



The Prince of Scrum

Using Scrum in a Prince 2 Environment



the reality of software development

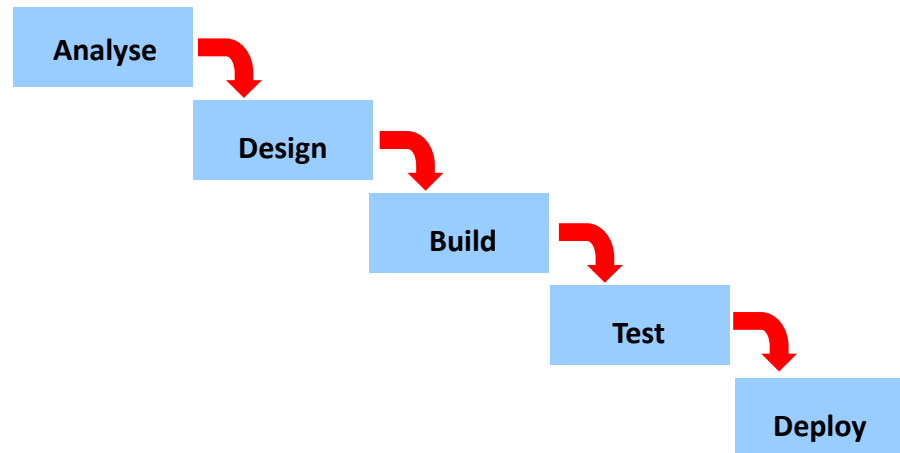
- 57% of projects fail due to poor project scoping
- 35% fail due to buggy software
- 30% fail due to unattainable business requirements.

Forrester Research (2004)



the waterfall methodology

- defined Process
- each stage completed and signed off before the next commences.
- big 'up-front' analysis/design fixes detail in requirements/solution
- communication through documentation





which often leads to

- extra/unused features (overproduction)
- partially developed work not released to production (inventory)
- intermediate/unused artifacts (extra processing)
- seeking information (motion)
- escaped defects not caught by tests/reviews (defects)
- waiting (including customer waiting)
- handoffs (transportation)

