pmod

Forward reasoning
or
learning from the past

Where the Agile Manifesto helps

October 2, 2012

Arie van Bennekum

people make the difference

Flow

- Introduction

- Definitions

- Consequences

- Solution

- Conditions

- At the moment

- Conclusion

Introduction, topic

Why many projects are not successful (defined in the early ‘90-ties)?

- The solution does not deliver what the *business needs*

- The solution has a lot of *hindering errors*

- The solution has overall a *poor performance*

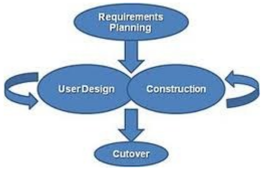
- The solution is *not accepted* by the end user population

- The solution is very *difficult to maintain*

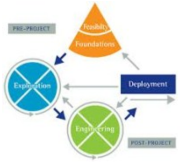
- The project runs *over time* and *over budget*

Introduction, observations

- Rapid Application Development, Scrum, RIPP innovate & improve

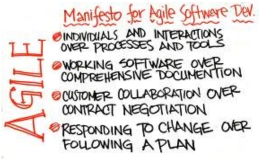


- DSDM tunes and structures improvement



- More innovation

- The Manifesto published



AGILE

Manifesto for Agile Softwere Dev.

- INDIVIDUALS AND INTERACTIONS OVER PROCESSES AND TOOLS
- WORKING SOFTWARE OVER COMPREHENSIVE DOCUMENTION
- CUSTOMER COLLABORATION OVER CONTRACT NEGOTIATION
- RESPONDING TO CHANGE OVER FOLLOWING A PLAN

- Agile gets more and more popular

- Repetition of problems

Did we forget something and can the Agile Manifesto help?

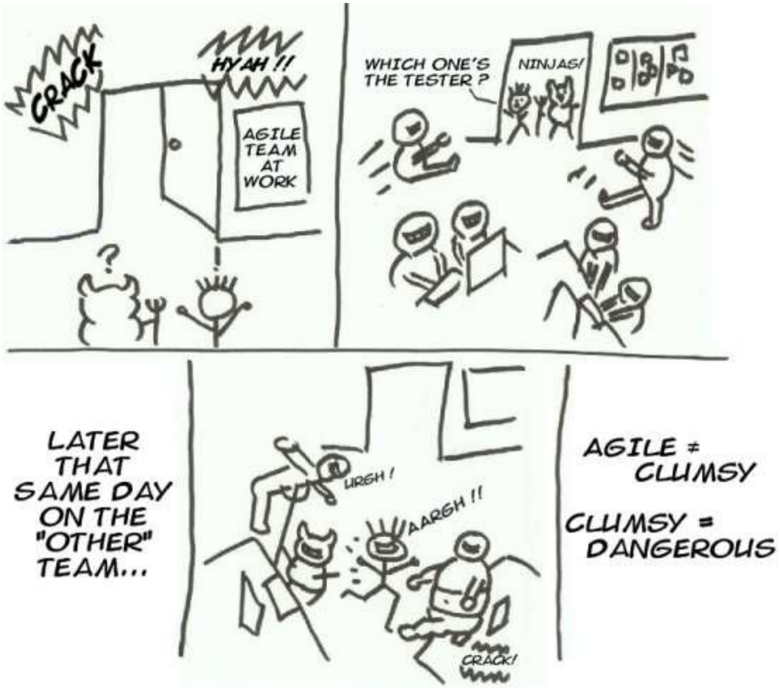
Introduction, personal

- 1987, start as developer
- 1994, switch to Rapid Application Development
- 1994, start as lecturer at the Rotterdam University for Professional Education
- 1997, switch to DSDM, involved as board member and assessor
- 1998, certified as Certified Professional Facilitator and assessor-CPF
- 2001, co-author of the Agile Manifesto
- 2007, involved in the start of the Agile Consortium International (ACI)
- 2010, involvement in ACI increased
  2011, joined People Make the Difference (www.pmtd.nl)
- 2012, elected als chair of the ACI

Definitions, Agile

My definition of Agile

Serving the business by being adaptive
(in stead of “deliver what has been described”)



Definitions, a succeful project

My definition of a successful project

helps the organisation to achieve their (SMART) business objectives as stated for the project within an acceptable time frame and for an acceptable investment

Definitions, “errors”

The errors that re-occur….

