

TESTWERK^{ICT}

Testen en RUP in vogelvlucht



info@testwerk.nl
Testwerk.nl



Martin Deiman

Inhoud

- ▶ Wat is testen
- ▶ Wat is RUP
- ▶ Fases
- ▶ Fases versus disciplines
- ▶ Disciplines en rollen
- ▶ Teamsamenstelling
- ▶ Iteraties en releases

*Waar denk je aan als je
het woord TESTEN hoort?*

TESTWERK^{ICT}

Wat gaan we behandelen?

- Wat is testen, waarom testen en wat is er te testen
- W-Model
- Op welke aspecten kun je testen
- Risico gebaseerd testen
- Testtechnieken
- Testtools
- Met wie heeft de tester te maken?

Wat is testen?

Wat is testen, waarom testen en wat is er te testen

- Simpele definitie: Vinden van fouten
- Veel gebruikte definitie: Vaststellen of iets werkt conform specificatie
- T-Map Next definitie: Testen is een proces dat inzicht geeft in- en adviseert over de kwaliteit en de daaraan gerelateerde risico's
- “Testing is an extremely creative and intellectually challenging task” – G.J. Myers

Waarom testen?

Wat is testen, waarom testen en wat is er te testen

- Inzicht krijgen in de kwaliteit
- Voorkomen van schade, afwegen van risico's
- Advies geven/krijgen voor in productie name
- Vertrouwen en draagvlak kweken

Testen hoeft niet

Het is een afweging tussen het risico dat je wilt lopen en het risico dat je kunt lopen.

Testen en RUP in vogelvlucht

Wat is testen, waarom testen en wat is er te testen



Stoeltje testen:

- Hoe ver ga je?
- Wat zijn de risico's?
- Wat wel en niet testen?
- Hoe testen?

TESTWERK^{ICT}

Wat is er te testen (testobjecten)?

- Hardware
- Systeemsoftware
- Applicatiesoftware
- Conversiesoftware
- (Bedrijfs-)Processen
- Documentatie
- Implementatie (-draaiboek)

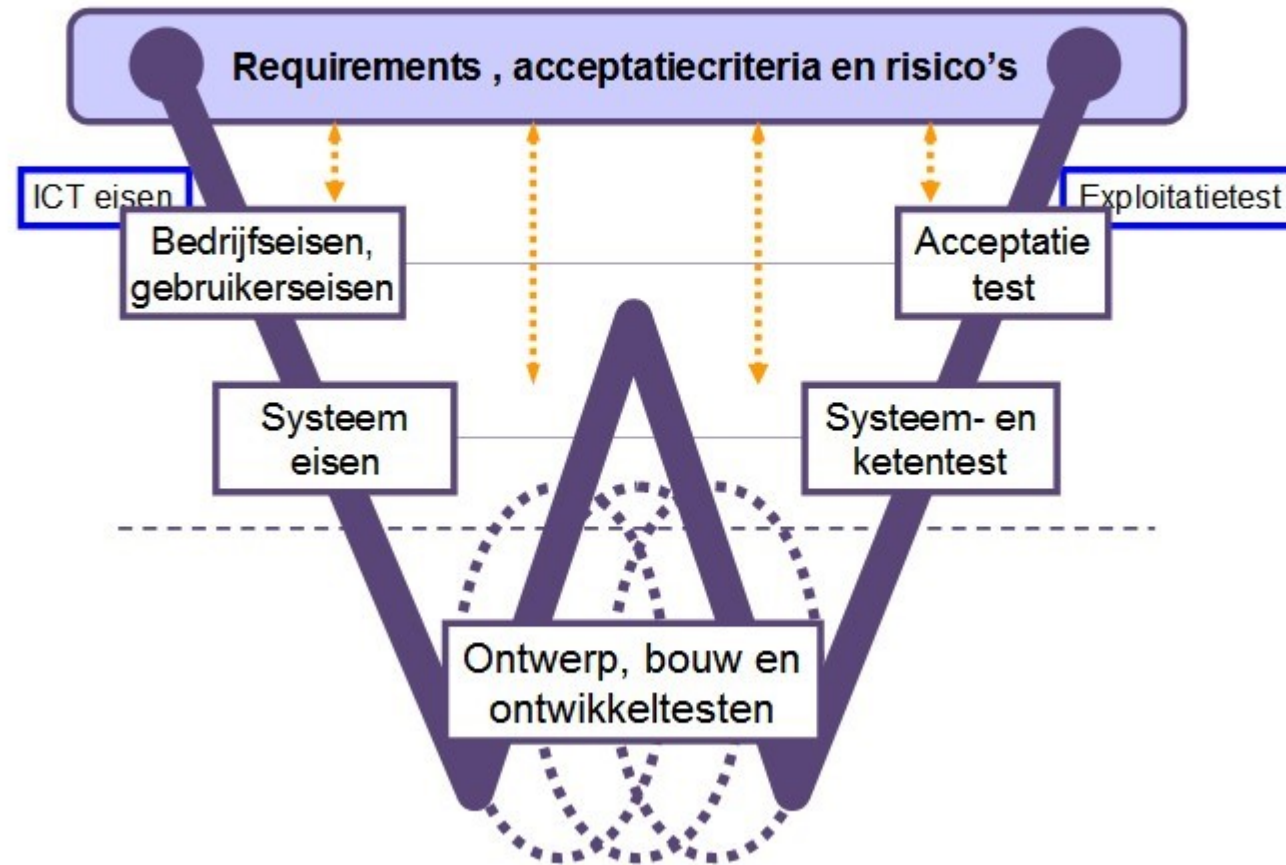
Wat is er te testen?

- Mobiele devices en Apps
 - Touch screen interface
 - Meerdere schermformaten
 - Ander gebruik dan desktop en web

- Andere nieuwe platformen
 - Smartwatch
 - Google Glass
 - Navigatie
 - Car apps
 - Cloud software



Het W-model



Op welke aspecten kun je testen? *(kwaliteitsattributen)*

Portability	Adaptability, Installability, Co-existence, Replaceability, Compliance
Efficiency	Time behavior (performance), Resource utilization, Compliance
Reliability	Maturity (robustness), Fault-tolerance, Recoverability, Compliance
Functionality	Suitability, Accuracy, Security , Interoperability, Compliance
Usability	Understandability, Learnability, Operability, Attractiveness, Compliance
Maintainability	Analyzability, Changeability, Stability, Testability, Compliance

