

# Een overzicht van vier veelgebruikte agile delivery modellen

30 augustus 2016 Consultancy.nl

*Als gevolg van de toenemende focus op agility ('wendbaarheid') in de afgelopen jaren zijn er anno 2016 tientallen agile change modellen waar managers en IT'ers uit kunnen kiezen om hun agile ambities vorm te geven. Rick van Benten, Managing Partner van Towson, een adviesbureau gespecialiseerd in verandering op basis van agile principes, geeft een overzicht van vier veel gebruikte agile delivery raamwerken en licht per model de belangrijkste eigenschappen toe.*

## Introductie

De maatschappij, klantbehoeftes en -verwachtingen veranderen voortdurend. Kwaliteit- en veiligheidseisen worden strenger, wet- en regelgeving worden geharmoniseerd, en technologische ontwikkelingen volgen elkaar in rap tempo op. Tegelijkertijd worden de kwantiteit, snelheid en complexiteit van veranderingen steeds groter.

Overheden, onderwijsinstellingen, bedrijven en medewerkers zullen de komende jaren geconfronteerd worden met kleinschalige en grootschalige veranderingen en een omgeving waarin maatschappelijke, economische en technologische veranderingen leiden tot de intrede van nieuwe marktpartijen, nieuwe markt-productcombinaties, innovatieve diensten en verschuiving van bestaande machtsstructuren.

Dit vergt aanpassingsvermogen en vraagt om een sterke klant- en productfocus, flexibiliteit en lenigheid. Om te kunnen overleven dienen organisaties in staat te zijn om effectief en tijdig te reageren op veranderingen, kansen en bedreigingen binnen bestaande en toekomstige markten. In de praktijk blijkt dit vaak een uitdaging en is het voor organisaties lastig om blijvende veranderingen door te voeren. Traditionele programma- en projectmanagement technieken zoals MSP en PRINCE2 blijken niet altijd het gewenste resultaat op te leveren en voldoende uitkomst te bieden.



In reactie hierop zijn vanaf de jaren '90 verschillende agile change modellen ontwikkeld waaronder Rational Unified Proces (RUP), Scrum, Nexus, Large Scale Scrum (LeSS), Scaled Agile

Framework (SAFe) en Disciplined Agile Delivery (DaD). Dergelijke modellen en methodieken stellen organisaties in staat om tijdig, effectief en efficiënt veranderingen door te voeren door hun lenigheid te vergroten.

Veel organisaties zijn inmiddels overtuigd van het agile gedachtegoed en gestart met het werken volgens de agile en Scrum principes. Bekende voorbeelden zijn Spotify, Coolblue, ING, Eneco en bol.com waar meerdere scrumteams gezamenlijk werken aan de succesvolle introductie van nieuwe producten en diensten om steeds meer toegevoegde waarde te leveren aan hun klanten, stakeholders en eindgebruikers.

Voor veel organisaties is het een uitdaging om de agile principes toe te passen en geleidelijk de transitie te maken naar een volwassen agile organisatie. Daarnaast is het meestal een uitdaging om continu de volwassenheid en effectiviteit van de agile organisatie te verbeteren en met meerdere agile teams tegelijkertijd producten en diensten te ontwikkelen.

## **Het Agile Manifesto**

In de jaren '90 werden organisaties en bedrijven steeds ICT-intensiever en steeds vaker geconfronteerd met technologische veranderingen, veeleisende klanten en een steeds korter wordende time-to-market van producten. Al snel bleek dat traditionele projectmanagement technieken onvoldoende flexibel waren om efficiënt en effectief proces- en systeemwijzigingen door te voeren.

In reactie hierop deden verschillende light-weight development technieken hun intrede, waaronder Dynamic System Programming Method (DSDM), Extreme Programming (XP), Rapid Application Development (RAD), Feature-driven Development (FDD) en Rational Unified Proces (RUP). De methodieken waren gericht op het incrementeel en iteratief ontwikkelen van software op basis van de behoefte van de klant en hadden een sterke focus op continu verbeteren en kwaliteit.

In 2001 kwamen 17 software-pioniers bij elkaar om de verschillende light-weight software technieken te bespreken en te vergelijken. De uitkomst van de meet-up was het Agile Manifesto waarin de 12 uitgangspunten van agile werken zijn vastgelegd die organisaties kunnen gebruiken om hun lenigheid, flexibiliteit en aanpassingsvermogen te trainen en te vergroten.