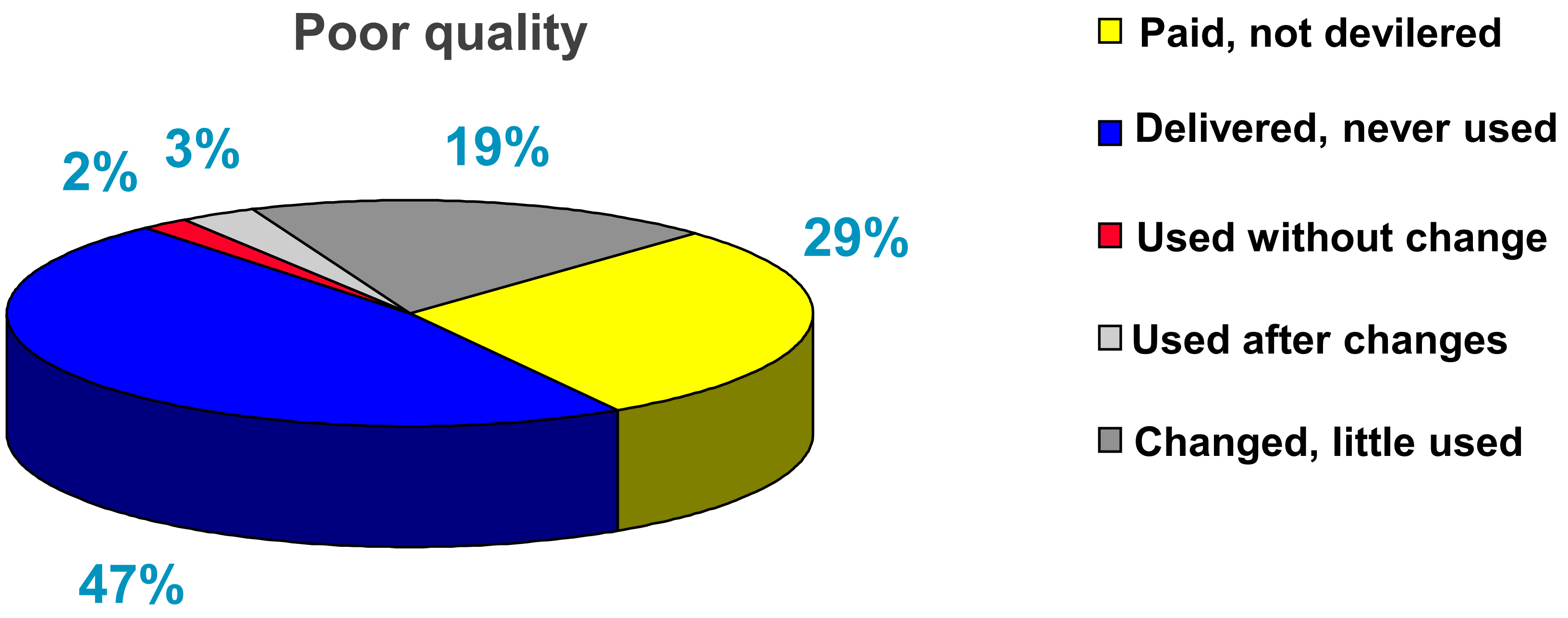
- The solution does not deliver what the *business needs*