Source: Mary & Tom Poppendieck, <http://www.poppendieck.com>

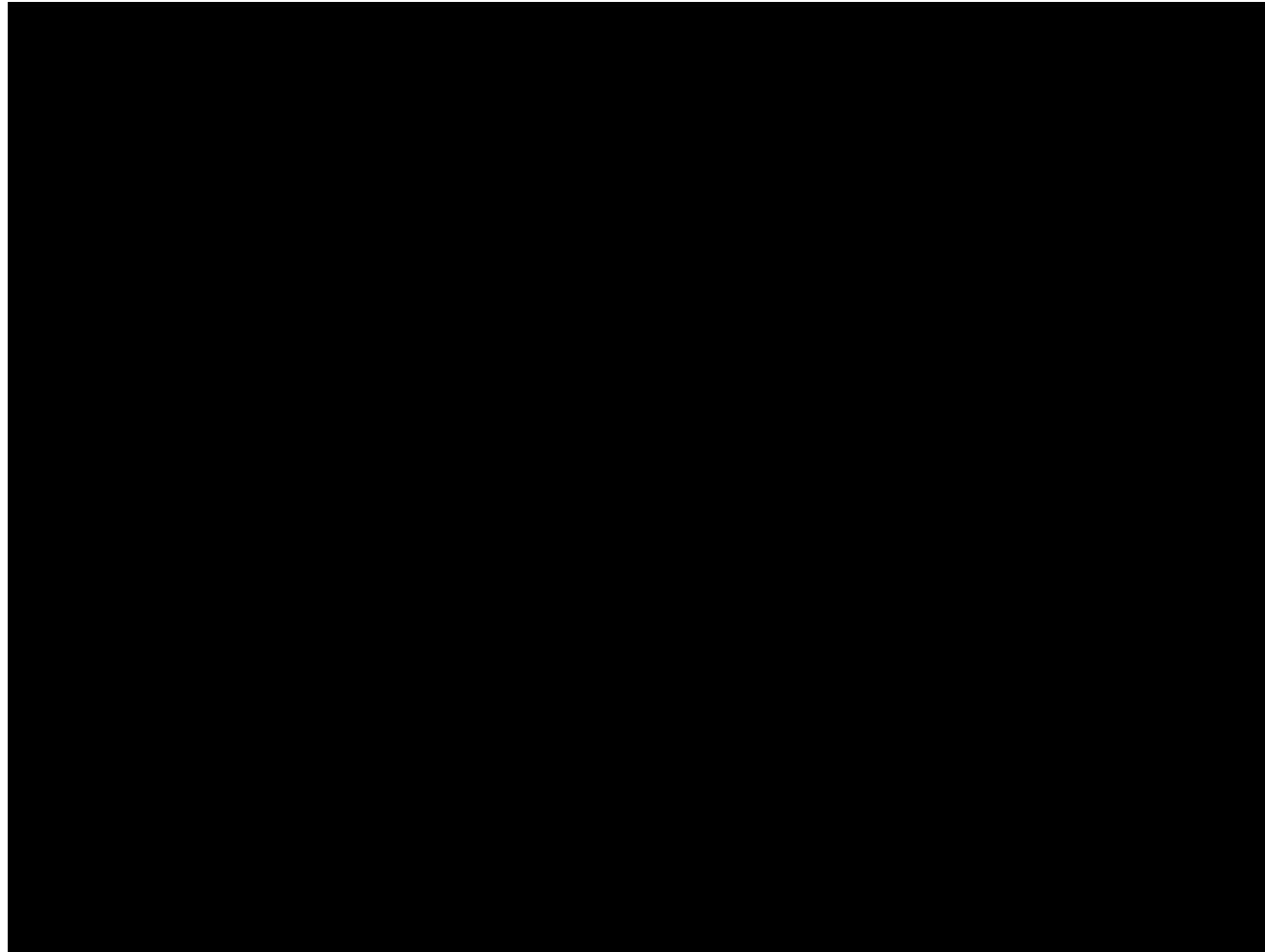


in the real world

- requirements change
- customers never know exactly what they need
- requirements are incomplete
- people rarely understand requirements from the beginning
- people make mistakes
- up-front design is too much and very expensive
- for most, apart from the very smallest systems it is hard to successfully design everything in advance
- defined processes don't work in complex environments