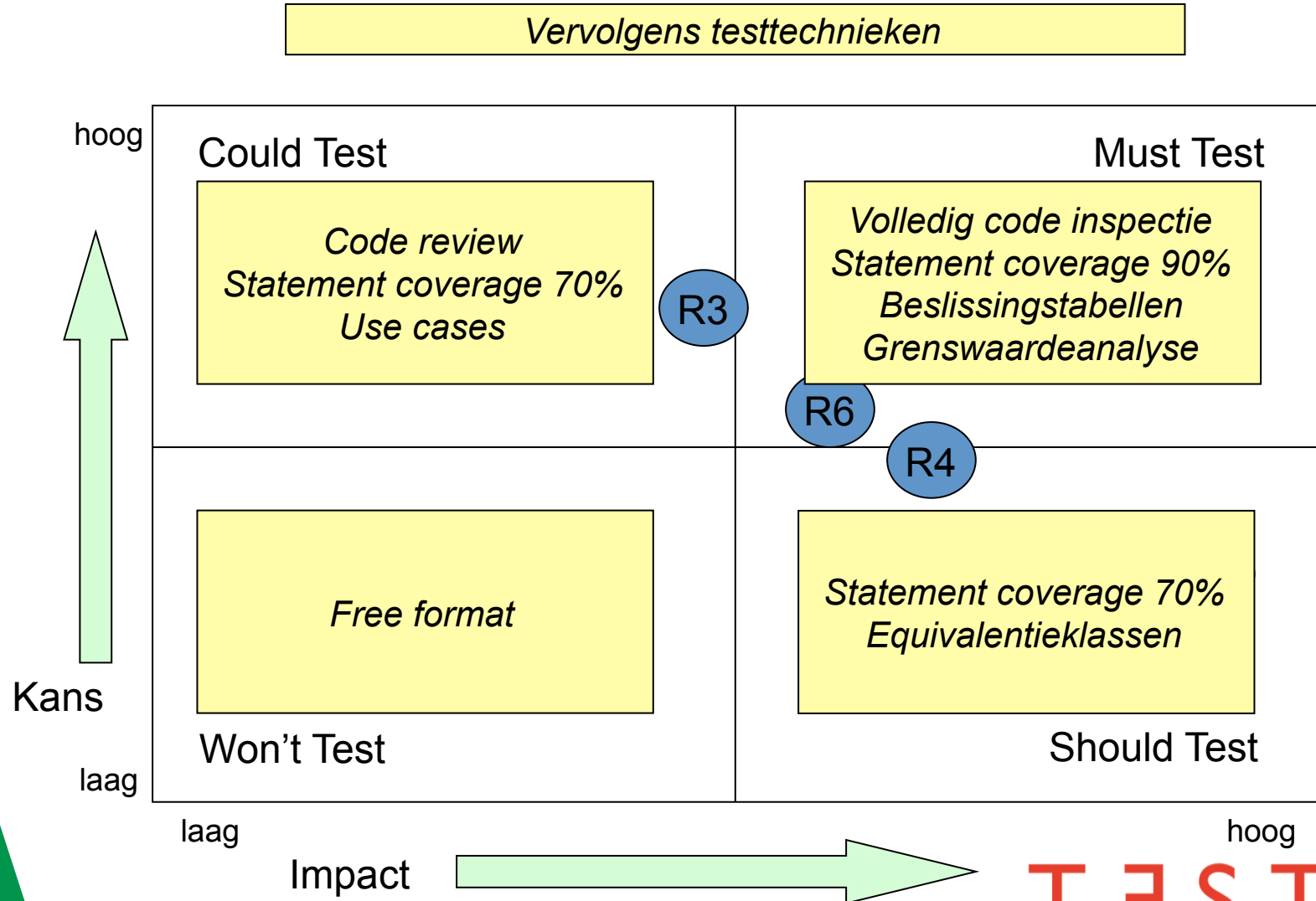
Testen op basis van risicoafweging

Risico gebaseerd testen

- Risico identificatie: bepaal per functionaliteit welke risico's er gelden (voorbeeld 10 risico's)
- Bepaal per risico kans dat het optreedt.
 - complexiteit
 - nieuwe ontwikkeling
 - interfaces
 - technologie
 - ervaring van bouwers
- Bepaal per risico de impact.
 - Belang voor gebruikers
 - Gebruiks intensiteit
 - Veiligheid

Testen op basis van risicoafweging

Risico gebaseerd testen



Met welke technieken kun je testen?

Statische testen (*zonder de te testen software uit te voeren*)

- Reviews
- Analyse

Dynamische testen (*de te testen software wel uitvoeren*)

- Structure based (white box)
- Experience based
- Specification based (black box)

Veel gebruikte testtechnieken

- Use Case testing
 - Compleet stuk functionaliteit met concreet resultaat
- Exploratory testing
 - Systeem stapje voor stapje ontdekken en dan ter plekke testen
 - Testontwerp tijdens testuitvoer
- Goedpaden en Foutpaden
 - Gedefinieerde foutsituaties aflopen
- Gegevens- of Procescyclustest
 - Flow van gegevens door het systeem met alle afslagen
 - Volledige proces van systeem en inpasbaarheid in bedrijfsproces

Testtools

➤ Testautomatisering

- Wegnemen van herhalende taken bij tester
- Regressietesten / Record & playback
- Load & performancetesten



➤ Testuitvoering

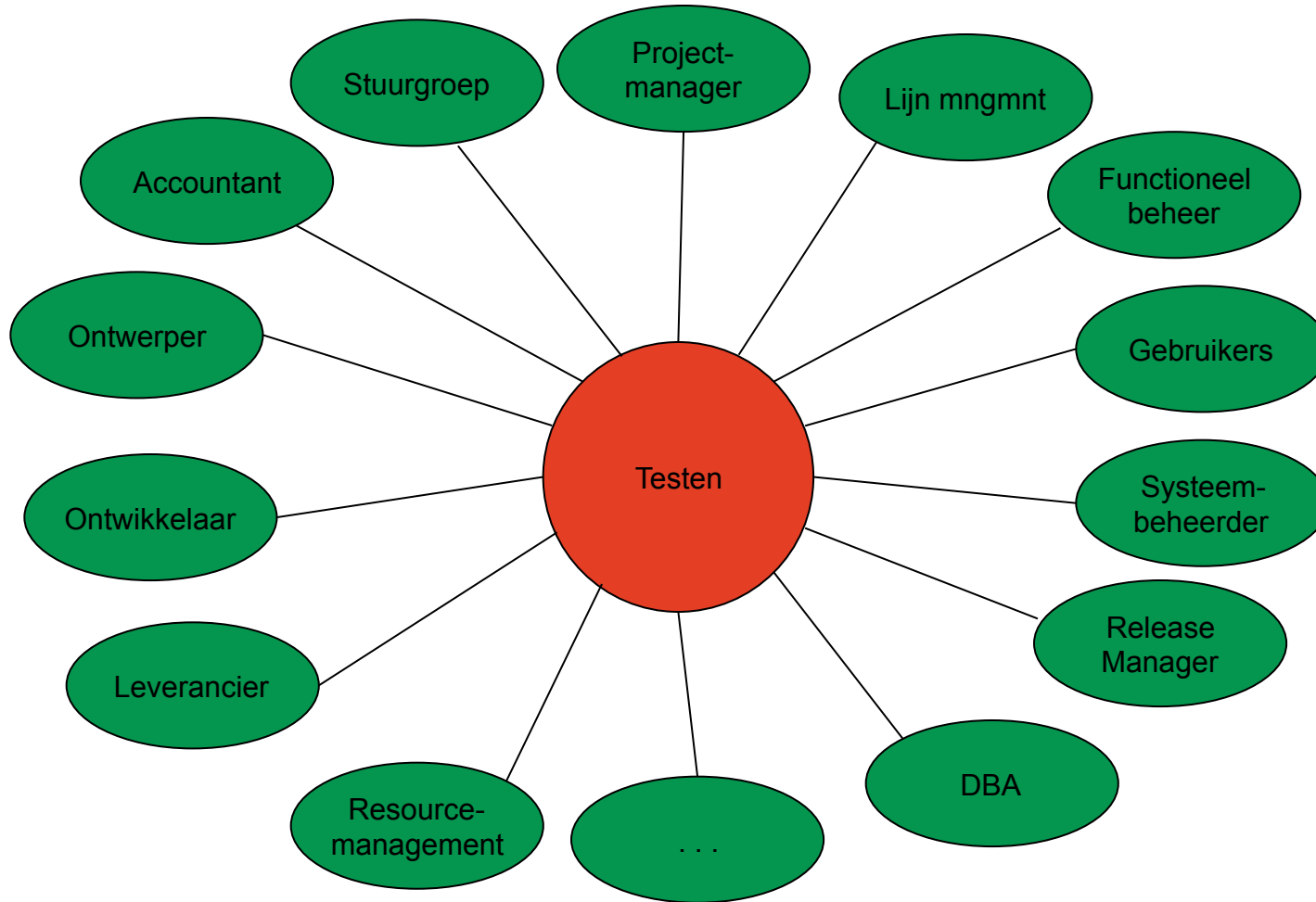
- Inschieten van berichten
- Tools om input/output te bekijken (bijv. XML)
- Bekijken van wijzigingen in systeem (database)
- Ook Stubs en Drivers