1. Customer satisfaction by early and continuous delivery of valuable software
2. Welcome changing requirements, even in late development
3. Working software is delivered frequently (weeks rather than months)
4. Close, daily cooperation between business, people and developers
5. Projects are built around motivated individuals, who should be trusted
6. Face-to-face conversation is the best form of communication (co-location)
7. Working software is the principal measure of progress
8. Sustainable development, able to maintain a constant pace
9. Continuous attention to technical excellence and good design
10. Simplicity—the art of maximizing the amount of work not done—is essential
11. Best architectures, requirements, and designs emerge from self-organizing teams
12. Regularly, the team reflects on how to become more effective, and adjusts accordingly



## Het Agile Manifesto: 12 uitgangspunten van agile werken

Samenvattend verkiest het Agile Manifesto individuen en interactie boven proces en tooling, werkende software boven documentatie, samenwerking met de klant boven contractonderhandelingen en het reageren op verandering in plaats van het volgen van een plan.

### **Single Team Scrum**

Het Agile Manifesto heeft de deur geopend voor nieuwe delivery methoden om in een complexe omgeving veranderingen door te voeren, waaronder Scrum de meest toegepaste en gebruikte methode is. Bedrijven zoals Spotify, Coolblue, ING, Eneco en bol.com maken dankbaar gebruik van Scrum om hun producten en dienstverlening continu te verbeteren.

Scrum is in de jaren '90 bedacht door Ken Schwaber en Jeff Sutherland met als doel om op een efficiënte en effectieve manier software te ontwikkelen. De term Scrum is afkomstig uit de rugbywereld en is een proces framework dat gebruikt wordt om complexe producten en software te ontwikkelen in een continu veranderende omgeving.

Het framework is gebaseerd op teamwork. Het Scrum Team bestaat uit een Product Owner, Scrum Master en Development Team en is gezamenlijk verantwoordelijk voor het realiseren van werkende producten en software die voldoen aan de wensen van de klant. Om de kwaliteit van de producten en software te borgen stelt het team een Definition of Done (DoD) op. In de DoD beschrijft het team de eisen waaraan producten en software moeten voldoen voordat ze in gebruik kunnen worden genomen. Hierbij kan gedacht worden aan de uitvoering van testwerkzaamheden, het documenteren van functionaliteiten en het voldoen aan interne security richtlijnen van de organisatie.

Het Scrum Team is zelf-organiserend, werkt iteratief, incrementeel en levert volgens een vast ritme, de sprint, een werkend product en werkende software op. De sprint is fixed en mag niet meer dan maximaal 4 weken duren. Transparantie, observeren en aanpassen vormen tijdens de sprint de kernwaarden van het team.

Binnen het Scrum Team is de Product Owner verantwoordelijk voor het inventariseren en prioriteren van de te realiseren producten en wijzigingen. De producten en wijzigingen worden centraal vastgelegd op een prioriteitenlijst de Product Backlog genaamd. In overleg met de

omgeving en betrokken stakeholders bepaalt de Product Owner in welke volgorde de producten en wijzigingen door het team worden opgepakt. De Scrum Master ondersteunt de Product Owner in zijn rol en is verantwoordelijk voor het wegnemen van issues en het ondersteunen en coachen van het team.

Aan het begin van elke sprint vindt er een Sprint Planning plaats. In dit overleg worden de richting en focus van de sprint bepaald en stelt het team een sprintdoel op. Daarnaast wordt de lijst met geprioriteerde producten en wijzigingen besproken en bepaald wat het team gaat oppakken in de sprint. Vervolgens wordt besproken hoe de wijzigingen worden opgepakt. De uitkomst van dit overleg is de Sprint Backlog: een korte lijst met geprioriteerde wijzigingen en taken op basis waarvan het team start met de realisatie van de producten en wijzigingen. De Sprint Backlog is eigendom van het Development Team en kan tijdens de sprint niet gewijzigd worden door de Product Owner. Gedurende de sprint wordt dagelijks de voortgang besproken in een Daily Scrum aan de hand van drie vragen: 1. Wat heb je gister gedaan?; 2. Wat ga je vandaag doen?; 3. Welke issues heb je?

Aan het eind van elke sprint worden tijdens de Sprint Review de producten en wijzigingen getoond aan de klant, stakeholders en eindgebruikers, en wordt de sprint afgesloten met een Sprint Retrospective. In dit overleg wordt de sprint door het team geëvalueerd en worden verbeterpunten benoemd die worden meegenomen in de volgende sprint. Op deze wijze kan het team zich continu verbeteren.



