Niels Malotaux

Getting and Keeping Projects under Control

London, 19 February 2009

N R Malotaux - Consultancy
The Netherlands
+31-30-2288868
+31-30-2288869
niels@malotaux.nl
www.malotaux.nl

Niels Malotaux

Niels Malotaux is an independent Project Coach specializing in optimizing project performance. He has some 35 years experience in designing electronic hardware and software systems, at Delft University, in the Dutch Army, at Philips Electronics and 20 years leading his own systems design company. Since 1998 he devotes his expertise to helping projects to deliver Quality On Time: delivering what the customer needs, when he needs it, to enable customer success. To this effect, Niels developed an approach for effectively teaching Evolutionary Project Management (Evo) Methods, Requirements Engineering, and Review and Inspection techniques. Since 2001, he taught and coached some 100 projects in 25+ organizations in the Netherlands, Belgium, China, Germany, Ireland, India, Israel, Japan, Romania, South Africa and the US, which led to a wealth of experience in which approaches work better and which work less in practice.

Niels puts development teams on the Quality On Time track and coaches them to stay there and deliver their quality software or systems on time, without overtime, without the need for excuses. Practical methods are developed, used, taught and continually optimized for:

- Evolutionary Project Management (Evo)
- Requirements Engineering and Management
- Reviews and Inspections.

Within a few weeks of turning a development project into an Evo project, the team has control and can tell the customer when the required features will all be done, or which features will be done at a certain date. Niels enjoys greatly the moments of enlightenment experienced by his clients when they find out that they can do it, that they are really in control, for the first time in their lives.

N R Malotaux
Consultancy

Niels Malotaux
project coach

Result Management

Bongerdlaan 53
3723 VB Bilthoven
The Netherlands
tel +31-30-228 88 68
fax +31-30-228 88 69
mob +31-6-5575 3604
niels@malotaux.nl
www.malotaux.nl

Getting and Keeping Projects Under Control

Niels Malotaux

N R Malotaux Consultancy

+31-30-228 88 68 niels@malotaux.nl www.malotaux.nl

1

Niels Malotaux

Project Coach

- Evolutionary Project Management (Evo)
- Requirements Engineering
- Reviews and Inspections

Result Management

- Researching problems in projects
- Finding ways to fundamentally overcoming these problems
- Ploughing back into projects
- Tuning of the results (because theory isn't practice)

2

Universal Project Goal

- Providing the customer with
  - what he needs
  - at the time he needs it
  - to be satisfied
  - to be more successful than he was 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

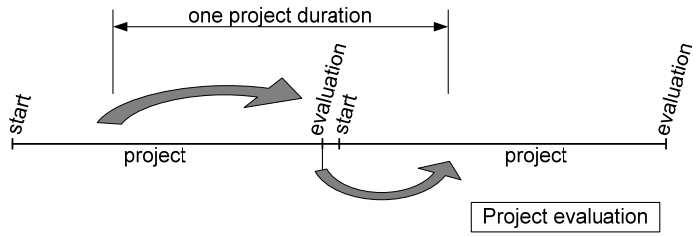
3

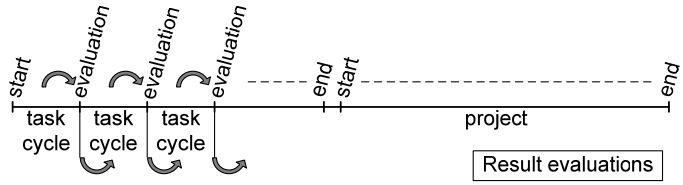
**If your previous project was late,
your current project will probably be late as well**

**If we don't learn from history,
we are doomed to repeat it**

4

Project evaluations





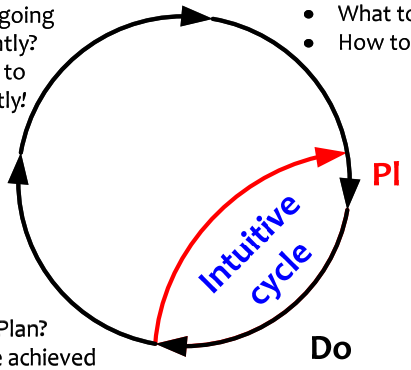
5

The essential ingredient: the PDCA cycle

(Deming cycle)

Act
- What are we going to do differently?
- We are going to do it differently!

Plan
- What to achieve
- How to achieve it

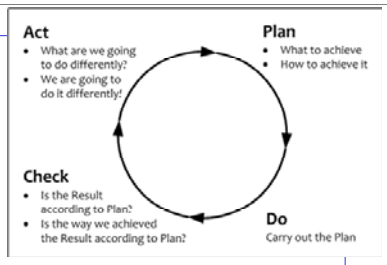


Check
- Is the Result according to Plan?
- Is the way we achieved the Result according to Plan?

