Evolutionary Planning

Producing even more in less time

www.malotaux.nl/conferences

Niels Malotaux

N R Malotaux
Consultancy

+31 655 753 604 niels@malotaux.nl www.malotaux.nl

Niels Malotaux

- Team and Organizational Coach
- Expert in helping optimizing performance

- Helping projects and organizations very quickly to become
  - More effective – doing the right things better
  - More efficient – doing the right things better in less time
  - Predictable – delivering as predicted
- Helping teams to shine

Result Management

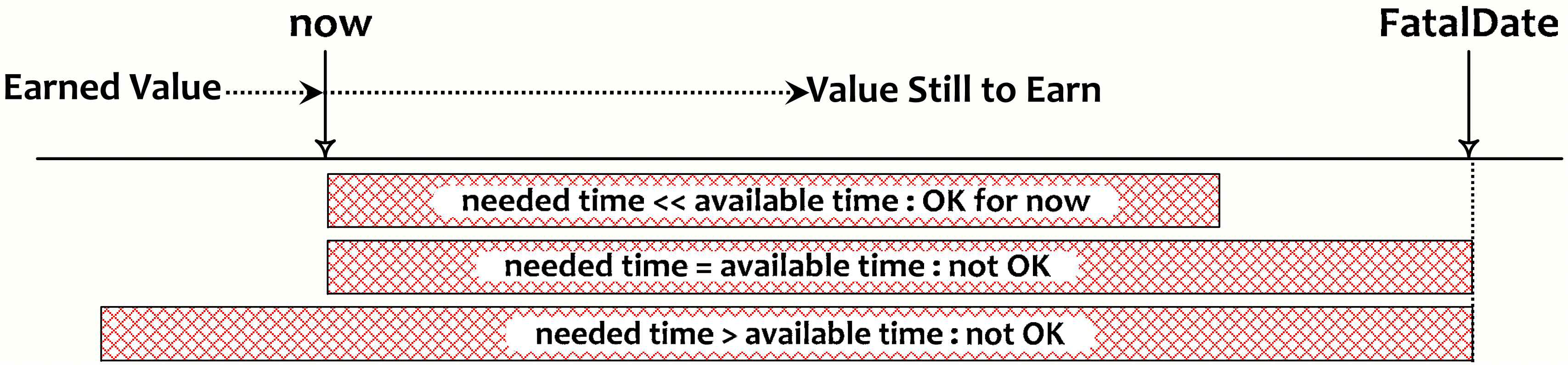
Goal of What We Do

Quality on Time

- Delivering the Right Result at the Right Time, wasting as little time as possible (= efficiently)

- 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

Any Deadlines ?



now

FatalDate

Earned Value

Value Still to Earn

needed time << available time : OK for now

needed time = available time : not OK

needed time > available time : not OK

- **Value Still to Earn**

versus

- **Time Still Available**

If the match is over, you cannot score a goal

Deceptive and dangerous options

- **Deceptive options**
  - **Hoping for the best** (fatalistic)
  - **Going for it** (macho)
  - **Working Overtime** (fooling ourselves and the boss)
  - **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
- **Dangerous** (but sometimes necessary) **option**
  - **Adding people**
    - Beware of Brooks' Law (1975)
      - Adding people to a late project … makes it later

Saving time

Continuous elimination of waste

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, instead of the solution we always used
  - The project
    - Doing the same in less time, instead 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** - right time, in the right order
- **TimeBoxing** - much more efficient than FeatureBoxing

Do you use Retrospectives ?

Do we really learn from what happened ?

