

Continue testverbetering

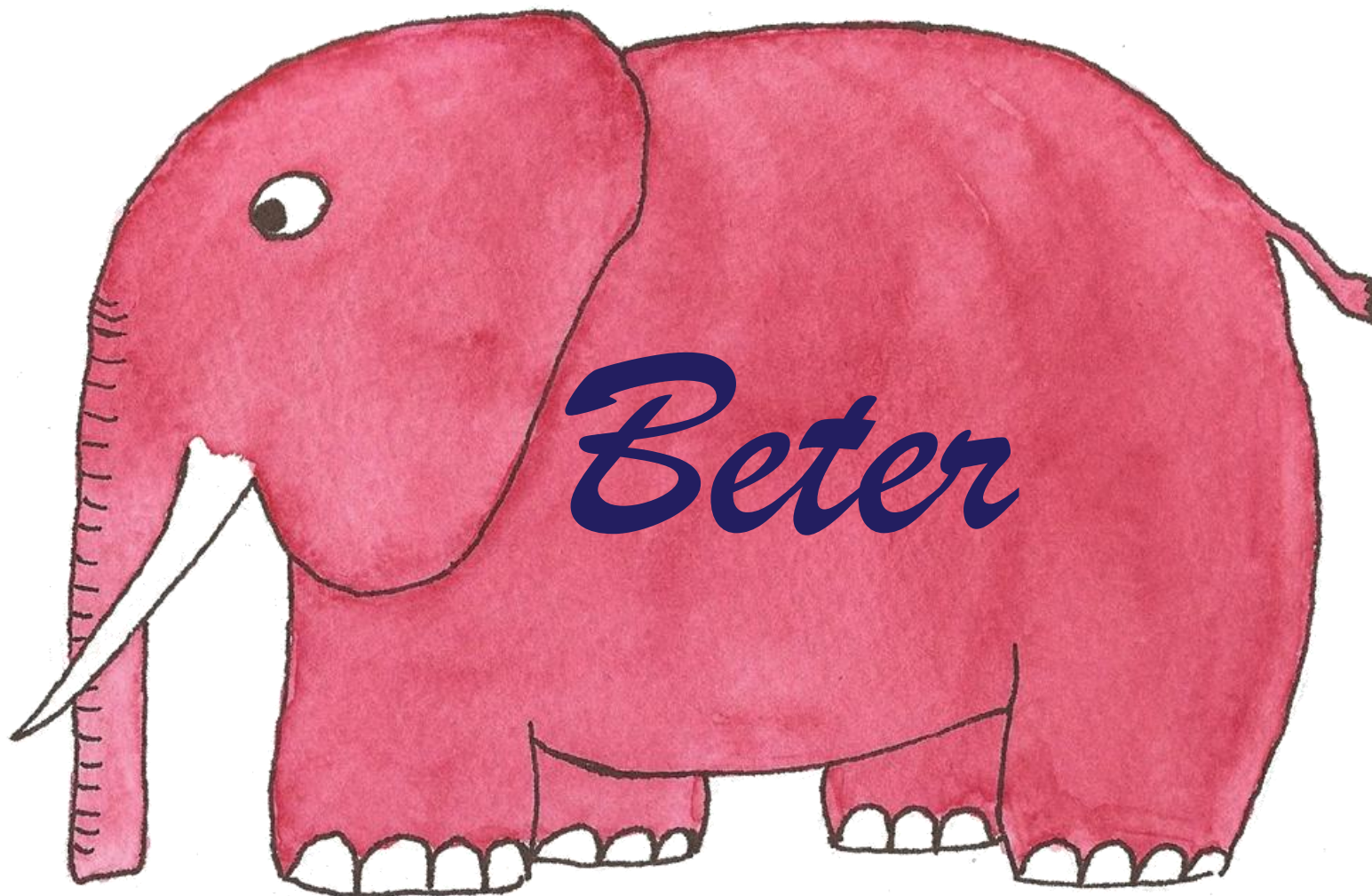
in elke situatie



Testverbetering

- Het probleem
- De ingrediënten
- De oplossing

Het probleem



Het probleem (toegelicht)

- Verbeteren van het testen lijkt op een roze olifant
 - Het ziet er aantrekkelijk en veelbelovend uit
 - Maar als je het wilt 'opeten' blijkt het veel te groot
 - Je weet niet waar je moet beginnen en waar te eindigen

Symptomen

- Maximale score
 - ... en toch problemen
- Prima prestaties
 - ... en toch een slechte score
- Goede verbeterplannen
 - ... en toch niets van terecht gekomen
 - ... en toch andere zaken verbeterd
- Duidelijk testbeleid
 - ... en toch grote verschillen in status en naleving
- Algemeen geaccepteerd
 - ... en toch een ivoren toren

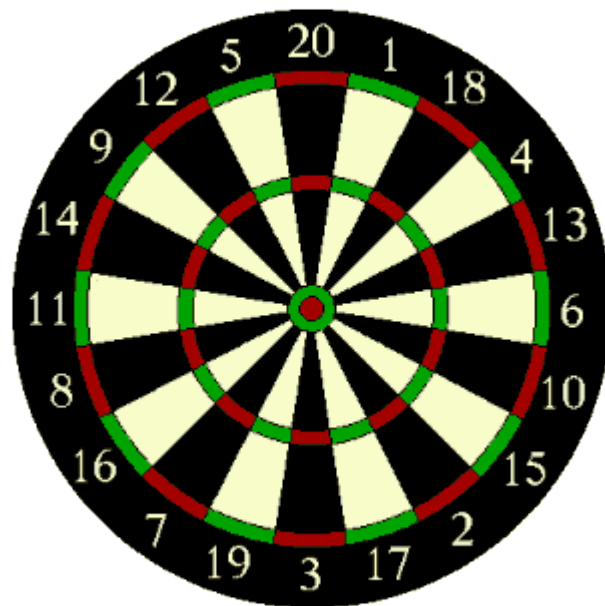


Symptomen (toegelicht)

- Er zijn goede methoden voor testverbetering, bijvoorbeeld TPI Next en TMMi
 - Daar hebben allerlei organisaties veel baat bij gehad
 - Het testen is ermee flink verbeterd
 - Toch begint het te knellen
 - Je merkt aan allerlei kleine strubbelingen dat - na aanvankelijke vorderingen - het verbeterproces steeds stroever verloopt

Eén doel, ...

