

Innovate2011

The Rational Software Conference

11th and 12th of October

Let's **build** a smarter planet.



Software Economics in Practice

Session AS01

Julian Holmes

Co-founder, UPMentors

Julian@UPMentors.com



This session...

Many software delivery projects fail...

- to deliver the value promised
- or achieve the expected results

Walker Royce presented

- Measure Improvement in Software Economics

I will provide my perspective on

- What it takes to make the changes suggested
- What happens in practice as these changes are made



Complexity Increases Uncertainty

My clients don't like uncertainty

- It results in waste and additional cost

They want predictability

- Faster and cheaper would be nice too!

So I encourage them to manage complexity effectively

- Linear / Waterfall approaches fail to do this
- What I encourage is the use of Iterative and Agile practices



Predictability Requires Adaptability

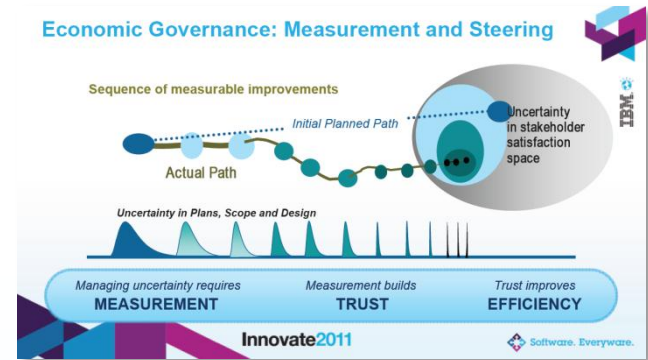
You need a Shared Vision

- But, up-front requirements != actual solution

Optimisation requires measurement

- Iterative practices increase feedback
- Increased feedback develops confidence
- With confidence comes trust

Blindly following a pre-defined path will not deliver the right result



Pivotal Culture Shifts

Continuous Integration of all work

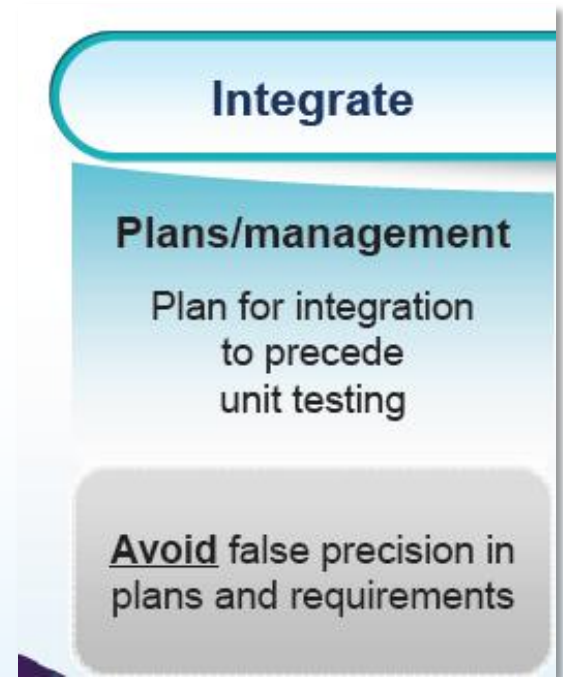
- Demonstrate true progress of working solutions

Avoid creating planning wallpaper

- You can't predict the future

Don't rely on "spec's"

- Early details can be wasted effort
- Template completion is not progress



Pivotal Culture Shifts

Balance risk against demonstrating value

- But risk management adds value too

Maintain a Strategic Architectural view

- Proving architecture decisions with valuable solutions

Regular demonstration is real progress

- Requires a “whole team” effort

Collaborate

Progress measures
Quantify progress trends from the integrated code and test base

Don't attack the easy things first



Pivotal Culture Shifts

Change is inevitable, but must be managed

- Maintaining “velocity” with change is key

Adopt Team Change Management

- Allow change to future work, there’s no “impact”
- But change to “done” work is new work

