goto; conference aarhus

Management myths: are we getting any better at this ?

Dave Clack

INTRODUCTION

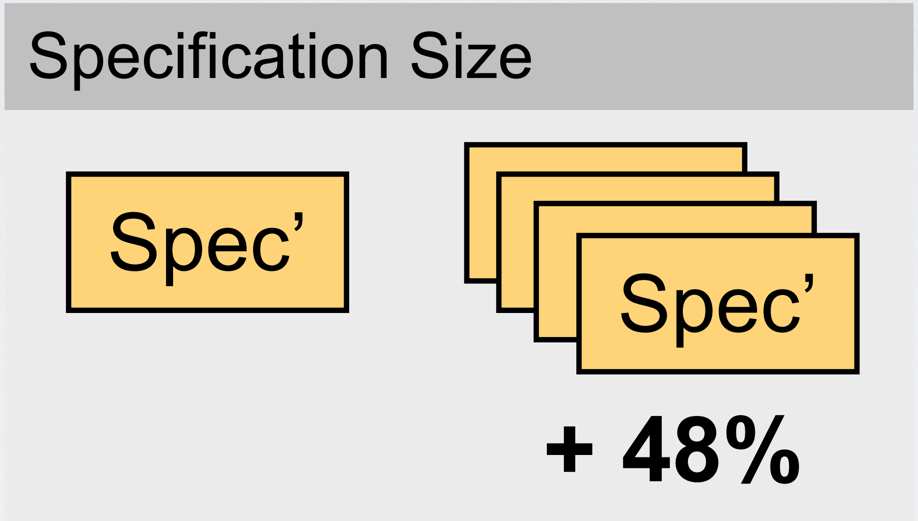
- Hi !
- Why am I talking today ?
- Mythbusting: Ceremony vs. Discipline

MYTHS ?

- we can estimate
- we can measure software size
- we can measure productivity
- pairing is expensive
- individual incentives work
- agile is high risk
- (a note on contracts)

PREMISE: WE CAN ESTIMATE

(WELL ENOUGH TO MAKE GOOD BUSINESS DECISIONS)



Specification Size

Spec'

Spec'

+ 48%

Magne Jørgensen, Stein Grimstad, Simula Research Laboratory, 2006

PREMISE: WE CAN ESTIMATE

(WELL ENOUGH TO MAKE GOOD BUSINESS DECISIONS)

Irrelevant Information

Spec’

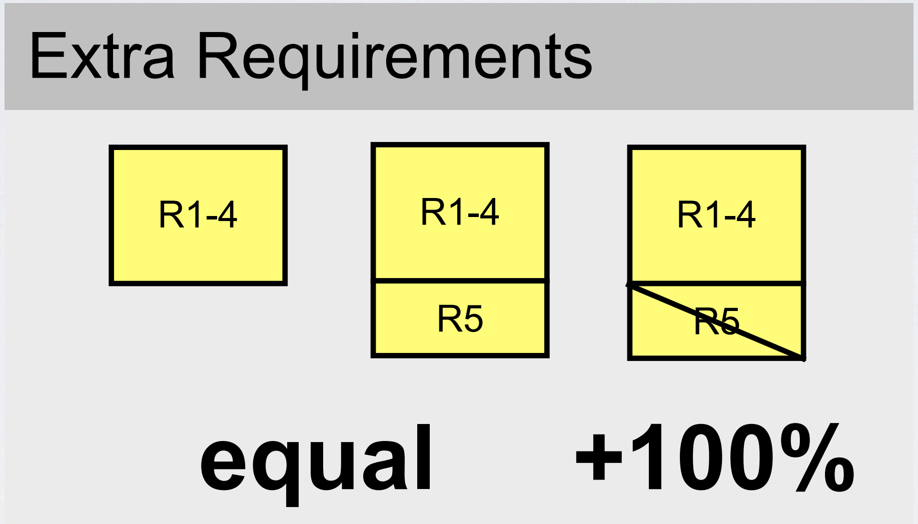
Spec’