- The solution is *not accepted* by the end user population

- The project runs *over time*

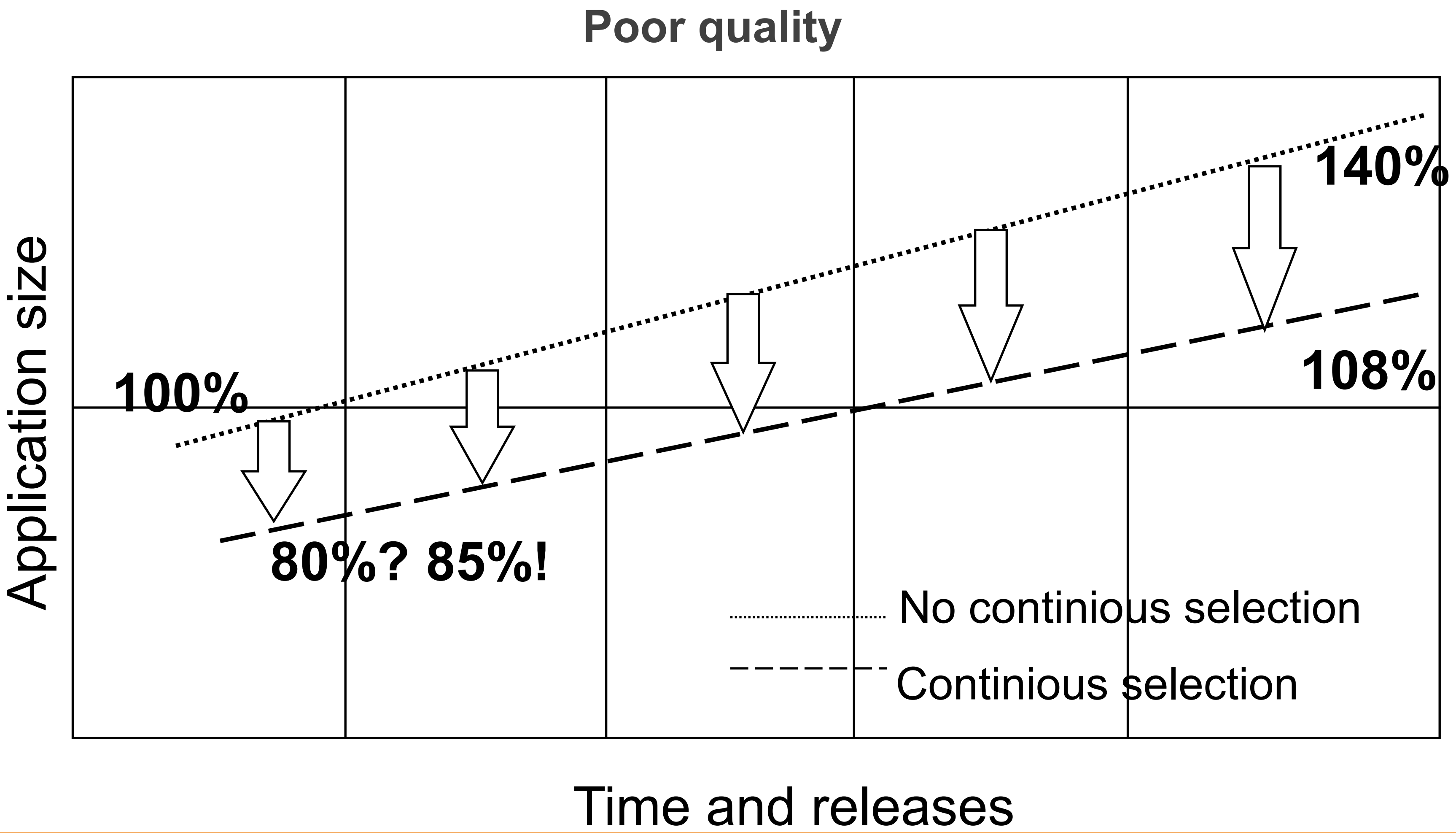
- The project runs *over budget*

Consequences



Gartner

Consequences

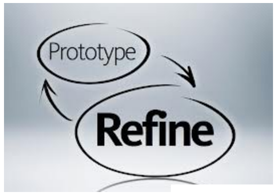


Solutions

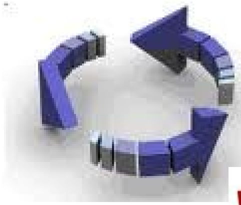
- Short delivery cycles for regular checks
  Brings quality, decreases re-work

- End user participation for validation and verification
  Brings acceptance and a match with the business