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

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

Use Prespectives

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

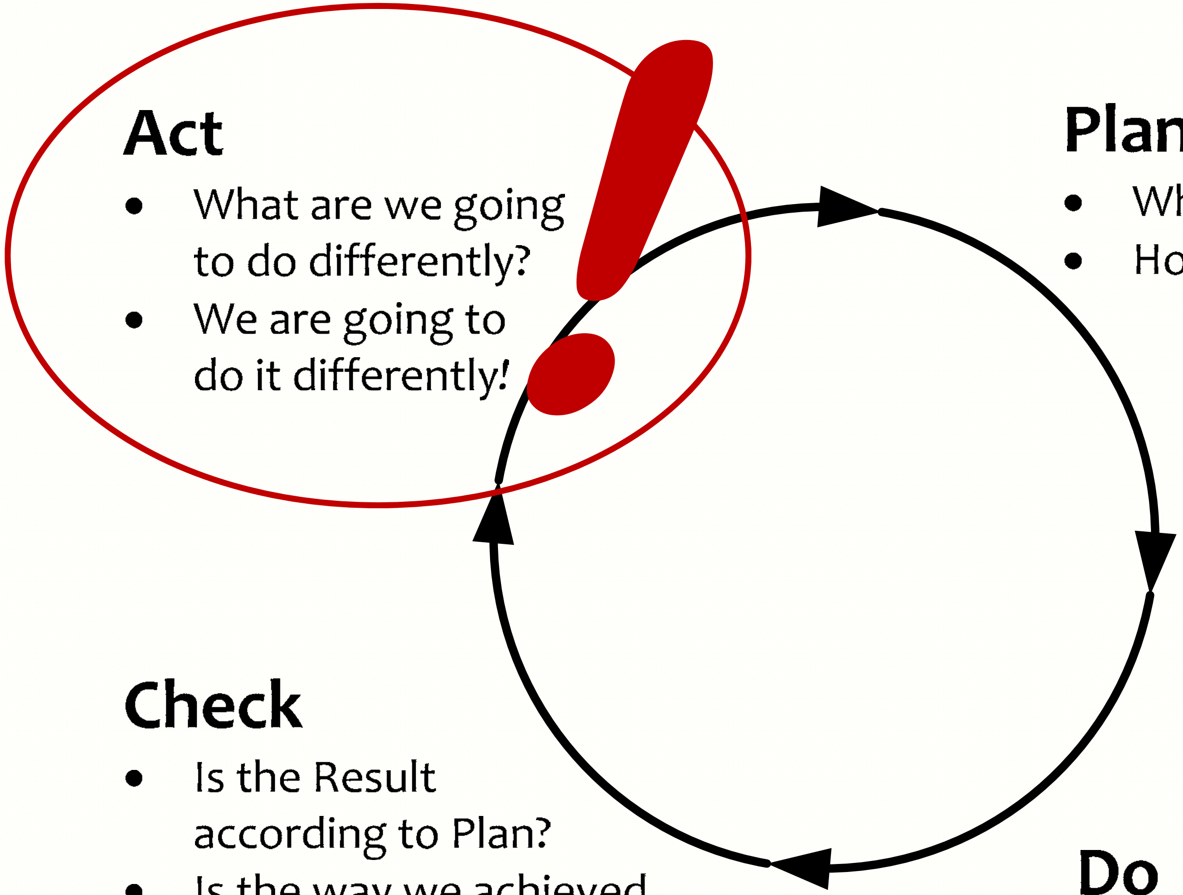
- Hindsight is easy, but reactive
- Foresight is less easy, but proactive
- Reflection is for hindsight and learning
- Preflection is for foresight and prevention

Only with *prevention* we can save precious time

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

The essential ingredient: the PDCA Cycle

(Shewhart Cycle - Deming Cycle - Plan-Do-Study-Act Cycle - Kaizen)



