Simplicity by Design

Niels Malotaux

www.malotaux.nl/conferences

Defining

- **Simple, simplicity**
  - Easy to understand or do
  - Uncomplicated in form or design
  - Question: *how easy, how uncomplicated*
- **Complex, complicated**
  - Involving many different and confusing aspects
- **Simplistic**
  - Treating complex issues and problems as if they were much simpler than they really are

If you think something is difficult, you didn't get it yet

(Once you know, it's not difficult anymore)

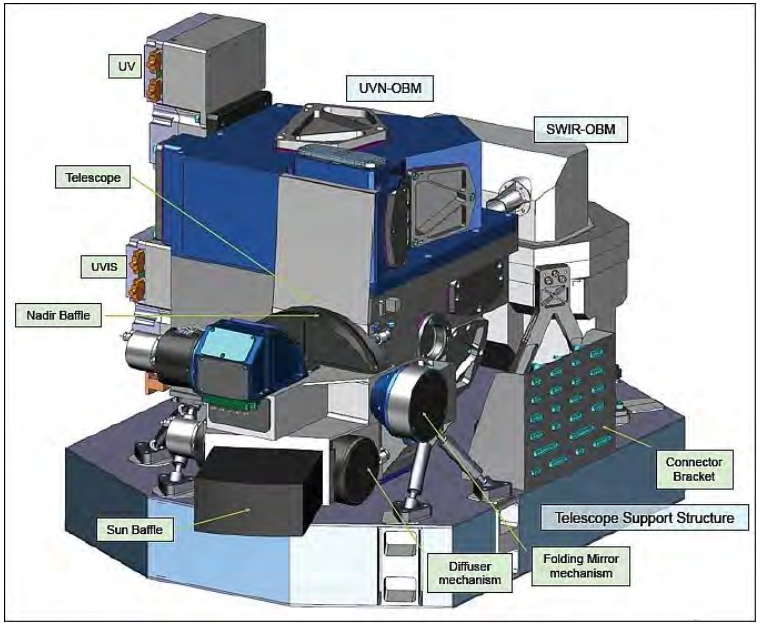
Evo Planning (TaskCycle – DeliveryCycle – TimeLine) is simple

- Still, many counter-intuitive elements make it difficult to start
- James
  - Deceptively Simple
- Do it 5 weeks for me

- Philips HQ
  - How can simple solutions solve difficult problems ?
  - Well: will difficult solutions give better results ?

- 2002 NATO conference on Evolutionary Development
  - We develop the Eurofighter. Surely these simple approaches do not apply to a project of 1000s of people, taking years and years

Earth Observation Satellite



- Very experienced Systems Engineers
- They use quantified requirements routinely
- They don't know exactly where they'll end up
- 6 year pure waterfall project (imposed by ESA)
- Only problem: They missed all deadlines
- 9 weeks later: They haven't missed any deadline since
- Recently: delivered 1 day early (instead of 1 year late)
- Savings: some 40 man-year (about €6M)
- How did they do that ?

Developing the problem is half the solution !

- Meeting with sub-contractors in three weeks
- Many documents to review
- Impossible deadline

- How many documents to review ?
- How much time per document ?

Simple exercise		
	per doc	hr
4 heavy	15	60
3 easy	2	6
	total	66
other work		33
	total	99

available	2 x 26	52

- Some suggestions …
- Result: well reviewed, great meeting, everyone satisfied

- Showing planning and scheduling as a *design problem* makes it simple for engineers