Methode

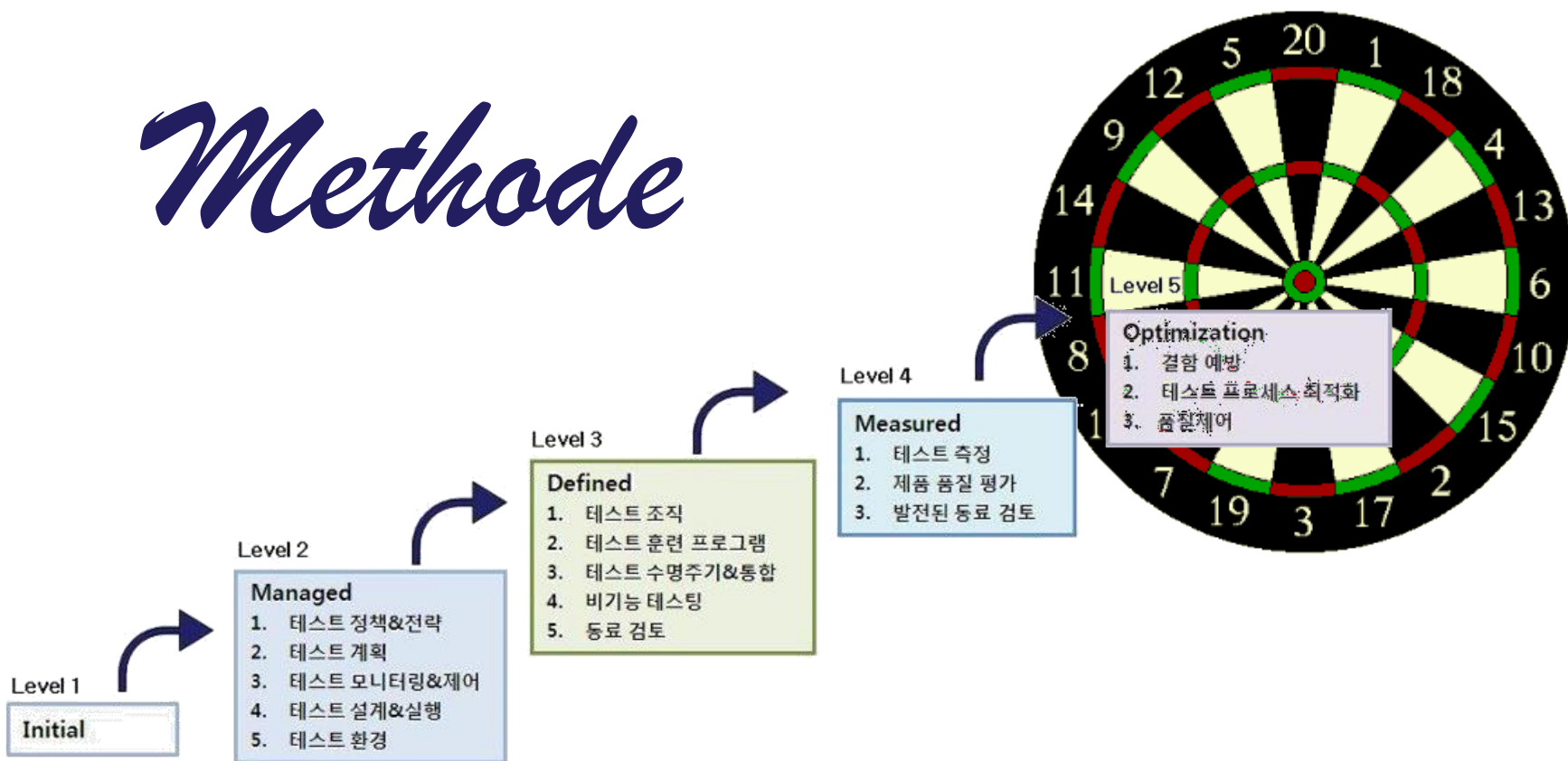


Eén doel, ... (toegelicht)

- De traditionele ('process reference') methoden denken vanuit één universeel einddoel voor het verbeterproces
 - TPI Next: alle aandachtsgebieden op weg naar volwassenheidsniveau 'optimizing'
 - TMMi: de hele testorganisatie op weg naar level 5 'optimization'

... één weg,

Methode

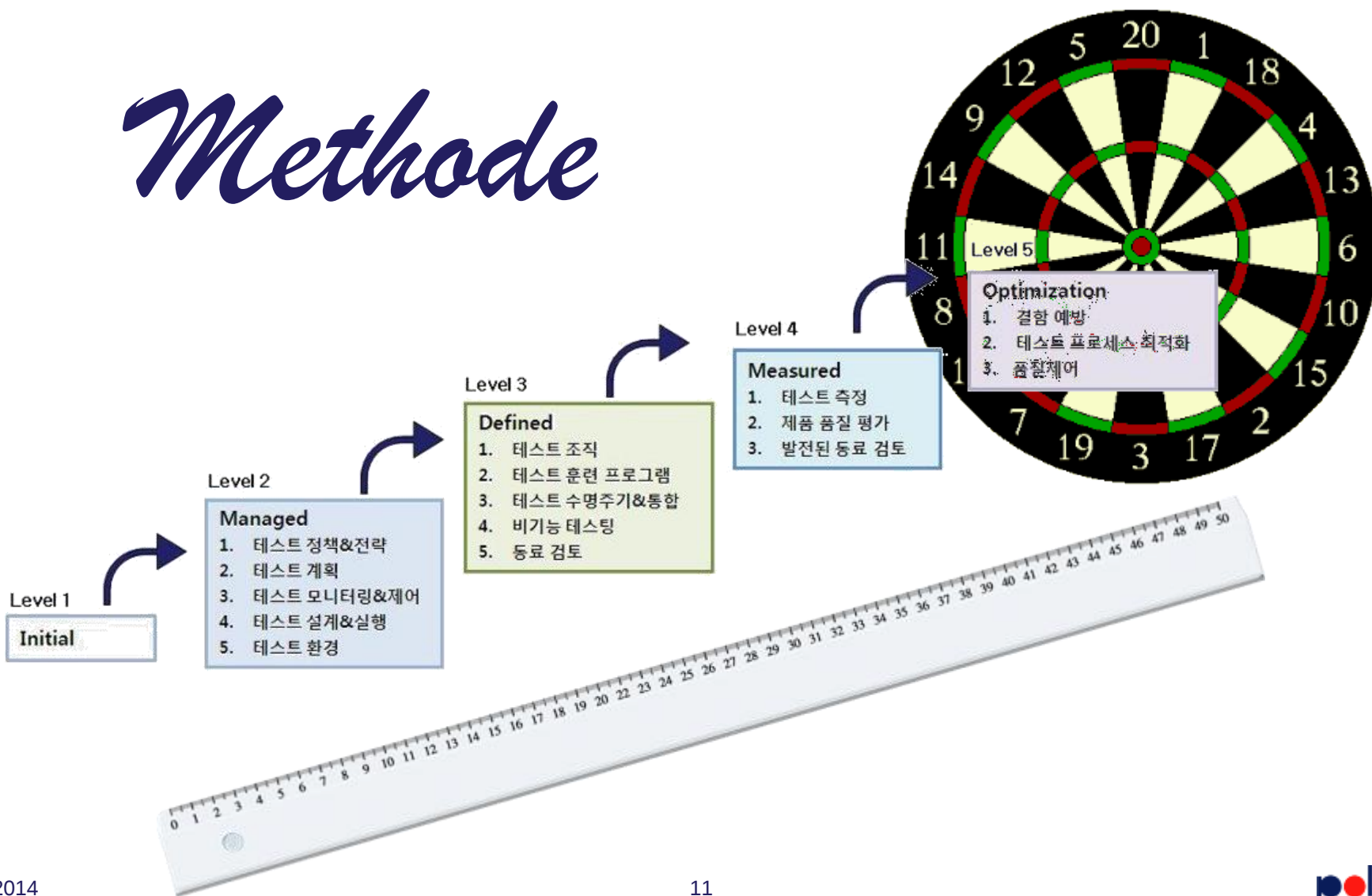


... één weg, (toegelicht)

- De weg naar dat universele einddoel is vrij strikt in stappen beschreven

... één maat,

Methode



... één maat, (toegelicht)

- De voortgang op de weg naar het einddoel wordt gemeten aan de hand van de afgelegde stappen

