

Quality costs less

What are we going to do about it?

www.malotaux.eu/conferences

www.malotaux.eu/booklets

Niels Malotaux

niels@malotaux.eu

www.malotaux.eu

Niels Malotaux



- Independent Engineering and Team Coach
- Expert in helping projects and organizations to quickly become
 - More effective - doing the right things better
 - More efficient - doing the right things better in less time
 - More predictable - delivering as needed
- Project rescue: getting projects back on track
- Project types
electronics, firmware, software, space, road, rail, telecom, industrial control, parking system
- Embedded Systems creator (electronics/firmware)

Delivering
Quality On Time
the Right Result
at the Right Time

Quality costs less

- What is more costly ?
 1. First doing it wrong, then finding out, then doing it right ?
 2. Doing things right the first time ?
- Which of these is happening in your work environment ?
 - Respond with 1 or 2 in the chat 😊
- Shouldn't we better focus on doing things Right the First Time?
- Will you ?
- Can you ?
- What are we going to do about it ?

How did I find out myself ?

- When I was 9 years old, my father bought a new house
- I had to paint the wooden beams in my room myself
- Grinding, primer, grinding, color paint
- He came checking regularly
- If something wasn't the quality he expected, he explained, and I had to do things over again
- I didn't like doing things over again
- This taught me about Doing It Right the First Time



Needing money for travel to Japan

- Student in last year of Masters (1974)
- Job advertisement for Quality Engineer
- Called the recruiter: Forget it !
- Called Quality Dept head: Good to see what we do as QA
- Go through that recruiter 😊



‘Quality’ department

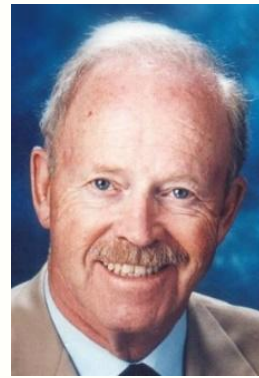
- They were testing product compliance, afterwards ...
 - Using AQL (‘Acceptable Quality Level’)
 - “100% quality would be too expensive”

What is AQL ?

(‘Acceptable Quality Level’)

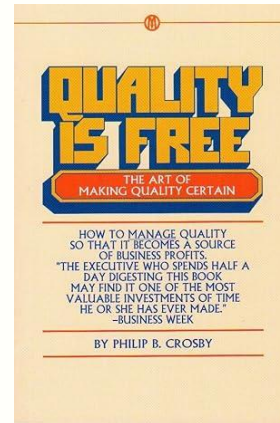
- Maximum acceptable number of defective items in a sample batch
“Economics of Quality”
- A company needed relays
- Decided to order 10,000 from Japan, AQL 2.5
- Japanese supplier: What is ‘AQL 2.5’ ?
- Take AQL table at 10,000, at AQL 2.5, find sample size Y, less than X defective
- Shipment arrived, with some relays in a separate bag
here are the defective ones ...

Inoculation



Crosby
(1926-2001)

- Found a 16-page brochure from ITT headquarters explaining quality management
- Confirmed my earlier experiences
- Phil Crosby in 'Quality is Free' (1978):
 - I sent a brochure to every unit
 - The results were amazing: No one paid the slightest bit of attention



Quality policy

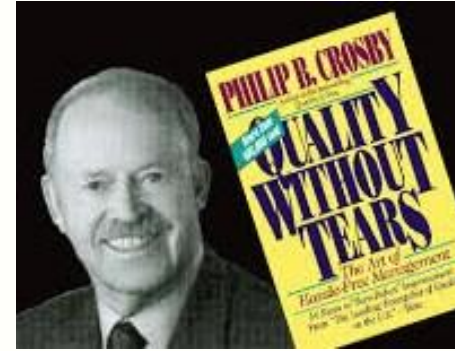
- Providing the customer with
 - what they need
 - at the time they need it
 - to be satisfied
 - to be more successful than they were without it
- Constrained by (win - win)
 - what the customer can afford
 - what we mutually beneficially and satisfactorily can deliver
 - in a reasonable period of time

Top-level requirement I'm using

- Doing what you said you would do
- More formally:
 - We will deliver products and services to our customers and co-workers
 - that meet the agreed requirements,
 - or we will have the requirements officially changed
 - to what we and they really need
- ITT every year saving more on the cost of non-quality

Absolutes of Quality Management

- **Conformance** to requirements
Not: goodness
- **Obtained through** prevention
Not: first doing it wrong and afterwards finding out
- **Performance standard is** zero defects
Not: that's close enough
- **Measured by the** price of non-conformance (PONC)
Not: how many defects
- **The purpose is** customer success
(~~not~~ customer satisfaction)
Added by Philip Crosby Associates, 2004



