

INCOSE International Symposium 2009 Singapore

Tutorial
July 2009

Niels Malotaux

Evolutionary Project Planning

**How Systems Engineers can Contribute
to
Getting and Keeping the Project On Time**

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

Niels Malotaux

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Niels Malotaux

Niels Malotaux is an independent Project Coach specializing in optimizing project performance. He has over 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 over 100 projects in 25+ organizations in the Netherlands, Belgium, China, Germany, India, Ireland, 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 the practice of real projects.

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.

<p>N R Malotaux Consultancy</p>	
<p>Niels Malotaux project coach</p>	<p>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</p>
<p><i>Result Management</i></p>	

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

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

Booklets:

www.malotaux.nl/nrm/pdf/MxEvo.pdf - www.malotaux.nl/nrm/pdf/Booklet2.pdf
www.malotaux.nl/nrm/pdf/EvoQA.pdf - www.malotaux.nl/nrm/pdf/EvoRisk.pdf
www.malotaux.nl/nrm/pdf/TimeLineIS09.pdf - www.malotaux.nl/nrm/pdf/HumanBehavior.pdf

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

The Right Result at the Right Time

- Do you regularly deliver the Right Result at the Right Time?
 - Why not?
 - Is this normal?
 - Can we do something about it?
-
- What is the Right Result?
 - What is the Right Time?

3

The problem

- Many projects don't deliver the right Results
- Many projects deliver late

or, more positively:

- I want my project to be more successful
- In shorter time

4

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

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

5

What is the most important requirement ?

6

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Time as a Requirement

- **Delivery Time is a Requirement, like all other Requirements**
- **How come most projects are late ???**
- **Apparently all other Requirements are more important than Delivery Time**
- **Are they really?**

7

Fallacy of *all* requirements

- **We're done when *all* requirements are implemented**
- **Isn't delivery time a requirement ?**
- **Requirements are always contradictory**
- **Perception of the requirements**
- **Who's requirements**
- **Do we really know the real requirements?**
- **Are customers able to define requirements ?**
 - Customers specify things they do not need
 - And forget things they do need
- **What we *think* we have to do should fit the available time**
- **Use the Business Case**

8

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Will your current project be on time ?

- **Was your previous project successful and on time ?**
- **Will your current project be successful and on time ?**
- **How do you know ?**

9

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

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

10

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Causes of Delay

- **Some typical causes of delay are:**
 - Developing the wrong things
 - Unclear requirements
 - Misunderstandings
 - No feedback from stakeholders
 - No adequate planning
 - No adequate communication
 - Doing unnecessary things
 - Doing things less cleverly
 - Waiting (before and during the project)
 - Changing requirements
 - Doing things over
 - Indecisiveness
 - Suppliers
 - Quality of suppliers results
 - No Sense of Urgency
 - Hobbying
 - Political ploys
 - Boss is always right (culture)
- **A lot of delay is avoidable and therefore unjustifiable**
- **Excuses, excuses: “external factors” being the cause of delays**

11

But we're Systems Engineers !

- **What caused the project being late?**
- **Could we have prevented the project being late?**
- **Was delivery time important?**
- **Was delivery time a requirement?**
- **Were *all* other requirements really more important?**

12

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Systems Engineers

- **Other Engineers**

- Silo thinking
- Sub-optimizing
- Gold plating (hobbies)
- Little attention to interfaces



- **Systems Engineers**

- Multi-dimensional thinking
- Optimizing design decisions over all dimensions
- All disciplines
- Whole life-cycle (cradle to cradle)
- Balancing requirements
- Including delivery time

13

What is On Time ?

- **Yesterday?**
- **Before the next exhibition?**
- **Managers dream?**
- **Time to market?**
- **Time to profit?**

**Compromise between what is needed
and what is possible**

just like any other requirement

14

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

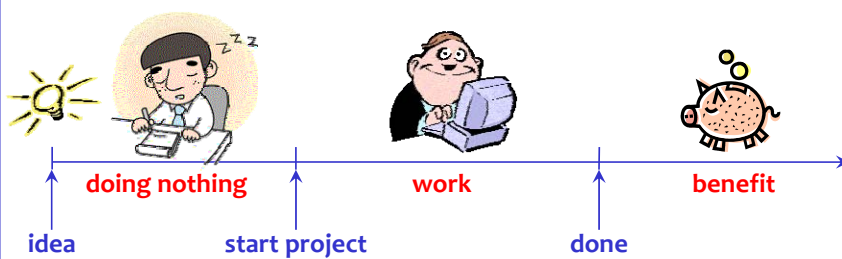
Time to market

- 5000 products per year \approx 20 products per day
- € 5000 per product
- Profit € 500 per product
- Profit € 10.000 per day

*Every day you start later, you'll be done a day later
and miss € 10.000*

15

What's the Cost of Doing Nothing ?

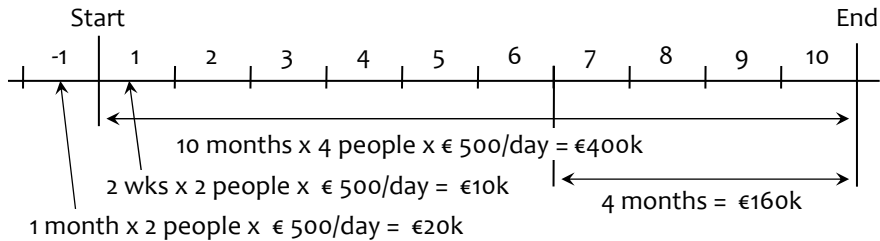


16

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

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 € 16M/24 = € 670k
- And saving 4 months brings € 16M/3 = € 5M extra
- Invest that €200k NOW and don’t - waste - time !