Simple question that opens a can of worms

- **What should you have achieved by the end of the week ?**

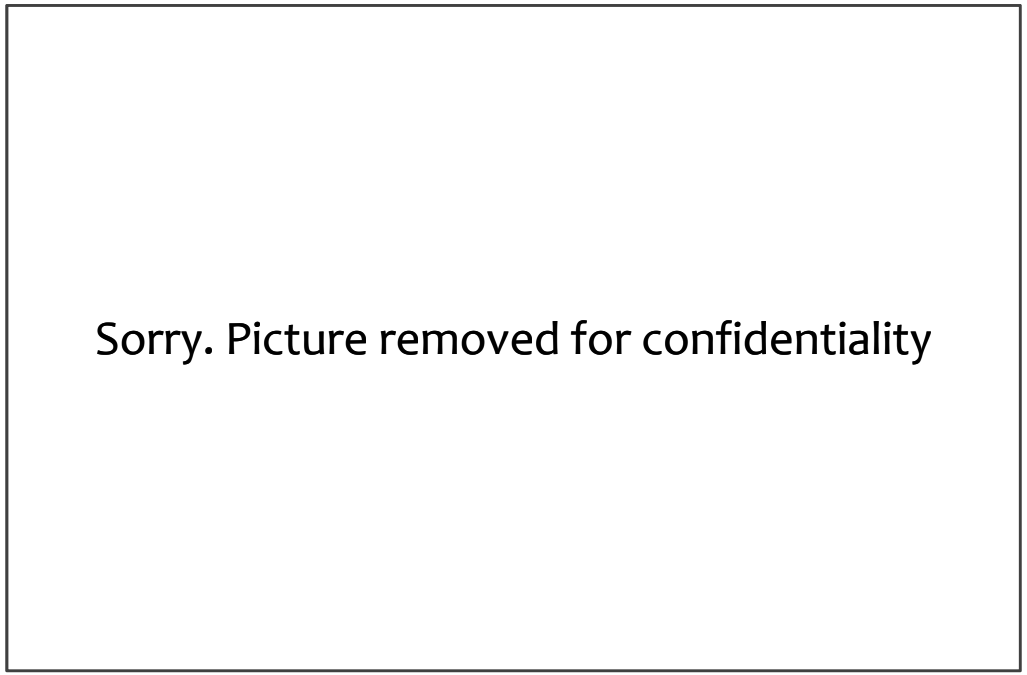
and

- **What do you have to do to achieve that ?**

Transparency makes keeping overview simpler

Not hidden
in a tool
in the computer

Using a large whiteboard



I usually miss design

- **What is design ?**
  - Fashion design
  - Graphic design
  - User interface design

- **I mean**
  - Representation of the structure of a solution in a way that is easily understood
  - How easy?
    - A year later it enables you to be up and running within one day

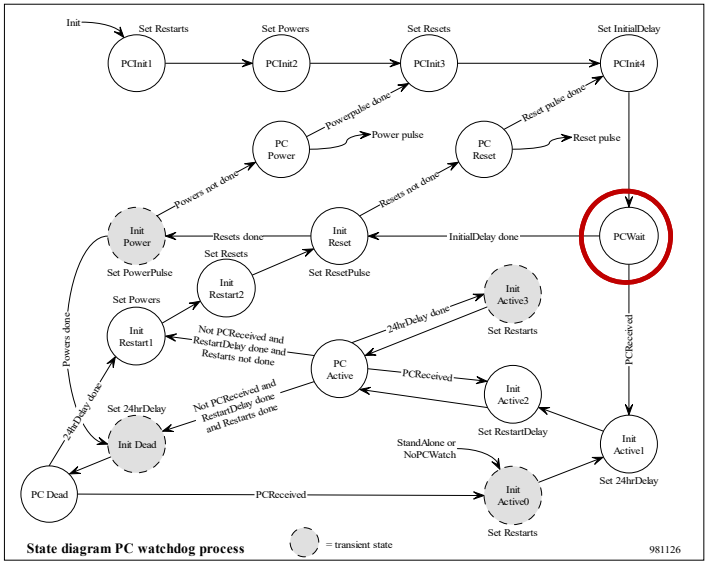
A design should be understood ‘at once’

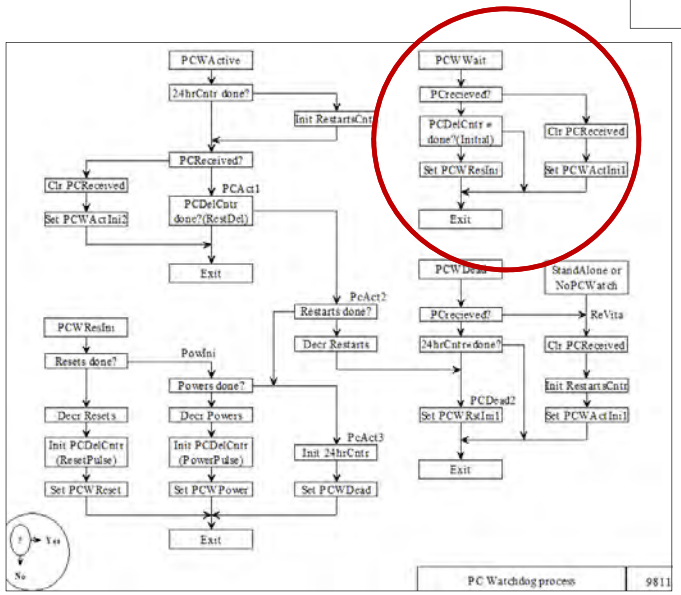
- There are many ways to represent a design
  - Text ?
  - Text and pictures
  - A picture says more than a 1000 words
- A design should be understood ‘at once’ (intended readership)
- Therefore it shouldn’t be complicated
- It should be simple:
  Easily understood or done; presenting no difficulty