➤ Ondersteunende tools

- Vastleggen testgevallen
- Vastleggen bevindingen

Met wie heeft de tester te maken?



Inhoud

- ▶ Wat is testen
- ▶ Wat is RUP
- ▶ Fases
- ▶ Fases versus disciplines
- ▶ Disciplines en rollen
- ▶ Teamsamenstelling
- ▶ Iteraties en releases

RUP =

- ▶ Rational Unified Process (IBM)
- ▶ Sinds 1998 op de markt
- ▶ Een iteratief software-ontwikkelp proces
- ▶ Een verzameling Best Practises, afgeleid van projecten die geslaagd zijn

Best Practices

- ▶ Ontwikkel Software incrementeel/iteratief
- ▶ Management van Software requirements
- ▶ Component gebaseerde architectuur
- ▶ Modeleer Visueel (UML) / Prototypes
- ▶ Bewaak de kwaliteit / Test
- ▶ Maak gebruik van versiebeheer.

Inhoud

- ▶ Wat is testen
- ▶ Wat is RUP
- ▶ Fases
- ▶ Fases versus disciplines
- ▶ Disciplines en rollen
- ▶ Teamsamenstelling
- ▶ Iteraties en releases

Fases

- ▶ Inception : Scope
 - ▶ Elaboration : Risico's
 - ▶ Construction : Functionaliteit
 - ▶ Transition : Overdracht
-
- ▶ Fases kunnen nooit overlappen
 - ▶ Fases hebben exit-criteria, als die niet gehaald worden kan de fase niet afgesloten worden
 - ▶ Fase afsluiting is een opdrachtgevers actie.

Inception

Elaboration

Construction

Transition

Fase: Inception

- ▶ Focus op **Scope**
- ▶ Wat gaan we doen?
- ▶ Wie zijn de stakeholders? En waarvoor?
- ▶ Met welke (kandidaat) architectuur?
- ▶ Welke risico's zijn er?
- ▶ Wat zijn de eisen, met welke prio?
- ▶ Is het haalbaar? Business case.

Inception

Elaboration

Construction

Transition

Fase: Elaboration

- ▶ Focus op **risico**'s
- ▶ Hoe gaan we het doen?
- ▶ Werkt de (kandidaat) architectuur?
- ▶ Ontwerp de belangrijkste(risicovolle) Use cases
- ▶ Minimaliseer de (hoogste) risico's; bouw en test
- ▶ Zet de architectuur neer en toon aan.

Inception

Elaboration

Construction

Transition

Risico's

Hoog	Avoid	Do First (elaboration)
Laag	Do last (construction)	Do second (construction)
Risico Business value	Laag	Hoog

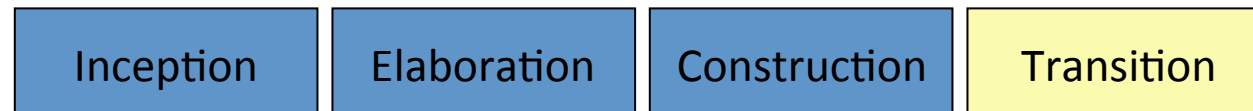
Fase: Construction

- ▶ Focus op **functionaliteit**
- ▶ Doe het: ontwerp, bouw en test
- ▶ Zorg dat het voldoet aan eisen van de gebruikers
- ▶ Doe gebruikersacceptatietesten tijdens iteraties.