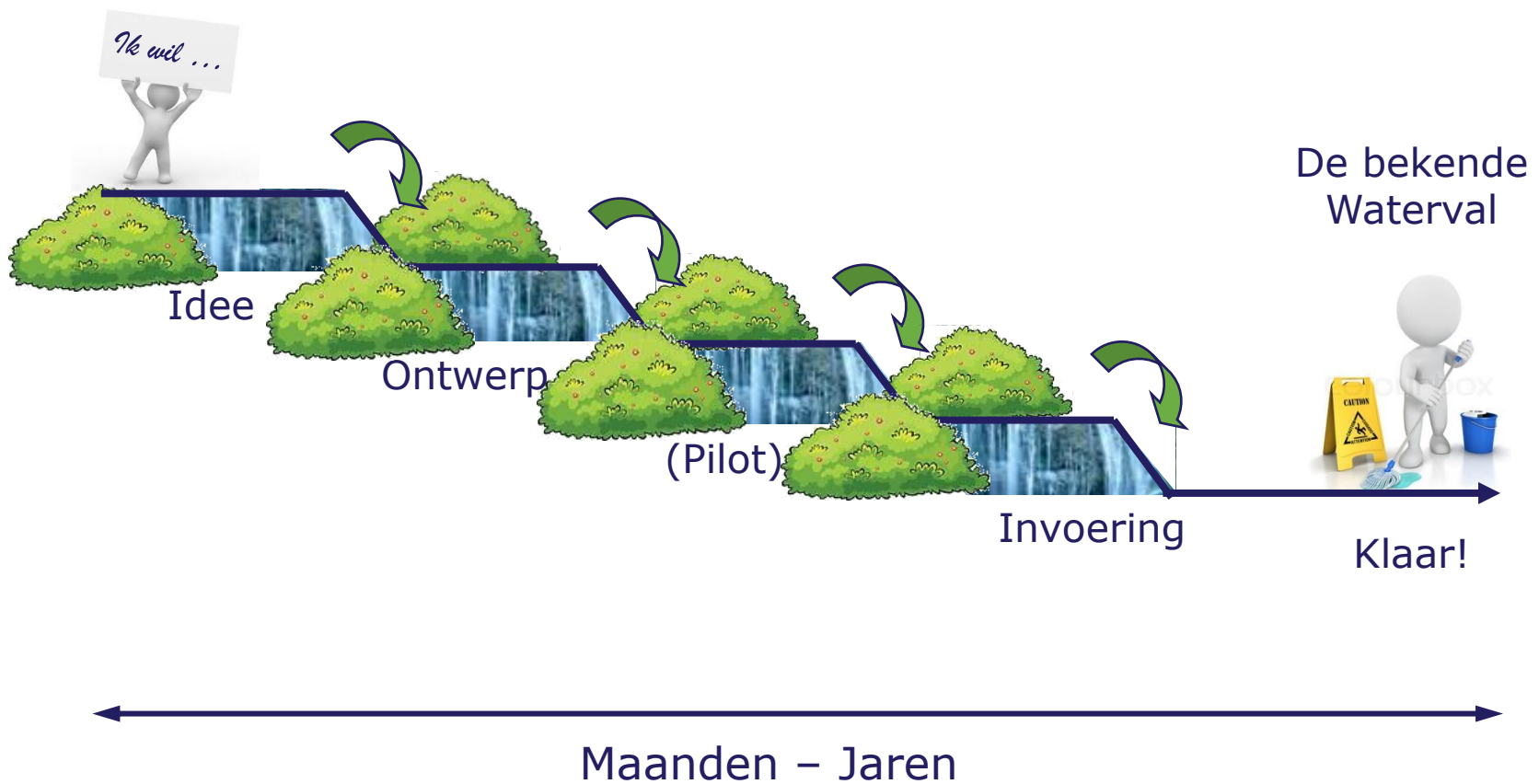
... één scope,



... één scope, (toegelicht)

- Veelal wordt het verbeterproces universeel uitgerold over de hele (test)organisatie
- Het 'landschap' waarbinnen wordt verbeterd lijkt uniform, overal even 'plat'
 - Alsof er alleen gelijksoortige, omvangrijke systemen op één bepaalde standaard infrastructuur volgens één bepaalde standaard aanpak worden ontwikkeld
- Iedereen wordt geacht zich in gelijke mate aan de verbeteringen te conformeren
 - Zelf als het niet helemaal goed past

... één aanpak,



... één aanpak, (toegelicht)

- Dat leidt tot verbeterprojecten met een vooraf geplande, vaste fasering
- ... en een geplande doorlooptijd van vele maanden
 - (en vaak ongeplande uitloop tot in jaren...)

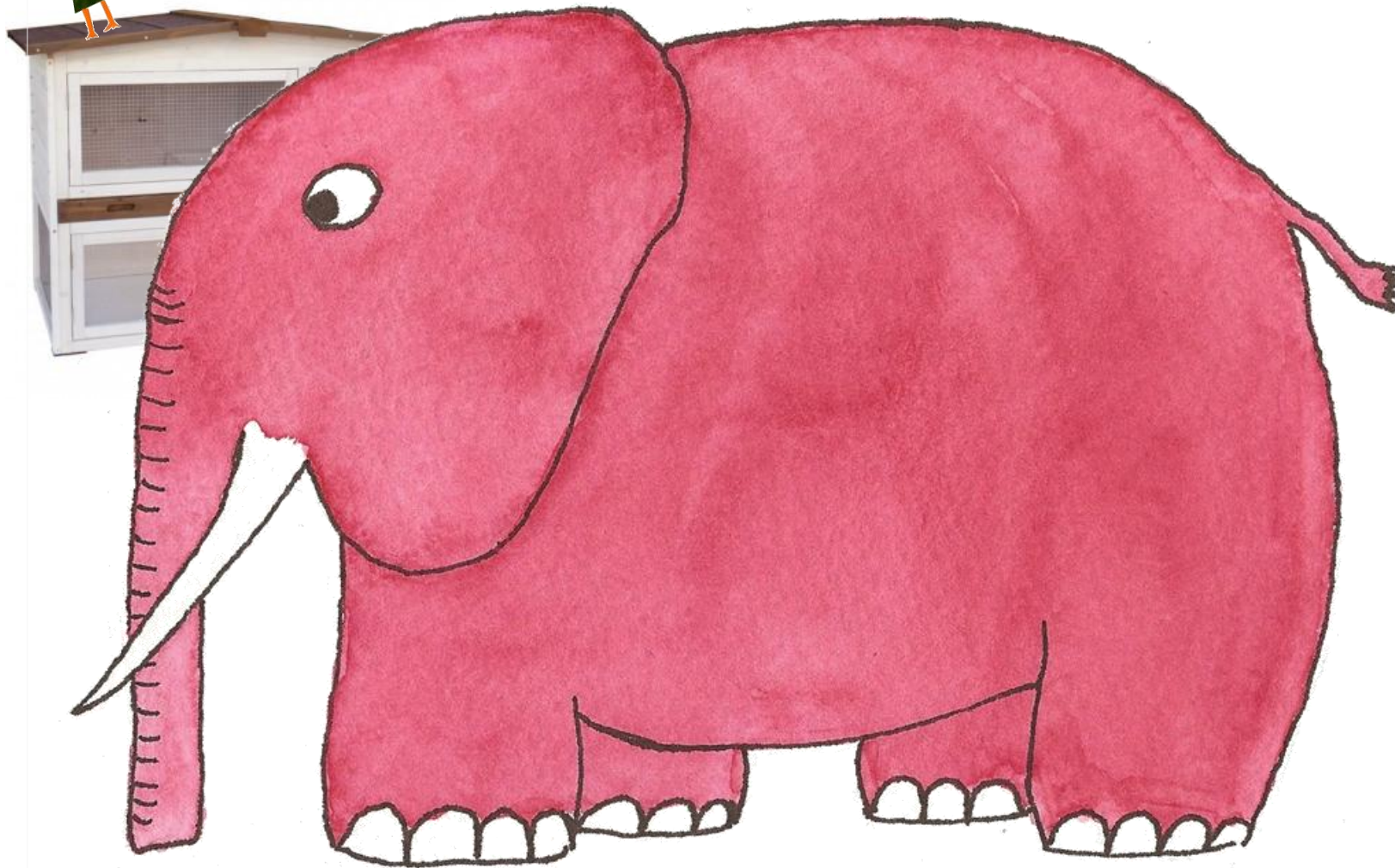
... 'bound'



... 'bound' (toegelicht)

- Dergelijke verbeteraanpakken leggen je vast in structuren, die weinig ruimte bieden om in te spelen op karakteristieke kenmerken van de organisatie en op tussentijdse veranderingen

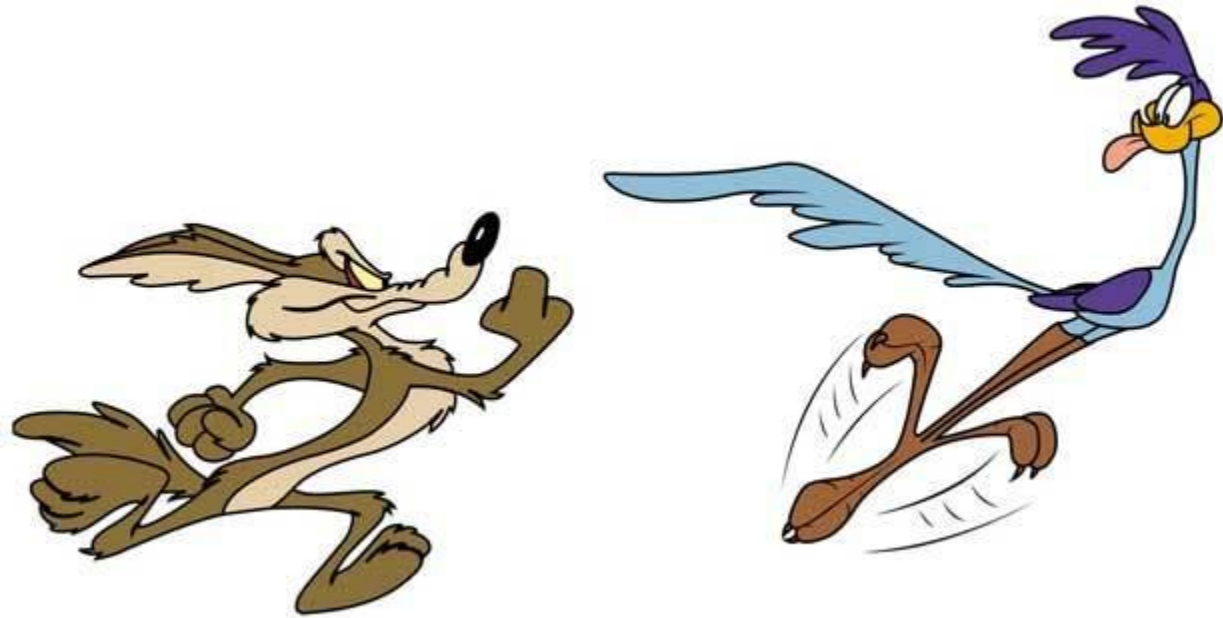
Het past niet meer!