## Multiple Team Scrum

Het agile gedachtegoed wordt inmiddels door veel organisaties omarmd en Scrum wordt in veel organisaties succesvol toegepast. Het op korte termijn starten met agile en Scrum blijkt in de praktijk voor organisaties relatief snel veel voordelen op te leveren ten opzichte van traditionele projectmanagement- en delivery technieken, waaronder het gericht en regelmatig opleveren van werkende software.

Het succes van Scrum heeft geleid tot een toename van het aantal Scrum Teams binnen tal van organisaties en van de behoefte om tegelijkertijd met meerdere Scrum Teams producten en software te ontwikkelen. In de praktijk is echter gebleken dat het niet zonder meer mogelijk is om op te schalen en dat de uitgangspunten van het Agile Manifesto tezamen met de spelregels van Scrum niet altijd voldoende uitkomst bieden.

In reactie hierop zijn de afgelopen jaren in korte tijd een aantal agile frameworks ontwikkeld waaronder Nexus, Large Scale Scrum (LeSS), Scaled Agile Framework (SAFe) en Disciplined Agile Delivery (DaD). De frameworks zijn bedoeld om organisaties handvatten te bieden voor het opschalen en doorontwikkelen naar een volwassen agile organisatie die in staat is om met meerdere teams tegelijkertijd agile te kunnen werken en producten te ontwikkelen.



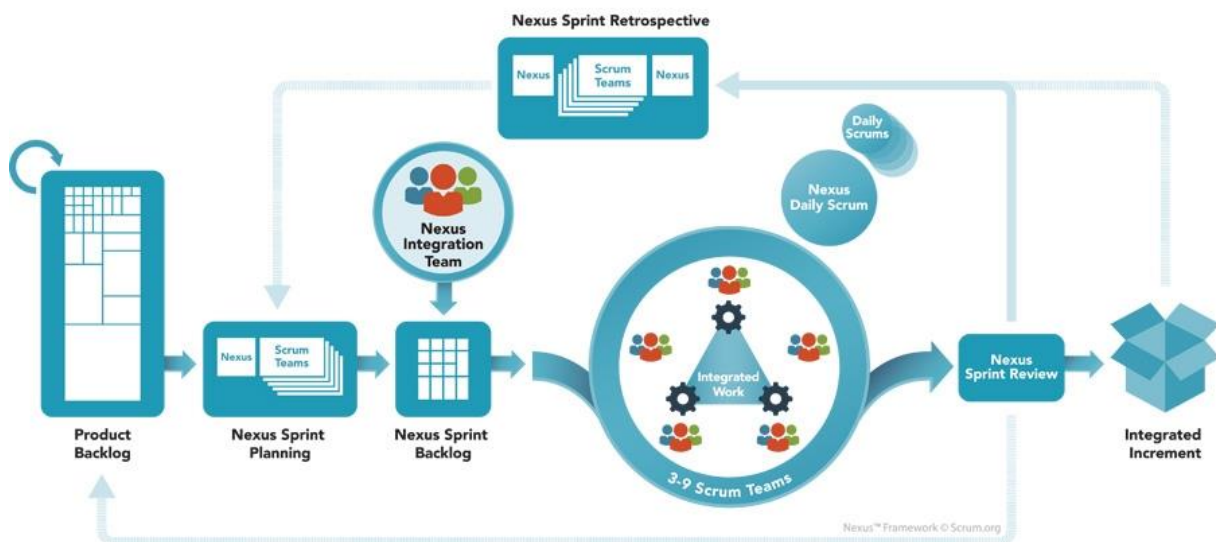
De modellen zijn complex, omvangrijk en de uitdaging voor organisaties is om het juiste model of gedeeltes van de modellen geleidelijk te implementeren.

## 1. Nexus

Het Nexus framework is het nieuwe wapenfeit van Ken Schwaber en een reactie op de verschillende agile frameworks die de afgelopen jaren zijn ontwikkeld. Het raamwerk, exoskeleton, van Nexus is bedoeld om met 3-9 Scrum Teams op basis van één Product Backlog te werken aan een geïntegreerd product.

De agile en Scrum principes vormen de basis van het framework waarbij de focus ligt op het verbeteren van de onderlinge samenwerking tussen de teams en de integratie van de te realiseren producten en oplossingen. Om dit te bewerkstelligen wordt naast de scrum teams een apart Scrum Team ingericht: het Nexus Integration Team.