+ 95%

Magne Jørgensen, Stein Grimstad, Simula Research Laboratory, 2006

PREMISE: WE CAN ESTIMATE

(WELL ENOUGH TO MAKE GOOD BUSINESS DECISIONS)



Extra Requirements

R1-4

R1-4
R5

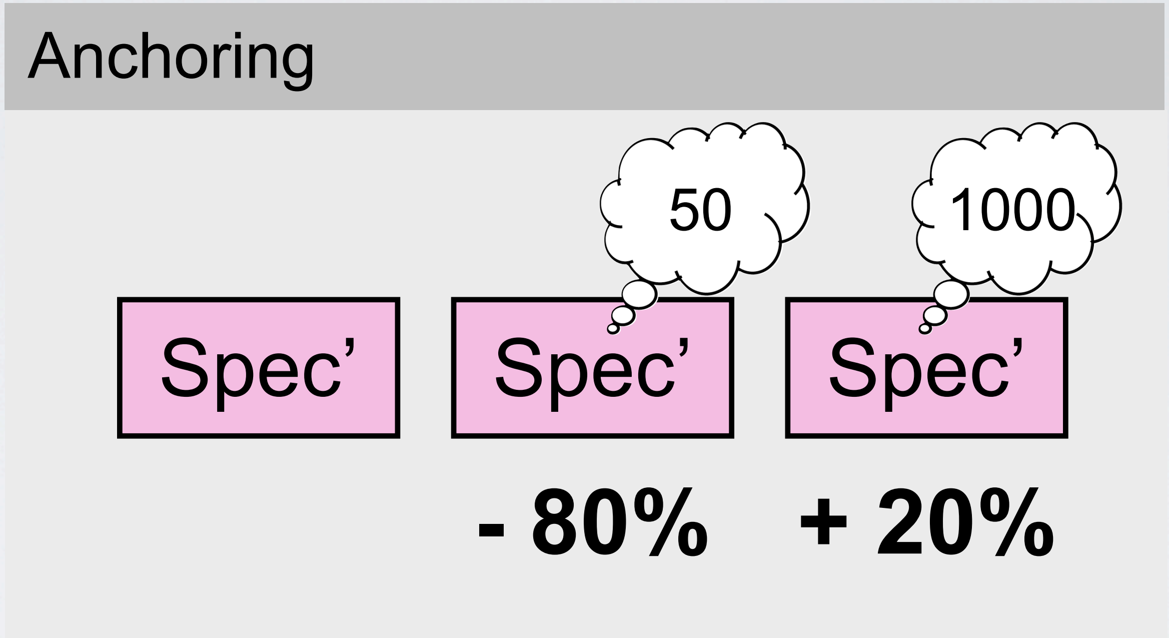
R1-4
~~R5~~

equal **+100%**

Magne Jørgensen, Stein Grimstad, Simula Research Laboratory, 2006

PREMISE: WE CAN ESTIMATE

(WELL ENOUGH TO MAKE GOOD BUSINESS DECISIONS)



Anchoring

Spec'

Spec'

50

- 80%

Spec'