17

Cost of one day of delay of our projects

- Cost of doing nothing
- Cost of doing
- Revenues once done
- (delay and lost revenues of the next project)

18

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

What could we have done to save time?

19

Excuses, excuses, excuses ...

- **We have been thoroughly trained to make excuses**
- **We always downplay our failures**

- **At the Fatal Day, any excuse is in vain: we failed**
- **Even if we “couldn’t do anything about it”**
- **No pain, no gain**

20

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

How about this challenge ?

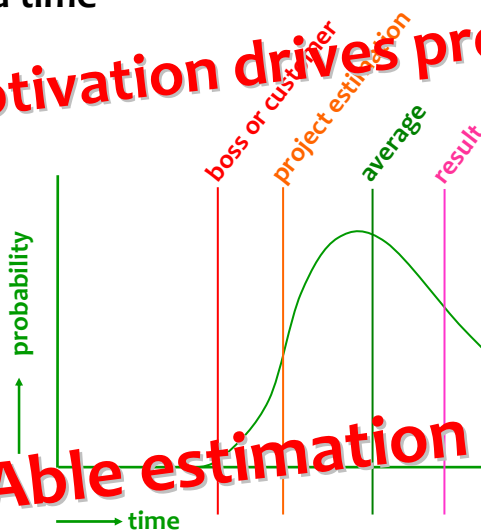
- Getting and keeping the project under control
- Never to be late
- We don't want to fail, we're not going to let it happen
- No excuses needed
- Not stealing from our customer's (boss') purse
- If it's impossible, how quickly will we know ?
- The only justifiable cost is the cost of developing the right things at the right time

21

Lead time

Motivation drives productivity

Able estimation is vital



22

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Estimation Exercise



Are you an optimistic or a realistic estimator?

Let's find out !

**Project:
Multiplying two numbers of 4 figures**

**How many seconds would you need to
complete this Project?**

23

Is this what you did?

24

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Defect rate

- Before test ?
- After test ?

25

Alternative Design (*how to solve the requirement*)

26

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Another alternative design

27

What was the real requirement?

Assumptions, assumptions ...

Better assume that many assumptions are wrong.

Check !

28

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Elements in the exercise

- Estimation, optimistic / realistic
- Interrupts
- Test, test strategy
- Defect-rate
- Design
- Requirements
- Assumptions

29

Murphy's Law

- Whatever can go wrong, will go wrong
- Should we accept fate?

Murphy's Law for Engineers:

- Whatever can go wrong, will go wrong ...

Therefore:

- We should actively check all possibilities that can go wrong and *make sure that they cannot happen*

30

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Preflection, foresight, prevention

Insanity is doing the same things over and over again and hoping the outcome to be different (let alone better)

Albert Einstein 1879-1955, Benjamin Franklin 1706-1790, it seems Franklin was first

Only if we change our way of working, the result may be different

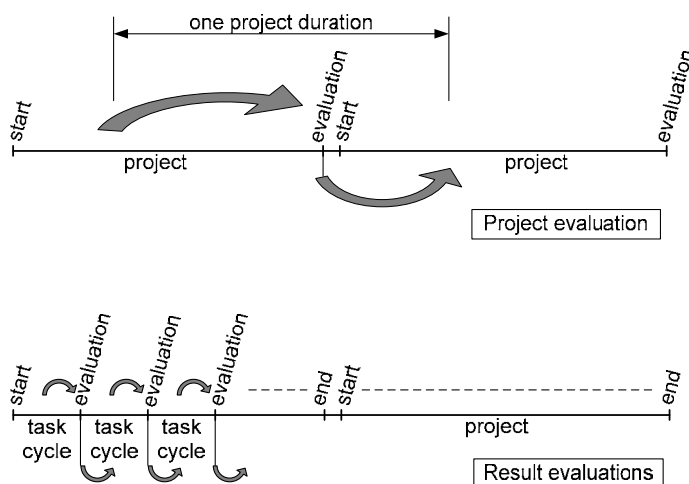
- **Hindsight** is easy, but reactive
- **Foresight** is less easy, but proactive
- **Reflection** is for hindsight and learning
- **Preflection** is for foresight and prevention