Constant demonstration validates change

- Increases confidence in what’s being delivered



Optimize

Quality measures
Quantify cost-of-change trends to demonstrate true agility

Don't rely on subjective and speculative measures

IBM
UPM

Silos v Whole Teams Documents v Software

Conventional

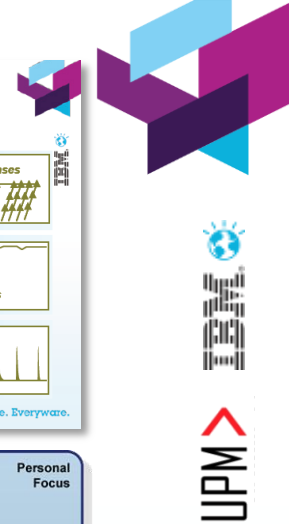
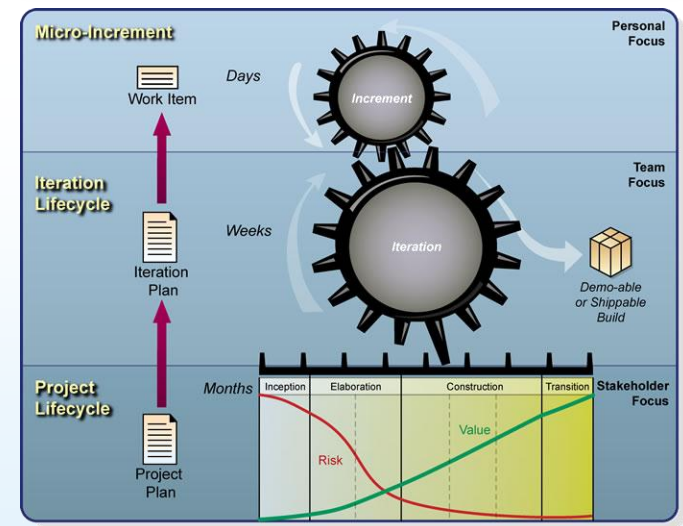
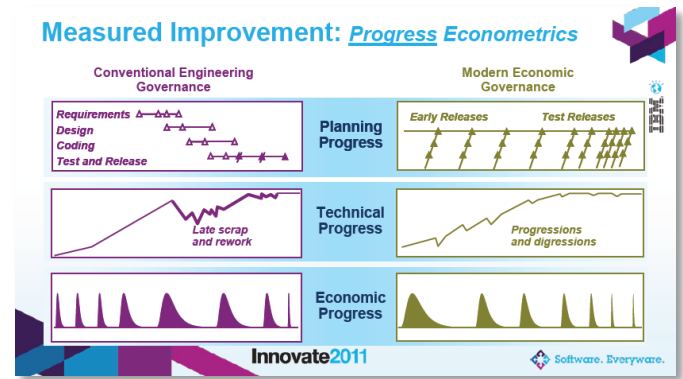
- Speculation, rework and delays

Modern

- Early discovery, consumable solutions

Balance Risk and Value

- Avoid late surprises



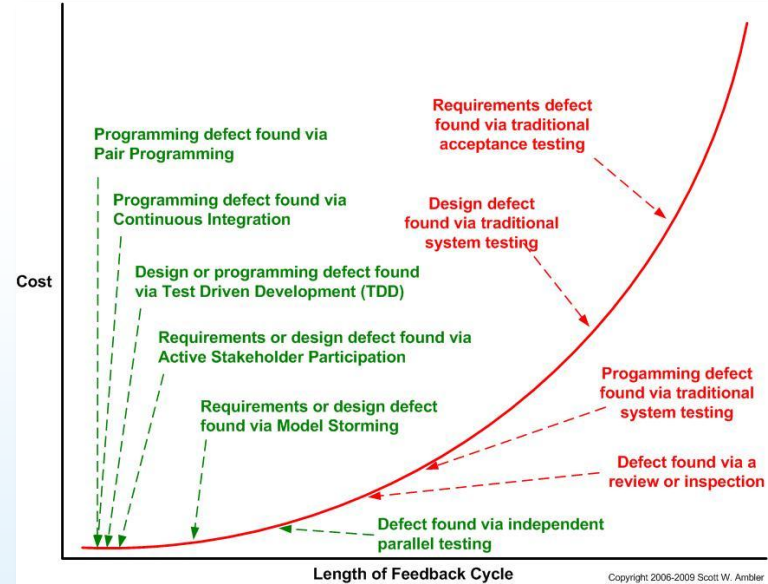
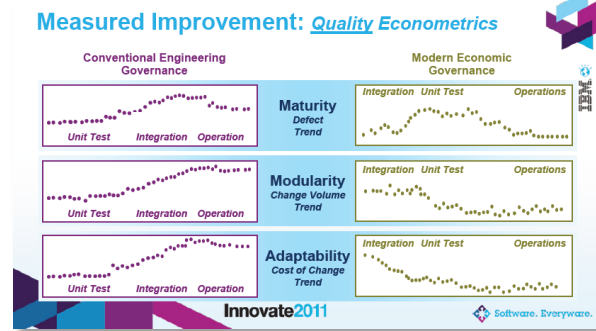
Quality-Driven Activity