### NEXUS™ FRAMEWORK



Het Nexus Integration Team bestaat dus uit een Product Owner, Scrum Master en leden afkomstig uit de overige Scrum Teams. De Product Owner is verantwoordelijk voor het vaststellen en prioriteren van de Backlog en de Scrum Master helpt de Scrum Teams de Nexus principes correct toe te passen. De teamleden van het Nexus team kunnen tevens onderdeel uitmaken van en participeren in andere Scrum Teams. Het team werkt volgens een vast ritme en is verantwoordelijk voor het afstemmen en mitigeren van de onderlinge afhankelijkheden tussen de teams en het integreren van de eindoplossing.

De sprintvergaderingen van het Nexus team zijn leidend en worden zoveel mogelijk geïntegreerd in de vergaderingen van de afzonderlijke Scrum Teams. Vertegenwoordigers vanuit de overige Scrum Teams worden uitgenodigd om deel te nemen aan het Nexus teamoverleg.

Het samenspel van het Nexus team en de overige Scrum Teams start met een Nexus Sprint Planning waarin het doel van de sprint wordt vastgesteld en wordt bepaald welke producten, oplossingen en wijzigingen worden opgepakt. De Backlog items vormen input voor de Sprint Planning van de

individuele teams. Daarnaast vindt er een dagelijks overleg, de Nexus Daily Scrum, plaats waarin de onderlinge afhankelijkheden en integratie issues worden geïdentificeerd die worden besproken in de Daily Scrums van de overige teams.

Aan het eind van elke sprint wordt een Nexus Sprint Review georganiseerd waarin het geïntegreerde product getoond wordt aan de omgeving en volgt een Nexus en Sprint Retrospective. In de Retrospective wordt de sprint geëvalueerd en worden verbeteracties vastgesteld.

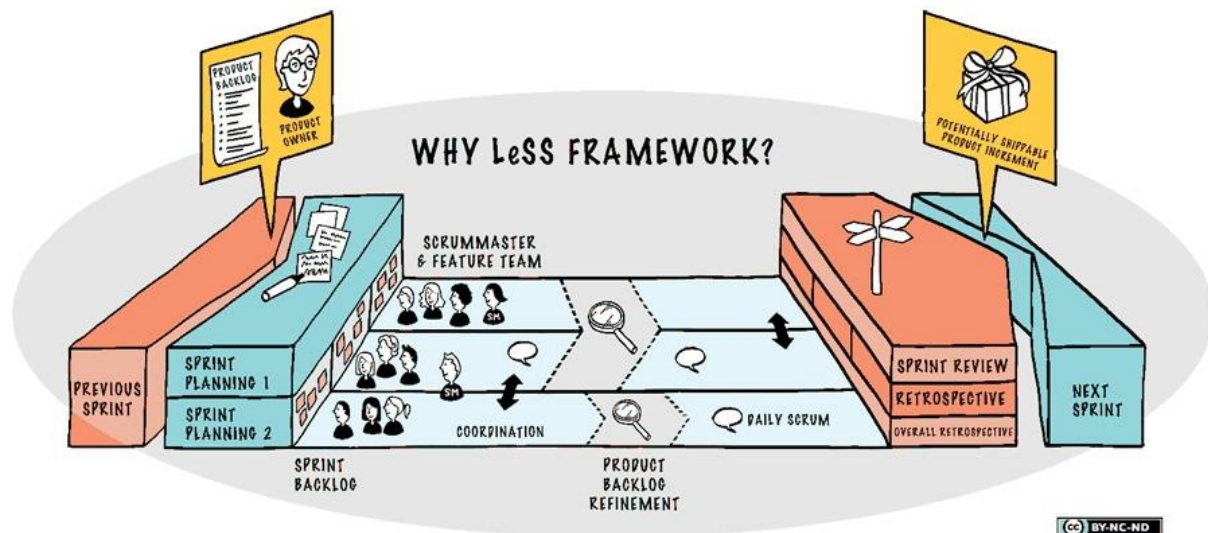
## **2. Large Scale Scrum (LeSS)**

Het LeSS framework is bedacht door Craig Larman en Bas Vodde en is bedoeld voor organisaties die met meerdere teams tegelijk willen scrummen. Inmiddels hebben onder andere BMW, Thales, Port of Rotterdam en Ericsson het framework geadopteerd.

Het framework heeft een sterke klant- en productfocus, omarmt lean- en systems thinking en doet nadrukkelijk afstand van tayloristische opvattingen, organogrammen en het denken in lokale optimalisatie. Onderstaand een overzicht van de belangrijkste principes.

