3. **DBTD-DBTD:** Agile teams volgen elke iteratie het DBTD-proces wat staat voor Design, Build, Test, Deploy. Dit betekent dat in elke iteratie alle fasen van productontwikkeling worden doorlopen. Teams die DDDBBBTTTD doen, zijn (stiekem) aan het waterfallen binnen SAFe. Mag niet, maar ik weet dat het soms ook niet anders kan, zelfs bij doorgewinterde SAFe bedrijven.



1. **Potentially Shippable Product:** Aan het einde van elke iteratie moeten alle Agile Teams een *Potentially Shippable Product* opleveren. Iets wat klaar is, af is en kan worden ingezet bij de *klant/gebruiker* van het Agile team. Let op, deze gebruiker kan zowel een ander agile team zijn, een andere ART, de lijn, de business of zelfs een externe partij . Het product kan een proces binnen een applicatie zijn, een deel van een groter geheel, een nieuwe service, een hoofdstuk van een PV, of een volledig werkende oplossing. Zolang dat 'iets maar werkt' en voldoet aan de DoD en direct kan worden gebruikt.

2. **DevSecOps compliant:** SAFe Agile Teams integreren DevSecOps-principes en werken met een continue leveringsmentaliteit. Security en Privacy zijn vanaf het begin ingebed in het ontwikkelproces, waardoor producten veilig en compliant zijn. Agile Teams zijn bij voorkeur van A tot Z verantwoordelijk voor het product, dus Dev en Ops. SAFe vermijdt 'Silo's' zoals jullie weten, met het gevaar van *wij-zij* mindsets (Zie ook kenmerk T-Shaped hieronder).

3. **Scrum of Kanban:** SAFe Agile Teams hebben de keuze tussen *SAFe-Scrum* en *SAFe-Team Kanban* als agile werkwijzen, afhankelijk van de specifieke behoeften van het team en de aard van het werk. Dit zorgt voor flexibiliteit binnen het raamwerk.

Team membership kenmerken

1. **High Performance Teams (HPT)**: Een belangrijk aspect bij het vormen van Agile Teams is dat teamleden goed, *uitstekend*, met elkaar moeten kunnen samenwerken. En dit geldt ook voor het LACE (-: Door bij het samenstellen van Agile Teams te letten op deze sociale en interpersoonlijke aspecten, kun je teams creëren die beter presteren en effectiever samenwerken en zich door kunnen ontwikkelen tot High Performance Teams. Ook al betekent dit verwijdering van niet sociaal passende teamleden. Het bereiken van de HPT-status kan veel gemakkelijker, omdat zoals ik eerder al vertelde, agile teams geen *start-stop* kennen en dus veel langer met elkaar samenwerken met veel meer rust. Een van de belangrijke taken van alle Agile coaches is te zorgen dat Agile Teams zich doorontwikkelen tot HPT's. (*voor meer info zie het hoofdstuk 'High Performance Teams: waar vertrouwen, focus en discipline samenkomen')

2. **T-Shaped:** Teamleden binnen een SAFe Agile Team hebben een diepgaande expertise in een specifiek vakgebied, maar zijn ook in staat om andere taken op te pakken buiten hun specialisatie zoals ‘maintenance’ of ‘incident management'. Deze flexibiliteit zorgt voor een betere samenwerking en verhoogt de effectiviteit van het team.

3. **Autonomie binnen kaders:** Dit kenmerk sluit aan bij ‘self-organizing’**.** Hoewel Agile Teams autonoom werken, opereren ze binnen de bredere doelstellingen van de ART en de organisatie. Dit waarborgt dat hun werk is afgestemd op de strategische doelen. Het LACE bekijkt de kaders van de autonomie zoals bijvoorbeeld: mogen Agile Teams hun eigen mensen werven, intakes doen en wegsturen? Mogen ze zelf kiezen waar het geld naartoe gaat? Mogen ze hun eigen tooling kiezen?

4. **Kaizen-mentaliteit (Relentless Improvements):** Continu verbeteren staat centraal binnen SAFe Agile Teams. Alle leden reflecteren *regelmatig* op hun processen en proberen deze voortdurend te optimaliseren. Agile Teams houden regelmatig Retro’s, reflecteren en verbeteren elke iteratie aantoonbaar, en wel aan de hand van *concrete (flow) Metrics.* Of verbeteren *Ad-hoc,* ingeval er sprake is van urgentie.

5. **SAFe principles:** Agile Team members zijn getraind in, en beheersen, het Agile Manifesto zoals bijvoorbeeld Principle #6 *‘The most efficient method to convey information is…’,* de SAFe Values, The House of Lean, de 10 SAFe principles en alle SAFe’s Best Practices.

Conclusie

De basis van SAFe ligt niet alleen in strategische concepten zoals Lean Portfolio Management, maar vooral in de Agile Teams die de daadwerkelijke waarde creëren. Het succes van een SAFe-implementatie hangt af van goed georganiseerde, autonome, gesynchroniseerde en cross-functionele DevSecOps teams die continu verbeteren, zelforganiserend zijn, en samenwerken volgens Agile principes.

Een van de *grote voordelen* (en waardoor SAFe voor een hogere productiviteit zorgt) is het feit dat Agile Teams Long-Lived zijn, en daardoor de HPT-status kunnen bereiken en niemand meer resources hoeft aan te vragen of te zoeken. Het hele Workforce & Resource Management wordt met SAFe veel gemakkelijker en veel simpeler.

Agile teams zijn dus de kern van elke waardecreatie in SAFe, en hun manier van werken, de kwaliteit van de inrichting, de kwaliteit van het ontwerp, bepaalt straks mede het succes voor de gehele organisatie. Dit is een taak voor het LACE om zo goed als mogelijk te modelleren.

Samenvatting: Agile Teams vormen het enige en echte fundament van SAFe, antwoord D !

Bijlage: soorten Agile Teams in SAFe

In SAFe zijn verschillende soorten Agile teams actief, waarmee SAFe modelleerders kunnen ‘spelen’ bij het ontwerp van een TOM. Deze teams kunnen worden ingedeeld in drie hoofdgroepen: gespecialiseerde teams, business teams, en technologie teams. Elk type team speelt een belangrijke rol in het realiseren van bedrijfswaarde en het faciliteren van de Agile transformatie binnen de organisatie.

1. Gespecialiseerde Agile Teams

• **LACE (Lean-Agile Center of Excellence)**

• Dit team is verantwoordelijk voor het stimuleren van de Agile transformatie binnen de organisatie.

• Ze faciliteren training, coaching en ondersteuning, en zorgen ervoor dat de SAFe-implementatie consistent en effectief verloopt.

• Meestal opgehangen op LPM niveau of ergens ‘centraal’

• **System Team**

• Dit team ondersteunt de Agile Release Train (ART) door het integreren van werk, het uitvoeren van systeemtests, en het ondersteunen van de deployment van de oplossing.

• Ze spelen een cruciale rol bij continue integratie (CI) en continue levering (CD).

• Ophangen op ART niveau (=1 vd 5 ART rollen).

• (Pas op: heeft overlap met het Enabler team)

• **Shared Services (SS)**

• Shared Services teams zijn niet *altijd fulltime* betrokken bij een specifiek team, maar worden ingeschakeld op basis van de behoeften van verschillende teams. Dit team biedt *specialistische* expertise en ondersteuning aan meerdere Agile teams of ART's.

• Is onderdeel van SAFe's 'Spanning Palette', wat betekent dat dit team over alle lagen van SAFe kan worden ingezet (scaled)

• Ze kunnen experts omvatten in domeinen zoals beveiliging, Privacy, UX, compliance, etc

• **Architectuur team / Design Authority Board (Geen SAFe!)**

• SAFe voorziet niet in het construct 'Architectuur team, Architectuur board', etc.

• Je kunt dit inrichten als Specialized Agile team, bijvoorbeeld op LPM level en het LACE Construct kopiëren.

2. Business Teams

• **Business Agile Teams**

• Deze teams zijn gericht op het toepassen van Agile principes binnen niet-technische domeinen zoals marketing, HR, of financiën.

• Ze helpen bij het sneller leveren van bedrijfswaarde door Agile werkwijzen zoals iteratief werken, korte feedbackloops en zelforganisatie toe te passen.

3. Technologie Teams

Binnen technologie teams maken we onderscheid tussen vier specifieke typen die bijdragen aan de waardecreatie binnen SAFe:

• **Stream-Aligned Teams**

• Deze teams zijn direct verantwoordelijk voor het leveren van waarde *aan de klant* door *end-to-end* te werken binnen een bepaalde waardestroom.

• Je kunt een integratie Stream-Aligned Team maken.

• **Complicated Subsystem Teams**

• Deze teams werken aan onderdelen of subsystemen die speciale kennis en expertise vereisen, bijvoorbeeld complexe algoritmes, AI, Big Data of gespecialiseerde architectuurcomponenten. Je wilt deze toch (tegen Silo denken in !) in 1 hok hebben en niet 'scattered' over alle Agile Teams.

• Hun werk ondersteunt andere teams binnen de waardestroom, en is dus vaak niet zichtbaar voor de eindgebruiker.

• De klanten van deze teams zijn dus andere teams

• **Platform Teams**

• Platform teams bieden ondersteunende diensten en technologieën *aan andere teams*, zoals infrastructuur (AWS, SAP, Azure, SAN's), tools, of shared services.

• Ze stellen de stream-aligned teams in staat om sneller te werken door herbruikbare oplossingen en technische ondersteuning te leveren.

• **Enabler Teams**

• Enabler teams zijn gericht op het ondersteunen van de technologische en infrastructurele aspecten van de oplossing, bijvoorbeeld door te werken aan technische schulden, research spikes, of het voorbereiden van de infrastructuur.

• De klanten van deze teams zijn dus andere teams.

37. Wanneer is een Agile Team écht een topteam?

In veel organisaties worden mensen simpelweg bij elkaar gezet en vervolgens een *team* genoemd. Maar een groep mensen in dezelfde ruimte, of op hetzelfde Jira-board, maakt nog geen team.

Misschien heb je het zelf wel eens meegemaakt: je zit in een meeting waarin een teamlid vertelt waar hij of zij mee bezig is, maar voor jouw werk heeft dat eigenlijk geen enkele relevantie. Op dat moment voelt zo'n meeting al snel als een *waste of time*.

Dat is vaak een signaal dat er geen echte onderlinge afhankelijkheid bestaat. En zonder onderlinge afhankelijkheid is er feitelijk geen team, maar slechts een groep individuen die naast elkaar werken.

Zeker in een Scaled Agile (SAFe) omgeving is het belangrijk om je af te vragen of je eigenlijk wel een *echt team* bent.

(bron: https://framework.scaledagile.com/scrum-master-team-coach)

Binnen SAFe zijn Agile Teams **long-lived teams**. Ze blijven langdurig bij elkaar, waardoor er voldoende tijd ontstaat om vertrouwen op te bouwen, samenwerking te verbeteren en uiteindelijk een echt high-performance team te worden. De **Scrum Master/Team Coach** speelt hierin een belangrijke rol door het team te begeleiden in deze ontwikkeling.

Maar de kernvraag blijft: **wanneer is een team eigenlijk een team?**

Een team ontstaat pas wanneer mensen elkaar nodig hebben

Een echte teamdynamiek ontstaat pas wanneer teamleden **van elkaar afhankelijk zijn om het werk af te krijgen**. Als iedereen zijn eigen taak uitvoert zonder anderen nodig te hebben, is er feitelijk geen team, maar slechts een verzameling individuen.

Kenmerken van High-Performance Teams (HPT's)

Wanneer een Agile Team zich ontwikkelt tot een High-Performance Team, zie je een aantal duidelijke kenmerken terug. Deze kenmerken vormen feitelijk het antwoord op twee belangrijke vragen: *zijn jullie een echt team?* en *functioneren jullie al als een High-Performance Team?*

Aan de hand van deze eigenschappen kun je vaak vrij snel herkennen of een groep mensen daadwerkelijk als team samenwerkt, of dat het in de praktijk nog vooral een verzameling individuen is die toevallig onder dezelfde teamnaam opereren.

#1: Een gedeelde visie en duidelijke *shared* doelen
Iedereen begrijpt waarom het team bestaat en waar het team naartoe werkt, aan welke gezamenlijke doelen we met z'n allen werken (The Solution). Die gezamenlijke richting zorgt voor focus en betrokkenheid, maar geeft ook energie.

#2: Focus op succes en waarde
In plaats van energie te steken in het vermijden van fouten, richten high-performance teams zich op het leveren van waarde en het behalen van resultaten.

#3: Commitment en betrouwbaarheid
Teamleden voelen zich gezamenlijk verantwoordelijk voor het resultaat. Afspraken worden nagekomen en het team levert consistent kwalitatief werk.

#4: Cross-functional: Diversiteit in kennis en vaardigheden
High-performance teams beschikken over verschillende perspectieven, kennis en vaardigheden. Juist die diversiteit maakt het mogelijk om complexe problemen effectief op te lossen. In SAFe noemen we dat een

multidisciplinair of cross-functional team.

#5: Psychologische veiligheid en vertrouwen
Teamleden voelen zich veilig om ideeën te delen, vragen te stellen en fouten te bespreken zonder angst voor schaamte of straf. Hierdoor ontstaat open communicatie en ruimte voor gezonde discussies.

#6: Effectieve besluitvorming en zelfhelend vermogen
Het team organiseert zijn werk grotendeels zelf en beschikt over voldoende kennis en mandaat om zelfstandig beslissingen te nemen, zonder steeds afhankelijk te zijn van externe sturing.

Daarnaast heeft het team een **zelfhelend vermogen**: problemen en conflicten worden binnen het team besproken en opgelost, zonder direct te escaleren naar management. Teamleden spreken elkaar aan en verbeteren samen de samenwerking.

#7: Uitdaging in balans met vaardigheden
Het werk is uitdagend genoeg om energie en betrokkenheid te creëren, maar niet zo moeilijk dat het team vastloopt.

#8: Werkplezier
Tot slot halen high-performance teams energie uit het samenwerken. Ze vinden het leuk om samen problemen op te lossen en resultaten te bereiken.

Van kenmerken naar concrete afspraken: het Team Sociaal Contract

Weten wat een High-Performance Team kenmerkt is één ding. Maar hoe zorg je er als team voor dat die kenmerken ook echt gaan leven? Een krachtig hulpmiddel daarvoor is het opstellen van een **Team Sociaal Contract**, ook wel een *Working Agreement* of *Team Norms* genoemd.

Een sociaal contract is een gezamenlijk opgesteld document waarin het team vastlegt *hoe* ze met elkaar willen samenwerken. Niet opgelegd van buitenaf, maar door het team zelf bepaald. Juist dat maakt het zo krachtig: iedereen heeft eraan bijgedragen, dus iedereen voelt zich er ook aan verbonden.

In zo'n contract legt het team zaken vast zoals:

• Welke waarden en gedragsnormen we hanteren

• Hoe we communiceren en besluiten nemen

• Hoe we elkaar aanspreken als afspraken niet worden nagekomen

• Wat we van elkaar verwachten rondom beschikbaarheid, samenwerking en kwaliteit

Een sociaal contract helpt dus direct bij het versterken van psychologische veiligheid (#5), onderlinge afspraken en commitment (#3), en het zelfhelend vermogen van het team (#6). Het is geen eenmalig document, maar een levend akkoord dat het team regelmatig evalueert en bijstelt, bijvoorbeeld tijdens een retrospective.

Voor de Scrum Master/Team Coach is het initiëren van een sociaal contract een van de eerste en meest waardevolle stappen om een groep mensen te laten groeien naar een echt team.

Van groep naar team

Een Agile Team wordt dus niet automatisch een High-Performance Team. Dat is een ontwikkelproces dat tijd, vertrouwen en leiderschap vraagt.

In de literatuur over teamontwikkeling wordt dit proces al lang beschreven. Zo introduceerde **Bruce Tuckman (1965)** het bekende model van *Forming, Storming, Norming en Performing*. Teams beginnen als een losse groep mensen, gaan vervolgens door een fase van discussie en frictie, ontwikkelen daarna gezamenlijke werkafspraken en bereiken uiteindelijk het niveau waarop ze echt effectief samenwerken.

Ook **Patrick Lencioni** benadrukt in zijn model *The Five Dysfunctions of a Team* dat vertrouwen de basis vormt van goed functionerende teams. Zonder vertrouwen ontstaan er geen open discussies, geen echte commitment en spreken teamleden elkaar niet aan op resultaten.

Deze inzichten sluiten sterk aan bij hoe **Agile en SAFe** naar teamontwikkeling kijken. Omdat Agile Teams in SAFe zogenaamde **long-lived teams** zijn, krijgen teams de tijd om deze ontwikkelfases daadwerkelijk door te maken en uiteindelijk uit te groeien tot High-Performance Teams, afgekort HPT's.

De basis is relatief eenvoudig en kun je samenvatten in drie kenmerken:

• Werk samen aan **één duidelijk doel;**

• Zorg dat teamleden **elkaar nodig hebben** om dat doel te bereiken;

• Creëer een omgeving van **vertrouwen, veiligheid en autonomie.**

Wanneer die elementen aanwezig zijn, verandert een groep mensen langzaam maar zeker in iets veel krachtigers: **een echt team, een High-Performance Team.**

38. High Performance Teams: waar vertrouwen, focus en discipline samenkomen

1.1. Waarom High Performance Teams ertoe doen

Iedere (SAFe) organisatie wil betere resultaten, snellere levering, hogere kwaliteit, vermogen om complexe problemen op te lossen en meer eigenaarschap. Daarom investeren organisaties in strategie, jaarplannen, processen, tooling, training en leiderschapsprogramma's. Toch blijft de praktijk vaak weerbarstig. Plannen worden niet volledig gerealiseerd, deadlines & prioriteiten verschuiven, besluitvorming vertraagt en teams leveren minder voorspelbaar dan gehoopt.

De oorzaak ligt zelden in één persoon of één verkeerd proces. Veel vaker zit het probleem in de manier waarop mensen als team samenwerken. Zijn de doelen helder? Zijn rollen duidelijk? Is er vertrouwen? Worden conflicten constructief besproken? Worden afspraken opgevolgd? Durven mensen elkaar aan te spreken? En is er voldoende focus op de paar dingen die er echt toe doen?

Een **High Performance Team** is een team dat deze vragen structureel goed weet te beantwoorden. Niet omdat alles vanzelf goed gaat, maar omdat het team bewust werkt aan vertrouwen, duidelijkheid, samenwerking, focus en accountability.

High Performance ontstaat dus niet toevallig. Het wordt gebouwd.

Zie deze video over "performance versus trust"

https://www.youtube.com/watch?v=J9CWOH8I4Ow

1.2. Wat is een High Performance Team?

Een High Performance Team is geen team dat alleen "gezellig" samenwerkt. Het is ook geen team zonder spanning, conflict of moeilijke gesprekken. Integendeel: in sterke teams worden verschillen juist zichtbaar en bespreekbaar gemaakt.

Een High Performance Team is een team dat:

Eigenschap

	Wat SAFe experts doen
Een helder en gedeeld, gezamenlijk doel en missie heeft;	Werken met ART level PI objective en iteration goals incl Confidence Vote scopes
Eigenaarschap en commitment toont voor resultaat;	Is een mindset, gedrag.
Begrijpt hoe het werk bijdraagt aan de bredere organisatieambitie;	doe je tijdens PI event, business context presentaties
Elkaar nodig heeft (onderling afhankelijk, complementair) om dat gedeelde doel te realiseren (cross functional, T-Shaped) met duidelijke rollen en verantwoordelijkheden. Respect voor ieders expertise.	Die maken een Team canvas en zorgen dat alle kennis nodig om het doel te realiseren aanwezig is.
Open en helder communiceert en transparant is	coaching door Scrum Master of SAFe/LACE Agile Coaches
Eigen Operating Protocols definieert	Coaching door de Scrum Master op duidelijke werkafspraken, meeting afspraken, agenda afspraken, hoe feedback geven en 'seek to understand before etc'
Gezonde conflicten durft te voeren;	coaching door Scrum Master of SAFe/LACE Agile Coaches
Psychologische veiligheid ervaart, open en eerlijk, respectvol, geeft en neemt	Coaching door Scrum Master of SAFe/LACE Agile Coaches

(zelf) effectief besluiten neemt en zelf conflicten kan oplossen	Coaching door Scrum Master of SAFe/LACE Agile Coaches
Diversiteit benut maar ook dezelfde normen en waarden (beliefs) deelt	Doe je door juiste samenstelling van de teamleden en door goede intakes, matching op cultuur
Continu leert en verbetert;	retro's uitvoeren/iteratie met mogelijk Ishikawa voor specifieke verdiepingen zoals op People, Proces en Technology
Personal Relationships: Plezier heeft in het werk (fun!) en met elkaar.	Heel belangrijk !

Het verschil tussen een gewoon team en een High Performance Team zit dus niet alleen in output. Het zit vooral in de kwaliteit van samenwerking die tot die output leidt. Een HPT werkt niet simpelweg harder. Een HPT werkt bewuster, scherper en beter samen.

"Building a High Performance Team (HPT) is a development journey that needs time, focus and monitoring. Facilitating development and using assessments both help to determine the status of the team and help define and execute the next needed actions for growth"

1.3. De kloof tussen ambitie en realisatie

Veel toekomstige HPT's starten met goede intenties. Er zijn plannen, doelen, roadmaps en prioriteiten. Toch ontstaat onderweg vaak verlies. Niet omdat mensen niet willen, maar omdat de uitvoering stokt.

Veelvoorkomende oorzaken zijn:

• slechte of onvolledige communicatie;

• onduidelijke prioriteiten;

• onvoldoende follow-up;

• onzichtbare bottlenecks;

• gebrek aan eigenaarschap;

• onduidelijke besluitvorming;

• onvoldoende vertrouwen;

• een cultuur waarin onderprestatie langzaam normaal wordt.

Dit is de bekende kloof tussen strategie en executie. De strategie kan goed zijn, maar als teams niet in staat zijn om samen te focussen, besluiten te nemen, te leren en elkaar accountable te houden, blijft realisatie achter.

Daarom zijn High Performance Teams geen luxe. Ze zijn een harde voorwaarde voor strategierealisatie.

1.4. Soft issues zijn harde prestatievoorwaarden

Vertrouwen, psychologische veiligheid, feedback, samenwerking en communicatie worden vaak "soft issues" genoemd. Dat klinkt alsof ze minder belangrijk zijn dan planning, budget, KPI's of processen.

In werkelijkheid is het precies andersom: als mensen elkaar niet vertrouwen, zeggen ze niet wat ze echt denken. Als er geen psychologische veiligheid is, worden fouten verstopt. Als conflicten worden vermeden of teamleden elkaar bewust ontlopen, ontstaan halfslachtige besluiten. Als rollen onduidelijk zijn, verdwijnt eigenaarschap. En als niemand elkaar aanspreekt, of durft aan te spreken, worden afspraken vrijblijvend.

Wat soft lijkt, bepaalt dus hard resultaat.

Een team zonder vertrouwen kan druk bezig zijn, maar zal moeilijk leren. Een team zonder duidelijke prioriteiten kan veel werk verzetten, maar weinig impact maken. Een team zonder accountability kan prettig samenwerken, maar structureel onderpresteren.

High Performance vraagt daarom om twee kanten tegelijk:

Kant	Betekenis
Menselijke kant	Vertrouwen, veiligheid, samenwerking, gezonde conflicten
Prestatiekant	Focus, prioriteiten, discipline, accountability en resultaat

De kracht zit in de combinatie.

1.5. Het fundament: vertrouwen en psychologische veiligheid

Vertrouwen (SAFe is een Trust Framework!) is de onderste laag van elk High Performance Team. Zonder vertrouwen ontstaat defensief gedrag. Mensen houden informatie achter, vermijden moeilijke gesprekken of beschermen vooral hun eigen positie.

Psychologische veiligheid betekent niet dat alles vriendelijk en comfortabel moet zijn. Het betekent dat mensen zich veilig genoeg voelen om eerlijk te zijn. Veilig genoeg om fouten toe te geven. Veilig genoeg om vragen te stellen. Veilig genoeg om afwijkende meningen te delen. En veilig genoeg om elkaar professioneel aan te spreken.

Een sterk team kent dus niet minder conflicten dan een zwak team. Het gaat anders met conflicten om. Conflicten worden niet persoonlijk gemaakt, maar gebruikt om betere besluiten te nemen.

Zonder vertrouwen geen gezonde conflicten. Zonder gezonde conflicten geen echt commitment. Zonder commitment geen accountability. Zonder accountability geen structureel resultaat.

1.6. Van losse mensen naar een HPT: de ontwikkelfasen

Teams worden niet in één keer High Performance, zeker niet als de teamleden elkaar niet kennen. Ze ontwikkelen zich in fasen. Een agile team doorloopt in de meeste gevallen de klassieke fasen uit het Tuckman-model (Tuckman, 1965): van Forming via Storming en Norming naar Performing, waarbij elke fase vraagt om een andere begeleiding van de Scrum Master.

1.6.1. Forming

In de beginfase zoeken mensen naar duidelijkheid. Wie zit er in het team? Wat is onze opdracht? Wie doet wat? Hoe werken we samen?

Deze fase vraagt richting, structuur en kennismaking.

1.6.2. Storming

Daarna komen verschillen naar boven. Mensen hebben verschillende meningen, belangen, stijlen en verwachtingen. Dat kan spanning geven.

Storming is geen teken dat het team faalt. Het is vaak een noodzakelijke fase. De vraag is niet of er conflicten ontstaan, maar of het team leert om die conflicten gezond te voeren.

1.6.3. Norming

In deze fase ontwikkelt het team gezamenlijke normen en afspraken. Mensen begrijpen elkaar beter, rollen worden duidelijker en samenwerking wordt voorspelbaarder.

1.6.4. Performing

In de Performing-fase werkt het team met vertrouwen, focus en kwaliteit. Er is minder ruis, meer eigenaarschap en betere besluitvorming. Het team kan omgaan met druk zonder direct terug te vallen in oud gedrag.

1.6.5. Adjourning

Teamwisselingen zijn onvermijdelijk, maar niet onschuldig: elke nieuwe samenstelling reset de groepsdynamiek en vraagt bewuste aandacht van de Scrum Master om het team zo snel mogelijk terug naar performance te begeleiden en vooral te monitoren wat er in het team gebeurt.

1.7. Prestatiediscipline: focus, prioriteiten en accountability

Een High Performance Team heeft niet genoeg aan vertrouwen alleen. Vertrouwen zonder accountability kan vrijblijvend worden. Mensen voelen zich prettig, maar spreken elkaar onvoldoende aan op resultaat. Daarom is prestatiediscipline nodig.

High Performance Teams:

• kiezen scherpe prioriteiten; (WSJF, Iteration Goals, PI Objectives)

• betrekken teamleden bij planning; (Iteration Planning events)

• maken afspraken concreet; (Refined User Stories)

• volgen voortgang structureel op; (goede Coach sync events & agile tooling)

• leren van successen en fouten; (Iteration Retro’s, I&A’s)

• maken bottlenecks zichtbaar; (Lean)

• spreken elkaar aan op gemaakte afspraken, bieden hulp aan elkaar aan; (soft skills)

• sturen bij als resultaten achterblijven. (allemaal een tandje bijzetten, niet 1 het weekend doorwerken)

Een sterk team heeft dus zowel veiligheid als scherpte.

Een veilig team zonder accountability wordt vrijblijvend. Een accountable team zonder veiligheid wordt defensief. Een High Performance Team heeft beide nodig.

1.8. De rol van MT & Lean Agile Leaders

Agile teams worden niet vanzelf HPT; leiders en de omgeving hebben daar grote invloed op. Lean-Agile Leaders zijn degenen die grote invloed hebben op het ontstaan van High Performance Teams. Zij creëren de condities waarin teams kunnen groeien of juist vastlopen.

Leiders helpen HPT’s door:

• richting te geven; (duidelijke visies tijdens PI events)

• hogere prioriteiten te beperken; (OKR’s, Strategic themes)

• duidelijke kaders te stellen; (bijvoorbeeld in Team Canvas)

• besluitvorming te versnellen, maar ook conform SAFe ‘te decentraliseren’; (Lean Agile Principle #9)

• veiligheid te creëren; (Trust)

• voorbeeldgedrag te tonen; (Walk the Talk)

• teams ruimte te geven;

• blokkades weg te nemen; (Impediments mee helpen op te lossen)

High Performance Teams vragen dus ook High Performance leiderschap.

1.9. De rollen die HPT-ontwikkeling dragen

High Performance Teams ontstaan niet vanzelf. Ze worden actief opgebouwd: door het team zelf, maar ook door de mensen die het team begeleiden en de organisatorische omgeving inrichten. Binnen SAFe zijn er twee rollen die hierin een centrale verantwoordelijkheid dragen: de Release Train Engineer en de Scrum Master. Beide hebben als expliciete opdracht om teams te helpen groeien naar High Performance, elk vanuit een eigen perspectief en werkterrein.

1.9.1. De Release Train Engineer (RTE)

De Release Train Engineer is verantwoordelijk voor de effectieve werking van de Agile Release Train als geheel en heeft als expliciete opdracht om de Train te laten doorgroeien naar High Performance. Dat betekent dat de RTE niet alleen stuurt op het leveren van waarde, maar ook actief werkt aan de randvoorwaarden waaronder teams kunnen floreren.

Concreet doet de RTE dit door:

• de waardestroom en oplossingen te optimaliseren om de PI-doelstellingen te realiseren;

• risico’s en impediments oplossen binnen de Train of, waar nodig, correct escaleren;

• inzicht en transparantie bieden op metrics als: Trainvolwassenheid, stabiliteit, voorspelbaarheid, medewerkerstevredenheid, Feature cycle time, stakeholdertevredenheid en kwaliteit van de geleverde waarde;

• de lean-agile mindset en organisatie-waarden actief te borgen in de Train;

• via de CDP de werkwijze van de Train continu te verbeteren en goede practices te delen;

• coaching en ondersteuning te organiseren voor alle rollen binnen en rondom de Train, zoals PO, SM, PM, SA, System team en BO’s met als doel High Performance teams en een High Performance Train;

• bij te dragen aan de RTE-CoP voor cross-train afstemming en verbetering;

• actief mee te bouwen aan een visie op de optimale High Performance Train en teamleden te stimuleren gezamenlijk naar die visie toe te werken;

• te zorgen dat de rituelen van de Train effectief en in cadence worden uitgevoerd.

• Hulp inroepen van de LACE voor het uitvoeren van assessments

Kort gezegd: als het team High Performance is, stroomt de waarde vanzelf

1.9.2. De Scrum Master (SM)

Waar de RTE werkt op het niveau van de Train, richt de Scrum Master zich op het agile team zelf. De Scrum Master/Team Coach heeft als kernverantwoordelijkheid het team te begeleiden naar High Performance, niet door te sturen, maar door ruimte te creëren, obstakels weg te nemen en het team te coachen in zijn groeipotentieel.

Concreet doet de Scrum Master dit door:

• de levering van oplossingen te borgen om de PI-doelstellingen van het team te realiseren;

• risico’s, operationele afhankelijkheden en impediments op te lossen binnen het team of correct te escaleren;

• inzicht en transparantie te bieden in teamvolwassenheid, voorspelbaarheid, voortgang en kwaliteit van de geleverde waarde;

• het team te laten groeien in het toepassen van organisatie-processen en de agile werkwijze gedreven door lean-agile mindset en principes;

• de team-rituelen effectief te laten verlopen en de bijdrage aan de Train te versterken;

• een High Performance Team mogelijk te maken;

• bij te dragen aan de Community of Practice (CoP) van Scrum Masters en onderling te leren

De Scrum Master is daarmee niet alleen een procesbegeleider, maar de spil in de ontwikkeling van het team. Als het team High Performance is, stroomt de waarde, en dat is precies het doel.

1.10. Mijn aanpak voor HPT-ontwikkeling

Een High Performance Team bouw je stap voor stap. Dat vraagt tijd, aandacht en een aanpak die past bij de situatie van het team. Want niet elk team start vanuit dezelfde positie. Een nieuw team dat elkaar nog nauwelijks kent, heeft andere begeleiding nodig dan een bestaand team dat al een tijdje samenwerkt maar wil doorgroeien. Daarom onderscheiden we twee trajecten: één voor nieuwe teams en één voor bestaande teams.

Wat beide trajecten gemeen hebben, is dat de ontwikkeling naar High Performance *geen eenmalige interventie* is. Het is een doorlopend proces waarbij het team regelmatig wordt begeleid, gemeten en bijgestuurd. Zowel de reis zelf als de ondersteuning hebben daarvoor tijd en focus nodig.

1.10.1. Nieuwe teams

Bij een nieuw team (ontstaan door een reorganisatie, een nieuwe teamleider, een gewijzigde teamsamenstelling of een volledig nieuwe opdracht) is de eerste periode bepalend. Teamleden kennen elkaar misschien nog nauwelijks, rollen zijn nog niet uitgekristalliseerd en er is nog geen gemeenschappelijke werkwijze. Juist in die beginfase wordt de basis gelegd voor vertrouwen, samenwerking en gedeeld eigenaarschap.

Het traject voor nieuwe teams start met een intake met de RTE, de SM of een vertegenwoordiger van de organisatie van het betreffende Agile Team. Vanuit die intake wordt een gefaseerd begeleidingstraject ingericht. In de eerste zes maanden (F6M) wordt het team intensief begeleid aan de hand van specifieke scripts die aansluiten op de fase waarin het team zich bevindt. Na zes maanden wordt de voortgang gemeten met een diagnostisch instrument. Op basis van die meting wordt het vervolg bepaald: een nieuwe begeleidingscyclus (A6M), opnieuw gevolgd door een meting.

Het traject voor nieuwe teams ziet er samengevat als volgt uit:

• Intake met onder andere matching op cultuur, normen en waarden.

• Een specifiek onboardingsprogramma

• Specifieke materietrainingen, soft en hard. Hard met name Agile specifieke trainingen.

• Begeleiding eerste 6 maanden op basis van F6M-scripts (First 6 Months), primair door de SM, waarbij een verandermanager uit de LACE of een senior Agile Coach over de schouder meekijkt.

• Meting met het diagnostisch instrument (bijvoorbeeld SAFe’s Team & Technical Assessment, maar er zijn nog vele andere op de markt beschikbaar);

• Oplossen van de geïdentificeerde verbeterpunten

• Wanneer gedrag structureel niet past bij de teamafspraken of de gewenste cultuur, moet dit zorgvuldig bespreekbaar worden gemaakt en waar nodig via de lijnorganisatie worden opgelost.

• Vervolgtraject na 6 maanden op basis van A6M-scripts (After 6 Months)

• Herhaalde meting met het diagnostisch instrument

1.10.2. Bestaande teams

Bestaande teams hebben al een geschiedenis. Ze kennen elkaars werkwijze, hebben patronen ontwikkeld en weten, bewust of onbewust, hoe ze met spanning en samenwerking omgaan. Dat kan een voordeel zijn, maar ook een valkuil: ingesleten gedrag dat het team belemmert om verder te groeien, wordt minder snel zichtbaar.

Voor bestaande teams begint het traject daarom met een meting: het diagnostisch instrument brengt in kaart waar het team staat op het gebied van doelen, rollen, samenwerking, vertrouwen en accountability. Die meting vormt de basis voor een gerichte begeleidingscyclus, met een lijst van concrete Improvement items, van zes maanden (A6M). Na die periode volgt een nieuwe meting om de ontwikkeling in beeld te brengen en het volgende traject te bepalen.

Het traject voor bestaande teams ziet er samengevat als volgt uit:

• Meting met het diagnostisch instrument

• Begeleiding op basis van A6M-scripts (After 6 Months)

• Herhaalde meting met het diagnostisch instrument

• Vervolgtraject op basis van A6M-scripts

• Herhaalde meting met het diagnostisch instrument

In beide gevallen geldt: de begeleiding is cyclisch en data-gedreven. Het diagnostisch instrument zorgt voor inzicht in de feitelijke stand van het team, terwijl de begeleiding gericht is op de specifieke groeistap die het team op dat moment nodig heeft. Zo wordt HPT-ontwikkeling geen eenmalig project, maar een structureel onderdeel van de manier waarop het team werkt en groeit.

1.11. Conclusie: vertrouwen plus discipline

High Performance Teams ontstaan wanneer vertrouwen, focus en discipline elkaar versterken. Vertrouwen zorgt dat mensen eerlijk durven zijn. Focus zorgt dat energie naar de juiste doelen gaat. Discipline zorgt dat afspraken worden opgevolgd en resultaten zichtbaar worden.

Een HPT is dus geen team zonder problemen. Het is een team dat problemen sneller bespreekbaar maakt, beter oplost en ervan leert.

De kern is eenvoudig:

High Performance ontstaat niet door harder te werken, maar door bewuster samen te werken aan het juiste resultaat.

Organisaties die structureel betere resultaten willen, moeten daarom niet alleen investeren in processen, plannen en systemen. Ze moeten investeren in de kwaliteit van hun teams. Want uiteindelijk wordt strategie niet gerealiseerd door PowerPoint, governance of rapportages. Strategie wordt gerealiseerd door teams die samen weten wat belangrijk is, elkaar vertrouwen, scherp blijven op resultaat en elke dag beter willen worden.

De vraag is niet óf jouw team High Performance kan worden. De vraag is wanneer je er serieus mee begint.

Begin klein. Bespreek met je team: wat gaat er goed in onze samenwerking, en wat kost ons het meeste energie? Dat gesprek alleen al is een eerste stap naar meer vertrouwen en focus.

Voor Scrum Masters en RTE's: gebruik een assessment om te meten waar het team nu staat. Niet als oordeel, maar als startpunt. Zonder meting geen richting, zonder richting geen groei.

En voor leiders: het grootste cadeau dat je een team kunt geven is niet meer resources of een beter proces. Het is duidelijkheid, ruimte en het vertrouwen dat ze het zelf mogen oplossen.

High Performance is geen eindbestemming. Het is een manier van werken die je elke dag opnieuw kiest.

39. Waarom roulerend voorzitterschap goed past in een SAFe-omgeving

Waarom roulerend voorzitterschap goed past in een SAFe-omgeving

In traditionele organisaties is het gebruikelijk dat vergaderingen een vaste voorzitter hebben. Die voorzitter bepaalt vaak de agenda, bewaakt de tijd en stuurt: bewust of onbewust: de discussie. In Lean-Agile organisaties ligt dat anders. Binnen SAFe en Agile omgevingen zien we vaak dat teams kiezen voor **roulerend voorzitterschap** in plaats van een vaste voorzitter.

Dat is geen toevallige keuze, maar sluit aan bij de kernprincipes van Lean-Agile werken: samenwerking, gelijkwaardigheid, zelforganisatie en continue ontwikkeling van mensen.

Ik pas dit principe zelf in veel omgevingen toe, meestal met goede resultaten. Wel altijd **situationeel**: voor sommige events: zoals PI Planning of Inspect & Adapt: is een vaste facilitator of event owner logischer. Maar voor reguliere overleggen binnen teams of bijvoorbeeld een LACE kan roulerend voorzitterschap zeer goed werken.

1. Zelfsturende teams en gelijkwaardigheid

SAFe gaat uit van **zelforganiserende Agile Teams**. Dat betekent dat teams gezamenlijk verantwoordelijk zijn voor hun werk, planning en samenwerking.

Wanneer één persoon structureel de rol van voorzitter vervult, kan dat onbedoeld een hiërarchische dynamiek creëren. De voorzitter lijkt dan al snel “de baas van het overleg”.

Door het voorzitterschap te **laten rouleren**, krijgt ieder teamlid de kans om een keer het overleg te faciliteren. Dat versterkt het gevoel dat iedereen een gelijkwaardige bijdrage levert aan het functioneren van het team.

2. Transparantere discussies

Een vaste voorzitter heeft: bewust of onbewust: invloed op de richting van een gesprek. Door vragen anders te stellen of bepaalde onderwerpen te benadrukken kan de discussie een bepaalde kant op bewegen.

Wanneer de rol rouleert, ontstaat vaak een **neutralere en transparantere dynamiek**. Teamleden ervaren dat het overleg van iedereen is, en niet van één persoon.

Dit bevordert open communicatie en zorgt ervoor dat verschillende perspectieven beter gehoord worden.

3. Meer betrokkenheid en empowerment

Lean-Agile organisaties proberen beslissingen zo dicht mogelijk bij het werk te leggen. Dat vraagt om teams die verantwoordelijkheid nemen.

Door de rol van facilitator of voorzitter te laten rouleren:

• voelen teamleden zich meer betrokken bij het overleg

• groeit het gevoel van eigenaarschap

• nemen mensen eerder initiatief

Het resultaat is vaak dat teams **actiever deelnemen aan discussies en besluitvorming**.

4. Iedereen de kans geven om te groeien

In veel teams zie je dat één persoon na verloop van tijd erg goed wordt in het leiden van meetings. Dat is positief, maar heeft ook een nadeel: anderen krijgen minder kans om die vaardigheid te ontwikkelen.

Door voorzitterschap te laten rouleren:

• leren meerdere teamleden hoe je een meeting structureert

• ontwikkelen mensen facilitation skills

• groeit de collectieve professionaliteit van het team

Dat past goed bij het SAFe-principe van **relentless improvement en continuous learning**.

5. Minder onnodige structuur en bureaucratie

Lean denkt kritisch na over elke rol en elk proces. Als iets geen waarde toevoegt, is het potentieel **waste**.

Een vaste voorzitter kan in sommige situaties nuttig zijn, maar in veel teamoverleggen voegt het weinig toe. Het overleg kan prima functioneren met een **lichte structuur en gedeelde verantwoordelijkheid**.

Door de rol simpelweg te laten rouleren voorkom je dat een extra formele laag ontstaat die eigenlijk niet nodig is.

6. Evenwichtige verdeling van verantwoordelijkheden

In teams zoals een LACE of Agile Team is het belangrijk dat werk en verantwoordelijkheden zo eerlijk mogelijk verdeeld zijn.

Roulerend voorzitterschap zorgt ervoor dat:

• verschillende stijlen en perspectieven worden gehoord

• iedereen een bijdrage levert aan het functioneren van het team

• de werklast niet altijd bij dezelfde persoon ligt

Dat draagt bij aan een **gezonde teamdynamiek**.

Situationeel toepassen

Roulerend voorzitterschap betekent niet dat het overal moet worden toegepast. Sommige events in SAFe vragen juist om een duidelijke facilitator of event owner.

Voorbeelden:

• **PI Planning**: geleid door de RTE

• **Inspect & Adapt**: duidelijke facilitatie nodig

• **ART events**: vaak vaste rolverdeling

Maar voor veel **team- of LACE-overleggen** kan roulerend voorzitterschap uitstekend werken.

Samengevat:
Roulerend voorzitterschap ondersteunt de Lean-Agile principes van gelijkwaardigheid, zelforganisatie, transparantie en continue ontwikkeling van mensen. Mits situationeel toegepast kan het een eenvoudige maar effectieve manier zijn om teams sterker en volwassener te maken.

40. Meer meetings door SAFe? Wat er werkelijk verandert

Vraag: *"Als we Agile/Scrum/SAFe gaan werken, krijgen we heel veel (extra) meetings!"*

Het korte antwoord

Valt reuze mee. Een SAFe-team besteedt circa **8%** van zijn tijd aan Scrum events bij een 2-weekse Sprint en circa **6,5%** bij een 3-weekse Sprint. Ruim 90% van de tijd blijft over voor het echte werk.

Aanleiding

Wanneer organisaties beginnen met SAFe, komt vaak de vraag op of er niet te veel tijd opgaat aan allerlei extra meetings. Zo ook tijdens een recente bedrijfsmatige "zeepkist-sessie". Er heerst een wijdverbreid gevoel dat de overhead bij agilisten hoog is en dat er te weinig tijd overblijft voor daadwerkelijk werk.

In dit hoofdstuk onderzoek ik die perceptie. Ik reken de werkelijke overhead van Scrum events in 2- en 3-weekse Sprints door in procenten, en maak het gevoel zo feitelijk.

Scrum events en hun doel

Voordat we in de cijfers duiken: welke events zijn er eigenlijk, en waarom? SAFe heeft goed nagedacht over welke meetings nodig zijn, hoe je zo efficiënt mogelijk afstemt en hoe de verschillende SAFe-niveaus met elkaar verbonden worden. En dat alles in een vaste **cadans**, met rust en regelmaat:

• dezelfde agenda

• dezelfde mensen

• dezelfde tijden en locatie

• hetzelfde ritme

Event	Wat gebeurt er?	Waarom is het nodig?
PI Planning	Eens per 10 weken stemmen alle teams samen hun nieuwe PI-doelen af en maken ze het increment-plan.	Daarna kunnen de teams 10 weken vooruit en leveren ze op wat ze tijdens de PI hebben beloofd.
Sprint Planning	Het team plant de doelen en het werk voor de komende Sprint.	Je zult het werk toch ergens moeten plannen.
Daily Stand-up (Team Sync)	Dagelijks kort overleg over voortgang en obstakels.	Alle teamleden zijn elke 24 uur ge-aligned.
Sprint Review	Aan het einde van de Sprint wordt het werk gedemonstreerd en feedback verzameld.	Je wilt niets bouwen waar gebruikers niet op zitten te wachten, en frequent horen of ze tevreden zijn.
Sprint Retrospective	Het team evalueert en verbetert proces en samenwerking.	Zonder retro ontstaan onuitgesproken frustraties en onopgeloste problemen, die van kwaad tot erger gaan.
Backlog Refinement	Geen ceremonie, maar een doorlopend proces (maximaal circa 10% van de tijd). Het team vormt gezamenlijk begrip over de vraag, acceptatiecriteria, value statements, etc.	Zonder refinement ontstaan misverstanden, lagere kwaliteit, discussies tijdens het bouwen en ad-hoc meetings.

Met andere woorden: **SAFe-agilisten vergaderen niet om het vergaderen**, maar om concrete doelen te realiseren: afstemmen, alignen, transparantie, planningen maken, problemen snel vinden, continu verbeteren en frequent gebruikersfeedback ophalen.

Hoeveel tijd kost het nu écht?

Uitgangspunten

• Theoretisch duren sommige meetings (Sprint Planning, Retrospective) langer, maar in de praktijk zijn onderstaande waarden goed aan te houden.

• **PI Planning** is buiten de berekening gelaten: SAFe kent hiervoor een aparte iteratie, puur voor 'Planning & Innovation'.

• **PO Sync** en **Coach Sync** zijn alleen voor Scrum Masters en Product Owners, niet voor teamleden.

• Een werkweek telt 40 uur.

Berekening

Event	2-weekse Sprint (10 dagen)	3-weekse Sprint (15 dagen)
Sprint Planning	2 uur	2 uur
Daily Stand-ups (15 min/dag)	2,5 uur	3,75 uur
Sprint Review	1 uur	1 uur
Sprint Retrospective	1 uur	1 uur
Totaal aan events	**6,5 uur**	**7,75 uur**
Beschikbare werkuren	80 uur	120 uur
Meeting-overhead	**8,13%**	**6,46%**

Rekenvoorbeeld 2 weken: 6,5 uur / 80 uur × 100% = 8,13%. Rekenvoorbeeld 3 weken: 7,75 uur / 120 uur × 100% = 6,46%.

Conclusie

SAFe brengt een zekere meeting-overhead met zich mee, maar die tijdsinvestering is klein in verhouding tot wat de meetings opleveren. Met slechts **8,13%** (2 weken) en **6,46%** (3 weken) blijft er ruim voldoende tijd over voor het daadwerkelijke werk, terwijl de events zorgen voor een effectievere en efficiëntere samenwerking.

De overhead kan zelfs nog verder omlaag door bijvoorbeeld:

• de Sprint Review korter te maken;

• de Sprint Planning compacter te houden;

• de Team Sync over te slaan op de dag van de Review.

Goed geïmplementeerd en gefaciliteerd leveren SAFe events dus meer op dan ze kosten aan tijd en middelen. **De volgende keer dat een collega klaagt over "te veel meetings", kun je hem of haar met deze feiten geruststellen.**

41. SAFe Iteration Review: handleiding voor een effectieve Iteration Review

De Sprint Review, in SAFe® de **Iteration Review** genoemd, is een kerngebeurtenis aan het einde van elke iteratie (doorgaans 2 tot 4 weken). Het is het moment waarop het team laat zien wat er is opgeleverd en stakeholders hun feedback geven.

Doel: transparantie creëren, samenwerking versterken en het product op koers houden.

Wat hoort er in een Iteration Review?

• Demo van opgeleverd werk: wat is er daadwerkelijk gedaan en werkend?

• Productincrement tonen: software, prototype of ander resultaat.

• Feedback ophalen: vragen, suggesties, richting bepalen.

• Team Backlog review: PO licht prioriteiten en volgend werk toe.

• Aanpassing en planning: backlog bijstellen op basis van feedback.

• Vieren: erken de inspanning en prestaties van het team.

Voorbereiding

Inhoud & focus

• Ken de Iteration Goals: demo uitsluitend wat is afgesproken. Twijfel? Check bij de SM of PO.

• Oefen en test grondig: navigeer door de demo, anticipeer op vragen en check of alles technisch werkt.

• Tijdscheck: neem jezelf op en controleer of je binnen de tijd blijft (bijv. max. 20 minuten).

• Agenda tijdig versturen: stuur de agenda vroeg genoeg zodat stakeholders kunnen aansluiten.

• Juiste deelnemers: nodig échte eindgebruikers uit, niet alleen management.

Keuze van de presentator

Niet iedereen is een even sterke presentator, en dat hoeft ook niet. Kies de meest geschikte persoon, maar geef ook anderen de kans om te groeien.

Visuele ondersteuning

• Gebruik visuals (schema's, screenshots) alleen waar ze iets verduidelijken.

• Dia's: maximaal 3 bullets per slide, grote leesbare fonts, schoon en rustig ontwerp.

• Minder is meer: geen gimmicks, geen overbodige tekst.

IT-setup (remote)

Checklist voor remote sessies

• Stabiel internet: gebruik bij voorkeur kabel.

• Goede microfoon: investeer in externe audio.

• Camera aan, altijd. Bij grote groepen (20+): spreek af wat de standaard is.