Discover issues early

- Test-first
- Non-solo development
- Continuous Integration
- Demonstration

Avoid “technical debt”

- Reduces the cost of change



Complexity ^{Agility} * Collaboration * Automation

Most projects are complex

IBM agility@scale addresses

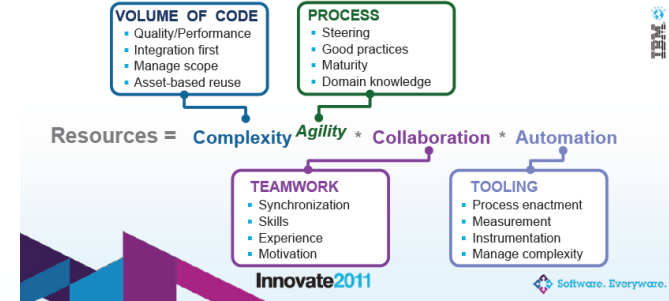
- Process, Teamwork, Tooling

But it's people that are most important

- Agility and Collaboration

They also require significant investment

Improving Software Economics



Innovate2011

 Software. Everywhere.

Improvement ROI

Automation – a foundation

- A simpler investment, quick results

Collaboration tools – an enabler

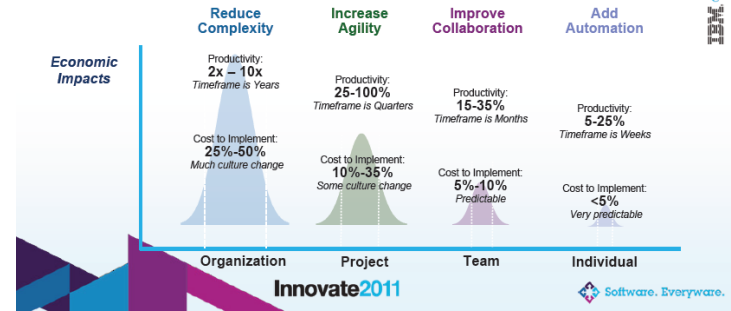
- Increased communication is always beneficial

Agile practices – multiply the returns

- But cultural change takes time, effort and significant investment

The tools bring the horse to water, Agility helps it drink

Productivity Improvement Leverage



“If you play better defense you can play better offense!”

You have to know where you stand

- Initial and continuous measurement is key

People and culture are major components of success

- It can't and won't happen over-night

Agility is key to delivery and operational success

- agility@scale takes more than simple method adoption



Your next steps toward Improving Software Economics



Read Walker's Whitepaper:

- http://walkerroyce.com/PDF/Improving_Software_Economics.pdf

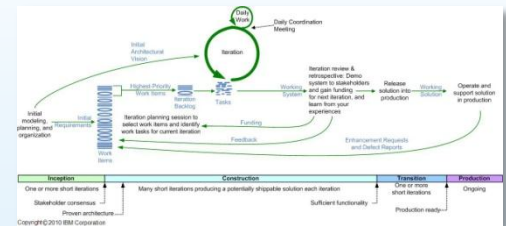
Learn from the experiences of others:

- Attend today's "TS2: Achieving Large Scale Distributed Agile Delivery"

Advance your Agile Education

- Attend tomorrow's "Introduction to [Disciplined Agile Delivery](#)"

Use the experience of others



Innovate2011