Do
Carry out the Plan

6

Evo



- Evo (short for Evolutionary...) uses PDCA consistently
- Applying the PDCA-cycle
  actively, deliberately, rapidly and frequently,
  for *Product*, *Project* and *Process*, based on ROI and highest value
- Combining Planning, Requirements- and Risk-Management into *Result Management*
- We know we are not perfect,
  but the customer should never find out
- Evo is about delivering Real Stuff to Real Stakeholders doing Real Things

"Nothing beats the Real Thing"

7

Evolutionary Project Management (Evo)

- Plan-Do-Check-Act
  - The powerful ingredient for success
- Business Case
  - *Why* we are going to improve *what*
- Requirements Engineering
  - *What* we are going to improve *and what not*
  - *How much* we will improve: quantification
- Architecture and Design
  - Selecting the optimum compromi[illegible]
- Early Review & Inspection
  - Measuring quality while doing, learning to prevent doing the wrong things

Right product

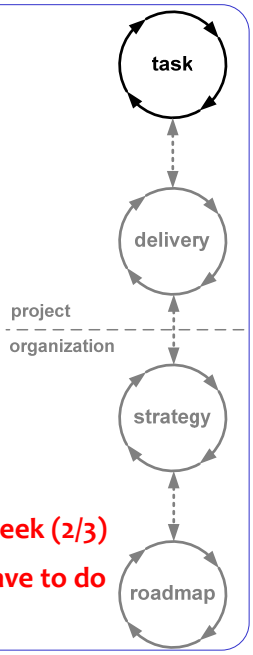
Evo planning

- Weekly TaskCycle
  - Short term planning
  - Optimizing estimation
  - Promising what you can achieve
  - Living up to your promises
- Bi-weekly DeliveryCycle
  - Optimizing the requirements and checking the assumptions
  - Soliciting feedback by delivering Real Results to e[illegible]
- TimeLine
  - Getting and keeping control of Time
  - Feeding program/portfolio/resource management

Right time

8

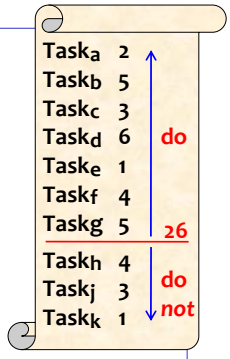
Weekly TaskCycle



- Are we *doing*
  the *right things*,
  in the *right order*,
  to the right *level of detail for now*
- Optimizing estimation, planning and tracking abilities to better predict the future
- Select highest priority tasks, never do any lower priority tasks, never do undefined tasks
- There are only about 26 plannable hours in a week (2/3)
- In the remaining time: do whatever else you have to do
- Tasks are always done, 100% done

9

Every week we plan



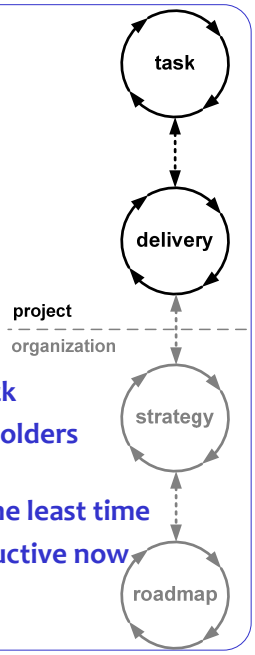
- How much time do we have available
- 2/3 of available time is net plannable time
- What is most important to do
- Estimate effort needed to do these things
- Which most important things fit in the net available time (default 26 hr per week)
- What can, and are we going to do
- What are we *not* going to do

2/3 is default start value
this value works well in development projects

10

DeliveryCycle

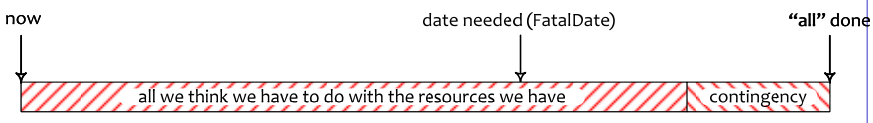
- Are we delivering
  the *right things*,
  in the *right order*
  to the right *level of detail for now*
- Optimizing requirements
  and checking assumptions
  a. What will generate the optimum feedback
  b. We deliver only to *eagerly waiting* stakeholders
  c. Delivering the juiciest, most important
     stakeholder values that can be made in the least time
     - What will make Stakeholders more productive now
- Not more than 2 weeks