Only with prevention we can save precious time

This is used in the Deming or Plan-Do-Check-Act cycle

31

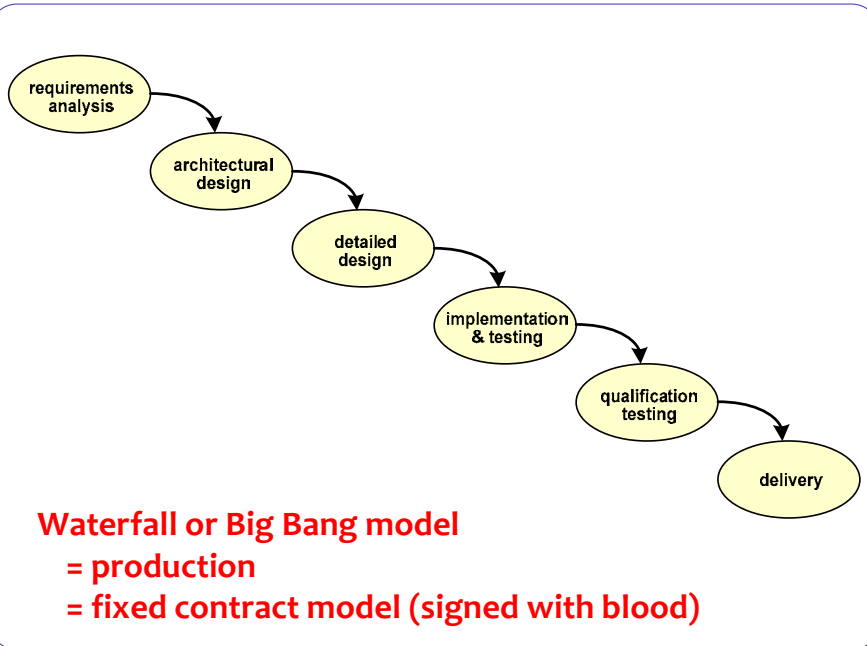
Project evaluations



32

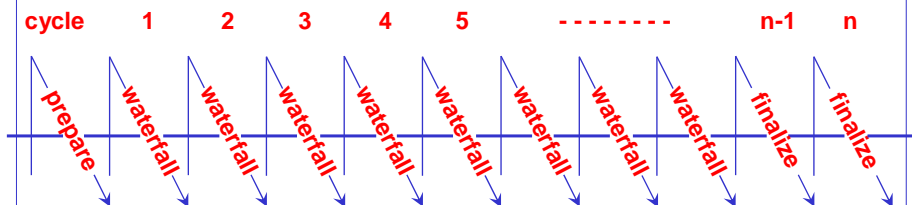
Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time



33

Using many waterfalls of growing functionality



34

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Knowledge

how to achieve the goal

If we

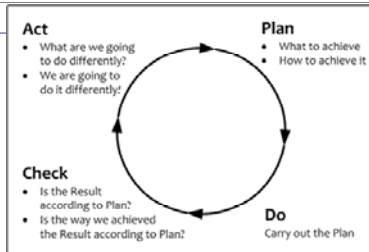
- Use very short Plan-Do-Check-Act cycles
- Constantly selecting the most important things to do

then we can

- Most quickly learn what the real requirements are
- Learn how to most effectively and efficiently realize these requirements

and we can

- Spot problems quicker, allowing more time to do something about them



doing the right things

doing the right things right

35

The essential ingredient: the PDCA cycle

(Deming cycle)

Act

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

Plan

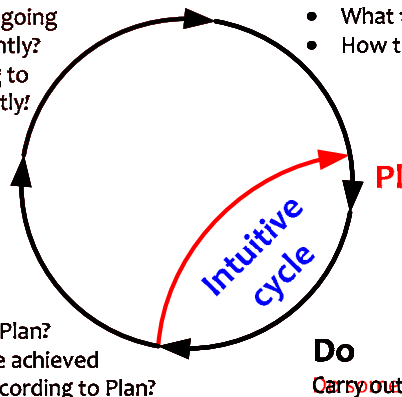
- What to achieve
- How to achieve it

Check

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

Do

Carry out the Plan

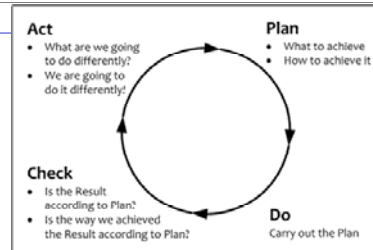


36

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

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”

37

- **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 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

38

Booklets:

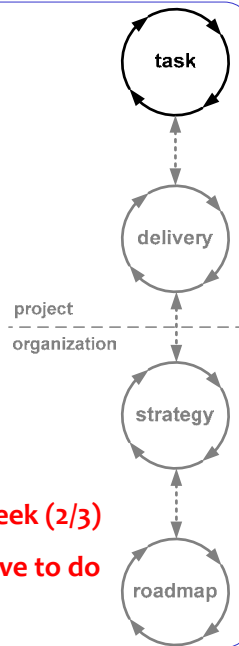
- www.malotaux.nl/nrm/pdf/MxEvo.pdf - www.malotaux.nl/nrm/pdf/Booklet2.pdf
- www.malotaux.nl/nrm/pdf/EvoQA.pdf - www.malotaux.nl/nrm/pdf/EvoRisk.pdf
- www.malotaux.nl/nrm/pdf/TimeLineISo9.pdf - www.malotaux.nl/nrm/pdf/HumanBehavior.pdf

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

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



39

Effort and Lead Time

- Days estimation → lead time (calendar time)
- Hours estimation → effort
- Effort variations and lead time variations have different causes
- Treat them differently and keep them separate
 - Effort: complexity
 - Lead Time: time-management
 - (effort / lead-time ratio)

40

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Weekly 3-Step Procedure

1. **Individual preparation**
 - Conclude current tasks
 - What to do next
 - Estimations
 - How much time available
2. **Modulation with / coaching by Project Management**
 - Status
 - Priority check
 - Feasibility
 - Commitment and decision
3. **Synchronization with group (team meeting)**
 - Formal confirmation
 - Concurrency
 - Learning
 - Helping
 - Socializing