Fase: Transition

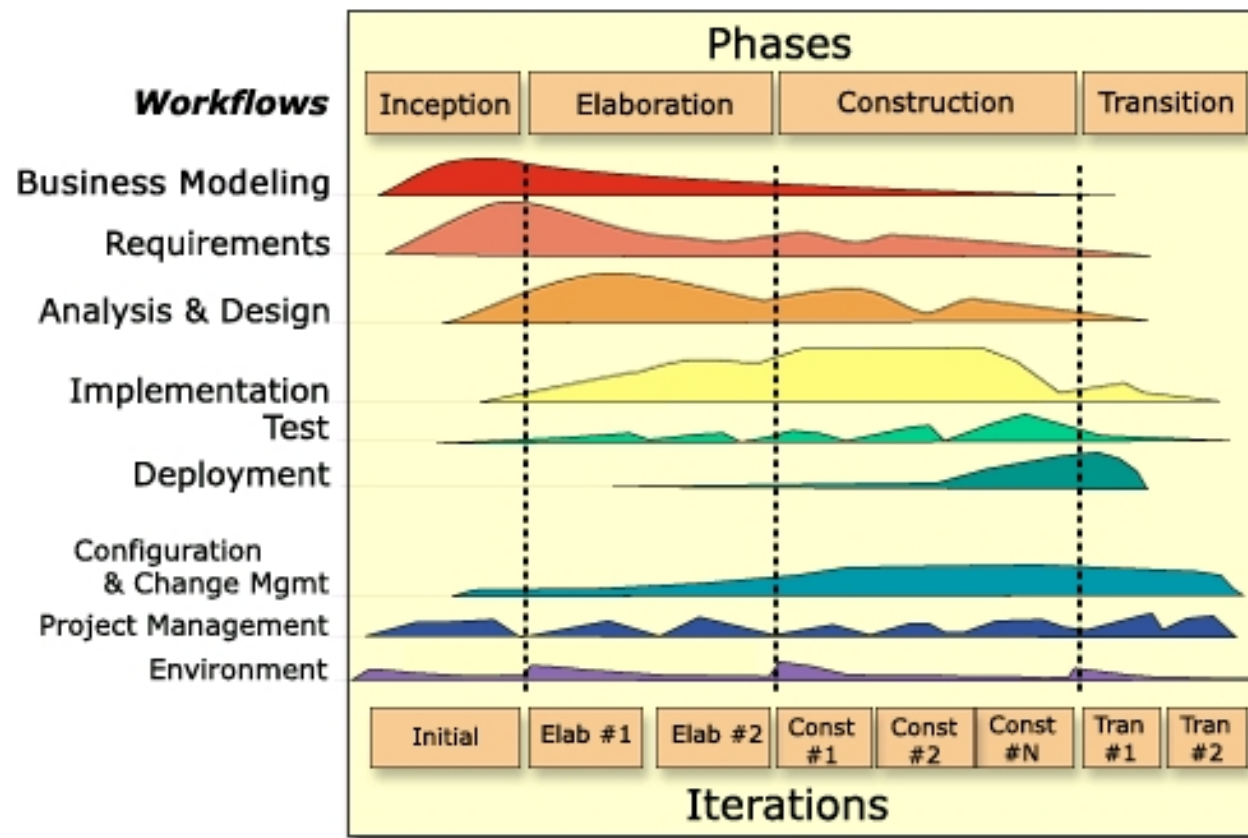
- ▶ Focus op **overdracht**
- ▶ Implementeer de laatste versie in productie
- ▶ Verzorg opleidingen
- ▶ Draag het systeem over aan gebruikers en beheer.



Inhoud

- ▶ Wat is testen
- ▶ Wat is RUP
- ▶ Fases
- ▶ Fases versus disciplines
- ▶ Disciplines en rollen
- ▶ Teamsamenstelling
- ▶ Iteraties en releases

Fases en disciplines

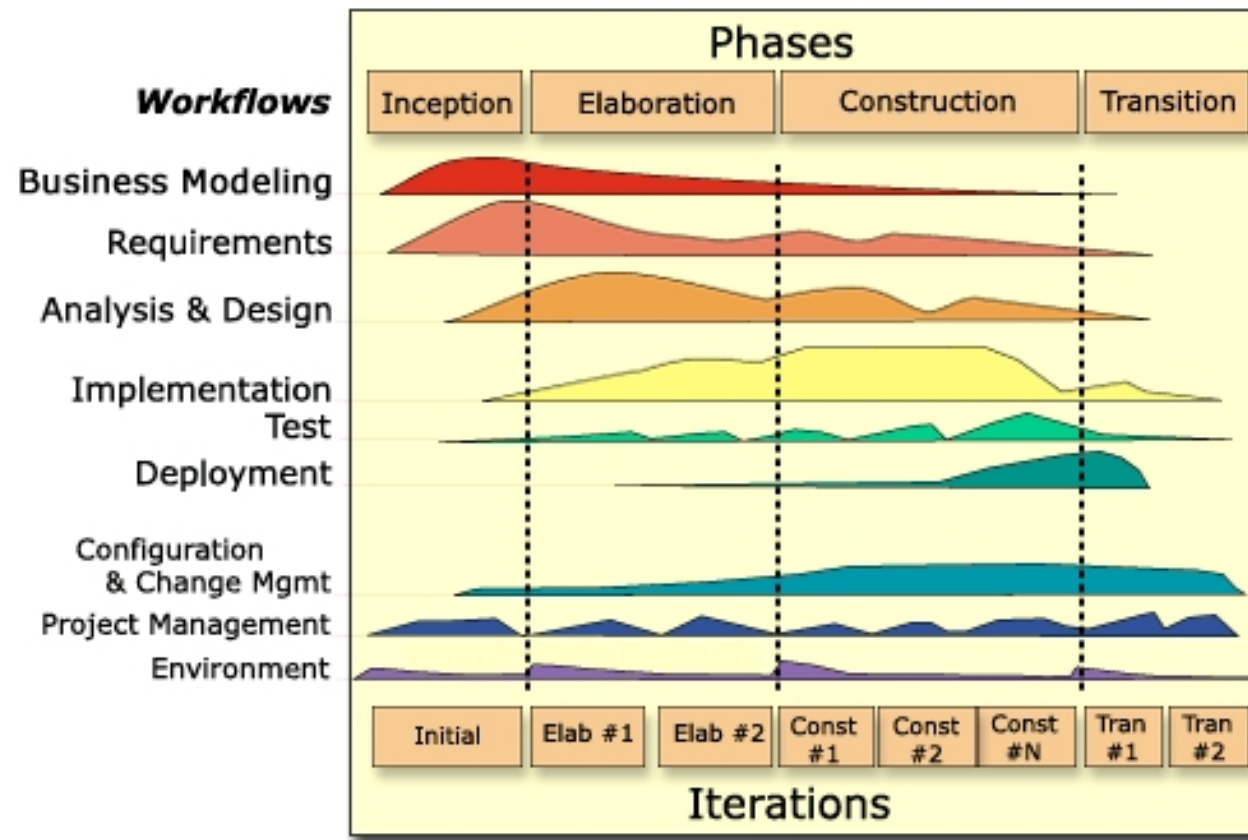


Business modeling

Geeft inzicht in hoe het product zal worden gebruikt in zijn operationele omgeving

- ▶ Beschrijf de huidige business
- ▶ Bepaal de scope
- ▶ Laat zien hoe het systeem de business ondersteunt
- ▶ Beschrijf de bedrijfsprocessen, business use cases en business actors (business use case model)
- ▶ Beschrijf het domeinmodel
- ▶ Beschrijf de business requirements.