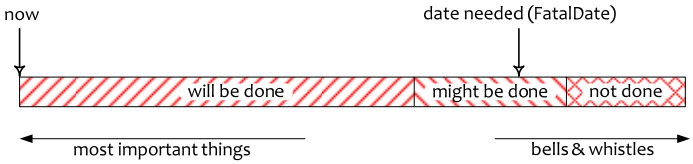
11

TimeLine

What the customer wants, he cannot afford



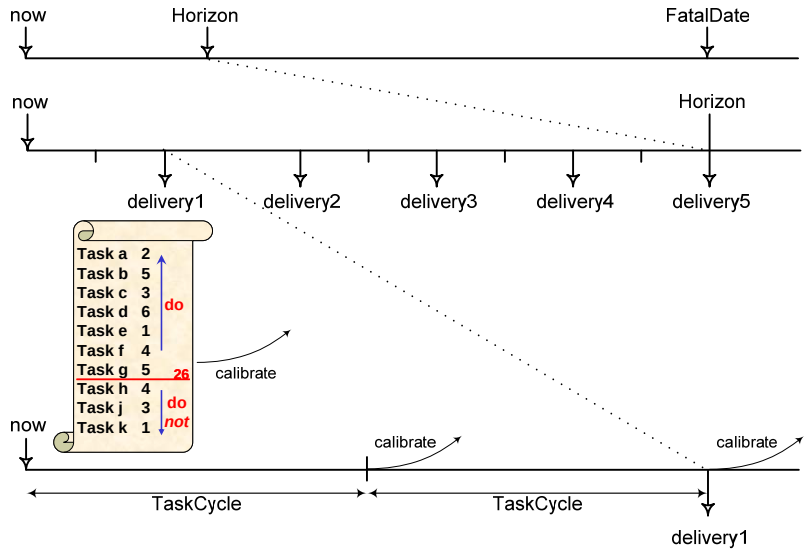
Standard Projects



Evo

12

Result to Tasks and back



13

Calibration

Activity	Estimate	Real
Act1	Ae1	Ar1
Act2	Ae2	Ar2
Act3	Ae3	Ar3
Act4	Ae4	Ar4
Act5	Ae5	Ar5
Act6	Ae6	Ar6
Act7	Ae7	Ar7
Act8	Ae8	Ar8
Act9	Ae9	Ar9
Act10	Ae10	Ar10
Act11	Ae11	
Act12	Ae12	
Act13	Ae13	
Act14	Ae14	
Act15	Ae15	
Act16	Ae16	
Act17	Ae17	
Act18	Ae18	
Act19	Ae19	
Act20	Ae20	
Act21	Ae21	
Act...	Ae...	

average ratio Ar/Ae in the past

now

predicted Value Still To Earn in the future

then

then2

Calibration Factor

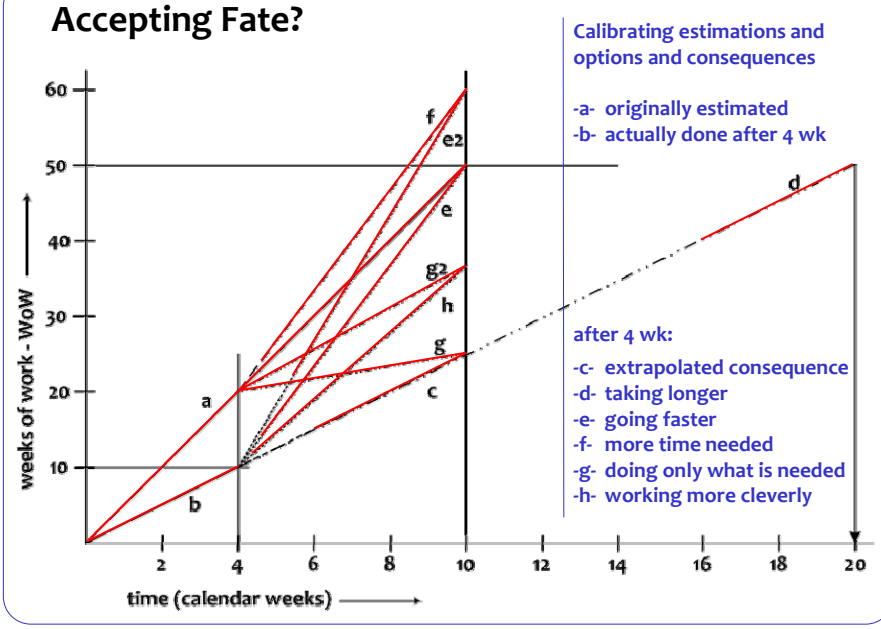
$$\frac{1}{n}\sum_{now-1}^{now-n}\frac{Ar}{Ae}$$

Value Still To Earn

$$Calibration\ Factor * \sum_{now}^{then} Ae$$

14

Accepting Fate?