• Schermresolutie: zorg dat content goed leesbaar is.

• Dubbel scherm: handig voor presentatie én deelnemersoverzicht.

• Geen afleiding op de achtergrond: rust, orde en professionele setting.

• Camera op ooghoogte of iets erboven, nooit van onderen.

• Backup device: klaarstaand bij belangrijke sessies (warm standby).

Uitvoering

Opname & toestemming

• Kondig aan dat de sessie wordt opgenomen.

• Vraag expliciete toestemming vóór je start.

• Respecteer wie niet wil meedoen aan de opname.

Demotips

• Blijf gefocust: demo alleen de highlights, geen informatie-overload.

• Neem de tijd per scherm: leg uit wat stakeholders zien, zeker bij nieuwe interfaces.

• Navigeer bewust: leg elke stap uit, ga er niet van uit dat iedereen bekend is met het systeem.

Opbouw van de demo

Bouw de demo als volgt op:

1. Start met de pijnpunten en context van de user story.

2. Schets de (PI-)doelstellingen: wat wilden we bereiken?

3. Demo de user story stap voor stap.

4. Bewijs dat de voordelen zijn gerealiseerd.

Storytelling & authenticiteit

• Vertel het vanuit de gebruiker: *'John, werkzaam bij Shell, solliciteert bij ASML. HRM-manager William verwerkt de aanvraag.'*

• Gebruik echte data: dummy-data werkt vervreemdend.

• Maak het concreet met echte namen en situaties.

Communicatie tijdens de demo

• Spreek duidelijk, in een rustig tempo.

• Maak oogcontact, fysiek of via de camera.

• Check regelmatig of iedereen er nog bij is.

• Haal direct feedback op: gebruik bijv. Mentimeter of Microsoft Forms.

Na de demo

Follow-up

• Deel documentatie, links en relevante bronnen na afloop.

• Communiceer hoe feedback wordt meegenomen in het verdere ontwikkelproces.

• Sluit af met de volgende stappen en een helder tijdpad.

Retrospectief

Loop deze handleiding na afloop punt voor punt door met het kernteam (PO, SM, etc.). Wat ging goed? Wat kan beter?

Een goede demo gaat verder dan features tonen: hij bouwt vertrouwen en versterkt communicatie met stakeholders.

Succes met je SAFe® Iteration Review!

Deel 6: Work items, portfolio en waarde

42. Missie, visie, strategie, Strategic Themes, OKR’s en kernwaarden bij Defensie

De samenhang tussen missie, visie, strategie, Strategic Themes, OKR’s en kernwaarden

Inleiding

Bij veel van mijn opdrachtgevers worden de directie- en managementtermen *missie, visie, strategie, strategic themes, OKR’s en kernwaarden* vaak door elkaar gebruikt. Dit leidt echter tot verwarring en versnippering van focus en een minder consistent en doordacht organisatieontwerp.

In dit hoofdstuk leg ik je uit wat deze begrippen betekenen, hoe ze samenhangen en hoe jouw organisatie ze effectief kan toepassen binnen een SAFe-transformatie.

De samenhang van deze managementelementen in SAFe

Laten we beginnen met een korte uitleg van de strategische ontwerpelementen, die ik hieronder per element uitleg, voorzien van tips en praktijkvoorbeelden.

1. **Missie -** bepaalt het *waarom* van de organisatie. Die moet ambitie en energie uitstralen.

2. **Kernwaarden** - bepalen *hoe* we ons gedragen in het realiseren van onze missie. Daar heb ik hele duidelijke beelden over (-:

3. **Visie** - geeft aan *waar* we naartoe willen, ook wel ‘de stip op de horizon’ genoemd

4. **Strategie** - beschrijft *hoe* we daar komen.

5. **Strategic Themes** - bepalen de *focusgebieden*.

6. **SAFe Portfolio’s -** Een of meer strategic themes worden gerealiseerd middels 1 of meer SAFe portfolio’s

7. **OKR’s per Portfolio** maken deze thema’s *meetbaar en uitvoerbaar*.

Element 1: Mission Statement

De missie beschrijft het fundamentele doel van een organisatie en geeft bovenal antwoord op de vraag: *waarom komen wij elke dag naar onze werkgever, aan welk hoger doel draag ik bij?*

Het definieert de primaire doelstellingen die verband houden met de behoeften van klanten en de waarden van de teamleden. De missie legt de brede doelen vast waarvoor een organisatie, afdeling, sector of bedrijf is opgericht en is primair bedoeld als een intern kompas voor leiderschap, werknemers en stakeholders.

Checklist voor een goede missie:

• Kort en krachtig: Op te zeggen door een kind van 10.

• Gericht op impact: Wat betekent onze missie voor Defensie en onze gebruikers?

• Inspirerend en positief: Moet medewerkers en stakeholders motiveren.

• Ambitieus en uitdagend: Mag een tikkeltje brutaal zijn, veelal in commerciële concurrentiegevoelige omgevingen

• Duurzaam en tijdloos: Blijft relevant in de loop der tijd.

• Duidelijke doelgroep en toegevoegde waarde: Voor wie doen we dit en waarom?

Ontwerpformule voor een missie

“WIJ [KERNACTIVITEIT X] VOOR [DOELGROEP Y] OM [IMPACT Z] TE BEREIKEN”

of

ONZE AMBITIE IS OM [IMPACT Z] VOOR [DOELGROEP Y] TE BEREIKEN DOOR [KERNACTIVITEIT X]

Voorbeelden

"Wij moderniseren de IT-infrastructuur van de landmacht om militairen optimaal te ondersteunen bij hun missies en zo de operationele effectiviteit te verhogen."

of

"Onze ambitie is om de meest wendbare en robuuste IT-organisatie binnen Defensie te worden. Dit doen we door SAFe in te zetten als raamwerk voor snelle innovatie, samenwerking en continue verbetering."

naslagwerk en theoretische kaders

• https://msb.franklincovey.com/

• https://www.nomonkeybusiness.eu/wat-is-een-missie/

• https://www.marketingscriptie.nl/missie-visie-formuleren/

Element 2: Kernwaarde

Kernwaarden vormen de basisprincipes en overtuigingen die het gedrag van een organisatie en haar medewerkers sturen. Ze bepalen *hoe* de missie wordt uitgevoerd en zorgen voor een gedeelde bedrijfscultuur.

Kenmerken van kernwaarden

• Tijdloos en fundamenteel voor de organisatie.

• Niet afhankelijk van de oprichter of het huidige management.

• Helpen bij het nemen van beslissingen en het bepalen van prioriteiten.

• Liggen ten grondslag aan alle strategische initiatieven en interacties.

• Indien gedefinieerd door een afdeling, vaak aanvullend of overlappend met de kernwaarden van de organisatie.

Voorbeeld (moeten in HRM gesprekken meetbaar en afvinkbaar zijn)

• **Afspraak is afspraak**: We zijn betrouwbaar en consequent.

• **Transparantie**: We communiceren open en eerlijk.

• **Continu verbeteren**: We streven naar First Time Right.

• **Samenwerking**: We werken cross-functioneel en klantgericht.

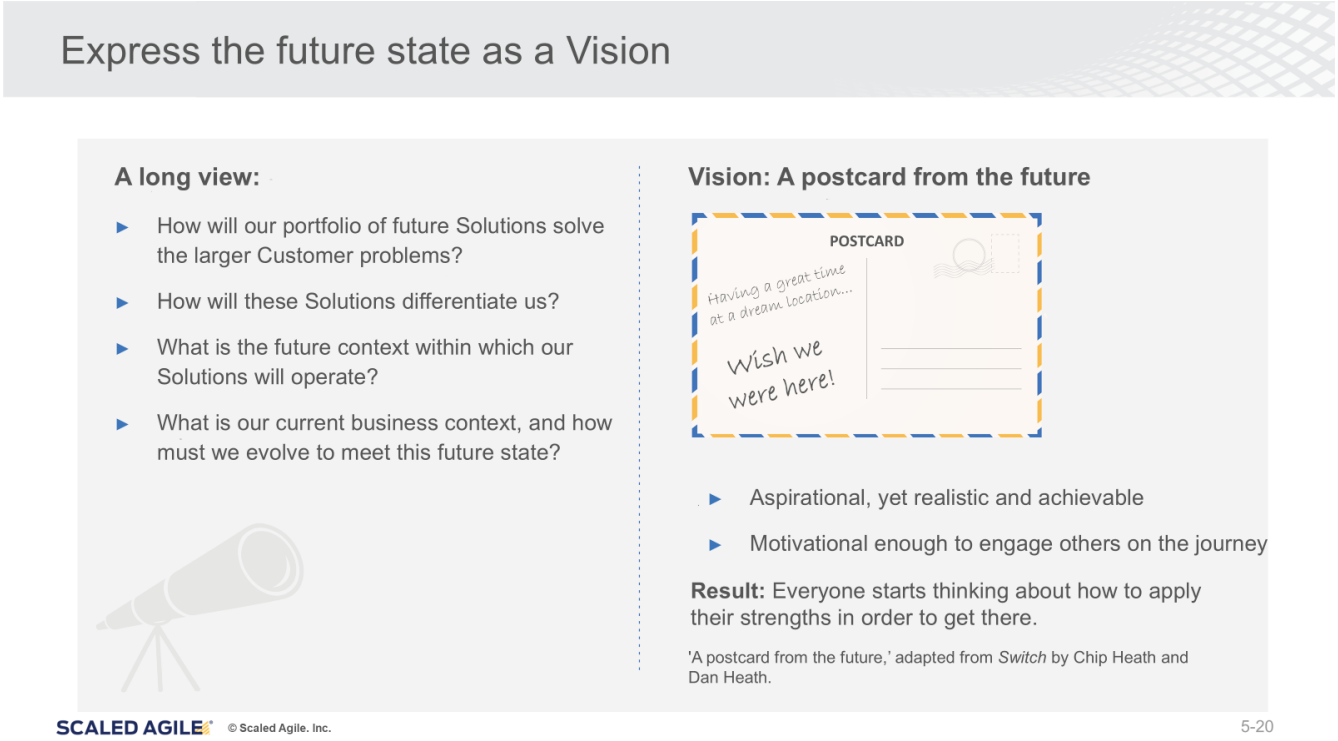
naslagwerk en theoretische kaders

• https://www.personio.nl/hr-woordenboek/kernwaarden-bedrijf

• https://hkb-advies.nl/kernwaarden/

Element 3: visie

Een visie schetst een toekomstbeeld en geeft richting aan waar de organisatie naartoe wil. Zowel organisatie als producten & diensten ('Solutions'), markten. Een missie beschrijft *waarom* de organisatie bestaat, terwijl een visie beschrijft *waar de organisatie naartoe wil* en wat de ultieme ambitie is.



Een goede visie moet voldoen aan de volgende eisen:

• Inspirerend en ambitieus: Het moet energie geven en richting bieden.

• Langetermijnperspectief: Minimaal 5 tot 10 jaar standhouden.

• Overkoepelend: De visie overstijgt operationele en financiële doelen.

• Helder en eenvoudig: Iedereen in de organisatie moet de visie kunnen begrijpen en uitdragen.

• Extern gericht: De visie moet inspelen op externe ontwikkelingen en de omgeving waarin de organisatie opereert.

• Beschrijft de SOLL-situatie.

Voorbeelden

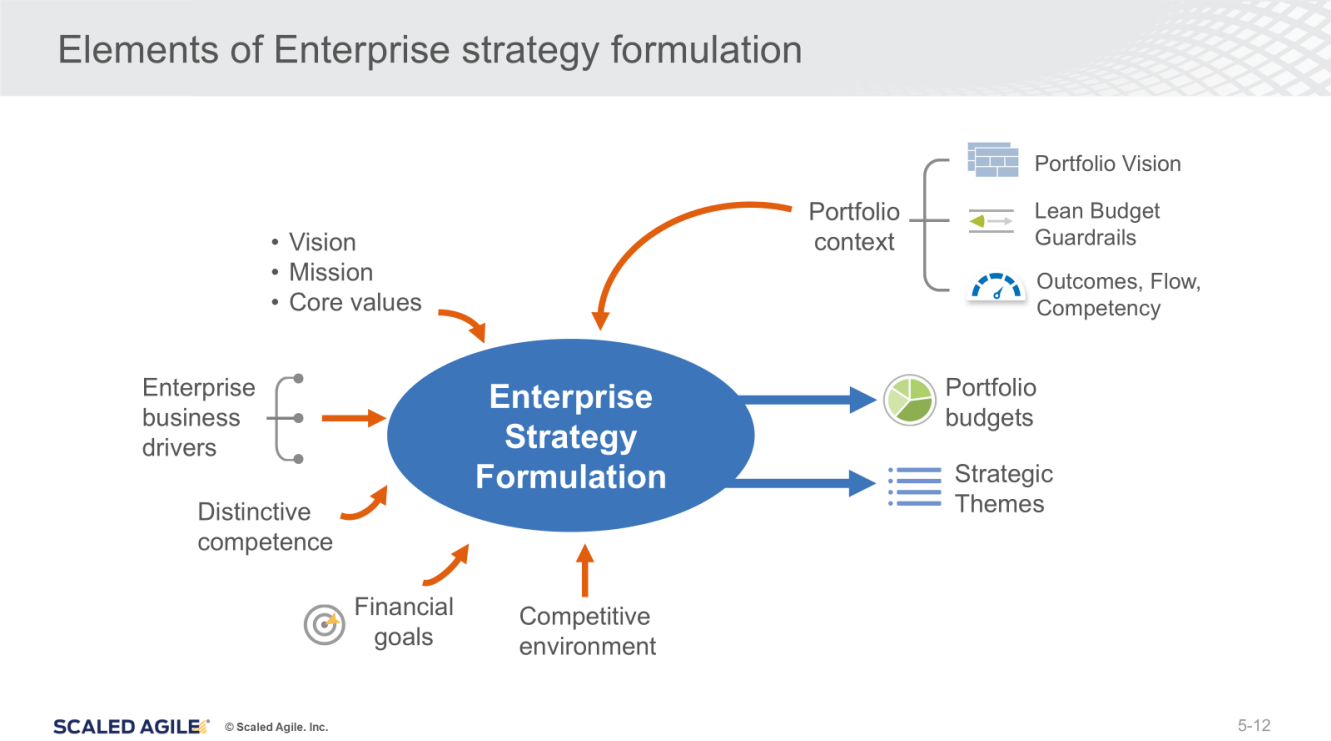
We willen toe naar een splitsing van de hiërarchie en willen de primaire business processen onderbrengen in een apart OS #2. We willen onze gebruikers ook in dat OS #2 opnemen en dichter bij de maak-mensen neerzetten.

Naslagwerk en theoretische kaders

• https://framework.scaledagile.com/solution-vision/

Element 4: strategie

Strategie beschrijft hoe de organisatie haar visie gaat realiseren. Dit zijn de concrete aanpak en keuzes die de organisatie maakt. Strategie gaat niet over *een lange lijst doelen of vage ambities*. 'Meer omzet' realiseren is geen strategie. Strategie is ook geen losstaand plan, maar een samenhangend geheel van keuzes die elkaar versterken. Zonder scherpe keuzes en duidelijke uitvoering blijft strategie slechts een intentie. Een organisatiestrategie is de combinatie van slimme elementen, besluiten, aanpakken, werkwijzen, die je een unieke marktpositie of organisatiekracht geven.



Een goede strategie gaat niet alleen over wat je *wel* doet, maar vooral ook over wat je *niet* doet.

Het draait om:

• Waar je concurreert (commercieel) of in welk speelveld je je begeeft (publieke sector)

• Welke unieke waarde je biedt

• Hoe je je middelen inzet

• Hoe je dit duurzaam maakt

IKEA is een perfect voorbeeld

Ze concurreren niet in de luxe meubelmarkt, maar focussen op betaalbaar en stijlvol meubilair. Hun strategie combineert zelfmontage met efficiënte supply chains en een uniek winkelconcept om concurrenten voor te blijven. En dat alles in handige dozen die iedereen zelf kan vervoeren met duidelijke instructies.

Strategie voorbeelden

• We gaan silo's elimineren en alleen nog werken in Value Streams

• We gaan long-lived Agile Teams bouwen, die we Cross Functional en Self Managing maken

• We gaan besluitvorming decentraliseren

• We gaan vergaand simplificeren

• We gaan standaardiseren

• We gaan een proces in van ‘HR for Culture’

• We gaan afscheid nemen van Legacy Policies teneinde flow te genereren.

• etc

naslagwerk en theoretische kaders

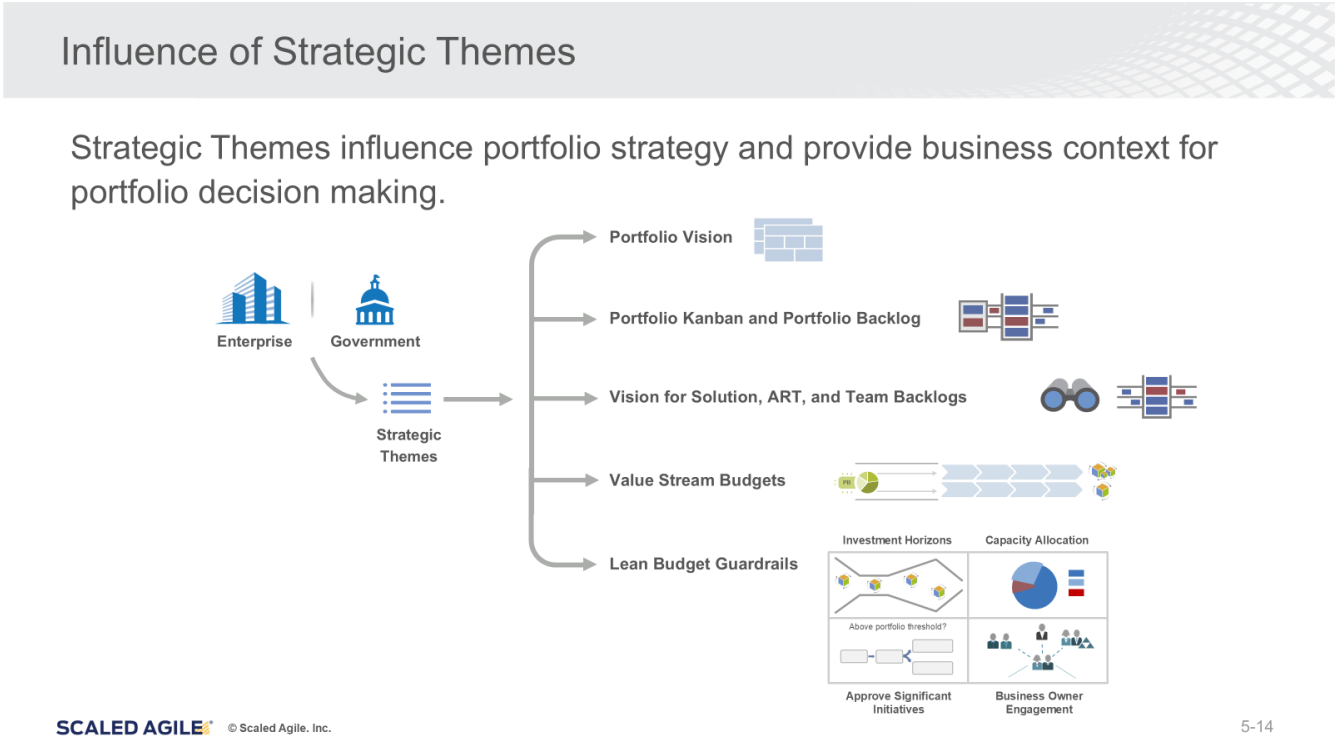
• https://framework.scaledagile.com/strategic-themes/

• https://framework.scaledagile.com/enabling-strategic-alignment-with-an-enterprise-strategy-room

• https://framework.scaledagile.com/?s=strategy

Element 5: Strategic Themes

Strategic Themes zijn de kernprioriteiten die de strategie ondersteunen en helpen om doelen te bereiken. Ze verbinden de strategische visie met de uitvoering in SAFe. Strategic Themes vertalen de strategie naar concreet uitvoerbare *focusgebieden* en verbinden de strategische visie met de uitvoering in SAFe.



Voorbeelden voor Defensie:

• Theme #1: Digitale veerkracht verhogen door cyberbeveiliging en AI-toepassingen

• Theme #2: Snellere en schaalbare inzet van IT-systemen en organisatie (SAFe)

• Theme #3: Data-driven besluitvorming verbeteren binnen Defensie

• Theme #4: Ontwikkelen van dronetechnologie en oorlogsvoering

Strategic Themes worden verder uitgewerkt in SAFe Portfolio’s, die de uitvoering hiervan faciliteren.

naslagwerk en theoretische kaders

• https://framework.scaledagile.com/strategic-themes/

Element 6: Objectives & Key results

OKR’s maken strategic themes meetbaar en uitvoerbaar en moeten eens per half jaar worden bekeken en mogelijk worden geherdefinieerd. OKR’s zijn concrete, meetbare doelen die voortvloeien uit Strategic Themes en helpen om strategieën te realiseren.

Hoe werkt dit?

• *Objectives*: Wat willen we bereiken? (kwalitatief en ambitieus)

• *Key Results*: Hoe meten we of we slagen? (specifiek en meetbaar). Per objective zijn 2-5 key results ideaal, meer is niet goed te managen

Voorbeeld voor Defensie

Objective & Key Result	Meting
THEME #2: SNELLERE EN SCHAALBARE ORGANISATIE (SAFe)	
KR1: De Net Promoter Score (NPS) van interne en externe klanten stijgt met 20% binnen zes maanden na de eerste PI.	Online Survey
KR2: Minimaal 85% van de geplande features wordt op tijd geleverd zoals beloofd tijdens het PI event	Uit Jira Align
KR3: Het aantal urgente escalaties en prioriteitswisselingen in de Team backlog neemt met 40% af, doordat teams voorspelbaarder werken.	Uit Jira
KR4: De Time to Field voor het opleveren van Epics wordt na vier PI’s met 50% verkort, doordat het TOM goed is ontworpen.	Uit Jira Align

naslagwerk en theoretische kaders

• https://www.perdoo.com/resources/online-guides/okr-guide

• https://framework.scaledagile.com/strategic-themes/

Conclusie

Het begrijpen en toepassen van missie, visie, strategie, strategic themes, OKR’s en kernwaarden in samenhang zorgt voor een coherente en goed gedefinieerde organisatieaanpak. Dit voorkomt fragmentatie, inconsistentie, versnippering van focus en onduidelijkheid binnen de organisatie.

Door SAFe als denkraamwerk te gebruiken, kunnen strategische thema’s effectief worden vertaald naar concrete initiatieven en meetbare resultaten. Zoals het werktuigbouwkundig gezegde luidt: *Meten is weten, gokken is dokken en gissen is missen.*

Met een duidelijke missie en visie, ondersteund door strategische thema’s en OKR’s, kan je je voorbereiden op een toekomst waarin IT wendbaar en effectief blijft bijdragen aan operationeel succes en van je eigen TOM een waar huzarenstuk maken!

Hieronder staan enkele voorbeelden uit diverse sectoren.

Auto’s

1. **Tesla: "Accelerating the world’s transition to sustainable energy."**

• Tesla’s missie is niet alleen om elektrische auto’s te maken, maar om de hele wereld duurzamer te maken. Hun innovaties in batterijtechnologie en autonoom rijden maken dit een inspirerend voorbeeld.

2. **Porsche: "Driven by Dreams."**

• Porsche verbindt hun merk met passie en dromen, waarbij ze de ultieme rijervaring willen bieden voor autoliefhebbers. Dit gaat verder dan alleen snelheid; het gaat over emotie en beleving.

3. **Ferrari: "We build cars, symbols of Italian excellence, that bring driving emotions to life."**

• Ferrari’s missie draait om pure emotie en exclusiviteit. Ze bouwen geen gewone auto’s, maar rijdende kunstwerken die status, snelheid en passie uitstralen.

Hifi & Audio

1. **Bang & Olufsen: "Create magical listening and viewing experiences."**

• Dit Deense merk legt de nadruk op de ultieme audiovisuele beleving, waarin design, vakmanschap en technologie samenkomen.

2. **Bowers & Wilkins: "Listen and you’ll see."**

• Deze luidsprekerfabrikant richt zich op pure geluidskwaliteit, zodat muziek zo dicht mogelijk bij de originele opname blijft. Hun missie benadrukt hoe geluidservaring ook visueel en emotioneel kan zijn.

3. **Sonos: "Fill every home with music."**

• Sonos wil muziek naadloos integreren in het dagelijks leven en zorgt ervoor dat mensen overal in huis van audio van hoge kwaliteit kunnen genieten.

Beauty

1. **L'Oréal: "Create the beauty that moves the world."**

• L’Oréal wil schoonheid toegankelijk maken voor iedereen en combineert innovatie met duurzaamheid en inclusiviteit.

2. **Dior: "Dream in Dior."**

• Dior verbindt zijn merk met luxe, elegantie en de kracht van verbeelding. Hun missie draait om creativiteit en emotionele impact.

3. **Estée Lauder: "Bringing the best to everyone we touch and being the best in everything we do."**

• Dit beautymerk focust op premium kwaliteit, empowerment en innovatie in cosmetica en huidverzorging.

Kleding & Fashion

1. **Nike: "To bring inspiration and innovation to every athlete in the world."**

• Nike’s missie is breed en krachtig: ze zien iedereen als een potentiële atleet en willen inspireren om grenzen te verleggen.

2. **Patagonia: "We're in business to save our home planet."**

• Dit outdoormerk heeft een van de meest impactvolle missies: ze zetten zich volledig in voor duurzaamheid en milieubewust ondernemen.

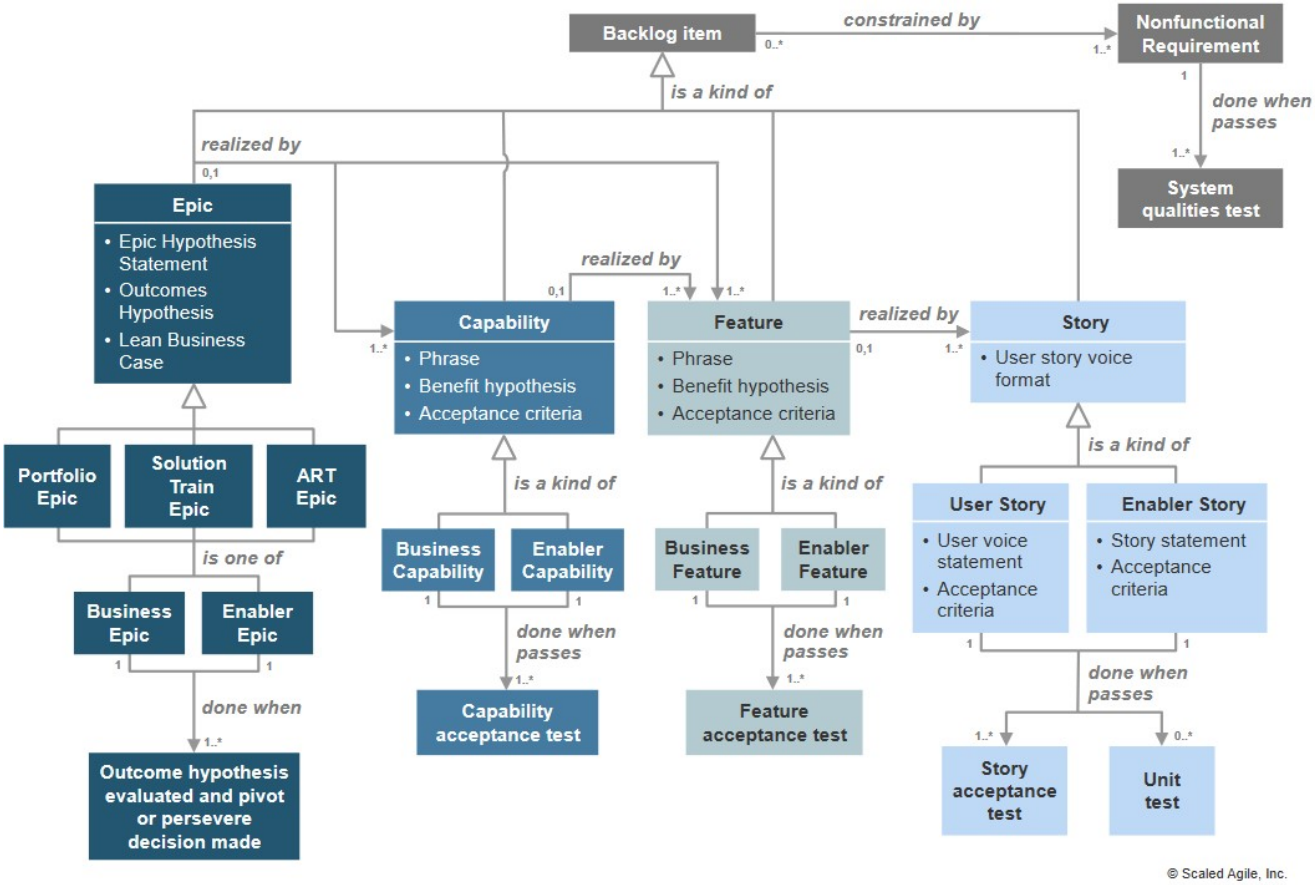
3. **Chanel: "To create, develop, manufacture and sell the best products in the world."**

• Chanel staat voor tijdloze elegantie en kwaliteit, met een missie om de beste mode en beautyproducten te leveren die de tand des tijds doorstaan.

43. SAFe Work Items: het governance model dat je goed in je tooling moet inrichten

Inleiding

Dit hoofdstuk is gebaseerd op **SAFe® versie 6.0.2**. Wat je hier leest is in feite de **praktische implementatie** van wat SAFe® formeel het *SAFe Requirements Model* noemt (zie SAFe® Extended Guidance: SAFe Requirements Model). SAFe beschrijft hierin een schaalbare hiërarchie van Epics, Capabilities, Features, Stories en NFR's om systeemgedrag uit te drukken op elk niveau van de organisatie.



Waarom 'work items' en niet 'requirements'?

Ik gebruik bewust de term **work items** in plaats van *requirements*. De reden is eenvoudig: het woord 'requirements' heeft bij systems engineers en architecten vaak een andere lading; het verwijst daar naar formele systeemeisen, traceability matrices en V-model documentatie. Dat is **niet** wat SAFe bedoelt. SAFe's 'requirements model' gaat over hoe je werk structureert en laat stromen van strategie naar uitvoering. In de Jira/Atlassian-wereld spreekt men van **issue types**, wat dichter bij de praktijk ligt. Om verwarring te voorkomen hanteer ik daarom consequent de term *work items*.

SAFe's Grow the Business (GTB) vs. Run the Business (RTB)

In veel organisaties wordt een fundamenteel onderscheid gemaakt tussen werk dat de business laat groeien (nieuwe producten, nieuwe markten, innovatie) en werk dat de business draaiende houdt: beheer, onderhoud, operationele continuïteit. In de IT-wereld kent men dit ook wel als Dev (development) en Ops (operations). In een van de vroegere SAFe LPM trainingen werden de woorden GTB (Grow the Business) en RTB gebruikt. Met andere woorden, investeringen die toekomstige waarde creëren versus de capaciteit die nodig is om bestaande dienstverlening in stand te houden.

Directe versus indirecte businesswaarde: Business Epics vs. Enablers

Naast het GTB/RTB-onderscheid kent SAFe een tweede belangrijke dimensie: het verschil tussen werk dat direct klantwaarde of businessresultaat oplevert, en werk dat daarvoor de randvoorwaarden creëert. SAFe noemt dit respectievelijk Business work items en Enablers. Een Business Epic of Business Feature levert direct waarde aan eindgebruikers of klanten: een nieuwe functionaliteit, een verbeterd proces, een nieuw product. Een Enabler-Epic of Enabler-Feature levert geen directe businesswaarde, maar is een noodzakelijke investering in de architecturale of technische fundering waarop toekomstige business-waarde gebouwd kan worden (denk aan het opzetten van een CI/CD-pipeline, het migreren naar een nieuw cloudplatform, of het realiseren van een architecturale runway).

From Strategy to Task: de executive management traceability-keten

Een goed ingericht SAFe® Work Item Model, met bovenstaande inachtneming, maakt niet alleen portfolio-sturing mogelijk, maar zorgt ook voor **traceability van strategie naar uitvoering**. In een goed werkend SAFe-systeem kun je namelijk altijd terugvinden hoe een strategische beslissing uiteindelijk resulteert in concreet werk in een team. De keten ziet er conceptueel als volgt uit:

Enterprise Epic → Portfolio Epic → Solution Train Epic → Capability → ART Epic → Feature → Story → Task

In SAFe wordt dit vaak samengevat als de verbinding tussen **strategy, portfolio investment en value delivery**.

Elke stap in deze keten vertegenwoordigt een **verfijning van intentie naar uitvoerbaar werk**. Wanneer deze structuur goed is ingericht in tooling zoals **Jira Align, Jira of Azure DevOps**, ontstaat een krachtige vorm van transparantie.

Executive Management kan bijvoorbeeld rapportages opvragen over:

• hoeveel % van onze capaciteit is besteed aan business work items ten opzichte van enablers.

• hoeveel % van de tijd moesten zaken worden uitgezocht voordat het devteam kon starten met werken? In welke mate beheersen we ons vak en hebben we direct toegang tot de juiste kennis (enabler work items type Exploration)

• hoeveel procent van de totale capaciteit wordt besteed aan improvements ten opzichte van de totale beschikbare capaciteit

• de verhouding tussen 'Grow the Business' en 'Run the Business'

• inzage krijgen in de totale verhouding en naleving van LPM's level Capacity Allocation Guardrail

• etc

Let op: een Jira Epic kan in Jira Align een Feature zijn

De naam *Epic* betekent niet in ieder pakket hetzelfde. In de gebruikelijke koppeling tussen **Jira** en **Jira Align** correspondeert een Jira Epic met een Feature in Jira Align. Dat is een afspraak in de integratie en geen algemene definitie van een SAFe Epic.

Jira heeft standaard een hiërarchie met Epics, daaronder Stories en andere standaardwerkitems, en daaronder Sub-tasks. In **Jira Cloud Premium en Enterprise** kun je met Plans extra hiërarchieniveaus boven Epic toevoegen voor company-managed projecten. Daarmee kun je ook grotere initiatieven en SAFe Epics modelleren. De mogelijkheden hangen af van de producteditie en de gekozen inrichting; Jira Align is hiervoor dus niet verplicht. Zie de Atlassian-documentatie over extra hiërarchieniveaus.

Mijn advies: leg eerst de betekenis en mapping van ieder work item vast. Als je Jira Epics via de connector als Features in Jira Align gebruikt, kan het hernoemen van het Jira-type naar Feature die keuze verduidelijken. Richt je de volledige hiërarchie in Jira zelf in, dan kan een andere naamgeving passend zijn. Stem

typebenamingen, hiërarchieniveaus, rapportages en eventuele koppelingen op elkaar af voordat je de configuratie wijzigt.

Terminologie in de tooling: Jira vs Jira Align

Elk pakket hanteert zijn eigen begrippen voor hetzelfde concept. Dat maakt het extra verwarrend als je niet bewust keuzes maakt. Hieronder een overzicht:

Begrip	SAFe® noemt het	Jira Align noemt het	Jira noemt het	ADO noemt het
Het type werk	Requirement	Work Item	Issue Type	Work Item
Strategisch initiatief	Epic	Epic	Eigen work item boven Epic, bijvoorbeeld SAFe Epic (Jira Cloud Premium/Enterprise met Plans; zelf in te richten)	Epic
Leverbaar binnen PI	Feature	Feature	Epic (standaardnaam; bij deze mapping gebruikt als Feature)	Feature
Teamwerk binnen iteration	Story	Story	Story	User Story
Uitvoerbare taak	Task	Task	(Sub)Task	Task

Veel pakketten bieden de mogelijkheid om de terminologie van de requirements, issue types aan te passen

Denk dit door vóórdat je begint: dit is een enterprise-besluit

Dit kan ik niet genoeg benadrukken: **je work item model is een enterprise-besluit, geen team- of ART-keuze.** Het is niet zo dat Team A met deze set work items werkt en Team B met een andere. Of nog erger: dat ART 1 een ander model hanteert dan ART 2. De hele organisatie (alle portfolios, alle Solution Trains, alle ART's, alle teams) werkt met **dezelfde set work items en dezelfde naamgeving**.

Waarom? Omdat:

• **Jira Align aggregeert over alle ART's en portfolios heen.** Als de ene ART een ander issue type model gebruikt dan de andere, kan Jira Align geen betrouwbare flow metrics, portfolio-overzichten of LPM-rapportages genereren.

• **De bidirectionele sync tussen Jira en Jira Align veronderstelt uniformiteit.** De connector mapt specifieke Jira issue types op specifieke Jira Align work items. Afwijkingen per project of ART breken die mapping.

• **Wijzigen achteraf is complex, tijdrovend en risicovol.** Als je eenmaal live bent en honderden of duizenden items hebt aangemaakt in Align, ADO en/of Jira, is hernoemen of herstructureren een enorme operatie. In Jira betekent dat bulk changes over alle Jira-projecten, het aanpassen van filters, dashboards, automation rules en JQL-queries. In Jira Align moet je de Platform Terminology aanpassen en alle bestaande rapportages en views valideren. Dat wil je niet.

Mijn advies: investeer de tijd om dit model **éénmaal goed door te denken** voordat je je tooling inricht. Betrek hierbij mogelijk Atlassian top consultants, de LACE, je Enterprise Architect, je RTE's, je LPM-team en je Jira-beheerders. Zorg dat iedereen dezelfde taal spreekt, in de vergaderzaal én in de tooling.

Met die context in het achterhoofd, lees je hieronder een voorbeeld work item model waarmee je jouw SAFe work item hiërarchie en governance kunt inrichten. SAFe compliant (-:

Guardrail Routing: sturingsniveau bepaalt het work item type

SAFe® kent een guardrail-gedreven governance. Routing is op basis van **sturingsimpact**, niet op basis van omvang of complexiteit:

Sturingsimpact	Work Item Type
Cross-portfolio	Enterprise Epic
Strategisch portfolio	Portfolio Epic
Multi-ART	Solution Train Epic
Binnen ART	ART Epic
PI delivery	Feature
Iteration work	Story / (Sub)Task

Denk aan deze ontwerpprincipes voor je work item model

• Niet de omvang, maar het **sturingsniveau** bepaalt welk issue type je gebruikt

• Zorg dat de verhouding **business vs enabler** werk meetbaar is op elk niveau

• NFR's zijn **backlog constraints** (quality attributes), geen backlog items

• In mijn praktijkmodel classificeer ik retro-verbeterwerk als **Enabler**: dat houdt het model simpel en planbaar. Retro is hierbij een eigen categorie, geen officieel SAFe Enabler-type.

• Bij routing-conflicten tussen ST en ART heeft **LPM het laatste woord**

Wie stuurt: Wie routeert: Wanneer wordt het geboren?

Wie stuurt	Artefact	Wie beslist waar het landt	Wanneer wordt het type geboren?
Enterprise level	Enterprise Epic	Enterprise Leadership Team	Bij enterprise-brede strategische initiatieven (jaarplan, strategiewijziging, cross-portfolio transformatie)
LPM Portfolio	Portfolio Epic	Het operationeel LPM team	Bij portfolio intake / lean business case fase wanneer investering portfolio-impact heeft. Design thinking.
Solution Train	ST Epic	STE + SM	Wanneer een Epic meerdere ART's raakt of solution-integratie vereist
	Capability	SM	In LPM analysing stage als onderdeel van de Lean Business Case
ART	ART Epic	RTE + PM	Wanneer initiatief groter is dan één feature of meerdere PI's beslaat binnen één ART
	Feature	Product Management	In LPM analysing stage als onderdeel van de Lean Business Case of gedurende ART-level backlog refinement
Team	Story	Product Owner + Team	Ontstaat continu tijdens ART backlog refinement; wordt gecommitteerd tijdens PI- en iteration planning.
	(Sub)Task	Team	Tijdens iteration planning wanneer story wordt opgesplitst in uitvoerbare taken. Frequent gedurende iteratie.

SAFe® Work Item Model: Business, Enabler en NFR-rol

Om bovenstaande verder inzichtelijk te maken heb ik hieronder een **concreet voorbeeld van het SAFe® Work Item Model** uitgewerkt dat iedereen nog verder kan finetunen naar zijn eigen situatie en context. Zowel GTB als RTB zijn hierin gemodelleerd, naargelang aan te passen naar Maintenance, Ops of andere herkenbare woorden.

In dit overzicht is per governance-niveau aangegeven welk type work item wordt gebruikt, hoe Business- en Enabler-werk daarin voorkomen, welke typen Enablers bestaan, en op welke manier NFR's op dat niveau worden geborgd. Dit maakt zichtbaar hoe strategische keuzes uiteindelijk doorwerken tot op teamniveau en zelfs tot individuele taken. SAFe onderscheidt vier typen Enablers: Exploration, Architectural, Infrastructure en Compliance. In mijn praktijkmodel voeg ik Retro en Maintenance toe als eigen categorieën; dit zijn geen officiële SAFe Enabler-typen.

SAFe® governance level	Work Item	Business	Enabler (Architectural Runway)	Enabler subtype	NFR rol
Enterprise	Epic	✓	✓	Architectural / Infrastructure / Compliance /	Enterprise policies

SAFe® governance level	Work Item	Business	Enabler (Architectural Runway)	Enabler subtype	NFR rol
				Exploration	
Portfolio	Epic	✓	✓	Architectural / Infrastructure / Compliance / Exploration / Retro	Portfolio guardrails
Solution Train	ST Epic	✓	✓	Architectural / Infrastructure / Compliance / Exploration	System qualities
	Capability	✓	✓	Architectural / Infrastructure / Compliance / Exploration / Retro	Constraints op features
ART	ART Epic	✓	✓	Architectural / Infrastructure / Compliance / Exploration / Retro	Acceptance constraints
	Feature	✓	✓	Architectural / Infrastructure / Compliance / Exploration / Retro	Acceptance criteria
Team	Story	✓	✓	Architectural / Infrastructure / Compliance / Exploration / Retro / Maintenance	Definition of Done
	(Sub)Task	✓	✓	(niet relevant)	Impliciet via Story

Formele definities

Enterprise Epic
Strategisch initiatief met enterprise-brede impact dat meerdere portfolios of value streams kan raken; realisatie duurt meestal meerdere jaren of portfolio investment cycles.

Portfolio Epic
Groot investeringsinitiatief binnen een portfolio dat significante businesswaarde of strategische enablement levert; realisatie duurt meestal meerdere PI's tot meerdere jaren.

Solution Train Epic
Initiatief met impact op meerdere ART's binnen een Solution Train dat solution-level coördinatie vereist; realisatie duurt doorgaans meerdere PI's.

ART Epic
Groot initiatief binnen één Agile Release Train dat te omvangrijk is voor één Feature en daarom wordt opgesplitst in meerdere Features; realisatie duurt meestal enkele PI's.

Capability
Functionaliteit op Solution-niveau die vaak meerdere ART's omvat en wordt opgesplitst in Features. Een Capability wordt zo gedimensioneerd dat realisatie binnen één Planning Interval (PI) mogelijk is. Is de functionaliteit te groot, splits deze dan in kleinere Capabilities. Zie de SAFe-definitie van Features en

Capabilities.

Feature
Functionaliteit die waarde levert aan de gebruiker of klant en door een ART kan worden gerealiseerd; wordt normaal binnen één Planning Interval al opgeleverd.

Story
Kleine eenheid van functionaliteit die door een Agile Team wordt gerealiseerd; wordt binnen één iteratie afgerond.

(sub) Task
Een Task is een concrete activiteit waarmee het team een Story realiseert. Als praktische richtlijn houd ik Tasks klein, bij voorkeur uitvoerbaar binnen één werkdag. Dit is een eigen werkafspraak, geen vaste SAFe-norm.

Business Work Item
Werk dat direct waarde levert aan gebruikers, klanten of bedrijfsprocessen, zoals nieuwe functionaliteit of verbeteringen aan bestaande producten.

Enabler Work Item
Werk dat nodig is om toekomstige businessfunctionaliteit mogelijk te maken, zoals architectuur, infrastructuur, onderzoek (exploration) of compliance-activiteiten.

Enabler Types
SAFe onderscheidt verschillende typen enablers, waaronder Architectural Enablers, Infrastructure Enablers, Compliance Enablers en Exploration Enablers, die de technische en organisatorische basis leggen voor toekomstige businesswaarde.

Tot slot

Dit model is geen theoretische exercitie, het is een concreet-voorbeeld-blauwdruk waarmee je straks dagelijks werkt. Van de Enterprise Epic die de Enterprise voedt tot de sub task die een team oppakt in een iteratie: elk work item heeft een helder sturingsniveau, een eigenaar en een plek in de keten. Zodanig dat executive management inzage krijgt in de capacity allocation verdeling, conform LPM's guardrails versus de actuals op de werkvloer. Inzage levert interessante stuurinformatie op.

Gebruik dit hoofdstuk als referentie bij het inrichten van je **ADO, Jira- en Jira Align-omgeving**. Bespreek het met je stakeholders, valideer het tegen jullie specifieke context, en leg de keuzes vast voordat je de eerste configuratie aanraakt.

Want zoals eerder gezegd: éénmaal goed inrichten is vele malen goedkoper dan achteraf repareren.

44. Projecten en SAFe Epics: overeenkomsten en verschillen

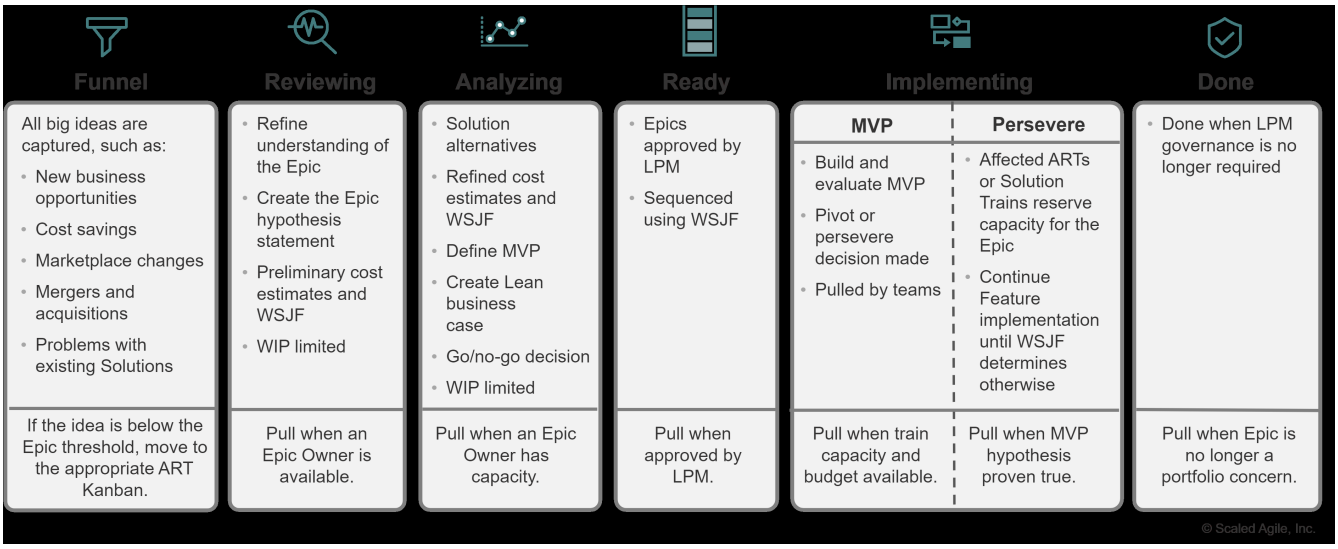
In traditionele watervalorganisaties en methodologieën zoals PPM wordt vaak gedacht in termen van projecten. Daarentegen werken SAFe (Scaled Agile Framework) experts met Epics. Dit fundamentele verschil in aanpak is essentieel om te begrijpen, vooral voor professionals die de overstap maken naar een SAFe-omgeving, zoals die op het niveau van Portfolio Management en Agile Release Trains.

Zonder een eerste lijst van 'gerefinede Epics' kan de LACE de hele SAFe-machine niet opstarten. Deze stap is dus van cruciaal belang en een kritische succesfactor (KSF).

In dit hoofdstuk begin ik met een korte introductie over het managen van Epics in SAFe, gevolgd door een overzichtelijke tabel met de belangrijkste verschillen.

Hoe SAFe Epics beheert ten opzichte van PPM

Alle grote nieuwe initiatieven worden beheerd binnen de SAFe Portfolio Management-laag en bijgehouden op het Portfolio Kanban Board, zoals hieronder weergegeven. Dit board kan worden geïmplementeerd in Jira, Azure DevOps of Fortes.



Alle nieuwe initiatieven worden eerst toegevoegd aan de **Funnel State**, die vaak een uitgebreide lijst van mogelijke epics bevat. Vervolgens worden deze epics verder uitgewerkt in de **Reviewing State**, waar een "Epic Hypothesis Statement" wordt opgesteld. Dit proces lijkt sterk op de aanpak van de PPM Projectkaart.

Daarna verplaatst de Epic Owner de epic naar de **Analyzing State**, waar een Lean Business Case wordt opgesteld. Tijdens deze fase vindt vaak een Feature Storming-sessie plaats (een praktijkwerkvorm, geen officieel SAFe-event), waarbij de epic wordt opgedeeld in verschillende features. Deze features worden vervolgens toegevoegd aan een algemeen Agile Release Train (ART) Kanban-board, omdat op dat moment nog niet duidelijk is welke ART de features zal oppakken.

Na een Go-besluit van Lean Portfolio Management gaat de epic naar de **Ready State**. Zodra er capaciteit is, start de Implementing State: eerst wordt het MVP gebouwd, en op basis van de uitkomst wordt besloten door te gaan (persevere), bij te sturen (pivot) of te stoppen. De Features worden ingepland tijdens de PI Planning van de betrokken ART’s.

Bij SAFe is het niet gebruikelijk om uitgebreide projectvoorstellen of PID's te schrijven. In plaats daarvan wordt alles beknopt samengevat op maximaal vier A4'tjes.