1. Large-scale scrum is scrum - Multiple team Scrum versus multiple Scrum Teams
2. Customer Centric - Focus op klant en klantbehoefte vormt basis voor productontwikkeling
3. Whole product focus - Focus op product en toegevoegde waarde voor de klant
4. Lean Thinking - Creëer flow, vermijd rijen en verwijder de 7 verspillingen
5. System Thinking - Denk in systemen in plaats van oorzaak gevolg
6. Empirical proces control - Inspecteer, observeer en pas aan op basis van bevindingen
7. Queing theory - Reduceer en beperk rijen, work in progress, multi-tasking en variatie
8. Transparency - Creëer transparantie en geef inzicht in de voortgang en complexiteit van werkzaamheden
9. More with LeSS - Bouw een agile organisatie bottom-up en niet top-down met behulp van management
10. Continuous Improvement - Verbeter continu het proces van product delivery

LeSS is gebaseerd op multiple team scrum en gaat in tegenstelling tot het Scaled Agile Framework (SAFe) uit van één groot Scrum Team, bestaande uit meerdere teams die gezamenlijk producten realiseren op basis van één centrale Product Backlog. SAFe daarentegen gaat uit van een agile team dat bestaat uit meerdere afzonderlijke Scrum Teams met elk een eigen Product Owner en Product Backlog. Binnen het framework wordt een onderscheid gemaakt tussen LeSS (< 9 teams) en LeSS Huge (> 9 teams).



Het LeSS Scrum Team bestaat uit zelforganiserende cross-functional feature teams. Feature teams zijn multidisciplinaire teams die in staat zijn om ketenwijzigingen en component overstijgende producten te ontwikkelen. Het team beschikt over één productowner en meerdere Scrum Masters, waarbij de Scrum Masters verantwoordelijk zijn voor de LeSS adoption van de organisatie en de ondersteuning van de teams. In de praktijk vervult een Scrum Master vaak voor drie Scrum Teams tegelijk de rol van Scrum Master.

In het geval van LeSS Huge worden de teams vaak gekoppeld aan een area. Een area is een cluster van functionaliteiten of een gedeelte van een product dat sepeeraat waarde toevoegt voor de klant. De indeling in areas wordt gebruikt om focus aan te brengen in de productontwikkeling. Een area staat niet gelijk aan een SAFe valuestream, maar bewerkstelligt hetzelfde doel. Ter ondersteuning van de Product Owner wordt er vaak per area een Area Product Owner aangesteld die samen met de Product Owner een team vormt en de Product Owner ondersteunt bij het opstellen en prioriteren van de Backlog.

Overeenkomstig 'single' team Scrum start de realisatie met een overall Sprint Planning waarbij alle Scrum Teams aanwezig zijn en bepaald wordt wat er in de sprint wordt opgepakt. Vervolgens wordt in de teams afzonderlijk bepaald hoe de userstories worden opgepakt en gestart met de sprint.

Aan het einde van de sprint vindt er een gezamenlijke overall Sprint Review plaats, waarin de gerealiseerde producten en software worden getoond aan de betrokken stakeholders.

Tot slot wordt er een overall Sprint Retrospective en Sprint Retrospective op teamniveau georganiseerd, waarin de sprint gezamenlijk en afzonderlijk door de teams wordt geëvalueerd.