15

Estimation techniques used

- **Just-enough estimation** (don't do unnecessary things)
  - Maximizing Return-on-Investment and Value Delivered
- **Changing from optimistic to realistic predictions**
  - Estimation of Tasks in the TaskCycle
  - Prediction what will be done when in TimeLine
- **0th order estimations** (ball-park figures)
  - For decision-making in Business Case and Design
- **Simple Delphi**
  - For estimating longer periods of time in TimeLine
  - For duration of several (15 or more) elements of work
- **Simpler Delphi**
  - Same, but for quicker insight
  - Recently added by practice
- **Calibration**
  - Coarse metrics provide accurate predictions
- **Doing something about it** (if we don't like what we see)
  - Taking the consequence
  - Saving time

16

Simple and Simpler Delphi

1. List things to do
2. Distribute the list
3. Add and estimate
4. List estimates
5. Discuss if differences
6. Estimate again
7. Repeat until consensus
8. Add up all the estimates

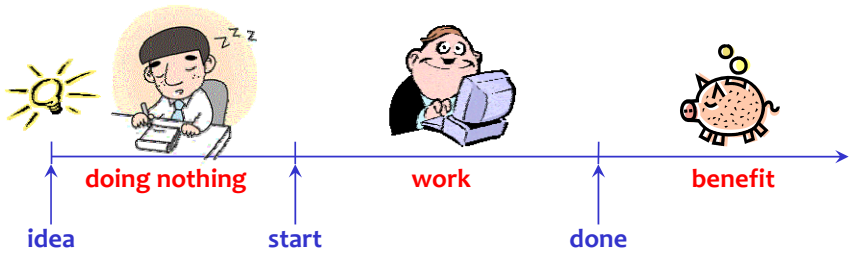
1. List things to do
2. Distribute the list
3. Add and estimate

4. List estimates: min and max
5. Discuss if differences
6. Agree on value between min and max
7. Add up all the estimates

**Even with coarse estimates per element of work,
the sum averages out the variations and can be quite predictive**

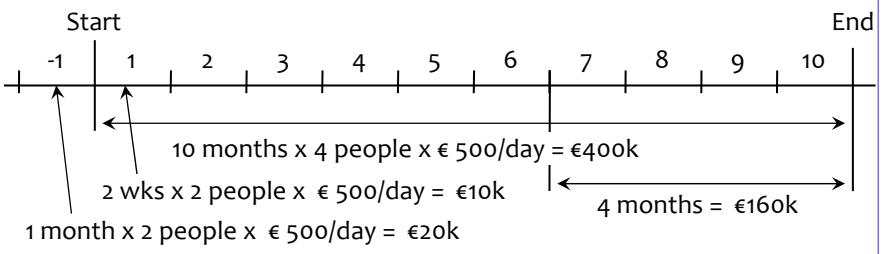
17

What's the Cost of Doing Nothing ?



18

The Cost of Time

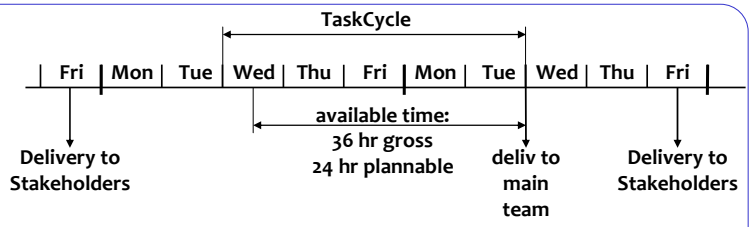


- We can save 4 months by investing €200k → "That's too much !"
- It's a *nicer* solution - Let's do 2 weeks more research on the benefits
- What are the expected revenues when all is done? → €16M/yr (1.3M/mnd)
- So 2 weeks extra doesn't cost €10k, but rather €16M/24 = €670k
- And saving 4 months brings €16M/3 = €5M extra

➔ Invest that €200k *NOW and don't waste time* !

19

Designing a Delivery



Serge (ProjLead)	
MbWA	3
Planning nxt wk	3
Work for deliv	4
-	6
-	2
-	1
-	5
Total	24

Gregory	
Draft design	6
Finish design	6
Work for deliv	3
-	1
-	2
-	2
-	3
-	5
-	6
XMLa	4
XMLb	4
Total	42

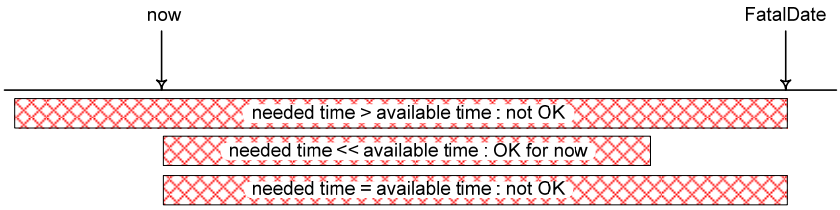
Gregory (later)	
→ Draft design	0
→ Finish design	0
...	
Repair deliv	0
...	