Verschillen en overeenkomsten overzicht

Aspect	Project	Epic (SAFe)
Management	Gecentraliseerd, bevel-en-controle, hiërarchische structuur, stuurgroepen	Gedecentraliseerd, zelfsturend, cross-functioneel, lean aanpak. Geen grote stuurgroepen, maar Reviews, PI events en Inspect & Adapt.
Besluitvorming	Gecentraliseerd: besluiten worden hoog in de lijn of in de stuurgroep genomen, met lange beslislijnen.	Gedecentraliseerd waar mogelijk (SAFe Lean-Agile Principle #9): strategische besluiten centraal in LPM, operationele besluiten zo dicht mogelijk bij het werk, bij de ART's en teams.
Planning	Gedetailleerde projectplannen, vaak met rigide tijdlijnen over lange periodes met deadlines	Lichtgewicht planning, flexibel aangepast op basis van feedback. Planningdetails per PI (doorgaans 10 weken).
Scope	Bepaalde, specifieke doelen met vaste resultaten	Brede, strategische doelen die evolueren op basis van markt- of klantbehoeften. Scope kan per dag worden aangepast conform het tweede principe van het Agile Manifesto
Scope en flexibiliteit	Beperkte flexibiliteit, veranderingen beheerd via formele (trage) wijzigingsverzoeken, impactanalyses, etc	Hoge flexibiliteit, snelle aanpassing op basis van feedback en veranderende eisen.
Uitvoering	Sequentiële, phase gateways benadering	Iteratieve, incrementele levering in lijn met Agile praktijken
Rollen	Projectmanager leidt en controleert het team en beheert 1) de inhoud, 2) de technologie en 3) het proces	Meerdere rollen leveren de Epic: de Epic Owner, de RTE's, de Product Managers, de Architecten. Elke rol kent zijn eigen expertisegebied.
Tijdschema	Bij de overheid kan 1 project jaren duren en heeft het een vooraf gedefinieerde einddatum.	Epic liefst klaar binnen 6 PI's, waar 1 PI doorgaans 10 weken is (SAFe: 8 tot 12 weken). Met een Minimum Viable Product (MVP) toets je de verwachte businessresultaten. Op basis daarvan besluit je door te gaan, bij te sturen of te stoppen. Een Epic wordt afgesloten wanneer is besloten te stoppen of wanneer verdere sturing op portfolioniveau niet meer nodig is. Vervolgwerk kan daarna via de ART-backlogs worden geprioriteerd. Niet alle ooit voorziene Features hoeven daarvoor gerealiseerd te zijn. De verwachte waarde, kosten en alternatieve investeringen bepalen of verder investeren verstandig is.
Van praten naar starten	Kan maanden duren voordat de 1ste paal wordt geslagen.	SAFe experts schakelen en starten snel: Binnen weken, maximaal 1 Planning Interval (doorgaans 10 weken).
Risicomanagement	Via uitgebreide risicologboeken en mitigatieplannen, periodiek besproken in de stuurgroep.	Continu: risico's worden bij elke PI Planning en in de ART Sync zichtbaar gemaakt en geROAMd (Resolved, Owned, Accepted, Mitigated). Het MVP beperkt het investeringsrisico.
Succescriteria	Succes gemeten aan naleving van scope, tijd en budget (GOKIT en TGK)	Succes gemeten aan de geleverde waarde en klanttevredenheid en continue meting van de business outcomes en leading indicators uit het Epic Hypothesis

		Statement.
Financiering	Financiering is meestal vast en vooraf volledig toegewezen.	Incrementele, Lean financiering op basis van waarde, budget kan verschuiven afhankelijk van de geleverde resultaten en MVP.
Stakeholder betrokkenheid	Beperkt tot specifieke fasen; vaak formele bijeenkomsten (bijv. stuurgroepen)	Continue betrokkenheid van stakeholders gedurende het hele proces, met focus op constante feedback via Iteration Reviews, System Demos, PI Planning en Inspect & Adapt.
Teamstructuur	Functiespecifieke teams met gescheiden verantwoordelijkheden	Agile Release Trains met Agile Teams.
Focus op klantwaarde en MVP	Minder directe focus op klantwaarde tijdens projectuitvoering	Hoge focus op constante waardelevering voor de klant. Epics worden zo gedefinieerd dat het MVP zo snel mogelijk live kan, liefst binnen 6 maanden.
Leveringstempo	Volledige levering pas aan het einde van de geplande tijdlijn	Levering vindt plaats in incrementele releases, wat zorgt voor vroege feedback en iteratie
Resourcing	Projectmanagers moeten aanvragen doen voor resources in de lijn en zeggen ‘hoeveel uur ze hoelang nodig hebben’	Epics worden ontwikkeld door bestaande ARTs en bestaande Agile teams. De capaciteit is vooraf via Lean Budgets per waardestroom verdeeld; tijdens PI Planning plannen de ART’s de Features van de Epic in.
Informatie Managers	Projecten ontstaan door input van informatie managers	Epics ontstaan vanuit de strategie (Strategic Themes) en worden in de Portfolio Funnel ingebracht door onder meer Epic Owners, Enterprise Architects, Business Owners en Solution/Product Management. Op ART-niveau wordt het werk verder verkend in de Continuous Exploration-fase van de Continuous Delivery Pipeline.

45. Business of Enabler? Waarom het verschil begrijpen essentieel is

In het kader van SAFe-transformaties is het cruciaal om het onderscheid te begrijpen tussen **Business werk** en **Enabler werk**. Hoewel beide typen werk essentieel zijn om waarde te leveren, verschillen ze sterk in zichtbaarheid, timing en doel. Daarnaast speelt **de Architectural Runway** een sleutelrol bij het verbinden van Enabler werk met de toekomstige levercapaciteit van het systeem. In de praktijk blijkt dat de agilisten het verschil en de relatie met SAFe's Architectural runway heel lastig vinden en vaak door elkaar halen.

Echter, om straks goede performancemetingen te kunnen verrichten, zoals onder andere de ART level "Flow Distribution Metric" is het voor SAFe coaches en voor LPM heel belangrijk direct bij start de juiste Epic- en Feature-typen te duiden. Want hoeveel funding is besteed aan direct aantoonbare klantwaarde en hoeveel aan indirect, onzichtbare waarde?

In dit hoofdstuk leg ik dit verschil uit: helder, concreet en toepasbaar binnen jouw organisatiecontext.

Business vs. Enabler: wat is het verschil?

Kenmerk	**Business Epics / Features / Stories**	**Enabler Epics / Features / Stories**
Doel	Levert directe waarde voor gebruiker of eindklant van de **eindoplossing** :warning:	Ondersteunt toekomstige business value voor de eindklant of maakt technische uitvoering mogelijk
Zichtbaarheid	Ja: herkend door gebruiker, zichtbaar als functionaliteit	Nee: onder de motorkap, maar cruciaal voor werking, prestaties of veiligheid van de eindoplossing. Primair 'voorwerk' etc
Requirement komt van*	Business Owners, Product Managers, eindgebruiker. Degene met het budget of de portemonnee. Deelnemers van de gebruikersorganisatie of proxies.	Architecten, System Team, DevOps, LACE, technische teams. Deelnemers van de 'fabriek'
Waarde-timing	Direct. De klant krijgt het na een iteratie of system demo. Hij is direct blij	Klant ziet niks, beoordeelt ook niks. Heeft hierin vaak geen interesse. Waarde indirect en vaak op middellange of lange termijn zichtbaar. ('we moeten nu weten hoe blockchain werkt!')

**Als een OEM iets nodig heeft om iets te leveren, betekent dit niet dat deze OEM de 'eindklant is' en daarmee een 'Business Epic of Feature'. Een OEM is een koppelvlak, supplier in de keten van waarde levering. Een OEM noch een interne afdeling gaat zelf 'oorlog voeren' en is dus per definitie niet 'de business' en werkt niet met Business Types. Ze leveren 'halffabricaten' in de totale oplossing.*

Voorbeelden binnen Defensie:

• **Business Feature**: "Commandant kan live positiedata van voertuigen zien op dashboard."

• **Enabler Feature**: "Opstellen van documentatieset met details voor inbouw in voertuigen."

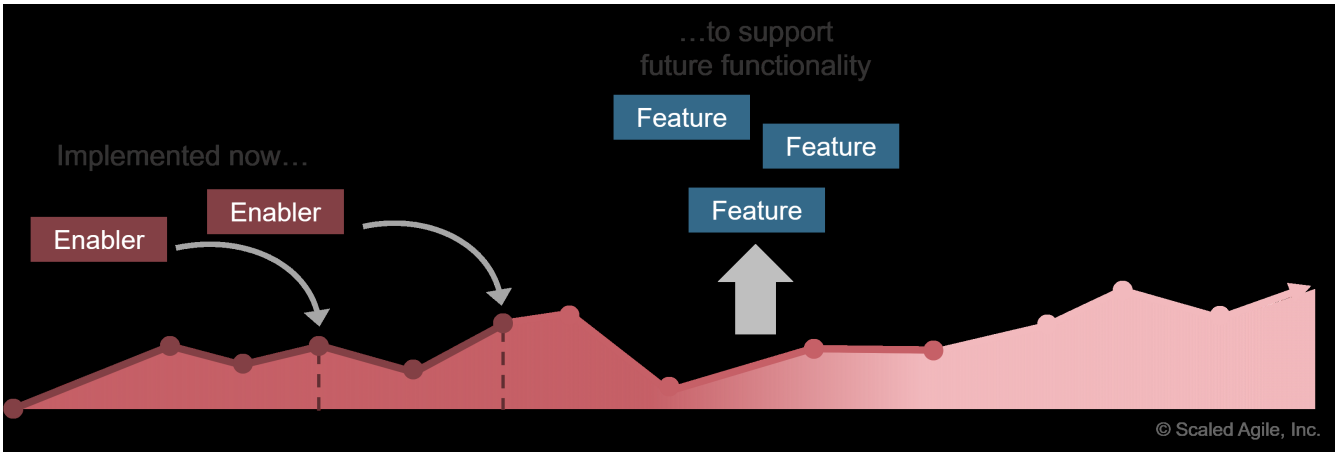
De eerste is functioneel zichtbaar. De tweede is een randvoorwaarde om de eerste überhaupt mogelijk te maken: maar blijft vaak onzichtbaar voor de eindgebruiker.

SAFe kent vier soorten Enablers

SAFe classificeert Enablers in vier types, die elk een specifieke functie vervullen. Ze kunnen op elk niveau van de backlog voorkomen: Epics, Capabilities, Features, Stories.

Type Enabler	Doel	Voorbeeld Defensie
Architectural Enabler	Ondersteunt de architectuur van de oplossing	Opzetten van een referentiearchitectuur. Ontwerpen van deelarchitecturen. Ontwikkelen van glossaries of taxonomieën
Infrastructure Enabler	Bouwt of onderhoudt ontwikkel-, test- of productie-infrastructuur	Virtualisatielaag voor testen met gesimuleerde drones
Exploration Enabler	Onderzoekt onzekerheden, verkent technische of functionele opties	onderzoekswerkzaamheden naar de inzet van AI voor objectherkenning in satellietbeelden. Je weet hier niet wat je niet weet. Inschatten is lastig; werk daarom met een timebox.
Compliance Enabler	Zorgt dat aan regelgeving of interne richtlijnen wordt voldaan	Implementatie van beveiligingslagen volgens MIVD-normen. Uitvoeren van PIA's, black box testing. Audits etc

De relatie met SAFe's Architectural Runway



De **Architectural Runway** is de technische basis die nodig is om toekomstige business functionaliteit snel en betrouwbaar te kunnen leveren. Zie het als het spoor dat moet liggen voordat de trein (business features) kan rijden.

• **Enabler werk** bouwt en onderhoudt dit spoor.

• Zonder runway geen snelheid, geen stabiliteit, en geen schaalbaarheid. De ART PM 'kan dan niet door'

• De ART level **System Architecten (SA's) zijn** verantwoordelijk voor het in kaart brengen, bewaken en uitbreiden van deze runway.

• De RTE bepaalt met LPM's Guardrail Capacity Allocation de verhouding tussen Business Epics en Enabler Epics, Features etc. Dit om te zorgen voor een evenwichtige balans.

Concrete vertaling naar de praktijk

Wanneer we een nieuwe Solution ontwikkelen, kunnen de **zichtbare business features** pas geleverd worden als onderliggende elementen zoals data-integratie, beveiligingslagen, documentatie en netwerkarchitecturen (enablers!) aanwezig zijn. Deze worden gepland, gebouwd en getest via het werk aan de **Architectural Runway**. Pas als we gaan bouwen aan directe klantwaarde, werken we aan Business Type Epics of Features.

Waarom dit relevant is voor jouw organisatie

Het verkeerd typeren van Epics en Features zorgt straks voor onduidelijkheid bij toekomstige lezers, SAFe experts en coaches, en zorgt ervoor dat de Metrics vanaf de start fout zijn.

46. De kernelementen van een Feature: theorie versus praktijk

Dit hoofdstuk is bedoeld voor LPM-ers, Portfolio managers, Product Managers, Release Train Engineers en andere SAFe-practitioners die al bekend zijn met de basisprincipes van SAFe en op zoek zijn naar een praktisch handvat voor het definiëren van kwalitatief goede Features, iets wat tot hun takenpakket hoort hetzij als Content Authority, hetzij als participant.

In het Scaled Agile Framework (SAFe V6.0) is een Feature een *behoorlijk belangrijk type requirement* dat waarde levert aan klanten en belanghebbenden binnen een *specifieke tijdspanne*: zijnde een Planning Interval (PI) van doorgaans 8 tot 12 weken; standaard vier ontwikkeliteraties van twee weken plus één Innovation & Planning (IP) iteratie. Hoewel één Agile team verantwoordelijk kan zijn voor de uitvoering, kunnen Features ook worden gerealiseerd door samenwerking tussen meerdere Agile teams. Een Feature bevindt zich op het niveau van de Agile Release Train (ART) en maakt deel uit van de Essential SAFe-configuratie conform SAFe V6.0.

Features zijn *bij voorkeur* altijd gekoppeld aan een Epic. Die koppeling, inclusief koppeling aan één of meerdere PI's, geeft maximale stuurbaarheid: het maakt duidelijk waar een Feature in het grote geheel aan bijdraagt. 'Zwevende' Features, ook wel *Orphan Features* genoemd, zijn een bekende valkuil in de praktijk. Professionele tooling zoals Jira Align biedt dan ook specifieke functionaliteit om deze orphans op te sporen en te adresseren via de functie "Viewing orphan objects", waarbij een object een Epic, Capability, Feature of Story kan zijn.

De decompositie van een Epic naar Features vindt primair plaats in de Portfolio Kanban, in de staat 'Analyzing' (verfijning) en verder tijdens 'Implementing', bijvoorbeeld in een "Feature Storming"-sessie. Let op: Feature Storming is een praktijkwerkvorm, geen officieel SAFe-event. Epic Owners, Product Management en Enterprise/System Architects werken samen om Epics die de staat 'Reviewing' hebben doorlopen op te splitsen in uitvoerbare en waardevolle onderdelen. Dit proces zorgt voor duidelijke doelen en haalbare uitvoeringsplannen binnen de kaders van SAFe.

Een goed gedefinieerde Feature omvat zowel de theoretische aspecten (zoals beschreven op de SAFe-website en gedoceerd in trainingen zoals de Leading SAFe-training) als aanvullende praktische details die in de praktijk vaak nodig zijn. Deze details ondersteunen de samenwerking tussen teams en bevorderen effectieve besluitvorming.

Theoretische Feature definitie elementen volgens SAFe

1. **Titel**: een korte en duidelijke en 'self-explanatory' naam die de essentie van de Feature weergeeft.

2. **Beschrijving (probleemstelling)**: een korte uitleg van wat de Feature inhoudt en waarom deze waardevol is, het probleem of de wens

3. **Benefits hypothese statement:** wat de Feature oplevert in termen van outcome in plaats van output, met andere woorden wat gaat het opleveren voor de eindgebruikers in eindgebruikers termen.

4. **Acceptatiecriteria**: Duidelijke specificaties die aangeven wanneer een Feature als "klaar" wordt beschouwd. Deze criteria moeten testbaar en meetbaar zijn. BDD is een mogelijk format hiervoor.

✕ ***Don't:*** *Beschrijf een Feature niet uitsluitend in het User Story Voice-format ("Als … wil ik … zodat …"). Een Feature ondersteunt meestal meerdere persona's en functionaliteiten. Bovendien draait een Feature om een outcome voor de business, niet om de behoefte van één gebruiker. De user voice hoort daarom thuis in de onderliggende User Stories, waar elke story zijn eigen persona heeft.*

✓ ***Do:*** *Gebruik het Features-and-Benefits-format: een korte zin die de functie en de context noemt, aangevuld met een meetbare benefit hypothese. Voorbeeld*
: Online zelf afspraken verzetten → 30% minder telefoontjes naar het servicecenter.

Ref: Leading SAFe en POPM training.

Aanvullende praktijk elementen voor effectieve implementatie

1. **Juiste Kanban:** Features bewegen door de ART Kanban.

2. **Feature nummer:** Komt meestal uit de software

3. **Feature state:** De staten van de ART Kanban, conform SAFe: Funnel, Analyzing, Ready (ART Backlog), Implementing, Validating on Staging, Deploying to Production, Releasing en Done. Teams en tooling kunnen dit vereenvoudigen.

4. **Eigenaarschap (Product Management):** In SAFe is Product Management eigenaar van Features en de ART Backlog: zij bewaken de benefit hypothesis, de acceptatiecriteria en de prioritering (WSJF). In de praktijk wijst men vaak per Feature één Product Manager aan als vast aanspreekpunt voor de teams over scope en prioriteit; een aparte rol ‘Feature Owner’ kent SAFe niet.

5. **PI nummer realisatie:** In welke PI deze, geschat, wordt opgeleverd. Planned Features staan primair op het ART Planningsboard. Zo niet, dan is het handig wel een oplevering te forecasten/roadmappen.

6. **Feature type:** Business Feature of Enabler Feature? Ingeval van Enabler, welk van de vier SAFe-typen: Exploration, Architectural, Infrastructure of Compliance. Exploration-enablers worden op storyniveau vaak uitgewerkt als spikes.

7. **Leading ART:** Features worden gedecomponeerd op portfolioniveau, waarbij de Features (vaak) nog niet zijn toegewezen aan een ART. Het is handig om daarvoor tijdens de Feature Storming-sessie al een inschatting van te maken welke ART de leading ART wordt. Dit kan ook tijdens de refinement van de Portfolio Backlog.

8. **WSJF (Weighted Shortest Job First)**: De WSJF score, zijnde CoD ÷ Job Size (Job Duration Proxy). Dit helpt bij het bepalen van welke Feature eerst wordt uitgevoerd.

1. **Feature Estimation:** Bij Feature schattingen in SAFe wordt gebruik gemaakt van Story Points, een relatieve maatstaf om de grootte en complexiteit van het werk in te schatten. Het draait hierbij niet om exacte tijdsduur, maar om de verhouding tussen verschillende Features of user stories. Estimation door degenen die het werk gaan doen of representanten daarvan.

2. **Cost of Delay (CoD):** De economische impact van het uitstellen van een Feature. In SAFe is CoD de som van drie componenten: User-Business Value, Time Criticality en Risk Reduction & Opportunity Enablement. Product Management schat deze relatief (aangepaste Fibonacci-reeks) samen met Business Owners en andere stakeholders.

9. **Alternatieve oplossingen:** Opties of scenario's voor hoe de Feature kan worden gerealiseerd, inclusief een risicoanalyse conform ‘Assume Variability, preserve options’ conform Lean-Agile Principle #3.

10. **Dependencies**: Identificeren van andere Agile Release Trains (ARTs), teams of externe stakeholders die betrokken zijn bij het leveren van de Feature. Binnen of buiten je Solution Train of eigen Portfolio.

11. **Risks:** Mogelijke risico’s, als die er zijn moeten deze direct worden geROAMd (Resolved, Owned, Accepted, Mitigated) en in de software, optie risks, worden verwerkt en gemonitord.

12. **Lijst met User Stories:** Uiteraard is een onderdeel van een Feature een lijst met ‘related user stories’, die samen en in onderlinge samenhang, de Feature opleveren.

Feature related events

1. **Portfolio Kanban**: Features ontstaan als decompositie van Epics in de Portfolio Kanban (staten ‘Analyzing’ en ‘Implementing’).

2. **ART Backlog refinement**: Features worden uitgewerkt en opgesplitst en gepulled.

3. **PI Planning Event**: Features worden gepland en toegewezen aan Agile teams en opgenomen op het ART Planning Board

4. **Iteration Planning**: Teams verfijnen de stories van de geplande Features en nemen ze op in de iteratie.

5. **Team Sync**: Dagelijkse sync van het team; de voortgang van Features wordt hier zichtbaar op het Planning Board.

6. **ART Sync/PO sync**: Voortgang van Features wordt besproken binnen de ART.

7. **System Demo**: Voltooide Features worden gedemonstreerd aan belanghebbenden.

8. **Inspect & Adapt**: een van de I&A onderdelen is de PI System demo, waar ART's hun Done-Features presenteren en feedback ophalen.

Specifieke Feature-metrics

• Flow Velocity (Feature throughput)

• Flow Time (lead time; cycle time als subset)

• Flow Load (WIP)

• Flow Efficiency

• Flow Distribution

• Flow Predictability (business value gepland vs. behaald, niet aantal Features)

• Outcome-metrics: leading indicators uit de benefit hypothesis

Zie SAFe's *Measure and Grow* theorie

Hoe groot is een Feature?

In de praktijk gaat het regelmatig mis met de omvang van Features. Ik zie Agile Release Trains (ART's) waarin bijna iedere kleine functionaliteit als Feature wordt geregistreerd, ook wanneer het slechts om enkele dagen werk gaat en de zelfstandige waarde onduidelijk is. Dat lijkt onschuldig, maar de gevolgen zijn groot. Stel dat een ART acht teams heeft die ieder vijftien mini-Features per Planning Interval (PI) plannen: dan verschijnen er 120 Features op het ART Planning Board. Elk daarvan vraagt aandacht bij refinement, prioritering, planning en voortgangsbewaking. Product Management, de Release Train Engineer (RTE) en de teams besteden daardoor meer tijd aan administratie, terwijl afhankelijkheden en risico's juist minder zichtbaar worden. Bovendien worden flow-metrics lastiger te interpreteren wanneer de omvang van wat men een Feature noemt sterk uiteenloopt.

Het omgekeerde komt eveneens voor: Features die zo omvangrijk zijn dat zij niet binnen één PI kunnen worden opgeleverd. De uitdaging is dus niet om zoveel mogelijk Stories onder één Feature te hangen, maar om herkenbare waarde in een beheersbare omvang te organiseren.

Wat schrijft SAFe voor?

Een Feature is een afgebakende functionaliteit die waarde levert aan gebruikers of stakeholders en door een Agile Release Train (ART) binnen één Planning Interval (PI) kan worden gerealiseerd.

Het aantal onderliggende User Stories is geen formeel SAFe-criterium. SAFe schrijft dus geen minimum of maximum voor het aantal Stories binnen een Feature voor. De belangrijkste bovengrens is dat de Feature binnen één PI gerealiseerd kan worden. Ook een Enabler Feature moet een herkenbaar doel hebben: de waarde kan daar bestaan uit het mogelijk maken van toekomstige functionaliteit, betere architectuur of het voldoen aan noodzakelijke randvoorwaarden.

Mijn praktijkvuistregel: 8 tot 25 User Stories

Om te voorkomen dat het ART Planning Board dichtslibt met te kleine werkitems, gebruik ik in de praktijk **8 tot 25 User Stories per Feature** als een bruikbare richtlijn. Dit is mijn praktische ontwerpkeuze en **geen officiële SAFe-standaard**. Het aantal Stories geeft een eerste signaal, maar is nooit doorslaggevend: een Story kan bij het ene team veel kleiner of groter zijn dan bij het andere.

Aantal Stories	Praktische beoordeling
1 tot 4	Relatief klein. Toets of het geheel zelfstandige waarde oplevert; zo ja, dan kan dit gewoon een Feature zijn. Zo niet, dan past een Story mogelijk beter.
5 tot 7	Onder mijn voorkeursbandbreedte. Controleer de benefit hypothesis, samenhang en gewenste demonstratie. Niet automatisch afkeuren.
8 tot 25	Mijn praktische voorkeursbandbreedte, mits de Feature waardevol, samenhangend en realiseerbaar binnen één PI is.
26 tot 30	Aan de grote kant. Onderzoek of kleinere, afzonderlijk waardevolle Features mogelijk zijn en toets de ART-capaciteit.
Meer dan 30	Duidelijk signaal om naar splitsing te kijken. Dit aantal maakt het werk niet automatisch tot een Capability of Epic; daarvoor is het sturingsniveau bepalend.

Een Feature met vier Stories kan dus uitstekend een volwaardige Feature zijn als deze zelfstandige waarde oplevert. Omgekeerd kan een Feature met twintig Stories te groot zijn wanneer de capaciteit of afhankelijkheden oplevering binnen één PI onmogelijk maken. Uiteindelijk bepalen de functionele samenhang, de verwachte businesswaarde en de beschikbare ART-capaciteit de werkelijke omvang.

Story Points, uren en kosten: houd de begrippen gescheiden

Story Points zijn een relatieve schattingsmaat en geen vaste omrekenfactor naar uren of werkdagen. Een eventueel initieel ijkpunt voor nieuwe teams is bedoeld om schattingen op gang te helpen, niet om later iedere Story Point permanent aan acht arbeidsuren gelijk te stellen. Uit 8 tot 25 Stories volgt daarom **geen algemeen geldige onder- of bovengrens in uren**.

In mijn hoofdstuk “Een rekenmodel voor de kosten van de SAFe fabriek” gebruik ik bewust een ander type voorbeeld: drie medewerkers besteden elk drie werkdagen aan één Story. Dat is negen persoonsdagen, oftewel 72 arbeidsuren per Story bij acht uur per dag. Een Feature van twintig van zulke Stories vertegenwoordigt dan 1.440 arbeidsuren en in dát rekenmodel €90.000 aan kosten. Het is een **illustratief kostenscenario**, niet de norm voor alle Features. Een ander team, product of type werk kan een heel andere gemiddelde Story-omvang hebben.

Wanneer splitsen of opschalen?

Kan een Feature niet binnen één PI worden gerealiseerd, dan onderzoek je of deze in kleinere, zelfstandig waardevolle Features kan worden opgedeeld. Het werk wordt niet automatisch een Epic of Capability omdat het veel Stories bevat of kostbaar is. Zoals beschreven in mijn hoofdstuk over SAFe Work Items is vooral het **sturingsniveau** bepalend: denk aan coördinatie op ART-, Solution Train- of portfolioniveau. Omvang, kosten en doorlooptijd helpen bij de beoordeling, maar zijn niet op zichzelf de formele definitie van het work item type.

De praktische toets blijft daarom eenvoudig: is de beoogde waarde helder, vormt de functionaliteit één logisch geheel, zijn de acceptatiecriteria toetsbaar en kan de verantwoordelijke ART het werk binnen één PI realiseren? Dan heb je een verdedigbare Feature, ongeacht of deze vier, twaalf of twintig Stories bevat.

Conclusie: theorie en praktijk in balans

Een goed gedefinieerde Feature is meer dan een theoretisch sjabloon. De SAFe-richtlijnen bieden een solide basis, maar pas met praktische elementen zoals WSJF, Cost of Delay en alternatieve oplossingen wordt een Feature een echt stuurinstrument voor samenwerking, prioritering én waardeoplevering.

Drie lessen springen eruit. Een Feature is geen User Story: beschrijf de functionaliteit met een korte omschrijving en een meetbare benefit hypothesis, niet uitsluitend in het formaat ‘Als … wil ik … zodat …’. Als praktische richtlijn hanteer ik 8 tot 25 User Stories, maar dat is geen SAFe-norm en er hoort geen vaste urengrens bij. Doorslaggevend zijn zelfstandige waarde, functionele samenhang en realisatie door een ART binnen één PI. Zorg bovendien voor traceability naar de relevante strategische doelen en, waar passend, een bovenliggende Epic; koppel de planning aan de beoogde PI zodat Product Management en Business Owners inzicht houden in de bijdrage aan het grotere geheel.

47. User Story sizing in SAFe: hoe groot mag een User Story zijn?

Binnen het Scaled Agile Framework (SAFe) wordt het gedrag van een systeem beschreven via een hiërarchie van werkitems. Deze artefacten vormen samen het **SAFe Requirements Model**.

De belangrijkste niveaus zijn:

• **Epic**

• **Capability**

• **Feature**

• **Story**

• **Task / Sub-task**

Deze hiërarchie zorgt ervoor dat strategische initiatieven uiteindelijk worden vertaald naar concreet werk voor Agile Teams.

Een **User Story** bevindt zich op het teamniveau en beschrijft een kleine, concrete functionaliteit die waarde levert aan de gebruiker. Volgens SAFe moet een User Story een **kleine verticale slice van functionaliteit** zijn die binnen één iteratie gerealiseerd kan worden.

De ideale grootte van een User Story

Een goede User Story heeft een omvang die door een Agile Team **binnen één iteratie volledig afgerond kan worden**, inclusief ontwikkeling, testen en validatie volgens de Definition of Done (DoD).

In de praktijk betekent dit meestal:

• **doorlooptijd:** enkele uren tot maximaal enkele dagen werk

• **timebox:** altijd afgerond vóór het einde van de iteratie

• **scope:** één kleine functionaliteit of gedragsverandering

Wanneer een Story te groot wordt, moet deze worden **gesplitst in meerdere kleinere Stories**.

Waarom kleine User Stories belangrijk zijn

Een belangrijk Lean-principe binnen SAFe is **Small Batch Sizes**. Kleinere werkpakketten bewegen sneller door het ontwikkelproces en verbeteren de flow van waarde.

Kleine stories zorgen voor:

• snellere feedback van gebruikers

• kortere doorlooptijden

• minder complexiteit

• betere voorspelbaarheid

• snellere detectie van fouten

Wanneer werk in kleinere batches wordt ontwikkeld, kunnen teams sneller leren en sneller bijsturen.

Problemen met te grote User Stories

Te grote Stories veroorzaken vrijwel altijd problemen in het ontwikkelproces:

1. **Delivery risk**

Wanneer een Story niet binnen de iteratie afkomt moet deze opnieuw worden opgesplitst en herpland.

1. **Langzamere waardelevering**

Waarde bereikt de klant later omdat grote stukken werk langer onderweg zijn.

1. **Slechtere voortgangsmeting**

Het wordt moeilijker te bepalen hoeveel werk werkelijk is afgerond.

1. **Hogere complexiteit**

Grotere Stories bevatten vaak meerdere functionaliteiten of afhankelijkheden.

1. **Minder feedbackmomenten**

Feedback van stakeholders komt pas laat in het proces.

1. **Minder flexibiliteit**

Grote Stories maken het moeilijker om prioriteiten te veranderen.

Wanneer een Story te groot of te klein is

Binnen SAFe gelden een paar eenvoudige richtlijnen:

Situatie	Actie
Story te groot	splitsen in meerdere Stories
Story te groot voor teamniveau	promoveren naar **Feature**
Story te klein	verwerken als **Task of Sub-task**

Tasks of Sub-tasks beschrijven vaak het **technische werk binnen een Story**, zoals testen, configureren of documenteren.

Story Points en grootte

Story Points worden gebruikt om **relatieve complexiteit** te schatten. Ze hebben **geen vaste relatie met uren of dagen**. Teams bepalen zelf wat hun referentie-story is en schatten relatief ten opzichte daarvan.

Dat betekent dat:

• 1 Story Point niet automatisch 1 uur of 1 dag betekent

• snelheid (velocity) bepaalt hoeveel werk een team werkelijk kan doen

Conclusie

User Stories zijn het belangrijkste werkitem voor Agile Teams binnen SAFe. Hun kracht zit in **kleine, duidelijke en snel leverbare functionaliteit**. Door Stories klein te houden ontstaat een snellere flow van waarde, betere voorspelbaarheid en frequentere feedback.

De vuistregel blijft daarom: **maak Stories niet te groot, zo klein mogelijk maar ook niet te klein**

48. SAFe-risico’s en impediments: wat is het verschil?

In dit hoofdstuk wil ik graag een veelvoorkomend misverstand rond *risico’s (risks)* en *belemmeringen* (*impediments*) verduidelijken. Dit is vooral relevant voor degenen die werken met scaled agile en SAFe.

De verwarring: waarom worden risico’s en belemmeringen vaak door elkaar gehaald?

Het komt vaak voor dat agile teams moeite hebben om het onderscheid tussen risico’s en belemmeringen te definiëren. Een mogelijke reden hiervoor is dat een onbehandelde belemmering uiteindelijk kan uitgroeien tot een risico. Stel dat een bepaalde technische belemmering niet tijdig wordt opgelost, dan kan dit de oplevering van een Feature in gevaar brengen en daarmee het succes van een gehele ART-release riskeren.

Risks vs. Impediments: wat is het verschil?

In SAFe onderscheiden we risico’s en belemmeringen als twee verschillende soorten artefacten. Hoewel ze gerelateerd kunnen zijn, is het belangrijk om hun verschillen goed te begrijpen: 1) om je velocity te maximaliseren, 2) om effectieve besluitvorming te ondersteunen, 3) om effectieve probleemoplossing te bevorderen en 4) om de efficiëntie van PI Planning te verhogen.

Risico’s: Dit zijn potentiële *Impediments* die in de *toekomst* kunnen optreden en die invloed kunnen hebben op het succes van een Epic of Planning Interval. Risico’s kunnen optreden op zowel ART-niveau als teamniveau. Tijdens PI Planning identificeren teams in de team-breakouts de risico’s voor hun eigen plan en bespreken zij welke maatregelen zij zelf kunnen nemen. Risico’s die teamoverstijgend zijn of besluitvorming buiten het team vragen, worden op ART-niveau samengebracht en met ROAM behandeld. Niet ieder teamrisico hoeft dus plenair te worden besproken, maar teamrisico’s maken wel deel uit van PI Planning.

Impediments: Dit zijn huidige problemen die de voortgang van een team belemmeren. Een belemmering is iets dat het team *nu (!!)* tegenhoudt. Dit kan iets zijn als een tekort aan middelen, iemand die nu ziek is geworden of een technische uitdaging die opgelost moet worden. Ook deze impediments kunnen op teamniveau zijn of teamoverstijgend.

Formuleringstips

Risico’s

• Formuleringstip: Ik observeer *<het risk event>* en ik zie een *<% kans>* dat *<potentiële gevolgschade door optreden risk event>*. **Als** *<het risk event>* plaatsvindt, dan heeft dit *<ingrijpende/grote/medium/beperkte>* gevolgen voor *<ST, ART of team level>*.

• Voorbeeld: Ik las recent in de krant dat *<de WHO waarschuwt voor een uitbraak van apenpokken>* en ik zie een *<5%>* kans dat daarom *<alle teamleden ziek worden>*. **Als apenpokken bij ons uitbreken**, dan heeft dit *<ingrijpende>* gevolgen voor *<alle Solution Trains, ART’s en alle agile teams>*.

Impediments

1. Formuleringstip: <wie heeft het probleem> heeft op dit moment *<het probleem>* wat *<de gevolgen>*. *<De oplossing>* moet voor *<datum/tijd>* zijn geregeld.

1. Voorbeeld: Ons Agile Team *<teamnaam>* heeft op dit moment geen toegang tot de testomgeving, wat de voortgang van de tests blokkeert. Toegang moet voor <datum/tijd> zijn geregeld.

Samenvatting

Risks

1. Beoordeel risico’s zodra je ze identificeert en bepaal welke opvolging nodig is. Gebruik ROAM: Resolved (het risico is afgehandeld en vormt geen zorg meer), Owned (iemand neemt verantwoordelijkheid voor de verdere beoordeling en opvolging), Accepted (het risico is bewust geaccepteerd) of Mitigated (er is een plan om de kans

of impact te beperken). Leg de gekozen aanpak, het eigenaarschap en eventuele maatregelen vast en volg deze actief op. Je hoeft dus niet te wachten tot een risico optreedt; ook hoeft niet ieder risico te worden gemitigeerd.

2. Mogelijke indicatoren: het aantal risico's zonder eigenaar of afgesproken aanpak, de tijd tot een besluit over de risicoaanpak en de resterende risico-impact na maatregelen. Beoordeel geaccepteerde risico's apart en bewaak of mitigerende maatregelen volgens afspraak worden uitgevoerd.

Impediments

1. Je kunt per direct schakelen en starten met oplossen.

2. KPI's: Impediment Resolutietijd / #Aantal Onopgeloste Belemmeringen

Tip: maak taalafspraken

Houd het liefst 2 woorden aan bijvoorbeeld 1) Risks en 2) Impediments. Bij voorkeur woorden die aansluiten bij de Agile Software menu opties en de taalinstellingen.

Alternatieven zijn

• Risico's en

• Belemmeringen / knelpunten

Deel 7: LACE, governance en enterprise

49. Het LACE in SAFe: van bottom-up implementaties naar enterprise agile governance

Introductie

Steeds meer grote organisaties in Nederland starten met de implementatie van het Scaled Agile Framework (SAFe). Vaak begint zo'n transformatie niet centraal, maar juist lokaal: een afdeling, business unit of IT-organisatie besluit zelf om met SAFe te starten. Vanuit die eerste implementaties ontstaan vervolgens Agile Release Trains, rollen zoals Product Owners en Scrum Masters, en wordt een eerste **Lean-Agile Center of Excellence (LACE)** ingericht om de transformatie te ondersteunen.

In de praktijk leidt dit regelmatig tot een interessante organisatorische vraag:
Moet een SAFe-transformatie bottom-up groeien vanuit afdelingen, of juist top-down worden gestuurd vanuit de organisatieleiding?

Het antwoord op die vraag bepaalt in grote mate **hoe het LACE wordt ingericht, gepositioneerd en gebruikt binnen de organisatie**. Het LACE kan namelijk functioneren als een klein ondersteunend expertteam voor één afdeling, maar ook als een centraal transformatie-orgaan dat de Agile ontwikkeling van een complete organisatie of zelfs een gehele enterprise aanstuurt.

In dit hoofdstuk verkennen we daarom drie belangrijke onderwerpen:

• Wat een **Lean-Agile Center of Excellence (LACE)** precies is volgens SAFe 6.0

• Welke **rol het LACE speelt in een Agile transformatie**

• Wat de verschillen zijn tussen **bottom-up en top-down implementaties** en welke LACE-modellen daarbij passen

Door deze perspectieven te combineren ontstaat een beter beeld van **hoe organisaties hun SAFe-transformatie structureel en duurzaam kunnen organiseren**.

Wat is een Lean-Agile Center of Excellence (LACE)?

Een Lean-Agile Center of Excellence (LACE) ondersteunt de SAFe-transformatie. De passende inrichting (centraal, decentraal of hub-and-spoke) hangt af van de schaal en complexiteit van de organisatie. Het LACE lijkt op het in de SAP-wereld bekende 'SAP competence centre'.

In theorie fungeert het LACE als een drijvende kracht achter de Agile transformatie en ondersteunt het divisies, afdelingen en secties bij de adoptie van Lean-Agile-DevOps principes op zeer grote schaal. Het LACE zorgt ervoor dat de benodigde ondersteuning, middelen en kennis beschikbaar zijn om SAFe succesvol te implementeren.

SAFe 6.0 definieert een LACE als een groep van cross-functioneel samenwerkende, ervaren en gecertificeerde SAFe-experts en transformatieconsultants (SPC's en SPCT's), die verantwoordelijk zijn voor de voortdurende ondersteuning en groei van de Lean-Agile transformatie binnen de organisatie. Het is een multidisciplinair team (hard en zacht) dat nauw samenwerkt met managementlagen om consistentie in de transformatieaanpak te waarborgen en de Executive Metrics (OKR's) en Strategische Thema's te realiseren en waarborgen.

Verantwoordelijkheden van een LACE

Een LACE heeft verschillende verantwoordelijkheden volgens SAFe 6.0, waaronder:

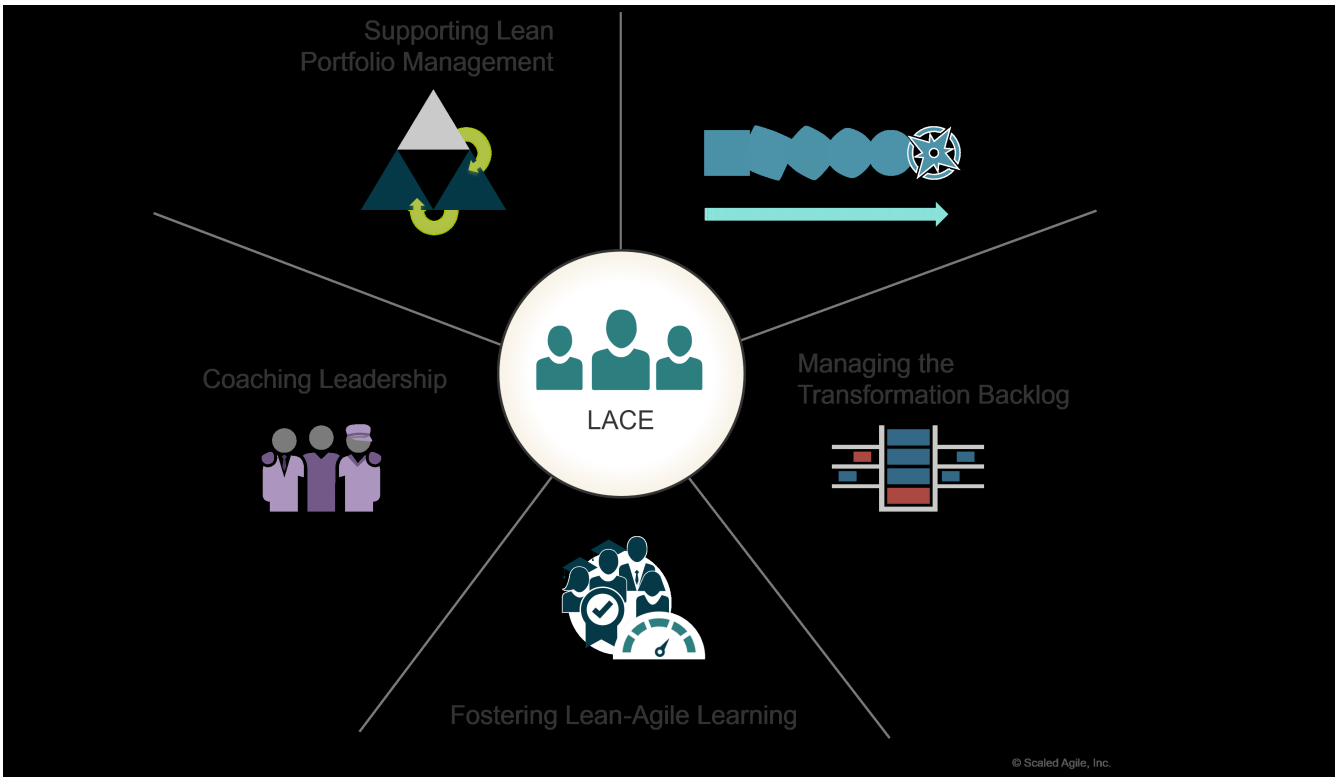
1. **Transformatiestrategie ontwikkelen**: het LACE helpt de richting en visie van de transformatie uit te stippelen, afgestemd op de bedrijfsdoelstellingen.

2. **Coaching en training bieden**: het team faciliteert trainingen en biedt ondersteuning aan teams en management om hen in staat te stellen Agile principes te omarmen.

3. **Continu verbeteren**: het LACE monitort de voortgang van de transformatie en identificeert verbeterpunten.

4. **Organisatiebrede communicatie**: het LACE zorgt voor transparantie en samenwerking door communicatiekanalen op te zetten die de gehele organisatie bereiken.

5. **Ondersteunen van LPM, ART's en alle Agile Teams via de BAT's of via de LACE-Spokes**: ze helpen Agile Release Trains (ART's) en teams om hun Lean-Agile werkwijzen te verbeteren en obstakels te overwinnen.



LACE versus bottom-up & top-down implementatie

Bij het implementeren van SAFe kunnen organisaties kiezen voor een bottom-up of een top-down aanpak. Hieronder staat een niet-limitatieve opsomming met voordelen en nadelen:

1. **Top-down implementatie**:

• **Voordelen**: Deze aanpak zorgt voor duidelijke sturing en visie vanuit het management, wat helpt bij het creëren van alignment tussen verschillende lagen van de organisatie. Het kan bredere, strategische veranderingen ondersteunen en waarborgen dat de hele organisatie dezelfde richting volgt. Vaak vinden medewerkers deze aanpak prettiger want 'executive management wil dat wij dit doen'. Een ander voordeel is dat een organisatie er baat bij heeft de top SAFe experts bij elkaar in een Centrale LACE te zetten om zo krachtenbundeling te stimuleren en kosten te besparen. De kans op falen is daardoor mogelijk minder bij deze aanpak omdat er 1 centrale SAFe denktank is met the-best-of-the-best SAFe experts met proven track records. Top-Down implementatie bevordert voorts SAFe's Customer Centricity omdat er vanuit 1 Target Operating Model wordt gemodelleerd.

• **Nadelen**: Dit kan leiden tot weerstand op teamniveau, vooral als teams zich beperkt voelen in hun autonomie of als ze niet voldoende betrokken worden bij het veranderproces, immers de afstand tot het centrale LACE wordt groter (wij/zij denken). Ook een 'one size fits all' aanpak kan gevaarlijk zijn, mogelijk moeten er decentrale

aanpassingen worden toegelaten. Door benodigde afstemming en extra overleggen, kan deze aanpak trager zijn (eindeloze discussies tussen Centrale LACE en alle Spokes).

2. **Bottom-up implementatie**: Elke business unit of organisatiedomein voert zijn eigen decentrale SAFe implementatie uit.

1. **Voordelen**: Deze aanpak stimuleert teamniveau-innovatie en creëert *grassroots-bewegingen*, waarbij teams zelf Agile werkwijzen omarmen. Dit leidt vaak tot snellere buy-in en eigenaarschap op het operationele niveau. Prima, mits het uitgangspunt is dat alle IT afdelingen solitair en onafhankelijk van elkaar opereren en kunnen opereren.

2. **Nadelen**: De transformatie kan fragmentarisch aanvoelen en niet consistent ('*inconsistent practices*') door de gehele organisatie worden uitgerold, wat leidt tot een gebrek aan alignment en sturing vanuit het hogere management. Na verloop van tijd kan opschalen naar enterprise niveau lastig blijken als cadences, PI events, Requirements models, taal, tooling en uitgangspunten overal op een andere wijze zijn uitgevoerd en er geen eenduidige naamconventies voor bijvoorbeeld een Epic zijn. SAFe kennis is gedecentraliseerd met gemis van synergetisch effect. Implementatie wordt relatief duurder zonder het vierogenprincipe door SAFe super-experts met dito ‘countervailing power’. Kans op misalignment met enterprise OKR’s.

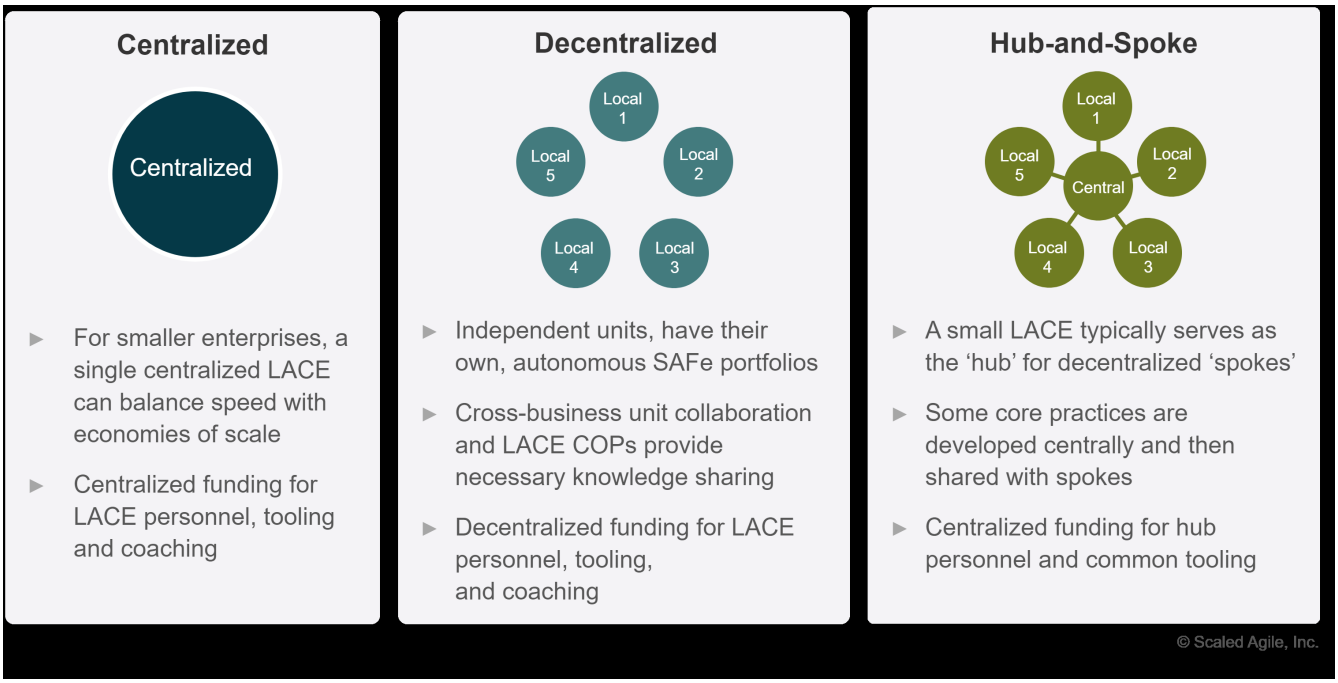
Verschillende LACE modellen in te zetten in verschillende situaties

Organisaties kunnen kiezen uit verschillende LACE-modellen, afhankelijk van hun behoeften en de complexiteit van de transformatie:

1. **Gecentraliseerd LACE-model**: Een centrale LACE die de gehele organisatie/afdeling/business unit ondersteunt. Dit model is efficiënt voor kleine tot middelgrote organisaties of als de Agile transformatie zich in een vroege fase bevindt.

