



Stichting NIOC en de NIOC kennisbank

Stichting NIOC (www.nioc.nl) stelt zich conform zijn statuten tot doel: het realiseren van congressen over informatica onderwijs en voorts al hetgeen met een en ander rechtstreeks of zijdelings verband houdt of daartoe bevorderlijk kan zijn, alles in de ruimste zin des woords.

De stichting NIOC neemt de archivering van de resultaten van de congressen voor zijn rekening. De website www.nioc.nl ontsluit onder "Eerdere congressen" de gearchiveerde websites van eerdere congressen. De vele afzonderlijke congresbijdragen zijn opgenomen in een kennisbank die via dezelfde website onder "NIOC kennisbank" ontsloten wordt.

Op dit moment bevat de NIOC kennisbank alle bijdragen, incl. die van het laatste congres (NIOC2023, gehouden op donderdag 30 maart 2023 jl. en georganiseerd door NHL Stenden Hogeschool). Bij elkaar bijna 1500 bijdragen!

We roepen je op, na het lezen van het document dat door jou is gedownload, de auteur(s) feedback te geven. Dit kan door je te registreren als gebruiker van de NIOC kennisbank. Na registratie krijg je bericht hoe in te loggen op de NIOC kennisbank.

Het eerstvolgende NIOC vindt plaats op donderdag 27 maart 2025 in Zwolle en wordt dan georganiseerd door Hogeschool Windesheim. Kijk op www.nioc2025.nl voor meer informatie.

Wil je op de hoogte blijven van de ontwikkeling rond Stichting NIOC en de NIOC kennisbank, schrijf je dan in op de nieuwsbrief via

www.nioc.nl/nioc-kennisbank/aanmelden-nieuwsbrief

Reacties over de NIOC kennisbank en de inhoud daarvan kun je richten aan de beheerder:

R. Smedinga kennisbank@nioc.nl.

Vermeld bij reacties jouw naam en telefoonnummer voor nader contact.



Het (niet) leren van je fouten

Eric van der Vliet

LogicaCMG IDT/TSWE

Senior Performance Improvement Consultant

Uw presentator?

- Eric van der Vliet
- Senior Performance Improvement Consultant
- Verantwoordelijk voor PQI competentie centrum
- 18 jaar ervaring in IT
 - Hardware ontwikkeling
 - Software ontwikkeling (embedded)
 - Project Management
 - Architectuur
 - Assessments (CMM, CMMI)
 - Six Sigma Green Belt

Wat is kwaliteit?

- Verschillende perspectieven
 - Klant tevredenheid
 - Perceptie van de gebruiker
 - Voldoen aan de eisen
 - Fouten in een product
 - ...
- Focus in deze presentatie is op product fouten
 - Waar zijn fouten gemaakt?
 - Waar worden fouten gevonden?
 - En zelfs meer belangrijk

Leren we nu van onze fouten???

Waar worden fouten gemaakt?

- 38% van de fouten worden geïntroduceerd tijdens de bouw fase
- 45% van de fouten worden geïntroduceerd in de requirements en design fase
 - Misinterpretaties
 - Gebrek aan kennis van methoden / technieken
 - Gebrek aan kennis van tools
 - Beperkte inspanning besteed aan reviews en inspecties
 - Etc.