Fases en disciplines

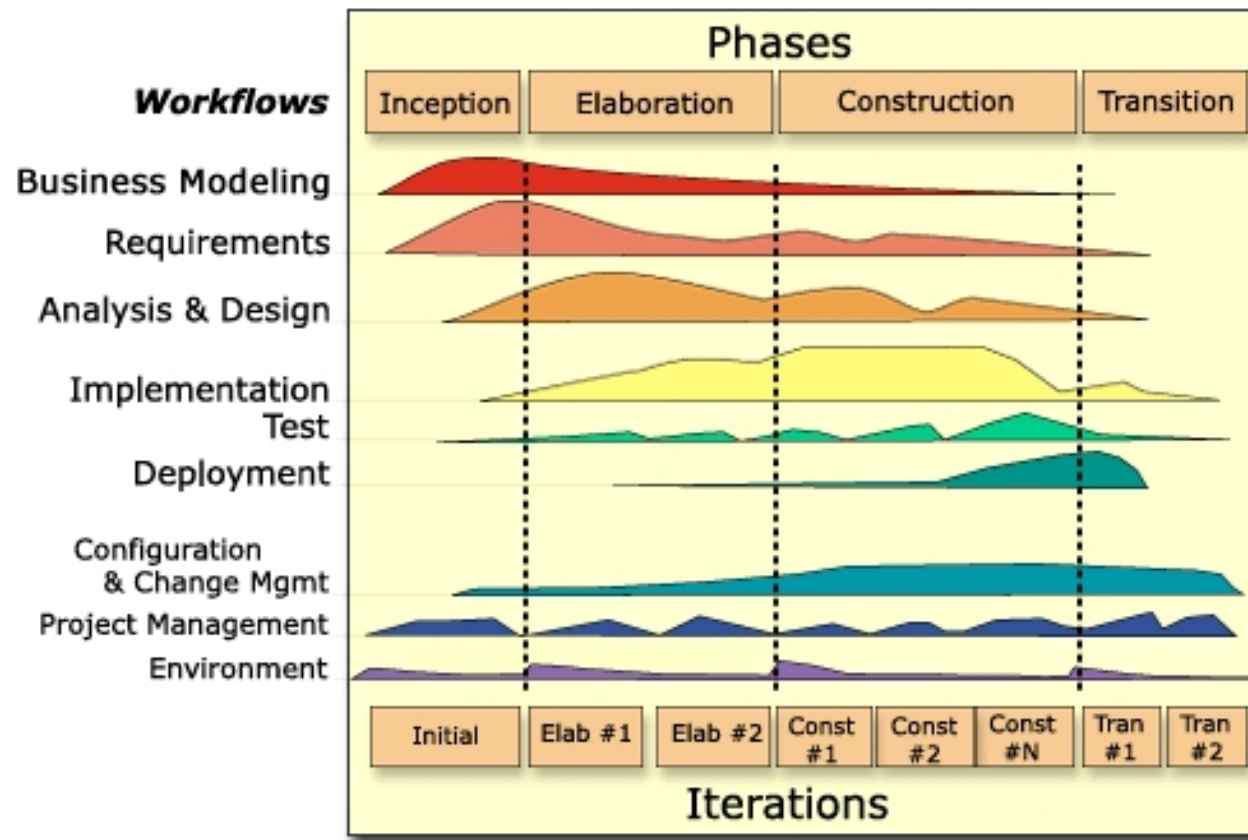


Requirements

Beschrijft wat het product moet bieden en hoe goed het product dat moet doen.

- ▶ Achterhalen en beschrijven requirements
- ▶ Bepaal wat het systeem moet doen
- ▶ Ontwikkel Use Cases
- ▶ Aanvullende technische en functionele specificaties
- ▶ User interface prototypes.

Fases en disciplines

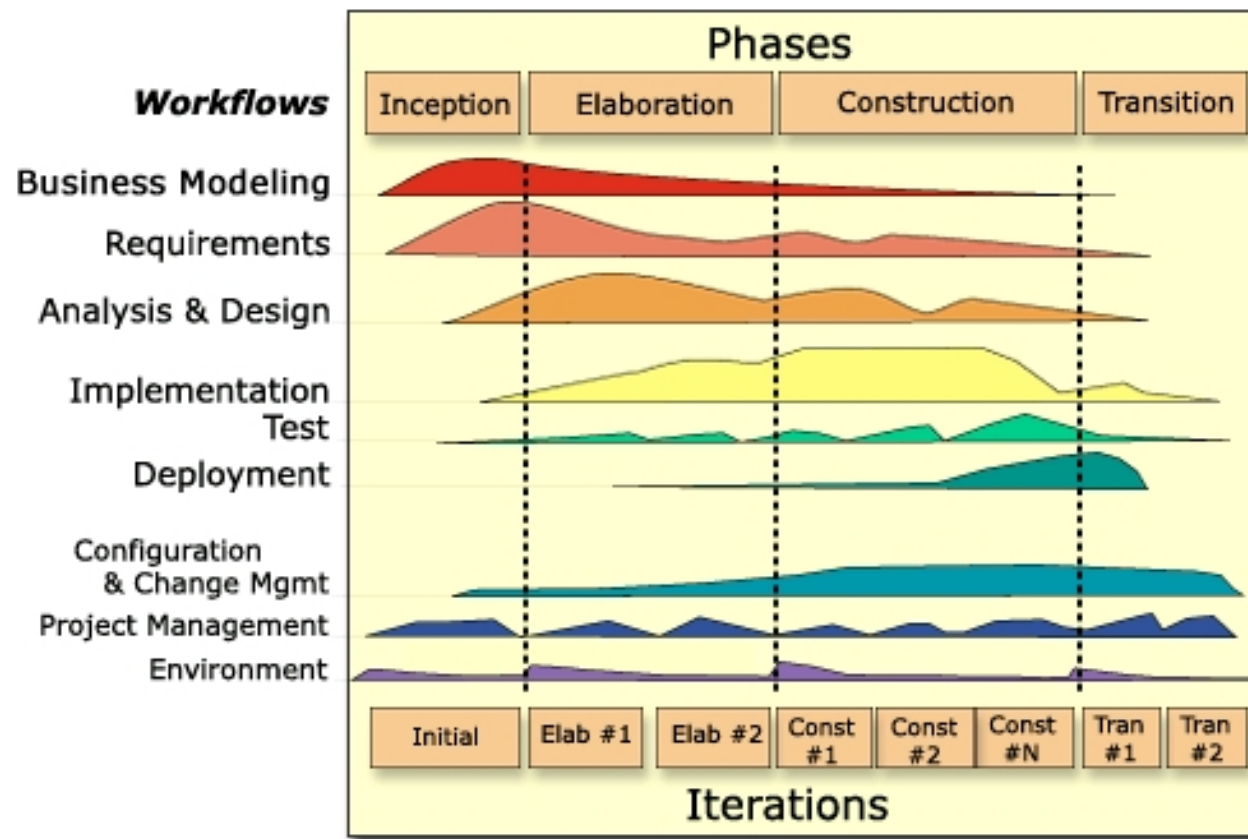


Analysis & Design

Bepaalt hoe het product zal gaan voldoen aan functionele & non-functionele eisen.

- ▶ Bepaal hoe de requirements gerealiseerd worden
- ▶ Beschrijf classes, modules, subsystemen en interfaces
- ▶ Ontwerpmodel: beschrijf in een plan hoe de code gestructureerd en geschreven moet worden.