Het past niet meer! (toegelicht)

- Waar de roze olifant vroeger een prima aanpak was, zie je dat het nu niet meer past
 - Voorheen grote IT-/testorganisaties die één centraal IT-landschap ontwikkelden en beheerden
- Een dergelijke aanpak is tegenwoordig te groot, te log, te star
 - ... omdat het landschap is versnipperd
 - Pakketten
 - Websites
 - Outsourcing / regievoering
 - Apps
 - Cloud
 - Solomo (So^{cial} Lo^{cal} Mo^{bile})
 - ...

Het moet ...



Het moet ... (toegelicht)

- Sneller
 - Releasecycli in weken in plaats van maanden
 - Daily builds
- Dan kun je niet aankomen met een verbeteringstraject van een (half-) jaar

en ...



en ... (toegelicht)

- Flexibeler
 - De markt / omgeving is continu in beweging
 - De business wil dat op de voet volgen
- Dan kun je je niet vastleggen op resultaten voor over een half jaar

en ...



en ... (toegelicht)

- Veelkleuriger
 - Eén IT-landschap bestaat niet meer
 - Een mainframe systeem is iets heel anders dan een app
- Dus kun je niet meer volstaat met één uniforme verbeteraanpak
 - Je zult moeten differentiëren
 - Bepalen waar welke knelpunten liggen en die individueel oplossen

en ...



en ... (toegelicht)

- Lean
 - ‘De crisis’ gaat nooit meer over
 - De bomen groeien niet meer tot in de hemel
- Een oversized aanpak wordt niet geaccepteerd
 - Just in time, right-sized
 - “If it ain’t broke, don’t fix it”
 - Geen kostbare bijwerkingen

en ...



en ... (toegelicht)

- Rendabeler
 - De business case
- Verbeteracties moeten meer opleveren dan ze kosten
 - In harde euro's
 - Onderaan de streep
 - “Wat verdienen/besparen we met deze verbetering?”

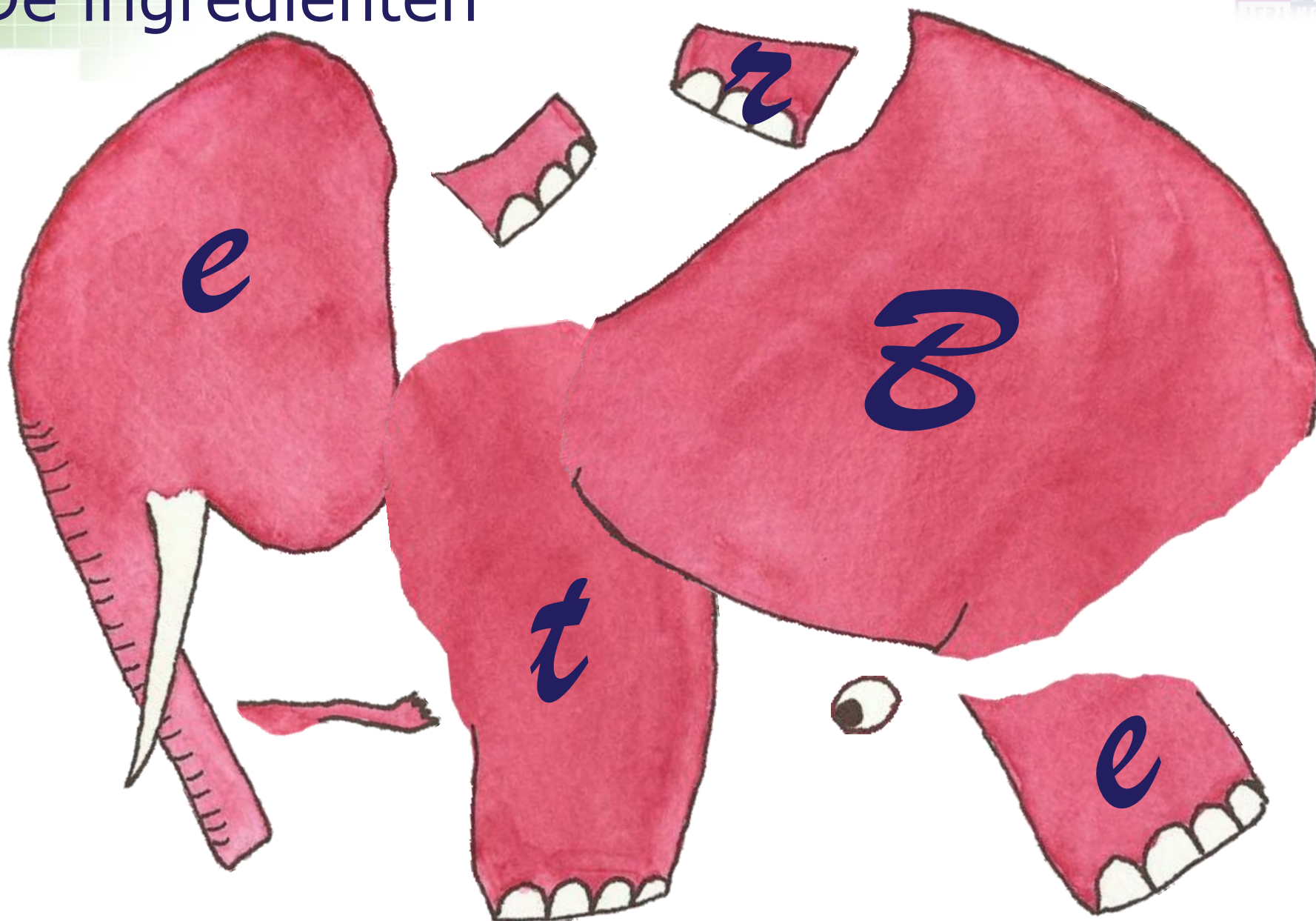
en ...



en ... (toegelicht)

- Vrijer
 - Geen keurslijf
- ... maar de vrijheid om in te spelen op
 - De business
 - De context
 - De infrastructuur
 - De tussentijdse veranderingen
 - ...

De ingrediënten



De ingrediënten (toegelicht)

- Wat kun je allemaal gebruiken om een testverbeteraanpak te verbeteren?

Beter verbeteren

- ... door gebruik te maken van bestaande verbeteringen
 - Agile
 - Scrum
 - Kanban
 - Exploratory
 - Test driven
 - Context driven
 - Kaizen
 - Continuous delivery
 - ...

Beter verbeteren (toegelicht)

- In onze test- en ontwikkelwereld zijn de afgelopen jaren allerlei verbeteringen doorgevoerd
- Die kun je analoog doortrekken naar een verbeteraanpak