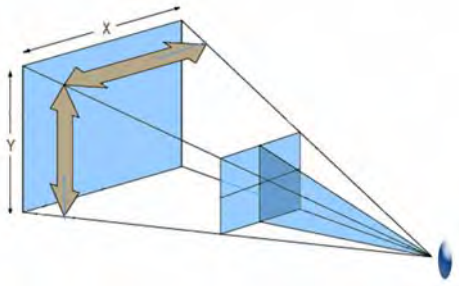
If you ask for a review of software

- **What do you get** (if you ever do) **?**
- **What should we ask for ?**
  - May I review the design first ?
  - If no design: reconstruct the design

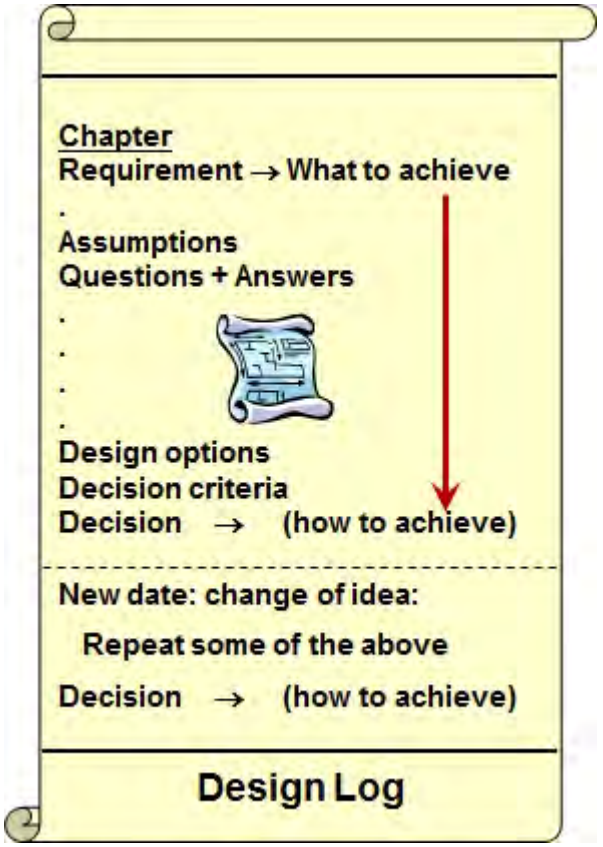
```
        MovLW   WaitPC          ; Select next phase
;       MovWF   PCPhase         ; (See EndPCX)
        Goto    EndPCX          ; Exit PC
;
;*****************************************************************
; Phase Restart init1 PCW
;
PCRIn1  Call    EEtoPCR         ; Init powers counter
        MovLW   RIn2PC          ; Select next phase
;       MovWF   PCPhase         ; (See EndPCX)
        Goto    EndPCX          ; Exit PC
;
;*****************************************************************
; Phase Restart init2 PCW
;
PCRIn2  Call    EEtoPCR         ; Init resets counter
        MovLW   RelnPC          ; Select next phase
;       MovWF   PCPhase         ; (See EndPCX)
        Goto    EndPCX          ; Exit PC
;
;*****************************************************************
; Phase Active init 1 PCW
;
PCAIn1  Call    Pre24h          ; Init 24h counter
        MovLW   AIn2PC          ; Select next phase
;       MovWF   PCPhase         ; (See EndPCX)
        Goto    EndPCX          ; Exit PC
;
;*****************************************************************
; Phase Wait PCW
;
PCWait  BTFSS   PCStat,PCRecvd  ; PC received?
        Goto    PCWait1         ; Branch if not
        BCF     PCStat,PCRecvd  ; Acknowledge PC received
        MovLW   AIn1PC          ; Select next phase
;       MovWF   PCPhase         ; (See EndPCX)
        Goto    EndPCX          ; Exit PC
;
PCWait1 MovF    PCDCntr,f       ; Check delay counter (initial delay)
        Skpz                    ; Skip if counter done (=zero)
        Goto    EndPC           ; Exit PC if not yet done
;
        MovLW   RelnPC          ; Select next phase
        MovWF   PCPhase         ; (See EndPCX)
        Goto    EndPCX          ; Exit PC
;
;*****************************************************************
; Phase Reset PCW
;
PCRes   BSF     PCStat,ResPls   ; Reset pulse on
        MovF    PCDCntr,f       ; Check delay counter (reset pulse)
        Skpz                    ; Skip if counter done (=zero)
        Goto    EndPC           ; Exit PC if not yet done
;
        BCF     PCStat,ResPls   ; Reset pulse off
        MovLW   In1PC           ; Select next phase
;       MovWF   PCPhase         ; (See EndPCX)
        Goto    EndPCX          ; Exit PC
;
;*****************************************************************
; Phase Power PCW
```