- Prototyping, from an early stage
  Brings understanding and quality

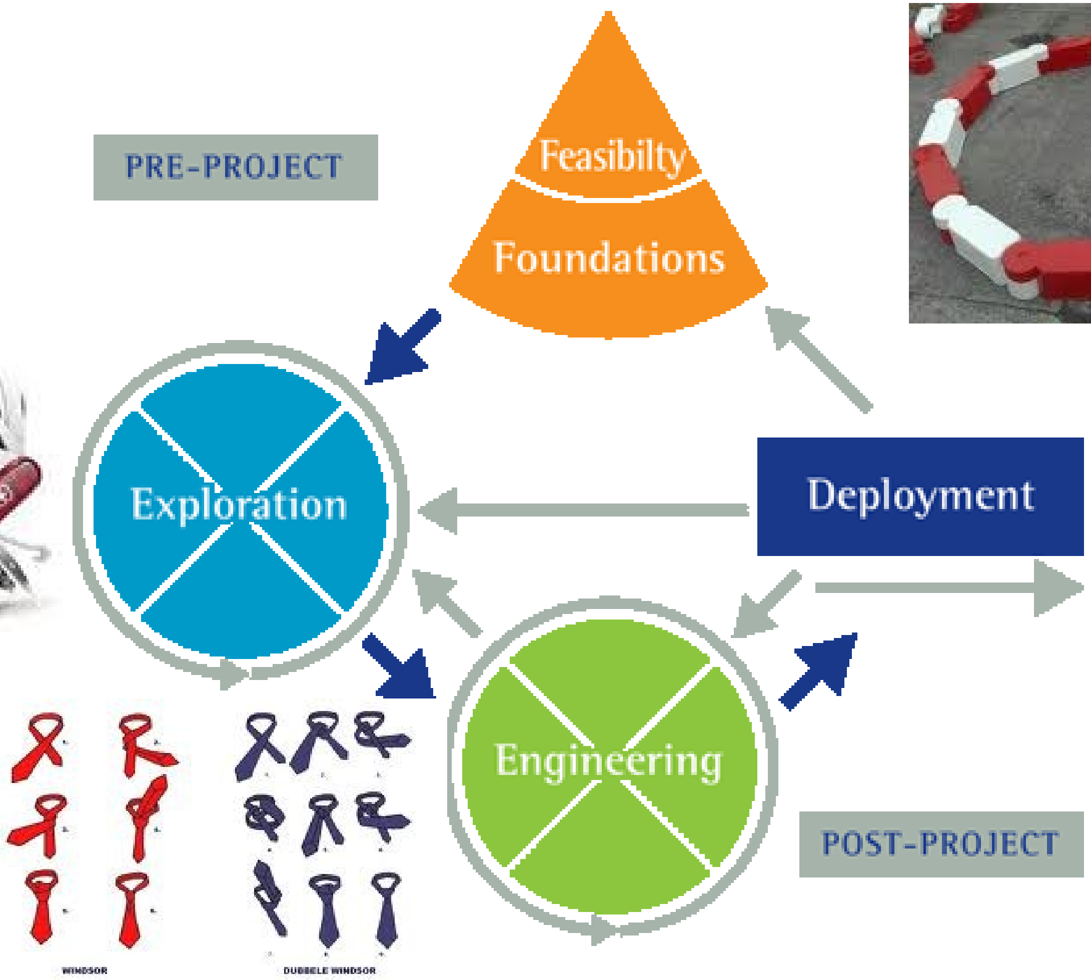


- Continious testing through the project cycle
  Brings quality, decreases re-work