2. **Gedecentraliseerd LACE-model**: Hier hebben verschillende delen van de organisatie hun eigen LACE's, wat autonomie en flexibiliteit bevordert. Dit model werkt goed in organisaties met diverse domeinen binnen de enterprise. Maar er is geen synergetisch effect en alle implementaties kunnen anders zijn. Later schalen kan zeer complex worden.

3. **Hub-and-Spoke model**: Een hybride model waarbij er een centrale LACE (de "Hub") op top niveau bestaat, die wordt ondersteund door decentrale LACE’s binnen afzonderlijke afdelingen of divisies (de "Spokes"). Dit model combineert de voordelen van centralisatie (kosten, bundeling van expertise, makkelijker schalen) met flexibiliteit voor individuele onderdelen. De kans op falen is hoogstwaarschijnlijk kleiner.



Waarom het Hub-and-Spoke model bij top-down implementaties de voorkeur heeft

Het LACE *Hub-and-Spoke model* is vaak het meest geschikt voor een top-down implementatie, vooral binnen grote organisaties. Dit model zorgt voor centrale sturing en consistente uitrol van SAFe-principes, terwijl het tegelijkertijd flexibiliteit biedt voor afzonderlijke onderdelen om de specifieke behoeften en uitdagingen van hun teams en divisies aan te pakken. Het helpt ook om strategische alignment te creëren, terwijl teams op lagere niveaus zich niet beperkt voelen door een rigide, gecentraliseerd systeem.

De centrale LACE bepaalt met alle SAFe experts de speelruimte en kaders voor de decentrale LACE's.

Executive Management ontvangt alle metrics van alle SAFe implementaties via het Centrale LACE, waar ook kan worden gecontroleerd en gemonitord welke SAFe implementaties goed lopen en welke niet goed.

Zo wordt de top-down visie van het management effectief gecombineerd met een pragmatische, gedecentraliseerde uitvoering.

50. SAFe transformaties met het SAFe LACE Hub-and-Spoke model

Inleiding

In organisaties als ASML, Defensie, ABN AMRO en NS, waar complexe systemen en multidisciplinaire samenwerking essentieel zijn, is het succesvol opschalen van Lean-Agile principes cruciaal voor duurzame waardecreatie. De implementatie van SAFe (Scaled Agile Framework) biedt een gestructureerde aanpak om deze principes op grote schaal toe te passen. Toch brengt dit uitdagingen met zich mee, zoals het verkrijgen van consistente metrics. Het ontbreken van een eenduidig inzicht in prestaties belemmert strategische besluitvorming, maakt prioritering van initiatieven moeilijker en leidt tot inefficiënte resourceallocatie.

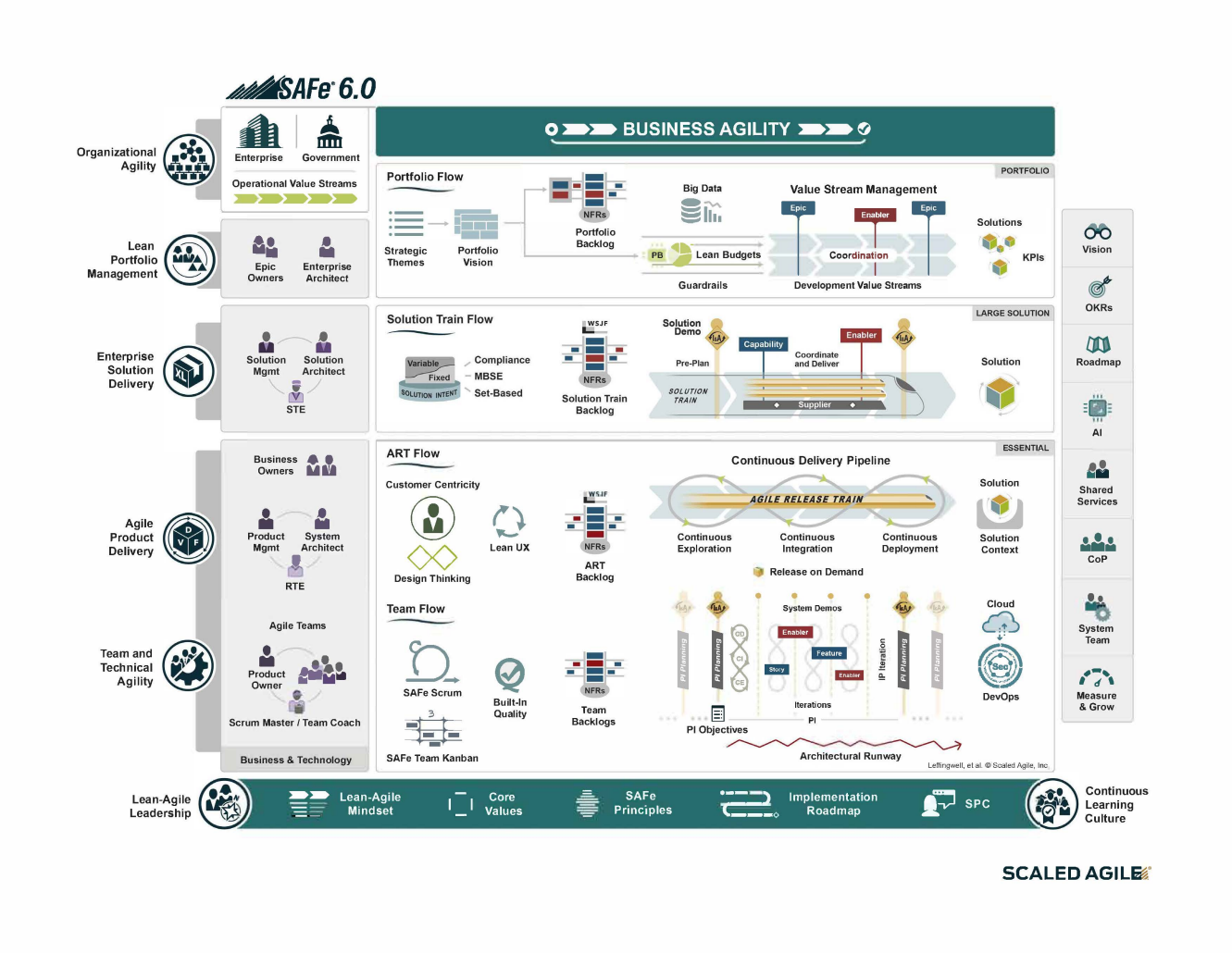
Sommige business units hebben al belangrijke stappen gezet in hun SAFe-reis door een SAFe-compliant Lean-Agile Center of Excellence (LACE) op te richten. Dit strategische centrum fungeert als drijvende kracht achter de transformatie en biedt ondersteuning aan de organisatie bij het omarmen van Lean-Agile werken.

In dit hoofdstuk bespreek ik hoe organisaties met behulp van het LACE Hub-and-Spoke-model een succesvolle transformatie op enterpriseniveau kunnen realiseren. Daarbij focus ik op de rol van de centrale Enterprise LACE (E-LACE) en hoe deze consistentie waarborgt, terwijl de afdelingen via Operational LACEs (O-LACEs) voldoende autonomie behouden om lokale uitdagingen aan te pakken. Deze aanpak stelt organisaties in staat om effectief te transformeren en duurzame verandering te realiseren in een complexe organisatieomgeving.

Wat is SAFe?

SAFe, of het Scaled Agile Framework, is een model en filosofie voor het opschalen van Lean-Agile best practices naar grote, complexe organisaties. Het biedt een gestructureerde aanpak voor het coördineren van meerdere Agile teams, Agile Release Trains en portfolio’s, terwijl het focus houdt op waardecreatie, klantgerichtheid en continue verbetering. SAFe stelt organisaties in staat om flexibel te blijven, zelfs op grote schaal.

Voor meer informatie over SAFe, zie de officiële SAFe-website.



Het SAFe 6.0 Framework

Wat is een LACE?

Een LACE, of Lean-Agile Center of Excellence, is een gespecialiseerd team dat verantwoordelijk is voor het leiden en ondersteunen van de Lean-Agile transformatie binnen een organisatie. Een LACE zorgt voor eenduidige trainingen, toolingstandaarden en best practices, wat leidt tot lagere kosten, hogere productie-efficiëntie en minder transformatiefouten.

Het kan opereren op strategisch, tactisch en operationeel niveau afhankelijk van de transformatie-inrichting. In essentie is een LACE een strategisch instrument voor het implementeren van SAFe en het ondersteunen van een cultuurverandering.

Wat is het Hub-and-Spoke-model?

Het Hub-and-Spoke-model is een organisatorisch framework afkomstig uit de logistiek, transport en informatienetwerken waarbij een centrale hub functioneert als coördinerend centrum en meerdere gedecentraliseerde spokes autonomie hebben binnen hun specifieke context. Dit model biedt meerdere voordelen:

• **Consistentie:** de centrale hub waarborgt uniforme richtlijnen en standaarden.

• **Autonomie:** de spokes hebben vrijheid om specifieke oplossingen te implementeren binnen de kaders van de Hub.

• **Snelle feedbackcycli:** Lokale teams kunnen ervaringen terugkoppelen naar de hub.

• **Efficiëntie:** Middelen zoals rolbeschrijvingen, cadence, training en TOM ontwerp worden centraal beheerd, waardoor duplicatie wordt voorkomen.

De decentrale aard van operaties vraagt om een model dat autonomie faciliteert zonder het grotere geheel uit het oog te verliezen. En dat kan met het Hub-and-Spoke-model.

Hub-and-Spoke-model binnen organisaties: Enterprise-LACE en Operational-LACE

Om het onderscheid tussen de centrale en decentrale LACEs duidelijk te maken, gebruiken we de termen **E-LACE** (Enterprise LACE) en **O-LACE** (Operational LACE). Er komt dus één centrale Enterprise LACE met daaronder meerdere spokes: de operational LACEs per afdeling.

Eisen en rollen binnen een LACE

Een LACE is een gespecialiseerd Agile team en volgt daarom alle regels die gelden voor SAFe compliant Agile teams:

• **Klein & zelforganiserend:** Compact, cross-functioneel, self-managing en zelfsturend. Conform SAFe: 5 tot maximaal 10 personen in omvang.

• **Iteratief werken:** LACEs werken iteratief en focussen op continue verbetering.

• **T-shaped vaardigheden:** Teamleden hebben brede kennis en diepe expertise, wat verzekert dat teamleden flexibel kunnen inspringen waar nodig.

• **SAFe level:** Opereert op LPM niveau.

Kernrollen:

• **Product Owner (PO):** SAFe Content Authority. Weet wat er moet worden gedaan. Beheert de backlog, schrijft de PBI’s en prioriteert werkzaamheden conform WSJF.

• **Scrum Master (SM):** Faciliteert het team (zalen, middelen, meetings enzovoort), verwijdert obstakels en zorgt voor high-performance teamwork.

• **Development Team Members:** Voeren de backlog items uit, refinen en produceren deliverables direct of indirect.

Ideale LACE-deelnemers

E-LACE:

Een cross-functioneel team met:

• Een mix van harde en zachte vaardigheden.

• 1 tot 3 senior SAFe-experts (liefst SPC’s aangevuld met 1 SPCT)

• Organisatie-experts.

• Afgezanten van de O-LACEs, primair hun PO’s of gemandateerde afgevaardigden.

• Senior leiders met ervaring in Agile transformaties. Key decision-makers.

O-LACE:

• Een cross-functioneel team waarvan de samenstelling specifiek wordt bepaald door de behoeften van de afdeling.

• Dit wordt afgestemd door de PO en SM van de O-LACE.

• Verder zie boven, maar minder ‘zwaar’

Overzicht verantwoordelijkheden

E-LACE:

• Content Authority van de totale Enterprise transformatie backlog

• Ontwerpt en beheert het organisatie-specifieke Manifesto.

• Ontwikkelt en deelt kennis over SAFe standaarden en best-practices.

• Biedt coaching en dagelijkse ondersteuning aan de O-LACEs.

• Ontwerpt en beheert al het trainingsmateriaal, zowel over SAFe als over de change.

• Ontwerpt metrics en dashboards voor Enterprise prestatiemonitoring.

• Voert audits uit op de O-LACEs, analyseert en ondersteunt verbeteringen.

• Ontwerpt en beheert de Agile Toolstack.

• Faciliteert enterprise en O-LACE specifieke VSAIW sessies (Value Stream and ART Identification Workshops).

• Bewaakt en monitort het SAFe Enterprise Target Operating Model in samenwerking met alle O-LACE’s.

• Beheert de SAFe Enterprise Knowledge Base met de totale SAFe opzet binnen organisaties: Cadence, toolingkeuze, rollen, TOM, Tipping Point, Metrics, enzovoort.

• Faciliteert de diverse CoP per vakgebied

• Monitort afwijkingen ten opzichte van de E-LACE ontwerpen en guidelines alsmede de impact bij afwijken daarvan

O-LACE:

• Implementeert SAFe binnen de afdeling.

• Faciliteert lokale adoptie van Lean-Agile principes.

• Beheert de lokale transformatie backlog.

• Rapporteert metrics aan de E-LACE.

• Coacht lokale rollen zoals PO’s, SM’s, RTE’s enzovoort.

• Implementeert veranderingen in samenwerking met lokale teams.

Voorbeelden van standaarden te ontwikkelen door de E-LACE

1. Het centrale enterprise dashboard met enterprise brede metrics

2. Het organisatiebrede SAFe TOM

3. Standaardisatie in enterprise tooling en vooral ook de opzet ervan

4. Standaardisatie in Cadence, zelfde heartbeat

5. Afstemming tussen alle lopende PI's (shifted PI's gedurende IP iteratie)

6. Standaardisatie in rollen en vooral rolbenamingen (met bepaalde lokale vrijheidsgraden)

7. Standaardisatie in Epic namen met naamconventie

8. Standaardisatie LPM Guardrails

9. Standaardisatie in SAFe hulpmiddelen zoals Capacity and Velocity calculators

10. Standaardisatie in de formele events, de opzet, de benamingen. Zoals bijvoorbeeld opzet PI event en inspect & adapt (ART's krijgen daarbinnen bepaalde vrijheidsgraden)

11. Standaardisatie van audits op team- en ART-niveau, scans, enzovoort

12. Standaardisatie en vooral consensus van het formele SAFe Requirements model (wat wel en wat niet) inclusief integratie hiervan in de Agile tooling.

13. enzovoort

Werkvormkeuze binnen Essential SAFe

Het LACE kan kiezen tussen Kanban en Scrum, afhankelijk van de context:

• **Scrum:** Geschikt voor iteratief en incrementeel werk.

• **Kanban:** Ideaal voor continue workflows met variabele prioriteiten.

Belangrijk is dat de E-LACE een 1:N backlogrelatie onderhoudt met de O-LACEs.

Hoeveel tijd kost het lidmaatschap van een LACE?

LACE-leden investeren tijd afhankelijk van hun rol. Indicatief:

• **Recurring meetings:** 4 uur per maand.

• **e-mails en stukken lezen en goedkeuren:** 4 tot 8 uur per maand, afhankelijk van SAFe-kennis en routine

• **SAFe training inclusief certificering:** 10 dagen per jaar.

• **Artifact-creatie:** Afhankelijk van de rol en context en mate van routine in SAFe.

• **Coaching, support, training:** Afhankelijk van je rol, context en mate van routine in SAFe.

Samengevat

Het LACE Hub-and-Spoke-model biedt organisaties een robuust fundament om een Lean-Agile transformatie op enterpriseniveau effectief te realiseren. Door de centrale rol van de Enterprise LACE (E-LACE) te combineren met de autonomie van de Operational LACEs (O-LACEs), wordt consistentie in processen en standaarden gewaarborgd, terwijl afdelingen flexibel kunnen inspelen op specifieke uitdagingen.

Deze balans tussen centralisatie en decentralisatie is cruciaal voor complexe organisaties, waar samenwerking tussen diverse onderdelen van groot belang is. Met duidelijke richtlijnen, effectieve samenwerking en een focus op waardecreatie, kan een organisatie niet alleen haar Agile transformatie succesvol leiden, maar deze ook duurzaam verankeren in haar organisatiecultuur.

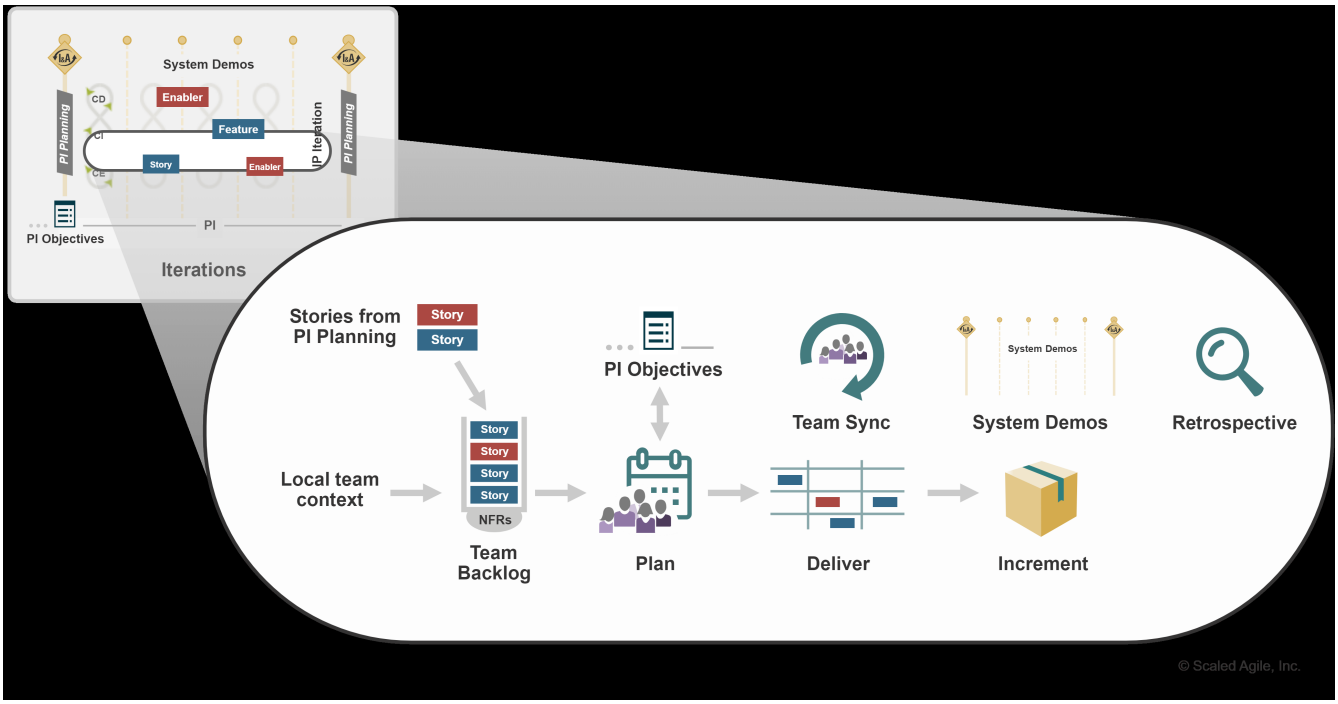
De sleutel tot succes ligt in het continu blijven leren, verbeteren en samenwerken op alle niveaus, zodat de transformatie niet alleen een middel tot een doel wordt, maar een integraal onderdeel van de manier waarop de organisatie opereert.

Met het LACE Hub-and-Spoke-model en een duidelijke eenduidige sterke visie op Lean-Agile werken is jouw organisatie goed gepositioneerd om haar doelen te bereiken en de operationele slagkracht te vergroten.

51. Ontwerp van het LACE Kanban board

Voor de Kanban-leken geef ik in dit hoofdstuk een mini-uitleg over Kanban. Kanban is een visuele methode voor het beheren van werkstromen en is een alternatieve aanpak ten opzichte van scrum. Beide zijn agile werkwijzen. Het helpt teams om processen te visualiseren, bottlenecks te identificeren en continu verbeteringen door te voeren. Kanban stamt uit de Japanse productie en wordt breed toegepast in Agile-omgevingen. Kanban bestaat uit de woorden KAN en BAN wat staat voor de Japanse woorden Visual + Board en is afkomstig van Toyota's TPS. In SAFe hebben de Agile Teams de keuze wat ze willen toepassen: Kanban of Scrum.

Mijn Kanban ontwerp aanpak bevat nog wat extra tekst en uitleg per design onderwerp voor degenen die nog geen experts in Kanban-ontwerp zijn.



SAFe Team Kanban

SAFe team Kanban of SAFe Scrum?

Kies voor SAFe Kanban als je werkt in een omgeving met veel variabele eisen, behoefte hebt aan continue workflow, en dynamische prioritering wilt toepassen en geen scope fix voor 2 weken wilt hebben.

Gebruik SAFe Scrum wanneer je vaste sprints/iteraties nodig hebt, duidelijke rolverdeling en ceremonies wilt, en stabiele eisen per iteratie hebt, wetende dat de Work Items 2 weken vastliggen en er bij voorkeur niet kan worden ingebroken op een iteratie (alleen de PO kan dit). Scrum biedt weliswaar meer structuur, maar Kanban biedt meer flexibiliteit voor snel veranderende taken met dus een *adapt to change* per minuut.

Keuze Kanban school + spelregels

We volgen primair de SAFe Kanban werkwijze en regels aangevuld met de theorie bij de opleiding van de *Kanban University* met de marktbekende certificaten TKP, KMP, KCP, etc. Parallel houden we ons aan de spelregels zoals beschreven in het Agile Manifesto. Onze specifieke Kanbans opereren niet als losse entiteit, maar als onderdeel van het grote SAFe met PI events, Planning Intervals, etc.

Doel van ons Kanban board

Het Kanban-board dat we opzetten, is bedoeld om de *workflow v*an een agile team, of bepaalde entiteit, te optimaliseren, het team te helpen prioriteiten te stellen, snel keuzes te maken en een goede balans te vinden tussen doorvoer en werkcapaciteit (demand/capacity). Een groot voordeel van een Kanban board is dat iedereen ook weet waar het team staat: Wat is onze werkvoorraad, wat zijn we aan het ontwikkelen, wat staat klaar voor acceptatie, waarover moeten we besluit nemen, wie moet wat doen om onze visie en missie te realiseren, etc.

En afsluitend ook voor het volgen en verbeteren van onze werkprocessen door visualisatie en focus op flow.

Fundamenteel Kanban principe: Pull system

Kanban werkt op basis van een pull-systeem, waarbij werk niet wordt "gepusht" naar de Agile-teamleden of de volgende Kanban states. Er wordt alleen werk opgepakt als er beschikbare (Story Points) capaciteit is. Dit voorkomt overbelasting en zorgt voor een gelijkmatige werkstroom en dus een continuous flow of value. Doel is dus het beheer van werk in een constante flow zonder het team te overbelasten.

Kanban roles

Hoewel Kanban geen formele rollen voorschrijft, zoals in Scrum, zijn er enkele rollen die vaak worden gebruikt om de workflow te organiseren en optimaliseren. Deze worden ook zo gedoceerd bij de Kanban University. Deze theoretische Kanban rollen bestaan uit: Service Request Manager (SRM), Service Delivery Manager (SDM), Team Members en Stakeholders

Daar wij SAFe Team Kanban principes volgen passen wij de 3 SAFe rollen toe: SM, PO en generiek teammembers. Deze rollen moeten bij het ontwerp ook worden toegewezen.

Kanban taal

De Kanban wordt opgezet in een bepaalde taal, die keuze dient te worden gemaakt. SAFe is primair Engels, de eerste state heet bijvoorbeeld Funnel en als iets klaar is DONE. Aanpassen kan, maar wees je bewust van de consequenties.

Kanban Work Item Types, Cards

In een Kanban systeem worden Work Items (geeltjes, blokjes, vierkantjes) gebruikt om het werk te visualiseren en te beheren. Elk Work Item vertegenwoordigt een Roadmap Artifact, een Core Competentie Artifact, taak of activiteit dat door de Kanban members moet worden opgepakt en voltooid. Work Items kunnen onderdeel van een groter geheel zijn, zoals een Feature of Epic. (SAFe of Scrum). Deze items worden door verschillende states geleid op het Kanban-board.

Door het werk op te delen in concrete items, kan het team de voortgang beter volgen en prioriteiten stellen. Wij volgen te allen tijde SAFe Lean-Agile Principle #6 om de Flow te maximaliseren.

Toekennen van assignees of task forces aan Work Items

Work Items kunnen worden toegewezen aan specifieke assignees of groepen van assignees, zoals task forces of werkgroepen. Dit gebeurt meestal door het benoemen van een persoon of team die verantwoordelijk is voor het uitvoeren van een Work Item primair in de “In Progress State”.

Elke kolom op het bord kan worden toegewezen aan een bepaalde groep, of een specifieke persoon, kan worden aangewezen op basis van hun capaciteit, expertise of beschikbare tijd, bijvoorbeeld in geval van development, testing, deployment etc.

Bij het toekennen van taken is het belangrijk om rekening te houden met de capaciteitsallocatie van teamleden en de geldende WIP Limits.

SAFe Lean-Agile Principle #6: WIP Limits (Work In Progress)

WIP-limieten beperken het aantal Work Items dat op een bepaald moment in een stage kan zijn. Dit voorkomt multitasking, te veel hooi op je vork, en bevordert een constante doorstroom van werk.

WIP limits zijn alleen nuttig als alle teamleden aan Artifacts werken en er gevaar dreigt van onderpresteren, te weinig flow of value. Zolang alles goed gaat, zijn WIP limits minder hard nodig. Note, WIP limits werken ook alleen als je gelijksoortig werk per stage hebt.

Kanban basisstructuur / States & Workflows

In ons Kanban-board worden werk-items door verschillende stadia van voltooiing geleid, ook wel "states" genoemd. Elke state is vertegenwoordigd door een kolom op het bord en helpt bij het visualiseren van de workflow.

State naam	Wip Limit	Doel / inhoud
Backlog/Funnel	None	Alles wat we moeten doen. 3 Sources: Roadmap, Core Competenties en vanuit de Org
Todo	None	Selectie van Funnel waar we aan gaan werken.
Refinement	Bepaal met je organisatie	Hierin wordt door de PO aan alle members uitgelegd wat een bepaald backlog Work Item inhoudt qua content, acceptatiecriteria, details, benefits, etc
In progress	Bepaal met je organisatie	Het Commitment point. Hier wordt het werk echt opgepakt en uitgevoerd en iemand, of een groep, ‘geassigned’
Impediment	Bepaal met je organisatie	Wanneer een taak niet kan worden voltooid door een impediment.
On Hold	Bepaal met je organisatie	taken die om wat voor reden dan ook even on hold komen te staan. Maw, er is al aan gewerkt, maar stopt even.
Besluitvorming	Bepaal met je organisatie	Work Items die een besluit nodig hebben. (bepaal wie er mag besluiten?)
Acceptance	Bepaal met je organisatie	Ontworpen Artifacts gaan naar deze state voor finale acceptatie. De LACE bepaalt wie de resultaten accepteert en of instemming van iedereen nodig is
Done	Bepaal met je organisatie	Alle work items die voldoen aan de DoD en formeel zijn geaccepteerd door de PO

Economische prioritering

In SAFe is Economische Prioritering, conform SAFe Principe #1 ‘Take an Economic View’ een cruciaal element om werk efficiënt te beheren en ervoor te zorgen dat de meest waardevolle en urgente taken eerst worden aangepakt. De LACE members moeten continu economisch prioriteren om waarde te maximaliseren en continu inspelen op veranderingen (MT’s die mogelijk plots een deliverable verwachten). Tijdens het prioriteren moet rekening worden gehouden met dependencies. In de praktijk kan een work item met een hogere prioriteit later moeten worden gedaan dan een lagere prio Work Item, puur vanwege een afhankelijkheid.

Er zijn drie prioriteringsmethodes:

• Door het toepassen van Kanban Standard Class of Service (Cost of Delay bepaald)

• WSJF (Weighted Shortest Job First): Economische prioritering

• Pull-System met een Dynamische Prioritering

Entry - & Exit Policies (per state)

In Kanban zijn *entry* en *exit* policies afspraken die duidelijk definiëren wanneer een werkitem een nieuwe fase mag *binnenkomen* of *verlaten* op het Kanban-bord. Deze policies helpen de workflow te stroomlijnen en zorgen ervoor dat werkitems alleen verder gaan wanneer ze voldoen aan de nodige criteria. Policies lijken sterk op de DoD, maar zijn niet hetzelfde.

Classes of Service / Swimlanes

Classes of Service zijn een belangrijk onderdeel van Kanban en worden gebruikt om verschillende soorten werkitems te categoriseren op basis van hun urgentie, risico of prioriteit. Oftewel de *Cost of Delay.* Elke "class of service" bepaalt hoe een item wordt behandeld in het systeem en helpt het Kanban Team om snel te reageren op veranderende omstandigheden, zonder de workflow te verstoren. Door Work Items te categoriseren, kan een team beter omgaan met de diverse soorten taken die in hun systeem circuleren.

Wij passen deze standaard Classes of Service toe:

• Expedite (Spoed, bloed, must do)

• Fixed Date (Vaste datum)

• Standard (Standaard)

• Intangible (niet tastbaar)

Alternatieve Classes of Service zijn categoriserings methodes op Epics of op Assignees. Agile software kan hier heel handig mee omgaan en multiple views genereren. Filters op een naam bouwen is ook een handige en snelle sorteringsmechaniek.

Commitment Point

Dit is het punt waarop het team zich ertoe verbindt een werkitem op te pakken en te voltooien. Dit kan helpen om verwachtingen te managen en het team te focussen op werk dat daadwerkelijk wordt uitgevoerd. Tot dit moment bevinden de items zich meestal in de Funnel en/of Backlog state, waar ze kunnen worden toegevoegd, geprioriteerd of aangepast zonder directe actie. Zodra een werkitem het commitment point passeert, is het team toegewijd om het binnen een redelijke termijn af te ronden. Teams kunnen een Work Item pas oppakken als alles m.b.t. het Work Item duidelijk is. Zie de states voor ons Commitment point.

(Flow) metrics & reporting

Kanban maakt gebruik van metrics zoals lead time, cycle time, throughput en cumulative flow diagrams om de workflow te analyseren en verbeteren. Wij gebruiken deze Metrics om onze Flow te analyseren:

Metrics:

• Lead time

• Cycle time

• Throughput

• Cumulative Flow Diagram (CFD)

Events (incl PDCA loop)

Zie SAFe Team Kanban voor de events, veel events zijn nagenoeg identiek aan de formele SAFe Iteration events.

Concreet aan de slag!

In Kanban begin je met werkitems altijd **links** op het bord, waar de eerste status staat (bijvoorbeeld "To Do" of "Funnel"). Vanaf daar doorlopen de items de workflow van **links** naar **rechts**, tot ze de eindstatus bereiken, zoals "Done". Je werkt items af door ze naar de **volgende** kolom te verplaatsen zodra ze aan de **exit policies** van de huidige status voldoen en de **entry policies** van de volgende status voldoen. Je werkt dus niet van rechts naar links en je begint ook niet bij Done om vervolgens naar To Do te werken.

Classes of Service toegepast?

Als je werkitems hebt in de **Expedite Class of Service**, heeft dit de hoogste prioriteit en is het verstandig hiermee te starten zodat deze 100% zeker worden besproken. Deze items worden snel door de workflow getrokken, vaak met voorrang op ander werk, omdat ze van kritisch belang zijn.

Hoe werk je de items weg van state naar state?

1. **Pull-systeem**: De Kanban teamleden trekken werk van het bord zodra ze beschikbaar zijn om een nieuwe taak op te pakken, te beginnen bij de meest linkse kolom met items die klaar zijn voor de volgende fase. Items zijn klaar als ze in de OUT kolom van de state staan. (als de software dat aankan en toelaat)

2. **WIP-limieten**: WIP limits hebben alleen zin als alle teamleden regulier Work-Items oppakken en het werk evenredig is verdeeld onder de Kanban members. Met andere woorden, per state, werken er meerdere team members solitair aan een Work Item.

Wanneer wordt het Kanban-bord bijgewerkt?

Het Kanban bord wordt conform SAFe **continu bijgewerkt** om de actuele status van werkitems en de workflow weer te geven. BVIR’s moeten in SAFe altijd up-2-date zijn zodat stakeholders, andere teams altijd een actuele stand van zaken inzichtelijk hebben.

Het bijwerken van het bord is een teamverantwoordelijkheid, waarbij teamleden en formeel de Kanban Service Delivery Manager (SDM) een actieve rol spelen. Echter, onze keuze is de SAFe Kanban regels te volgen, waarbij de **Kanban PO** rol de **content authority** is en de **Value Maximiser.** Deze rol is primair verantwoordelijk voor de ‘iteration- en team backlog’, of items waaraan gaat worden gewerkt,

Hier is hoe en wanneer een Kanban-bord wordt bijgewerkt:

• **Tijdens het werk**: Zodra een werkitem van status verandert (bijvoorbeeld van "In Progress" naar "Done"), moet het bord direct worden bijgewerkt door het teamlid dat aan het toegewezen work-item werkt.

• **Team Sync Event**: Tijdens de dagelijkse 15 min team sync wordt het bord gezamenlijk doorgenomen. Teamleden geven updates over hun werk en verplaatsen items naar de juiste kolom.

• **Bi-weekly sync events**: Dit is geen officieel SAFe-event en kent een flexibele agenda die wordt afgestemd op de behoeften van het team. Een vast onderdeel is het kort doornemen van de algehele status met het team. Normaal gesproken is iedereen op de hoogte van de voortgang, tenzij het *Daily Team Sync*-event niet plaatsvindt.

• **Bij nieuwe input**: Als er nieuwe taken, prioriteiten of blokkerende issues opkomen, registreert het team deze en stemt het af hoe ze worden verwerkt. Ook bij Scrum kunnen nieuwe inzichten tot aanpassingen leiden. Het team stemt wijzigingen af met de Product Owner, zonder het Sprint Goal in gevaar te brengen.

• **Na voltooiing van een taak**: Zodra een taak of werkitem is afgerond, wordt het naar de OUT van de current Stage verplaatst (bijv. "In progress-out").

• **Blokkades aangeven**: als er een blokkade ontstaat (bijvoorbeeld door een afhankelijkheid van een ander team), wordt dit op het bord aangegeven, zodat de status duidelijk blijft voor iedereen.

Note: Als de WIP-limieten zijn bereikt, mogen er geen nieuwe taken naar een kolom worden verplaatst totdat er weer capaciteit vrijkomt. De tooling maakt een overschrijding van de WIP-limiet zichtbaar. Of nieuwe items ook daadwerkelijk worden geblokkeerd, hangt af van de gebruikte tooling en inrichting.

Werkafspraak voor dit LACE: de Product Owner bewaakt de inhoud en prioritering van de Team Backlog. We spreken af dat nieuwe work items worden aangemaakt door de PO of door iemand aan wie de PO dit heeft gedelegeerd. Teamleden brengen nieuwe ideeën en werkzaamheden daarom eerst in bij de PO. Zo blijven overzicht en samenhang behouden en worden dubbele items voorkomen. Dit aanmaakrecht is een lokale werkafspraak.

52. Van PMO naar VMO: de transformatie naar waardegedreven governance in SAFe

Inleiding

Veel organisaties kennen een Project Management Office (PMO) als een centraal orgaan voor projectgovernance en resourceplanning. Maar met de adoptie van Lean-Agile frameworks zoals SAFe 6.0 verschuift de focus van projectmatig werken naar waardegedreven governance. Dit brengt een *fundamentele* verandering met zich mee: het PMO moet transformeren naar een Value Management Office (VMO).

Maar wat betekent dit precies? Wordt een PMO automatisch een VMO? Is het simpelweg een naamswijziging, of vraagt dit om een compleet andere mindset en werkwijze? En welke rol speelt het Lean-Agile Center of Excellence (LACE) in deze transitie?

In dit hoofdstuk behandel ik:

• De verschillen en overeenkomsten tussen een PMO en een VMO.

• Waarom een PMO niet automatisch een VMO wordt.

• De competenties die nodig zijn voor een VMO-rol.

• Hoe de overgang van LACE naar VMO kan verlopen.

• De impact op governance en portfolio management.

1. Wat is een PMO in een traditionele organisatie?

Definitie en functie

Een Project Management Office (PMO) is een centraal ondersteunend orgaan binnen een traditionele projectorganisatie, vaak op basis van PRINCE2 of MSP. Het richt zich op standaardisatie, controle en administratieve ondersteuning van projecten en programma’s.

Kernverantwoordelijkheden

• Standaardisatie van projectmanagementmethodieken en processen

• Ondersteuning van project- en programmamanagers

• Risicobeheer en issue management

• Capaciteits- en resourceplanning

• Rapportages en compliance-controles

• Governance op tijdelijke projecten en budgetbeheer

Scope en beperkingen

• Projectgericht: PMO’s werken met tijdelijke projecten met een start en einde, en verplaatsen resources van het ene project naar het andere.

• Governance-gericht: de focus ligt op controle, deadlines en compliance, minder op continue waardecreatie.

• Bureaucratisch: PMO’s zijn vaak document-gedreven en minder wendbaar in snel veranderende omgevingen.

2. Wat is een VMO in SAFe 6.0?

Definitie en functie

Een Value Management Office (VMO) is het governance-orgaan binnen SAFe dat zich richt op strategische alignment, Lean Portfolio Management (LPM) en continue waardecreatie. Waar een PMO projecten beheert, faciliteert een VMO de flow van strategische waarde binnen en tussen Value Streams.

Kernverantwoordelijkheden van een VMO

• Lean-budgettering: Toewijzen en beheren van investeringen op basis van Epics en strategische doelen

• Portfolio Kanban: Sturen van strategische initiatieven en Epic-governance

• Strategische governance: Aligneren van portfolio's met bedrijfsdoelen

• Coördineren van waardestromen: Oplossen van afhankelijkheden tussen ARTs en Value Streams

• Meten van business outcomes: Focus op flow-efficiëntie en strategische impact

• Coachende rol: Ondersteunen van Lean Portfolio Management (LPM) en LACE

• Faciliteren van LPM events: Portfoliobeoordelingen, strategische alignment en PI-planning

Scope en werkwijze VMO

• Continu waardegericht: Richt zich niet op projecten, maar op voortdurende waardecreatie en flow-optimalisatie.

• Agile en strategisch: Flexibeler dan een PMO en gericht op langetermijnwaarde, niet op projectdeliverables.

• Data-gedreven: Beslissingen worden gebaseerd op flow metrics en business outcomes, niet op budgetcycli of deadlines.

3. Wordt een PMO automatisch een VMO?

In het kort: Nee. De rol en mindset van een VMO zijn fundamenteel anders dan die van een PMO. Een VMO opereert op een strategisch niveau en werkt waardegedreven, terwijl een PMO zich richt op projectgovernance en administratieve controle.

Kenmerk	PMO (PRINCE2/MSP)	VMO (SAFe 6.0)
Focus	Project- en programmamanagement	Lean Portfolio Management & waardecreatie
Governance	Controle & compliance	Wendbaarheid & strategische alignment
Scope	Tijdelijke projecten	Continue flow van waarde
Werkwijze	Bureaucratisch, documentgericht	Agile, adaptief, strategisch
Meetmethode	Oplevering & deadlines	Business impact & flow-efficiëntie

Wat moet er veranderen?

Voor een succesvolle transitie van PMO naar VMO is het belangrijk dat de organisatie:

• De mindset verschuift van projectgebaseerde governance naar waardegestuurde governance.

• Projectbudgetten loslaat en overstapt op Lean Budgeting.

• Meer strategische competenties ontwikkelt binnen het team.

• Gaat werken met Portfolio Kanban en Flow Metrics

In veel organisaties transformeert een PMO naar VMO doorgaans middels een omscholingstraject en SAFe coaching die essentieel zijn voor een succesvolle transitie.

4. LACE als opstap naar een VMO

Wat doet een LACE?

Het Lean-Agile Center of Excellence (LACE) is het implementatieorgaan binnen een SAFe-transformatie. Het ondersteunt de organisatie bij de adoptie van Agile en Lean-principes en is primair bezet door veranderkundigen, SAFe-experts (SPC's, SPCT's) en organisatiedeskundigen van de organisatie zelf. Kan een LACE overgaan in een VMO? Ja, in sommige organisaties, als de focus verschuift van transformatie naar strategische governance en (heel belangrijk!) de organisatie SAFe-volwassen genoeg is.

5. Welke competenties heeft een VMO-team nodig?

Een VMO bestaat vaak uit een klein gespecialiseerd Agile Team (3 tot 10 FTE) dat zelf ook werkt conform de SAFe Lean-Agile Principes, heeft een PO, een SM en een eigen backlog. Een VMO draait gewoon mee in elke iteratie en plant ook tijdens het PI event.

Kerncompetenties

1. *Strategisch denken*: Kan schakelen tussen operationele en strategische doelstellingen en begrijpt hoe waardestromen bijdragen aan bedrijfsresultaten.

2. *Lean-Agile mindset* - Begrijpt en past Lean-Agile principes toe, inclusief flow-based werken en continue verbetering.

3. *Portfolio Governance & Lean Budgettering*: Ervaring met financieel management op portfolio-niveau en het sturen van investeringsbeslissingen op basis van waarde.

4. *Stakeholdermanagement* - Kan effectief communiceren met RTE's, STE's en Scrum Masters en kan groepen faciliteren en begeleiden in besluitvorming.

5. *Facilitator skills* - Kan grote groepen begeleiden, structureren en faciliteren bij besluitvorming en samenwerking.

6. *Data-gedreven besluitvorming*: Werkt met metrics zoals flow-efficiëntie, cycle time en business outcomes om beslissingen te onderbouwen.

7. *Systeemdenken* - Kan organisaties als samenhangende systemen bekijken en optimaliseren om bottlenecks in waardestromen te elimineren.

8. *Organisatieontwikkeling en verandermanagement*: Begrijpt hoe men een cultuurverandering begeleidt en een organisatie helpt bij het adopteren van Lean Portfolio Management (LPM).

9. *Technologisch bewustzijn*: Begrijpt de impact van technologie op de organisatie en kan samenwerken met architecten en IT-leiders.

10. *Oplossingsgericht en besluitvaardig*: Kan snel complexe problemen analyseren en tot actie overgaan zonder te verzanden in bureaucratie.

11. *Multi-level SAFe-samenwerking* - Werkt samen met Agile Release Trains (ARTs), Solution Trains en externe leveranciers in complexe waardestromen. Kan effectief communiceren met RTE's, STE's en Scrum Masters en

hun taal en terminologie begrijpen.

Opleidingsniveau en ervaringsniveau

• Minimaal WO/HBO+ niveau in bijvoorbeeld bedrijfskunde, informatiemanagement, finance, of een technische richting.

• Ervaring op senior managementniveau (bij voorkeur in portfolio management, LACE, of Lean-Agile implementaties).

• Doorgewinterde kennis van SAFe, bij voorkeur Leading SAFe, LPM, DevOps, Agile Product Delivery gecertificeerd.

• Ervaring met strategisch portfolio management.

• Kennis van en ervaring met Agile Tooling zoals ADO, Jira, Jira-Align, Confluence etc.

6. Conclusie

Een PMO transformeert *niet automatisch* in een VMO.

Dit vraagt namelijk om:

✓ Een fundamentele mindset-shift van projecten naar waarde
✓ Nieuwe governance- en budgetprocessen binnen SAFe
✓ De juiste competenties en strategische skills
✓ Gedegen SAFe kennis, ervaring en een aantal certificeringen.

Het LACE speelt een sleutelrol in deze overgang, maar een VMO heeft een bredere focus dan implementatie alleen.

Voor een organisatie die echt waardegedreven wil werken, is er meer nodig dan alleen een naamswijziging van PMO naar VMO. Maar met de juiste hulp en middelen is dat meerdere organisaties succesvol gelukt!

53. SAFe-governance: structuur in een wendbare organisatie

Introductie

De term *governance* komt vaak voorbij en is nu opnieuw zeer actueel, mede door het onderzoek van Capgemini. Toch wordt governance door sommigen ervaren als een vage managementterm, wat leidt tot vragen zoals:

• Wat wordt er precies mee bedoeld?

• Wat valt er wel en niet onder?

• Hoe verschilt SAFe-governance van traditionele governance?

• Hoe passen marktpartijen in een SAFe-governancestructuur?

Governance is een essentieel onderdeel van SAFe en moet in het TOM (Target Operating Model) helder en gedetailleerd worden uitgewerkt. Het vormt de basis voor structuren en processen die nodig zijn om doelen te bereiken, risico's te beheersen en verantwoording af te leggen.

Bij traditionele organisaties, zoals overheden en defensie, is governance vaak hiërarchisch ingericht, met uitgebreide rapportages en stuurgroepen. In een wendbare omgeving zoals binnen SAFe verandert governance echter fundamenteel. Hier ligt de focus op waardecreatie, snelheid en flexibiliteit, wat vraagt om een andere aanpak.

In dit hoofdstuk neem ik je mee in het onderwerp governance. Ik begin met een algemene definitie, bespreek de belangrijkste componenten van governance en verdiep me vervolgens in hoe SAFe-governance werkt en de totale besturing plaatsvindt. Daarnaast wil ik kennis over het modelleren van governance in het TOM met je delen.

Definitie

"Een volledige beschrijving van de ophanging, rollen, taken, verantwoordelijkheden & bevoegdheden (TVB's), besluitvormingsprocedure en -structuren en communicatielijnen, etc. Het SAFe operating system is specifiek voor het behalen van de gewenste strategische veranderingen (strategic Themes met OKR's) naast het bestaande hiërarchische organisatiemodel, The Hierarchy', gepositioneerd. Deze constructie dient geen belemmering en bedreiging te zijn voor elkaars werkzaamheden en verantwoordelijkheden"

Governance componenten en hoe het werkt in SAFe

Hieronder de diverse building blocks, componenten van het ontwerp van een governance model, waarbij ik traditioneel uitzet tegen SAFe.

Component	Traditionele governance	SAFe governance
Structuur / ophanging	Hiërarchisch, harkjes en silo-gebaseerd	Value stream- en team -georiënteerd. Scaled. Ook externe partijen moeten ergens worden 'opgehangen in SAFe'
Opdrachten geven en werk verdelen	Notes, Projectplannen, mensen naar werk	Werk naar Agile Teams. Via LPM Epics, naar leading ARTs naar Agile Teams
Accountability	Top management, stuurgroepen, PGM, PM etc. Top down. PGM is overal van: Process, Technology en Content.	Lastiger verhaal, want in SAFe zijn Agile Teams zelforganiserend en verantwoordelijk voor het leveren van een potentieel inzetbaar product (Potentially Shippable Product) aan het einde van elke sprint. Ook is in SAFe de Accountability gesplitst in Process, Technology en Content. Accountability is veel transparanter: Teams

		committeren zich tijdens PI Planning aan hun team-PI Objectives. (De persoon aan wie het backlog item is 'geassigned' *is er ook van)*
Rollen	Stuurgroepen, PM, PGM, boards, etc, ongereguleerd. Elk project bepaalt zelf.	Alle SAFe rollen zijn beschreven en qua takenpakket voor 90% vastgelegd. 10% org afhankelijk.
Besluitvorming	Trage cycli via MT-vergaderingen en stuurgroepen. Vaak hiërarchisch en gepaard met veel papier/stukken.	Decentralisatie via LPM, ART's en Agile teams. SAFe experts volgen 'decentralised decision making, conform SAFe principe #9. Tempo van besluitvorming ligt veel hoger. Tijdens PI events.
Meeting structuren	Stuurgroep, programmamanagement overleg, teamoverleg, bilats, etc	Alle meetings in SAFe zijn scaled, predefined, uitgedacht en ge-aligned. En voor iedereen hetzelfde met dezelfde naam, dezelfde agenda en dezelfde tijden. Maar kunnen wel op maat worden gemaakt voor de specifieke org.
Papier	Veel papier: eerst denken en dan doen	Weinig papier: learn fast through experimentation' (start fast (MVP), fail fast, fail often)
Focus	Controle en risicovermijding	Waardecreatie en continu leren
Rapportages	PRINCE2-achtige "highlight reports" met focus op voortgang. Stuurgroep cycli	SAFe kent een 7x24 uurs rapportage, want alle Big Visual Information Radiators (BVIRs) dienen altijd up2date te zijn met 100% transparantie (iedereen mag erbij). 1x per PI vindt er een Inspect & Adapt plaats om de laatste PI Metrics te delen, plus elke iteratie gedurende de Iteration Review. LPM heeft sync events met status updates.
Sturing	Op deadlines en op voortgang	Primair op de LPM backlog en vervolgens op de PI objectives. (en realisatie OKR's etc)
Scope	Projectmatig en afgebakend. Changes via trage CAB's	Productmatig en gericht op doorlopende levering. Change kan 7x24 uur, met impact bepaling door de PO en herprioritering van de team backlog. (en de ART backlog)
Escalatie	Via PM, via PGM naar stuurgroep en mogelijk management	Via Team naar ART naar LPM. Escalaties blijven in SAFe, tot die er niet meer uitkomen. Dan escalatie naar The Hierarchy.
Heart beat	Ritme wordt bepaald door MT cyclus, stuurgroep cyclus en op PGM en PM level wat er in het Projectplan staat. Elk project heeft zijn eigen heart beat	Er is 1 grote heart beat voor iedereen in SAFe. Een LPM, ST, ART en een team heart beat. SAFe heeft daar al over nagedacht, kleine aanpassingen situationeel mogelijk.
Prioritering	Bijvoorbeeld HSE (hardste schreeuwer eerst), MoSCoW of prioriteitenlijsten in spreadsheets.	WSJF allemaal in de Agile tooling met gecascadeerde prio's van LPM naar ART naar Teams.
Metrics	Tijd, geld, kwaliteit en Business Case	SAFe experts sturen op een volledige set aan Metrics en bouwen uitgebreide managementdashboards over de *hele SAFe machine heen.* Deze boards zijn altijd up2date ('refresh') en volledig

		geautomatiseerd. Management heeft inzage in alle ARTs, teams, LPM wat er in SAFe gebeurt op alle dimensies. LPM stuurt Epics op Benefits Realisation met primair Leading Indicators.
Gevaren	Risk en issues, vaak in aparte xlsx logs. Exception logs	Roamed risks en Impediments, allemaal gelogd in de Agile tooling met 1 groot overzicht over heel SAFe voor Executive Management.