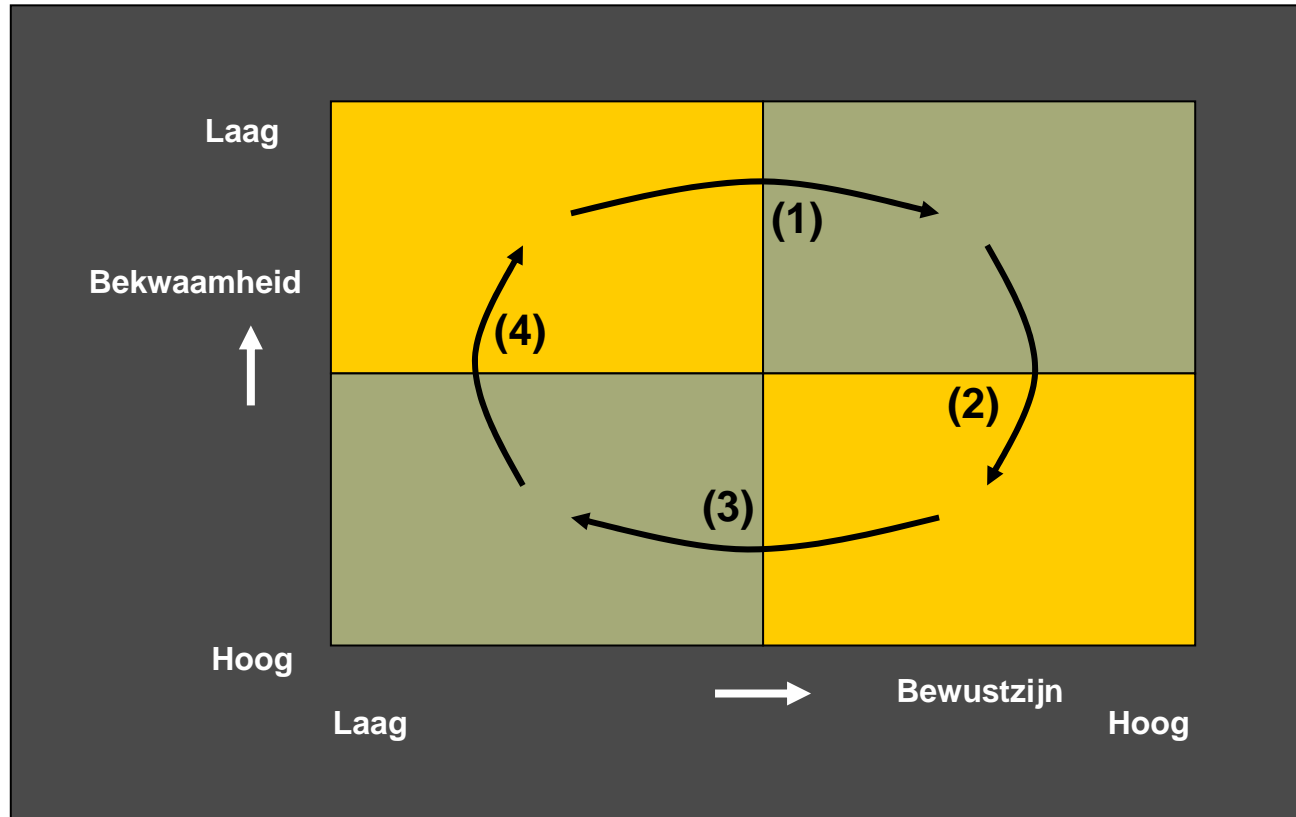
De cijfers!

Fase gedetecteerd

Fase geïnjecteerd

	Requirements	Ontwerp	Bouw	Module test	Integratie test	Systeem test	Na release	Rij
Requirements	9	4	2	6	4	2	1	28
Ontwerp		3	2	2	2	6	2	17
Bouw			6	23	2	5	2	38
Module test				8	1	1	0	10
Integratie test					3	1	1	5
Systeem test						1	1	2
Kolom	9	7	10	39	12	16	7	

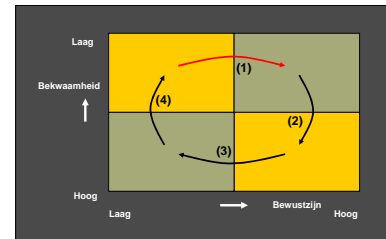
Hoe kunnen we deze situatie verbeteren?



In vier stappen

Step 1: Bewustzijnsfase

- **Onbewust onbekwaam => Bewust onbekwaam**
- Start met het meten van fouten
 - Inspectie fouten
 - Test defects
- Creëer bewustzijn m.b.t. productkwaliteit
- Vergelijk projecten
- Benchmark met andere organisaties / literatuur
- Overtuig jezelf dat actie nodig is



Praktische metingen

- Inspectie data
 - Gebruik een spreadsheet of database
 - Metingen
 - Major defects (fouten) gevonden
 - Aantal logische pagina's (~300 woorden) geïnspecteerd

- Test data
 - Gebruik een defect tracking tool
 - Metingen
 - Aantal gevonden fouten
 - Omvang van het programma (bijv. KLOC)