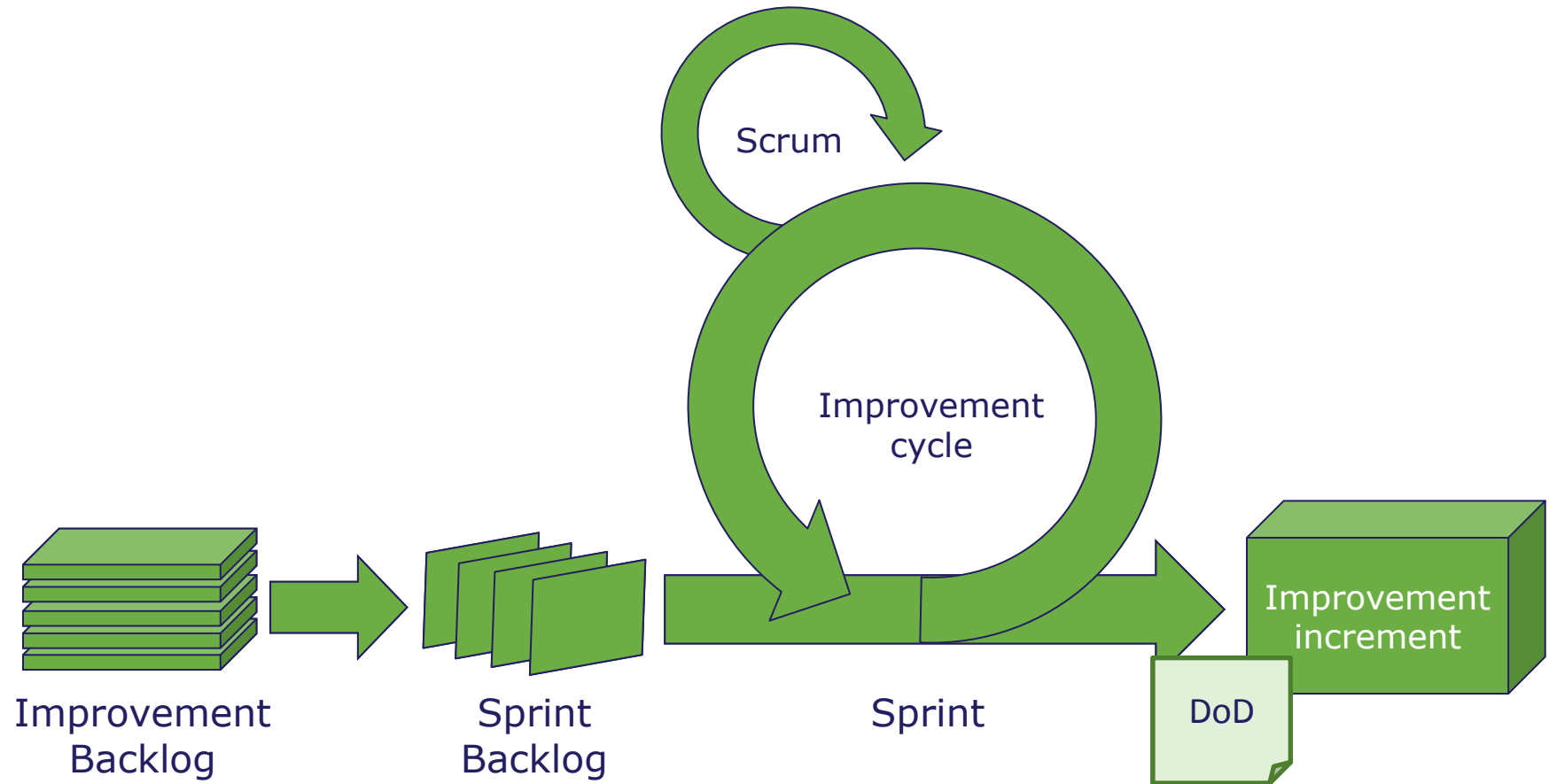
Agile



Agile (toegelicht)

- Uit het Agile gedachtegoed kun je veel bruikbare ideeën overnemen
 - Individuals and interactions / zelfsturende teams
 - Customer collaboration / owner uit de business
 - Working software
 - In dit geval geïmplementeerde verbeteringen
 - Welcome changes
 - Simplicity (“Maximizing work not done”)
 - ...

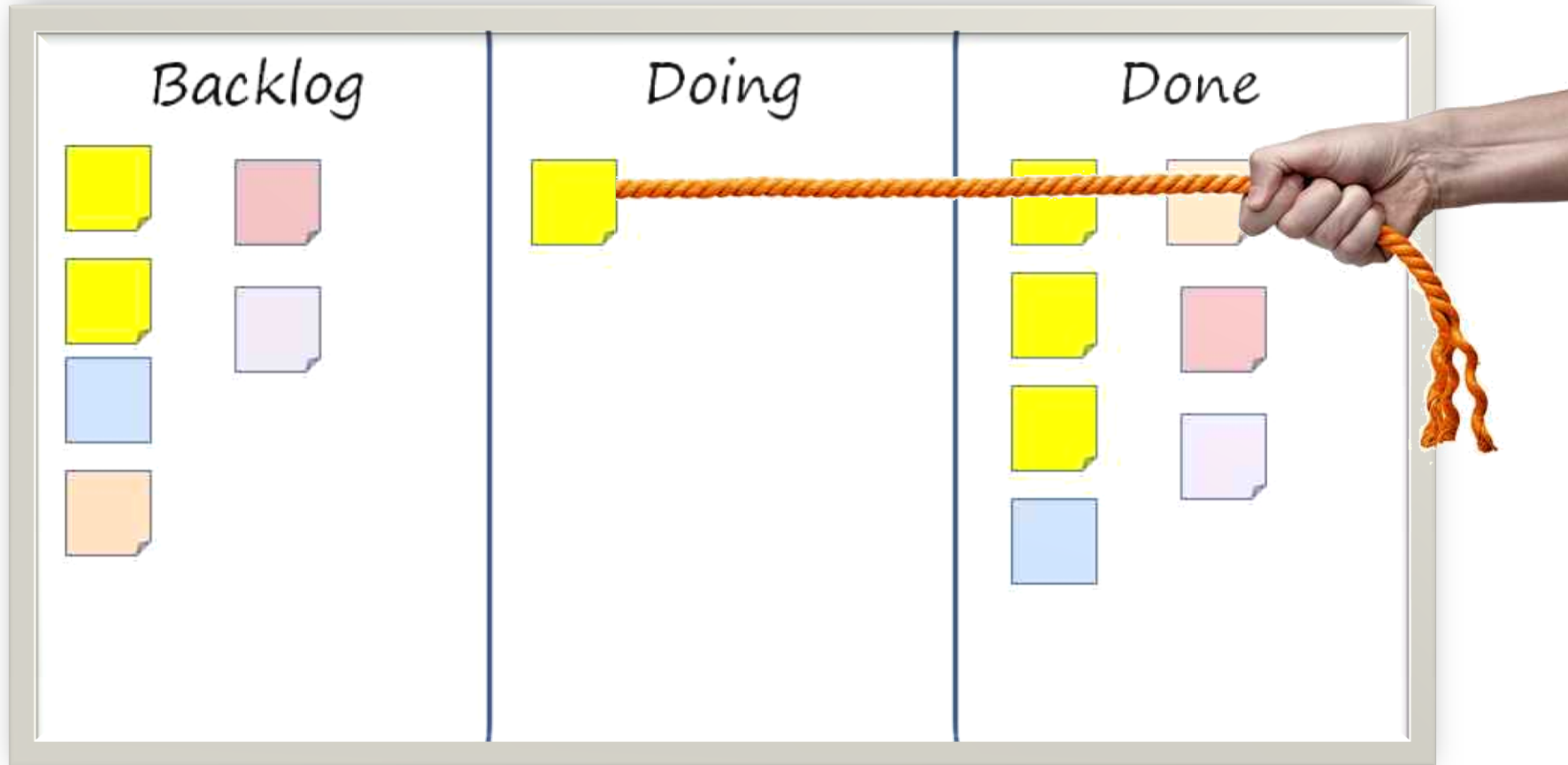
Scrum



Scrum (toegelicht)

- Uit de Scrum aanpak kun je veel bruikbare ideeën overnemen
 - Kort-cyclisch
 - Backlogs met user stories
 - As [role] I want [result] so that [goal]
 - Gesized voor één iteratie
 - Met business value en prioriteit
 - ‘Definition of done’
 - Met acceptatie criteria
 - ‘Shipable’ resultaten
 - ...

Kanban



Kanban (toegelicht)

- Kanban leert je te denken vanuit een pull- in plaats van een push-mechanisme
 - Stem werk in uitvoering af op de capaciteit van de organisatie (niet te veel hooi op de vork)
 - Pak een nieuwe verbetering pas op als de vorige 'klaar' is
 - Vaak bouwen verbeteringen voort op eerdere of andere verbeteringen
 - Beperk 'tussenproducten': verbeteringen die geen direct voordeel opleveren
 - ...