Conclusie

De SAFe-governance biedt een heldere structuur en aanpak voor organisaties die wendbaarder willen worden in een dynamische omgeving. Waar traditionele governance vaak gericht is op controle en risicovermijding binnen hiërarchische kaders, verschuift de focus binnen SAFe naar waardecreatie, snelheid en transparantie. Dit vraagt om een *fundamentele herinrichting* van rollen, besluitvormingsprocessen en rapportagestructuren.

De kracht van SAFe-governance ligt in de decentrale besluitvorming, transparante accountability en de uniforme ritmes (cadence) binnen Agile Release Trains en de agile teams. Dit zorgt niet alleen voor hogere snelheid en flexibiliteit, maar ook voor een betere afstemming tussen teams en managementlagen.

Door een zorgvuldig ontworpen *Target Operating Model* te hanteren (taak van de LACE), kunnen organisaties traditionele en wendbare structuren naadloos integreren zonder elkaar in de weg te zitten.

Met SAFe-governance wordt duidelijk dat het niet gaat om het afschaffen van bestaande structuren, maar om het toevoegen van een wendbare dimensie waarmee strategische doelen sneller en efficiënter kunnen worden bereikt.

54. Waarom we géén werkrollen op ART-niveau willen, en welke alternatieven SAFe biedt

Waarom we géén werkrollen op ART-niveau willen, en welke alternatieven SAFe biedt

Inleiding

In veel SAFe-implementaties ontstaat de verleiding om **teamoverstijgende specialisten** zoals inkopers, integratiemanagers of testcoördinatoren een plek te geven “op ART-niveau”. Naast de **faciliterende rollen** als Release Train Engineer (RTE), Product Manager (PM) en System Architect (SA) duiken er dan allerlei ‘losse rollen’ op die werk verrichten, maar geen onderdeel zijn van een Agile Team. Een RTE heeft geen User Stories, een PM en SA ook niet. Alhoewel de PM Features aanmaakt en de SA Enablers, worden deze door derden ontwikkeld. De Agile Teamleden (= Dev Team members) zijn eigenaar van de User Stories en 'doen het werk' zoals gedefinieerd in de Team Backlog(s).

In een Target Operating Model (TOM) kun je bewust onderscheid maken tussen tactische afstemming op ART-niveau en operationele uitvoering binnen teams. In dit hoofdstuk leg ik uit waarom ik losse uitvoerende rollen op ART-niveau afraad en welke **SAFe-conforme alternatieven** er zijn om deze specialisten te positioneren.

Waarom geen 'werkende rollen' op ART-niveau?

1. Onduidelijkheid over verantwoordelijkheid

Als mensen buiten de teams werk verrichten, is het onduidelijk:

• Wie verantwoordelijk is voor de uitvoering

• Waar de backlog van hun werk zich bevindt (een ART Kanban board kent Features die worden geïmplementeerd middels user stories)

• Hoe hun werk wordt gepland, gecoördineerd en beoordeeld

2. Verkeerde beeldvorming: lijken 'sturend' te worden

Specialisten op ART-niveau worden onbedoeld vaak gezien als *‘managers van de ART’*. Ze zijn immers niet in een team gepositioneerd en nemen toch beslissingen of voeren werk uit. Dit verstoort het beeld van *decentrale verantwoordelijkheid.*

3. Geen operationeel planningsevent

Op ART niveau vindt er geen Iteration Planning event plaats waar Dev Team Members worden geassigned aan backlog Items. Features zijn namelijk geen User Stories, Features bestaan uit x User Stories die op iteration niveau worden ontwikkeld.

Welke alternatieven biedt SAFe?

Om deze problemen te voorkomen, biedt SAFe drie duidelijke alternatieven:

1. Complicated Subsystem Team

Toepassing: voor specialisten die werken aan complexe of specialistische domeinen die meerdere teams raken.

Voorbeelden:

• Inkoop & contractmanagement

• Security compliance

• Privacy

• UX/GUI designers

Kenmerken:

• Volwaardig Agile Team in termen van WoW. (dit team kan in principe uit minder leden bestaan).

• Werkt met iteraties, backlog, DoD, PI Planning

• Neemt deel aan System Demo en Inspect & Adapt

Voordeel: Helder teamkader, duidelijke werkverdeling, inspecteerbaar en schaalbaar.

2. Shared Services

Toepassing: voor schaars beschikbare experts die tijdelijk bijdragen aan één of meer teams, zonder volledig lid te zijn van een team.

Voorbeelden:

• UX-designers

• Juridisch adviseurs

• Inkopers

• Performance-testers

Kenmerken:

• Worden op het juiste moment betrokken bij werk van Agile Teams

• Hun werk wordt, via de PO, ingepland op de Team Backlog van het desbetreffende team.

Voordeel: Shared resources, flexibel inzetbaar, zonder een vaste teamstructuur te verstoren.

Let op: Vereist goede afstemming in PI Planning (10 week forecast) en Iteration Planningen per team.

3. System Team op ART-niveau

Toepassing: voor ART-brede technische ondersteuning, tooling, integratie, systeemtesten en deployment pipelines.

Voorbeelden:

• CI/CD support

• Testen (Het team moet zelf zorgdragen voor *integrated-increments, het System Team ondersteunt dit*)

• Testdata- en omgevingsbeheer

Kenmerken:

• Teamoverstijgend: Ondersteunt alle teams binnen de ART

• Werkt ook agile en incrementeel en is een formeel Agile Team in haar WoW.

• Vaak samengesteld uit technische specialisten

Voordeel: Ondersteunt flow van alle teams, voorkomt versnippering van technische enablement.

Conclusie

Operationeel werkende specialisten op ART-niveau leiden tot structuurruis, rolverwarring en gebrek aan transparantie.

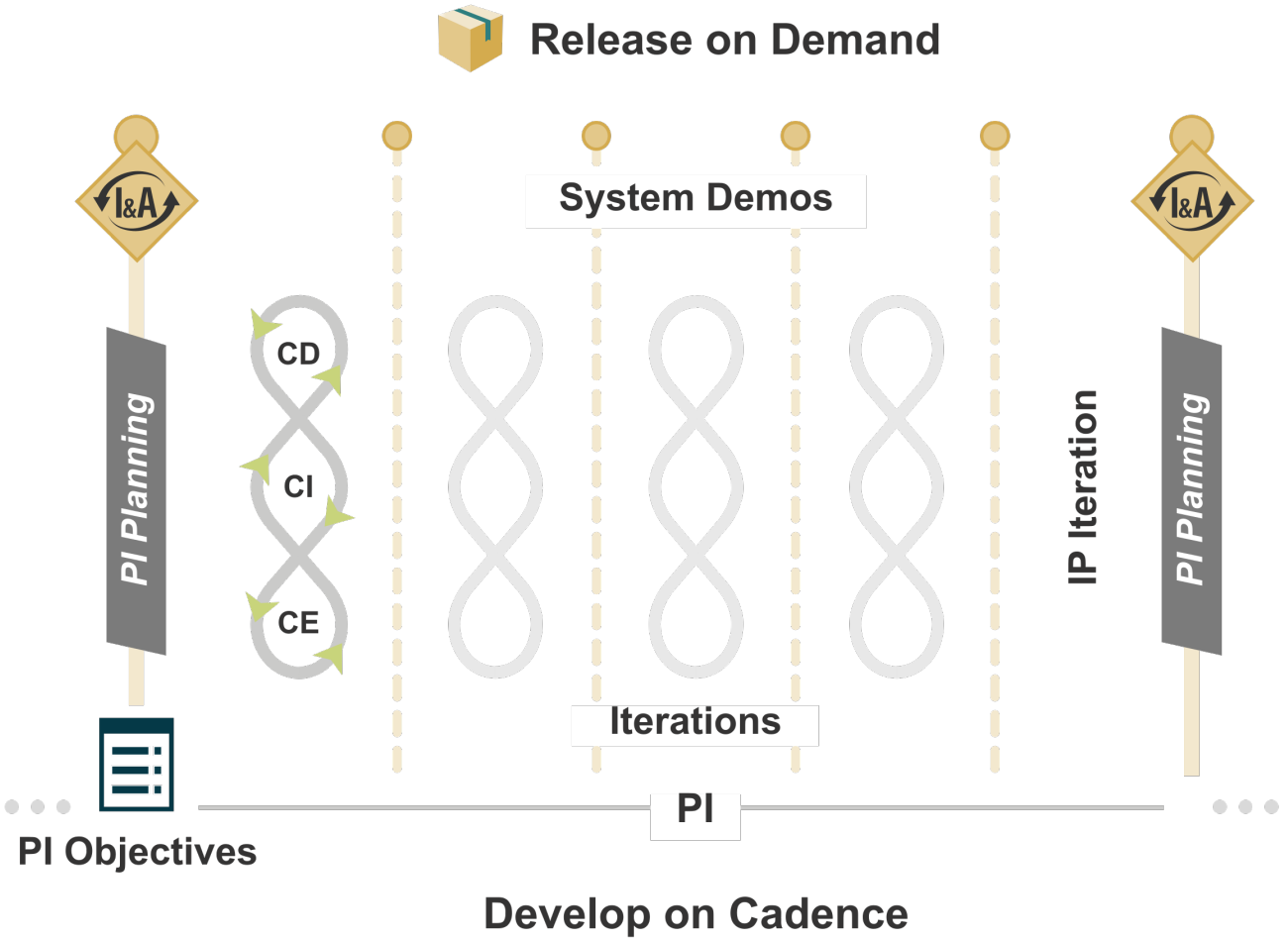
SAFe biedt drie goede alternatieven:

1. **Complicated Subsystem Team**- wanneer werk structureel en complex is
2. **Shared Services**- wanneer de inzet incidenteel of specialistisch is
3. **System Team**- wanneer de inzet technisch en ART-overstijgend is

Deze oplossingen houden je ART **schaalbaar, zuiver en professioneel**, zonder afbreuk te doen aan wendbaarheid en samenwerking.

55. De enterprise cadence: ontwerpcriteria voor een werkend SAFe-ritme

Inleiding



© Scaled Agile, Inc.

Binnen SAFe is een “Cadence” niet “slechts planning”; het is een primair mechanisme dat grootschalige coördinatie economisch haalbaar maakt door voorspelbaarheid te creëren en synchronisatie mogelijk te maken over meerdere domeinen (business, technologie, compliance, operations).

SAFe-principe #7 positioneert cadence expliciet als het middel om voorspelbaarheid te creëren, terwijl synchronisatie ervoor zorgt dat meerdere perspectieven “gelijktijdig worden begrepen, opgelost en geïntegreerd”, precies wat enterprise-brede planning en integratie vereisen.

Daarnaast behandelt SAFe flow als een eerste-orde zorg (“SAFe is een flow-gebaseerd systeem”), en flow wordt gemakkelijk verstoord door niet-afgestemde ritmes, te grote batches en ad-hoc onderbrekingen. De flow-richtlijnen binnen SAFe benadrukken dat organisaties systematisch onderbrekingen in de waardestroom moeten identificeren en oplossen als onderdeel van een continu waardeleveringssysteem.

Daarmee is het bepalen van de cadence een van de meest impactvolle beslissingen in een enterprise SAFe®-transformatie, naast bijvoorbeeld ook het ontwerpen van je formele requirementsmodel, waarin je je work-itemtypen vaststelt. Als een van beide niet goed is ontworpen, heeft de hele enterprise daar veel last van. Beide zijn besluiten die bij een SAFe-transformatie niet aan individuele O-LACEs of ART’s overgelaten kunnen worden: ze moeten enterprise-breed en uniform worden genomen, anders valt de basis onder samenwerking, rapportage en portfoliobeheer weg.

In dit hoofdstuk neem ik je mee in het proces waarmee je, vóórdat je ook maar één datum prikt, de randvoorwaarden voor een enterprise cadence vastlegt. Deze criteria zijn gebaseerd op SAFe® 6.0.2 en vele jaren cadence-ontwerpervaring, en vormen de basis voor het consensusproces met alle betrokken stakeholders.

Stap 1: bepaal eerst de fundamentele uitgangspunten

Voordat je aan ontwerpcriteria toekomt, moeten de volgende fundamentele keuzes gemaakt worden:

1. Bepaal de SAFe-configuratie

De configuratiekeuze (Essential, Large Solution, Portfolio of Full SAFe) bepaalt welke cadence-niveaus je überhaupt ontwerpt. Bij Essential SAFe ontwerp je alleen de ART- en teamcadence. Bij Large Solution komt de Solution Train-cadence erbij. Bij Portfolio en Full SAFe voeg je daar de LPM-cadence aan toe. Zorg dat je hierover eerst met elkaar consensus bereikt, en leg vast welke portfolio's met welke configuraties werken.

2. Bereik overeenstemming over het gebruik van drie Lean-Agile principes als ontwerpleidraad

Drie SAFe-principes zijn direct leidend voor cadence-ontwerp:

• **Principle #6: Make Value Flow Without Interruptions.** Dit principe introduceert acht flow-accelerators die direct relevant zijn voor cadence-ontwerp, waaronder de belangrijke "Optimize Time In the Zone" flow accelerator en "Remediate Legacy Policies and Practices".

• **Principle #7: Apply Cadence, Synchronize with Cross-Domain Planning.** Het cadence-principe bij uitstek. Cadence transformeert onvoorspelbare events in verwachte events, verlaagt transactiekosten van planning en integratie, en maakt wachttijden voorspelbaar. Synchronisatie zorgt ervoor dat meerdere perspectieven tegelijkertijd begrepen, opgelost en geïntegreerd worden.

• **Principle #9: Decentralize Decision-Making.** Dit principe stelt dat beslissingen zo dicht mogelijk bij de bron van informatie en expertise moeten worden genomen: op het laagst mogelijke niveau waar de benodigde kennis en context aanwezig zijn. Voor cadence-ontwerp betekent dit dat niet alle timing- en formatbeslissingen enterprise-breed hoeven te worden vastgelegd: teams houden vrijheidsgraden voor hun eigen ceremonies, mits ze binnen de enterprise-kaders opereren. Expliciete grenzen stellen (wat is enterprise-breed vastgelegd, wat is lokaal bepaalbaar) voorkomt zowel centrale overregulering als lokale fragmentatie.

3. Afwijking van formele SAFe-benamingen

Bepaal op enterprise-niveau in welke mate mag worden afgeweken van de formele SAFe-eventbenamingen, en leg dit vast. Noemen we het PI Planning-event "PI Planning" of "kwartaalbijeenkomst"? Leg ook de onderbouwing voor elke afwijking vast.

1. **Geografische spreiding en tijdzones**

Bij grote internationale organisaties is distributie een first-order ontwerpfactor. SAFe bevat expliciete richtlijnen voor distributed PI Planning. Anders ontstaat er onvermijdelijk conflict tussen "gelijke data voor iedereen" en praktische beperkingen zoals reistijd, kosten en schaarse experts.

Stap 2: bepaal de ontwerpkeuzes op LPM-, ST- en ART-niveau

De volgende ontwerpkeuzes vormen de randvoorwaarden voor het ontwerp van de cadence in een Full SAFe-configuratie.

Bepaal de governance: SAFe zelf is hierover stil: het framework beschrijft wát de cadence inhoudt, maar benoemt geen eigenaar. Leg als ontwerpkeuze vast wie de cadence mag aanpassen, op welke gronden en met welke aankondigingstermijn richting teams. Zonder dit besluit worden cadence-aanpassingen ad hoc doorgevoerd, stapelen uitzonderingen zich op en verliest het ritme zijn kracht. Cadence is een architectuurkeuze.

Architectuur wordt niet ad hoc aangepast.

Besluitritme versus leveringsritme: Een veelvoorkomende ontwerpfout is het verwarren van leveringsritme met besluitritme. Deze ritmes lopen namelijk in de harde praktijk zelden synchroon.

• Leveringsritme: PI-cyclus

• Besluitritme: Portfolio approvals

• Strategieritme: Jaarplannen en beleidsperioden

• Budgetritme: Financiële herijkingen

Ontwerp expliciet hoe strategie wordt vertaald naar PI-cadence, en voorkom strategische herprioritering midden in een lopend PI. Stabiliteit van uitvoering vereist voorspelbaarheid van besluitvorming.

Release-strategie als onderdeel van cadence-ontwerp: Cadence kan niet los worden ontworpen van release-mechaniek. Belangrijke ontwerpkeuzes zijn:

• Release per PI

• Release per iteratie

• Continue deployment

• Hardware-release cadence

Wanneer release-strategie niet expliciet wordt gekoppeld aan cadence, ontstaat impliciet PI-batching. Dat vergroot Flow Time en vertraagt waarde-levering. Cadence moet waarde versnellen, niet opslaan.

Aansluiting bij bestaande organisatie-cadence/rapportages: De SAFe-cadence moet aansluiten bij bestaande plannings- en besluitvormingscycli van de organisatie: begrotingsrondes, beleidsperioden, jaarplanningen en enterprise rapportage rollups. Neem op enterprise-niveau een bewust besluit: past SAFe zich aan de bestaande cadence aan, of andersom? Leg vast welk besluit is genomen en met welke onderbouwing.

Maximaliseer het aantal Planning Intervals per jaar: Tel de beschikbare werkweken, trek feestdagen en collectieve vrije perioden af, en bereken hoeveel volledige PI's (inclusief IP-iteratie) erin passen. Waar PI's over vakanties of feestdagen vallen, ontwerp je bewust aangepaste (extended) PI's in plaats van teams te dwingen in onrealistische iteraties. Leg een lijst vast met alle vakantiedagen en de weken waarop de cadence is ontworpen, zodat je hierop later kunt terugvallen bij fouten of discussies.

LPM-cadence: De LPM-cadence bestaat uit drie events die elk feitelijk op een eigen ritme lopen. De Portfolio Sync is maandelijks en heeft een operationele focus, terwijl de Strategic Portfolio Review elk kwartaal plaatsvindt om strategie en budget te herijken. Participatory Budgeting vindt twee keer per jaar plaats: frequent genoeg om wendbaar te blijven, maar stabiel genoeg om commitment mogelijk te maken. Samen vormen deze drie events het kloppende hart van portfoliobesturing binnen SAFe.

Stel samen de kwetsbare perioden vast: Kom tot consensus om geen PI-event of Inspect & Adapt te plannen vóór de derde week van januari, ná 20 december, of tijdens de zomervakantie. Energie en beschikbaarheid zijn in die perioden te laag voor ceremonies van deze omvang. Pas de cadence aan op dit besluit.

Bepaal de lengte van de IP-iteratie: De Innovation & Planning-iteratie kent conform SAFe een duidelijke tweedeling: week 1 is voor teamwerk (innovatie, technische schuld, backlog refinement, voorbereiding), week 2 is voor ceremonies. In de praktijk vinden veel organisaties dit te kostbaar, of de twee PI-planningsdagen te lang. Ik heb van alles voorbij zien komen. Stel de enterprise-brede lengte van de IP-iteratie gezamenlijk vast.

Inspect & Adapt is 3 à 4 uur, PI Planning is 2 dagen: Hoewel dit in principe vaste, niet-onderhandelbare doorlooptijden zijn, wordt er in de praktijk toch van afgeweken. Leg op enterprise-niveau wellicht de doorlooptijd van de I&A vast maar zeker die van het PI Planning-event. Bepaal daarbij ook of ART's hiervan mogen afwijken, en zo ja, onder welke voorwaarden.

Solution Train cadence: Wanneer meerdere ART's bijdragen aan dezelfde solution of binnen hetzelfde portfolio opereren, moeten hun PI-grenzen synchroon lopen. Dit is een voorwaarde voor effectief afhankelijkheidsbeheer en betekenisvolle gezamenlijke demo's.

Beschikbaarheid van schaarse rollen: Executive Management en Business Owners zijn niet de enige schaarse rollen binnen een enterprise. Denk ook aan Enterprise Architect, Security, Finance controllers etc. Een fundamentele ontwerpvraag luidt: Kunnen deze rollen inhoudelijk deelnemen aan meerdere ART-ceremonies zonder structurele overbelasting? Indien het antwoord nee is, moet de cadence worden herontworpen. Een ritme dat afhankelijk is van structurele overbelasting is geen stabiel operating system.

PI Planning-dagen optimaliseren naar beschikbare ruimte: PI Planning is een groot fysiek event. Bij meerdere ART's die tegelijk plannen (wat kan oplopen van 35 tot 60+ ART's parallel), is de beschikbaarheid van geschikte locaties en zalen een harde randvoorwaarde die in het cadence-ontwerp moet worden meegenomen. Overweeg in het ontwerp vast te leggen welke ART op maandag en dinsdag plant, en welke op woensdag en donderdag.

Stel een PI- en iteratie-naamgevingsconventie op: Bij organisaties met meerdere portfolio's is het handig de PI-naam te voorzien van een korte portfolio-prefix, bijvoorbeeld HY-PI25.4.1 voor het Hypotheken-portfolio en FIN-PI25.4.1 voor het Finance-portfolio. Dit voorkomt naambotsingen in gedeelde tooling en maakt portfolio-overstijgende rapportage direct leesbaar. PI25.4.1 staat voor de vierde PI in jaar 25, teamiteratie nummer 1.

Inventariseer bestaande legacy events: Breng alle bestaande meetings en vergaderstructuren in kaart en beoordeel per event: afschaffen, samenvoegen met een SAFe-ceremonie, of vervangen. De nieuwe cadence komt niet bovenop het bestaande. Ze vervangt het. Dit is Principle #6 (Remediate Legacy Policies) in de praktijk. Maak een lijst van die events en leg het besluit per event vast.

Externe leveranciers en contractritme: In enterprise-omgevingen spelen contractuele verplichtingen een belangrijke rol. Leveranciers hanteren vaak eigen kwartaalritmes, facturatiemomenten en milestone-afspraken. Ontwerpvraag: Sluit de SAFe-cadence aan bij contractuele verplichtingen, of worden contracten aangepast aan de Lean-Agile werkwijze? Wanneer deze afstemming niet expliciet wordt gemaakt, ontstaat spanning tussen LPM en juridische realiteit. Cadence moet ook juridisch uitvoerbaar zijn.

Stap 3: bepaal de ontwerpkeuzes op Agile Team-niveau

Op teamniveau worden de ontwerpkeuzes vertaald naar de dagelijkse werkpraktijk.

Vast team, vaste agenda, vast tijdstip, vaste locatie, vaste tooling: Dit is het SAFe-fundament op teamniveau: rust, regelmaat en voorspelbaarheid. Elke afwijking moet bewust en beargumenteerd zijn. In het cadence-ontwerp kun je een lijst opstellen van alle events met bijbehorende deelnemers, locaties en tijdstippen, en deze voor twee jaar vastleggen. Dat geeft houvast.

Iteraties starten op? SAFe heeft aanbevelingen ontwikkeld voor het starttijdstip van iteraties op teamniveau. Start je agile teams op maandagmorgen om 8.30 uur met het Iteration Planning-event en sluit je af met de retrospective op vrijdag 16:00, of start je de iteratie op dinsdag of woensdag? Dit is een bewuste keuze die je vooraf moet maken.

Stel een expliciete eventlijst op per niveau: Bepaal bewust welke events op LPM-, ART-, team- en eventueel Solution Train-niveau standaard zijn en welke optioneel. Niet elk event dat SAFe beschrijft is in elke context nodig, maar de keuze moet expliciet zijn, niet impliciet. Deze lijst is ook belangrijk voor toekomstige audits: als agile teams geen formele SAFe-events uitvoeren terwijl dat wel zo is ontworpen, leidt dat tot een lagere score.

Ceremonies zoveel mogelijk in de ochtend: Door vaste meetings in de ochtend te plannen, creëer je ononderbroken blokken productieve tijd in de middag. Dit minimaliseert task-switching en maximaliseert de "Time in the Zone" die teams nodig hebben voor complex werk (Principle #6). Een uitgangspunt kan zijn dat agile teams

zelf bepalen hoe laat zij starten.

Maximaal één ad-hoc blok per week. Ongeplande meetings zijn de vijand van voorspelbaarheid. Reserveer één vast moment per week in je Cadence voor ad-hoc overleg en bescherm de rest van de week.

Teams houden vrijheidsgraden: Niet alles kan of moet op enterprise-niveau worden vastgelegd. Teams hebben speelruimte nodig om hun cadence aan te passen aan hun eigen realiteit: beschikbaarheid van teamleden, onsite/remote-balans, vrije dagen. Niet elk team zal het Iteration Planning-event conform het enterprise-ontwerp op dinsdagmorgen 8:30 kunnen uitvoeren.

% Overhead. Zodra de cadence definitief is ontworpen, is het raadzaam de procentuele event-overhead per iteratie nauwkeurig door te rekenen en vast te stellen, om de verhouding tussen vergadertijd en productietijd expliciet te maken.

Remote versus fysieke ceremonies: de vorm van ceremonies beïnvloedt energie, duur en event-overhead. Ontwerp expliciet:

• Volledig fysiek (verplicht vanuit je organisatie)

• Hybride

• Volledig remote

• 1+1 model (dag 1 fysiek, dag 2 remote)

Dit geldt met name voor PI Planning en Inspect & Adapt. Fysieke capaciteit, reisbelasting en concentratievermogen zijn harde randvoorwaarden in enterprise-ontwerp.

HR- en performance-cycli: Enterprise cadence raakt ook HR-processen zoals Beoordelingsgesprekken, POP-cyclus, KPI-jaarcyclus, Opleidingsplanning etc. Wanneer HR-cycli botsen met I&A of PI Planning, ontstaat energieverlies. Ontwerp daarom alignment tussen performance- en delivery-cycli.

Stap 4: werk de cadence uit in een gedetailleerd plan

De cadence moet nu per dag, per week en per maand in detail worden uitgewerkt, wat doorgaans in een xlsx-bestand gebeurt. Alle agilisten moeten exact weten op welke data welke events plaatsvinden. Ook alle PI-dagen moeten bekend zijn, zodat locaties tijdig kunnen worden gereserveerd.

Bereken daarbij de procentuele event-overhead voor twee iteratietypen: een normale iteratie en de IP-iteratie, die een aanzienlijk hogere event-overhead kent. Dit maakt de verhouding tussen vergadertijd en productietijd inzichtelijk en helpt bij het weerleggen van de veelgehoorde klacht dat SAFe te veel meetings heeft.

Conclusie

Een enterprise cadence is geen agenda-oefening. Het is een strategisch ontwerpbesluit en dient naleesbaar, schriftelijk te worden vastgelegd inclusief versiebeheer.

Wanneer cadence en requirements-model niet uniform, expliciet en enterprise-breed zijn ontworpen, ontstaat versnippering. ART’s gaan hun eigen ritmes draaien. Portfolio’s verschuiven onafhankelijk van elkaar. Executive stakeholders moeten kiezen tussen conflicterende PI-events. Rapportages worden onvergelijkbaar. Flow stokt. Een goed ontworpen enterprise cadence doet precies het tegenovergestelde:

• Het creëert voorspelbaarheid.

• Het minimaliseert wachttijd.

• Het beschermt "Time in the Zone".

• Het maakt synchronisatie mogelijk.

• Het maakt portfoliobesturing betekenisvol.

• Het voorkomt lokale optimalisatie.

• Het reduceert verborgen vergaderstructuren.

• Het maakt overhead expliciet en beheersbaar.

Cadence is daarmee geen operationeel detail, maar een architectuurkeuze van het operating system. Wie cadence enterprise-breed ontwerpt (gebaseerd op SAFe-principes, expliciete randvoorwaarden en consensus), legt een stabiele basis onder samenwerking, portfoliosturing en waarde-levering. Wie dat niet doet, ontwerpt impliciet chaos.

En in een enterprise-omgeving is impliciete chaos altijd duurder dan expliciete structuur.

Bijlage: referentieoverzicht SAFe-events per niveau

Dit overzicht dient als referentie bij het ontwerpen van de enterprise cadence. Niet elk event is in elke configuratie verplicht. De keuze welke events worden opgenomen is een expliciete ontwerpkeuze die in de eventlijst (zie Stap 3) wordt vastgelegd. De cadence-ontwerpcriteria in dit hoofdstuk verwijzen naar dit overzicht in plaats van alle events afzonderlijk te benoemen.

LPM-niveau events

• LPM Backlog Refinement: wekelijks of tweewekelijks, binnen iteratie

• Portfolio Sync: maandelijks

• Feature Storming event: frequentie afhankelijk van aantallen Epics (geen formeel event)

• Strategic Portfolio Review: kwartaal, minimaal 4-6 weken vóór volgende PI Planning

• Participatory Budgeting: halfjaarlijks

• LACE: (twee) wekelijks

• VMO: (twee) wekelijks

Solution Train-niveau events (alleen Large Solution / Full SAFe)

• Solution Train Backlog Refinement: periodiek binnen het PI, volgens de afgesproken enterprise cadence

• Pre-PI Planning: 1-2 dagen vóór ART PI Planning; alignment Solution Train met ARTs en suppliers

• Post-PI Planning: 1 dag ná ART PI Planning; integratie PI-plannen, risico-identificatie, Solution Train PI Objectives

• Solution Demo: per PI, einde PI; geïntegreerde demo van alle ARTs

• Solution Train Inspect & Adapt: per PI, gecombineerd met of aansluitend op ART I&A

ART-niveau events

• ART Backlog Refinement: wekelijks of tweewekelijks, binnen iteratie

• Prepare for PI planning: aantal dagen voor het PI event

• Feature Socializing Event (optional)

• PI Planning: per PI, 2 dagen

• Inspect & Adapt (I&A): per PI

• PI System Demo

• Kwantitatieve en kwalitatieve beoordeling

• Problem Solving Workshop

• System Demo: per iteratie of elke 2 iteraties

• ART Sync: wekelijks of 2x per week (optional)

• PO Sync: wekelijks of 2x per week

• Coach Sync: wekelijks of 2x per week

• Feature Storming event: frequentie afhankelijk van aantallen Epics (geen formeel event)

Team-niveau events

• Iteration Planning: per iteratie, begin iteratie

• Iteration Review / Team Demo: per iteratie, einde iteratie

• Iteration Retrospective: per iteratie, einde iteratie

• Team Sync: dagelijks, ~15 min

• Backlog Refinement: wekelijks of tweewekelijks, binnen iteratie

• Design Thinking events: mogelijk workshops om tot ‘Solutions en Prototypes te komen’

• <en uiteraard multi teamlevel meetings met de omgeving om tot je Increment te komen>

Inventarisatie legacy events op afdelingsniveau

• Inventariseer en stel vast wat hiermee gebeurt

• wat blijft

• wat wordt gemigreerd

• wat kan weg (ook in verband met de kosten en overhead)

Inventarisatie legacy events op enterpriseniveau

• Portfolio events, grote prioriteringsevents, enterprise rapportage events, etc

• wat blijft

• wat wordt gemigreerd

• wat kan weg (ook in verband met de kosten en overhead)

Deel 8: Metrics en kosten

56. Data Driven Improvements: sturen op de juiste metrics binnen je totale operating model

Het huidige MT en de impact op metrics

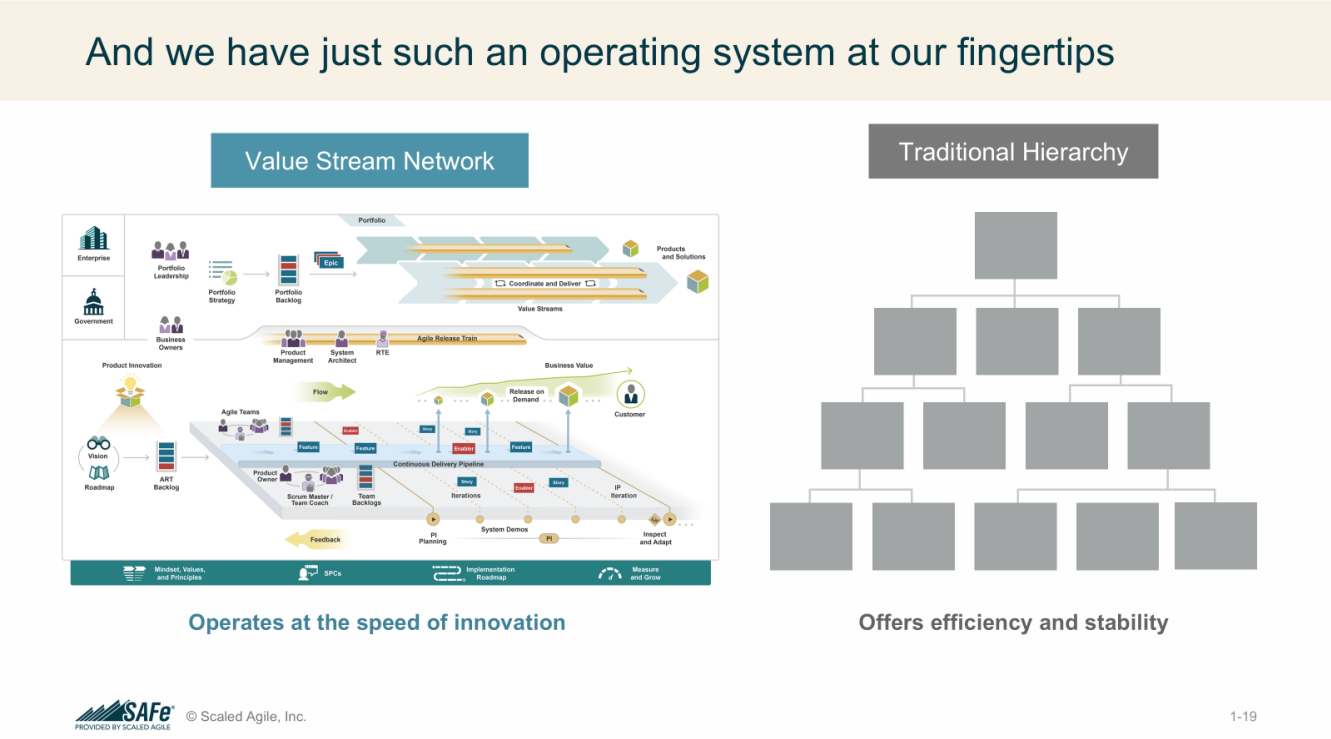
Introductie, doel van het hoofdstuk en doelgroep

Veel SAFe®-implementaties stranden niet op gebrek aan Agile kennis of motivatie, maar op een **gebrek aan inzicht**. Daarmee bedoel ik: organisaties werken volgens SAFe®, maar weten vaak niet precies **hoe hun totale operating model werkelijk presteert.** En met werkelijk bedoel ik aantoonbaar, reproduceerbaar en niet voor meerdere interpretaties vatbaar. Zonder een solide metrics-besturings-model blijft besturing grotendeels gebaseerd op aannames in plaats van fact based data.

Een belangrijke waarschuwing vooraf: wanneer SAFe® binnen de organisatie niet goed is ingericht, en met name het requirements-model, kunnen bepaalde metrics later niet of mogelijk nooit meer betrouwbaar uit de tooling worden gehaald. Metrics zijn geen dashboardvraagstuk, het is een SAFe systeemontwerpvraagstuk.

Dit hoofdstuk is geschreven voor practitioners, coaches en executive management die betrokken zijn bij het inrichten en besturen van een Lean-Agile organisatie, en die betrokken zijn bij het meetbaar maken van een SAFe®-organisatie en belang hebben bij die metrics om gericht te kunnen sturen.

SAFe® geeft in haar framework wel richtlijnen voor metrics, maar de harde praktijk is helaas iets weerbarstiger. Het doel van dit hoofdstuk is om jou een denkmodel te bieden met concrete handvatten en een set van voorbeeldmetrics en gezamenlijk na te denken welk orgaan overkoepelend monitort, audit en intervenieert.



Typische directievragen

Wat wil directie écht weten? Als je alle directievragen reduceert, komen ze neer op zes fundamentele behoeften:

1. **Vertrouwen:** Staat de voortbrengingsketen op groen? Kunnen we erop vertrouwen dat het systeem levert wat we beloofd hebben (predictability) ? Halen de ART's hun PI-objectives en milestones? Is de flow gezond, de predictability op orde? Is alles onder controle, of sturen we op hoop? Wat is de uitkomst van een assessment als we deze zouden initiëren?

2. **Risico:** Wat kan ons morgen verrassen? Wat weten we nog niet maar zouden we moeten weten? Wat is er initieel geschat en wat zijn de actuals? Welke Milestones staan op groen en welke niet, halen we de planningen? Welke dependencies (leveranciers?) hinderen flow?

3. **Waarde**: stoppen we ons geld en onze mensen op de goede plek? Wat levert het op? Welke Solutions zijn opgeleverd tegen welke totale kosten? Welke epics leveren daadwerkelijk waarde, welke hadden we beter niet kunnen doen? Hoeveel % is besteed aan Dev, hoeveel aan Ops, hoeveel aan Improvements en hoeveel aan Refactoring?

4. **Besluiten & acties:** Wat moeten we nú besluiten? Wat is de knoop die wij moeten doorhakken? Wat zijn de drie beslissingen die wij de komende 90 dagen moeten nemen?

5. **Vooruitkijken: zijn we toekomstbestendig? Zijn de juiste mensen beschikbaar om nieuwe ontwikkelingen en vragen op te vangen en aan de verwachte vraag te voldoen? Dit is zeker van belang voor uitvoeringsorganisaties.**

6. **Inzage:** Wat is de huidige configuratie van de fabriek: hoeveel 'solutions', aantal ART's, aantal teams, FTE's, verhouding Chiefs/Indians over de hele Fabriek, etc. Als er iets niet loopt, hoe snel vinden we dan de rootcause? Wordt het beter of wordt het slechter?

Zes vragen op enterpriseniveau. Hoeveel kun jij er beantwoorden: aantoonbaar, reproduceerbaar, zonder interpretatie met 1 druk op de knop?

DEEL I: HET METRICS-MODEL

1. Waarom meten?

Organisaties meten niet om dashboards te vullen, maar om beter te kunnen sturen en zodoende de einddoelen van een organisatie te behalen (missie, visie, OKR's). In een complex systeem zoals een SAFe®-organisatie zijn veel processen tegelijk actief: strategie, portfolio-besluiten, ontwikkeling, operatie en continue verbetering. Zonder goede metrics is het vrijwel onmogelijk om te begrijpen waar problemen ontstaan, waar de root-cause ligt of waar juist verbetering plaatsvindt.

SAFe® is van nature een **fact-based framework**. Dat is geen toeval, het is een bewuste ontwerpkeuze. Binnen SAFe® worden besturingsbesluiten niet genomen op basis van wandelganggesprekken, buikgevoelens of plausibele verhalen. De basis is altijd data: meetbaar, herhaalbaar en onweerlegbaar. Metrics zijn daarin geen bijzaak, ze zijn het instrument waarmee het framework zichzelf eerlijk houdt. Waar een mening aanvechtbaar is, is een getal dat niet. Dat maakt metrics in SAFe® niet alleen nuttig, maar noodzakelijk.

Metrics vervullen aanvullend drie belangrijke functies.

1. Ten eerste maken zij continue verbetering mogelijk. Teams en ART's kunnen hun werkwijze alleen verbeteren wanneer zij inzicht hebben in hun eigen prestaties.

2. Ten tweede maken metrics het mogelijk om tijdig te interveniëren. Wanneer bepaalde indicatoren verslechteren, kan de organisatie ingrijpen voordat problemen escaleren.

3. Ten derde ondersteunen metrics het bouwen van een duurzaam Lean-Agile systeem (OS #2), een van de hoofddoelen van elke SAFe®-transformatie. Het doel hiervan is immers niet afhankelijk blijven van externe SAFe® SPC's, consultants of ladingen agile coaches, maar een systeem creëren dat ook op lange termijn goed functioneert, ook wanneer deze groep op termijn wegvalt.

Voor werktuigbouwkundige ingenieurs zoals ik, maar ook voor het management geldt de eenvoudige regel: **Meten is weten, gokken is dokken, gissen is missen.** Zonder betrouwbare metrics blijft organisatiebesturing grotendeels gebaseerd op aannames.

Tip: Mijn advies is te starten met een beperkte MVP-set van metrics die snel inzicht geeft in de gezondheid van het systeem. In de vroege fase van een SAFe-transformatie ligt de nadruk daarom vooral op leading indicators, je wilt snel weten of je ontwerp loopt. Probeer niet meteen een volledig metrics-landschap te bouwen. Dat mislukt in de praktijk vrijwel altijd. Een klein dashboard met een paar goed gekozen indicatoren werkt vaak beter dan een uitgebreid model met tientallen metrics.

2. Het metrics-landschap

Het metrics-model van een SAFe®-ingerichte-organisatie is **gelaagd en multidimensionaal**. Metrics bestaan niet als één lijst met indicatoren, maar ontstaan op verschillende niveaus, in verschillende categorieën en met verschillende doelen. Dat maakt het onderwerp complex, maar ook krachtig: wanneer deze dimensies goed gecombineerd worden ontstaat een compleet beeld van hoe het systeem werkelijk functioneert.

Ik werk dit uit aan de hand van onderstaande vijf dimensies:

1. Organisaties kennen na een SAFe implementatie **twee operating systems**. Het hiërarchische systeem (OS #1) richt zich op traditionele bedrijfsvoering, zoals HR, finance en compliance. Het Lean-Agile netwerk (OS #2) richt zich op waardecreatie via ART's en teams. Beide systemen genereren hun eigen soorten metrics.

2. Binnen de hele organisatie worden meestal meerdere typen metrics onderscheiden (gebaseerd op SAFe® V6.0.2 en gangbare metrics-theorieën)

1. **Outcome-metrics** meten het uiteindelijke resultaat voor klanten of gebruikers.

2. **Flow-metrics** meten hoe snel waarde door het systeem beweegt.

3. **Grow the Business metrics** (Dev prestaties).

4. **Run the business metrics** (Ops).

5. **Competency-metrics** geven inzicht in de volwassenheid van The Value Stream Network.

6. **Improve-the-Business** (Continuous Improvement) metrics laten zien of de organisatie daadwerkelijk leert en verbetert.

7. **Financiële metrics** maken tenslotte zichtbaar welke economische waarde wordt gerealiseerd.

3. **Elke laag in SAFe heeft zijn eigen perspectief** op prestaties en waarde. Indicatoren hebben betrekking op individuele teams, op een volledige Agile Release Train (ART), op een Solution Train of op het portfolio-niveau binnen Lean Portfolio Management (LPM).

4. Elke laag kent het onderscheid tussen **leading en lagging indicators**. Lagging indicators beschrijven resultaten die al gerealiseerd zijn, zoals klanttevredenheid of gerealiseerde business value. Leading indicators daarentegen geven een vroeg signaal over toekomstige prestaties. Flow-metrics zijn bijvoorbeeld vaak leading indicators: wanneer flow verslechtert, zal business performance later vrijwel altijd volgen. Met name tijdens start van je agile transformatie wil je sterk sturen op je leading indicators.

5. Een andere belangrijke dimensie is het verschil tussen **MVP-metrics en Full Solutions metrics**. Organisaties die net starten met SAFe® proberen vaak meteen een compleet metrics-landschap te bouwen. In de praktijk werkt het beter om tijdens de SAFe transformatie te beginnen met een beperkte set indicatoren die snel inzicht geven in de gezondheid van het systeem. Naarmate de organisatie volwassener wordt, kan deze set geleidelijk worden uitgebreid tot het punt dat 'SAFe business as Usual wordt'. Dan kunnen de transitie metrics worden gemigreerd naar operationele metrics.

Samen vormen deze dimensies het complete **metrics-landschap van een SAFe®-organisatie**. In de volgende secties werken we uit hoe deze dimensies per SAFe®-laag concreet worden ingevuld en dan onderverdeeld conform bovenstaande dimensies.

3. Databronnen, dashboarding, meetarchitectuur en vastlegging

De informatie die nodig is om een SAFe®-organisatie goed te sturen, zit vrijwel altijd verspreid over meerdere systemen. Er bestaat geen enkele tool die alle relevante metrics direct en volledig ontsluit, en die zal er ook niet snel komen. Agile teams registreren hun werk in verschillende typen enterprise tooling, DevOps-pipelines genereren hun eigen data, en financiële en HR-informatie leeft in weer andere systemen. Wie echt wil sturen op metrics, moet deze bronnen daarom actief combineren. Een centrale BI-laag (bijvoorbeeld Power BI of Tableau) brengt de data via API-koppelingen samen in één managementdashboard.

In de praktijk komen metrics daarbij doorgaans uit een combinatie van de volgende bronnen:

1. **Enterprise Agile tooling**: Jira Align, Planview, TargetProcess, etc.

2. **Agile tooling**: Jira, Rally, ADO, etc.

3. **ServiceNow**: ITIL, releases, incidenten, bugs, etc.

4. **DevOps tooling**: pipelines, deployments, test coverage

5. **Ops tooling**: monitoring, availability, performance, SLA-dashboards

6. **SAP of ERP-systemen**: vaak de financiële data

7. **HRM**: inzet, ziekte, uitval, FTE-metrics

8. **Assessments, surveys en retrospectives**: zoals o.a. de formele SAFe® assessments, specifieke survey tooling, etc

9. **Overige enterprise tooling**

Een goed ingericht dashboard toont daarom niet alleen OS #2 (het Value Stream Network), maar ook de relatie met de traditionele organisatie, zijnde OS #1 (**de hiërarchische organisatie**).

In de praktijk beginnen getransformeerde organisaties meestal met een relatief **beperkte set van metrics**: een MVP-dashboard dat snel inzicht geeft in de gezondheid van het systeem. Naarmate de organisatie meer ervaring opdoet met SAFe® en beter begrijpt welke signalen belangrijk zijn, groeit het dashboard geleidelijk uit tot een rijker en volledig beeld van de organisatie met de Full Solution Metrics.

DEEL II: METEN EN MONITOREN VAN HET GROTE GEHEEL

Nu het metrics-landschap en de onderliggende architectuur helder zijn, gaan we concreet kijken wat je per operating system en per SAFe® laag meet en monitort. Maar voordat we dat doen, is er een fundamentele vraag die je eerst moet beantwoorden: wat meet je eigenlijk in jouw organisatie? Het antwoord is direct afhankelijk van hoe je Target Operating Model (TOM) is ontworpen. Welke activiteiten, processen en verantwoordelijkheden zijn belegd in OS #1 (de hiërarchische organisatie) en welke in OS #2 (het Lean-Agile value network)? En misschien nog belangrijker: wat is het demarcatiegebied daartussen? In de praktijk is dat grensgebied zelden scherp, en juist daar ontstaan de meeste meet- en sturingsproblemen. De keuze van metrics is dus geen technische vraag, maar een ontwerpvraag. Wie is eigenaar van welk proces? Wie rapporteert aan wie? En waar overlappen OS #1 en OS #2? Pas als die vragen beantwoord zijn, kun je een metrics-model bouwen dat de werkelijkheid van jouw organisatie weerspiegelt. In dit deel starten we met de OS #1-metrics, omdat die in de meeste organisaties al bestaan, zij het verspreid over verschillende systemen en rapporten. Daarna werken we de OS #2-metrics uit per SAFe®-laag.

1. “De hiërarchie” metrics (OS #1)

Naast OS #2 blijft in vrijwel elke organisatie ook het traditionele managementsysteem bestaan. Binnen dit OS #1-domein worden andere soorten prestaties gemeten dan in het Lean-Agile netwerk: niet beter of slechter, maar anders van aard en gericht op andere sturingsvragen. De lijnorganisatie en de verhouding tussen OS #1 en OS #2 worden verder uitgewerkt in hoofdstuk 33, ‘Leiderschap in SAFe: schakelen tussen hiërarchie en waardestroomnetwerk’, en hoofdstuk 34, ‘Lijnmanager én gelijkwaardige collega? Waarom SAFe duidelijke rolgrenzen vraagt’. Hieronder volgt een concrete set van potentiële OS #1-metrics als vertrekpunt. Ik splits deze metrics bewust niet op in MVP, leading en lagging. OS #1 bestaat in de meeste organisaties al jaren, en veel van deze metrics worden al geruime tijd gemeten en gerapporteerd. De opgave hier is dan ook geen nieuwe meting te ontwerpen, maar te inventariseren wat er al is, en vervolgens te bepalen welke bestaande signalen relevant genoeg zijn om te combineren met de OS #2-metrics in één integraal bestuurlijk overzicht. Want juist in het grijze gebied tussen deze twee systemen ontstaan de meeste meet- en sturingsproblemen. Wie dat grensgebied niet expliciet adresseert, meet veel, maar stuurt weinig.

Voorbeeld metrics HRM	**From score**	**To score**
eNPS score	De IST state	De gewenste SOLL state.
Medewerkersverloop (turnover rate) organisatie	-	-
Teamlid verloop (turnover agile teams)	-	-
Ziekteverzuim en duurzame inzetbaarheid	-	-
Lead/cycle time levering resources naar OS #2	-	-
Matching Score OS #1 naar OS #2	-	-
Compliance-score (bijv. AVG, ISO, BA)	-	-
%certified SAFe agilisten	-	-
% medewerkers met actief leerplan	-	-
Actueel overzicht van aanvragen OS #2 en status per aanvraag.	-	-
%succesvol gevuld per SAFe level	-	-

Voorbeeld metric out of scope OS #2	**From score**	**To score**
Wat valt er nog binnen OS #1 en gaat niet meer naar OS #2?		
<bepaal samen met de organisatie>		

Wellicht financial metrics	**From score**	**To score**
Wie beheert de financiën, OS #1 of OS #2?		
Budget vs. Actuals per ART	-	-
Capex/Opex-verhouding	-	-

Hoewel waarschijnlijk de meeste metrics vaak buiten de Enterprise Agile Tooling ontstaan, zijn ze wel degelijk relevant voor het functioneren van **het totale systeem**. In een volwassen LACE/organisatie worden daarom OS #1 - en OS #2-metrics gecombineerd in één bestuurlijk overzicht c.q. management dashboard.