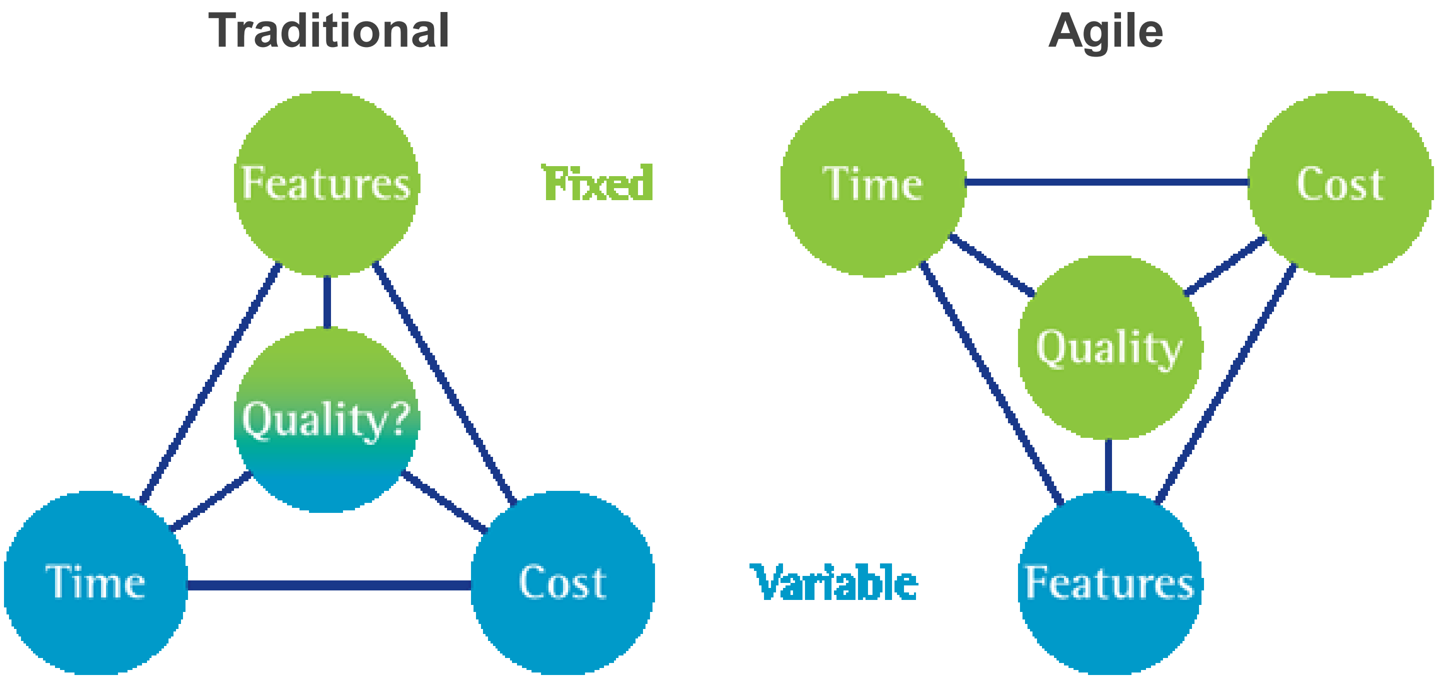


- Continious selection
  Avoids projectobese and brings therefor quality

Solutions


PRE-PROJECT
Feasibilty
Foundations
Exploration
Deployment
Engineering
POST-PROJECT
WINDSOR
DUBBELE WINDSOR

Solutions



people make the difference

Forwarde reasoning or learning from the past

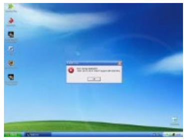
Solutions

Testprinciples

- Validation
  Test if we understood eachother right

- Verification
  Test if we really achieve the stated business benefits

- Error centric
  Test to find the software bugs



- Integrated
  Test during the complete project cycle to find issues at an early stage

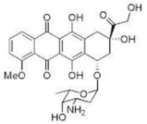
- Independent
  Test to avoid tunnel vision

- Repeatable
  Test in a wat that makes it easy to reproduce and prove the fix

Solutions

What does it take?

- Structure

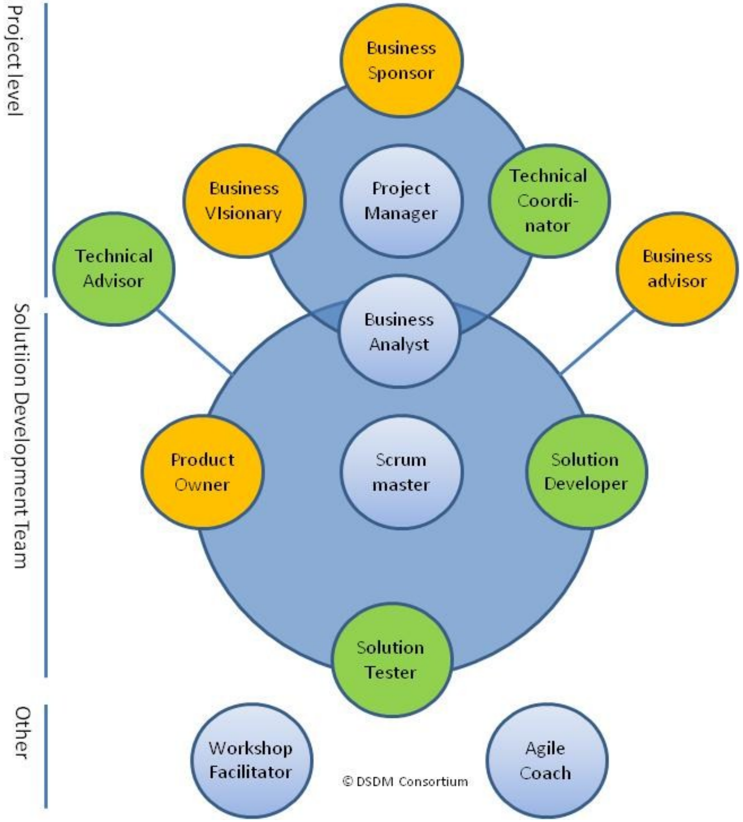


- Discipline

- The right people

- The right way

- Support and guidance



© DSDM Consortium

Solutions

In general:

Better is worse than good enough

At the moment

- The Agile domain has changed
- New innovations
- Scrum dominates
- DSDM re-surfaces

- Salt Lake Experience
- Bad acceptance
- Often little or known recognized business value
- Little user involvement
- Little holistic focus
- Return of some of the earlier errors/problems
- “No tester”

Conslusions

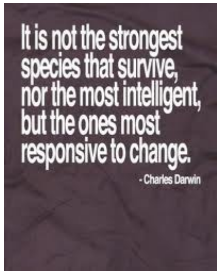
The Agile Manifesto

- Individuals and interaction over processes and tools

- Working software over comprehensive documentation

- Customer collaboration over contract negotiation

- Responding to change over following a plan



It is not the strongest species that survive, nor the most intelligent, but the ones most responsive to change.