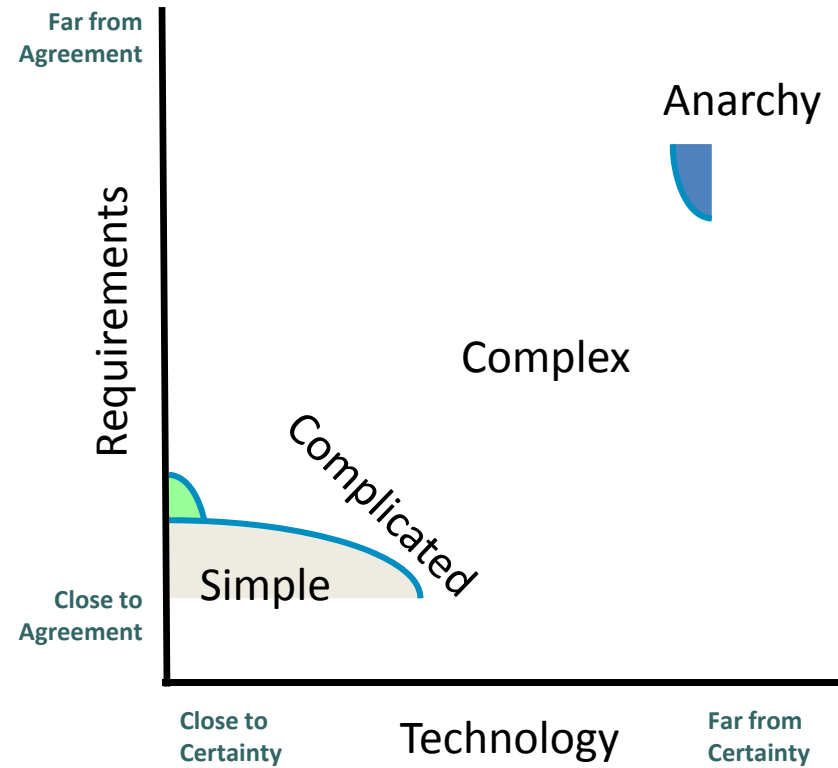


the users' perspective



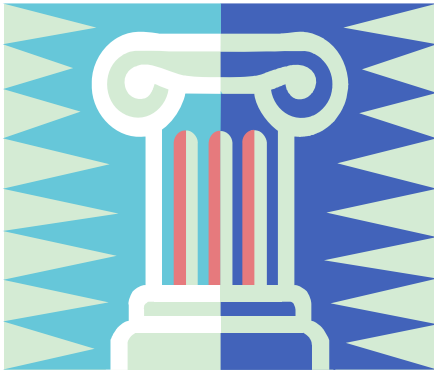


defined process vs empirical process

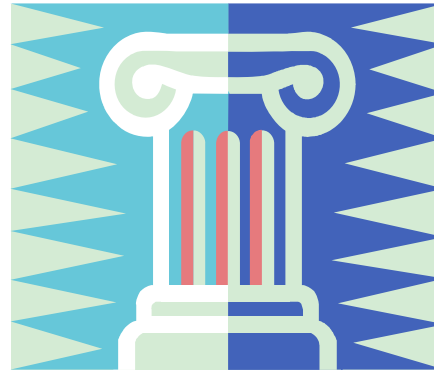




3 pillars of empirical process control



Transparency



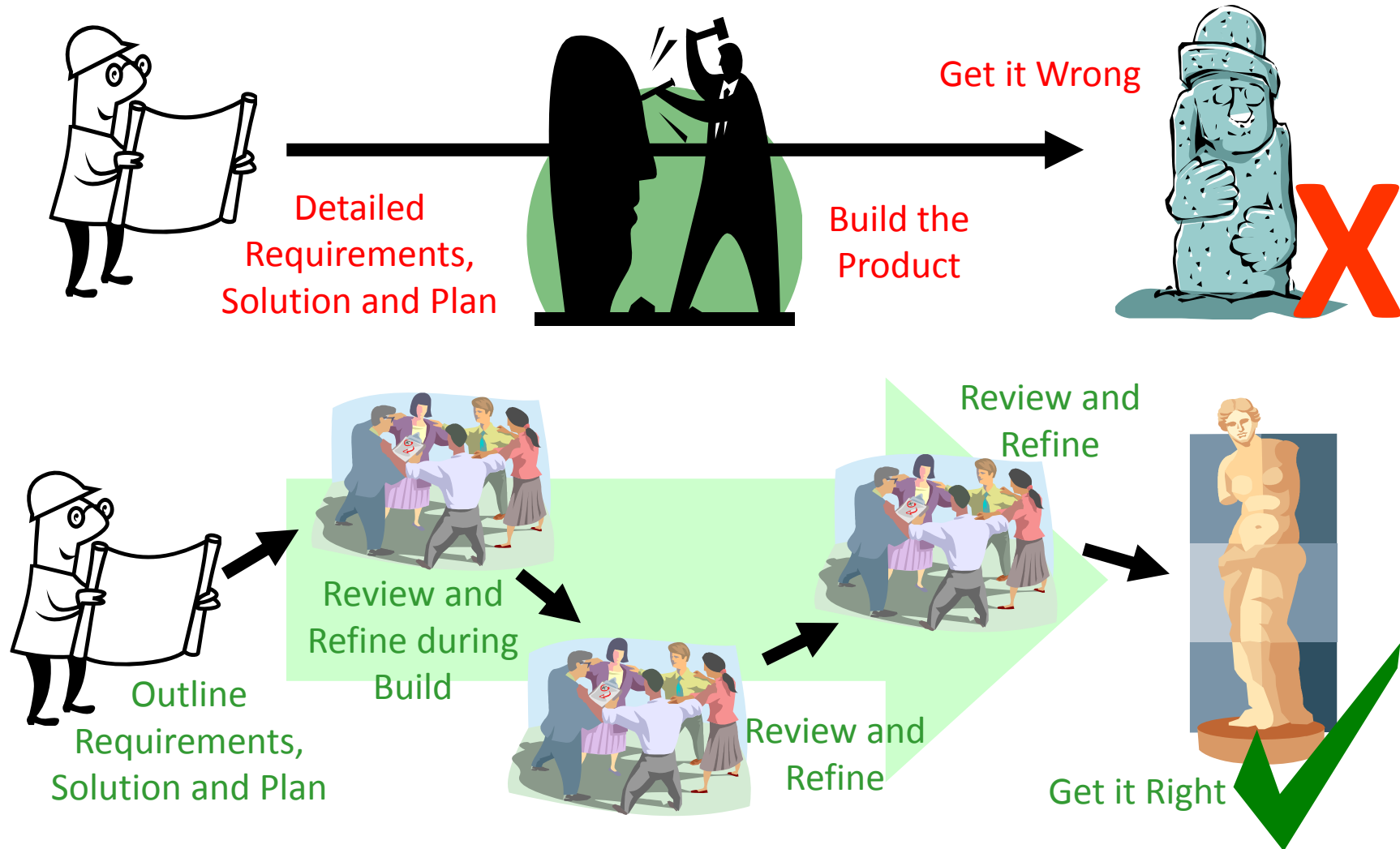
Inspection



Adaptation



empirical (agile) product development





agile myths

- implementing agile is easy
- agile gives instant benefit
- agile means no documentation
- agile means 'hacking'
- agile means poor quality
- agile is a 'silver bullet'
- you just need to read a book
- agile only relates to software
- agile is new
- traditional methods don't work
- agile replaces everything that has gone before
- agile means 'JFDI'



manifesto for agile software development

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.



what is agile?

- as the manifesto says, it's a set of values
- supported by a set of good software engineering practices
- valuing people
- these values and practices together drive behaviour
- the aim of agile enablement is to change people's behaviours to improve software delivery



what are the agile practices?

- short cycle time
 - allows rapid inspection
 - early delivery of value
- plan only for the next cycle in detail
 - deliver value and accommodate change
- build only what you need
 - unnecessary features cost money which they will never pay back
- plan based on value
 - deliver high value first for early return on investment
- user stories
 - common language and business metaphor
- close collaboration
 - make small decisions often, not big decisions



what are the agile practices?

- delivery team pick their own tools
 - would you insist on the tools a brain surgeon could use to operate on you?
- constant review
- automated testing
 - when we change things, we know we haven't broken anything
 - focus on new features, not spending lots of time regression testing
- continuous integration
- evolutionary design
- test driven development
 - focus on business need
 - simplified code



what are the agile practices?

- refactoring
 - keep the code clean and tidy to reduce the cost of change
- start simple and build iteratively
 - deliver high business value early and start benefits realisation
- small releases
 - little and often to reduce risk and deliver value sooner
- inspect and adapt
 - expose issues with ways of working/relationships etc and fix them
- drive behavioural change
 - focus on doing the right thing at the right time
 - continuous improvement
- have fun!