Zero Defects Attitude

Jerome	
→ XMLa	3
→ XMLb	3
...	

20

If it doesn't fit ... count backwards



21

Deceptive options

- Hoping for the best (fatalistic)
- Going for it (macho)
- Working Overtime (fooling ourself)
- Moving the deadline
  - Parkinson's Law
    - Work expands to fill the time for its completion
  - Student Syndrome
    - Starting as late as possible, only when the pressure of the FatalDate is really felt

22

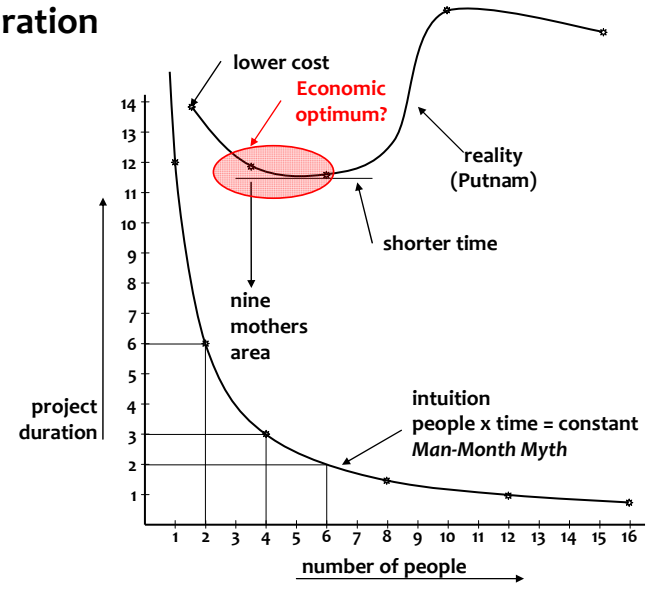
Adding people to a late project ...

makes it later

(Brooks' Law, 1975)

23

Project-duration



24

Saving time

We don't have enough time, but we can save time *without negatively affecting the Result !*

- Efficiency in *what* (*why*, for *whom*) we do - doing the right things
  - *Not* doing what later proves to be superfluous
- Efficiency in *how* we do it - doing things differently
  - The product
    - Using proper and most efficient solution, in stead of the solution we always used
  - The project
    - Doing the same in less time in stead of immediately doing it the way we always did
  - Continuous improvement and prevention processes
    - Constantly learning doing things better and overcoming bad tendencies
- Efficiency in *when* we do it - doing things in the right order, at the right time
- TimeBoxing - much more efficient than FeatureBoxing

25

Evolutionary Project Management (Evo)

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

Zero Defects Attitude

- Weekly TaskCycle
  - Short term planning
  - Optimizing estimation
  - Promising what you can achieve
  - Living up to your 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
  - Feeding program/portfolio/resource management