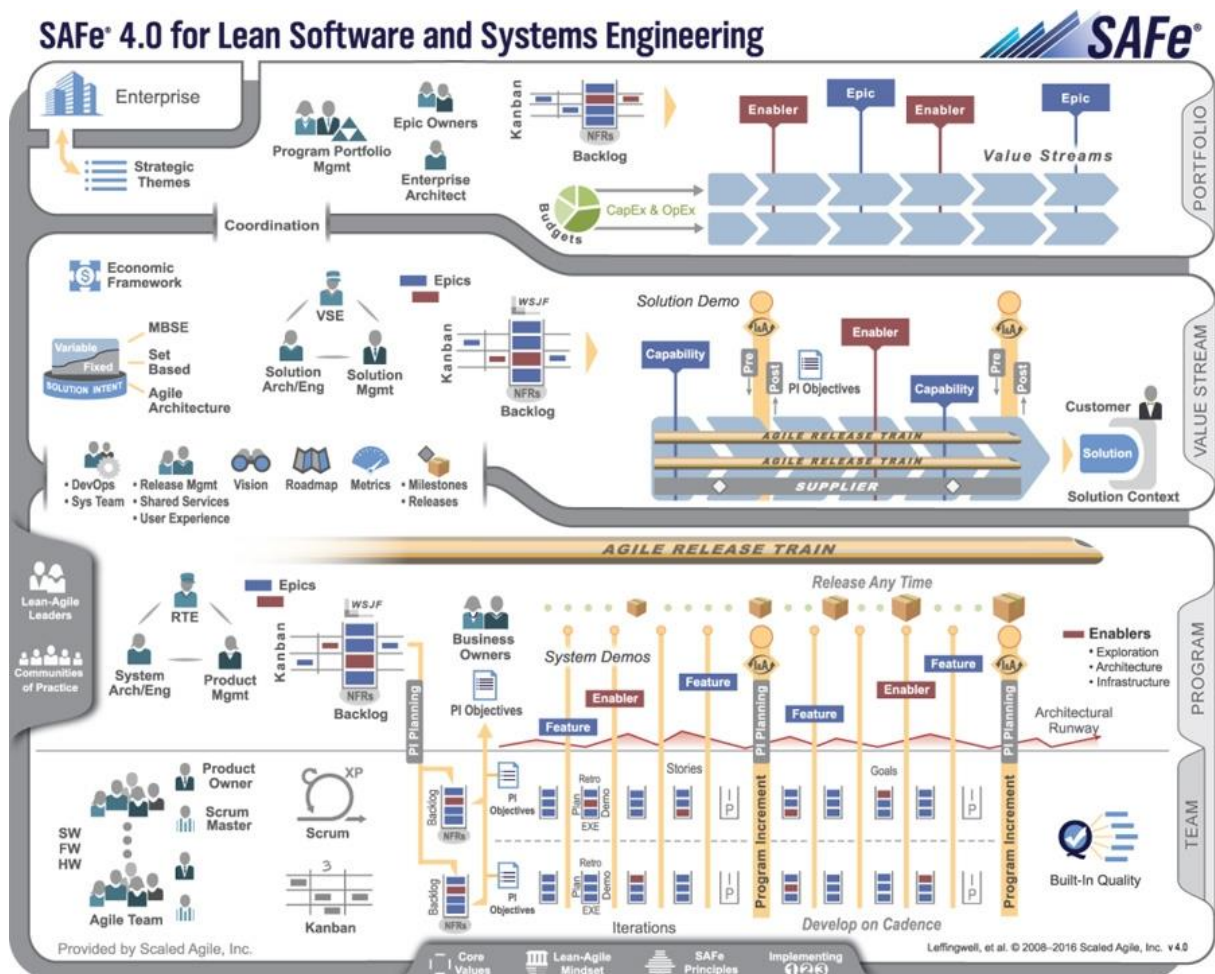
### 3. Scaled Agile Framework (SAFe)

Het Scaled Agile Framework (SAFe) is een bekend agile framework dat is bedacht door Dean Leffingwel. SAFe is een framework dat organisaties helpt om met meerdere Scrum Teams tegelijkertijd veranderingen door te voeren. Inmiddels hebben grote bedrijven zoals onder andere TomTom, Lego en Intell het SAFe framework geadopteerd en toegepast.

Het framework is gebaseerd op lean principes en probeert alignment te creëren tussen de strategie van een organisatie en de daadwerkelijke realisatie van producten en diensten door agile Scrum

Teams. SAFe maakt in tegenstelling tot LeSS hierbij gebruik van bestaande governance- en programmastructuren zoals het strategische portfolio overleg dat veel gebruikt wordt binnen organisaties om de prioriteit van veranderingen te bepalen.

1. Take an economic View - Change op basis van economische afwegingen (cost of delay)
2. Apply systems thinking - Optimaliseer ketens in plaats van (stand alone) componenten
3. Assume variability, preserve options - Ga uit van variëteit, wees flexibel en beschik over alternatieven
4. Build incrementally with fast, integrated learning cycles - Gebruik iteraties, test en evalueer periodiek (PDCA)
5. Base milestones on objective evaluation of working systems - Stel realistische doelen en vermijd tolgates
6. Visualize and limit WIP, reduce batch sizes, and manage que lengths - Creëer flow en monitor work in progress
7. Apply cadence, synchronize with cross domain planning - Hanteer een vast ritme, synchroniseer en plan delivery
8. Unlock the intrinsic motivation of knowledge workers - Motiveer en activeer intrinsieke motivatie
9. Decentralize decision making - Decentraliseer zoveel mogelijk de besluitvorming





SAFe onderkent respectievelijk vier niveaus: het portfolio-, value stream-, programma- en teamniveau. Op portfolio niveau worden door de directie de strategische thema's gedefinieerd die richting geven aan de gewenste verandering. Daarnaast worden op dit niveau de belangrijkste value streams (waardeketens) geïdentificeerd waarbinnen de veranderingen plaats dienen te vinden.