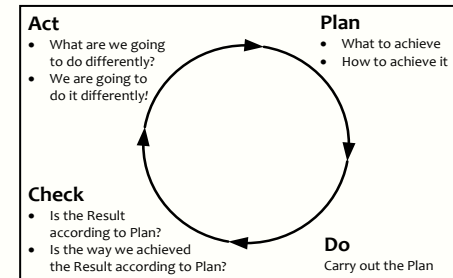
Conformance to requirements

- Management not allowing cutting corners anymore
- Requiring that *every individual* in the company makes sure they deliver
 - The Right Results
 - Efficiently
 - Preventing making errors, rather than repairing afterwards
 - Aware of the cost caused by errors, as an incentive of doing something about it
- Establishing what the Right Results are
 - Rather than everyone individually having a different view what is right
- Then we meet the agreed requirements *with care*
- Or, have the requirements *changed to what we and the customer really need*
- Management assisting people as needed
 - Perhaps hiring Tom Gilb for help

Quality on Time

Everyone in the organization

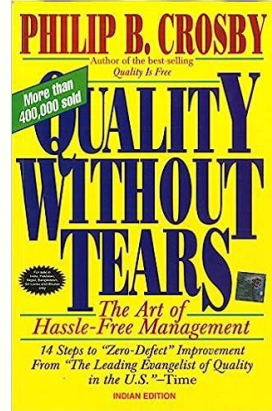
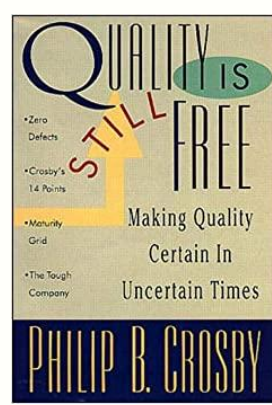
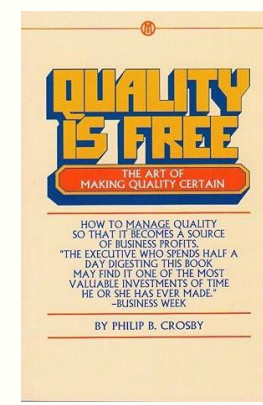
- Management, finance, sales, HR, production, developers ...
 - Identifying your customers
 - What do your customers need
 - Making sure to deliver that without error
 - Every time again
- Just common sense, right ?
- Prevention
 - Learning to prevent any errors that still happened



People make mistakes ?

If you think something is possible, or
you think it's impossible,
you're probably right (Henry Ford)

- Conventional wisdom says that error is inevitable
- As long as the performance standard requires it, this self-fulfilling prophecy will come true
- Most people will say: People are humans and humans make mistakes
- And people do make mistakes, particularly those who do not become upset when they happen
- Do people have a built-in defect ratio ?
- Mistakes are caused by two factors: lack of knowledge and *lack of attention*
- Lack of attention is an *attitude problem*



What is a defect ?



A defect is the cause of a problem
experienced by our customers
while relying on our results

How to feed Prevention

- Is Root Cause Analysis routinely performed - every time ?
not just cause analysis
- What is the Root Cause of a defect ?
- Cause:
The error that caused the defect
- Root Cause:
What caused **US** to make the error that caused the defect
- Without proper Root Cause Analysis, we're *doomed to repeat the same errors*

cause

solution

root cause

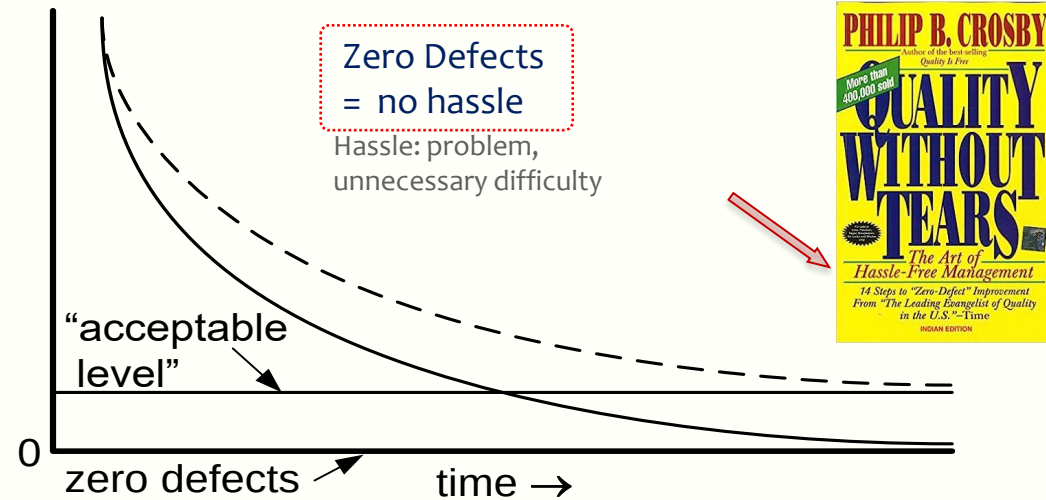
root solution

What is Zero Defects ?