Evo planning

26

Getting and Keeping Projects Under Control

Niels Malotaux

N R Malotaux Consultancy

+31-30-228 88 68 niels@malotaux.nl www.malotaux.nl

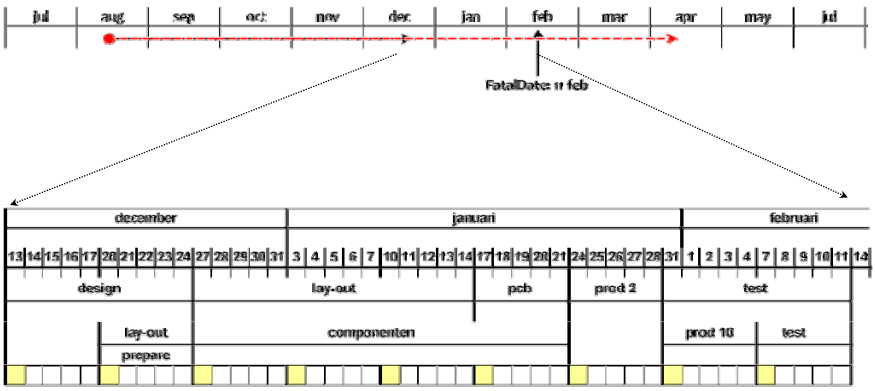
27

Links

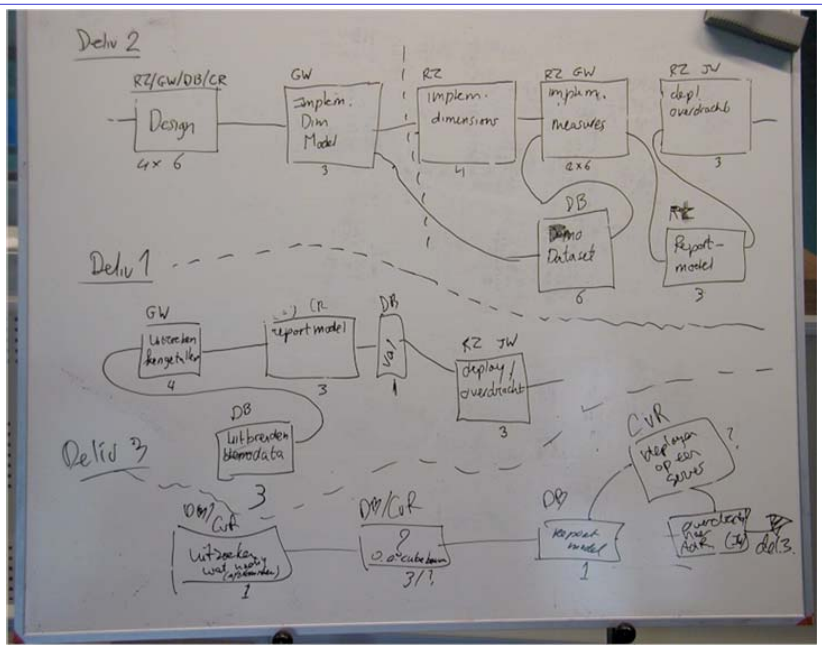
- **www.gilb.com**
  Tom Gilb's website: Evo guru
- **www.malotaux.nl**
  Niels' activities: Evo evangelist
- **www.malotaux.nl/nrm/Evo**
  Evo pages
- **www.malotaux.nl/nrm/Insp**
  Inspection pages
- **www.malotaux.nl/nrm/pdf/MxEvo.pdf**
  Evolutionary Project Management Methods (2001 - issues and first experience)
- **www.malotaux.nl/nrm/pdf/Booklet2.pdf**
  How Quality is Assured by Evolutionary Methods
  (2004 - after some 25 projects in 9 organizations - practical implementation experience)
- **www.malotaux.nl/nrm/pdf/EvoTesting.pdf**
  Optimizing the Contribution of Testing to Project Success (2005)
- **www.malotaux.nl/nrm/pdf/EvoRisk.pdf**
  Controlling Project Risk by Design (2006)
- **www.malotaux.nl/nrm/pdf/TimeLine.pdf**
  TimeLine: How to get and keep control over longer periods of time (2007)
- **www.malotaux.nl/nrm/pdf/HumanBehavior.pdf**
  Recognizing and Understanding Human Behavior to Improve Systems Engineering Results (2008)
- **www.malotaux.nl/nrm/pdf/TimeLineIS09.pdf**
  How to Achieve the Most Important Requirement (2008)
- **ww.malotaux.nl/nrm/pdf/InspManual.pdf**
  16 page mini Inspection Manual
- **www.malotaux.nl/nrm/Evo/ETAF.htm**
  Download the Evo Task Administrator (ETA) tool (expects MSAccess 2000~2003)

28

TimeLine planning



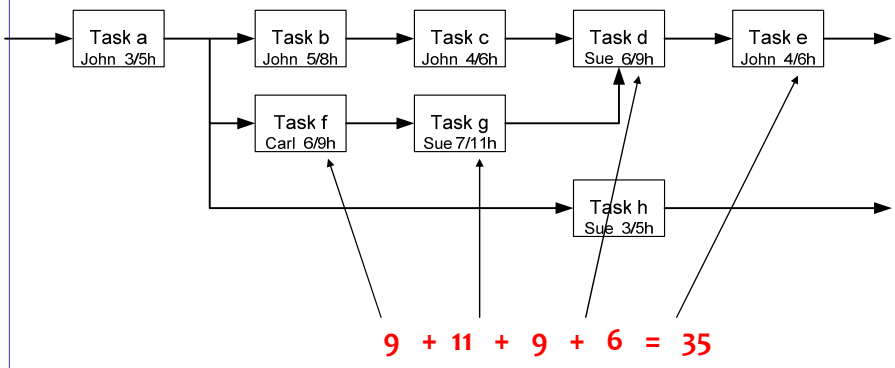
29



30

PERT (Project Evaluation Review Technique)