Ervaringscijfers

Software fouten

[Ron Radice]	24,4	Fouten / KLOC
[Customer sites]	12,7	Fouten / KLOC (C-code)
[Les Hattonn]	8	Fouten / KLOC (C-code)
“ “	12	Fouten / KLOC (Fortran)
[Putnam]	12	Fouten / KLOC (Cobol)
[Grady 1992]	23	Fouten / KLOC

Gemiddeld 15,35 fouten / KLOC

Minder dan 10 fouten / KLOC is acceptabel

Document fouten

[LogicaCMG-1]	3 – 5	Fouten / Logische pagina
[LogicaCMG-2]	0.5 – 3	Fouten / Logische pagina
[LogicaCMG-3]	0.5 – 1	Fouten / Logische pagina

Major defects (fouten) gevonden

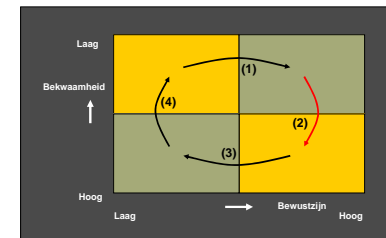
Logische pagina is ongeveer 300 woorden

Minder dan 1 defect (fout) / logical pagina is acceptabel

Step 2: Competentiefase

- **Onbewust onbekwaam => Bewust bekwaam**
- De bewustzijnsfase gaf inzicht in de hoeveelheid fouten
- Maar

Hoeveel tijd is besteed om deze fouten te vinden???



Ervaringscijfers

Fase geinjecteerd	Fase gedetecteerd						
	Requirements	Ontwerp	Bouw	Module test	Integratie test	System test	Na release
Requirements	1.2	8.8	9,85	12,4	14.8	15	18.7
Ontwerp		1,1	5,4	8,1	12,2	13,6	16,75
Bouw			3,2	4,03	9,7	12,2	14,8
Module test				3,93	8,6	11,15	13,65
Integratie test					6,7	12	17,3
System test						11,8	16,8

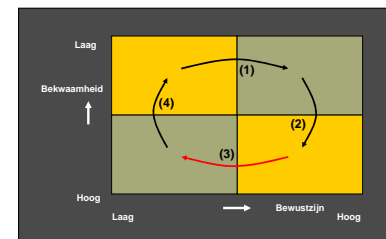
Hoe effectief is het vinden van een fout in een bepaalde fase?

Fase	Effectiviteit
Document Inspectie	50-60%
Code Inspectie	40-70%
Module test	20-60%
Integratie test	15-46%
Acceptatie test	46-50%

Step 3: Institutionaliseringsfase

- **Bewust bekwaam => Onbewust bekwaam**
- We weten hoeveel fouten we maken
- We kennen de kosten voor het vinden van de fouten
- We weten de effectiviteit en efficiency van verschillende methoden
- Zo, we kunnen keuzes maken uit de verschillende verificatie methoden
- Maar

Hoe kunnen we leren van deze fouten???