2. “The Value Stream Network” metrics (OS #2)

Binnen het Lean-Agile value network (OS #2) worden metrics meestal bekeken en gemanaged per SAFe®-laag. Ook, omdat op elke laag een ‘proces owner’ zit die verantwoordelijk is voor het goed laten functioneren van die laag. Elke laag heeft daarnaast een ander perspectief op prestaties en waarde. Bovendien verschilt de volwassenheid van de metrics-set in de tijd. Veel organisaties starten met een beperkte set MVP-metrics die snel inzicht geven (bijvoorbeeld na launchen van de eerste ART), en breiden deze later uit met een vollediger metrics-model wanneer het systeem stabiel draait.

Competency enables Flow, and Flow produces Outcomes.

2.1 Competence metrics set

Competence metrics meten de bekwaamheid van teams en individuen: beschikken we over de juiste kennis, vaardigheden en werkwijzen om waarde te leveren? Waar flow metrics (zie onder) laten zien *hoe soepel* werk door het systeem beweegt en outcome metrics meten *wat* dat oplevert voor klant en organisatie, richt de competence set zich op *wie* dat werk uitvoert, en of zij daarvoor toegerust zijn. Denk aan technische volwassenheid, agile volwassenheid en continue leerbereidheid als fundamenten onder duurzame prestaties. Tip, Scaled Agile biedt al out-of-the-box een volledige set van Metrics aan om de Competence op alle levels van SAFe te meten.

De metrics	From score	To score
Het overall “SAFe Business Agility Value Stream” radar	**De IST state**	**De gewenste SOLL state.**
Tal van andere SAFe assessments, per core competency / disciplines	-	-
%teams dat werkt met DoD, DoR en retrospectives	-	-
Beschikken over een Agile Mindset	-	-
Passen aantoonbaar SAFe’s best-practices toe	-	-
Forecasted versus Actuals	-	-
etc		

Pas deze voorbeeldset aan op de situatie en afspraken binnen je organisatie.

2.2. Concrete LPM level metrics set.

Het LPM-niveau stuurt op de gezondheid van een specifiek portfolio binnen mogelijk een set van portfolios. Stroomt waarde snel genoeg door het systeem, worden de juiste Epics gekozen en gerealiseerd, en draagt het portfolio aantoonbaar bij aan de strategische doelen van de organisatie? Op dit niveau kijkt de LACE zowel naar leading indicators die vroeg signaleren of het systeem goed functioneert, als naar lagging indicators die achteraf laten zien of de verwachte waarde ook werkelijk is gerealiseerd.

MVP metrics	De metrics	From score	To score
Leading	Epic cycle time	Gem. Epic doorlooptijd > 8 PI’s	Gem. Epic doorlooptijd is ~ 4 PI
	Portfolio WIP (Epics in progress)	53 actieve Epics gelijktijdig	≤ 15 actieve Epics gelijktijdig
	Epic Hypothese Validatiesnelheid	< 20% van Epics heeft EHS gevalideerd binnen 1 PI	≥ 70% van Epics heeft MVP-hypothese gevalideerd binnen 1 PI

MVP metrics	De metrics	From score	To score
	Flow Distribution	> 70% focus op nieuwe features	Gebalanceerd portfolio (bijv. 40% business / 30% enablers / 20% tech debt / 10% innovation)
	Portfolio Flow Balance (CFD stability)	CFD toont structurele WIP opstapeling in Analyse / Review / Approval stages	CFD bands blijven stabiel en ongeveer parallel
	Orphan Feature Ratio (Features zonder Epic / Strategic Theme)	> 25% van Features niet gekoppeld aan Epic of Strategic Theme	≤ 3% van Features zonder Epic link
	Schattingsnauwkeurigheid (Forecast/actuals)		Afwijking +/- 15%
	Innovatie adaptiegraad	<bepaal samen met de organisatie>	<bepaal samen met de organisatie>
	Portfolio Decision Lead Time	*Besluitvorming > 8 weken*	*Besluitvorming ≤ 2 weken (syncs)*

Lagging (outcomes)	Portfolio Predictability (realisatie vermogen)	< 50% van Epics opgeleverd conform scope & portfolio roadmap	≥ 80% van Epics opgeleverd conform scope & planning
	Epic Success Rate	< 30% van de Epics die beloofde business hypothesis realiseren	≥ 70% van de Epics die beloofde business hypothesis realiseren
♪	Customer Satisfaction (CSAT)	3	4.7
♪	Strategic Theme Realisatiegraad (OKRs)	< 40% van strategische thema's behaald	≥ 75% van strategische thema's behaald
♪	Time-to-Market (Idea → Production Lead Time)	Nieuwe capability > 24 maanden	Nieuwe capability ≤ 6 maanden
♪	Complete Capability Delivery Rate	Gedeeltelijk, delen missen	Volledig (NFR's, Interops, Usability, documentatie, O&T, Support, onderhoudbaar, operationele implementatie mogelijk)

Pas deze voorbeeldset aan op de situatie en afspraken binnen je organisatie.

*Bepaal zelf of je met een kleinere set wilt werken of hier mee start

2.3. Concrete ART level metrics set.

Het ART-niveau stuurt op de gezondheid van de Agile Release Train als geheel: levert de ART betrouwbaar en frequent waarde op per PI, stroomt werk soepel door het systeem, en werken teams effectief samen richting gedeelde PI-doelen? Op dit niveau kijkt de LACE naar signalen die vroeg aangeven of alle ART's in OS #2 goed functioneren, en naar resultaten die laten zien of de beloofde waarde ook daadwerkelijk is opgeleverd. Omdat SAFe®-teams steeds vaker end-to-end verantwoordelijk zijn voor zowel ontwikkeling als operations (DevOps), bevat de set ART-metrics ook de voornaamste DevOps- en flow-indicatoren. Organisaties waarbij Ops nog deels centraal is belegd, parkeren de SLA- en availability-metrics voorlopig in de OS #1-tabel. (dit moet in je TOM zijn gedefinieerd)

MVP metrics voorbeelden om mee te starten, te leren en te wennen

ART MVP metrics	De metrics	From score	To score
Leading	Flow Time	Gem. Feature doorlooptijd > 2 PI's	90% heeft Gem. Feature doorlooptijd ≤ 1 PI
	Flow Load / WIP	15 Features tegelijk in progress per ART	≤ 6 tot 8 Features in progress (WIP-limiet actief)
	Flow Distribution	80% Features, tech debt en enablers onzichtbaar	~60% Features / ~20% Enablers / ~20% Ops+Debt (bewust en zichtbaar verdeeld)
(meet op ART niveau)	Pull Ratio (Pulled Story Points / beschikbare Story Points-capaciteit)	< 60% van beschikbare SP capaciteit wordt gepulled	~85% van capaciteit wordt van backlog gepulled
	Dependency Resolution Time	>2 iteraties	≤1 iteratie
Lagging	PI Predictability (% PI Objectives behaald)	< 50% van PI Objectives behaald	≥ 85% behaald (incl. uncommitted objectives)
	Flow Velocity	< 10 Features opgeleverd per PI	≥ 20 Features per PI, stabiel en voorspelbaar
	Defect Escape Rate	>15% defects na release	≤5%
Continuous Improvements	Aantoonbare improvement I&A-Problem-solving workshop		
	<bepaal samen met de organisatie>	-	-

Uit te breiden met deze metrics in geval van Full Solution SAFe®

ART Full Solutions metrics	De metrics	From score	To score
Leading	Flow Efficiency	<20%	≥40%
	ART Flow Balance (CFD stability)	Bottlenecks, ophopingen	Stabiele CFD
Lagging	<bepaal samen met de organisatie>		
	<bepaal samen met de organisatie>		
Continuous Improvements	zie onder.		

Pas deze voorbeeldset aan op de situatie en afspraken binnen je organisatie.

2.4. Concrete Agile Team level metrics set.

Het team-niveau stuurt op de dagelijkse gezondheid van het agile team zelf: werkt het team in een duurzaam ritme, is de kwaliteit van het geleverde werk op orde, en voert het team continu verbeteringen door? In SAFe® zijn agile teams idealiter end-to-end DevOps-teams: zij bouwen én beheren wat ze opleveren. Dat betekent dat Dev- én Ops-metrics hier samenkomen in één teamverantwoordelijkheid. De Scrum Master en Team Coach zijn de primaire gebruikers van deze metrics; zij signaleren patronen en initiëren verbetergesprekken. Organisaties waarbij Run-the-Business nog centraal is belegd, plaatsen de SLA- en incidentmetrics in de OS #1-tabel totdat het team die verantwoordelijkheid volledig heeft overgenomen.

MVP metrics voorbeelden om mee te starten, te leren en te wennen

Team MVP metrics	Domain	De metrics	From score	To score
Misc	Outcome	Customer Satisfaction (CSAT)	3	4.7
	HRM	Teamstabiliteit / Team Happiness score	-	-
	HRM	Team Utilisation rate (pulled cap/beschikbare cap)		
	Tooling	Data Quality / Jira Align Data Freshness (Weekly Update Compliance)	nvt	≥95% teams hebben Jira Align vóór vrijdag 17:00 bijgewerkt
Leading	Dev	Flow Predictability	Geleverd vs. gepland > 30% afwijking per sprint	≤ 10% afwijking geleverd vs. gepland per sprint
		Suppliers leveren conform Dependencies	Onbekend	90% conform PI dependency commitment
		% niet afgemaakte stories naar de volgende iteratie	<bepaal samen met de organisatie>	< 5%
		#nr of defects		
	Ops	Automated Test Pass Rate		
		Deployment Frequency (DORA)	< 1x per sprint naar productie	≥ 1x per sprint (streven: on-demand)
		Change Failure Rate (DORA)	15% van deploys leidt tot incident of rollback	< 5% van deploys leidt tot incident of rollback
		Servicedesk responsetimes, SLA's etc	X	Y
Lagging	Dev	Sprint Velocity Stabiliteit	Velocity varieert > 40% tussen sprints	Velocity stabiel ≤ 10% variatie over 3 sprints
		Test Automation Coverage	< 30% van codebase gedekt door geautomatiseerde tests	≥ 80% van codebase gedekt (unit + integration tests)
		Tech Debt Ratio	30% van sprint-capaciteit gaat naar bug fixes en tech debt	≤ 10% van capaciteit naar debt (structureel beheerst)
	Ops	MTTR (Mean Time To Restore, DORA)	Gem. hersteltijd na incident > 4 uur	Gem. hersteltijd ≤ 1 uur (P1/P2 incidenten)
		Service Availability (%)	< 99% uptime (onafgesproken uitval)	≥ 99.9% uptime conform SLA-afspraken
Effectiviteit Retrospective	Een bepaalde metric is aantoonbaar verbeterd			
	Retro Action Completion Rate	% van retro-acties daadwerkelijk		

		geïmplementeerd vóór volgende retro		
	Improvement Story Ratio	% van sprint-capaciteit besteed aan improvement stories		

Pas deze voorbeeldset aan op de situatie en afspraken binnen je organisatie.

Uit te breiden met deze metrics in geval van Full Solution SAFe®

Team Full Solution metrics	Domain	De metrics	From score	To score
Leading	Dev	Flow Efficiency	< 20% actieve tijd; > 80% wachttijd in flow	≥ 40% actieve tijd (flow efficiency ratio)
	Ops			
Lagging	Dev			
	Ops			
Continuous Improvements				

DEEL III: VAN METEN NAAR VERBETEREN

Wie monitort, stuurt en intervenieert op het grote geheel?

Tijdens een transformatie draait een organisatie tijdelijk op twee operating systems tegelijk, en dat is nadrukkelijk nog geen *steady state*. Voor die situatie maar ook de situatie erna ('Business as Usual'), moeten we een governance model bouwen.

OS1, de traditionele hiërarchie, moet blijven functioneren want die betaalt de rekeningen en voert HRM uit. OS2, het Value Stream Network, moet groeien want dat is de toekomst. In een Full Solution SAFe configuratie betekent dat: de LPM laag, de Solution Train laag, de ARTs, alle agile teams, system teams en shared service centres. Beide systemen vragen om eigen metrics, maar iemand, *een orgaan,* moet die complete metrics-set samen bekijken, systeemdenken toepassen, interpreteren en ingrijpen als het fout gaat.

Metrics die niemand leest zijn decoratie. Metrics die wel gelezen worden maar niet leiden tot een besluit of interventie zijn even nutteloos. Het grootste gevaar van een goed ingericht metrics-model is dus niet dat het ontbreekt, maar dat het wél bestaat, netjes op een dashboard staat, en vervolgens geen aanleiding geeft tot actie. **Meten zonder handelen is energie verspillen**. Daarom is de vraag niet alleen wát je meet, maar wie er verantwoordelijk is om er iets mee te doen, en wat de consequentie is als dat niet gebeurt.

SAFe kent drie organen die elk een deel van dit werk doen.

1. **Het Portfolio Leadership Team** is het beslissende orgaan op LPM-niveau: het autoriseert Epics, stelt prioriteiten in de portfolio backlog en bewaakt de alignment met de strategische thema's.

2. **De LACE** ontwikkelt het Target Operating model, begeleidt de transformatie, coacht, traint en bouwt competenties op.

3. **De VMO**, als die er al is, verzamelt metrics, maakt flow zichtbaar en rapporteert richting de beslissende partij, maar heeft geen mandaat om in te grijpen.

In de praktijk zien we dat OS1-managers kijken naar hun KPI's, Agile coaches kijken naar flow metrics, en het systemische verband tussen beide blijft onzichtbaar. Bovendien is LPM tijdens een transformatie zelf onderwerp van verandering. Je kunt niet tegelijk rechter en verdachte zijn.

Wat er **eigenlijk** nodig is, is een **vierde onafhankelijke functie**, bijvoorbeeld een Transformation Governance Board. Dit orgaan rapporteert niet aan Portfolio Leadership maar direct aan de CEO. Het wordt gevoed door VMO-data, LACE-observaties én OS1-KPI's. Het bezit SAFe experts, Metrics experts, lokale organisatie experts met systeemdenken als kerncompetentie: het ziet dwarsverbanden tussen alle lagen, van LPM tot aan de agile teams. Het is politiek onafhankelijk omdat zijn mandaat van buiten de portfoliohiërarchie komt en boven de OS-en staat. En het is bevoegd om te escaleren naar de CEO wanneer een laag, inclusief LPM zelf, de transformatie blokkeert.

Cruciaal is dat dit orgaan zijn gezag niet ontleent aan positie of autoriteit, maar aan kennis en data. Het moet in OS1-taal én in OS2-taal kunnen spreken. Het moet kunnen zeggen: de cijfers tonen dit, en iedereen in de kamer weet dat het klopt.

De CEO, of een high level executive, is de enige die dit construct kan installeren én beschermen. Zonder expliciete executive-sponsorship is een Transformation Governance Board tandeloos, en dan strandt de transformatie precies op het punt waar de meeste transformaties stranden: niemand die het grote geheel bewaakt.

Tip: Er bestaat geen standaard SAFe antwoord op de vraag wie de metrics governance bewaakt. Bespreek dit expliciet binnen je organisatie, selecteer een orgaan, en leg het vast in je TOM. Bespreek ook de rol van huidige OS #1 MT leden in relatie tot dit high level Governance board.

Interventies per probleemtype

Wanneer metrics aantonen dat het systeem niet optimaal functioneert, moet de organisatie bepalen **welk type probleem zich voordoet**.

• Bij **flow-problemen** ligt de oorzaak vaak in te veel werk tegelijk, slecht dependency management, te grote werkpakketten, externe interrupts of wachtrijen tussen teams. Interventies richten zich dan bijvoorbeeld op het verlagen van WIP, het verkleinen van batch sizes of het oplossen van bottlenecks.

• Bij **competency-problemen** ligt de oorzaak meestal in kennis, kunnen, willen, leiderschap of samenwerking. In dat geval kunnen interventies bestaan uit training, DevOps-verbeteringen, leiderschapsontwikkeling of het versterken van T-shaped teams.

• Een derde categorie betreft **Improve-the-Business problemen**. Veel organisaties hebben geen gebrek aan verbeterideeën, maar wel aan aantoonbare opvolging. Daarom meten sommige organisaties bijvoorbeeld hoeveel retro-acties daadwerkelijk worden afgerond, hoeveel improvement stories er naast feature-werk worden uitgevoerd en hoeveel structurele bottlenecks per PI worden opgelost.

De **IP-iteration** wordt in veel SAFe®-omgevingen gebruikt om hier expliciet ruimte voor te creëren. Daar kan bijvoorbeeld worden bekeken hoeveel capaciteit wordt besteed aan improvement, innovatie of training.

57. Een rekenmodel voor de kosten van de SAFe fabriek

Introductie

Organisaties die SAFe® invoeren, stellen vroeg of laat de vraag: wat kost dit eigenlijk ten opzichte van wat we opgeleverd krijgen aan Waarde? Het antwoord is in een SAFe wereld relatief makkelijk te geven, daar Agile Teams Long-Lived zijn.

Dit hoofdstuk presenteert een praktisch rekenmodel waarmee je de structurele DevOps kosten van een SAFe®-fabriek in kaart brengt, van teamniveau tot portfolio. Het geeft je direct een goed beeld ‘wat alles exact kost’

Mijn rekenmodel maakt onderscheid tussen twee perspectieven:

1. De lopende **operationele kosten van een full SAFe configuratie** per iteratie en per maand.

2. De gemiddelde **investeringsomvang van een Epic** van idee tot realisatie

Beide zijn essentieel voor een eerlijk businessgesprek over wendbaarheid.

Waarom dit rekenmodel interessant is

Veel organisaties die SAFe implementeren komen uit een traditionele watervalomgeving waarin IT-werk wordt georganiseerd via projecten. In zo’n model wordt werk gefinancierd per project: elk project heeft zijn eigen budget, stuurgroep, projectmanager, eigen PMO, governance en rapportagestructuur.

Wanneer organisaties willen begrijpen wat hun IT eigenlijk kost, tellen zij alle projectbudgetten bij elkaar op. Dat lijkt logisch, maar geeft zelden een volledig beeld. Want zit alle overhead van stuurgroepen, coördinatoren, etc, daar ook in? Deze kosten zijn vaak lastig volledig inzichtelijk te maken.

In een SAFe-organisatie werkt dit fundamenteel anders: SAFe werkt met **Long-Lived Agile teams en - Agile Release Trains** die samen een **stabiele, voorspelbare delivery-organisatie vormen**, vaak aangeduid als een *factory*. Omdat deze teams structureel bemenst blijven, zijn de kosten van deze fabriek relatief eenvoudig te bepalen. Het aantal teams, de bezetting en de kosten per FTE liggen immers grotendeels vast. Daarmee ontstaat een interessant economisch perspectief.

Wanneer organisaties de kosten van hun SAFe-fabriek kennen, en deze vergelijken met de kosten van hun Lean Portfolio Management (LPM) laag, ontstaat een veel interessantere vraag dan alleen “wat kost SAFe?”: Welke fabriek levert uiteindelijk de meeste waarde voor de funding die erin gaat?

Met andere woorden: uit welke kraan komt uiteindelijk het meeste water?

Dit hoofdstuk helpt om in ieder geval **de kosten van de SAFe delivery-fabriek inzichtelijk te maken**. De waterfall kosten moeten organisaties zelf bepalen. Pas wanneer beide perspectieven bekend zijn, ontstaat een eerlijk gesprek over de verhouding tussen **waterfall en SAFe (alleen the Value Stream Network)**, en over de werkelijke economische efficiëntie van de organisatie.

Uitgangspunten kosten SAFe fabriek

Algemene aannames

• We rekenen met iteraties van 2 weken

• We rekenen met Nederlandse kosten (dus geen outsourcing, near sourcing etc)

• Er wordt gerekend met **11 werkmaanden per jaar** (vakantie en feestdagen buiten beschouwing)

• Alle rollen worden **dedicated ingevuld**: geen parttime of gedeelde bezetting in verband met eenvoud rekenmodel

• Kosten zijn gemiddeld all-in genomen, dus incl auto, laptop, externen en internen.

Structuur van de SAFe®-organisatie

Het model volgt de vijf niveaus van de SAFe®-configuratie: **Teamlid → Team → ART → Solution Train → Portfolio/LPM**. Elk niveau heeft eigen rollen en een eigen kosten-footprint. De inputs zijn variabel: je past het aantal teams, ART's en Solution Trains aan op je eigen organisatiegrootte.

Totale kosten van een full SAFe configuratie voor een Large Enterprise

Laag 1: kosten per FTE per iteratie

Een Agile-team bestaat uit ongeveer **10 teamleden,** conform de SAFe theorie. Laten we uitgaan van deze kosten per teamlid

Parameter	Waarde
gemiddelde kosten per teamlid	€10.000 per maand
gemiddelde kosten per teamlid	€5.000 per iteratie van 2 weken

Laag 2: Agile team kosten per iteratie

Een Agile-team bestaat uit gemiddeld **8 teamleden** plus **2 FTE** voor de Product Owner en Scrum Master.

Parameter	Waarde
Teamleden	8
PO + SM	2 FTE
Totaal per team	**10 FTE**
Kosten per team	€50.000/iteratie
Kosten per team/jaar (11 mnd)	**€1.100.000/jaar**

Laag 3: ART team kosten per iteratie

Een ART in dit rekenmodel bestaat uit **8 Agile-teams, wat de meeste organisaties ongeveer hebben.** Op ART-niveau worden 4 extra FTE toegevoegd voor de RTE, PM(s) en Architect(en). Het System Team is al meegerekend als één van de agile-teams.

Parameter	Waarde
RTE, PM(s), Architect(en)	4 FTE
Aantal Agile-teams	8
Totale ART-bezetting	84 FTE
Kosten ART	€420.000/iteratie

Parameter	Waarde
Kosten ART/jaar (11 mnd)	~€9,2M/jaar

Laag 4: kosten per Solution Train per iteratie

Een Solution Train coördineert in ons voorbeeld **8 ART's**. Op ST-niveau komen daar 10 FTE bij voor ST/SM’s/SA's.

Parameter	Waarde
ST, S-PM's, SA's	10 FTE
Aantal ART's per ST	8
Totale ST-bezetting	682 FTE
Kosten ST	~€3,4M/iteratie
Kosten ST/jaar (11 mnd)	~€75M/jaar

Laag 5: totale kosten op LPM niveau

Op portfolioniveau komen daar nog **15 Epic Owners/LPM-functies/EA's** en **15 Agile Coaches/SPC's** bij, verdeeld over de Solution Trains in ons rekenvoorbeeld.

Parameter	Waarde
Epic Owners, LPM, EA's, Leadership team	15 FTE
LACE, VMO	15 FTE
Aantal Solution Trains	6
Totale bezetting	~4122 FTE
Kosten Portfolio	~€20,6M/iteratie
Kosten Portfolio/jaar (11 mnd)	~€453M/Jaar

Gemiddelde kosten per Epic in een SAFe fabriek

Het tweede rekenmodel werkt **bottom-up**: wat kost een Epic gemiddeld, opgebouwd vanuit de kleinste eenheid? We rekenen met dezelfde uitgangspunten van het rekenmodel. Dit rekenmodel helpt om de verschillende niveaus van SAFe-werkitems beter te begrijpen voor SAFe nieuwelingen. In de praktijk zie ik namelijk regelmatig dat organisaties moeite hebben met het correct benoemen van werkitems.

Het komt vaak voor dat iets een *Epic* wordt genoemd terwijl het in werkelijkheid een *Feature* is, of zelfs slechts een *User Story*. Door de kostenstructuur van de SAFe-fabriek te begrijpen, ontstaat automatisch een gevoel voor de schaal van het werk en juiste duiding in je agile tooling. Een wijziging van enkele duizenden euro’s hoort vrijwel nooit op portfolio-niveau thuis. En een Epic van minder dan enkele tonnen is meestal geen Epic, maar eerder een Feature die binnen een ART kan worden opgepakt. Met andere woorden, correcte terminologie is daarom niet alleen een kwestie van SAFe-theorie, maar ook van **economisch inzicht**.

Laag 1: Team Level, kosten per User Story

Een gemiddelde User Story vergt **3 dagen** werk van 3 **developers**. Zeg een frontend developer, backend developer, ook benodigde tester, UX/UI designer, Architect rekenen we voor het gemak even als 1 FTE.

Parameter	**Waarde**
Doorlooptijd per Story	3 dagen
Aantal developers	3
Kosten per Story	**€4.500,-**
Moet af zijn	binnen 1 iteratie, in verband met small batch sizes zeg 3 à 4 dagen.

ART Level: kosten per Feature

Stel een Feature bestaat gemiddeld uit **15 Business Stories** + **5 Enabler Stories** = 20 Stories.

Parameter	Waarde
Business Stories per Feature	15
Enabler Stories per Feature	5
Totaal Stories per Feature	20
Kosten per Feature	**€90.000,-**
Moet af zijn	binnen 1 Planning Interval (PI)

Epic Level: kosten per Epic

Stel, een Epic bestaat gemiddeld uit **15 Business Features** + **5 Enabler Features** = 20 Features.

Parameter	Waarde
Business Features per Epic	15
Enabler Features per Epic	5
Totaal Features per Epic	20
Kosten per Epic	**€1.800.000,-**
Moet af zijn	Binnen zeg 3 tot max 6 PI's.

Conclusie

Dit rekenmodel laat twee dingen helder zien.

• Ten eerste: software ontwikkelen kost geld, en vaak aanzienlijk meer dan men intuïtief verwacht. Alleen al één Agile Team kost ongeveer **€50.000 per iteratie**. Dat inzicht is belangrijk, omdat het helpt te begrijpen hoe Business Owners naar waarde kijken. Voor hen staat elke iteratie feitelijk gelijk aan een investering van tienduizenden euro's. De vraag is daarom niet alleen *wat* teams bouwen, maar vooral **of de opgeleverde waarde opweegt *tegen* deze investering**. Wanneer Business Owners een Iteration Review of System Demo bijwonen, kijken zij dus impliciet naar de vraag: *hebben we in deze iteratie voldoende waarde geleverd voor wat het ons heeft gekost?* Het begrijpen van deze economische realiteit helpt teams om beter te prioriteren, scherper

te formuleren wat echte klantwaarde is, en bewuster om te gaan met capaciteit en backlogkeuzes.

• Ten tweede: een Epic is *geen kleine* wijziging. Met een gemiddelde investering van €1,8 miljoen is goede portfoliokeuzes maken geen luxe maar noodzaak. Dat is precies waarom SAFe® Lean Portfolio Management en Epic-kwalificatie zo centraal staan in een volwassen implementatie, niet als bureaucratie, maar als bewuste allocatie van schaarse capaciteit.

Het rekenmodel is beschikbaar voor je als Excel-rekenmodel en kan worden aangepast op het aantal teams, ART's en Solution Trains voor je eigen organisatie.

Deel 9: Persoonlijk vakmanschap

58. Goed pragmatisme zonder solide theoretische basis is gokken!

Anekdote 1: de pragmatische SAFe-implementatieconsultant

Laatst sprak ik een professional die graag wilde meewerken aan een SAFe-transformatie. Hij stelde zich voor als SAFe-implementatieconsultant en vertelde dat hij al meerdere transformaties had begeleid. Dat is altijd interessant, waarna ik hem vroeg waar hij zijn SAFe opleidingen heeft gevolgd en welke. Zijn antwoord was: *“Geen. Daar geloof ik niet in, want ik ben een pragmatisch mens.”* Dat zette me aan het denken. Want hoe begeleid je een zeer complexe enterprise-transformatie zonder je ooit serieus in de achterliggende principes en best practices te hebben verdiept? Het hele grote geheel begrijpen? Waarom zou je geen behoefte hebben om te begrijpen hoe het framework precies in elkaar zit voordat je organisaties helpt het toe te passen?

Anekdote 2: de IKEA-kast

Iedereen kent het moment. Je staat voor een berg onderdelen, een zakje schroefjes en een handleiding die je resoluut opzij legt. *Hoe moeilijk kan het zijn?* Een uur later staat de kast. Trots. Tot je ziet dat de deur scheef hangt. Je zoekt het schroefje dat je "over" had. En je realiseert je: de fout zat niet in de laatste stap, die zat in de eerste vijf minuten. Bij een IKEA-kast is dat irritant. Je schroeft hem gewoon opnieuw in elkaar en investeert 2 uur tijd extra. Maar wat als die kast een organisatie is van 65.000 mensen?

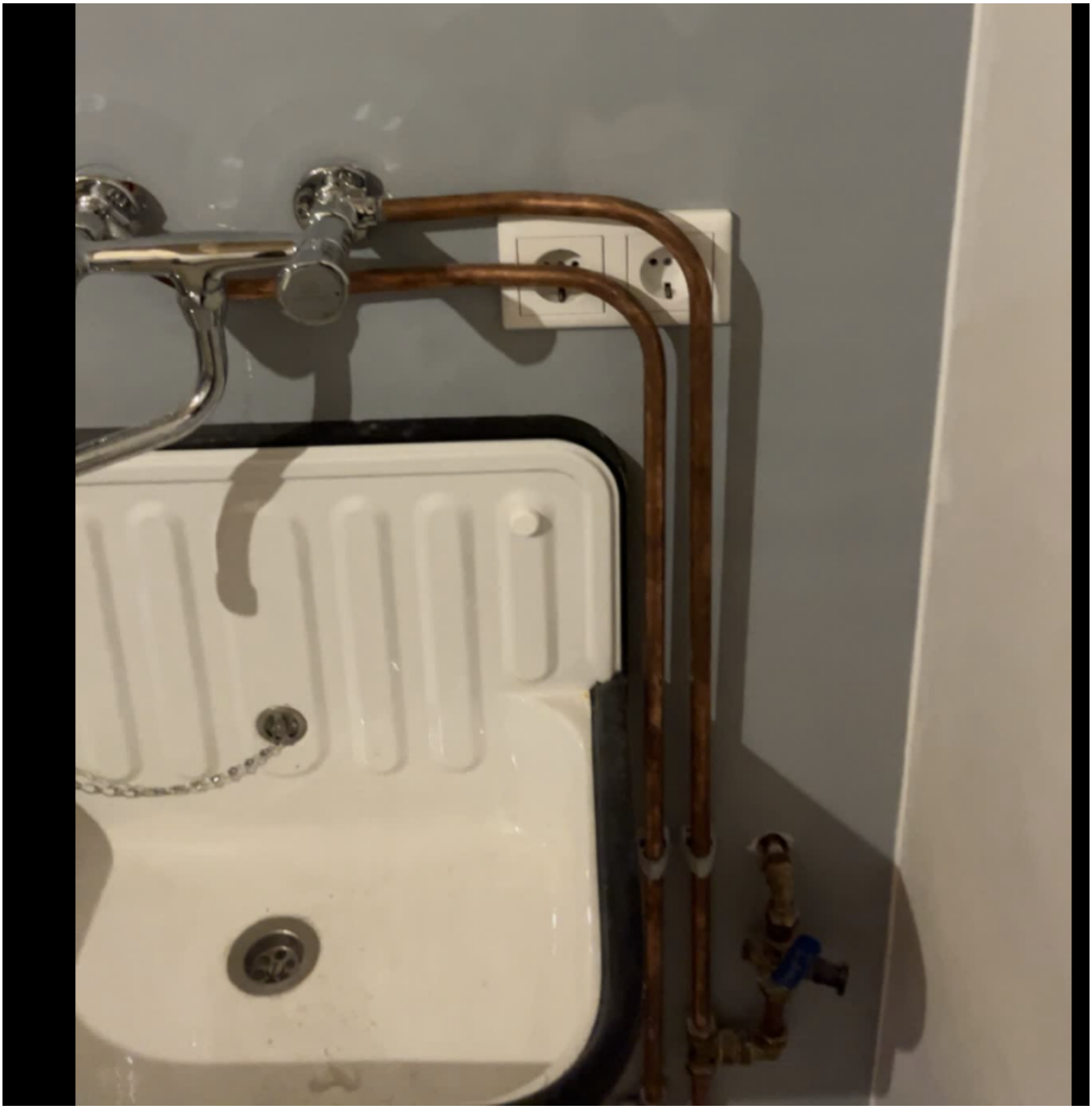
Inleiding

In de wereld van mijn opdrachten, trainingen en klanten voor SAFe wordt pragmatisme vaak geprezen als een effectieve benadering van probleemoplossing. Het vermogen om snel en doelgericht oplossingen te vinden kan onmiskenbaar waardevol zijn, vooral in een omgeving waar resultaten belangrijk zijn.

Vaak wordt het woord ‘pragmatisme’ gebruikt als excuus: om een (totaal) gebrek aan theoretische onderbouwing of formele training te maskeren. In plaats van pragmatisme als een bewuste keuze voor eenvoudige en praktische oplossingen, wordt het in zulke gevallen eerder een excuus voor een gebrek aan studie, examens halen, verdieping of voorbereiding. Deze professionals doen vaak hun werk 'op gevoel’, ‘monkey see, monkey do' of 'copy/paste vorige opdracht’.

Een te grote focus op pragmatisme brengt daarom ook risico’s met zich mee, waardoor ik altijd bij het horen van dat woord ‘rechtop in bed zit’. Ik ben altijd bang voor de gevolgen van hun oplossingen als ik streef naar excellente klantoplossingen, die toekomstvast en rock-solid in elkaar moeten zitten. Waarin alle COPAFIJTH-aspecten (Communicatie, Organisatie, Personeel, Administratie, Financiën, Informatie, Juridisch, Tijd en Huisvesting) grondig zijn geïdentificeerd, overwogen en meegenomen.

Hoe kunnen we dan toch een balans vinden tussen pragmatische oplossingen enerzijds en een diepgaande strategische visie en doordachte aanpak anderzijds? Dat onderzoek ik in dit hoofdstuk. Ik beschouw ‘pragmatisme’ eens vanuit een aantal gezichtspunten.



Het probleem is even ‘pragmatisch opgelost’

De kracht van pragmatisme

Pragmatische mensen richten zich, als ze het woord zelf overigens goed hebben begrepen, vaak op wat werkt in het hier en nu. Ze nemen beslissingen die directe, tastbare resultaten opleveren. Dit is ideaal voor het aanpakken van kortetermijnproblemen of het efficiënt bereiken van doelen met een lage onzekerheid.

Hun kenmerken zijn onder andere:

• Nuchterheid: gericht op feiten en realistische oplossingen.

• Resultaatgerichtheid: Focus op concrete uitkomsten.

• Oplossingsgericht denken: Prioriteit geven aan wat praktisch uitvoerbaar is.

• Out-of-the-box denken: Creatieve en soms onconventionele oplossingen.

• Efficiëntie: Snelheid boven langdurige analyse of overleg.

Voorbeelden van pragmatisch handelen zijn onder meer het verminderen van bureaucratie, het versnellen van processen, en het zoeken naar flexibiliteit binnen regels om sneller resultaten te behalen.



De mogelijke negatieve consequenties van pragmatisme

Hoewel pragmatisme positieve eigenschappen heeft, zijn er ook nadelen en vooral risico's die ik vaak zie en waarvoor ik bang ben.

Een pragmatische instelling kan zich vertalen in:

Gebrek aan theoretische opleiding & verdieping: Pragmatisme wordt soms ingegeven door een gebrek aan theoretische kennis of de motivatie, energie en zin om diepgaand onderzoek te doen, studies te volgen, boeken te lezen of (SAFe) certificaten te halen. Dit kan leiden tot een onderschatting van de waarde van gedragingen en principes zoals het Agile Manifesto of de Lean-Agile Principles en gemis van het zien en kennen van de grote verbanden, inzage in en overzicht over het grote geheel.

Kortetermijnfocus: Er is een risico dat men te veel gericht is op directe oplossingen zonder rekening te houden met langetermijngevolgen. Dit kan resulteren in structurele problemen die pas later zichtbaar worden. De oplossingen zijn niet 'sustainable', doordat er niet over cruciale aspecten is nagedacht.

Innovatiebeperkingen: Pragmatisme kan creativiteit en vernieuwing in de weg staan. Het vermijden van risico's en onconventionele ideeën kan de innovatiekracht van een organisatie beperken.

Beperkte visie: Een gebrek aan strategische diepgang en een solide theoretische basis (zoals Enterprise Architectuur denken, meervoudige SAFe-certificering, etc.) kan leiden tot een gemis aan richting en het mislopen van grotere kansen. Er zitten gaten in de oplossingen, waardoor het maximale rendement niet wordt gehaald.

Inefficiëntie door onvolledigheid: Pragmatische mensen starten vaak snel en lijken voortgang te boeken, maar missen soms cruciale aspecten van het grotere geheel. Met uiteindelijk vertraging, herontwerpen, terug naar de tekentafel als gevolg.

Beperkte oplossingsruimte: Een pragmatische benadering kan resulteren in standaardoplossingen *('monkey see, monkey do'*) die de ruimte voor maatwerk, innovatie en diepgaande analyses beperken.

Het vinden van de juiste balans

De sleutel tot succes ligt in het combineren van pragmatisme met strategische visie en innovatie. Een gezonde balans zorgt ervoor dat:

• Snelheid en efficiëntie behouden blijven.

• Vooruitdenken en innoveren een prominente plaats krijgen.

• Langetermijndoelen niet uit het oog worden verloren.

• Er een juiste balans is tussen denken en doen.

• Acteren en beslissingen nemen op basis van principes, best-practices, theoretische kennis en kunde. Een praktische vuistregel: stel bij elke pragmatische oplossing bewust de vraag "hebben we alle COPAFIJTH-aspecten gewogen, en wat zijn de langetermijngevolgen van deze keuze?"

Het vermogen om zowel praktisch als theoretisch-visionair te denken is essentieel voor duurzame groei en succes. Pragmatisme mag niet het enige kompas zijn, maar moet dienen als een nuttig hulpmiddel binnen een bredere strategie en theoretisch kader.

Tegelijkertijd zijn er valkuilen, zoals kortzichtigheid, een gebrekkige samenhang met bredere problemen, en een neiging om sociale en organisatorische grenzen te negeren. Daarom is het essentieel om ook het nut van analyse en overleg in te zien.

Conclusie

Pragmatisme is een waardevolle eigenschap, maar moet in balans worden gebracht met strategische visie, theoretische diepgang en concrete domain expertise (SME). Het combineren van al deze elementen leidt tot een aanpak die zowel snel als duurzaam is, en die ruimte biedt voor innovatie en groei. Een professional die pragmatisme, theorie en strategie weet te integreren, kan zowel kortetermijnresultaten behalen als toekomstbestendige oplossingen ontwikkelen.

Tot slot: Vraag jezelf eens af: wanneer jij of een collega 'ik ben pragmatisch' roept, is dat een bewuste keuze, of een comfortabele vlucht?

59. Wie ongevraagd wil adviseren moet goed weten wat hij doet

2. Inleiding

Vrijwel iedereen heeft ooit wel ervaren dat een goedbedoeld advies uiteindelijk niet werd opgevolgd. Zolang we het hebben over feedback op het kapsel van je partner, is dat nog redelijk onschuldig. Het wordt anders wanneer je in een zakelijke omgeving met een advies naar een meerdere toegaat. Niet zelden zal de ongevraagde raad dan in de wind worden geslagen. Als dit een of twee keer gebeurt, is het tot daar aan toe. Wanneer dit je voortdurend overkomt, zeker in teamverband, is het eindresultaat teleurstelling en frustratie. Er ontstaat een organisatie die niet meer van onderaf innoveert en waar een negatieve cultuur heerst. ”Het management luistert toch niet” en ”Hier lukt toch nooit iets, ik ben gestopt met meedenken”.

De vraag is natuurlijk of dit werkelijk zo is. Willen de managers echt niet luisteren of is het zo dat de ”adviseur” zijn hand op een of andere manier heeft overspeeld? Want wie ongevraagd wil adviseren moet goed weten wat hij doet. Oftewel: als je het doet, doe het dan goed!

> Een padvinder komt te laat bij het clubhuis, waarna de akela hem vraagt naar de reden. De padvinder antwoordt dat hij *een goede daad heeft verricht*, hij heeft namelijk een oud vrouwtje naar de overkant van de weg geholpen.
> De akela vraagt verbaasd: “Maar waarom ben je dan zo laat?” Waarop de padvinder antwoordt: “Ze wilde niet.”

3. Met een ongevraagd advies begeef je je op glad ijs.

Het dilemma bij veel ongevraagde adviezen is dat ze vaak hun doel missen. Dan is er veel kostbare organisatietijd en energie verloren gegaan. Als naar aanleiding van op zich goed bedoelde initiatieven succes uitblijft, kan dit uiteindelijk leiden tot frustratie en negativisme. Je komt dan in een negatieve spiraal terecht met als gevolg mogelijk reputatieschade en imagoverlies, zowel voor jezelf als voor de hele afdeling. In dit hoofdstuk volgt een beschrijving van een aantal problemen en ongewenste effecten die zich voordoen bij ongevraagd adviseren.

3.1. Ongewenste adviezen kosten onnodig veel tijd.

Adviseren is, als je het goed wilt doen, een tijdrovend proces. Veelal moet je eerst goed overleggen en je oor te luisteren leggen bij derden. Afsluitend wil je je advies wellicht in een mooi ingebonden rapport hebben. De ambitieuze adviseur wil dat de inhoud van het rapport zonder meer wordt geaccepteerd. Een kroon op zijn werk zou zijn als de ongevraagde adviezen ook nog eens één op één zouden worden geïmplementeerd. Wanneer men je adviezen meerdere malen naast zich neerlegt, rijst de vraag waar de fout ligt. Werd al die energie feitelijk niet verkeerd besteed? Had het adviesproces niet efficiënter en effectiever gekund? Had je in een veel eerder stadium al kunnen weten of het advies überhaupt kans van slagen had?

3.2. Continu afgewezen adviezen leiden tot frustratie.

Bij keer op keer nul op rekest krijgen beland je in een negatieve spiraal van frustratie en teleurstelling. Alle goed bedoelde initiatieven worden doodgeslagen. Terwijl je bij succesvolle adviezen in een positieve spiraal geraakt, aan kracht wint en succes aantrekt. Je doet mee en je hoort erbij!

3.3. Verkeerde adviezen leiden tot imagoschade.

Blijf niet doordrammen nadat je nul op rekest hebt gekregen op een ongevraagd advies. Kom ook niet steeds terug op eerder geventileerde meningen. Het wordt dan voor de ontvanger wellicht onaangenaam, met mogelijke reputatie- en imagoschade voor jezelf. Het probleem ligt niet bij de ontvanger, maar bij het niet goed volgen van het proces genaamd ongevraagd adviseren. Bijvoorbeeld door het overslaan van belangrijke processtappen of relationele belemmeringen. Gevolg is dat je steeds minder aan tafel zit met de keybeslissers en geleidelijk meer aan invloed inboet.

3.4. Constant niet gehoord worden bevordert een *laissez-faire*-cultuur.

Werknemers zullen stoppen met goedbedoelde ongevraagde adviezen als het zittende management ze consequent negeert. Dit werkt een *laissez-faire*-cultuur in de hand, met het gevaar dat werknemers en management elkaar gevangen gaan houden in hun werkrelatie. Hierdoor wordt het management afgesloten van veelal goede ideeën van de werkvloer.

4. Wees je bewust van je ware intenties

Voordat je ongevraagd gaat adviseren is het van belang bij jezelf na te gaan wat je ware intenties zijn. Is je advies wel volledig belangeloos, integer en onbaatzuchtig? Of is er eigenlijk sprake van een verborgen agenda achter je advies?

Mogelijke redenen voor ongevraagde adviezen zijn veelal:

· (Levens)overtuiging, de zogenaamde evangelisten

· Erkenning en herkenning

· Behoefte aan invloed en macht

· Erbij willen horen

· Liefde of bezorgdheid. Je wilt iemand ergens voor behoeden

· Onvrede. Je ziet punten die je irriteren (“Dit moet nu echt eens worden opgelost”)

5. Zorg eerst dat je aan tafel komt en begrijp de ontvanger.

Bij ongevraagd adviseren is de kern vaak een situatie, zakelijk of privé, die je wilt veranderen of verbeteren. Het mandaat voor implementatie ligt bij een ander, maar wel binnen jouw invloedssfeer. Met andere woorden, middels beïnvloeding zou je een bepaalde situatie kunnen veranderen. Hierbij kun je persoonlijk materieel - betere positie, meer budget, etc.- of immaterieel voordeel hebben - erkenning, liefde, minder stress, verhoogde afhankelijkheid bij de ander. Het grote verschil tussen ongevraagd adviseren en hiërarchisch leiding geven is dat het gewenste eindresultaat niet kan worden afgedwongen of topdown gedirigeerd.

De ongevraagde adviseur moet daarom beschikken over een arsenaal aan tactieken, en bovenal over kennis. Hij moet weet hebben van zowel relationele als (inhoudelijke) vertrouwensaspecten. Bekend zijn met voorkeursrepresentatiesystemen, aanvoelen welke adviesstijl in welke specifieke situatie het meest effectief is. En, bovenal, hij heeft passie voor het vak, de meest pure vorm van overtuiging en beïnvloeding.

5.1. Zorg dat je het vertrouwen krijgt en behoudt.

Voor het krijgen en behouden van vertrouwen moet je op meerdere vertrouwensdimensies het juiste niveau hebben.

Naar de theorieën van Paul Watzlawick vindt communicatie tussen mensen op twee niveaus plaats:

1. Inhoudsniveau

1. Betrekkingsniveau.

In het algemeen richt de meer harde sector van het bedrijfsleven (techneuten, bèta’s, blauwen) zich op inhoudsniveau, de meer zachte (hulpverleners, alfa’s) op betrekkingsniveau. Aan welk niveau meer belang wordt toegekend is mede afhankelijk van de hiërarchische positie binnen de organisatie. Stakeholders Management is op topniveau erg belangrijk. Daarom moeten topmanagers het betrekkingsniveau zeer goed toetsen en zo nodig bijsturen. Een relatie is nu eenmaal niet alleen gebaseerd op inhoud, maar ook op de communicatie hiervan. Zonder dit belangrijke fundament ontbreekt de basis voor een verder adviestraject.

In veel gevallen zullen zaken als bijvoorbeeld standsverschil ook een rol spelen. Een personeelslid dat zich direct tot een lid van de raad van bestuur wendt, moet van goeden huize komen om gehoord te worden. Mensen op dit niveau willen vaak door “gelijken” aangesproken worden. Zo is het voor een topmanager niet chique te zeggen dat bijvoorbeeld Piet Jansen zijn persoonlijk adviseur is, in tegenstelling tot bijvoorbeeld Manfred Kets de Vries. Je moet enig realiteitsbesef hebben om dit te kunnen inschatten.

5.1.1. Het moet emotioneel met je gesprekspartner ”klikken”.

Het is vrij moeilijk luisteren naar een adviseur die regelrecht in je allergiezone[1] zit. Daarnaast wil iedereen, werknemers, managers en directeuren, zo prettig mogelijk de dag doorkomen en zal er alles aan doen dit te bereiken. Mensen hebben de neiging pijn te vermijden en geluk na te streven. Daarom is het belangrijk alvorens op inhoud in te gaan eerst te zorgen dat het op relatieniveau klikt. Dit proces onder controle krijgen en de communicatie wederzijds zoveel mogelijk veraangenamen heet “rapport” maken. Contacten van hoge kwaliteit kenmerken zich door vertrouwen, openheid, betrokkenheid en respect. Pas als een dergelijk contact is opgebouwd, is er ruimte voor effectieve en wezenlijke communicatie. Rapport maken doe je door je verbale en non-verbale communicatie af te stemmen op je gesprekspartner. “Rapport” is het vermogen iemands wereld binnen te gaan, sympathie te verwerven en de ander het vertrouwen te geven dat je hem of haar begrijpt. Het is de basis en de essentie van succesvolle communicatie. Dat je rapport moet “maken” geeft al aan dat het niet iets statisch is, dat eraan gewerkt moet worden. Het is een proces dat bewust en constant moet worden geleid. Een proces dat een opgaande lijn kan vertonen, ineens weer terug kan zakken naar het beginpunt en zijn eigenaardigheden en grillen kan vertonen. Het is een voortdurende wisselwerking tussen zender en ontvanger. Dit opbouwen van rapport hoeft geen langdurig proces te zijn. Het kan enkele minuten tot enkele uren nemen. Vaak voel je al snel of het met iemand klikt. Of je elkaar “verstaat”. De juiste toon vindt. Maar rapport kun je ook verliezen. Passeer je de ontvanger, “overval” je hem wellicht? Laat je hem wel genoeg in zijn waarde? Is hij wellicht bang voor slimmere en sterkere medewerkers en luistert hij daarom per definitie niet meer? In dat geval ben je terug bij af en komt de inhoud ook niet meer bij de ontvanger aan.

Dit alles vergt een goed empathisch vermogen. Je moet gevoel ontwikkelen voor de sociale interactieprocessen tijdens het uitwisselen van de feitelijke inhoudsgeoriënteerde informatie. Met andere woorden, een adviseur moet over goede emotionele advieskwaliteiten en gevoelsantennes beschikken.

5.1.2. Je moet inhoudelijk voldoende niveau hebben.

Een ander aspect is de kwaliteit en diepgang van je inhoudelijke advieskwaliteiten. Die kwaliteit wordt mede bepaald door bijvoorbeeld je opleidingsniveau, je ervaring en de uitvoerbaarheid van je adviezen. Een goed advies heeft duidelijke eerste stappen. Inhoudelijk vertrouwen ontstaat door te laten zien dat je meervoudig kunt kijken. Bijvoorbeeld niet alleen naar techniek, maar ook naar impact, belangen en financiën. Vertrouwenwekkend zou bijvoorbeeld kunnen zijn de impact van je advies van tevoren te toetsen op alle assen van het McKinsey 7S-model. Met dit hulpmiddel weet je beter dat je in elk geval het advies van meerdere kanten hebt belicht.

Weet bijvoorbeeld waar de ontvanger in ieder geval op zal letten of wat hij belangrijk vindt. Anders veegt hij wellicht je advies onmiddellijk van tafel, alleen door een dimensie waar je niet aan hebt gedacht. Inhoudelijk vertrouwen ontstaat als de adviseur de inhoud kan vertalen naar de context van de ontvanger, of zelfs nog breder. Ik zeg daarom altijd: “Denk zo groot mogelijk”. Bezie je eigen advies vanuit management- en directieposities. Is het gegeven advies dan nog wel zo gunstig?