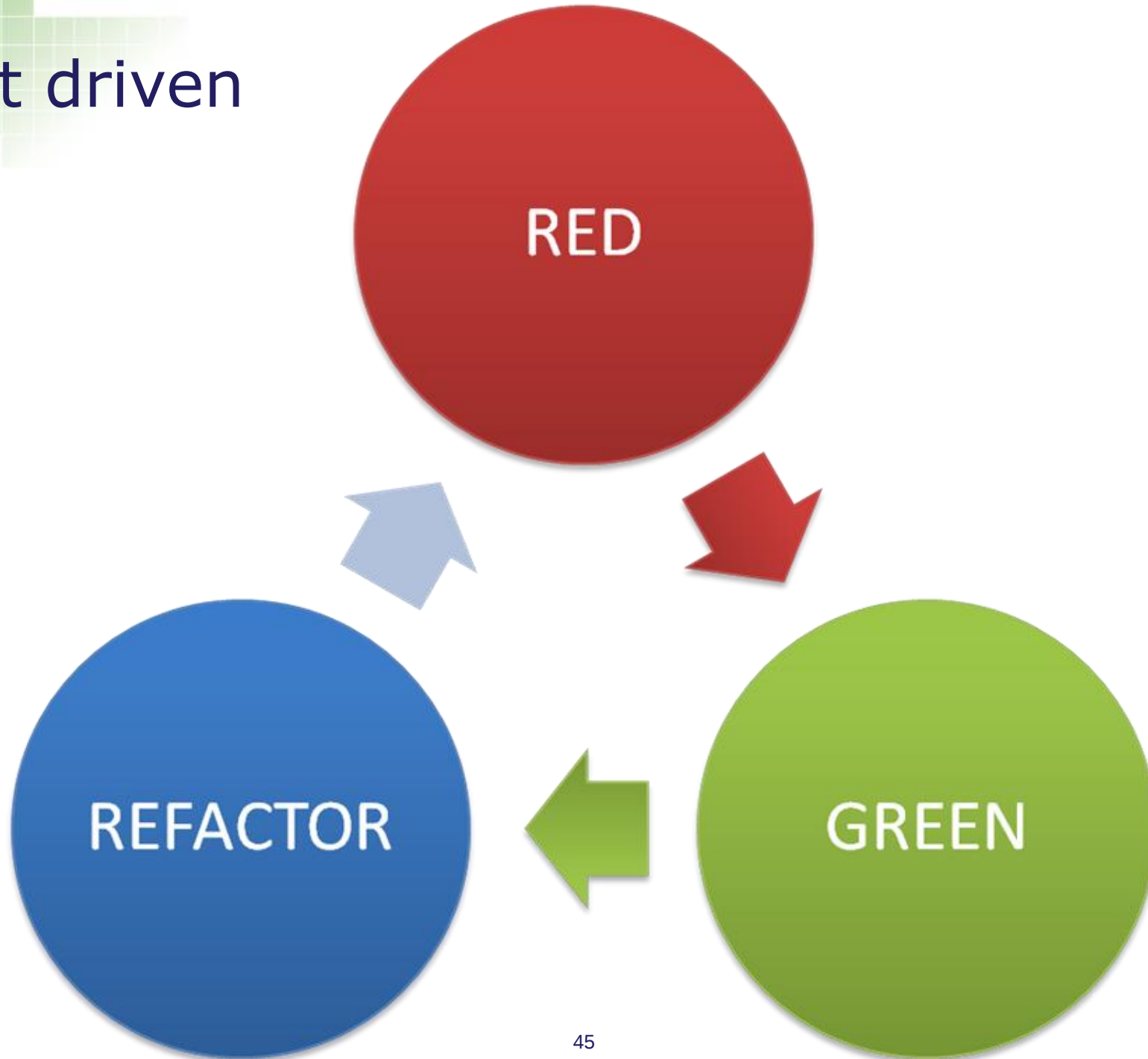
Exploratory



Exploratory (toegelicht)

- Denk in time-boxen, charters
 - Ieder charter een eigen thema
 - Leer tijdens de uitvoering
 - Volgende stap bepalen op basis van wat je in de huidige stap hebt geleerd
 - ...

Test driven



Test driven (toegelicht)

- Bedenk eerst hoe je de resultaten van een verbetering zichtbaar kunt maken, in termen die betekenis hebben voor de business
 - Euro's
 - Time-to-market
 - Concurrentiepositie
 - ...
- Dan pas de individuele verbetering bedenken en implementeren
- Af en toe 'refactoring' om de 'technical debt' weg te werken

Context driven



- The Seven Basic Principles of the Context-Driven School**
1. The value of any practice depends on its context.
 2. There are good practices in context, but there are no best practices.
 3. People, working together, are the most important part of any project's context.
 4. Projects unfold over time in ways that are often not predictable.
 5. The product is a solution. If the problem isn't solved, the product doesn't work.
 6. Good software testing is a challenging intellectual process. Only through judgment and skill, exercised cooperatively throughout the entire project, are we able to do the right things at the right times to effectively test our products.
 - 7.

Context driven (toegelicht)

- There are good practices in context, but there are no best practices
- The *improvement* is a solution. If the problem isn't solved, the *improvement* doesn't work
 - Kijk of het werkt in het 'hier en nu'
- Good software testing *improvement* is a challenging intellectual process
 - Zonder domeinkennis gaat het niet lukken

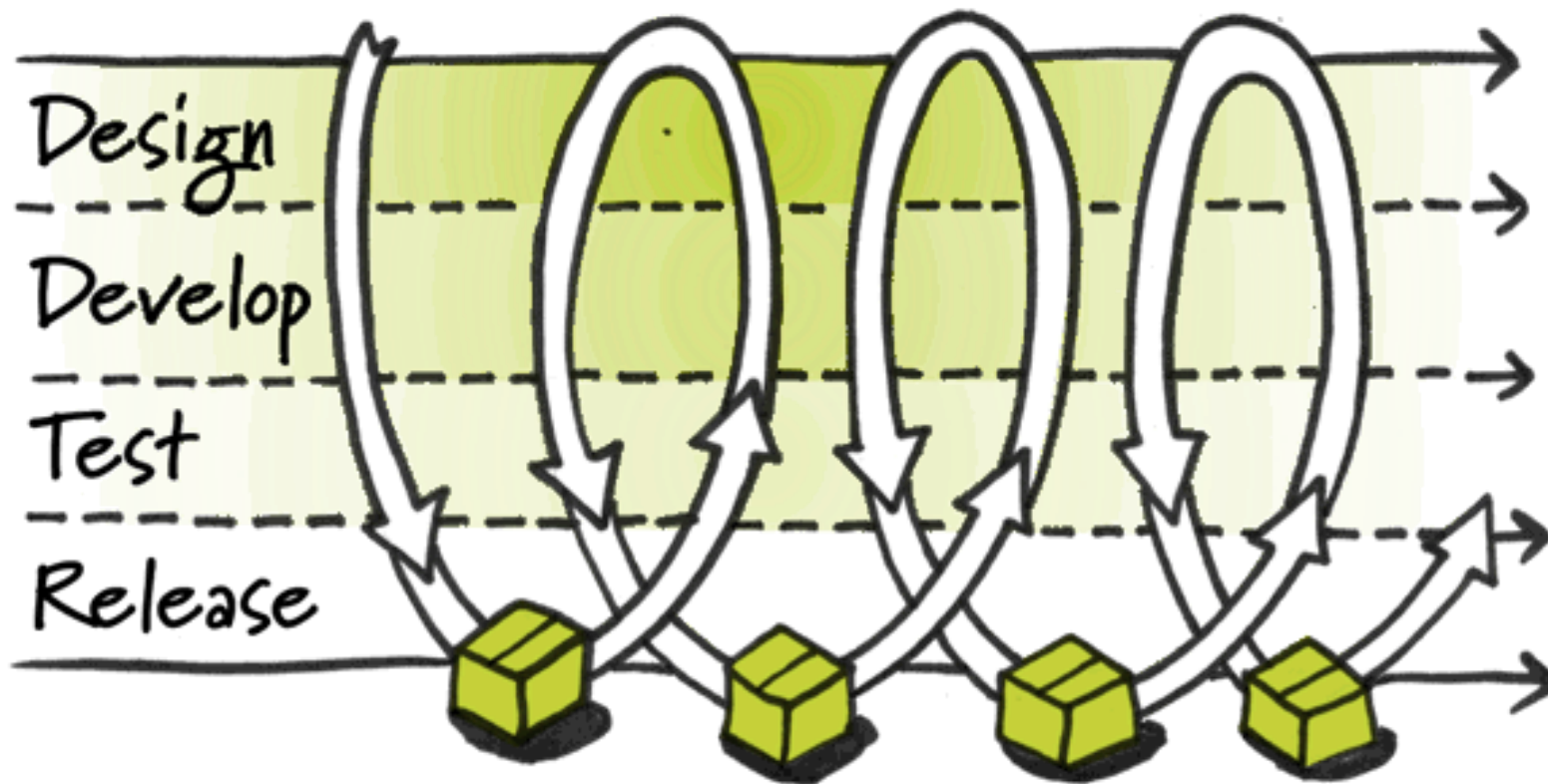
Kaizen



Kaizen (toegelicht)

- De beste verbeterideeën komen vanaf de werkvloer
- Betrek iedereen bij de verbeteringen, niet alleen de consultants en de managers
- Zorg voor een permanente, transparante informatie-stroom naar alle betrokkenen

Continuous delivery



Continuous delivery (toegelicht)

- Als het even kan...
 - Geen big bang
 - Elke uitgewerkte verbetering direct implementeren
 - ... en integreren in de 'normale' werkstroom