- Zero Defects is an *asymptote*



Niels Malotau:
In my experience the
'zero defects' attitude
results in 50% less defects
almost overnight



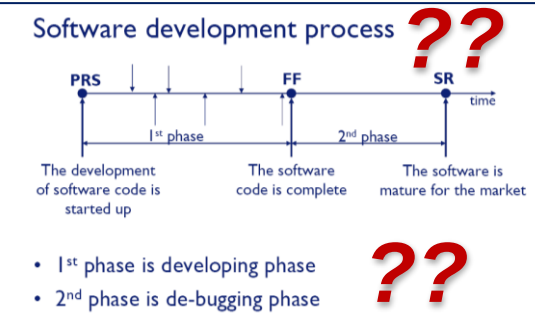
- When Philip Crosby started with Zero Defects in 1961, errors dropped by 40% almost immediately
- AQL > Zero means that the organization has settled on a level of *incompetence*
- Causing a *hassle* other people have to live with

Is Zero Defects software possible ?

- How many defects are acceptable ?
- Requirements specify a certain number of defects ?
- Do you check the required number has been produced ?

In your work

- How much time is spent putting defects in ?
- How much time is spent trying to find and fix them ?
- Do you sometimes see repeated issues ?
- How much time is spent on defect prevention ?



Requirement:

- Deliver 7 bugs in every sprint

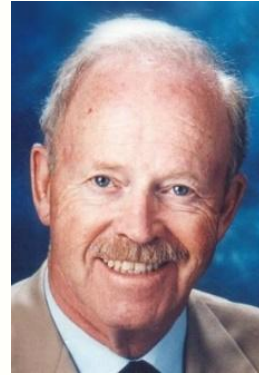
User Story:

- As a customer,
I like to experience some bugs,
so that I regularly can complain

Better quality costs less

If we do everything Right the First Time

- Crosby suggested to select suppliers based on no need for testing
 - Ditch suppliers who deliver defects
 - If something goes out, stop and fix it - It will **all** run smoothly soon (yield went up from 16% to 85% in one month)
- Deming said similar to Ford
 - Quality isn't a department - It's a decision
 - Quality inspectors are a sign of failure, not success
 - Quality is built in, NOT inspected in
 - You cannot test quality into a product
 - Hence: The best quality system is one that needs no inspection
 - Ford became profitable again



Crosby
(1926-2001)