DesigLog



- Design
- Review
- Code
- Review

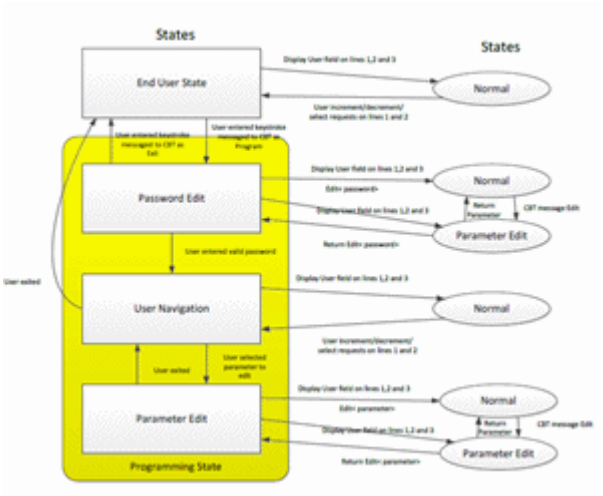
Iterate as needed

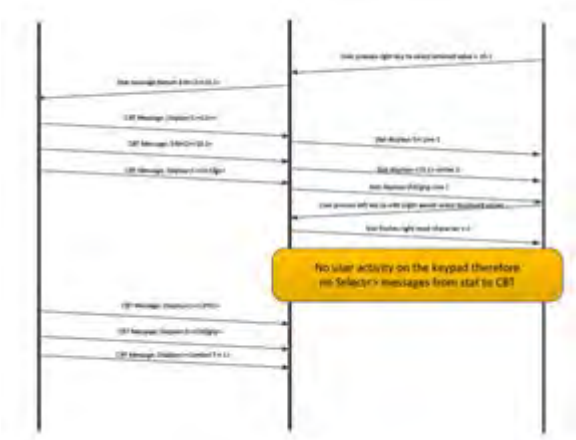
- Test (no questions, no issues)
- If issue in test: no Band-Aid: start all over again:
  Review: What's wrong with the design ?
- Reconstruct the design (if the design description is lacking)
- What happens if you ask "Can I see the DesignLog ?"

Sorry. Picture removed for confidentiality

Sorry. Picture removed for confidentiality

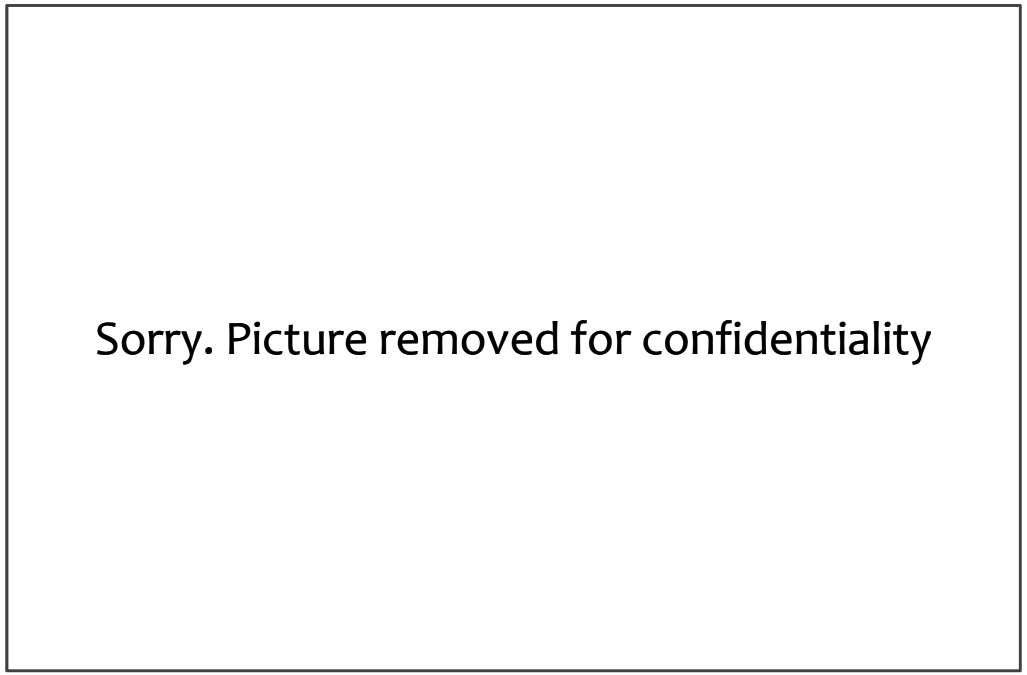
Sorry. Picture removed for confidentiality