De oplossing

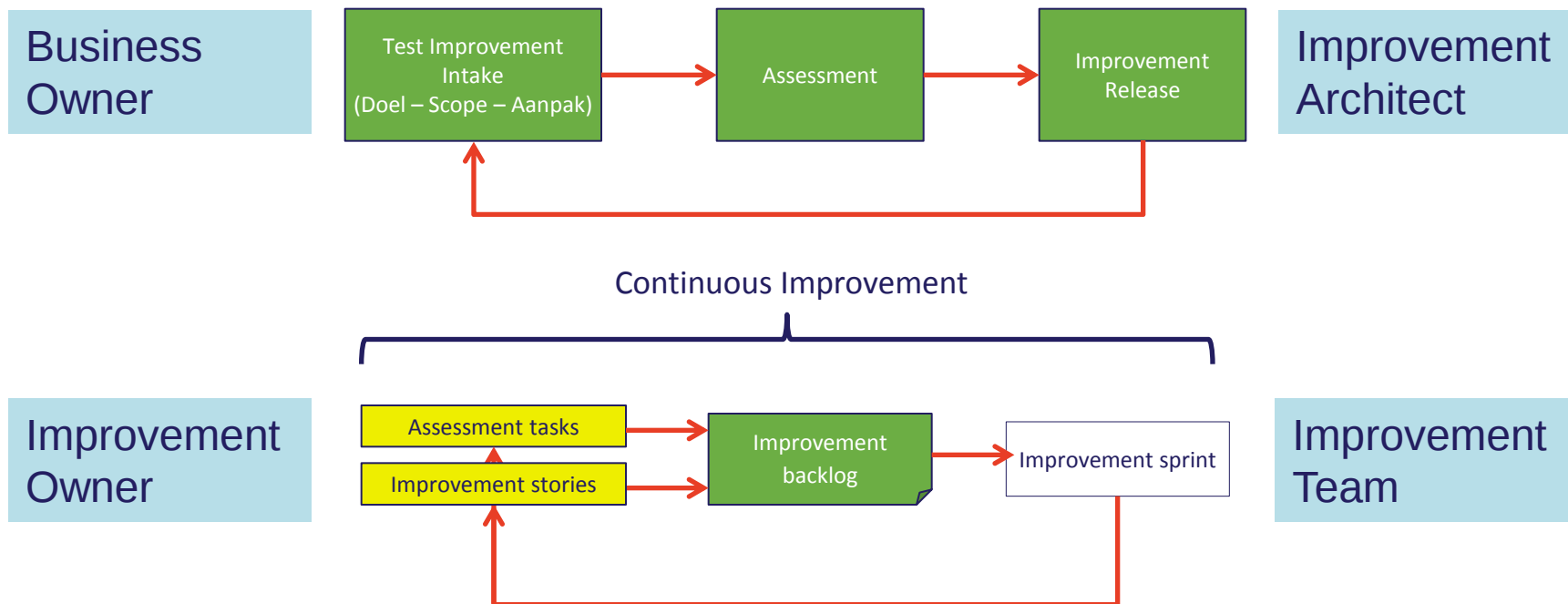
- The best of all worlds
 - Revolutie en Evolutie
 - Top Down en Bottom Up
 - Continu
 - Welcome changes
 - Mensen, Mensen, Mensen



De oplossing (toegelicht)

- Pas zinvolle elementen uit deze IT-verbeteringen analoog toe op testverbetering
 - Het is een revolutie in denken, geen evolutie in aanpak
 - Oude wijn in nieuwe zakken
 - Uiteindelijk gaat het om het (veranderde) gedrag van mensen

Twee niveaus



Twée niveaus (toegelicht)

- Architectniveau
 - Doelen, grote lijn, kaders
- Implementatieniveau
 - Individuele verbeteringen

Architectniveau

- Business owner aan het stuur
- Stip aan de horizon
 - Waar willen we heen?
 - Waar zijn we nu?
 - Hoe gaan we op weg?



Architectniveau (toegelicht)

- De business zet de stip aan de horizon
 - KISS (Keep It Small and Simple)
- Vanuit welke positie willen we waar naartoe en hoe doen we dat?

Intake

- Business & architect zetten de kaders
 - Doel
 - Wat wil de business bereiken met de testverbetering?
 - Scope
 - Waarop heeft de verbetering betrekking?
 - 'Approach matching' – gereedschapskist versus context
 - Bound: TPI, TMMi, CTP, STEP, ...
 - Unbound: ervaring, heuristiek, intuïtie, brainstorm, ...



Intake (toegelicht)

- De business aan zet
 - Welk probleem moet worden aangepakt?
 - Wat zijn de randvoorwaarden?
 - Wat zijn de tijdlijnen?
- Scope
 - Wat pakken we mee?
 - ... en wat niet (of later)?
- Approach matching
 - Wat (van de bestaande methoden en technieken) kunnen we waarvoor gebruiken?
 - Pick, mix & match
 - Niet dogmatisch, niet voor de eeuwigheid

-



Assessment (toegelicht)

- Waar staan we nu?
 - Soms kun je dat het beste analyseren met een beproefde aanpak zoals TPI: veel werk, gedetailleerd resultaat
 - Soms kun je beter een globale schets maken: snel en doelgericht maar weinig detail
 - De context bepaalt

Improvement release

- Improvement epic:
concreet, samenhangend, haalbaar verbeterdoel



Improvement release (toegelicht)

- Het grote plaatje
 - Wat willen we over een maand of drie (?) hebben bereikt?
 - Motiverend uit te leggen aan alle betrokkenen
 - 'What is in it for me?'

Improvement releaseplan

- Improvement epic
 - ➔ improvement stories
 - Business value
 - Prioriteit
 - Gesized
 - Acceptatiecriteria
- Improvement backlog
 - Verbetertaken
 - Assessment taken

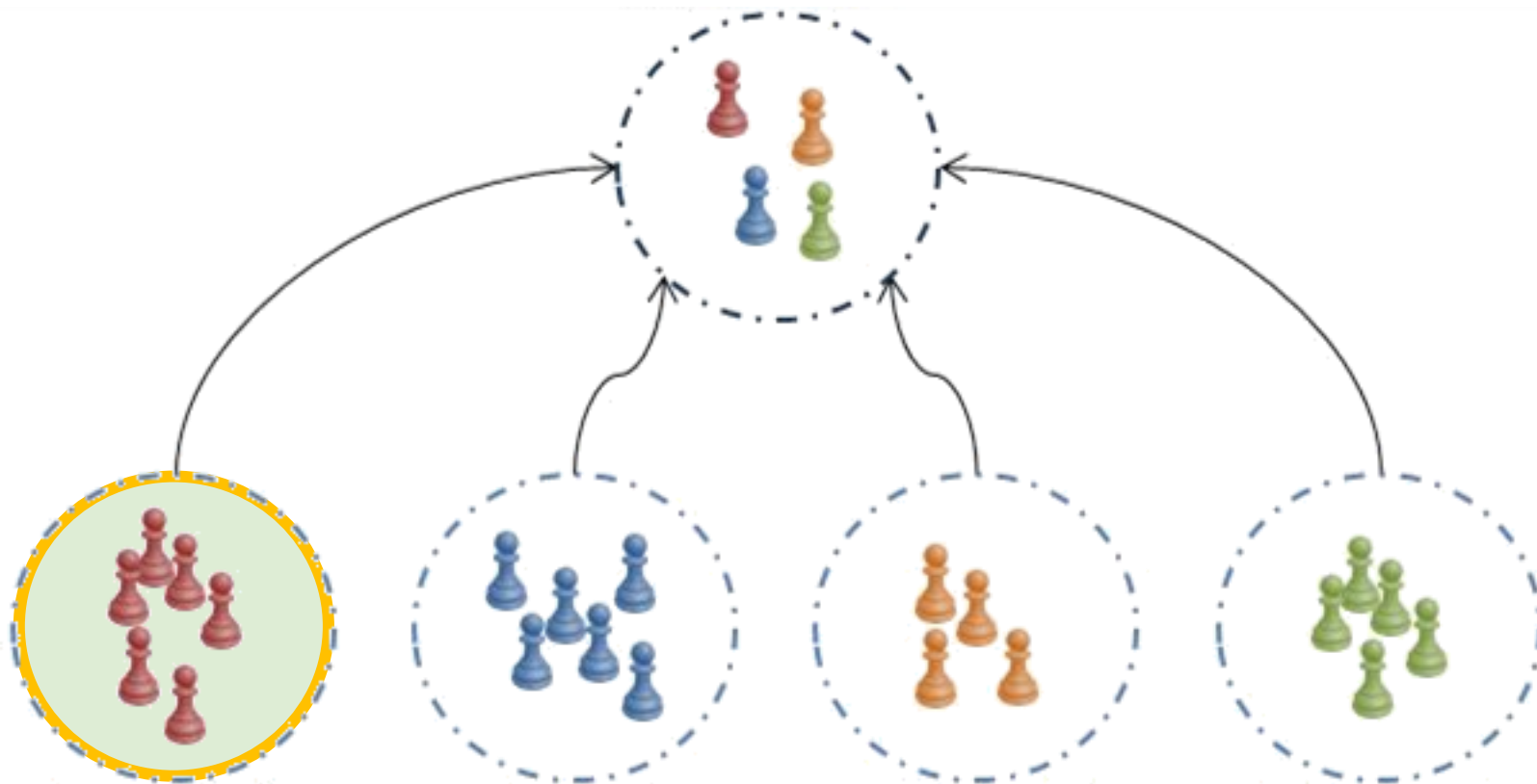


Improvement releaseplan (toegelicht)