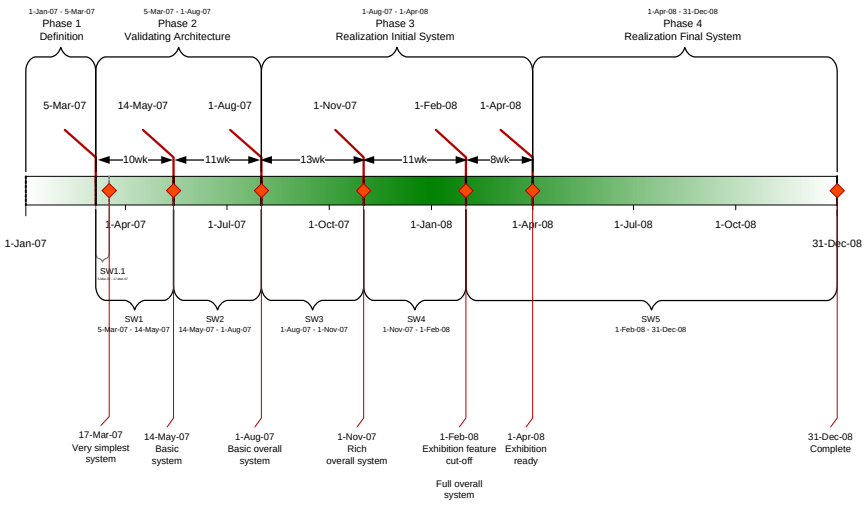
used for *Designing* a Delivery



9 + 11 + 9 + 6 = 35

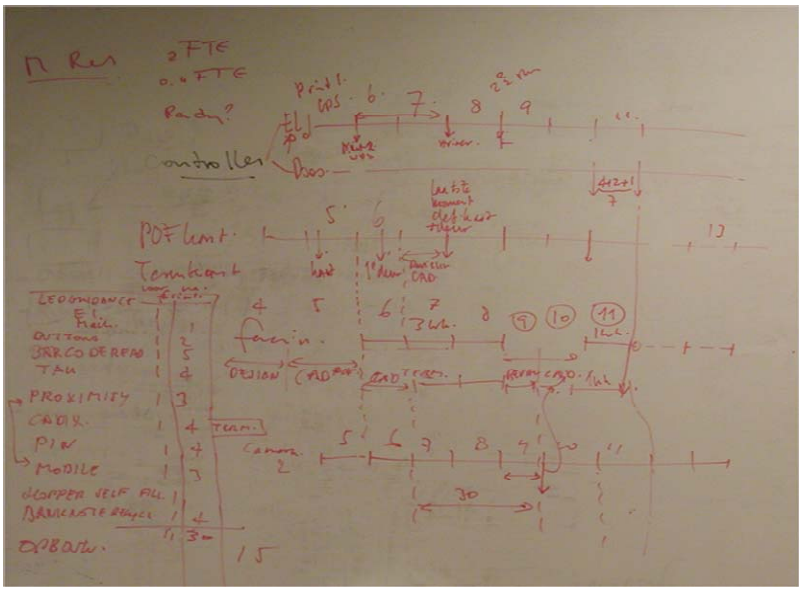
31

TimeLine example



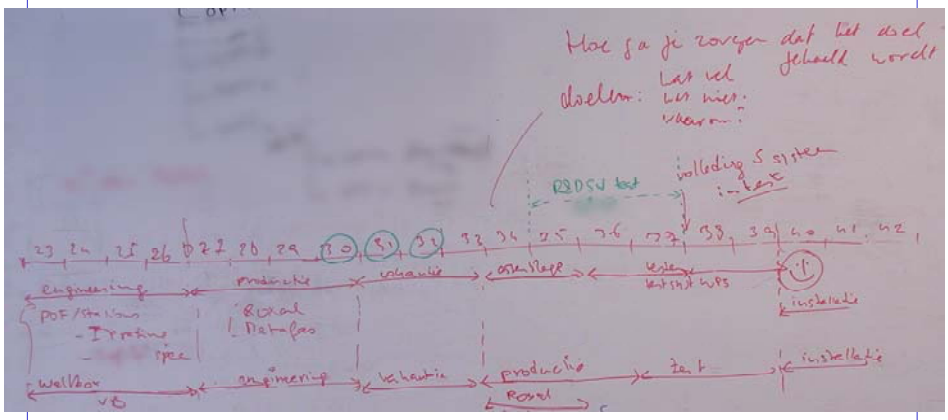
32

Whiteboard TimeLine Planning



33

Whiteboard TimeLine Planning



34

Design is always a compromise

- Design is the process of collecting and selecting options how to implement the requirements
- The Requirements are *always* conflicting

example:

- Performance 
- Budget (time, money) 

35

Design Process

- Collect obvious design(s)
- Search for *one* non-obvious design
- Compare the relative ROI of the designs
- Select the best compromise
- Describe the selected design
- Use a DesignLog to document the decision process

- Books:
  - Ralph L. Keeyney: Value Focused Thinking
  - Gerd Gigerenzer: Simple Heuristics That Make Us Smart