41

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

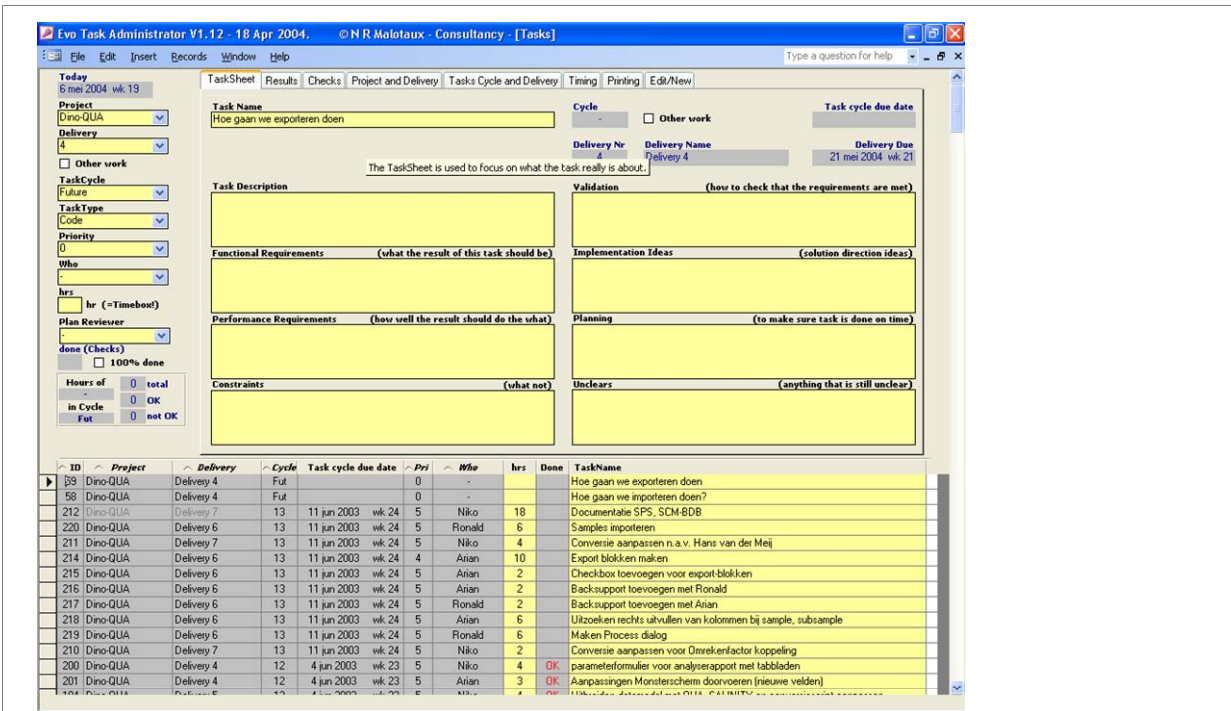
2/3 is default start value
this value works well in development projects

Taska	2	↑	
Taskb	5		
Taskc	3		
Taskd	6		do
Taske	1		
Taskf	4	26	
Taskg	5		
Taskh	4	↓	
Taskj	3		do
Taskk	1		not

42

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time



TaskCycle Exercise

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

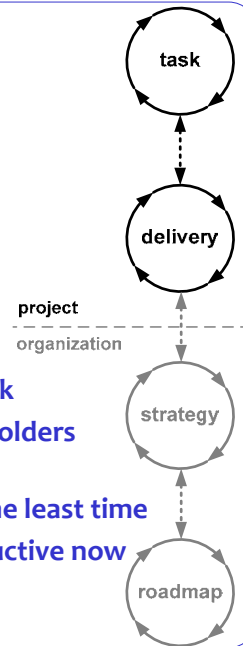
Task a	2	↑ do
Task b	5	
Task c	3	
Task d	6	
Task e	1	
Task f	4	
Task g	5	
Task h	4	↓ not do
Task j	3	
Task k	1	

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

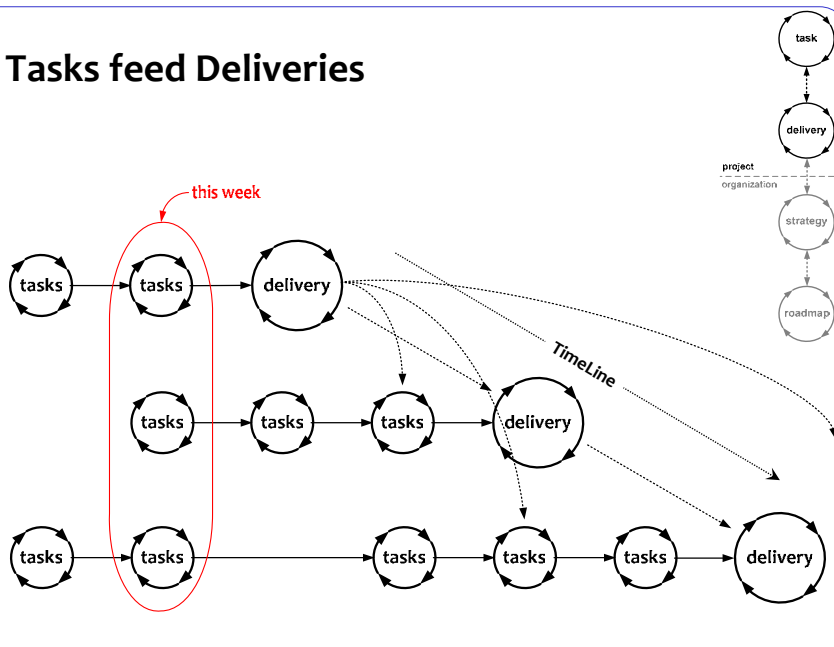
DeliveryCycle

- **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**



45

Tasks feed Deliveries



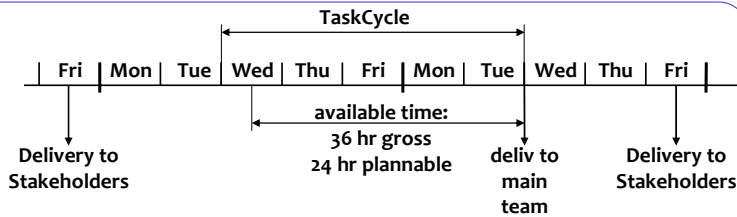
46

Niels Malotaux

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Designing a Delivery



Serge (ProjLead)		Gregory		Gregory (later)	
MbWA	3	Draft design	6	Draft design	0
Planning next wk	3	Finish design	6	Finish design	0
Work for deliv	4	Work for deliv	3	...	
-	6	-	1	Repair deliv	0
-	2	-	2	...	
-	1	-	2	...	
-	5	-	3	...	
Total	24	-	5	...	
		-	6	Jerome	
		XMLa	4	XMLa	3
		XMLb	4	XMLb	3
		Total	42	...	