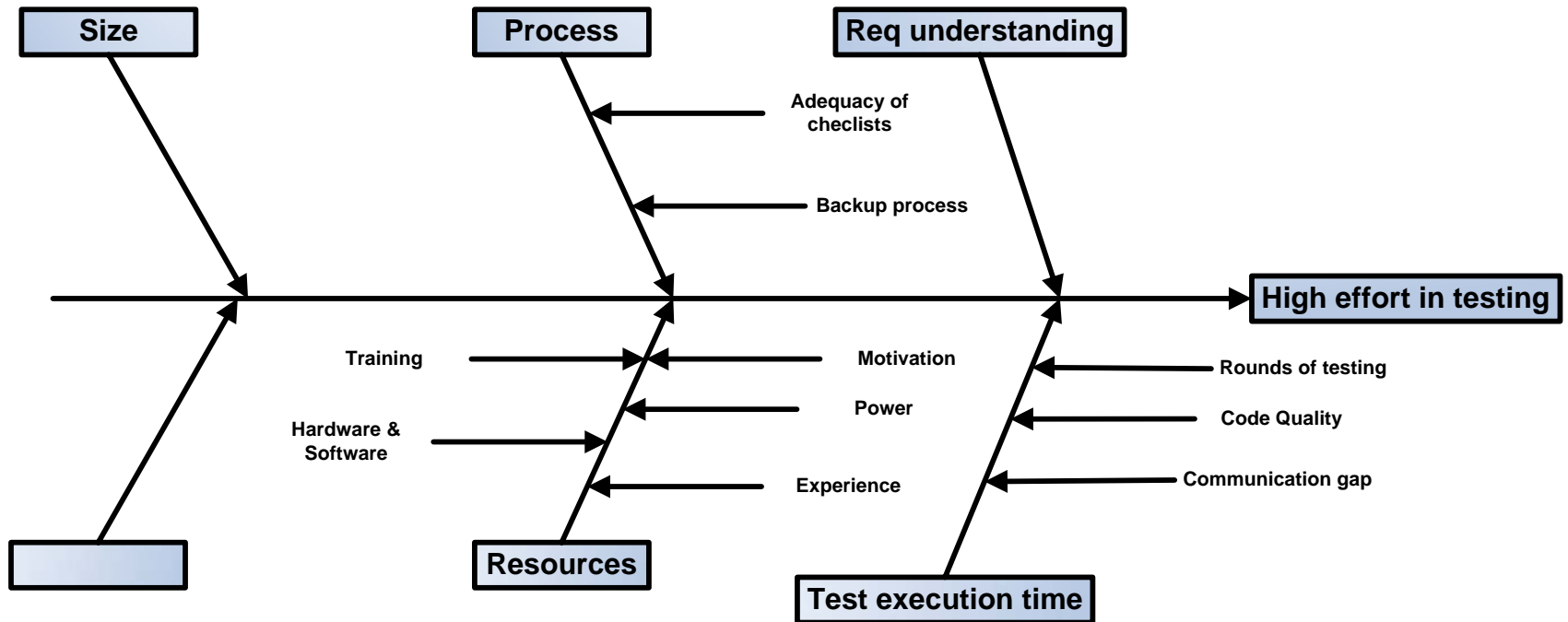
Root cause analyse

- Selecteer data voor analyse
- Vergaar data voor analyse
 - Bronnen van data kunnen zijn:
 - Fouten gerapporteerd door de gebruikers
 - Fouten gevonden gedurende inspecties
 - Fouten gevonden gedurende testen
 - Enz.
- Bepaal de root cause
- Definieer verbeter acties

Wanneer Causal Analysis?

- Na inspecties
- Na een test fase
- Na het passeren van een milestone
- Aan het eind van een iteratie
- Aan het eind van het project

Cause and Effect diagram



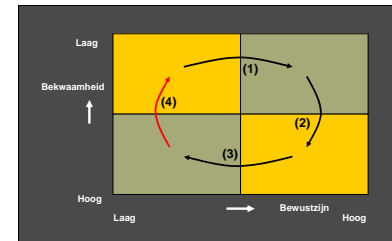
Kosten besparing?

Locatie	Huidig % van fouten gevonden per fase	Alle fouten gevonden in dezelfde fase als geïntroduceerd	Mogelijk besparingen
Requirements	9%	28%	9%
Ontwerp	7%	17%	12%
Bouw	10%	38%	31%
Module test	39%	10%	30%
Integratie test	12%	5%	9%
Systeem test	16%	2%	6%
Na release	7%	0%	3%
Gemiddeld aantal fouten	7,9	2,7	6,2
Percentage Reductie		65,8%	21,5%

Step 4: Heroriëntatiefase

- **Onbewust bekwaam => Onbewust onbekwaam**
- Na de eerste drie stappen ben je één niveau verder
- Maar

Er is altijd meer om te verbeteren



Conclusies

- We kunnen leren van onze fouten!
- Maar, zet niet alle fouten ineens
- Start met het verzamelen van de eerste foutdata
- Start met een eerste analyse van deze foutdata
- Start met het leren van je fouten op basis van de gedefinieerde stappen
- Een besparing tot 21,5% is mogelijk

Je hoeft geen mature organisatie te zijn om te leren van je fouten!!!

Dankuwei



Eric van der Vliet
Senior Performance Improvement Consultant
LogicaCMG Nederland B.V.
Divisie Industry, Distribution & Transport

E: eric.van.der.vliet@logicacmg.com

W: www.logicacmg.com