36

Impact Estimation

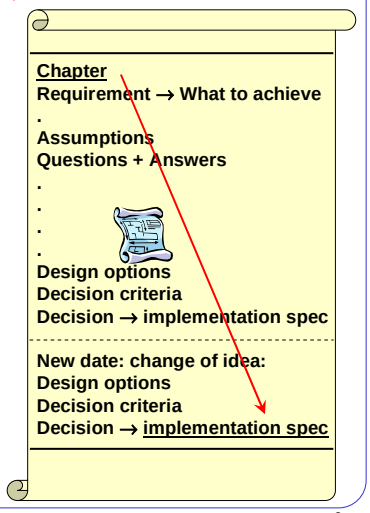
	On-line Support	*On-line Help*	*Picture Handbook*	*On-line Help + Access Index*
Learning 60 minutes <-> 10 minutes				
Scale Impact	5 min.	10 min.	30 min.	8 min.
Scale Uncertainty	±3 min.	±5 min.	±10 min.	±5 min.
Percentage Impact	110%	100%	60%	104%
Percentage Uncertainty	±6% (3 of 50 minutes)	±10%	±20%	±10%
Evidence	Project Ajax: 7 minutes	Other Systems	Guess	Other Systems + Guess
Source	Ajax Report, p.6	World Report, p.17	John B	World Report, p.17 + John B
Credibility	0.7	0.8	0.2	0.6
Development Cost	120 K	25 K	10 K	26 K
Performance to Cost Ratio	110/120 = 0.92	100/25 = 4.0	60/10 = 6.0	104/26 = 4.0
Credibility-adjusted Performance to Cost Ratio (to 1 decimal place)	0.92*0.7 = 0.6	4.0*0.8 = 3.2	6.0*0.2 = 1.2	4.0*0.6 = 2.4

ref
Tom Gilb
Competitive Engineering

37

DesignLog (project level)

- In computer, not loose notes, not in e-mails, not handwritten
  - Text
  - Drawings!
  - On subject order
  - Initially free-format
  - For all to see
- All concepts contemplated
  - Requirements
  - Assumptions
  - Questions
  - Available techniques
  - Calculations
  - Choices + argumentation:
    - If rejected: why?
    - If chosen: why?
- Rejected choices
- Final (current) choices
- Implementation

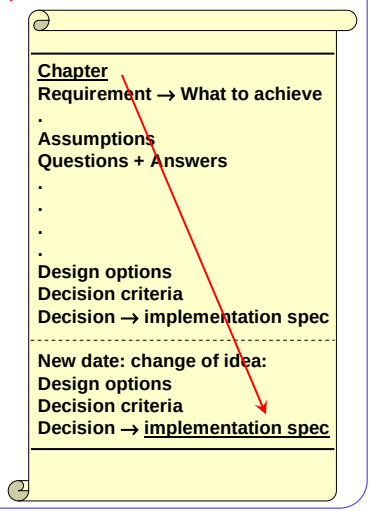


38

ProcessLog

(department / organization level)

- In computer, not loose notes, not in e-mails, not handwritten
  - Text
  - Graphics (drawings)
  - On subject order
  - Initially free-format
  - For all to see
- All concepts contemplated
  - Related requirement
  - Assumptions
  - Questions
  - Known techniques
  - Choices + argumentation:
    - If rejected: why?
    - If chosen: why?
- Rejected choices
- Final (current) choices
- Implementation



39

Evo Planning: Weekly TaskCycle

- Goal is not to be a good estimator
- Goal is to learn to promise what you can and will do and then to live up to your promises
- It's easier to estimate in hours than in pieces of cake
- We estimate net effort to do the work
- All work to be 100% done at the end of the week
- We plan 2/3 of the available time
- The other 1/3 is for all other things you'll do anyway
- We only work on planned things

40

Optimizing Estimation

- Immediately consuming the metrics for learning
- Change from optimistic to realistic estimation in 3 weeks
- Only if we are Serious about Time (Sense of Urgency)
- Also using the metrics for calibration of predictions
- Estimation method: Intuition + optimizing intuition
- The person doing the task is estimating
- Others should never challenge the estimation
- Estimates are non-negotiable!
- We can negotioate about the contents

41

Evo Planning: Bi-weekly DeliveryCycle

- What can and will you have realized at the end of the cycle
- Deliveries are never postponed
- Halfway we gauge whether we'll make it
- If not, what can we do to still deliver

42

The problems in projects are not the real problem,
the real problem is that we don't do something about it

43

Getting and Keeping Projects Under Control

Niels Malotaux

N R Malotaux
Consultancy

+31-30-228 88 68 niels@malotaux.nl www.malotaux.nl

44