Act

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

Plan

- What to achieve
- How to achieve it

Do

Carry out the Plan

Check

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

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

Why

What

How much

Are we done

How

Check as early as possible

Right Result

Zero Defects Attitude

Evo Project Planning

- **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

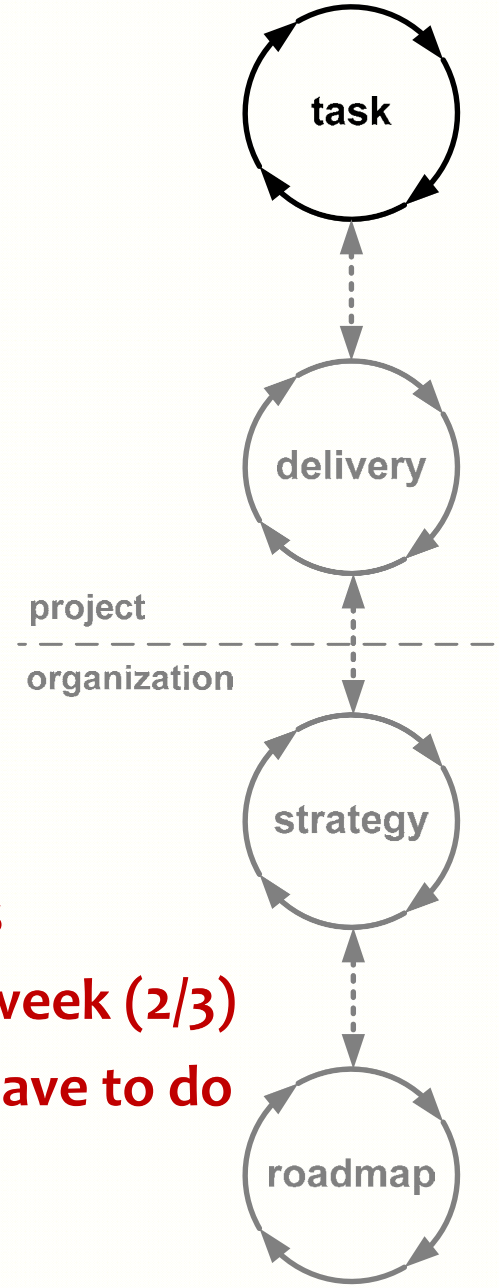
Efficiency of what we do

Effectiveness of what we do

Right Time

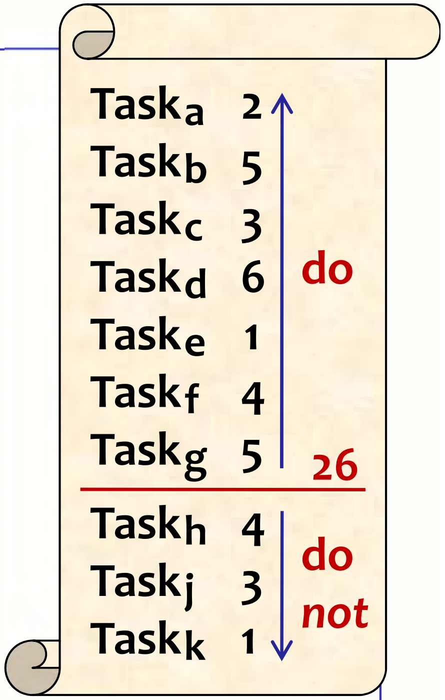
What will happen and what will we do about it ?

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

Weekly TaskCycle

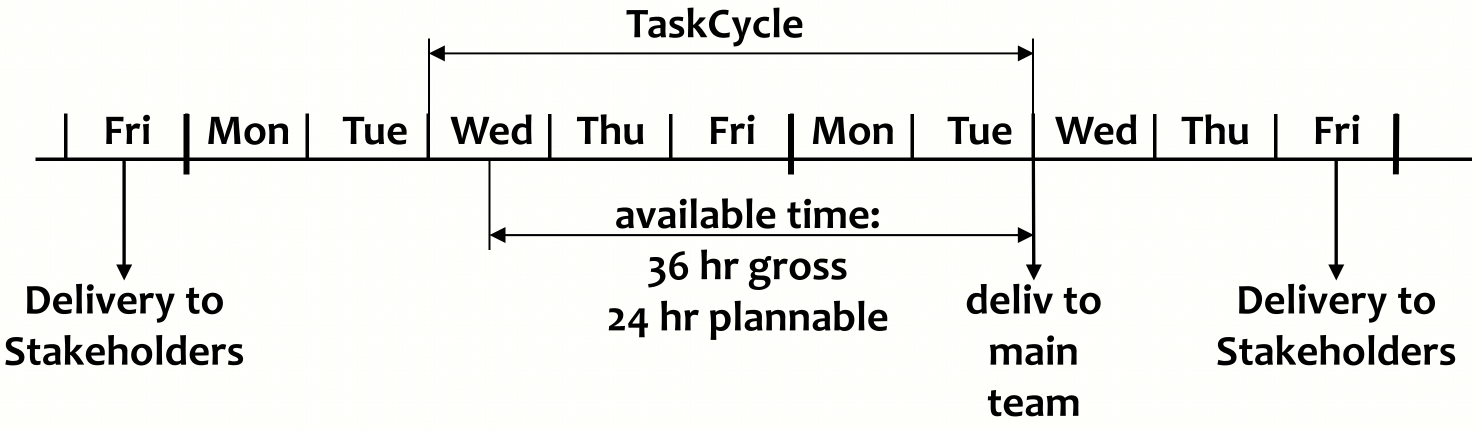


Task_a	2	
Task_b	5	
Task_c	3	
Task_d	6	do
Task_e	1	
Task_f	4	
Task_g	5	26
Task_h	4	do
Task_j	3	*not*
Task_k	1	