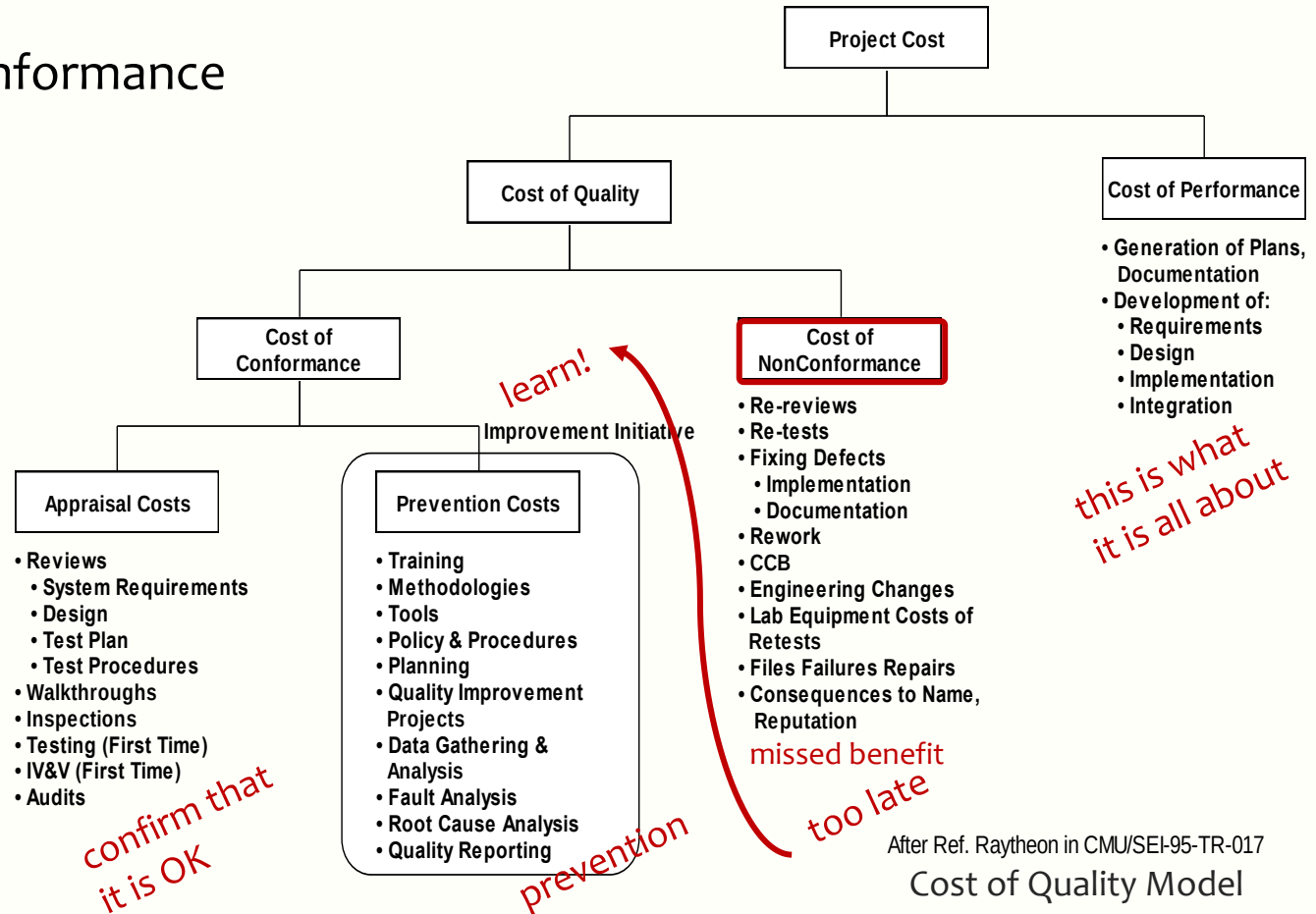


Deming
(1900-1993)

But we are agile !

- **Agility**
Quickly and appropriately responding to whatever is, and whatever happens
- **Can be used anytime, anywhere, with any approach**
so-called 'Agile Frameworks' are not automatically agile
- **Quickly**
indicates frequent feedback
- **Appropriately**
is the challenge