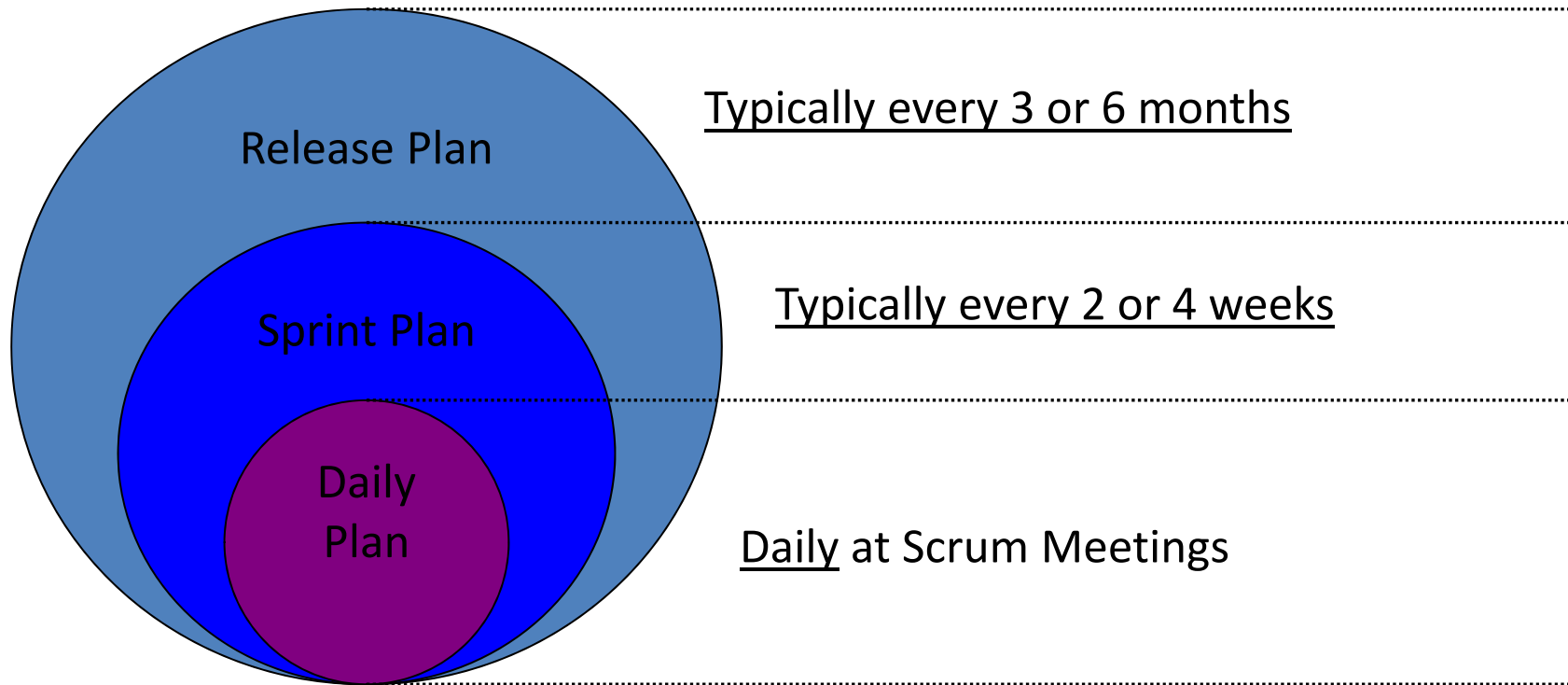


these practices lead to

- shared ownership by the team
- collaborative working
- openness
- eliminating waste
- delivering value
- continuous improvement



the scrum planning cycles





user stories

- the way requirements are documented
- differing levels of granularity
 - epic
 - theme
 - Story
- level of detail dependent on priority and proximity
- collectively the Product Backlog
- a reminder to talk



user stories

- epics
 - “as a Sales Director, I want an eCommerce platform so that I can sell online”
- themes
 - “as a customer I want a shopping basket so that I can buy multiple items at the same time”
- stories
 - “as a customer I want to be able to add an item to my shopping basket so that I can buy it”



the product backlog

Story	Estimate
As a Sales Director, I want an eCommerce platform so that I can sell online	128
As a customer I want a shopping basket so that I can buy multiple items at the same time	56
As a customer I want to be able to add an item to my shopping basket so that I can buy it	8
As a customer I want to be able to remove an item from my shopping basket so that I don't have to buy it	4



So why agile?

- value focussed
- quality focussed
- rapid feedback and ability to change
- simple code that meets business need
- production quality code delivered early
- constantly evolving working practices
- improved quality