Zero Defects Attitude

47

Interrupts

- Boss comes in: "Can you paint my fence?"
- What do you do?

- In case of interrupt, use interrupt procedure

48

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Interrupt Procedure "We shall work only on planned Tasks"

In case a new task suddenly appears in the middle of a Task Cycle (we call this an *Interrupt*) we follow this procedure:

1. Define the expected Results of the new Task properly
2. Estimate the time needed to perform the new Task, to the level of detail really needed
3. Go to your task planning tool (many projects use the ETA tool)
4. Decide which of the planned Tasks is/are going to be sacrificed (up to the number of hours needed for the new Task)
5. Weigh the priorities of the new Task against the Task(s) to be sacrificed
6. Decide which is more important
7. If the new Task is more important: replan accordingly
8. If the new Task is not more important, then do not replan and do not work on the new Task. Of course the new Task may be added to the Candidate Task List
9. Now we are still working on planned Tasks.

49

Active Synchronization

Somewhere around you, there is the bad world.

If you are waiting for a result outside your control, there are three possible cases:

1. You are sure they'll deliver Quality On Time
2. You are not sure
3. You are sure they'll not deliver Quality On Time
 - If you are not sure (case 2), better assume case 3
 - From other Evo projects you should expect case 1
 - Evo suppliers behave like case 1

In cases 2 and 3: Actively Synchronize: Go there !

1. Showing up increases your priority
2. You can resolve issues which otherwise would delay delivery
3. If they are really late, you'll know much earlier

50

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Agile, but will we be on time ?

- Organizing the work in very short cycles
- To make sure we are doing the right things
- And that we are doing it the right way
- So, we already work more efficiently

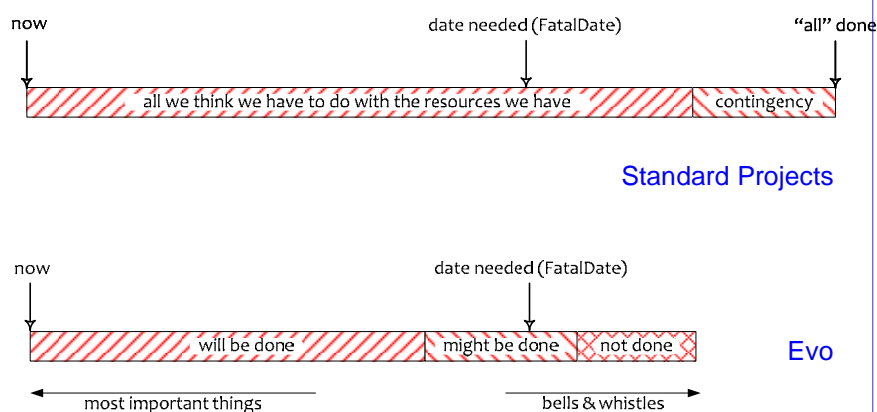
but ...

- How do we make sure the whole project is done on time?

51

TimeLine

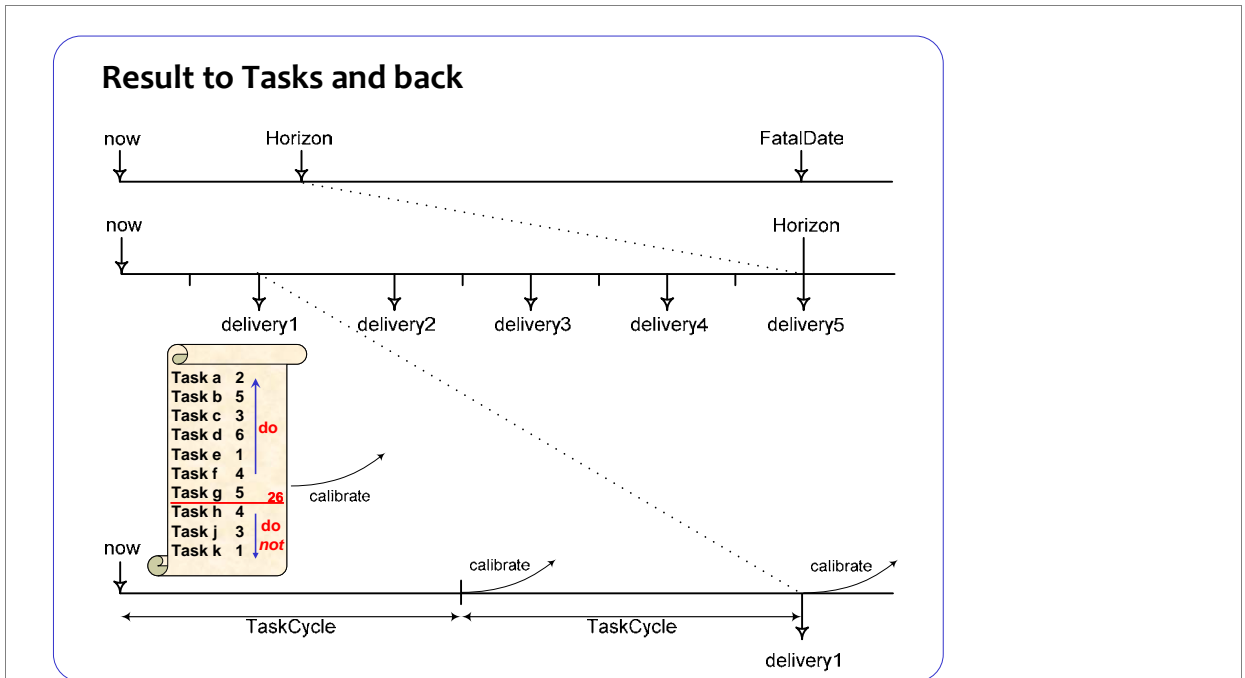
What the customer wants, he cannot afford