1000

+ 20%

Magne Jørgensen, Stein Grimstad, Simula Research Laboratory, 2006

PREMISE: WE CAN ESTIMATE

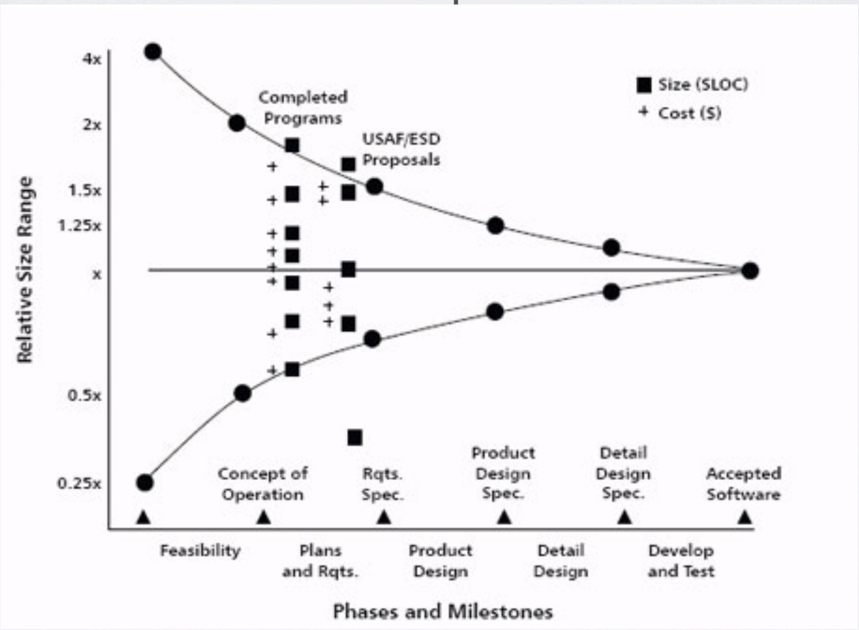
(WELL ENOUGH TO MAKE GOOD BUSINESS DECISIONS)

- Specification Size
- Irrelevant Information
- Complexity
- Anchoring

PREMISE: WE CAN ESTIMATE

(WELL ENOUGH TO MAKE GOOD BUSINESS DECISIONS)

- So we use relative estimates
- & Boehm's Estimate trumpet



- good enough ?

PREMISE: WE CAN ESTIMATE

(WELL ENOUGH TO MAKE GOOD BUSINESS DECISIONS)

- Most estimation is CEREMONY
- So What:
  - the usual agile and lean stuff
  - honesty
  - business objectives not features / epics / stories
  - commit when done ?
  - careful about contracts !

PREMISE: WE CAN ESTIMATE

(WELL ENOUGH TO MAKE GOOD BUSINESS DECISIONS)

BUSTED

- Most estimation is CEREMONY
- So What:
  - the usual agile and lean stuff
  - honesty
  - business objectives not features / epics / stories
  - commit when done ?
  - careful about contracts !

PREMISE: SOFTWARE SIZE CAN BE MEASURED

- how much x how complex
- sloc ? object points ? function points ? business value ? bug density ? A vs. B ?
- changes vs new ?
- in code: less is more
- real world: size ~ work

PREMISE: SOFTWARE SIZE CAN BE MEASURED

BUSTED

- how much x how complex
- sloc ? object points ? function points ? business value ? bug density ? A vs. B ?
- changes vs new ?
- in code: less is more
- real world: size ~ work

PREMISE: PRODUCTIVITY IS MEASURABLE

- productivity = output / input
- ... or size / work
- but we can’t measure size

PREMISE: PRODUCTIVITY IS MEASURABLE

BUSTED

- productivity = output / input
- ... or size / work
- but we can't measure size

SO WHAT ?

- be suspicious
  - XYZ tool / process increases productivity by N%
  - outsourcing / insourcing reorganisation saves £X
  - velocity increased by M%

- mostly decomposes to unit price savings
- is the work really comparable ?

SO WHAT ?

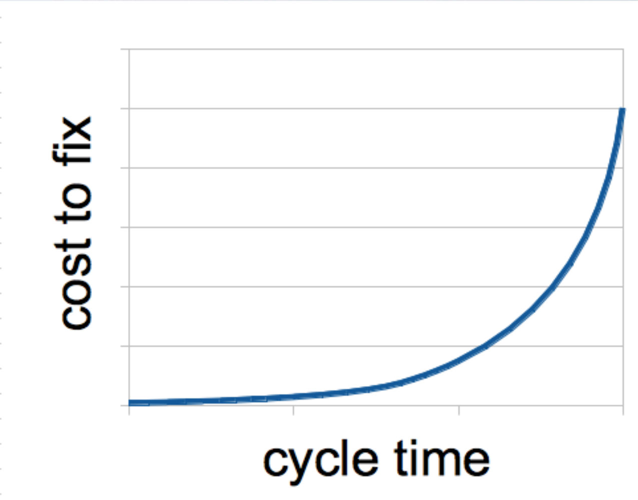
- Most software metrics are bad proxies
- doesn’t mean stop using them
- review / debate / **action**

PREMISE: PAIRING IS EXPENSIVE

(COMPARED TO SOLO PROGRAMMING)

2x

vs.