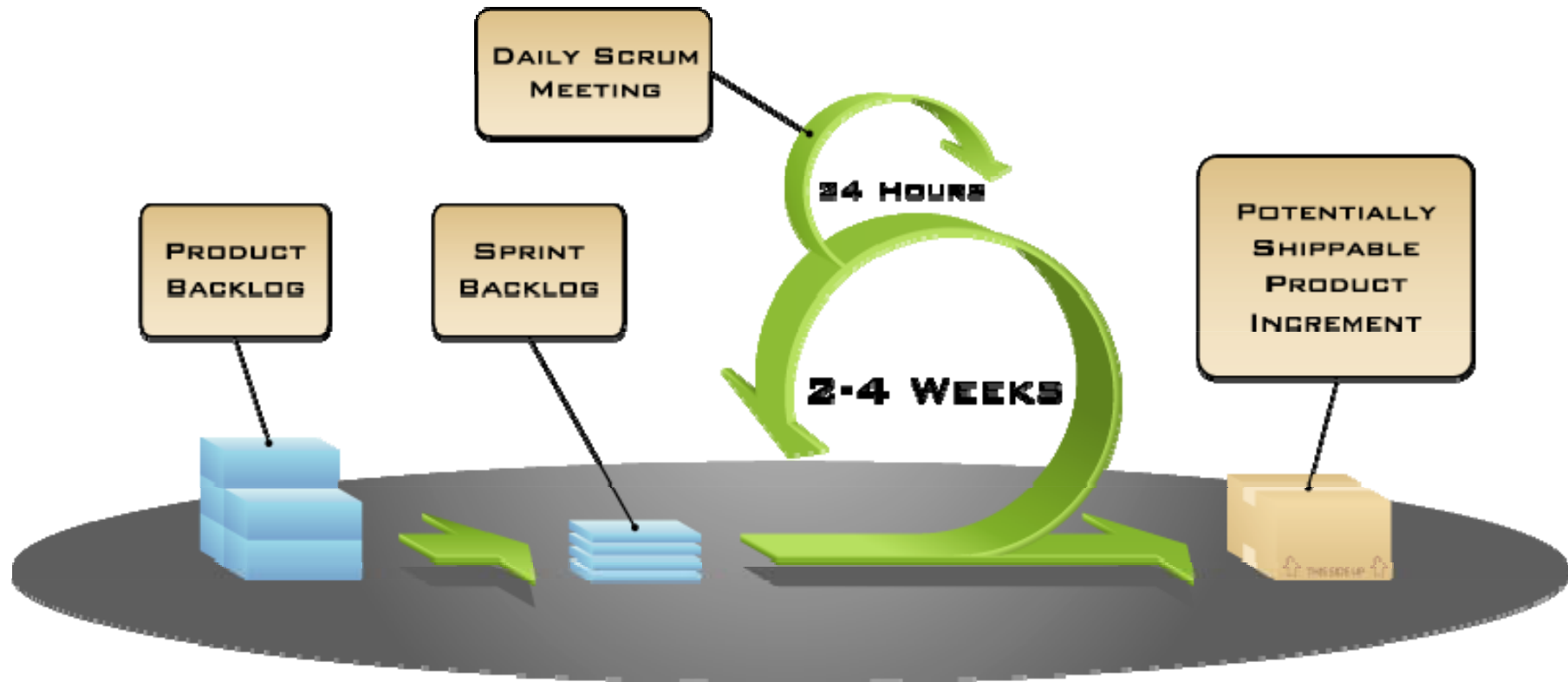


what does it look like in real life?

- prioritise backlog of requirements
- plan delivery for the cycle
- build – test first, code second
- demonstrate each user story when built
- showcase everything built in the cycle
- inspect and adapt working practices