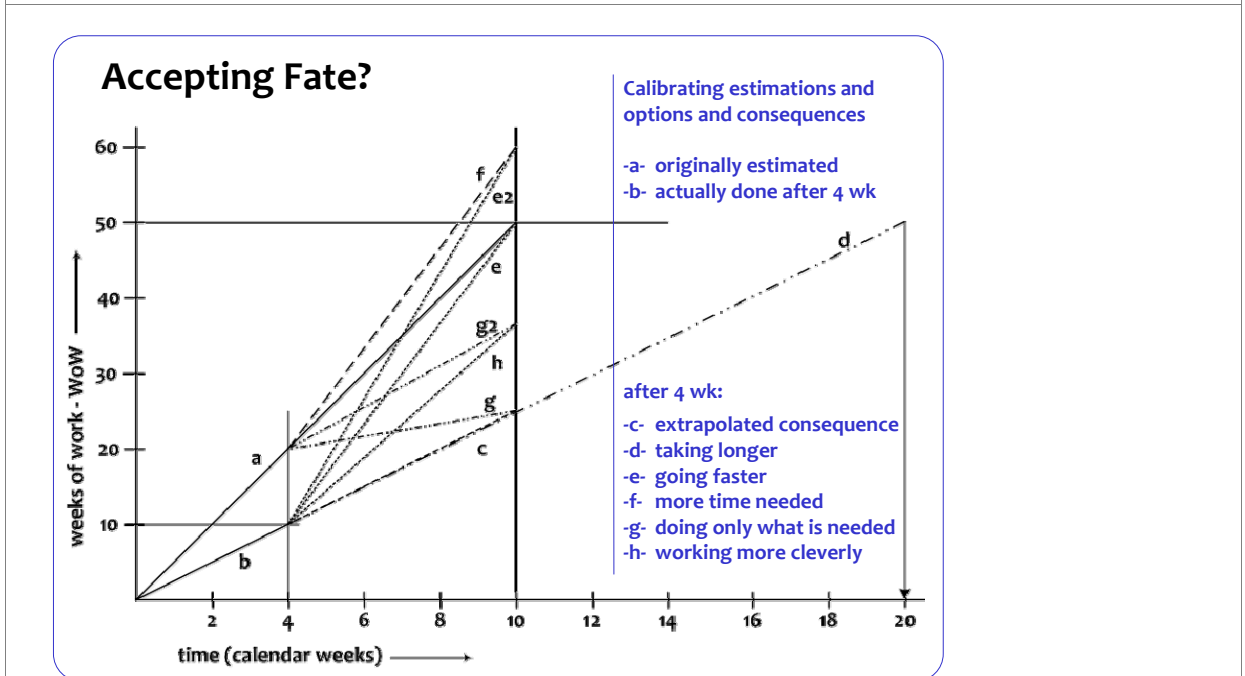
52

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time



53



54

Booklets:

- www.malotau.nl/nrm/pdf/MxEvo.pdf -
- www.malotau.nl/nrm/pdf/EvoQA.pdf -
- www.malotau.nl/nrm/pdf/TimeLineIS09.pdf -
- www.malotau.nl/nrm/pdf/Booklet2.pdf
- www.malotau.nl/nrm/pdf/EvoRisk.pdf
- www.malotau.nl/nrm/pdf/HumanBehavior.pdf

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Deceptive options

- **Hoping for the best** (fatalistic)
- **Going for it** (macho)
- **Working Overtime** (fooling oneself)
- **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

57

Adding people to a late project ...

makes it later

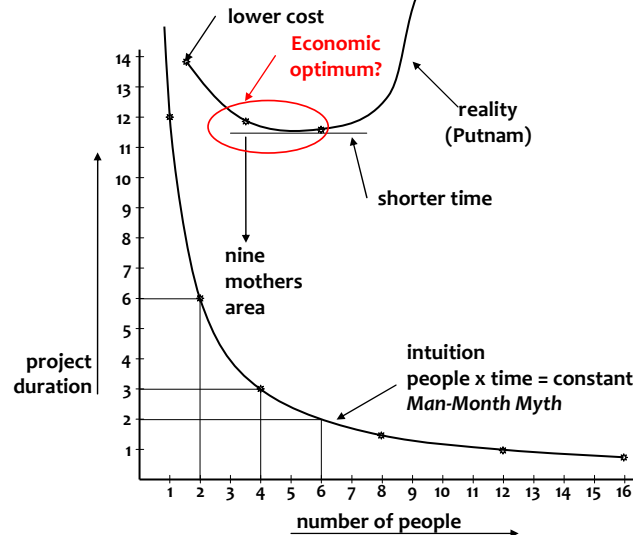
(Brooks' Law, 1975)

58

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Project-duration



59

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**

60

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

- **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

Evolutionary Project Management (Evo)

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

Evo Project Planning

61

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



62

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Culture

Ingrained customs

- How do we behave between each other
- We don't know any more why we do it
- Still, it's acquired by experience
- (Probably) not genetic

63

Things I heard

- **Group is important**
 - Project team is a group
- **Face**
 - We are not perfect, but the customer should never find out
- **Cannot say "No"**
 - How do you then indicate "no"?
- **Is that clear? - Yes**
 - If you don't understand:
 - Is the teacher unclear ?
 - Am I stupid ?
- **Authority**
 - Boss is always right
 - Teacher is always right

64

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Things I heard (2)

- **Group is responsible**
 - Personal responsibility ?
- **Survival**
 - Win or Win - win ?
- **Harmony**
 - With whom ?
 - To what extent ?