- Vertaal het grote releasedoel in een aantal verhaaltjes
 - Met aantoonbare waarde voor de business
 - Met een prioriteit voor de business
 - In één (korte) verbetercyclus te realiseren
 - Met een concrete manier om aan te tonen dat het werkt
- Zet ieder verhaal op een backlog
 - Naast verbeteringen ook assessments

Scrum of Scrums

- Taken zo veel mogelijk beleggen bij operationele teams: Business As Usual
 - [Apart team voor generieke verbeteringen ...]



Scrum of Scrums (toegelicht)

- Beleg de verbeteringen zo veel mogelijk bij lopende projecten / sprints
 - Dan gaan ze naadloos mee met de rest van de opleveringen
 - Natuurlijk wel de benodigde extra resources leveren
- Voor bepaalde generieke / overhead-verbeteringen is soms een separaat verbeterspoor / -sprint / -project nodig
 - Alleen als het niet anders kan

Implementatieniveau

- Stroom van verbeteringen binnen gestelde kaders
 - Improvement owner uit de business
 - Improvement teams: testers, ontwikkelaars, beheerders, ...
 - Improvement architect als Scrum master



Implementatieniveau (toegelicht)

- Sprint teams en projecten pakken individuele verbeteringen op die bij hen op de backlog / to do lijst staan
 - Onder aansturing van iemand uit de business
- De improvement architect loopt mee om de grote lijn in de gaten te houden, te zorgen dat alle verbeteringen op elkaar aansluiten

Plan

- Verbeteringen
 - Selecteren van backlog o.b.v. business value en prioriteit
- Sprint planning
 - Improvement owner
 - Sprint team
 - Sprint backlog
 - Linking pin naar Scrum of Scrums



Plan (toegelicht)

- Individuele verbeteringen worden planmatig opgepakt op basis van de behoeften van de business
- Vanuit de sprints en/of projecten wordt iemand afgevaardigd naar het architectniveau
 - Daar komen de verbeteringen samen

Do

- Verbeteren
 - Mensen, mensen, mensen
 - What's in for me?
 - Direct aan de slag
 - Direct meten
 - Olievlek



Do (toegelicht)

- Klein beginnen
- Laten zien (aantonen) dat het werkt
- Mensen enthousiast maken
- Van daaruit verder uitrollen

Check

- Retrospective
 - Wat gaat beter? ... *in de business*
 - Wat ging goed en waarom?
 - Wat viel tegen en waarom



Check (toegelicht)

- Iedere geïmplementeerde verbetering evalueren
 - Is het beoogde business resultaat bereikt?
(outputsturing)
 - Hoe verliep het verbeterproces?
 - Wat behouden, wat de volgende keer anders doen?

Act

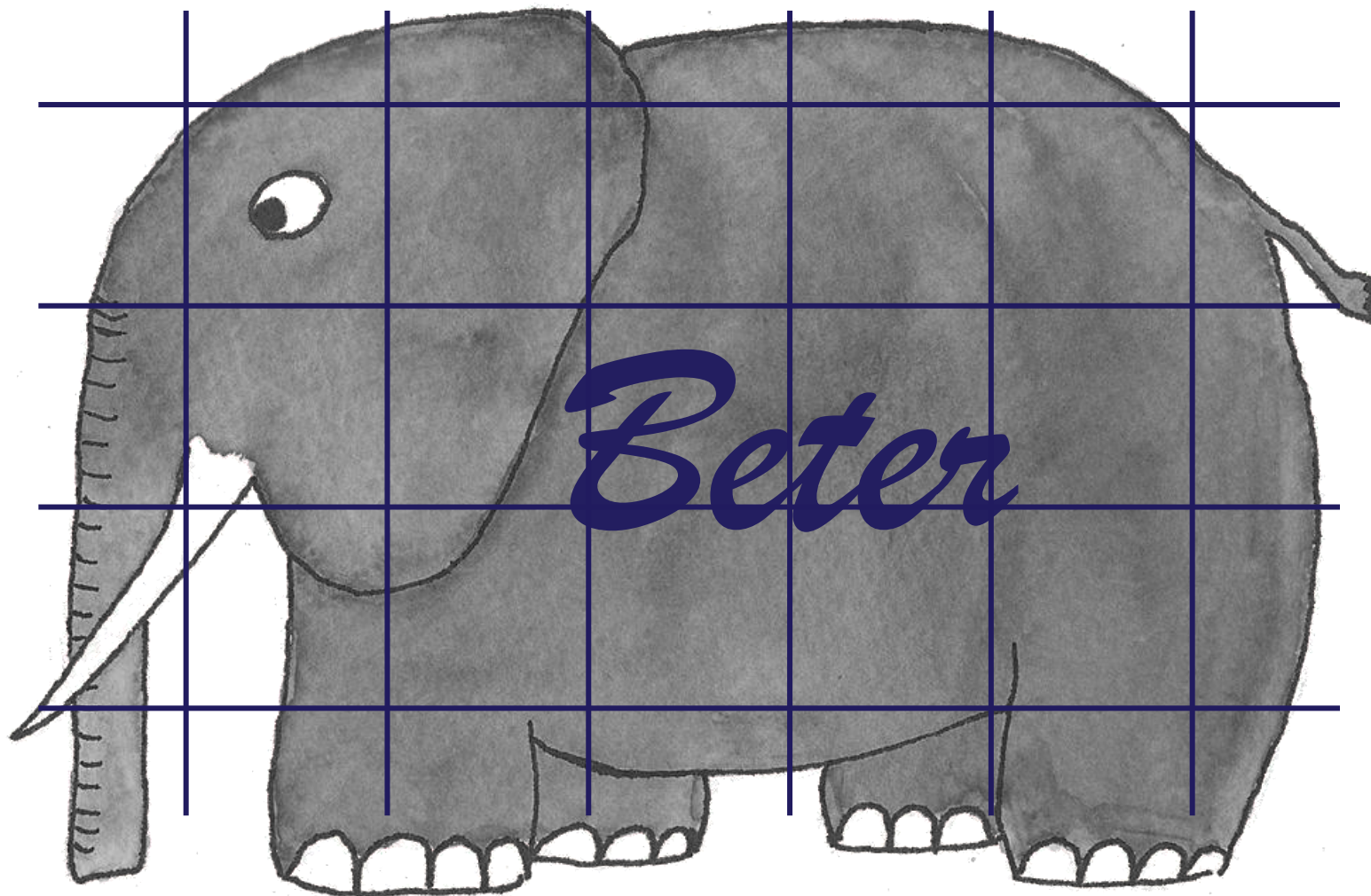
- Wat wordt de volgende stap?
 - Grooming, aanpassen, herprioriteren van backlog



Act (toegelicht)

- Leren van iedere verbetering
 - De backlog van nog te implementeren verbetering nalopen
 - Indien nodig aanpassen op basis van ervaringen uit het verleden en verwachtingen voor de toekomst
 - Business inbreng essentieel

De oplossing



De oplossing (toegelicht)

- Geen grote roze olifant
 - maar een gewone grijze
- zodanig verdeeld in hapklare brokken
 - dat het eetbaar is geworden
 - eet alleen wat voor jou lekker en gezond is

Testverbetering in iedere situatie

- Pragmatisch, doelgericht, snel
 - Focus op door business bepaalde scope
 - Approach matching
 - Hybride aanpak: mix van bound en unbound modellen
- Continue stroom van directe verbeteringen
 - Gericht op stip aan de horizon
 - Alert en flexibel afgestemd op omgeving in beweging
- Toegevoegde waarde voor business staat voorop
 - Gestuurd door improvement owner
 - Meetbaar in termen van bijdrage aan business doelen