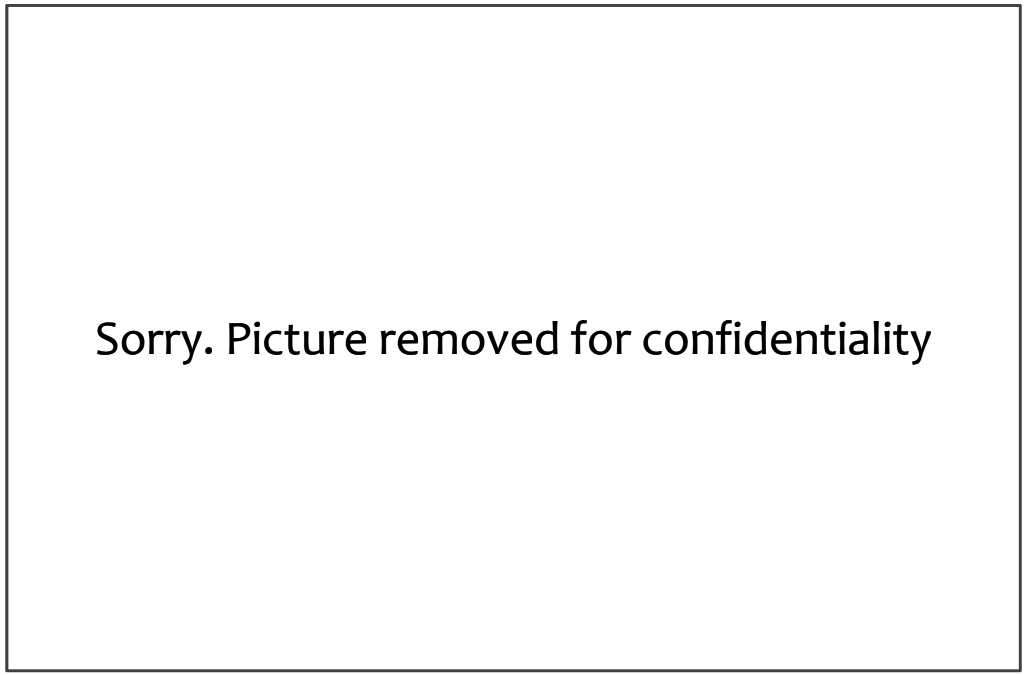


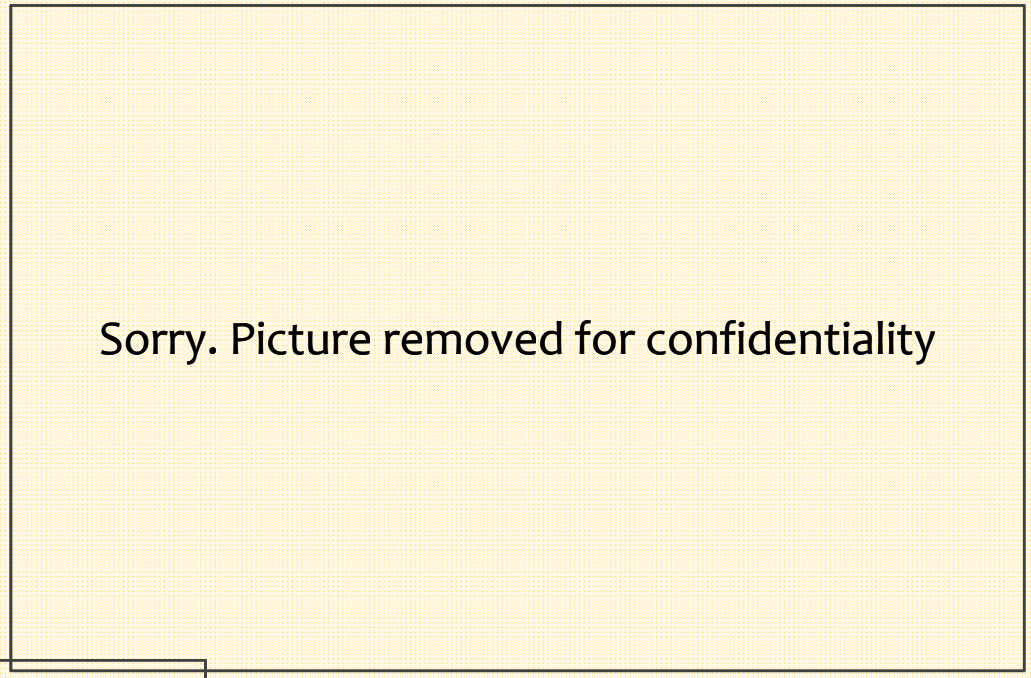
Choose the appropriate design

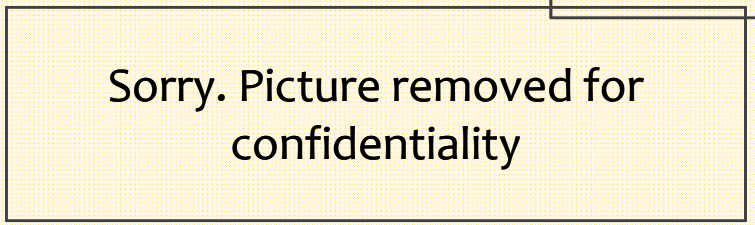
~25 pages documentation condensed into one page



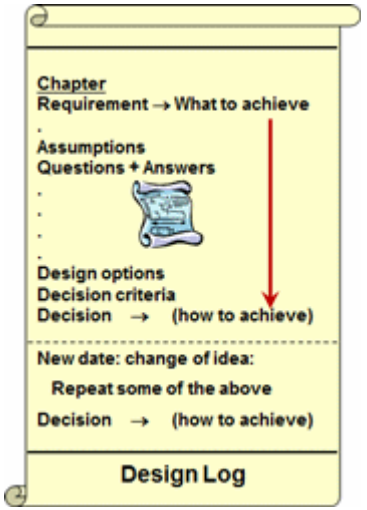
Interface program flow example




Sorry. Picture removed for confidentiality


Sorry. Picture removed for confidentiality

In the pub

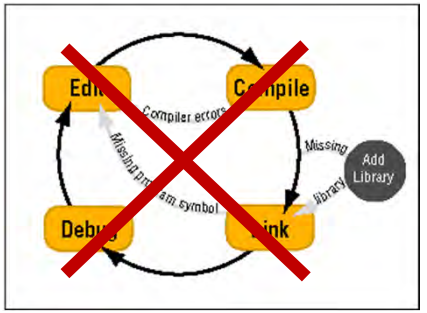


James:

Niels, this is Louise

Louise, this is Niels, who taught me about DesignLogging

Tell what happened



Louise:

- *We had only 7 days to finish some software*
- *We were working hard, coding, testing, coding, testing*
- *James said we should stop coding and go back to the design*
- *"We don't have time !" - "We've only 7 days !"*
- *James insisted*
- *We designed, found the problem, corrected it, cleaned up the mess*
- *Done in less than 7 days*
- *Thank you!*

What James told me recently

- Actually, two features were delivered and deployed
  - One that was design and code reviewed had no issues after deployment
  - Other one, was the source of quite a few defects.
- Furthermore, the final review of the design caused a complete redesign, which was then implemented

- In summary, this success has proved instrumental in buy-in for DesignLogs which are now embedded in the development process

Bottom line

- **Design – review until happy**
- **Implement – design until happy**
- **Testing doesn't find issues**

Proper design makes success simpler

Edsger Dijkstra

Simplicity is a prerequisite for reliability