Een voorbeeld: als operationele specialisten zich laten verleiden tot het geven van organisatieadviezen (“ICT moet centraliseren”) overzien zij vanuit hun perspectief, kennis en ervaring vaak maar tien procent van de totale problematiek. Dit doet de oplossing in hun ogen vaak simpel lijken, omdat negentig procent van de impact op de organisatie hun ontgaat.

Tot slot moet je actuele, valide en professionele kennis bezitten en niet te veel fouten maken. Met een opmerking dat diesels langzaam zijn, wek je als adviseur vandaag de dag geen echt vertrouwen meer. Professionele kennis betekent dat deze van toepassing is op de schaal waarop je adviseert: “MS-Access” adviseren als server voor een corporate database voor een top200 onderneming getuigt niet van op klanten toegespitste en relevante

kennis. Als een ontvanger door onkunde het advies vervolgens wel implementeert en er na x maanden achter komt dat het niet werkt, win je niet bepaald aan invloed en reputatie.

Er kan op twee manieren een verschil in kennisniveau tussen de adviseur en de ontvanger bestaan:

Denkt de ontvanger dat hij meer weet dan de adviseur?

Dit is lastig. Want wanneer neemt de ontvanger dan iets aan? Alleen als hij het gevoel heeft dat de adviseur hem kan aanvullen of als hij zichzelf wil toetsen of scherp houden. Meer denken te weten ligt dicht bij ‘je beter dan de ander voelen’, je ‘superieur ten opzichte van de ander’ voelen. Dit gevoel wordt meestal bepaald door een combinatie van afkomst, opleiding, ervaring en emotie.

Denkt de ontvanger dat hij minder weet dan de adviseur?

Dan is de vraag: in welk geval neemt de ontvanger iets aan? Waarschijnlijk als de adviseur in de ogen van de ontvanger een specialist is, een goeroe. Vaak heeft de ontvanger al een beeld gekregen middels verhalen van anderen. In dit geval komt het dus aan op reputatie. Als de adviseur een onbekende is, gaat het veelal om de eerste indruk en de intuïtie van de ontvanger. Hij vangt algauw signalen op die zijn oordeel al dan niet bevestigen. Vervolgens bepaalt hij voor zichzelf of hij te maken heeft met een goede of een slechte adviseur. Afhankelijk hiervan zit je wel of niet aan tafel.

5.2. Zorg voor de juiste beeldvorming.

Met andere woorden, zorg in de rol van ongevraagd adviseur altijd voor een positieve beeldvorming. Daarmee kun je de grondhouding van de ontvanger al beïnvloeden voordat een gesprek überhaupt begint. Te denken valt aan publicaties, interviews, televisie-optredens, verbonden zijn aan universiteiten, klinkende resumés of een indrukwekkende lijst met referenties van succesvol uitgevoerde opdrachten. Want belangrijk bij adviseurs is of je Piet van XYZ bent, Piet van McKinsey of een welbekende invloedrijke lobbyist.

Tot slot kun je je reputatie op een slimme manier positief beïnvloeden door je bijvoorbeeld door anderen te laten introduceren en neerzetten. Zaak is dat een positief en deskundig beeld van je wordt geschetst en je goed wordt gepositioneerd.

5.3. Zorg dat je de handvatten van de ontvanger kent.

Iedere ontvanger heeft zo zijn eigen stijl van communiceren, zijn eigen handvatten. Om goed aan tafel te komen zul je deze handvatten moeten leren kennen. Hoe ontvangt en verwerkt de ontvanger bij voorkeur informatie? Hoe laat hij zich overtuigen? Is hij overtuigd als de concurrent het ook zo doet? Als hij het kan zien? Als bijvoorbeeld Gartner het ook vindt? Als hij sterke argumenten hoort? Als zijn tweede man er ook in gelooft?

Deze zaken staan ook wel bekend als het voorkeursrepresentatiesysteem. Voor maximale advieskracht is het belangrijk de wijze van presenteren af te stemmen op het voorkeurssysteem van de ontvanger. Stem vervolgens goed af of hij het advies graag even per e-mail wil, ofwel dertig pagina’s in kleur, met volledige onderbouwing en alle feiten. Kwaliteit is immers *the fitness for use*.

5.4. Zorg dat je de juiste adviesstijl kiest.

Voor welke wijze van benadering is de ontvanger gevoelig? Zorg voor maximale invloed door de juiste adviesstijl te hanteren: overtuigen, overreden, stellen, inspireren, overbruggen, etc.

Kies daarbij de juiste rol: expert, dokter of begeleider. Soms wil de ontvanger alleen een snel werkende pil opdat hij van een probleem wordt verlost. Soms wil hij het de volgende keer zelf kunnen oplossen en zoekt hij hulp bij zijn eigen denkproces om tot een passende oplossing te komen.

5.5. Houd rekening met normen, overtuigingen, vooroordelen, status en paradigma's.

Goed klikken ontstaat niet alleen bij dezelfde normen, waarden en levensovertuigingen, maar ook bij gemeenschappelijke belevingswerelden en referentiekaders. Tijdens het adviesproces zal een ongevraagd adviseur hier oog voor moeten hebben.

Voorbeeld van een mogelijke norm van de ontvanger: "Ik praat niet met personeel, alleen met management of hoogleraren". Je moet beseffen dat mensen er status aan ontlenen alleen met gelijken of hoger geplaatsten te spreken. Daar boks je moeilijk tegenop. Ook uiterlijk speelt een grote rol. Het is bewezen dat mensen met symmetrische gezichten aardiger gevonden worden. Denk bijvoorbeeld aan een manager die luid en stellig spreekt tegenover een collega die aarzelend en zacht spreekt. De manier van spreken kan beïnvloeden hoeveel gezag iemand krijgt toegeschreven, los van de kwaliteit van het argument. Andere mogelijke overtuigingen: ”Ik geloof alleen mensen die hun sporen in de ICT hebben verdiend”. “Ik geloof niet in omgeschoolde biologen” of - een voorbeeld van een vooroordeel - ”Bevolkingsgroep X is traag, daar zitten nooit goede projectmanagers tussen”. Vooroordelen en overtuigingen zijn moeilijk te achterhalen, zodat je soms niet goed weet waar het probleem ligt.

5.6. Adviseer recht vanuit je hart en met passie.

Natuurlijk berusten overtuigingskracht en geloofwaardigheid op argumenten. Toch worden ze mede, of zelfs voornamelijk, bepaald door de wijze waarop je iets brengt. Je communiceert middels lichaamshouding, stemvolume en woorden. Het is interessant hoe hierbij de verhoudingen liggen:

• 55% wordt uitgedrukt middels de lichaamstaal

• 38% middels de stem en slechts

• 7% middels woorden

Lichaamstaal en gesproken taal ontlopen elkaar dus niet veel! Enig charisma om je ideeën over het voetlicht te brengen kan daarom geen kwaad.

6. Zorg ervoor dat je advies zo snel mogelijk een gevraagd advies wordt.

Doe eerst grondig onderzoek voor je veel tijd steekt in ongevraagde adviezen: houdt iemand anders zich al met deze problematiek bezig? Is het probleem nog wel een probleem? Lost het zich wellicht over een aantal weken automatisch op? Door in een vroeg stadium een idee voor te leggen en bespreekbaar te maken kun je alvast je voelhorens uitsteken. Als je idee überhaupt geen kans van slagen heeft, hoef je er geen kostbare tijd in te steken. Voor de verkennende ronde, waarin je het eerste idee voorlegt, moet je voldoende in huis hebben van wat ik in dit hoofdstuk beschrijf: aansluiting, diagnostische kennis, commerciële vaardigheden en politiek inzicht. Als je initiële idee interessant genoeg is, krijg je mogelijk de opdracht een en ander nader uit te werken. Plots is je ongevraagde idee dan een gewenst advies.

6.1. Leg je eigen onwetendheid bloot.

First understand to be understood! Zoek uit wat je al wel en wat je nog niet weet. Houdt zich al iemand met het probleem bezig? Is het ergens anders al eens opgelost? Het heeft geen zin diverse verbetervoorstellen op afdelingsniveau te schrijven terwijl de ontvanger al lang weet dat de hele club geoutsourced wordt. Probleem hierbij is dat de ontvanger jou dit mogelijk niet mag en kan vertellen. Hij geeft daarom een politiek antwoord.

6.2. Zorg ervoor dat je aansluiting hebt bij de ontvanger.

Verlok de ontvanger allereerst met eventuele toekomstige voordelen of winsten. Zet een visje uit. Wees sensitief en pols de belangstelling. Doseer daarbij slim je informatie, een olifant eet je ook niet in één keer op. Zorg dat je idee voor de ontvanger te behappen is.

6.3. Verplaats je in de ontvanger en presenteer de voordelen.

Denk niet alleen aan je eigen belangen en voordelen. Hoe helpt het ongevraagde advies de ontvanger zijn persoonlijke en zakelijke doelen te realiseren? Want “jouw belang is niet ontvangers belang” en “jouw agenda is niet ontvangers agenda”. Leg de focus dus ook aan de andere kant van de tafel.

6.4. De juiste timing is cruciaal, zorg daarom voor een gevoel van urgentie.

Veel managers hebben het druk en houden zich alleen staande als ze elke dag prioriteiten stellen. Dit doen ze door *P1=urgent-belangrijke* zaken (zie matrix) als eerste af te handelen. En wat in het *urgent-belangrijk* kwadrant valt, wordt vaak bepaald door de persoonlijke doelen en ambities van de manager. Als er tijd over is, kan hij aandacht besteden aan de *P2=laag urgent- belangrijk zaken*. Timing is hierbij dus cruciaal. Je kan op het verkeerde moment met de goede ideeën komen. Voor alles is een juiste tijd.



Eigen illustratie: Major Milestones.

Als ongevraagd adviseur wil je uiteraard graag een akkoord voor je advies of idee. Dan is het belangrijk te weten of het advies waarmee een bepaald knelpunt wordt opgelost ook een P1-knelpunt in de ogen van de manager is. Zo niet, dan zal hij je mogelijk als tijdconsumerend ervaren in plaats van als tijdbesparend. De truc is hier jezelf slim te promoveren van het *P4-* naar het *P1-kwadrant.* Zorg dus voor een gevoel van urgentie, schep een behoefte. In dat geval word je eerder serieus genomen en kom je sneller aan tafel. Dit veronderstelt wel een goed inzicht in de belangen van de ontvanger. Wat drijft hem? Wat is op dit moment belangrijk voor hem? Speel hierop in en leer jezelf maximaal in hem te verplaatsen.

6.5. Begrijp hoe de hazen lopen.

Het doel van ongevraagd adviseren is een verandering of verbetering te bewerkstelligen. Het nadeel hiervan is dat bij veel veranderingen winnaars en verliezers zijn. Winnaars steunen de verandering, verliezers saboteren haar vaak. Zorg daarom dat je goed doorziet wat je ongevraagd advies voor de organisatie betekent. Wat is de reikwijdte ervan? Zijn er andere afdelingen bij betrokken? Raken anderen macht of bepaalde rechten kwijt? Welk belang schaad je mogelijk? Wie moet je nog meer meekrijgen (*informal influencers*) voordat je advies wordt opgevolgd? Wie wordt er beter van, en wie slechter? Is volkomen duidelijk wie uiteindelijk beslist?

6.6. Zorg dat je je gesprekspartner meekrijgt.

Verder is het nu zaak de ontvanger mee te krijgen, van het initiële idee een opdracht te maken. Vraag concreet akkoord voor verdere uitwerking in exact dat formaat en die vorm als door de ontvanger gespecificeerd en gewenst. Houd contact en ga zo nu en dan na of de ingeslagen weg de goede is, want de wereld draait door en er kunnen nieuwe inzichten ontstaan. Verdwijn niet wekenlang onder water.

7. Als je het goed doet, levert het veel winst op.

7.1. Je wint aan reputatie, je wordt erkend en herkend.

Succesvolle implementaties van ongevraagde adviezen leveren een onderneming winst en voordeel op of voorkomen mogelijke problemen. Als men dit erkent, doe je mee, krijg je schouderklopjes en neemt men je serieus. Met andere woorden, je komt en blijft aan tafel.

7.2. Alles lukt, je kunt de wereld aan.

Succesvolle ongevraagde adviezen geven energie. Je wordt een winnaar en geraakt in een positieve spiraal, een bepaalde opgaande flow. En van het een komt het ander. Succes geeft energie, mislukking kost energie. Je begrijpt het proces. Zodra je merkt dat je niet meer aan tafel zit, zet je dit recht. Dit kan tijd kosten, want vertrouwen, bijvoorbeeld, is niet per ons te koop. Je zorgt ervoor dat je goed onderzoekt wat er gebeurde. Tot die tijd heeft ongevraagd advies geen zin, want de boodschap komt toch niet over.

7.3. Je wordt effectiever en wint aan tijd.

Je wint veel tijd door in een vroeg stadium te polsen of je advies überhaupt zin heeft. Geen onnodige tijdsinvesteringen meer die niets opleveren. Geen grootse plannen die in de la, of, nog erger, de prullenbak belanden. Door slim te manoeuvreren weet je al vroeg of je advies een losse flodder of een schot in de roos is. In de tijd die je bespaart, kun je werken aan andere nieuwe ongevraagde adviezen en zo maximale invloed verwerven. Er is meer tijd over. Dit betekent minder stress, minder frustratie en meer energie en lol in je werk.

7.4. Je krijgt nieuwe opdrachten en kunt harder groeien.

Initiële ongevraagde adviezen worden nu gewenste adviezen. Je wordt meer gevraagd, waardoor je meer meemaakt, meer ervaring krijgt. Je groeit harder dan je collega's, waardoor de wereld aan je voeten ligt.

7.5. Eerste stap naar excellentie.

De ongevraagde adviseurs worden nu niet meer als lastig en zeurderig ervaren maar als meedenkend en participerend. Immers, de P1's van de manager zijn nu dezelfde als de P1's van de ongevraagd adviseur. Het bedrijf innoveert weer van onderaf, continue verbetering vindt plaats. Een betere werkrelatie, betere interactieprocessen tijdens het ongevraagd adviseren. En dan hoort een manager in zijn onderneming nooit meer: "Het management luistert hier toch niet."

8. Hoe een ongevraagd advies een succesvol advies wordt

1. Leg in de verkennende ronde een eerste idee voor. Hierdoor kun je peilen of je idee kans van slagen heeft. Zo niet, dan hoef je er geen verdere tijd in te steken. Bereid dit gesprek voor met de drie aandachtspunten in stappen 2 tot en met 4 hieronder: je acceptatie als gesprekspartner, aansluiting en timing, en de politieke verhoudingen.

1. Onderzoek of je voor de ontvanger een acceptabele gesprekspartner bent, zowel emotioneel als inhoudelijk. Is dit niet het geval, kijk dan of je iemand anders (die wel acceptabel is) zover kunt krijgen jouw idee/voorstel te presenteren of te ondersteunen.

1. Zorg dat je voorstel aansluit bij de belevingswereld van de ontvanger. Is dit je baas: wat vindt hij belangrijk en hoe kun je daarop inspelen (geef voordelen). Zorg bovendien voor de juiste timing!

1. Ken de politieke verhoudingen. Als je idee/advies wordt uitgevoerd, wie voelt daarvan de impact, met name op hoger niveau? Kijk naar het grotere plaatje. Is je idee dan nog steeds acceptabel of moet je eerst ook op hoger niveau meer draagvlak creëren?

[1] Referentie kernkwadranten van ir D.D. Ofman

60. Waarom mijn zakelijke inbox elke dag leeg is en ik nooit iets vergeet of mis: ondanks 100+ mails per dag!

Waarom mijn zakelijke inbox elke dag leeg is en ik nooit iets vergeet of mis: ondanks 100+ mails per dag!

Laat ik meteen een misverstand wegnemen. Mijn zakelijke inbox is niet leeg omdat ik weinig e-mail krijg. Integendeel. Op een normale werkdag ontvang ik met gemak 100+ e-mails, van mensen én van geautomatiseerde systemen.

Mijn inbox is bijna altijd leeg omdat ik in Outlook een slim **Inbox-Zero** rule-systeem heb gebouwd dat voor mij al jarenlang werkt. Een lege inbox is geen doel op zich, maar het **resultaat van grip hebben op je werk, je aandacht en je hoofd**.

Ik zie bij veel mensen hetzelfde gebeuren: een volle mailbox zorgt voor onrust. Elke e-mail voelt als een open eindje. *Moet ik hier nog iets mee? Hoeveel werk ligt hier nog? Wanneer ga ik dit doen? Mis ik iets belangrijks?*

Een volle mailbox geeft veel professionals stress. Niet omdat zij het werk niet aankunnen, maar omdat ze geen overzicht meer hebben **en niet weten** hoe lang ze erover doen om alle e-mails weg te werken: 2 uur? 6 uur? 2 dagen? Een lege inbox daarentegen doet precies het tegenovergestelde: die geeft rust. En dát is de reden waarom ik al jaren geleden mijn e-mailsysteem ben gaan automatiseren zodat vrijwel ieder moment van de dag beheersbaar is.

Een inbox is geen takenlijst

Veel mensen gebruiken hun inbox als to-dolijst, archief, geheugensteun én stressgenerator tegelijk. Alle berichten staan nog in de inbox. Dat kan niet goed gaan! Een inbox is bedoeld voor **onverwerkte input en is geen archief**. Niet meer, niet minder. Alles wat in die inbox blijft liggen, blijft je aandacht trekken.

Je kunt jezelf voornemen om “elke dag alle e-mail weg te werken”. Dat werkt vaak een paar weken. Daarna wint de realiteit. Daarom heb ik mijn mailbox **zelforganiserend** gemaakt met een slimme set Outlook rules.

Het principe is simpel:

krijg ik een bepaald type mail vaker dan twee keer, dan automatiseer ik het.

Het resultaat is dat ongeveer **80 tot 90% van de naar mij gestuurde berichten mijn inbox niet eens raakt**, die gaan direct naar de juiste map. Belangrijk om te benadrukken: geen enkele e-mail verdwijnt of raakt zoek. Alles komt óf zichtbaar in mijn inbox óf in een map met een ongelezen teller. Ik mis dus niets; ik kies bewust wanneer ik iets zie en wat ik dan zie. *Dat* heb ik geautomatiseerd.

(Voor het gemak spreek ik vanaf nu alleen over rules; Gmail werkt vergelijkbaar met filters.)

De basis van een effectieve Rules-structuur

Als je je mailbox onder controle wilt krijgen, moet je één ding accepteren: je ontvangt grofweg twee soorten e-mail:

• e-mails van mensen

• e-mails van applicaties

Als je die twee door elkaar laat lopen in je inbox, ontstaat chaos. De oplossing is niet “harder werken”, maar **slim sorteren aan de poort** met rules.

Belangrijk om te weten: Outlook en Gmail filters verwerken rules, **op volgorde** (van boven naar beneden). Daardoor geldt:

• hoe hoger de rule staat, hoe belangrijker en definitiever deze is;

• als Rule 1 een e-mail al verplaatst, kan Rule 2 die mail vaak niet meer verwerken;

• als twee rules hetzelfde matchen, wint degene die hoger staat;

• gebruik waar mogelijk “stop processing more rules” bij regels die definitief zijn.

E-mailmappenstructuur = mentale structuur = rust

Mijn e-mailmappen zijn bewust genummerd (bijvoorbeeld ## Core, ## Support, ## Archief).

• Bovenin staan de mappen die horen bij mijn **core business**,

• daaronder de **ondersteunende** mappen en

• helemaal onderaan het **archief**.

Hoe lager een map staat, hoe minder aandacht die krijgt. Dat is geen toeval, dat is ontwerp. De belangrijkste mappen staan altijd bovenin en zijn direct zichtbaar.

Voordat je rules gaat aanmaken, is het verstandig om eerst stil te staan bij je e-mailmappenstructuur. In de praktijk ontstaat een goede structuur niet in één keer. Die ontwikkel je door dit proces een paar keer te doorlopen en tussentijds bij te stellen.

De meest praktische aanpak is om **mappen aan te maken terwijl je de rules bouwt**. Op dat moment zie je direct welke soorten e-mail binnenkomen, welke mappen je daadwerkelijk nodig hebt en welke namen logisch en herkenbaar zijn. Let daarbij ook op de nummering van je mappen. Door mappen bewust te nummeren bepaal je automatisch welke mappen bovenin staan en waar je aandacht naartoe gaat. In de praktijk is dit geen eenmalige actie. Eens per paar maanden loop ik mijn inbox en mappen kort door: zie ik nieuwe patronen of terugkerende mailtypes, dan voeg ik één rule toe en is mijn systeem weer actueel.

Voorbeeld zakelijke Rules-opzet (Als-Dan)

Onderstaande rules vormen een compacte basis die voor de meeste kenniswerkers werkt. Ze zijn bewust eenvoudig gehouden en zorgen ervoor dat in je inbox alleen e-mails overblijven waar jij **nu of snel** iets mee moet doen.

Rule 1: delete
Als ik e-mails ontvang die structureel geen enkele waarde hebben voor mijn werk, bijvoorbeeld terugkerende notificaties, automatische bevestigingen of informatieve mails waarvan ik zeker weet dat ik ze nooit nodig heb, dan verwijdert deze rule de e-mail direct.

Let op: deze rule gebruik ik alleen voor e-mails waarvan ik 100% zeker weet dat ze geen actie, informatie of bewijswaarde hebben.

Let op: In de praktijk test ik dit soort rules eerst door e-mails tijdelijk naar een map te verplaatsen, voordat ik ze automatisch laat verwijderen.

Rule 2: Key stakeholders / belangrijke personen
Als ik een e-mail ontvang van een vooraf gedefinieerde belangrijke persoon of key stakeholder, dan verplaatst de rule deze naar een specifieke map (bijvoorbeeld Klant X of Naam Persoon).

Rule 3: CC- en groepsmails
Als ik alleen in de CC sta, dan verplaatst de rule deze naar de map 030) CC → CC_Persoonlijk.
Als de e-mail naar een groep of distributielijst is gestuurd, dan verplaats ik deze naar 030) CC → CC_Groepen.

Rule 4: Agenda- en meetingreacties
Als het onderwerp woorden bevat zoals *Accepted*, *Declined*, *Tentative* of *Cancelled*,
dan verplaatst de rule deze naar de map MEETINGS.

Rule 5: Administratie & bewijs
Als het onderwerp woorden bevat zoals *factuur*, *invoice*, *order*, *bestelling*, *bevestiging*, *contract*, *betaling* of *referentie*, dan verplaatst de rule deze naar Administratie en Bewijs.

Rule 6: Nieuwsbrieven en updates
Als het onderwerp woorden bevat zoals *nieuwsbrief*, *newsletter*, *update*, *release notes* of *insights*,
dan verplaatst de rule deze naar de map 055) NIEUWSBRIEVEN EN UPDATES.

Rule 7: Systeem- en applicatiemails
Als een e-mail afkomstig is van applicaties of systemen zoals Jira, MS Teams, ServiceNow, HR-systemen, monitoring, noreply, Bol.com of Booking.com, dan verplaatst de rule deze naar de map 080) SYSTEEM of een bijbehorende submap.

Rule 8: De laatste “Rule” (geen Rule, wel essentieel)

Alles wat niet door bovenstaande rules wordt geraakt, blijft in je inbox. Dat is bewust: dit zijn e-mails waar jij eigenaar van bent.

Voor elke inbox-mail stel je jezelf vier vragen:

• Moet ik hier nu iets mee doen?
→ meteen doen en daarna archiveren

• Moet ik hier op korte termijn iets mee doen, morgen, overmorgen?
→ naar 010) OP ZEER KORTE TERMIJN (niet laten zweven)

• Hoef ik hier niets direct mee te doen?
→ direct archiveren (090) BEWAAR/ARCHIVERING of 020) LATER EN ALS TIJD)

• Zie ik een patroon en krijg ik dit type mail vaker?
→ maak er een rule van en voer die direct uit op je inbox.

Belangrijke nuance: niet alleen je inbox lezen en verwerken telt

Het doel van deze aanpak is **niet** dat je alleen nog maar je inbox beheert en de rest negeert. Integendeel. De inbox is uitsluitend bedoeld voor **ASAP-werk**: e-mails waar je nu of heel snel iets mee moet doen. Andere soorten e-mail worden bewust verplaatst naar aparte mappen, zoals:

• nieuwsbrieven → 055) NIEUWSBRIEVEN EN UPDATES

• ICT-incidenten → 080) SYSTEEM → ICT Helpdesk

• CC-mails → 030) CC → CC_Persoonlijk

• systeemupdates → 080) SYSTEEM

Deze mappen beheer je niet continu, maar bewust en alleen op momenten dat je tijd, ruimte of zin hebt. Belangrijk om te weten is dat ook deze mappen gewoon “xx ongelezen” tonen, je ziet dus wél dat er iets is binnengekomen, maar het vraagt niet direct je aandacht.

Deze aanpak gaat niet primair over tijd besparen, maar over mijn **energiemanagement**. Ik bepaal zelf wanneer iets mijn aandacht krijgt, in plaats van dat mijn mailbox dat voor mij doet.

Tot slot

“Inbox zero” is geen trucje en geen doel op zich. Het is een bewuste keuze om je aandacht niet te laten gijzelen en structureel rust te creëren. Outlook Rules of Gmail Filters zijn geen nerd-gedoe, maar een persoonlijk operating system voor kenniswerk.

Zie je dit soort patronen in je eigen mailbox terug? Begin eens met één of twee inbox rules en test eens een poosje. Dat is vaak al genoeg om rust te creëren!

Bijlage: voorbeeld mappenstructuur

Onderstaande conceptmappenstructuur is bedoeld als inspiratie en startpunt, niet als blauwdruk. Gebruik haar om je inbox te ontlasten en je aandacht te sturen: alles wat direct actie vraagt blijft in de inbox of gaat naar *010) OP ZEER KORTE TERMIJN*, wat kan wachten gaat naar *020) LATER EN ALS TIJD*, en informatie, systeem- en administratieve e-mails krijgen een vaste plek buiten je directe aandachtsveld. Begin klein, pas de namen en submappen aan naar je eigen context en voeg alleen mappen toe als daar een duidelijke regel en functie achter zit. Het doel is niet perfect sorteren, maar rust en overzicht creëren door bewuste keuzes.

INBOX

<alle emails die niet zijn weggeruled naar mappen en je gedurende komend uur nog wilt processen>

<move naar 010) OP ZEER KORTE TERMIJN als je vandaag geen tijd hebt, maar wel op zeer korte termijn 'iets mee moet doen'

010) OP ZEER KORTE TERMIJN

├─ <kies zelf of je nog submap wilt>

├─ <kies zelf of je nog submap wilt>

020) LATER EN ALS TIJD

├─ Lezen, leren, consumeren als ik tijd heb en dan verder verwerken

├─ <kies zelf>

030) CC

├─ CC_Persoonlijk

└─ CC_Groepen

040) ADMIN EN BEWIJS

├─ Facturen 2025,2026, etc

├─ Contracten

├─ Bestellingen

├─ Bevestigingen

└─ Betalingen

050) PROJECTEN OF WERKGROEPEN

├─ Werkgroep 1

├─ LPM

└─ Project ABS

055) NIEUWSBRIEVEN EN UPDATES

├─ Nieuwsbrief 1

├─ Nieuwsbrief 2

└─ <kies zelf>

070) MEETINGS

├─ Accepted

├─ Declined

└─ <kies zelf>

080) SYSTEEM

├─ Jira of ADO

├─ MS_Teams

├─ ICT Helpdesk

├─ HR_Systemen

└─ Overig

090) BEWAAR/ARCHIVERING

├─ 2024

├─ 2025

└─ 2026

61. E-mailonderwerpregelconventie: grip op je inbox tijdens een SAFe-transformatie

Je opent je inbox en weet niet waar te beginnen. Wat is urgent? Wat kan wachten? Wat verwacht iemand eigenlijk van jou? Wie een SAFe®-transformatie start, weet dat dit gevoel al snel toeneemt. Nieuwe stakeholders, nieuwe overlegstructuren, nieuwe teams, en dus: meer mail.

Veel meer mail. Zonder afspraken over hoe je die stroom organiseert, verlies je snel de grip. Je mist acties, vergeet vragen te beantwoorden, en besteedt kostbare tijd aan het doorzoeken van een overvolle inbox.

Mijn ECC (E-mailonderwerpregelconventie) biedt een eenvoudige oplossing: door het onderwerp van elke e-mail te beginnen met een korte code, weet de ontvanger direct wat er van hem of haar verwacht wordt, nog vóórdat de mail geopend wordt.

De conventie: één of twee codes, directe duidelijkheid

Zet aan het begin van elke onderwerpregel één of twee van de volgende codes. Iedere code wordt gevolgd door een dubbele punt. Je kunt bijvoorbeeld U: en A: combineren tot U: A: voor een urgente actie.

Code	Betekenis	Verwachte reactietijd
U:	Urgent: hoge prioriteit, lees en reageer direct	Zo snel mogelijk, asap
Q:	Question: ik heb een vraag en verwacht een antwoord	liefst < 1 werkdag zodat de zender doorkan
A:	Action: ik verwacht dat je iets doet	< 2 werkdagen, anders kan de afzender niet door.
I:	Information: ter informatie, geen actie vereist	Geen
LACE: , VMO:, ARTX:	Optioneel om te zorgen dat mail over 1 onderwerp naar één e-mailmap gaat	Afhankelijk van de Q: of A: in combinatie met het topic domain
<geen code>	informeel bericht, geen actie verwacht of van buitenstaander	

Voorbeelden in de praktijk

I: We zijn van 1 t/m 5 januari buiten kantoor

Q: Kunnen we morgen overleggen? Mijn auto staat in de garage

U: A: Contractuele issues: actie vereist vandaag

A: ART Launch: Start met de voorbereiding van de PI Planning sessie

Q: Kan jij de PI Objectives van ART 2 nalopen vóór vrijdag?

A: Iedereen: lever je beschikbaarheid aan voor het volgende PI

Waarom dit in een SAFe-omgeving onmisbaar is

Een SAFe-transformatie is documentintensief en communicatie-intensief per definitie.

Denk aan:

• PI Planning-voorbereiding en de bijbehorende afstemming met meerdere ART's

• Stakeholdercommunicatie vanuit het LACE richting management en teams

• Contractuele en logistieke afstemming met leveranciers en trainingspartners

• Terugkerende updates over voortgang, risico's en besluiten

Zonder structuur stapelt dit zich op. Met de ECC zie je in één oogopslag wat urgent is, wat een actie vereist, en wat je rustig later kunt lezen. Dat scheelt je dagelijks meerdere minuten, en voorkomt dat je iets mist.

Combineren met automatisch sorteren: Outlook rules!

De ECC wordt nóg krachtiger in combinatie met automatische Outlook-regels. Door regels in te stellen op basis van de code in de onderwerpregel (bijvoorbeeld alle mails met I: automatisch naar een aparte map) organiseer je je inbox zonder enige handmatige actie.

Meer hierover lees je in hoofdstuk 60, ‘Waarom mijn zakelijke inbox elke dag leeg is en ik nooit iets vergeet of mis: ondanks 100+ mails per dag!’.

Waar deze conventie vandaan komt

Deze conventie heb ik niet zelf verzonnen. Ik heb hem geleerd in het begin van mijn carrière, toen ik werkte bij **Digital Equipment Corporation (DEC)** aan de **Europalaan 44 in Utrecht**, destijds een van de toonaangevende computerbedrijven ter wereld, bekend van de PDP- en VAX-minicomputersystemen.

Wat me altijd is bijgebleven: Digital Equipment was het enige bedrijf waar ik ooit heb gewerkt of mee samengewerkt dat zo'n e-mailconventie **consequent en bedrijfsbreed** toepaste. Niet als experiment, niet als voorstel van een enthousiaste medewerker. Nee, iedere nieuwe collega die binnenkwam werd er direct in getraind. Dag één. Inclusief executive management. Het was gewoon zo. En het werkte. Ik heb die gewoonte nooit vergeten. En in al die jaren erna heb ik geen enkel ander bedrijf gezien dat dit op dezelfde consequente manier heeft volgehouden.

Als jij ook bij Digital Equipment hebt gewerkt en je deze conventie ook herinnert: stuur me dan een LinkedIn-bericht op https://www.linkedin.com/in/roelpeels/. Dat vind ik altijd leuk.

De enige randvoorwaarde: iedereen doet mee

Net als bij de BNC Bestandsnaamconventie (zie hoofdstuk 63, ‘Documentchaos in je Agile transformatie? Een slim ontworpen BNC lost het op’) geldt ook hier: de ECC werkt het beste als het hele team, ART, SAFe fabriek, de afspraak consequent hanteert. Eén persoon die de code weglaat, doet de systematiek voor dat deel teniet. De investering is minimaal: een korte uitleg en een gedeeld document is voldoende. De terugverdientijd is onmiddellijk.

Digital Equipment begreep dat al in de jaren tachtig. Elke nieuwe medewerker werd direct getraind. Geen uitzonderingen. Dat is precies waarom het daar werkte, en waarom het overal werkt waar men het serieus neemt.

62. Documentchaos in je Agile transformatie? Een slim ontworpen GMS lost het op (deel 1)

2. Inleiding

In kleine organisaties lijkt documentbeheer/informatiemanagement vaak nog overzichtelijk. Maar zodra organisaties groter worden groeit het aantal documenten explosief. Dit geldt in het bijzonder voor Agile transformaties. Een serieuze transformatie met meerdere portfolio's, Solution Trains en Agile Teams produceert enorme hoeveelheden informatie: rollenbeschrijvingen, policies, Target Operating Model met tientallen hoofdstukken, architectuurdocumenten, werkinstructies, presentaties, assessments, trainingsmateriaal en honderden detailontwerpen. Dit kan gemakkelijk oplopen tot >100.000 documenten binnen 2 jaar tijd. Ik heb meerdere keren bij opdrachtgevers gezeten met >300.000 confluence pages of sharepoint documenten.

Het probleem is niet dat de documenten ontbreken. Het probleem is dat de organisatie ze kwijt is doordat onduidelijk is welke informatie je waar moet opslaan. Medewerkers zoeken naar het laatste Target Operating Model, het laatste LPM-ontwerp, contracten, besluiten, beleidsdocumenten. Ze weten zeker dat die documenten bestaan. Maar waar ze staan, is een ander verhaal. Laat staan wat exact de laatste versie is.

Daarmee ontstaat een fundamentele vraag:

Hoe zorg je ervoor dat je agile transformatie in control blijft over haar transformatie-informatie?

3. Leren via herkenbare privévoorbeelden met 2 bouwstenen

Het interessante is dat dit probleem niet alleen in organisaties bestaat. Vrijwel iedereen kent het ook uit zijn privésituatie. Documenten bestaan ergens op de harde schijf van je gezin, maar vaak weet niemand precies waar. Je zoekt naar een polis, een factuur, je hypotheek nota, een diploma of een document waarvan je zeker weet dat je het ooit hebt gemaakt maar waar had je die toen opgeslagen?

Dit waarschijnlijk voor iedereen herkenbare probleem vormt het vertrekpunt van dit hoofdstuk. Omdat elke organisatie anders is, leg ik de oplossing niet uit aan de hand van een specifieke organisatie, maar aan de hand van een privésituatie die iedereen begrijpt. Iedereen kent namelijk zijn eigen huishouden en zijn eigen documenten. Daardoor zijn voorbeelden direct herkenbaar.

Wanneer je de principes eenmaal begrijpt, kun je ze vervolgens eenvoudig vertalen naar de informatiehuishouding van je eigen organisatie.

De aanpak die ik gebruik bestaat uit twee bouwstenen:

1. **GMS: generieke mappenstructuur:** de logische structuur waarin documenten worden opgeslagen.

2. **BNC: BestandsNaamConventie:** een standaard manier om bestanden te benoemen zodat documenten ook buiten hun map herkenbaar blijven.

Dit hoofdstuk gaat over het **eerste onderdeel:** de **Generieke mappenstructuur (GMS)**. Het tweede onderdeel, de **BestandsNaamConventie (BNC)**, wordt behandeld in hoofdstuk 63, 'Documentchaos in je Agile transformatie? Een slim ontworpen BNC lost het op'. Samen vormen deze twee onderdelen een eenvoudige maar krachtige methode om jouw documentbeheer structureel op orde te brengen.

In de rest van dit hoofdstuk laat ik zien hoe je een **Generieke mappenstructuur (GMS)** ontwerpt waarmee je documenten niet alleen opslaat, maar ook altijd weer kunt terugvinden i.c.m. een BNC.

4. Waarom Full Text Retrieval *alleen* niet werkt

In discussies over documentbeheer hoor je regelmatig het volgende argument: als zoekmachines tegenwoordig zo goed zijn, waarom zouden we dan nog mappenstructuren en naamgevingsregels nodig hebben? Het idee klinkt aantrekkelijk. Je gooit alle bestanden in één grote bak, of 1 map, en laat een Full Text Retrieval engine alles indexeren. Daarna hoef je alleen nog maar een zoekterm in te typen.

In de praktijk werkt dit echter minder goed. Het eerste probleem is dat je nog steeds moet weten wát je zoekt. Je moet een woord kunnen bedenken dat in het document voorkomt. Als je dat woord niet meer weet, of als het document andere terminologie gebruikt dan jij in gedachten hebt, dan blijft het document onvindbaar. Het tweede probleem is dat niet elk document doorzoekbare tekst bevat. Denk aan gescande PDF's zonder OCR, foto's, afbeeldingen, ZIP-bestanden of oude documenten. Voor een zoekmachine bestaan deze bestanden wel, maar ze zijn inhoudelijk niet te indexeren. Ze verdwijnen daardoor stilletjes in de vergaarbak.

Daarom zie ik Full Text Retrieval niet als oplossing, maar als hulpmiddel in combinatie met mijn GMS en BNC aanpak.

De voorbeelden hierboven schetsen allemaal hetzelfde onderliggende probleem: documenten hebben meerdere kenmerken tegelijk, maar een mappenstructuur kent slechts één hiërarchie. Je kunt een bestand nu eenmaal maar op één plek opslaan. *Dit* is het *kernprobleem* dat een goede GMS moet oplossen.

5. Vingeroefening

Voordat we de ontwerpprincipes van een goede GMS behandelen, is het nuttig om eerst een eenvoudige oefening te doen. Zie onder ongeveer dertig documenten die vrijwel ieder huishouden kent. Probeer daar zelf eens een logische mappenstructuur voor te bedenken. Het doel van deze oefening is niet om direct de perfecte indeling te maken, maar om te ervaren hoe snel ogenschijnlijk eenvoudige documenten al classificatievragen oproepen. Kijk bij elk document in welke soort map je het zou opslaan, volgens welk ordeningsmechanisme ga je ordenen?

Praktijkvoorbeeld:

Document	Document	Document	Document
Paspoort	Rijbewijs	Geboorteakte	Huwelijksakte / samenlevingscontract
DigiD-gegevens / BSN-overzicht	Koopakte / huurcontract woning	Hypotheekakte	Eigendomsbewijs (kadaster)
Verzekeringspolis opstal	Verzekeringspolis inboedel	Energiecontract (stroom/gas)	Servicecontract cv-ketel
Jaarlijkse belastingaangifte (IB)	WOZ-beschikking	Bankafschriften (jaaroverzicht)	Salarisstroken
Jaaropgave werkgever	Pensioenopgave	Kentekenbewijs auto	Verzekeringspolis auto (groene kaart)
APK-keuringsrapport	Zorgverzekeringspolis	Vaccinatiebewijs / medisch dossier uittreksel	Tandartsdossier (behandeloverzicht)
Geboorteakte kind	Schooldiploma's / rapporten	Kinderbijslag-beschikking	Reisverzekeringspolis
Levensverzekering / uitvaartverzekering	Testament / notariële volmacht	**Verzekeringspolis Volvo V70**	

Waar ga je in je mappen ontwerp bijvoorbeeld de **verzekeringspolis van je Volvo V70** opslaan? Dat document past namelijk in meerdere categorieën, maar mag uiteindelijk maar **op één logische plek bestaan, lees 1 in 1 map**. Je zou deze polis kunnen opslaan:

• in de map **Polissen**, samen met al je andere verzekeringen.

• in de map **Volvo V70**, samen met alle autodocumenten van die auto

• in een map **Achmea**, zodat je per leverancier een volledig overzicht hebt wat je daar hebt afgesloten.

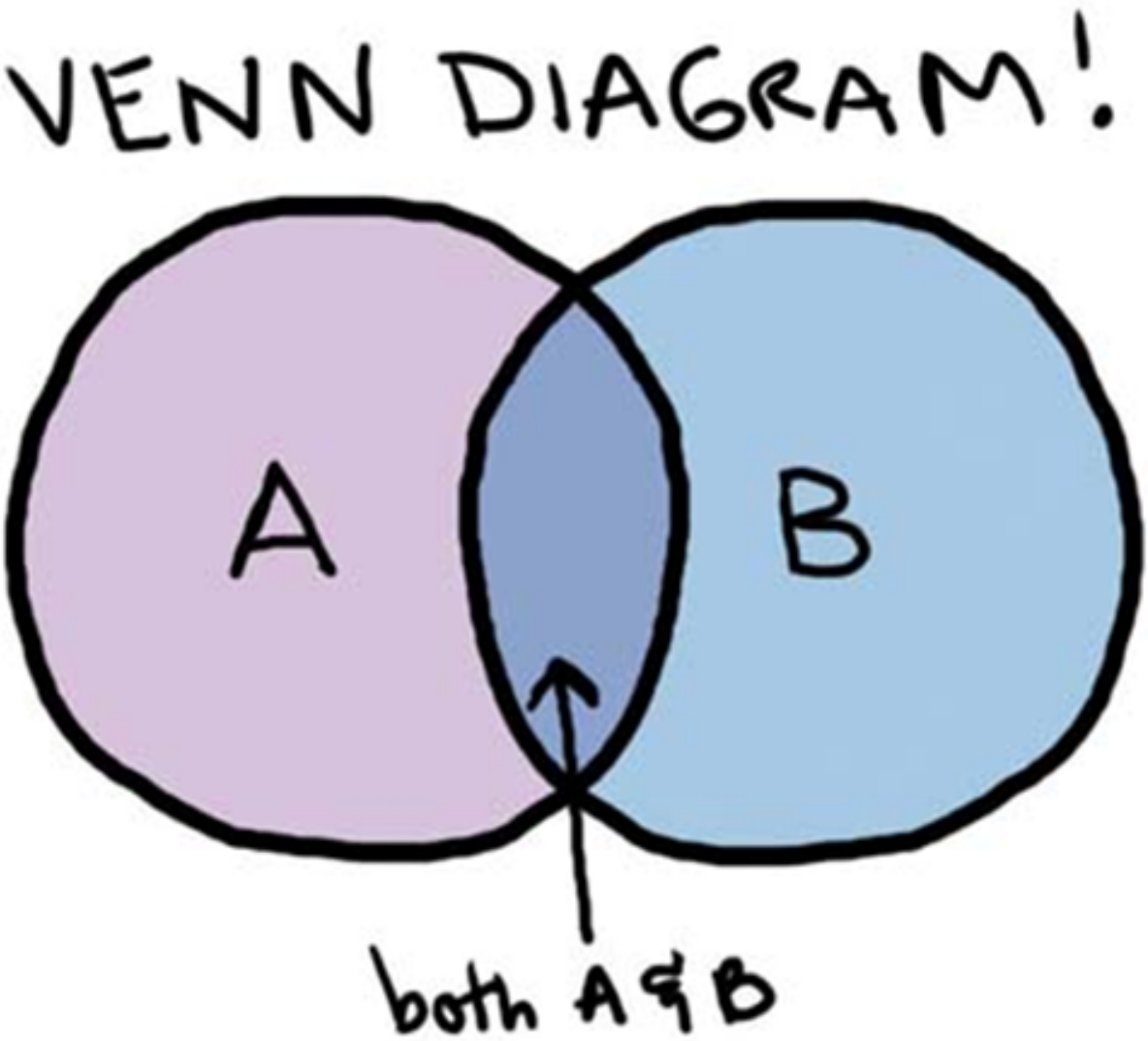
• Maak je die map dan aan op D:\Volvo V70, of is de Volvo een submap van een andere map. Welke en waarom, hoe ben je tot die ordening gekomen?

Precies dit soort twijfelgevallen maakt het ontwerpen van een goede mappenstructuur zeer lastig en snel complex. Privé is dat al ingewikkeld genoeg. In agile transformaties wordt het nog veel complexer, omdat niet alleen het volume toeneemt, maar ook de context, de afhankelijkheden en het aantal mogelijke invalshoeken (document kenmerken)

6. Twee denkmodellen achter een goede structuur

6.1. Venn-diagrammen: waarom mappenstructuren zo lastig zijn

Een Venn-diagram is een manier om verzamelingen en hun overlap visueel te tonen. Elke cirkel staat voor een categorie; de overlap tussen cirkels toont elementen die in meerdere categorieën passen.



Stel je drie cirkels voor als mappen:

Map A: alle verzekeringspolissen in één map. Dit is 100% zuiver

Map B: alle documenten over je Volvo V70. Dit is 100% zuiver

Map C: wat is Map C?

Nu ontstaat het interessante probleem. Neem bijvoorbeeld de autoverzekeringspolis van je Volvo V70. Dat document hoort tegelijk bij drie categorieën:

• het is een verzekeringspolis

• het is een autodocument

• het gaat specifiek over je Volvo V70

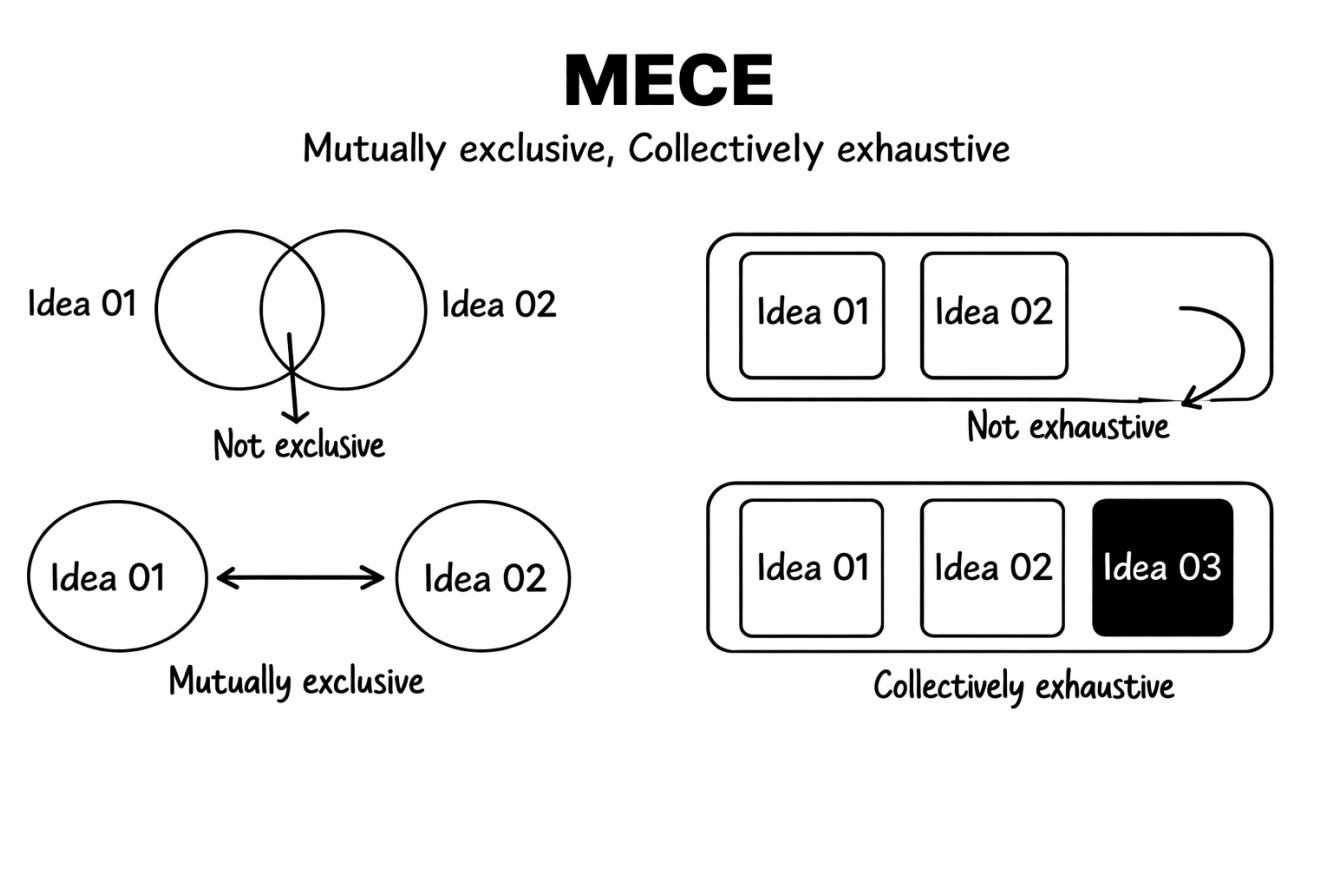
In een Venn-diagram ligt dat document dus precies in de overlap van alle drie de cirkels. Maar in een digitale mappenstructuur kan een bestand maar op één plek staan. Je kunt het document niet tegelijk opslaan in drie verschillende mappen zonder duplicaten of links te maken.

En hier ontstaat het kernprobleem van mappenstructuren: **documenten hebben meerdere kenmerken, maar mappen zijn enkelvoudig**. Zodra je categorieën gebruikt die elkaar overlappen, weet je nooit meer zeker waar een document thuishoort. De één zoekt het document onder "verzekeringen", de ander onder "auto", en weer iemand anders onder "Volvo". Dat is 1 van de kernproblemen van mappenstructuren.

De conclusie is daarom eenvoudig maar belangrijk: overlappende categorieën werken niet als hoofdstructuur. Je moet één ordeningsprincipe kiezen en dat consequent toepassen.