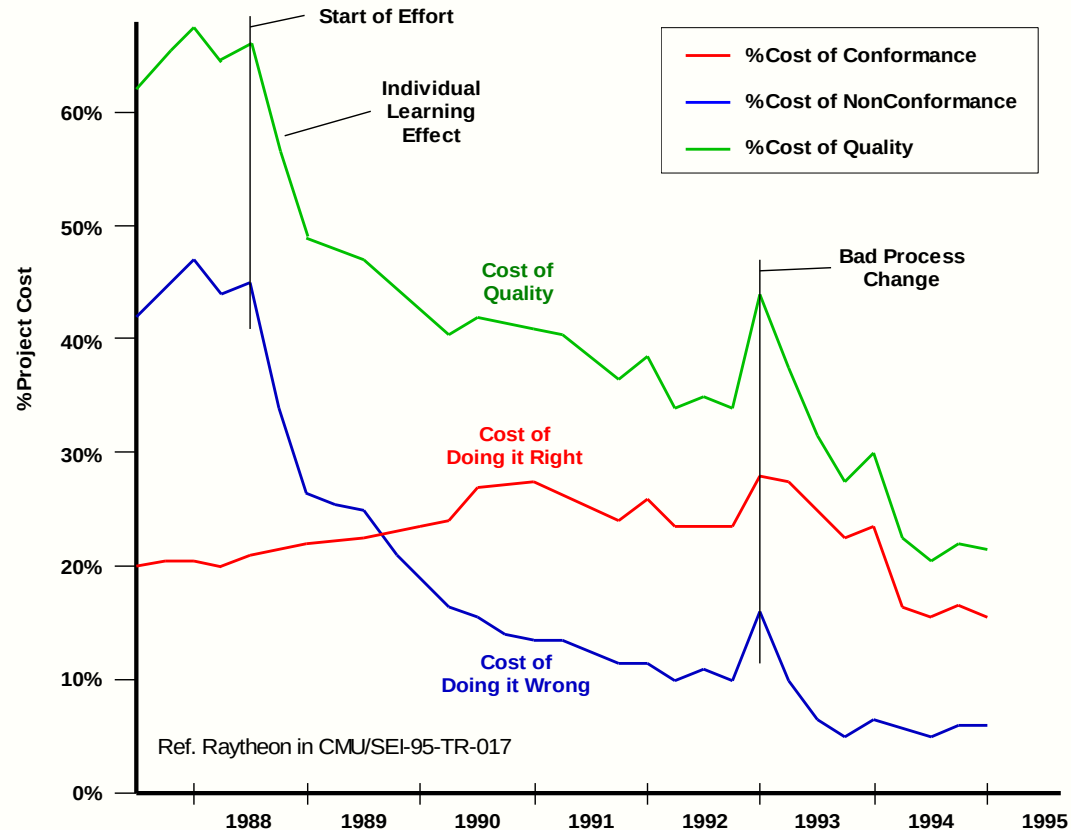
Price of non-conformance



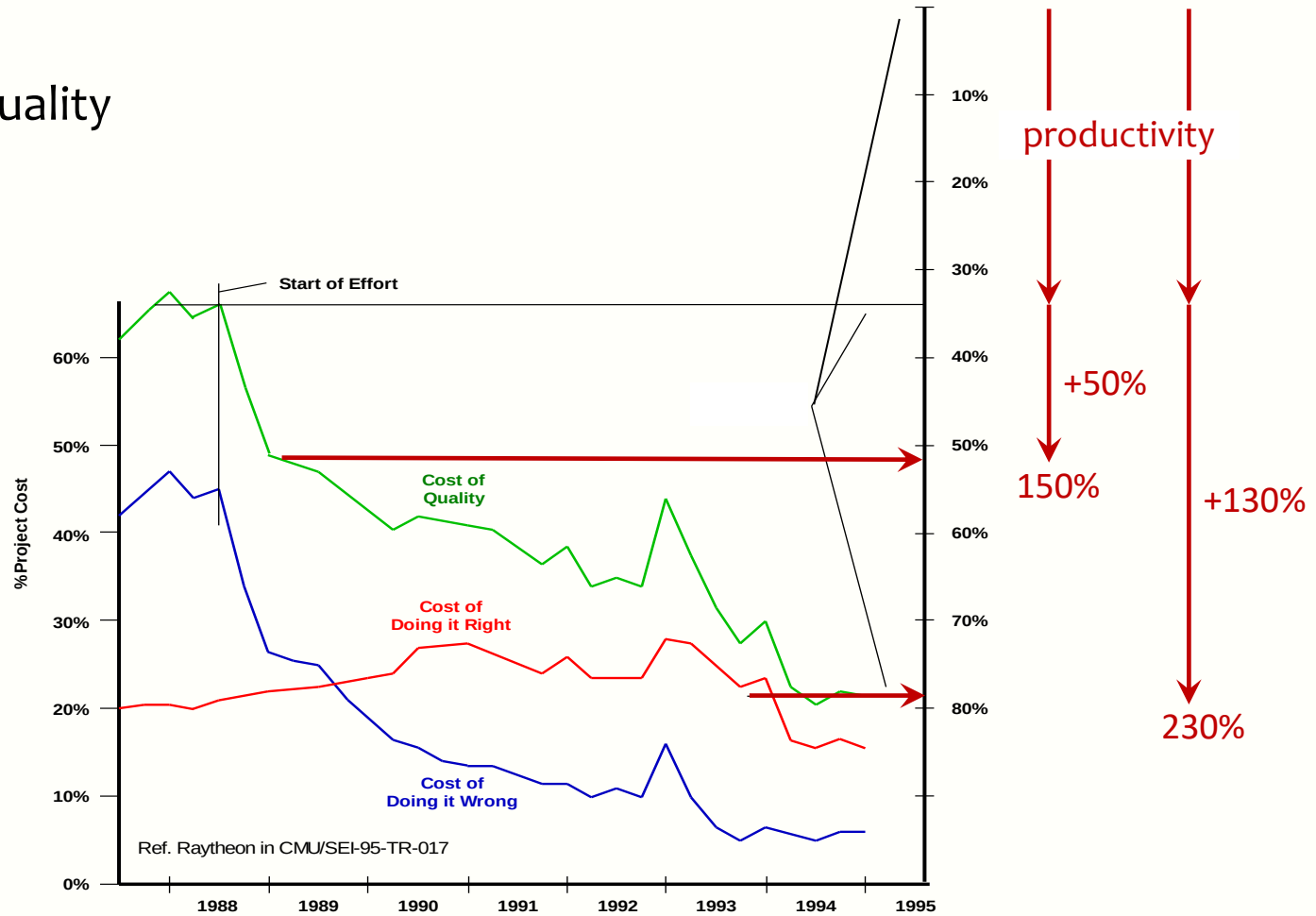
After Ref. Raytheon in CMU/SEI-95-TR-017

Cost of Quality Model

Cost of Quality



Benefit of Quality



Developing electronic products

1980 - 1998

- Many electronic products developed
- Getting into production: 'simply worked'
- Couldn't afford getting bothered by problems, when working for the next client
- Customers didn't need their 'panic budget'
(the high costs caused by repairs in and after production)
- Zero Defects at work !

Jules May: Extreme Reliability

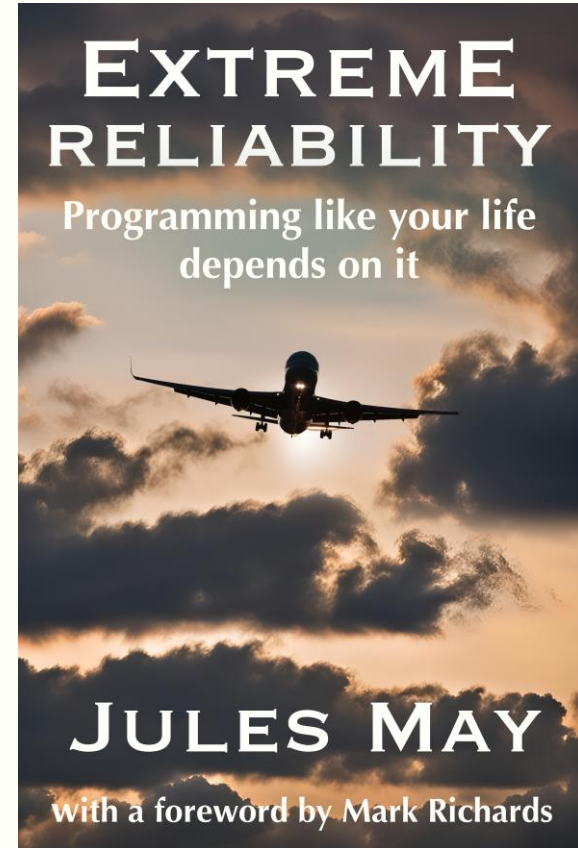
All my experience has convinced me
that it takes less time to code it right first time
than knowingly doing it wrong