the scrum cycle



COPYRIGHT © 2008, MOUNTAIN COCA™ SOFTWARE

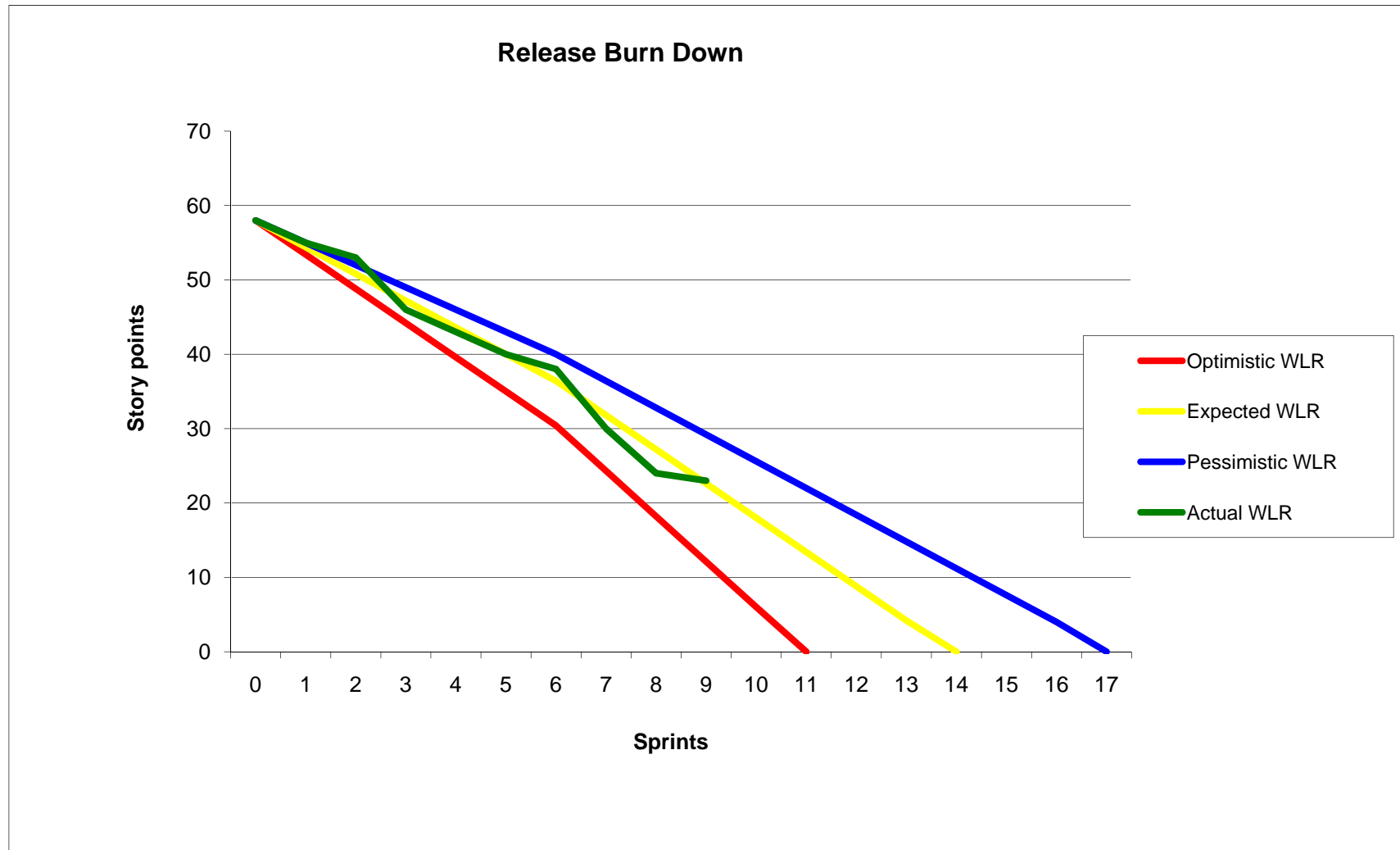


how do we know it's working?

- progress tracked using burn down charts
- multiple types – same format
- sprint burn down shows progress within current sprint
- release burn down shows progress across a number of sprints which form a release
- budget burn down
- use of retrospectives



the burn down chart





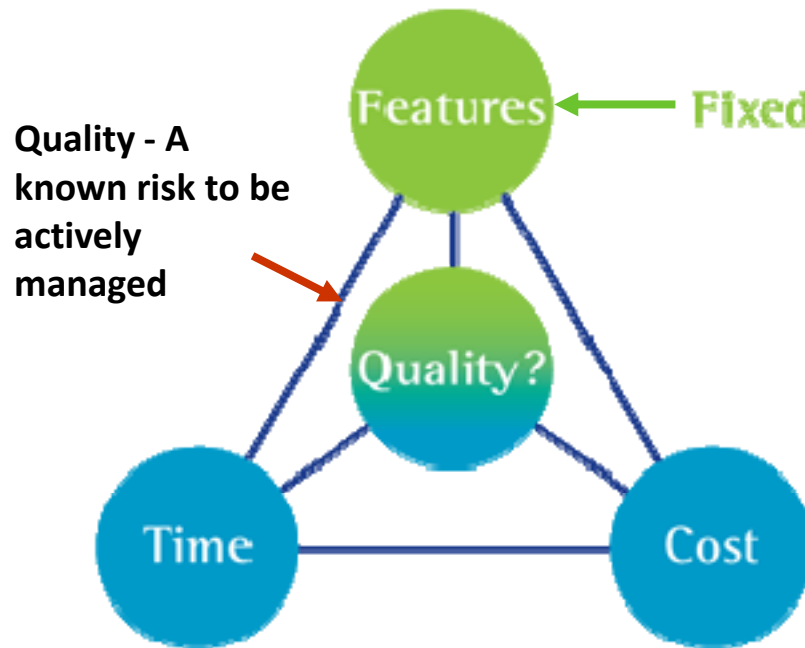
what are the benefits?

- constant feedback loop
- improved quality
- inspection (test) after every code change
- simple code that meets business need
- working code delivered early
- constantly evolving working practices
- cost, schedule and quality fixed – only scope changes