6.2. Het McKinsey MECE-principe

Het Venn-diagram laat zien waarom mappenstructuren zo lastig zijn. Documenten hebben meerdere kenmerken tegelijk. Een autoverzekeringspolis is bijvoorbeeld tegelijk een verzekering, een autodocument en een document over een specifieke auto. Maar een bestand kan maar op één plek staan hadden we al gezegd. Daarom moet je een structuur ontwerpen waarbij categorieën elkaar niet overlappen. Daarvoor gebruiken we het MECE-principe.



MECE toegelicht

MECE staat voor:

• Mutually Exclusive: categorieën mogen elkaar niet overlappen. Dus een document staat in Map A *of* in Map B.

• Collectively Exhaustive: samen moeten ze alles afdekken, wat betekent dat we een GMS moeten ontwerpen waar we *al onze (gezins) documenten* in kwijt kunnen.

Met andere woorden: elk document past in precies één categorie, en geen enkel document valt buiten de structuur. De relatie met het Venn-diagram is eenvoudig. Het Venn-diagram laat zien dat documenten meerdere eigenschappen hebben. Het MECE-principe dwingt je vervolgens om één van die eigenschappen te kiezen als hoofdstructuur.

Je kiest dus niet tegelijk voor:

• documenttype (verzekeringspolis, financiën)

• object (Volvo V70)

Je kiest één ordeningsprincipe als basis voor je mappenstructuur. De andere eigenschappen kun je vervolgens oplossen met submappen of met een goede bestandsnaam. Daarmee wordt de structuur voorspelbaar. Iedereen weet waar een document hoort, ook zonder te zoeken.

Korte denk oefening: Stel dat je een aantal documenten hebt met deze content: mannen, vrouwen, kinderen, transgenders, hermafrodieten en neutraal. Je moet hiervoor een mappenstructuur ontwerpen waarin elk document eenduidig kan worden opgeslagen.

Je besluit twee mappen te maken om deze documenten op je schijf te archiveren:

• c:\mannen

• c:\vrouwen

Dan ontstaat direct het probleem. Waar sla je het document "kind" op? En waar hoort een document met de inhoud "hermafrodiet" thuis?

Een korte MECE-check:

• **Man / vrouw mappen** Een man is geen vrouw en een vrouw is geen man. Samen vormen ze: alle volwassenen naar geslacht. ✓ MECE

• **Kind / volwassene:** Een kind is geen volwassene en een volwassene is geen kind. Een volwassene kan een man of vrouw zijn. Samen vormen ze: alle mensen naar leeftijdscategorie. ✓ MECE

• **Man / vrouw met extra categorieën:** Wanneer je mappen toevoegt zoals een map voor *kind*, *transgender* of *hermafrodiet*, ontstaat overlap: want, een kind kan immers zowel man als vrouw worden, een transgender kan biologisch één geslacht hebben maar zich met een ander identificeren, een hermafrodiet past niet eenduidig in man of vrouw. Deze categorieën overlappen dus en dus raken derden documenten kwijt. Waar heb jij dat opgeslagen? ✗ **Niet MECE**

Deze eenvoudige oefening laat precies zien waarom een structuur die niet MECE is direct tot onvindbaarheid leidt. Sommige documenten passen niet duidelijk in één map, terwijl andere documenten in meerdere mappen tegelijk zouden kunnen horen.

7. Toe te passen GMS ontwerpprincipes

7.1. Het levensarchief

LifeTree Architecture is ontworpen als een levenslang persoonlijk archief. Het doel is om het volledige privéleven van een gezin systematisch te bewaren zoals al je woningen, alle voertuigen (gereden en huidig), hypotheken, belastingaangiften, vermogensbeheer, diploma's, medische dossiers, individueel maar ook gezinsfoto's en -video's, aankopen en bijvoorbeeld vakantie informatie. De structuur vormt daarmee een digitale representatie van het privéleven van het gezin.

7.2. Scheid actieve documenten van historisch archief

In plaats van oude informatie te verwijderen, wordt geschiedenis expliciet bewaard voor bijvoorbeeld de belastingdienst, AIVD screening doeleinden maar ook ingeval van overlijden.

Daarom gebruiken we patronen zoals:

• Huidige woning / Historische woningen

• Huidige werkgever / Historische werkgevers

• Huidige school / Historische scholen

Hierdoor ontstaat een volledig historisch levenspad. Ingeval van een tekort aan schijfruimte, of geen interesse meer, kun je gewoon een paar historische mappen deleten.

7.3. Innovatie binnen je mappenstructuur

Een robuuste mappenstructuur ontstaat zelden perfect in één keer. Vooral in het begin (from scratch) zul je regelmatig moeten bijschaven, mappen verplaatsen of een structuur iets aanpassen. Dat is normaal. De eerste periode gebruik je de structuur intensief en ontdek je waar iets beter kan. Controleer continu hoe MECE je ontwerp is.

Na verloop van tijd stabiliseert je structuur. Uit mijn eigen ervaring blijkt dat wanneer het hoogste niveau en de eerste twee of drie niveaus goed zijn ontworpen, deze daarna vaak tientallen jaren vrijwel ongewijzigd blijven.

7.4. Zoek het hoogste aggregatieniveau

Bij het ontwerpen van een mappenstructuur moet je altijd eerst zoeken naar het hoogste logische aggregatieniveau. Op dat niveau probeer je de hoofdmappen zo MECE mogelijk te maken: Mutually Exclusive en Collectively Exhaustive. Pas daarna werk je verder naar lagere niveaus (lees: submappen) in de boom.

Veel mensen maken de fout om direct mappen aan te maken voor concrete dingen zoals "d:\Auto" of "d:\Hypotheek". Dat lijkt logisch, maar leidt bijna altijd en met name snel tot *chaos*. Binnen korte tijd ontstaan er overlappende mappen, dubbele documenten en onduidelijkheid over waar iets hoort. Daarom is de informatiehuishouding bij veel bedrijven zo'n bende.

Begin je GMS ontwerp altijd met een inventarisatie van je documenten en zoek naar een hoger abstractieniveau waar deze onder vallen. Neem bijvoorbeeld een auto. Je zou een map "Auto" kunnen maken. Maar een auto is eigenlijk onderdeel van een grotere categorie: voertuigen. En voertuigen zijn weer onderdeel van een nog hogere categorie: bezittingen of assets.

De vraag is dus steeds: wat is het hoogste logische niveau waar dit onder valt?

Voor een gemiddeld huishouden zou een eenvoudig en logisch hoogste niveau er bijvoorbeeld zo uit kunnen zien:

1. Fotos

2. Video’s

3. Geluid

4. Documenten

5. Assets

Wanneer dit hoogste niveau goed is gekozen en MECE is, kun je daarna pas beginnen met het verder uitwerken van de onderliggende mappenstructuur.

7.5. Perfecte MECE mappenstructuur bestaat niet

Hoewel je een GMS zoveel mogelijk MECE ontwerpt, is 100% MECE in de praktijk onhaalbaar. De reden is fundamenteel: documenten hebben (bijna) altijd meerdere kenmerken tegelijk, maar een mappenstructuur kent slechts één hiërarchie. Je kunt een bestand nu eenmaal maar op één plek opslaan.

Neem de factuur van je Volvo V70. Dat document is tegelijk een factuur, asset document en een document over een specifiek voertuig. Alle drie de categorieën zijn correct. Toch moet je kiezen: gaat het in de map 'Volvo V70', in 'Aankoop facturen 2026' of ergens anders? Er is geen objectief juist antwoord. Wat telt is dat je een keuze maakt en die consequent volhoudt.

Dit brengt ons bij een belangrijk principe dat hier direct uit volgt:

Kies één ordeningsprincipe en pas het consequent toe en onderwijs de mapgebruikers hierin.

Dus niet per map een andere logica, maar overal dezelfde redenering en dezelfde ordeningsprincipes.

7.6. Herhaalbare patronen maken je GMS goed schaalbaar

Veel onderdelen van het leven volgen dezelfde schaallogica. Denk aan gezinsleden, werkgevers, voertuigen, opleidingen, belastingjaren of woningen. Deze categorieën hebben vaak een vergelijkbare interne structuur. Daarom werken we in een Generieke Mappenstructuur met **herhaalbare mappatronen**. Wanneer een nieuw element ontstaat, hoef je de structuur niet opnieuw te ontwerpen. Je kopieert simpelweg een bestaand patroon en past de naam aan.

Voorbeelden:

• **Nieuw gezinslid:** Komt er een nieuw kind in het gezin, dan kopieer je de bestaande mappenstructuur van een ander gezinslid en hernoem je deze naar de naam van het nieuwe kind.

• **Nieuw belastingjaar:** Wanneer een nieuw fiscaal jaar begint, kopieer je bijvoorbeeld de map *Aangifte IB 2024* naar *Aangifte IB 2025*. De onderliggende structuur blijft intact en kan direct opnieuw gebruikt worden.

• **Nieuwe werkgever of opleiding:** Ook hier kun je een bestaand patroon kopiëren zodat contracten, correspondentie, beoordelingen en relevante documenten direct logisch geordend zijn.

Het grote voordeel hiervan is dat het systeem **meegroeit met je leven** zonder dat je telkens opnieuw hoeft na te denken over de structuur. En, je hoofdstructuur, de rootmappen blijven onaangetast.

7.7. Beperk padlengte door mapnamen niet te lang te maken

Windows kent een maximale lengte voor een volledig bestandspad. Zo'n pad bestaat uit de volledige route naar een bestand: alle mapnamen + de bestandsnaam zelf.

In oudere Windows-versies en in veel software geldt nog steeds de klassieke MAX_PATH-limiet van 260 tekens. Moderne versies van Windows 10 en Windows 11 kunnen technisch langere paden aan (tot ongeveer 32.767 tekens), maar veel programma's, backupsoftware en cloudopslag ondersteunen dit niet altijd volledig. In de praktijk is het daarom verstandig te ontwerpen alsof de 260-tekenslimiet nog steeds bestaat.

Daarom gebruikt een goed GMS ontwerp korte en compacte mapnamen, zoals *Diplomas*, *Medisch*, *Werk*, IB *2025*, Voertuigen

Door namen kort te houden blijft er ruimte over voor extra submappen en met name BNC-conforme bestandsnamen (die zijn namelijk langer dan gemiddeld), zonder dat het systeem tegen technische grenzen aanloopt en je windows errors genereert.

7.8. Nummer je mappen om de juiste volgorde af te dwingen

Besturingssystemen sorteren mappen standaard alfabetisch. Dat betekent dat de volgorde niet altijd overeenkomt met de logica die jij in je hoofd hebt. Door mappen te nummeren kun je de volgorde zelf bepalen. Daarom is het verstandig om een vast formaat te gebruiken, bijvoorbeeld:

• ###) mapnaam1

• ###) mapnaam2

In zowel privé- als organisatiesituaties zijn drie cijfers ruim voldoende, omdat je zelden meer dan 999 submappen onder de hoger liggende-map nodig hebt. Het voordeel daarvan is dat je later nog makkelijk een map ertussen kunt plaatsen, bijvoorbeeld 015) of 025), zonder dat je alle mappen opnieuw moet nummeren. Daardoor blijft de mappenboom overzichtelijk en logisch opgebouwd.

Tip: gebruik voor gedateerde mappen, zoals vakanties en werkgevers, het formaat YYMMDD, conform de BNC (hoofdstuk 63). Voor jaargebonden dossiers, zoals de aangifte IB, volstaat het jaartal. YYMMDD sorteert chronologisch zolang de datums binnen dezelfde eeuw vallen; over een eeuwgrens is die volgorde niet betrouwbaar. De sorteervoorbeelden in de bijlage gebruiken daarom datums binnen dezelfde eeuw.

7.9. Map maken of wens oplossen in de bestandsnaam?

Bij het ontwerpen van een mappenstructuur ontstaat regelmatig de vraag:

"moet ik hiervoor een nieuwe map maken, of kan ik dit beter oplossen in de bestandsnaam?"

Veel mensen maken automatisch een nieuwe map per document attribuut. Dat lijkt logisch, maar elke extra map maakt de structuur complexer. Na verloop van tijd ontstaat dan een steeds diepere boom met overlappende categorieën en verliest men het overzicht. Daarom is het verstandig eerst te bepalen of iets een **echte categorie** is, of slechts een **attribuut van een document**.

Neem het voorbeeld van **abonnementen**. Denk aan Vodafone, Netflix of een sportschool. Je zou hiervoor een map kunnen maken:

020) Financiën

└── 090) Abonnementen

Maar een abonnement is beter geen aparte documentcategorie. Het is meestal een **contract of financieel document (de factuur) met het kenmerk "abonnement"**. In dat geval kun je het document ook gewoon opslaan binnen je financiële administratie en het attribuut in de bestandsnaam verwerken.

Bijvoorbeeld:

240315 contract mobielabonnement -VODAF.pdf

240201 factuur Abonnement streaming -NETFL.pdf

230901 polis autoverzekering VolvoV70 -ACHME.pdf

Met zoektools zoals **Windows Search (F3)** of **Everything** kun je deze documenten vervolgens eenvoudig terugvinden door te zoeken op bijvoorbeeld:

abonnement.pdf

Het principe is dus eenvoudig:

• **Gebruik mappen voor stabiele categorieën.**

• **Gebruik bestandsnamen voor documentkenmerken (attributen).**

Hoe je zulke bestandsnamen systematisch opbouwt, bespreek ik in **hoofdstuk 63: de BestandsNaamConventie (BNC)**. Samen met de GMS vormt dit de tweede bouwsteen om documenten altijd snel terug te vinden.

7.10. Mapnamen zijn zelfverklarend en tijdloos

Een mapnaam is een belofte aan de toekomstige gebruiker over wat erin zit. Die gebruiker ben jij zelf, over tien jaar, of een gezinslid, of agile transformatie lid dat de structuur niet heeft ontworpen.

Vermijd daarom altijd namen als Nieuw, Oud, Diversen, Overig, Tijdelijk of Misc. Deze namen zeggen niets over de inhoud en groeien zonder uitzondering uit tot een vergaarbak. En dat willen we niet. Alles wat nergens anders past, belandt er vroeg of laat in. Daarmee ondermijnen ze precies wat je met een GMS wil bereiken.

Een goede mapnaam is kort in verband met max padlengte, concreet, contextonafhankelijk en moet begrijpelijk zijn zonder dat je de rest van de structuur hoeft te kennen. Denk aan: Voertuigen, Diplomas, Belasting, Investments, Werkgevers, Medisch: iedereen begrijpt direct wat erin zit, nu en over tien jaar.

7.11. Wijs de GMS Admin aan

Een GMS werkt alleen als de structuur stabiel blijft. Zodra meerdere mensen zonder spelregels mappen mogen aanmaken, groeit de structuur ongecontroleerd. Binnen korte tijd ontstaan parallelle mappen voor hetzelfde doel en verliest de GMS zijn voorspelbaarheid. De ontwerper weet vaak alles van de opbouw en wat waar en is getraind in MECE denken en doen, Venn-diagrammen.

Daarom geldt een eenvoudig principe: er is altijd één hoofd admin (binnen de familie) die de hoofdstructuur beheert en bewaakt. In een transformatie is dat bijvoorbeeld iemand uit de LACE, VMO of iemand van een Enabler team.

Een absolute spelregel is dat de rootstructuur, tot ongeveer mapniveau 3, nooit zonder toestemming wordt aangepast, die vormt de ruggengraat van het systeem. Lagere niveaus mogen beperkt worden uitgebreid, zolang ze het vastgestelde herhaalbare patroon volgen.

In agile transformaties is dit principe extra relevant. Teams maken al snel hun eigen mappenstructuren die logisch voelen vanuit hun eigen context, maar niet aansluiten op de bredere GMS. Daarom, één admin die de spelregels bewaakt voorkomt dit.

8. Kers op de taart: een uitgewerkte ~MECE prive GMS

Nu we alle ontwerpprincipes hebben behandeld, is het tijd voor de kers op de taart: een volledig uitgewerkte ~MECE privé GMS die je direct kunt gebruiken. Je vindt deze in de bijlage. Hieronder lopen we elk ontwerpprincipe één voor één langs en toetsen we of de GMS hieraan voldoet, en waarom we soms bewust afwijken.

8.1. Principe: het levensarchief

✓ Compliant. De GMS is ontworpen als een levenslang persoonlijk archief. De rootmap D:\ dekt het volledige privéleven van een gezin, als uitgangspunt van mijn voorbeeld gezin (!):

• personen (Vader, Moeder, kinderen), huisdieren,

• gezinsfinanciën,

• media en

• assets.

Er is geen einddatum, geen projectmap die vervalt, geen tijdelijke structuur. De structuur is ingericht om tientallen jaren mee te gaan en het volledige gezinsleven te bevatten, inclusief historische lagen.

8.2. Principe: Scheid actieve documenten van historisch archief

✓ Compliant. Dit principe is consequent doorgevoerd. Bij werk geldt het patroon

• 010) Huidige werkgever / 020) Historische werkgevers”

Bij school: “010) Huidige school / 020) Historische scholen”. Bij woning: “010) Huidige woning / Historische woningen Woning1, Woning2, Woning3”. Actief en historisch zijn zo altijd gescheiden, maar wel samenhangend bewaard. Oude informatie verdwijnt niet: ze krijgt een vaste historische plek.

8.3. Principe: Innovatie binnen je mappenstructuur

✓ Compliant. De GMS bevat expliciet een map “070) Gezins Innovaties” onder Administratie, met submappen Onderzoek, Offertes, Vergelijkingen en Besluiten. Dit is de plek waar een gezin nieuwe ideeën, aanschaffingen of verbeteringen doordenkt voordat ze een besluit nemen. Zo is innovatie (het continu bijstellen van je eigen informatiehuishouding) structureel verankerd in de GMS zelf.

8.4. Principe: Zoek het hoogste aggregatieniveau

✓ Compliant. De vier rootmappen van de GMS zijn: 010) Administratie, 020) Financiën, 030) Media en 040) Assets. Dit zijn de hoogst mogelijke abstractieniveaus voor een huishouden. Er staat geen map “Auto”, “Hypotheek” of “Achmea” op rootniveau; die zitten dieper in de boom, op de juiste logische plek. Door eerst het hoogste niveau te bepalen en pas daarna verder te verfijnen, blijft de hoofdstructuur stabiel en voorspelbaar.

8.5. Principe: Perfecte MECE mappenstructuur bestaat niet

✓ Bewust ~MECE. De naam van deze GMS heet niet voor niets “~MECE”. Er zijn grensvlakken in de structuur waar een document in meerdere mappen zou kunnen passen. Neem de verzekeringspolis van een auto: die past zowel onder 040) Assets > Roerend Goed > Voertuigen > Auto Vader als onder een eventuele polissenmap. De keuze in deze GMS is om alles rondom een specifiek object bij dat object te bewaren. Dat is een bewuste, consistente keuze, geen fout. Het principe geldt: kies één ordeningsprincipe en pas het consequent toe.

8.6. Principe: Herhaalbare patronen maken je GMS goed schaalbaar

✓ Compliant. De GMS werkt uitgebreid met herhaalbare patronen. De persoonsmappen volgen een vergelijkbaar basispatroon met Identiteit, Diplomas en Medisch. Bij Vader en Moeder wordt dit aangevuld met Werk en Pensioen; bij Kind1 en Kind2 met School en Vrije tijd. Elk historisch voertuig volgt hetzelfde patroon (Auto1, Auto2, Auto3). Bij de belastingaangiften wordt hetzelfde patroon per jaar herhaald; in de voorbeeldstructuur zijn 2024 en 2025 opgenomen. Vakantiefoto's worden per vakantie geordend, met een datum en bestemming in de mapnaam. Wanneer er een Kind3 bijkomt, kopieer je simpelweg de structuur van Kind1 of Kind2 en pas je de naam aan. De GMS groeit mee met het leven zonder herontwerp.

8.7. Principe: Beperk padlengte door mapnamen niet te lang te maken

✓ Compliant. Alle mapnamen in de GMS zijn kort en compact gehouden. Voorbeelden: Identiteit, Diplomas, Medisch, Werk, Contracten, Voertuigen, Witgoed, Bruingoed, Grijsgoed, Grond. Langere alternatieven zoals "Voertuigen en Transportmiddelen" of "Historische Werkgevers Archief" zijn vermeden. Ook de rootnamen zijn compact: Administratie (13 tekens), Media (5 tekens), Assets (6 tekens). Dit houdt het volledige pad ruim binnen de praktische Windows-limiet van 260 tekens, ook bij diepere submappen en lange BNC-bestandsnamen.

8.8. Principe: Nummer je mappen om de juiste volgorde af te dwingen

✓ Compliant. De GMS gebruikt het formaat "###) Mapnaam" voor inhoudelijk geordende categorieën. Jaarmappen en datummappen hebben hun eigen chronologische ordening en krijgen daarom geen extra volgnummer. Op rootniveau: 010) Administratie, 020) Financiën, 030) Media, 040) Assets. Binnen Administratie: 010) Vader, 020) Moeder, 030) Kind1, 040) Kind2, 050) Huisdier 1 enzovoort. Bij Vader en Moeder: 010) Identiteit, 020) Diplomas, 030) Medisch, 040) Werk, 050) Pensioen. Dit dwingt de logische volgorde af: de meest essentiële en meest geraadpleegde mappen staan bovenaan, onafhankelijk van alfabetische sortering door het besturingssysteem.

8.9. Principe: Map maken of wens oplossen in de bestandsnaam?

Bij het indelen van documenten maken we onderscheid tussen de vaste plek in de GMS en kenmerken die in de bestandsnaam thuishoren. Datum, documentsoort, onderwerp, versie en eigenaar kunnen via de BNC herkenbaar worden gemaakt, zonder voor elk kenmerk een extra map aan te maken. Zo toetsen we ook dit ontwerpprincipe: is een nieuwe map echt nodig, of kan de zoekvraag met een duidelijke bestandsnaam worden opgelost?

8.10. Principe: Mapnamen zijn zelfverklarend en tijdloos

✓ Compliant. Geen enkele mapnaam in de GMS heet "Diversen", "Overig", "Tijdelijk", "Nieuw" of "Misc". Alle namen zijn concreet en contextonafhankelijk: Identiteit, Diplomas, Medisch, Vaccinaties, Chip registratie, Witgoed, Bruingoed, Grondstoffen, Vastgoed investeringen, Jaaroverzichten broker. Een willekeurige gebruiker die deze structuur voor het eerst ziet, begrijpt direct wat er in elke map te vinden is, zonder uitleg en ook tien jaar later nog.

8.11. Principe: Wijs de GMS Admin aan

✓ Compliant. De GMS is ontworpen vanuit één ordeningsprincipe, met een stabiele rootstructuur die niet zomaar mag worden aangepast. In een gezinscontext is dit de “GMS Admin”, doorgaans degene die de structuur heeft ontworpen en begrijpt. De rootmappen (niveau 1 t/m 3) vormen de ruggengraat en worden niet zonder overleg gewijzigd. Lagere niveaus mogen beperkt worden uitgebreid, zolang het vastgestelde herhaalbare patroon wordt gevolgd. Voor wie deze GMS als startpunt gebruikt: wijs direct bij ingebruikname één persoon aan als beheerder.

De uitgewerkte GMS in de bijlage is hiermee getoetst aan alle elf ontwerpprincipes. Waar we bewust afwijken, zoals bij de ~MECE keuze om alles rondom een object bij dat object te bewaren, is dit een doordachte en consistente beslissing, geen vergissing. De structuur is vrij te gebruiken als startpunt voor je eigen privé informatiehuishouding. Pas de namen van gezinsleden, voertuigen en huisdieren aan naar je eigen situatie, en de GMS is direct inzetbaar.

9. Tot slot: tips voor een robuuste GMS

9.1. Scheid data van applicaties, zorg voor goede backups

Het is raadzaam om je programma- en systeemschijf (C:\) te scheiden van de schijf waarop je persoonlijke gegevens staan, vaak D:\. De reden hiervoor is eenvoudig. De C-schijf kan worden vervangen, mag crashen. Je programma’s en besturingssysteem zijn vervangbaar. Je gegevens zijn dat niet. Je documenten, foto’s, contracten en administratie moeten daarom op een aparte schijf staan die je zo veilig mogelijk beheert. Idealiter staat deze gegevensschijf op een redundante opslag, bijvoorbeeld een RAID-opstelling (zoals RAID-5 of RAID-1), zodat een defecte schijf niet direct dataverlies betekent.

Daarnaast hoort bij belangrijke gegevens altijd een tweeledige backupstrategie:

• een frequente online backup (bijvoorbeeld cloud of NAS-backup)

• een periodieke offline backup op een externe schijf die niet permanent met internet verbonden is (i.v.m. Crypto locking)

Die offline backup bewaar je bij voorkeur op een andere fysieke locatie of op een veilige plek in huis.

9.2. Automatisch de hele Lifetree op je PC zetten?

Wil je de volledige Lifetree GMS direct op je eigen computer hebben staan, zonder zelf alle mappen handmatig aan te moeten maken? Stuur me dan even een berichtje. Ik heb namelijk een klein programmaatje ontwikkeld dat de complete mappenstructuur automatisch aanmaakt op je desktop. Zo kun je direct aan de slag en de kracht van een goed ontworpen GMS in je eigen praktijk toetsen, zonder enige voorkennis of technische handelingen.

10. Samenvatting

Na het lezen van dit hoofdstuk weet je het volgende.

Documentchaos is geen toeval. In zowel privésituaties als grote Agile transformaties ontstaat informatiechaos vrijwel altijd door hetzelfde grondprobleem: documenten hebben meerdere kenmerken tegelijk, maar een mappenstructuur kent slechts één hiërarchie. Wie dit niet onderkent, bouwt onbewust een systeem dat zichzelf ondermijnt.

De oplossing bestaat uit twee bouwstenen: een Generieke Mappenstructuur (GMS) en een BestandsNaamConventie (BNC). Dit hoofdstuk behandelt de GMS. De BNC komt aan bod in hoofdstuk 63.

Een goede GMS ontwerp je op basis van elf principes: (1) richt de structuur in als levensarchief; (2) scheid actieve documenten van historisch archief; (3) geef innovatie een plek in de mappenstructuur; (4) kies het

hoogste logische aggregatieniveau; (5) accepteer dat een perfecte MECE-indeling in de praktijk niet altijd haalbaar is; (6) werk met herhaalbare patronen; (7) beperk de padlengte met korte mapnamen; (8) nummer mappen voor de gewenste volgorde; (9) los documentkenmerken waar mogelijk op in de bestandsnaam in plaats van extra mappen te maken; (10) kies zelfverklarende, tijdloze mapnamen; en (11) wijs één GMS Admin aan die de structuur bewaakt.

Dezelfde principes die thuis werken, werken ook in een Agile transformatie, net zoals in elke andere organisatie. De schaal verschilt, de logica niet. Een privé GMS met vier rootmappen dekt een heel gezinsleven. Een organisatie GMS met vergelijkbare principes dekt honderdduizenden documenten over meerdere portfolios, Solution Trains en teams.

Morgen direct beginnen? De volledig uitgewerkte ~MECE privé GMS staat in de bijlage van dit hoofdstuk, klaar om als startpunt te gebruiken. Pas de namen van gezinsleden, voertuigen en huisdieren aan naar je eigen situatie, en je GMS is direct inzetbaar. Wil je de mappenstructuur automatisch laten aanmaken op je eigen pc, stuur me dan even een berichtje (zie sectie 9.2).

11. de bijlage: een uitgewerkte ~MECE prive GMS

Onderstaand overzicht toont een voorbeeld van een **Generieke mappenstructuur** voor een huishouden. De structuur is logisch opgebouwd, MECE op hoofdniveau en bedoeld om tientallen jaren bruikbaar te blijven.

11.1. Hoofdniveau

Dit is het **hoogste aggregatieniveau** van de Generieke Mappenstructuur. Op dit niveau zijn de categorieën ontworpen volgens het MECE-principe (Mutually Exclusive, Collectively Exhaustive).

```
D:\
├── 010) Administratie
├── 020) Financiën
├── 030) Media
└── 040) Assets
```

11.2. Detailniveau

```
D:\
│
├── 010) Administratie
│   ├── 010) Vader
│   │   ├── 010) Identiteit
│   │   ├── 020) Diplomas
│   │   ├── 030) Medisch
│   │   ├── 040) Werk
│   │   │   ├── 010) Huidige werkgever
│   │   │   └── 020) Historische werkgevers
│   │   │       ├── 010315) Werkgever1
│   │   │       ├── 051101) Werkgever2
│   │   │       └── 140601) Werkgever3
│   │   └── 050) Pensioen
```

```
│   │
│   ├── 020) Moeder
│   │   ├── 010) Identiteit
│   │   ├── 020) Diplomas
│   │   ├── 030) Medisch
│   │   ├── 040) Werk
│   │   │   ├── 010) Huidige werkgever
│   │   │   └── 020) Historische werkgevers
│   │   │       ├── 010315) Werkgever1
│   │   │       ├── 030901) Werkgever1
│   │   │       ├── 051101) Werkgever2
│   │   │       ├── 070301) Werkgever2
│   │   │       ├── 140601) Werkgever3
│   │   │       └── 150901) Werkgever3
│   │   └── 050) Pensioen
│   │
│   ├── 030) Kind1
│   │   ├── 010) Identiteit
│   │   ├── 020) Diplomas
│   │   ├── 030) Medisch
│   │   ├── 040) School
│   │   │   ├── 010) Huidige school
│   │   │   ├── 020) Historische scholen
│   │   │   │   ├── 040901) Lagere school
│   │   │   │   ├── 100901) Middelbare school
│   │   │   │   └── 160901) Hogere school
│   │   │   └── 030) Cursussen
│   │   └── 050) Vrije tijd
│   │
│   ├── 040) Kind2
│   │   ├── 010) Identiteit
│   │   ├── 020) Diplomas
│   │   ├── 030) Medisch
│   │   ├── 040) School
│   │   │   ├── 010) Huidige school
│   │   │   ├── 020) Historische scholen
│   │   │   │   ├── 040901) Lagere school
│   │   │   │   ├── 100901) Middelbare school
│   │   │   │   └── 160901) Hogere school
```


030) Cursussen
050) Vrije tijd
050) Huisdier 1
010) Dierenarts
020) Vaccinaties
030) Verzekering
040) Chip registratie
050) Verzorging
070) Gezins Innovaties
010) Onderzoek
020) Offertes
030) Vergelijkingen
040) Besluiten
020) Financiën
010) Belastingen
010) Aangifte IB
010) Vader
2024
2025
020) Moeder
2024
2025
020) Toeslagen
010) Zorgtoeslag
020) Kinderbijslag
030) Huurtoeslag
020) Facturen
2024
2025
030) Notarieel
010) Testament
020) Volmachten
030) Huwelijk of samenleving
040) Erfrecht


040) Bankzaken
050) Beleggingen
010) ETF
020) Crowdfunding
030) Grondstoffen
040) Cryptos
050) Vastgoed
030) Media
010) Fotos
010) Vader
020) Moeder
030) Kind1
040) Kind2
050) Huisdier 1
060) Gezin
070) Vakanties
210403 Spanje
220504 Zweden
231023 Mexico
020) Videos
010) <bepaal zelf>
030) Geluid
010) <bepaal zelf>
040) Assets
010) Software
010) Installatie kits
020) Licenties
020) Hardware
010) Roerend Goed
010) Voertuigen
050301) Auto vader
010) Historische auto's
990410) Auto1

```
│   │   ├── 120615) Auto moeder
│   │   │   └── 010) Historische auto's
│   │   │       └── 120615) Auto1
│   │   ├── 150901) Aanhanger
│   │   └── 190315) Elektrische fiets
│   │
│   ├── 020) Witgoed
│   │   └── 010) Koelkast
│   │
│   ├── 030) Bruingoed
│   │   ├── 010) Televisie
│   │   └── 020) Platenspeler
│   │
│   └── 040) Grijsgoed
│       ├── 010) Laptop
│       ├── 020) Netwerk
│       └── 030) Randapparatuur
│
└── 020) Onroerend Goed
    ├── 010) Huidige woning
    │   └── 010) Historie
    │       ├── 030614) Woning2
    │       └── 181205) Woning3
    │
    ├── 020) Vakantieverblijf
    └── 030) Grond
```

63. Documentchaos in je Agile transformatie? Een slim ontworpen BNC lost het op (deel 2)

1.1. De bestandsnaam als communicatieafspraak: simpel, collectief en direct nuttig

2. Inleiding

In het vorige hoofdstuk over de Generieke Mappenstructuur zagen we hoe een goed ontworpen Generieke Mappenstructuur (GMS) voorkomt dat documenten verdwijnen in de verkeerde map. Maar een mappenstructuur alleen is niet genoeg.

Wie een SAFe-transformatie start, weet niet altijd wat er documentair op hem afkomt. Een middelgrote transformatie met meerdere Agile Release Trains, een LACE, portfoliomanagement en betrokken lijnorganisaties produceert al snel meer dan 100.000 documenten binnen 2 jaar tijd met zomaar >1000 mappen. Visiedocumenten, beleidsdocumenten, Target Operating Models, planningen, roadmaps, requirements, architectuurbeslissingen, HRM zaken, suppliers zaken, contracten, facturen, het stapelt zich razendsnel op.

Zonder een eenduidig afsprakensysteem voor bestandsnaamgeving ben je simpelweg verloren. Je moet weten wie een document heeft gemaakt, welke versie de actuele is, en wanneer het document voor het laatst is bijgewerkt. Drie vragen, de bestandsnaam moet ze alle drie kunnen beantwoorden.

De BNC (BestandsNaamConventie) is de tweede bouwsteen van dit zelfbedachte concept, al bewezen in de praktijk sinds 1999, en biedt hiervoor een simpele en doeltreffende oplossing.

3. Het probleem: mappen zijn niet genoeg

De meeste organisaties vertrouwen op een mappenstructuur om documenten terug te vinden. Mappen zijn nuttig: ze zijn ondersteunend als je een document écht kwijt bent. Maar ze zijn niet de primaire oplossing.

Een goede bestandsnaamconventie plus goede, en vooral snelle, tooling stelt je in staat om tot 99% van je documenten te vinden zonder ook maar één map te openen. Je zoekt direct op naam middels een query (in de verkenner of via een gespecialiseerde zoektool) en je bent er. Mappen zijn het vangnet. De bestandsnaam is het systeem.

De praktijk zonder BNC ziet er zo uit:

• Niemand weet welke versie de meest recente is.

• Versies liggen in persoonlijke archieven.

• Zoeken levert niets op omdat sleutelwoorden niet in de bestandsnaam zitten.

• Documenten worden onnodig opnieuw aangemaakt omdat het origineel 'niet te vinden' is.

• Status en eigenaarschap zijn onduidelijk: wie heeft dit gemaakt? Is dit concept of definitief?

Dit zijn geen incidenten, dit is structureel, iets wat veel lezers zullen herkennen. In een SAFe-omgeving met honderden betrokkenen en tienduizenden documenten per jaar heeft het een serieuze prijs: verloren productiviteit, frustratie, en het risico op besluitvorming op basis van verouderde informatie.

4. De oplossing: de BNC-structuur

De BNC lost dit op door elk bestand een vaste, opgebouwde naam te geven op basis van zes eenvoudige indexen:

Index	Format / voorbeeld	Toelichting
[DATE]	221101	Datum in YYMMDD-formaat
[Type]	Invoice, Planning, Notulen…	Documentsoort, één woord
[Subject]	Migratie datacenter	Compacte beschrijving van de inhoud
-[V#.#]	-V2.1, -VDEF, -VBSL	Versie of status (optioneel)
-[OWNER]	-PEELR, -GATEB	Achternaam (4) + initiaal voornaam, hoofdletters
.EXTENSIE	.docx, .pdf, .xlsx	Bestandstype

Elke index wordt gescheiden door één spatie. Elke vaste code (versie, eigenaar) wordt voorafgegaan door een koppelteken. Een correcte bestandsnaam ziet er zo uit:

De conventie in het kort

[YYMMDD] [TYPE] [Subject] [-V#.#] [-OWNER].[EXT]

Voorbeelden

• 260311 Invoice SAFe Training POPM -VDEF -GLOBK.pdf

• 260215 Planning PI-12 ART Logistiek -V1.0 -JANSS.xlsx

• 260101 Minutes Retrospective Sprint 5 Team Falcon -VDEF -SMITK.docx

• 251203 Vision SAFe Transformatie 2026 -V2.1 -PEELR.pptx

• 250918 Contract Leverancier Azure Services -VBSL -MICRO.pdf

• 260305 Notulen Stuurgroep Q1 2026 -VDEF -BRUIJ.docx

• 251115 Risicolog ART Platform -V3.0 -PEELR.xlsx

• 260210 Factuur-in Trainingsprogramma POPM -VDEF -SMITK.pdf

• 260211 Factuur-uit Trainingsprogramma LPM -VDEF -SMITK.pdf

Het grote voordeel is dat je harde schijf er altijd keurig en geordend uitziet.

4.1. Waarom YYMMDD vooraan staat

Dit is een bewuste keuze, en een belangrijke. Door de datum **vooraan** te zetten, sorteert je verkenner automatisch chronologisch. Zonder enige extra handeling zie je direct welk document het nieuwst is en heb je meer zekerheid ‘wat de laatste versie is’. Dit is cruciaal wanneer je werkt met documenten die meerdere versies kennen over langere periodes.

De volgorde **YYMMDD** (jaar-maand-dag) in plaats van de gangbare Europese notatie DD-MM-YY heeft één doorslaggevend voordeel: **alfabetische sortering is gelijk aan chronologische sortering**. Schrijf je 01-11-22, dan sorteert de verkenner op dag, en staat november 2022 tussen januari 2019 en december 2023. Schrijf je 221101, dan staat alles uit 2022 netjes bij elkaar, gesorteerd per maand en dag. Het systeem werkt vanzelf, zonder dat je ooit handmatig hoeft te sorteren.

Note: We kiezen bewust voor YYMMDD in plaats van YYYYMMDD; de millenniumovergang is geen issue meer en het spaart tekens in de bestandsnaam.

5. Zoeken zonder mappen: twee manieren

5.1. Methode 1: Windows Verkenner met PC-indexering (F3)

Windows heeft een ingebouwde zoekfunctie die werkt via de verkenner (F3 of het zoekvak). Om dit goed te laten werken, moet je schijf geïndexeerd zijn. Zo stel je dat in:

1. Ga naar **Configuratiescherm** → **Indexeringsopties**

2. Klik op **Wijzigen** en voeg de mappen toe die je wilt indexeren (bijv. je hele documentenmap of SharePoint-synchronisatiemap)

3. Klik op **Geavanceerd** → **Nu opnieuw opbouwen** om de index te vernieuwen

4. Wacht tot de indexering klaar is (kan enkele uren duren bij grote volumes)

Zodra de index klaar is, kun je via F3 zoeken **met de windows wildcards**:

Zoekopdracht	Wat je vindt
08*off*shel*.doc	Het offerte-document richting Shell uit 2008
03*cours*	Alle cursus-gerelateerde documenten uit 2003
*plan*mark*.doc	Elk marketingplan, ongeacht datum of eigenaar
08*-GATEB*.*	Alles wat Bill Gates heeft aangeleverd in 2008
*factuur-in*BOL*	Alle facturen van bol.com

5.2. Methode 2: gebruik “Everything” (aanbevolen voor grote volumes)

Voor wie echt grote hoeveelheden documenten beheert (en in een SAFe-programma is dat al snel het geval) is **Everything** van Voidtools de betere keuze met de kanttekening dat hiervoor een lokale installatie nodig is. Everything is een gratis zoektool die de volledige mapstructuur en bestandsnamen van je schijf inleest en indexeert.

Het resultaat: je kunt door **meer dan een miljoen documenten zoeken in minder dan een seconde**. Geen wachttijd, geen spinner, geen frustratie.

• Download via: https://www.voidtools.com

• Installeer en start: Everything indexeert automatisch bij de eerste start

• Kies bij settings de juiste schijf of de juiste mappen die je wel en niet wilt doorzoeken

• Typ je zoekopdracht (wildcards werken standaard): bijv. *invoice*skillsoft*

• Resultaten verschijnen real-time terwijl je typt

Super handig!!

Everything integreert ook met de Windows verkenner en heeft een optionele HTTP-server voor gedeeld gebruik in teams. Voor iedereen die serieus met documentbeheer bezig is, is dit de standaardtool.

note: Locate32 is een gratis alternatief voor Everything waarmee je snel bestanden en mappen kunt zoeken op basis van bestandsnaam; download via: https://sourceforge.net/projects/locate32/

6. Waarom dit in een SAFe-implementatie onmisbaar is

Een SAFe-transformatie is documentintensief per definitie. Zonder BNC verlies je de grip.

Met BNC:

• Zie je direct **wanneer** het document is gemaakt of voor het laatst bijgewerkt (DATE-index)

• Weet je altijd **wie verantwoordelijk is (OWNER-index) en wie je moet bellen** voor de inhoud

• Ken je de content al min of meer zonder het bestand te openen

• Ken je de **status** zonder het bestand te openen (-VDEF, -V2.1)

• Kun je **raak zoeken** omdat altijd minimaal één index bekend is

De BNC is geen bureaucratisch instrument. Het is een **communicatieafspraak**, vergelijkbaar met een branching-strategie in softwareontwikkeling of een PI-cadans in SAFe. Simpel, collectief, en onmiddellijk nuttig. En het werkt alleen als iedereen meedoet: dat is de enige randvoorwaarde.

7. Samenvatting

In hoofdstuk 62 behandelden we de Generieke Mappenstructuur (GMS), de logische structuur waarin documenten worden opgeslagen. De GMS werd uitgelegd aan de hand van een privéhuishouden: herkenbaar, concreet en direct toepasbaar. Dezelfde principes, MECE-denken, herhaalbare patronen, historisch archief scheiden van actief archief gelden zonder uitzondering ook in een Agile transformatie. De schaal verschilt, de logica niet.

Dit hoofdstuk voegt de tweede bouwsteen toe: de BestandsNaamConventie (BNC). Waar de GMS bepaalt wáár een document staat, bepaalt de BNC hoe een document heet zodat het ook buiten zijn map herkenbaar en vindbaar blijft. Samen vormen GMS & BNC een **eenvoudig maar krachtig documentmanagementsysteem** waarmee je documenten niet alleen opslaat, maar met name altijd weer terugvindt.

Net als thuis geldt ook in SAFe-transformaties: de conventie werkt het beste als iedereen meedoet. Dat is de enige randvoorwaarde en tegelijk de grootste uitdaging. Wijs daarom ook voor de BNC een beheerder aan, leg de afspraken vast, en train nieuwe teamleden actief in gebruik, coach, help, leg uit, monitor.

Een goede GMS zonder BNC is als een goed geordend archief zonder etiketjes op de dozen. Samen maken ze het verschil.

64. Werknemer van de maand. Klinkt leuk. Maar is het ook slim?



Stel je voor: het is maandagochtend. De teamlead loopt naar voren met een glimlach en een ingelijst certificaat. “En de werknemer van de maand is… Sarah!” Applaus. Sarah lacht verlegen. Drie rijen verder staart Mark naar zijn bureau. Hij weet dat hij afgelopen maand het werk van twee collega’s heeft opgevangen, een kritieke release heeft gered en elke dag als eerste aanwezig was. Maar Mark is niet zo zichtbaar als Sarah. Mark praat niet zo veel op meetings. Mark wint nooit. Sarah is extrovert, Mark introvert.

Dit tafereel speelde zich meer dan eens af tijdens mijn jaren bij grote internationale IT-organisaties. En elke keer zag ik hetzelfde patroon: een moment van erkenning voor één persoon, gevolgd door stille frustratie bij een handvol anderen. Niet omdat die mensen jaloers waren. Maar omdat ze voelden dat het systeem niet klopte en dat er achter de schermen andere ideeën leefden over wie het had moeten worden.

Het echte probleem zit niet in de bedoeling, maar in het instrument

Wie een Employee of the Month-systeem introduceert, heeft meestal goede intenties. Meer motivatie, hogere betrokkenheid, een sterkere cultuur. Dat zijn legitieme doelen. De vraag is alleen of dit instrument daar het juiste antwoord op is.

In sommige omgevingen werkt het prima. In een pure salesorganisatie waar prestaties **objectief en individueel meetbaar zijn**, is het klip en klaar wie bovenaan staat. Immers, cijfers liegen niet! Maar in de meeste moderne (SAFe) organisaties ligt dat heel anders.

Steeds meer werk ontstaat door samenwerking, helemaal in SAFe omgevingen. Resultaten zijn het product van gedeelde verantwoordelijkheid, onderlinge afhankelijkheden en collectieve inspanning. Juist in die context roept een individueel prijsmechanisme een fundamentele vraag op: beloon je de persoon die het meest zichtbaar is, of degene die echt de meeste waarde toevoegt? En hoe weet je dat eigenlijk zeker?

Zodra de criteria voor werknemer van de maand in een niet-commerciële omgeving niet glashelder zijn, en dat zijn ze zelden, ontstaat er ruimte voor subjectiviteit. En subjectiviteit in beloningssystemen is een voedingsbodem

voor precies de dingen die je als organisatie niet wilt: interne competitie, jaloezie, verminderd vertrouwen en een subtiele maar schadelijke verschuiving in gedrag. Mensen richten zich niet meer op wat goed is voor het geheel, maar op **wat nodig is om zelf op te vallen.**

Moderne organisatievormen zoals SAFe draaien **niet om het optimaliseren van één individu**. Ze draaien om het verbeteren **van het systeem als geheel conform SAFe's Systems Thinking**. De vraag is niet "wie was deze maand de held?" maar veel beter "welk agile team of ART heeft als geheel gepresteerd, geleerd en de meeste bruikbare (cijfermatig aantoonbare) waarde geleverd?"

Een individueel prijsmechanisme staat haaks op dat principe. Het suggereert dat succes een solistische prestatie is, terwijl het in de praktijk altijd een teamprestatie is.

Individuele waardering is niet verkeerd. Integendeel. Mensen persoonlijk bedanken, successen zichtbaar maken en oprechte erkenning uitspreken is essentieel en je kunt het niet vaak genoeg doen. Maar doe het bewust, tijdig en vooral persoonlijk, 1 op 1, niet als maandelijks plenair zeepkisten HR-ritueel met een ingelijst certificaat.

Een concreet alternatief dat ik organisaties aanraad: verschuif de erkenning van individu naar team. Vier teamprestaties publiekelijk. Benoem specifiek welk gedrag of welke samenwerking het verschil maakte. En geef individuele waardering direct, op het moment dat iemand iets bijzonders doet, niet vier weken later bij de koffieautomaat en ook niet plenair. Iedereen vindt schouderklopjes van zijn meerdere leuk.

Afsluitend

Erkenning werkt pas echt als mensen het ervaren als eerlijk, passend en betekenisvol. Een systeem dat dat niet garandeert, creëert applaus aan de voorkant en frustratie achter de schermen. De vraag is dus niet of je mensen moet waarderen. Natuurlijk wel. De vraag is of je dat met dit instrument op de juiste manier doet.

Ing. R.P. Peels, SAFe expert & Authorised trainer | safe@majormilestones.nl

OVER DE AUTEUR

ing. R.P. Peels
Certified SAFe® Implementatie Consultant & Authorised Trainer

Roel Peels combineert diepgaande en excellente kennis van het Scaled Agile Framework (afgekort SAFe®) met meer dan veertig jaar praktijkervaring in project-, programma- en agilemanagement. Waar veel auteurs over digitale transformaties schrijven vanuit theorie, werkt Peels al meer dan 15 jaar midden in de praktijk van grootschalige Lean-Agile SAFe®-implementaties bij grote complexe organisaties.

Hij is twaalfvoudig SAFe®-gecertificeerd en vervulde in zijn carrière rollen als CEO, Founder, Startup-expert, Lean Portfolio Management practitioner (LPM), Release Train Engineer (RTE), SAFe® Program Consultant (SPC) en Executive Agile Coach. Zijn achtergrond als werktuigbouwkundige en technisch bedrijfskundige, gecombineerd met een interimmanagementopleiding aan onder meer de Erasmus Universiteit, geeft hem een zeldzame combinatie van technisch inzicht, bedrijfskundige scherpte en organisatorisch leiderschap.

Wat Roel onderscheidt is zijn vermogen om op meerdere niveaus tegelijk te opereren. Hij kan denken in Enterprise-strategieën, missies en visies op directieniveau, maar net zo goed zelf de vertaalslag maken naar de operatie op de werkvloer. Van portfolio-sturing en governance tot het daadwerkelijk inrichten en gebruiken van tools zoals Jira en Jira Align: Roel combineert strategisch denken met hands-on uitvoering.

Hij begeleidde SAFe®-transformaties bij organisaties zoals ASML, Coca-Cola, Sanoma, het Ministerie van Justitie en het Ministerie van Defensie. Bij de laatste organisatie leidde hij als voorzitter van het SAFe® Lean-Agile Center of Excellence (LACE) de transformatie van Defensie Landgebonden IT, met aantoonbare resultaten en brede steun van zowel Business Owners als Executive Management.

Zijn persoonlijke drijfveren — sterke prestatiegerichtheid, scherpe focus, leergierigheid en een krachtige communicatiestijl — maken hem tot een interimmanager, trainer, adviseur en auteur die complexe materie begrijpelijk maakt en organisaties helpt om daadwerkelijk te veranderen in plaats van er alleen over te praten.

KLAAR VOOR DE VOLGENDE STAP?

Over een SAFe®-transformatie lezen is één ding. Er één leiden is iets anders. Major Milestones B.V. begeleidt organisaties van strategie tot executie — geen standaardaanpak, maar maatwerk gebaseerd op wat in de praktijk werkt.

→ Enterprise Agile & SAFe® transformaties (strategie tot uitvoering)
→ Leadership enablement & SAFe® executive training
→ Interim transformatieleiderschap (LACE, RTE, Agile leadership)
→ Value Stream design & Jira / Jira Align operationalisatie
→ Keynotes & masterclasses over Lean-Agile leadership

Neem contact op via LinkedIn voor een vrijblijvend gesprek.

Major Milestones B.V.
linkedin.com/in/roelpeels