- 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

Designing a Delivery



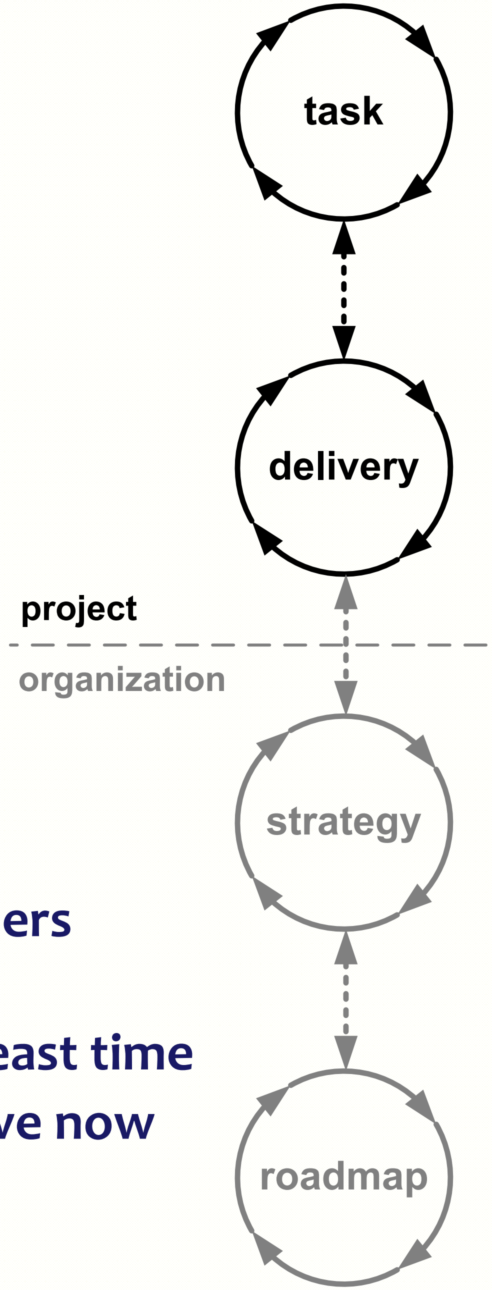
Serge (ProjLead)	
MbWA	3
Planning nxt wk	3
Work for deliv	4
-	6
-	2
-	1
-	5
Total	24

Gregory			Gregory (later)	
Draft design	6	→	Draft design	0
Finish design	6	→	Finish design	0
Work for deliv	3		...	
-	1			
-	2			
-	2			
-	3			
-	5			
-	6		**Jerome**	
XMLa	4	→	XMLa	3
XMLb	4	→	XMLb	3
Total	42		...	

Why is this important?

- Half (±30%) of what people do in projects later proves not having been necessary
- During the TaskCycle planning we can very efficiently see
  - What our colleagues think they're going to do
  - Make sure they're going to work on the most important things
  - Not on unnecessary things
  - In line with the architecture and design
  - Leading most efficiently to the goal of the delivery