65

Cultural differences ?

Dutch

open, direct, explicit, blunt
informal
arrogant
preaching
assertive
can say no
egalitarian, not showing wealth
consensus
rules, tasks
win-win
little power distance
authority must be earned
little brand value
not spending more than necessary

You ?

66

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

TimeLine

- The TimeLine technique doesn't solve our problems
- They help to expose the real status early and continuously
- Instead of *accepting* the undesired outcome, we do something about it
- The earlier we know, the more we can do about it
- We start saving time from the very beginning
- We can save a lot of time in any project, while producing a better outcome
- If, and only if, we are serious about time !



67

Some Examples from Practice

Niels Malotaux

N R Malotaux
Consultancy

+31-30-228 88 68

niels@malotaux.nl

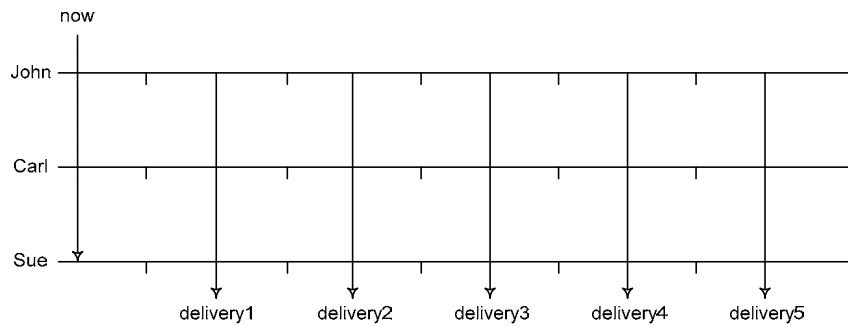
www.malotaux.nl

68

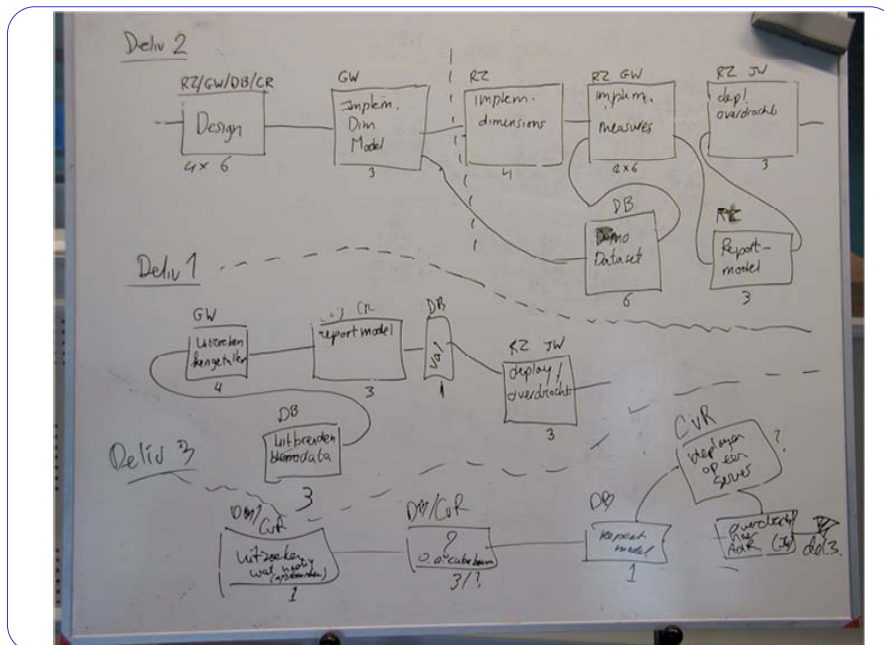
Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Making individual TimeLines



69



70

Booklets:

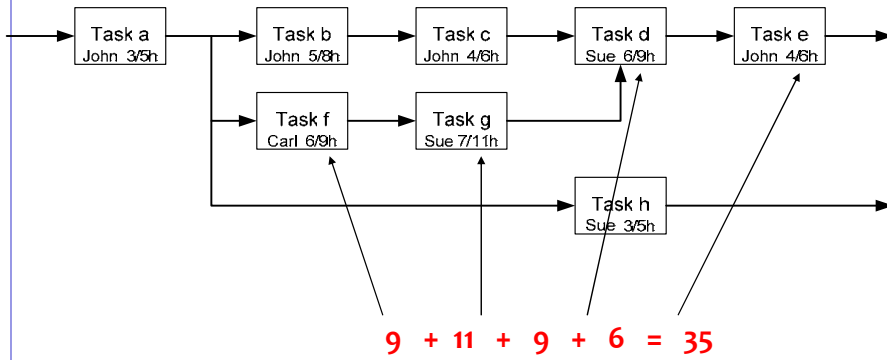
- www.malotaux.nl/nrm/pdf/MxEvo.pdf -
- www.malotaux.nl/nrm/pdf/EvoQA.pdf -
- www.malotaux.nl/nrm/pdf/TimeLineISo9.pdf -
- www.malotaux.nl/nrm/pdf/Booklet2.pdf
- www.malotaux.nl/nrm/pdf/EvoRisk.pdf
- www.malotaux.nl/nrm/pdf/HumanBehavior.pdf