Fases en disciplines

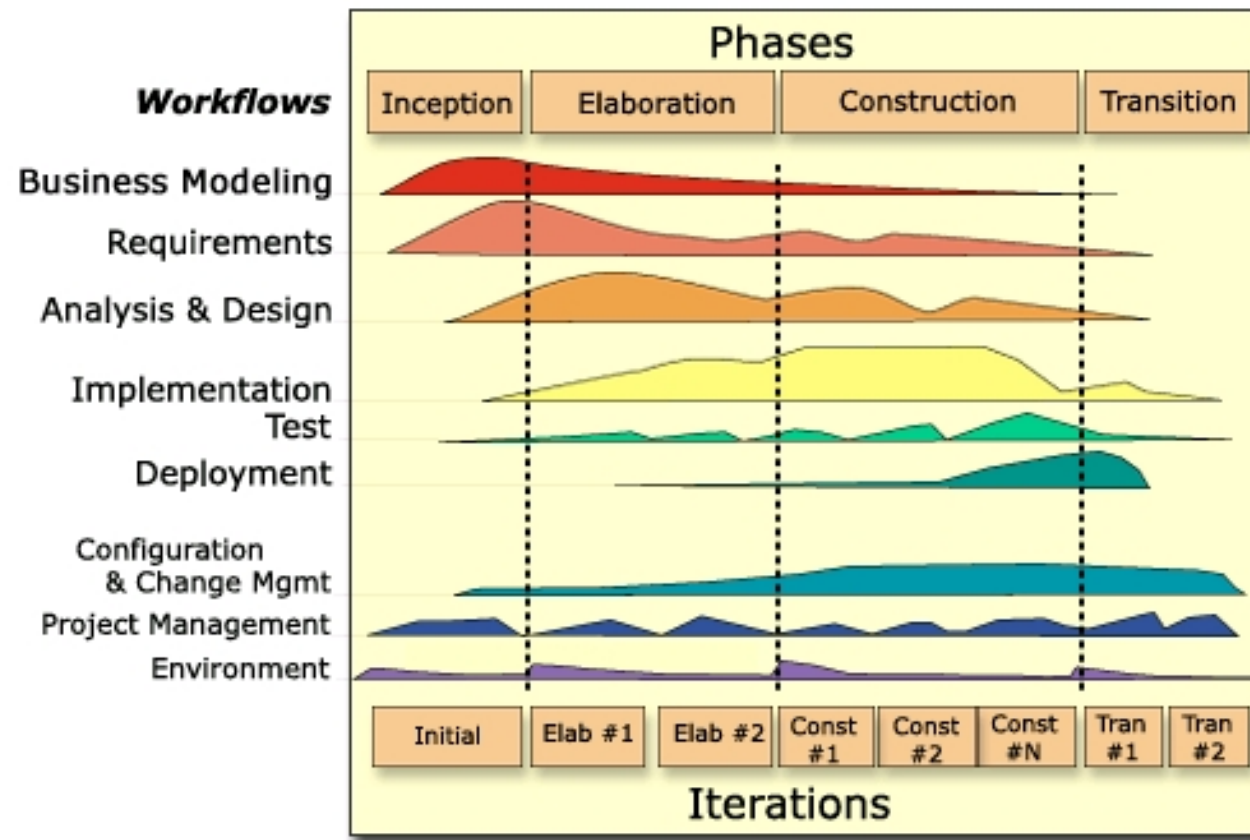


Implementation

Produceert een werkend (deel)product op basis van het ontwerp.

- ▶ Vastleggen van classes en objecten
- ▶ Schrijf de code
- ▶ Unittesten
- ▶ Integreren van alle code.

Fases en disciplines



Kenmerken testen binnen RUP

- Komt terug binnen het hele proces ipv aan het einde van het project
- Belanghebbenden worden betrokken en zijn verantwoordelijk voor de kwaliteit (klanten, programmeurs)
- Veel interactie tussen de verschillende disciplines
- Voornamelijk testen op basis van usecases
- Gebruik van test tools

Voordelen ten opzichte van traditionele testaanpak

- Fouten worden in een eerder stadium opgespoord
- Herstelkosten zijn lager
- (Her)testen kunnen sneller uitgevoerd worden
- Meer communicatie en begrip onderling
- Belanghebbenden krijgen verantwoordelijkheidsgevoel

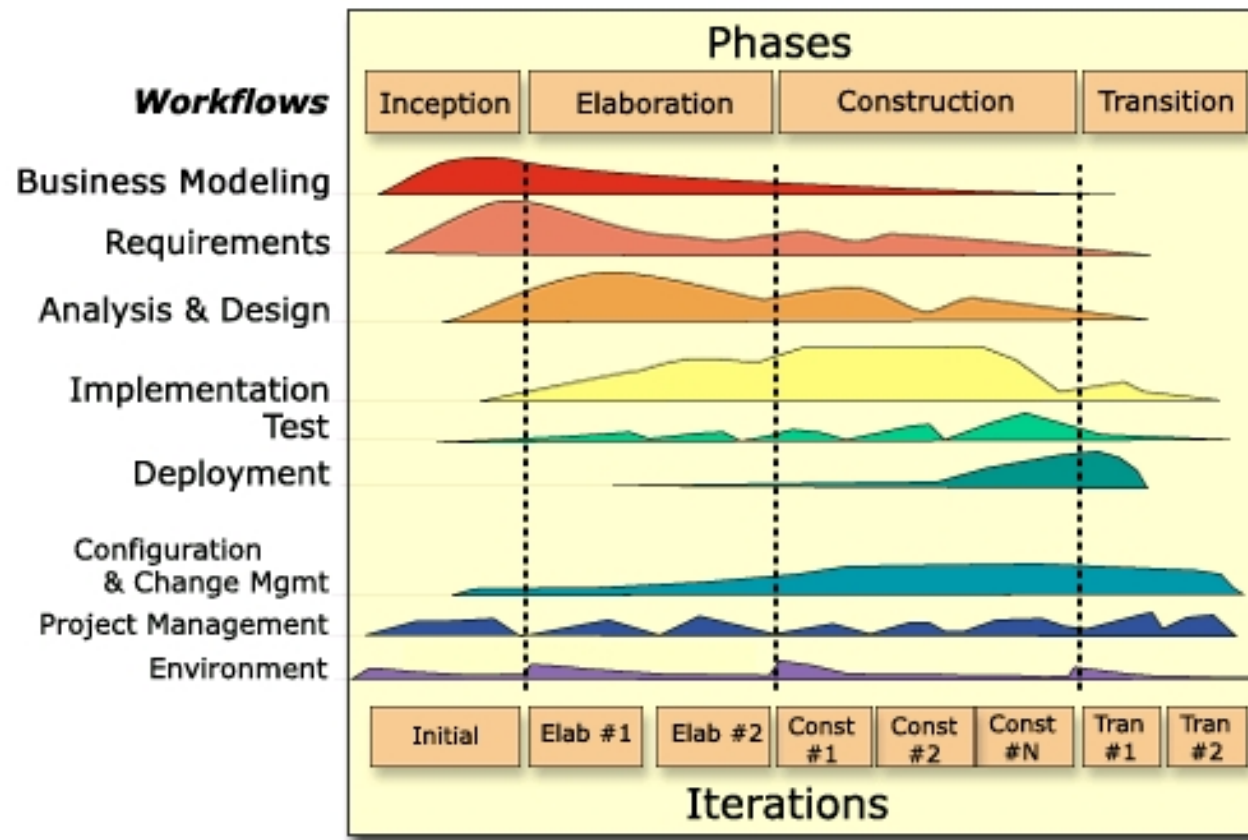
Nadelen

- Het testproces bevat geen gestructureerde testanalyse (voorbereiding van de testuitvoering)
- Weinig documentatie
- Initieel opzetten en inrichten van testtooling kost tijd en geld

Test tools

Veel gebruik van test tools, voornamelijk om testgevallen automatisch uit te voeren en voor het uitvoeren van regressietesten.