It's not just a long-term benefit:
it's quicker in the short term as well

This book is for developers and technical leads who want to
eliminate not just 95% of their bugs, but all of them

For developers who want to deliver flawless software, faster

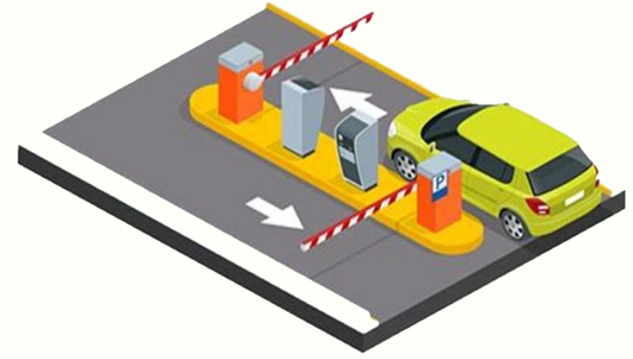
*Now we have Ai !
does Ai produce flawless software ???*



New QA manager at a large bank

No defects in the first two weeks of use

It simply works ...



- What we deliver simply works
- Does that mean without bugs ??? said the project manager and architect in chorus
- It simply works - do I speak in a foreign language ?

Handheld device crashing all the time

- **First requirement: get it stable**
 - Stop adding features - get it to the bare minimum - get it stable
 - Any other requirements are irrelevant if the product crashes all the time
- **Two weeks to understand I was dead serious about it**
- **Third week testers coming to me**
 - Niels, something weird is going on. It simply works ! We don't find bugs anymore !
 - Now they could add more functionality, which (surprise!) simply worked.
- **Developers quite capable to deliver zero-defect software, only by challenging them**
(shouldn't you try that out ?)
- **Challenge for testers: How to prove there are no bugs, if there aren't any**
- **If they still find anything, feeding the prevention process**

Quality for products: How well ...

3 or 7 ?

- | | | | | |
|--------------------|--------------------|----------------------|-----------------------|------------------------|
| • accessibility | • debuggability | • installability | • capabilities | • serviceability |
| • accountability | • degradability | • integrity | • proactivity | • securability |
| • accuracy | • determinability | • interactivity | • producibility | • simplicity |
| • adaptability | • demonstrability | • interchangeability | • provability | • stability |
| • administrability | • dependability | • interoperability | • recoverability | • standards compliance |
| • affordability | • deployability | • intuitiveness | • redundancy | • survivability |
| • agility | • discoverability | • learnability | • relevance | • sustainability |
| • analyzability | • distributability | • localizability | • reliability | • tailorability |
| • auditability | • durability | • maintainability | • repairability | • testability |
| • autonomy | • effectiveness | • manageability | • repeatability | • timeliness |
| • availability | • efficiency | • mobility | • reproducibility | • traceability |
| • compatibility | • elasticity | • modifiability | • resilience | • transparency |
| • composability | • evolvability | • modularity | • responsiveness | • ubiquity |
| • confidentiality | • extensibility | • observability | • reusability | • understandability |
| • configurability | • familiarity | • operability | • robustness | • upgradability |
| • convenience | • fault-tolerance | • orthogonality | • safety | • usability |
| • correctness | • fidelity | • portability | • scalability | • vulnerability |
| • credibility | • flexibility | • precision | • seamlessness | |
| • customizability | • inspectability | • predictability | • self-sustainability | |