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

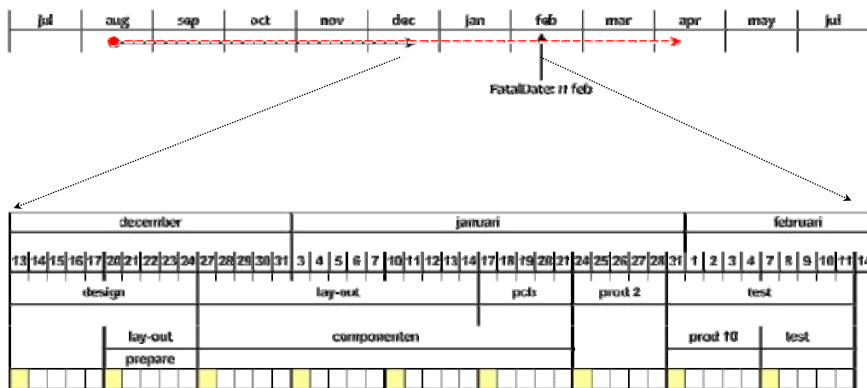
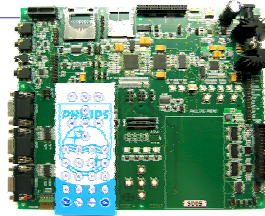
PERT (Project Evaluation Review Technique)

used for *Designing a Delivery*



71

TimeLine planning

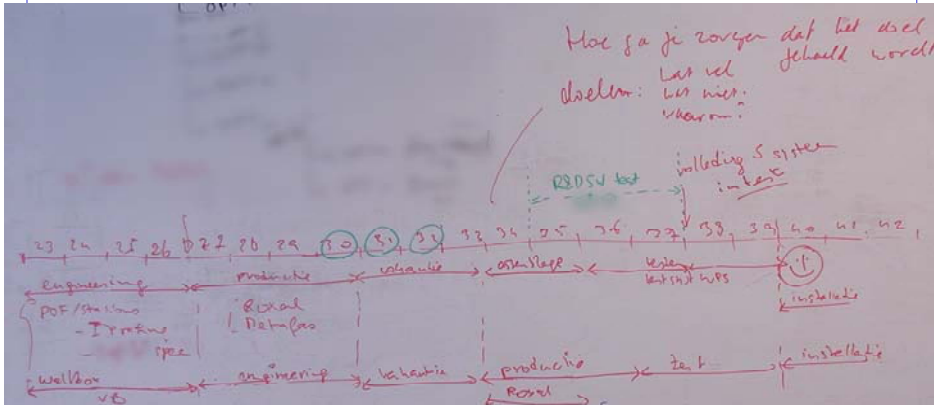


72

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Whiteboard TimeLine Planning



75

We have a QA problem !

- Large stockpile of modules to be tested
- Estimate: will cost half year of testing
- You shall do Full Regression Tests
- Full Regression Tests take about 15 days each
- QA is bottleneck
- Can we do something about this?



76

Niels Malotaux

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

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/Booklets
 1. Evolutionary Project Management Methods (2001)
 2. How Quality is Assured by Evolutionary Methods (2004)
 3. Optimizing the Contribution of Testing to Project Success (2005)
 - 3A. Optimizing Quality Assurance for Better Results (2005)
 4. Controlling Project Risk by Design (2006)
 5. TimeLine: How to Get and Keep Control over Longer Periods of Time (2007)
 6. Human Behavior in Projects (2008)
 7. How to Achieve the Most Important Requirement (2008)
- www.malotaux.nl/nrm/Evo/ETAF.htm
Download the Evo Task Administrator (ETA) tool
(expects MSAccess 2000-2003)

79

Getting and Keeping Projects Under Control

Niels Malotaux

N R Malotaux
Consultancy

+31-30-228 88 68

niels@malotaux.nl

www.malotaux.nl

80

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

Simple Delphi estimation



1. **Make a list of things we think we have to do in just enough detail**
2. **Distribute the list among people who will do the work, or who should be knowledgeable about the work**
3. **Ask them to add what we apparently forgot, and to estimate how much time the elements of work would cost, "as far as you can judge"**
4. **In a meeting the estimates are compared**
5. **If estimates differ significantly between estimators, do not take the average, but discuss about the contents of the work, not about the estimate** (some may forget to include things that have to be done, some others may think that more has to be done than necessary)
6. **After discussion, people estimate individually again and the estimates are compared again**
7. **Repeat until sufficient consensus (usually not more than once or twice)**
8. **Add up all the estimates to end up with an estimate for the whole project**

81

Simple and Simpler Delphi



- | | |
|------------------------------------|--|
| 1. List things to do | 1. List things to do |
| 2. Distribute the list | 2. Distribute the list |
| 3. Add and estimate | 3. Add and estimate |
| 4. List estimates | 4. List estimates: min and max |
| 5. Discuss if differences | 5. Discuss if differences |
| 6. Estimate again | 6. Agree on value between min and max |
| 7. Repeat until consensus | 7. Add up all the estimates |
| 8. Add up all the estimates | |

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

82

Evolutionary Project Planning

How Systems Engineers can Contribute to Getting and Keeping the Project On Time

0th- order approximations

- In the Business Case we often use 0th- order estimations
- Order of magnitude
- Better than $0 < \text{guess} < \infty$ (any number is better than no number)
- 0th order is better than *no clue*
- 1st order is often less accurate than 0th order
- Using two different ways of estimation for crosscheck
- Errors may average if we estimate several pieces

83

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)
- 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 and should negotiate about the *contents*

84