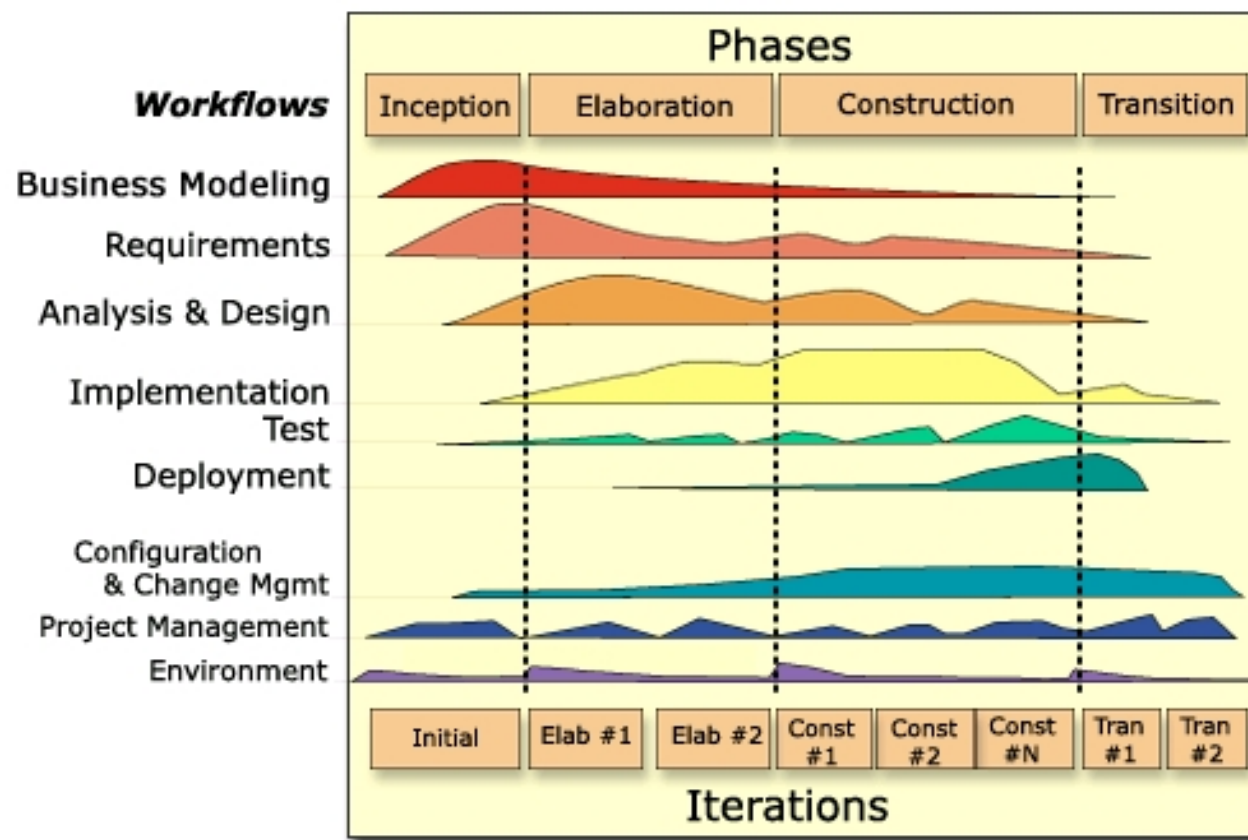
Fases en disciplines



Deployment, C&C, PM, Environment

- ▶ Deployment: Maakt het product beschikbaar voor de eindgebruikers
- ▶ Configuration & Change Management: Beheerst de integriteit van alle project en product artifacts
- ▶ Project Management: Plant en coördineert het behalen van de projectdoelstellingen.
- ▶ Environment: Levert het projectteam een passende omgeving om het plan te kunnen uitvoeren.

Fases en disciplines



Inhoud

- ▶ Wat is testen
- ▶ Wat is RUP
- ▶ Fases
- ▶ Fases versus disciplines
- ▶ Disciplines en rollen
- ▶ Teamsamenstelling
- ▶ Iteraties en releases

Disciplines en rollen (1)

- **Business Modeling**
 - **Business Process Analyst** Discovers all business use cases.
 - **Business Designer** Details a single set of business use cases.
- **Requirements**
 - **Systems Analyst** Discovers all requirement use cases.
 - **Requirements Specifier** Details a single set of requirement use cases.
- **Analysis and Design**
 - **Software Architect** Decides on technologies for the whole solution.
 - **Designer** Details the analysis and design for a single set of use cases.
- **Implementation**
 - **Integrator** Owns the build plan that shows what classes will integrate with one another.
 - **Implementer** Codes a single set of classes or a single set of class operations.
- **Test**
 - **Test Manager** Ensures that testing is complete and conducted for the right motivators.
 - **Test Analyst** Selects what to test based on the motivators.
 - **Test Designer** Decides automated vs. manual and creates automations. Implements automated portions
 - **Tester** Runs a specific test.




























Disciplines en rollen (2)

- **Deployment**
 - **Deployment Manager** Oversees deployment for all deployment units.
 - **Tech Writer, Course Developer, Graphic Artist** Create detailed materials to ensure a successful deployment.
- **Project Management**
 - **Project Manager** Creates the projectplan; prepares go/no go decisions. Plans, tracks, and manages risk.
- **Environment**
 - **Process Engineer** Owns the process for the project.
 - **Tool Specialist** Creates guidelines for using a specific tool.
- **Configuration and Change Mgt**
 - **Configuration Manager** Sets up the CM environment, policies, and plan.
 - **Change Control Manager** Establishes a change control process, Reviews and manages change requests.