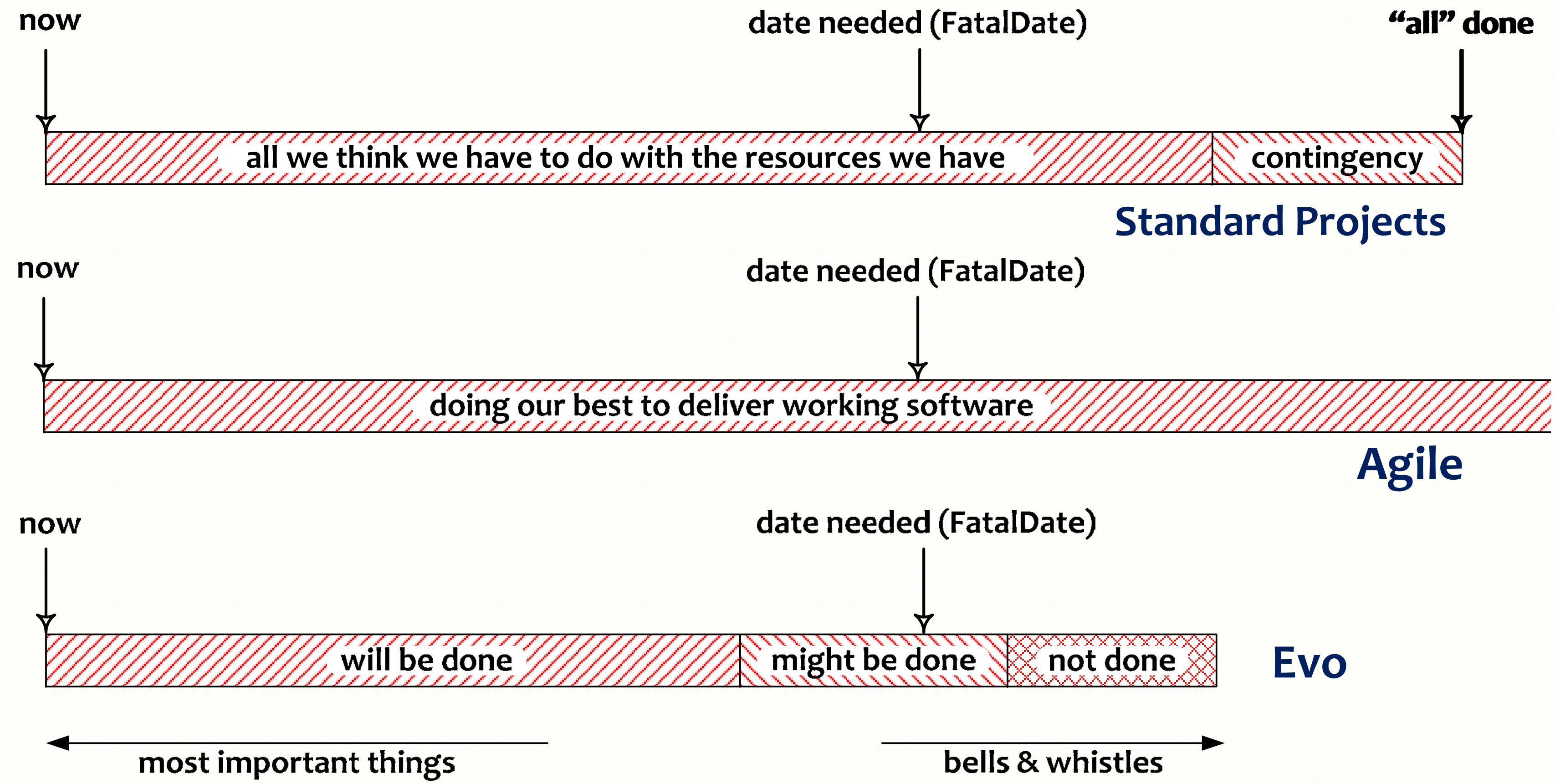
DeliveryCycle



- Are we *delivering*
  the right things, in the right order
  to the right level of detail for now
- Optimizing requirements
  and checking assumptions
  1. What will generate the optimum feedback
  2. We deliver only to eagerly waiting stakeholders
  3. 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 (it can be less !)

TimeLine

How de we know that the business gets *what* they need, *when* they need it?



now

date needed (FatalDate)

"all" done

all we think we have to do with the resources we have

contingency

Standard Projects

now

date needed (FatalDate)

doing our best to deliver working software

Agile

now

date needed (FatalDate)

will be done

might be done

not done

Evo

most important things

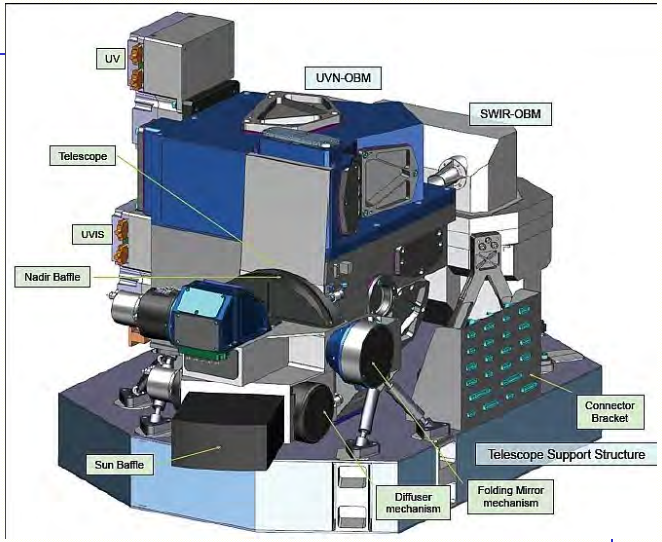
bells & whistles

- **Better 80% 100% done, than 100% 80% done**
- **Let it be the most important 80%**

Sorry.
Picture removed for confidentiality

Sorry.
Picture removed for confidentiality

Immediate savings



Space Instrument development project

- Savings: some 40 man-year

Oscilloscope development

- Delivery 50% faster than average overrun over the last 5 years
- Fastest time-to-market, highest quality at intro in more than 10 years
- Team won a prestigious Team Award as part of the company's Technical Excellence recognition program

More

www.malotaux.nl/booklets

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

RS **Measurable Value with Agile (Ryan Shriver - 2009)**
Use of Evo Requirements and Prioritizing principles

www.malotaux.nl/inspections

Inspection pages

Evolutionary Planning

Producing even more in less time

www.malotaux.nl/conferences

Niels Malotaux

N R Malotaux
Consultancy

+31 655 753 604 niels@malotaux.nl www.malotaux.nl