De strategische thema's worden gekoppeld aan de value streams en per waardeketen wordt een veranderbudget toegekend. Vervolgens worden de strategische thema's per value stream op programma niveau vertaald in hapklare brokken en wijzigingen, de epics, die vastgelegd worden op een programma backlog.

Tegelijkertijd wordt er per value stream een Release Train ingericht. Een Release Train is een agile team bestaande uit meerdere Scrum Teams die gezamenlijk verantwoordelijk zijn voor het ontwikkelen van werkende producten en software binnen een waardeketen. Elk team binnen de Release Train beschikt over een afzonderlijke Product Backlog, Product Owner en Scrum Master en werkt overeenkomstig de principes van agile Scrum.

De programma backlog wordt vervolgens verder opgeknipt en vertaald in een Product Backlog op basis waarvan de Scrum Teams binnen een Release Train kunnen starten met de realisatie. Om de onderlinge samenwerking en alignment tussen de teams te vergroten wordt er gemiddeld één keer in de 10 weken een Release Planning georganiseerd. Het overleg duurt vaak twee dagen. Bij dit overleg zijn het management en alle teams aanwezig, en worden gezamenlijk de doelstellingen en de inhoud van de eerst komende sprints bepaald. Daarnaast worden in de vergadering de belangrijkste afhankelijkheden en risico's tussen de teams afgestemd.

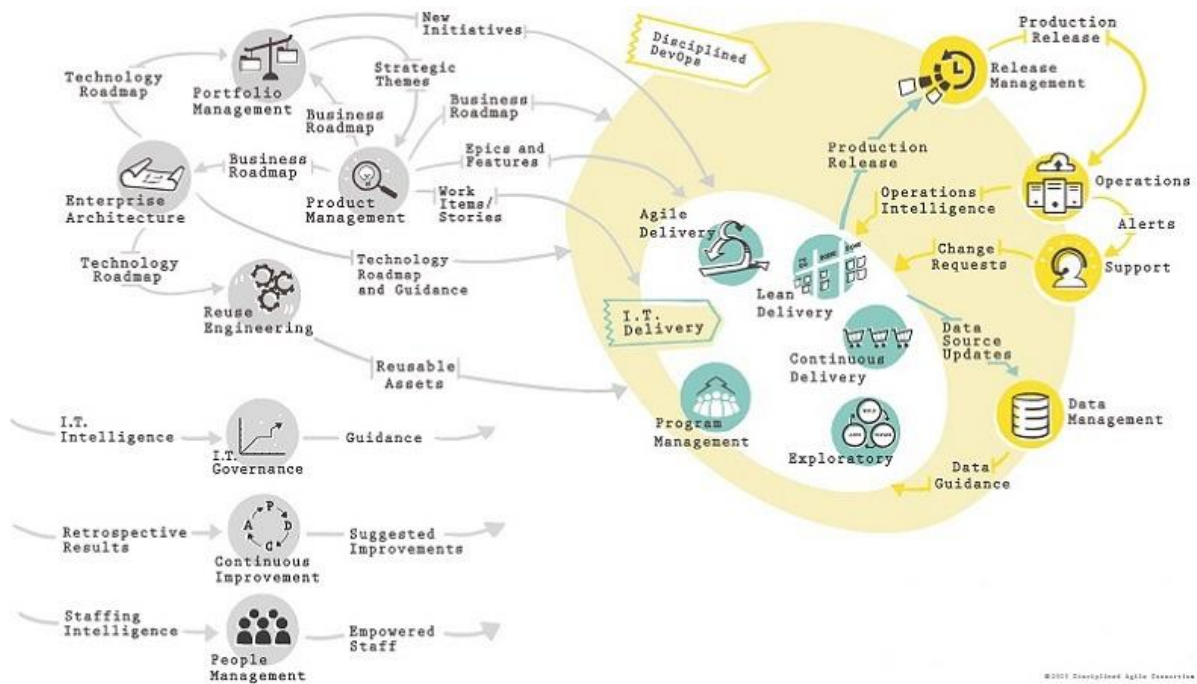
Tot slot benoemt het SAFe framework in tegenstelling tot LeSS verschillende ondersteunende rollen die de productontwikkeling en het voortbrengingsproces binnen het framework en de teams ondersteunen. Voorbeelden van rollen zijn de Agile Release Train Engineer, het Release Management en Enterprise Architecten. Rollen die sterke overeenkomst vertonen met leidinggevende functies in niet agile organisaties.