Inhoud

- ▶ Wat is testen
- ▶ Wat is RUP
- ▶ Fases
- ▶ Fases versus disciplines
- ▶ Disciplines en rollen
- ▶ Teamsamenstelling
- ▶ Iteraties en releases

Teamsamenstelling:

Waterval	RUP
Bedrijfs Analyse 	Bedrijfs Analyse 
Informatie Analyse 	Inception    
Ontwerp 	Elaboration    
Bouw   	Construction      
Test  	Transition    

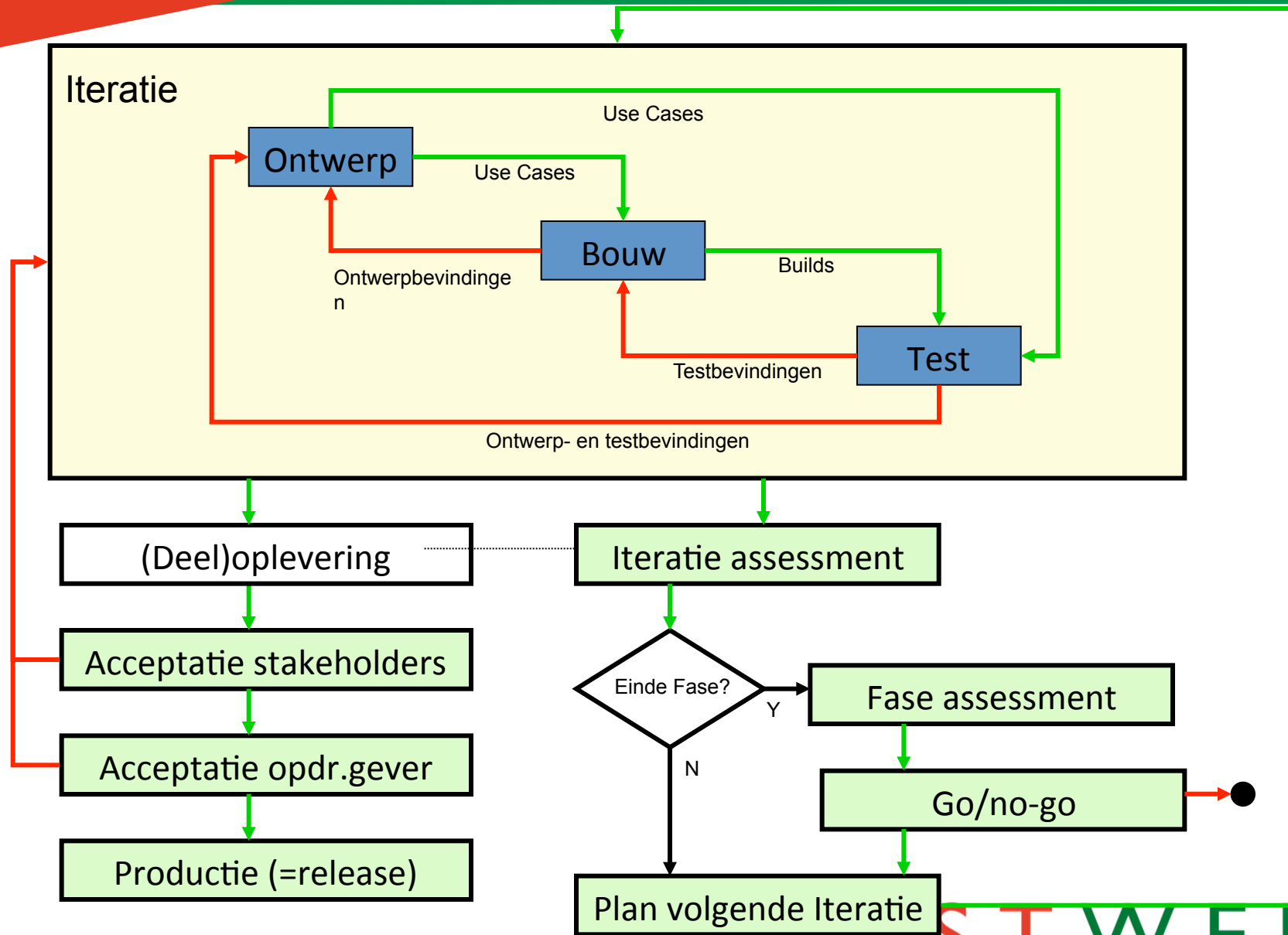
Inhoud

- ▶ Wat is testen
- ▶ Wat is RUP
- ▶ Fases
- ▶ Fases versus disciplines
- ▶ Disciplines en rollen
- ▶ Teamsamenstelling
- ▶ Iteraties en releases

Iteraties

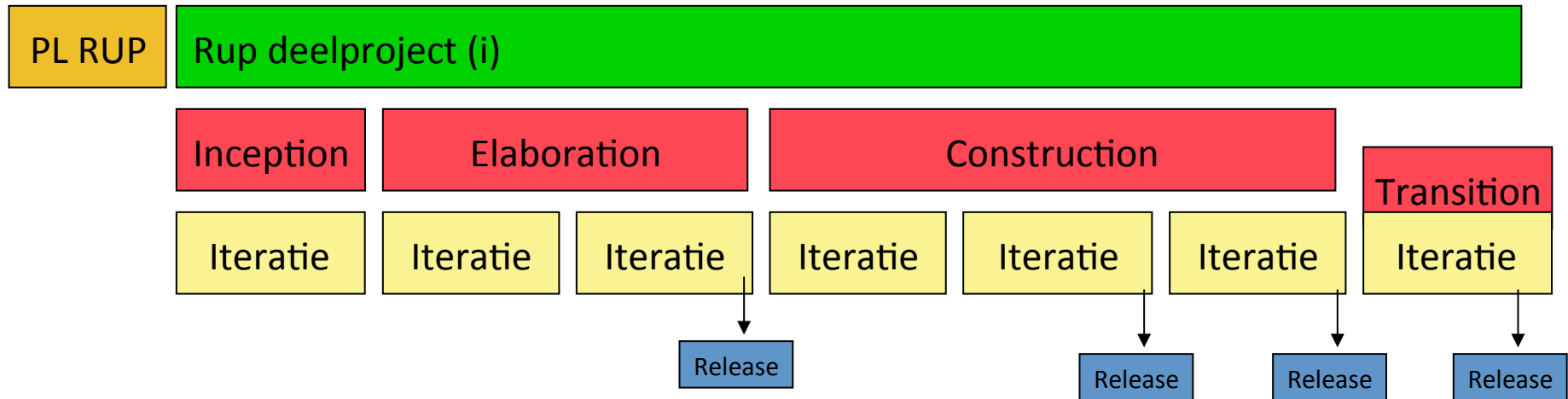
- ▶ Project is een aaneenschakeling van iteraties
- ▶ Iteraties tussen 2 – 4 weken
- ▶ Elke iteratie levert werkende software
- ▶ Iteraties bevatten:
 - ▶ Opgeleverde requirements
 - ▶ Goed genoeg software
 - ▶ Wijzigingen
 - ▶ Risico-minimalisatie
 - ▶ Opgeloste bevindingen
 - ▶ Technische oplossingen in verhouding met de kosten
- ▶ Wie het werk doet plant het werk!

Testen en RUP in vogelvlucht



Releases

- ▶ Meerdere releases mogelijk binnen een project
- ▶ Voordelen:
 - ▶ Eerder resultaat voor de klant
 - ▶ Aantonen van werkbaarheid in Productieomgeving
- ▶ Nadelen:
 - ▶ Meer overhead



Wat wil je nog meer weten?

?