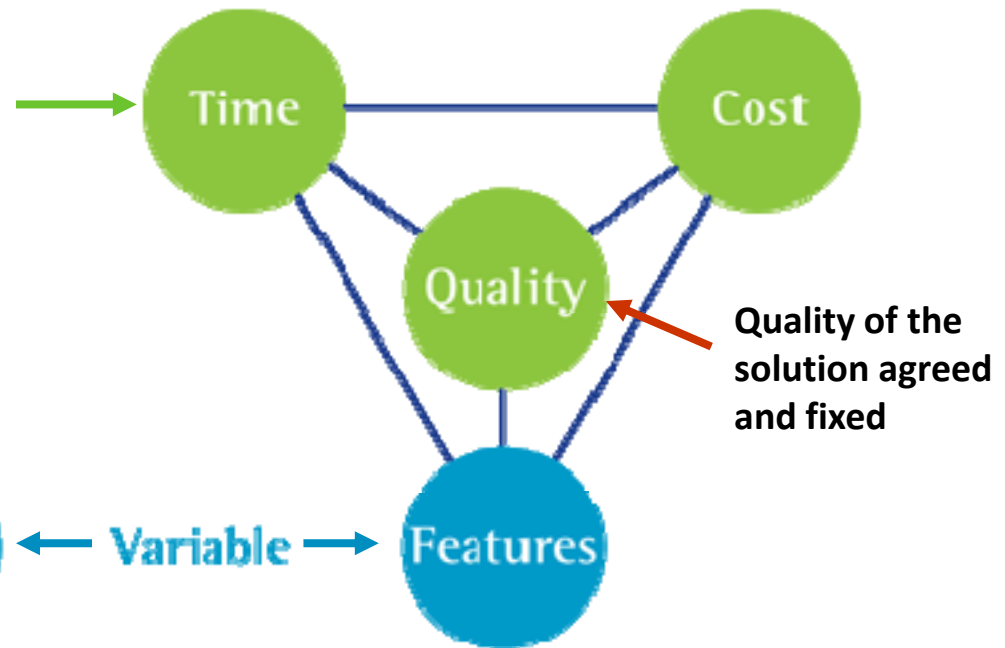


in the world of project constraints

Traditional Approach



Agile Approach





where does agile work?

- FDA-approved, life-critical systems
- satellite-control software
- handheld software
- mobile phones
- network switching applications
- commercial software
- financial applications
- ISO 9001-certified applications
- embedded systems
- 24x7 systems with 99.999% uptime requirement
- the Joint Strike Fighter
- some of the largest applications in use



who's using agile successfully?

- Microsoft
- Yahoo
- Google
- Lockheed Martin
- Time Warner
- Turner Broadcasting
- Siemens
- Nokia
- Capital One
- BBC
- BSkyB



but we use Prince 2

- agile methods can be used alongside Prince 2
- several areas of overlap
- starting up a project (SU)
- directing a project (DP)
- initiating a project (IP)
- controlling a stage (CS)
- managing stage boundaries (SB)
- managing product delivery (MP)
- closing a project (CP)
- planning (PL)



starting up a project (SU)

- creating the project management team
 - identify the product owner (executive)
 - identify the scrum master
 - identify any additional project management capability required
- prepare project brief
 - include product vision



directing a project (DP)

- sprint planning determines what is required from the next sprint and commits
 - funding
 - resources
- sprint review authorizes completion of a stage
- sprint review used to communicate with external interested parties
- Final sprint review authorizes project closure
- daily stand ups provide ad-hoc direction



initiating a project (IP)

- form initial product backlog
- use the product backlog to form an initial release plan
- agree initial spring length
- define schedule of scrum ceremonies
 - planning game
 - daily stand up
 - sprint review
 - Retrospective
- define what “done” looks like
- don't forget to refine the business case and risk log



controlling a stage (CS)

- sprint planning ensures
 - the right products are delivered
 - plans are updated
- daily stand up ensures
 - progress is tracked – via burn down charts
 - deviations from plan are managed
 - sprints are aborted if the sprint goal is no longer relevant or can not be achieved
 - issues are captured and dealt with
- sprint review ensures
 - quality is achieved
 - all interested parties are informed
 - the right products have been delivered



managing stage boundaries (SB)

- sprint review assures product owner and project board that the current stage plan is completed
- sprint planning prepares next stage plan
- sprint review and sprint planning allow check for continuing viability of the project
- sprint planning confirms authority to continue
- retrospective records lessons learnt and actions to improve



managing product delivery (MP)

- sprint planning used to negotiate work to be done and plan it
- daily stand ups keep track of progress which is reported through the burn down charts
- close collaboration ensures the work being done is the right work
- sprint review ensures quality is achieved and approves products



closing a project (CP)

- final sprint review authorizes closure of project
- final retrospective covering all sprints identify follow on action recommendations
- final sprint review should also plan any further post-project reviews (benefits realisation)



planning (PL)

- identify products required
- prepare product backlog for each product
- product backlog contains non-functional as well as functional requirements
- dependencies between stories identified and if possible removed
- each product backlog estimated in story points
- release plan prepared for each product



where has agile been used with Prince 2?

- used by AOL for telecoms OSS delivery
- used by BSkyB for
 - content web sites
 - ecommerce
 - customer self service
 - CRM delivery
- used by Fitness First for delivery of global membership system



thank you

Iain McKenna

agile consultant

iain.mckenna@project-success.co.uk

www.project-success.co.uk

+44 (0)7879 631122