- 2x creation cost vs. saved downstream cost
- visible & planned vs. hidden & unplanned
- traditional vs. agile/lean

PREMISE: PAIRING IS EXPENSIVE

(COMPARED TO SOLO PROGRAMMING)

- We have a meta study - COOL !
- Pairing is faster, better and more expensive
- particularly for complex tasks

PREMISE: PAIRING IS EXPENSIVE

(COMPARED TO SOLO PROGRAMMING)

- BUT ...
- mostly synthetic studies
- no downstream costs
    - business continuity
    - avoided waste
    - better maintainability
    - fewer late defects

PREMISE: PAIRING IS EXPENSIVE

(COMPARED TO SOLO PROGRAMMING)

IT DEPENDS

- BUT ...
- mostly synthetic studies
- no downstream costs
  - business continuity
  - avoided waste
  - better maintainability
  - fewer late defects

PREMISE: INDIVIDUAL INCENTIVES WORK

- we work and deliver as a team
- pairing and rotation
- collective code ownership
- team building

- review processes seek differences
- we reward individually

PREMISE: INDIVIDUAL INCENTIVES WORK

FAIL

- we work and deliver as a team
- pairing and rotation
- collective code ownership
- team building

- review processes seek differences
- we reward individually

PREMISE: INDIVIDUAL INCENTIVES WORK

- How about for commitment based incentives ?

PREMISE: AGILE/LEAN IS HIGH RISK (VS. TRADITIONAL APPROACHES)

- ie
- Gates, Reviews, Reports, Project Managers and Segregation of Responsibilities are safer
- than
- CI, Automated Testing, Continuous Deployment, systems controls, Direct Business Control

PREMISE: AGILE/LEAN IS HIGH RISK (VS. TRADITIONAL APPROACHES)

- knowing quality now vs. in weeks
- separation of duties vs. automated security
- direct, immediate business control vs. delayed proxies
- live tracking and planning vs. reporting assets / review
- knowledge in several heads vs. paper and silos
- improve as you go vs. sometimes at the end
- self organisation vs. project management

PREMISE: AGILE/LEAN IS HIGH RISK (VS. TRADITIONAL APPROACHES)

BUSTED

(BUT IT MIGHT NOT LOOK THAT WAY!)

- knowing quality now vs. in weeks
- separation of duties vs. automated security
- direct, immediate business control vs. delayed proxies
- live tracking and planning vs. reporting assets / review
- knowledge in several heads vs. paper and silos
- improve as you go vs. sometimes at the end
- self organisation vs. project management

PREMISE: AGILE/LEAN IS HIGH RISK (VS. TRADITIONAL APPROACHES)

BUSTED

(BUT IT MIGHT NOT LOOK THAT WAY!)

- agile / lean appears chaotic
- democracy is noisy
- every issue / refactor / mistake is visible
- nowhere to hide (tech and business)
- live planning & working to business objectives (not spec's) is a tough, mental switch
- transparency means we need to know why we work as we do

A NOTE ON CONTRACTS

- Proxy for trust
- inhibitor to doing the right things
- try and set estimate precision too early
- can enforce SDLC
- often win / lose

A NOTE ON CONTRACTS

- Option 1: Play the Game
  - agile stuff: regular acceptance, small deliverables
  - set fixed prices
  - pay for time
  - pay for points
  - change control rip off

- Option 2: Get Creative
  - pay if happy
  - cancel the project every iteration
  - share of value
  - supplier pays and leases
  - risk profit

MYTHS ?

- we can estimate
- we can measure software size
- we can measure productivity
- pairing is expensive
- individual incentives work
- agile is high risk

MYTHS ?

- we can estimate **BUSTED**
- we can measure software size **BUSTED**
- we can measure productivity **BUSTED**
- pairing is expensive **IT DEPENDS**
- individual incentives work **FAIL**
- agile is high risk **BUSTED**