Quality

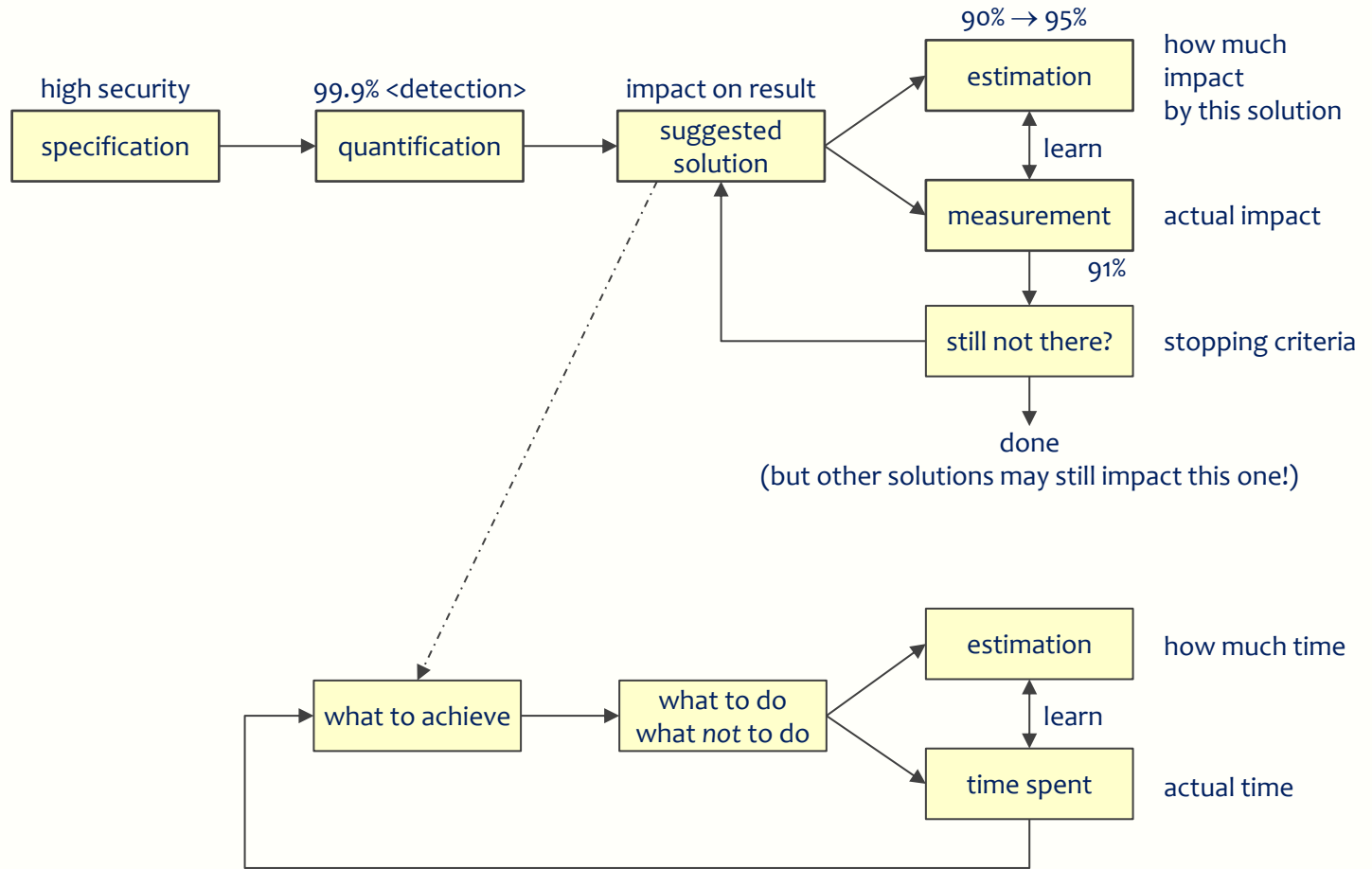
organising the

- what
- why
- for whom
- how much

on Time

organising the

- how
- when



Weekly TaskCycle plan

- What can we learn from last week (retrospecting)
- What should we **prevent** the coming week (prespecting)
- What are we supposed to achieve
- How much time do we have available
- 2/3 of gross available time is net plannable time
- What is most important to do
in order to achieve what we're supposed to achieve
- Estimate net effort needed to do these things
- Which most important things fit the net plannable time
(default 2/3 of gross available time, 26hr per week at 40hr work-week)
- What can, and what are we going to do
- What are we **not** going to do
- Writing it down ! Our fuzzy mind isn't good enough !

2/3 is default start value - works well with development work

Evolutionary Project Management elements (Evo) - Tom Gibb
www.malotau.eu/?id=processes

What How much Are we doing

Check and learn as early as possible

Zero Defects Attitude

Weekly TaskCycle

- Short term planning
- Optimizing estimation
- Promising what we can achieve
- Living up to our promises

Efficiency of what we do

Bi-weekly DeliveryCycle

- Optimizing the requirements and checking the assumptions
- Soliciting feedback by delivering real results to eagerly waiting Stakeholders

Effectiveness of what we do

TimeLine

- Getting and keeping control of Time: Predicting the future
- Feeding program/portfolio/resource management

What will happen, and what will we do about it?

Malotau - Quality on Time

www.malotau.eu/processes

148

Task _a	2	↑ do
Task _b	5	
Task _c	3	
Task _d	6	
Task _e	1	
Task _f	4	↓ do not
Task _g	5	
Task _h	4	
Task _i	3	
Task _k	1	

26

Prespecting with 'innocent' questions

Task _a	2
Task _b	5
Task _c	3
Task _d	6
Task _e	1
Task _f	4
Task _g	5
Task _h	4
Task _i	3
Task _k	1

People come in with their week plan,
come out with a modified plan,
more going to work on the right things,
less spending time on unnecessary things.
Immediate savings.
From day one.