#### **4. Disciplined Agile Delivery (DaD)**

De ontwikkeling van het Disciplined Agile Delivery framework is in 2009 gestart door IBM onder leiding van Scott Ambler en Mark Lines. Het framework is bedoeld voor organisaties die op grote schaal lean-agile willen werken. DaD onderscheidt zich van LeSS en SAFe door de procesgerichte benadering van delivery, de focus op een werkbare oplossing, een significant andere teamsamenstelling en een sterke focus op de omgeving.

Het framework is een lean-agile hybride model en is gebaseerd op het Agile Manifesto dat de bedenkers hebben aangescherpt en vertaald in het Disciplined Agile Manifesto bestaande uit 15 simpele uitgangspunten waarin de focus ligt op het realiseren van werkbare oplossingen voor stakeholders in plaats van producten voor de klant.

In het framework zijn duidelijk invloeden van Scrum, Extreme Programming, XP, Kanban, ITIL, SAFe en Agile Modeling terug te vinden. Maar bovenal is DaD een procesgericht framework met een duidelijke focus op de interactie tussen personen en de brede omgeving van een organisatie.



Producten, oplossingen en software worden op basis van heldere doelstellingen overeenkomstig een vast delivery proces gerealiseerd. De basis Delivery Lifecycle van DaD bestaat uit 5 fasen: concept, inception, construction, transitie en productiefase. In de concept- en inceptionfase worden de visie, roadmap en doelarchitectuur opgesteld die richting geven aan de realisatie van de oplossing. Vervolgens wordt in de constructionfase gestart met de realisatie van de oplossing die in de transitiefase wordt gedeployed en in de productiefase in gebruik wordt genomen door de stakeholders voor wie de oplossing wordt gerealiseerd.

Gedurende dit proces is het DaD team, evenals een traditioneel Scrum Team, gezamenlijk verantwoordelijk voor de realisatie van de oplossing. De samenstelling van het team verschilt echter aanzienlijk van de samenstelling van traditionele Scrum Teams. Binnen het team wordt namelijk een onderscheid gemaakt tussen primaire en secundaire rollen. Een DaD team bestaat uit een Team Lead, Product Owner, Architectuur Owner, team members en vaak maken stakeholders ook onderdeel uit van het team. Daarnaast wordt het team aangevuld met secundaire rollen, waaronder proces-, systeem en technische experts, testers en integrators. Allen zijn volgens het framework noodzakelijk om een werkbare oplossing te realiseren en de interactie tussen het team en de omgeving succesvol te laten verlopen.

Bij de start van de realisatie wordt door het team overeenkomstig de Scrum principes een Iteration Planning georganiseerd, waarin bepaald wordt met welke oplossingen gestart zal worden. In de dagelijkse Coordination Planning wordt de voortgang besproken. Aan het eind van de sprint wordt in de Iteration Review de oplossing getoond aan de omgeving en wordt er afgesloten met een Retrospective waarin de sprint wordt geëvalueerd.

## De agile organisatie

De maatschappij, klantbehoeftes en verwachtingen veranderen voortdurend. Kwaliteit- en veiligheidseisen worden strenger, wet- en regelgeving worden geharmoniseerd, en technologische

ontwikkelingen volgen elkaar in rap tempo op. Tegelijkertijd worden de kwantiteit, snelheid en complexiteit van veranderingen steeds groter.

Om tijdig in te spelen op deze veranderingen zijn in de afgelopen jaren verschillende agile delivery modellen ontwikkeld, waaronder Scrum, Nexus, Large Scale Scrum (LeSS), Scaled Agile Framework (SAFe) en Disciplined Agile Delivery (DaD). De modellen en methoden zijn gebaseerd op het Agile Manifesto en hebben allen als doel om gericht, iteratief, incrementeel werkende producten, oplossingen en software te realiseren die voldoen aan de wensen van de klant.

Daarnaast proberen de modellen organisaties handvatten te geven om de transitie te maken naar een volwassen, effectieve agile organisatie die in staat is om tijdig in te spelen op veranderende marktomstandigheden en efficiënt en effectief veranderingen kan doorvoeren.

Welk model het beste past bij een organisatie is afhankelijk van het type, de governance en het volwassenheidsniveau van de organisatie, maar de praktijk heeft inmiddels voldoende aangetoond dat de agile principes en frameworks werken.