- Really ?
- Should we ?
- Why would we do that ?
(never use 'you')
- Who's waiting for that ?
- What do they need ?
- How much do they need ?
- When do they need it ?
- Is it really necessary ?
- Is it really necessary now ?
- How do we know ?

- Why ?
- What happened ?
- What could we do ?
- Will we be on time ?
- What makes us think that ?
- What's different this time ?
- Is this the best way to do it ?
- Does it fit the available time ?
- *If there is any problem:*
What-are-we-going-to-do-about-it ?

Never challenge an estimate !
If we don't understand the estimate,
only say:

- What are you planning to do ?
- Now estimate again ...

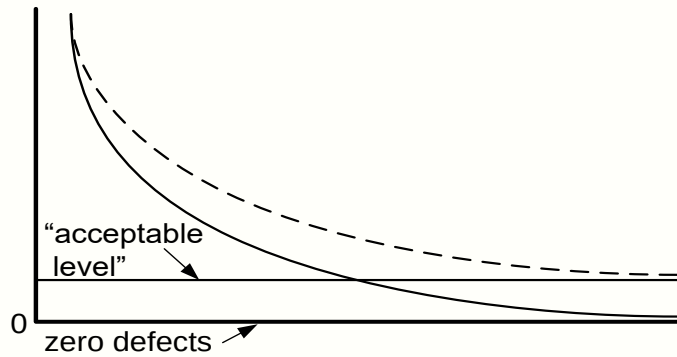
If they insist:

- Let them find out themselves

What are we going to do about it ?

- Who is our customer ?
- What do they need ?
- When do they need it ?
- Will they be happy with it ?
- Will they be more successful ?
- Can the customer afford it ?
- Is it win-win ?

- Providing the customer with
 - what they need
 - at the time they need it
 - to be satisfied
 - to be more successful than they were without it
- Constrained by (win - win)
 - what the customer can afford
 - what we mutually beneficially and satisfactorily can deliver
 - in a reasonable period of time



Approaching Zero Defects is Absolutely Possible

If in doubt, let's talk about it

Niels Malotaux

niels@malotaux.eu

www.malotaux.eu

Quality costs less

What are we going to do about it?

www.malotaux.eu/conferences

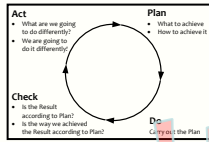
www.malotaux.eu/booklets

Niels Malotaux

niels@malotaux.eu

www.malotaux.eu

- **Plan-Do-Check-Act**
 - The powerful ingredient for success
- **Business Case**
 - Why we are going to improve what



Evolutionary Project Management elements (Evo)

Tom Gilb

- **Requirements Engineering**
 - What we are going to improve and what not
 - How much we will improve: quantification
- **Architecture and Design**
 - Selecting the optimum compromise for the conflicting requirements
- **Early Review & Inspection**
 - Measuring quality while doing, learning to prevent doing the wrong things

What
How much
Are we done

How

Check and learn
as early as possible

Zero
Defects
Attitude

- **Weekly TaskCycle**
 - Short term planning
 - Optimizing estimation
 - Promising what we can achieve
 - Living up to our promises
- **Bi-weekly DeliveryCycle**
 - Optimizing the requirements and checking the assumptions
 - Soliciting feedback by delivering Real Results to eagerly waiting Stakeholders
- **TimeLine**
 - Getting and keeping control of Time: Predicting the future
 - Feeding program/portfolio/resource management

Efficiency
of what we do

Evo Project Execution

Niels

Effectiveness
of what we do

What will happen, and
what will we do about it?

- 1 Evolutionary Project Management Methods (2001)
Issues to solve, and first experience with the Evo Planning approach
 - 2 How Quality is Assured by Evolutionary Methods (2004)
After a lot more experience: rather mature Evo Planning process
 - 3 Optimizing the Contribution of Testing to Project Success (2005)
How Testing fits in
 - 3a Optimizing Quality Assurance for Better Results (2005)
Same as Booklet 3, but for non-software projects
 - 4 Controlling Project Risk by Design (2006)
How the Evo approach solves Risk by Design (by process)
 - 5 TimeLine: How to Get and Keep Control over Longer Periods of Time (2007)
Replaced by Booklet 7, except for the step-by-step TimeLine procedure
 - 6 Human Behavior in Projects (APCOSE 2008)
Human Behavioral aspects of Projects
 - 7 How to Achieve the Most Important Requirement (2008)
Planning of longer periods of time, what to do if you don't have enough time
 - 8 Help ! We have a QA Problem ! (2009)
Use of TimeLine technique: How we solved a 6 month backlog in 9 weeks
 - 9 Predictable Projects (2012) - How to deliver the Right Results at the Right Time
 - 9a 予測可能なプロジェクト - 適切なタイミングで適切な結果を出す方法 - same, in Japanese
- RS Measurable Value with Agile (Ryan Shriver - 2009)
Use of Evo Requirements and Prioritizing principles