- Charles Darwin

Conslusions

The 12 principles of Agile software

- Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- Business people and developers must work together daily throughout the project.
- Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
- Working software is the primary measure of progress.
- Agile processes promote sustainable development.
  The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
- Continuous attention to technical excellence and good design enhances agility.
- Simplicity--the art of maximizing the amount of work not done--is essential.
- The best architectures, requirements, and designs
  emerge from self-organizing teams.
- At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

Conclusion

- It was and still is in the frameworks
  Re-use and don't think lightly
  Focus on the process, Quality will follow

(c)www.asianjobportal.com

- It is a framework so it needs some attention and proper application
  Make sure activities are well assigned to the right professionals (and yes, also a professional tester)

- It has been done, so there is best practice
  Get it at the proper Agile platforms

To let you know….

Agile Consortium International

- Knowledge sharing, innovation, standardization
- Accreditation, independent quality label
  Agile Foundation, Practitioner, Project Manager and Master

Links:

- www.agileinthecore.com
- www.certifytoinspire.com
- www.dsdm.org (ABC, October 2012)
- www.agileconsortium.net (under construction)
- www.iaf-world.org
- www.iaf-world.eu (October 2012)
- www.pmtd.nl

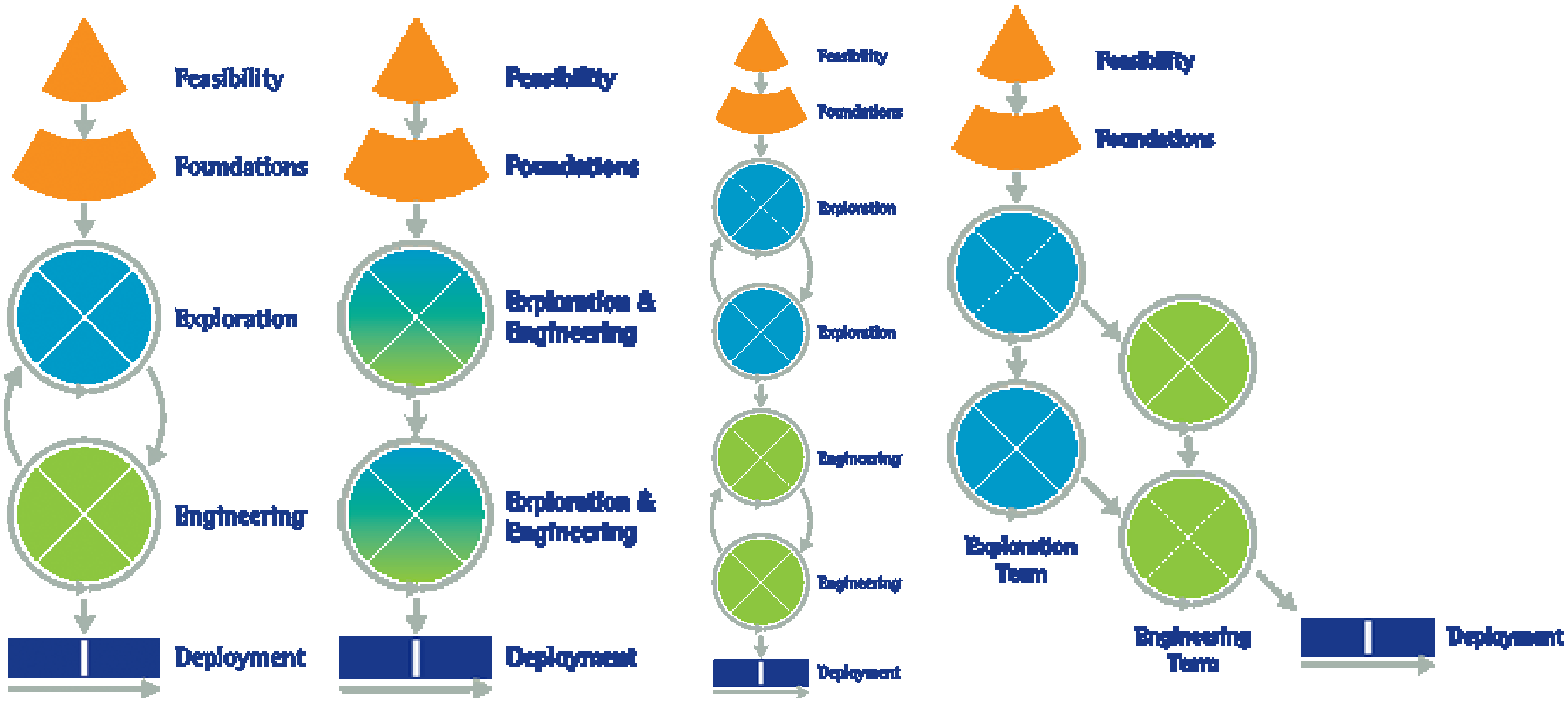
De duistere kant van de kenniseconomie

people make the difference

Arie van Bennekum | arie.van.bennekum@pmtd.nl | +316-16535445

PMtD Van Nelle Ontwerpfabriek | Koffie 1.20 | Van Nelleweg 1, 3044 BC Rotterdam | Postbus 13120, 3004 HC Rotterdam
telefoon +31(0)10 751 65 35 | fax +31(0)10 751 65 39 | info@pmtd.nl | www.pmtd.nl

Algemene inleiding Atern



Forwarde reasoning or learning from the past

Algemene inleiding Atern

Hoe zit het met toepasbaarheid?

- *Altijd is iets* van Atern toe te passen en *meestal alles*
- Het gaat met name om een *situationele en pragmatische* aanpak.

Algemene inleiding Atern

Systeem- projectkarakteristieken voor geschiktheid:

- Zichtbare functionaliteit (interactief)
- Aanwijsbare gebruikersgroepen
- Deelbare functionaliteit
- Decomponeerbare functionaliteit
- Tijdsdruk
- Verschillende onderlinge prioriteit van de functionaliteit
- Nog onduidelijke informatiebehoefte

Algemene inleiding Atern

Systeemkarakteristieken met extra aandacht:

- Projecten met reuse als DOEL
- Procesbesturingsapplicaties
- Safety critical applicaties
- Real time applicaties

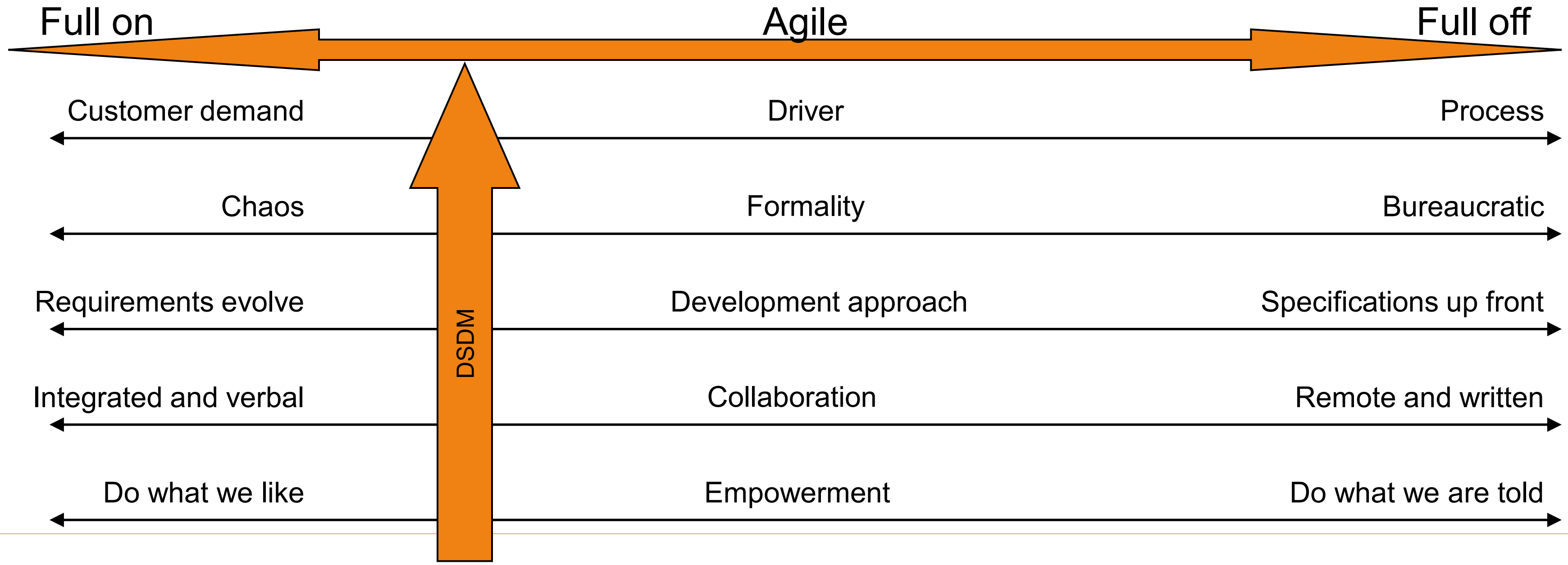
Algemene inleiding Atern

Basistechnieken:

- Cultuur (no blame, samen, zelfsturing, communicatie)
- Timeboxing
- Facilitated workshops
- MoSCoW
- Prototyping
- Gecontroleerd itereren
- Schattingen
- Incrementen

Algemene inleiding Atern

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



Full on Agile Full off

Customer demand Driver Process

Chaos Formality Bureaucratic

Requirements evolve Development approach Specifications up front

Integrated and verbal Collaboration Remote and written

Do what we like Empowerment Do what we are told

DSDM

Algemene inleiding Atern

Product

Plug-in

Element

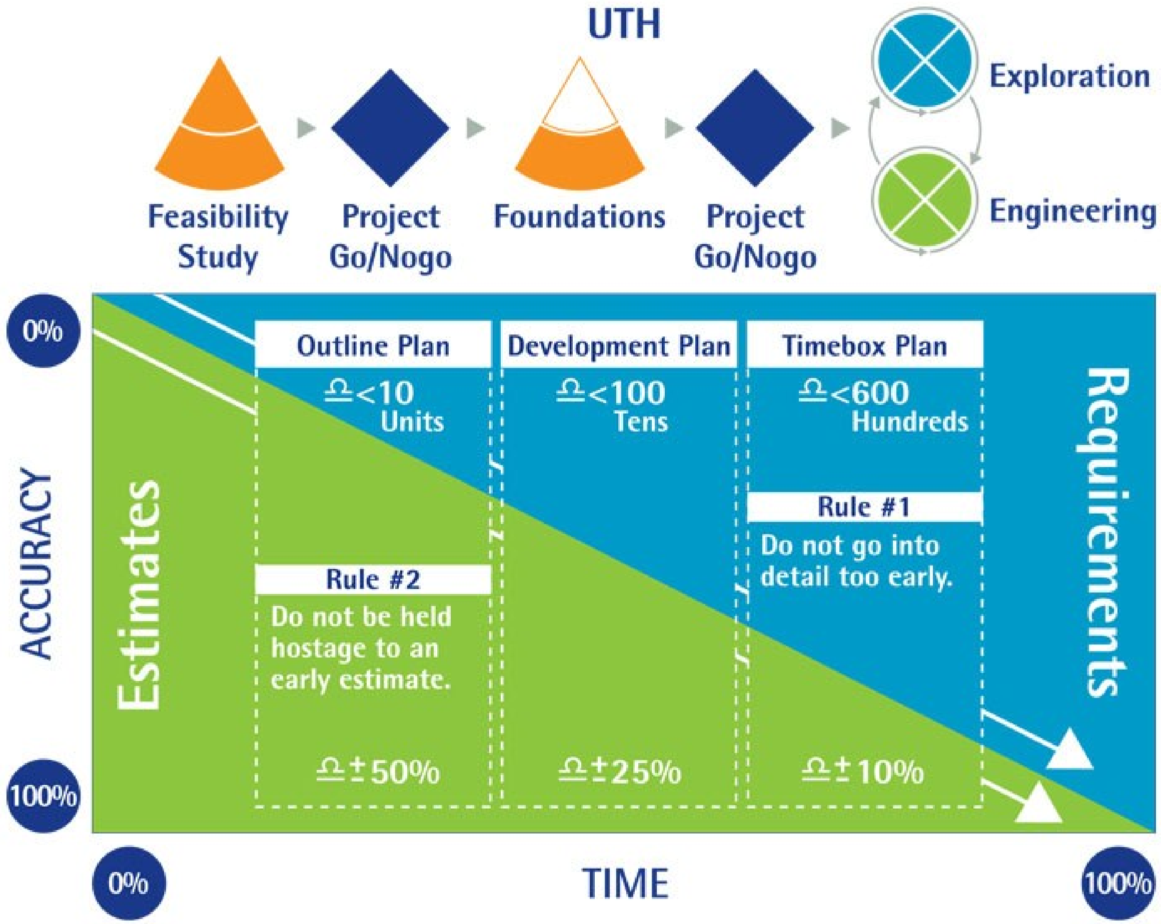
Algemene inleiding Atern

Fase	🎁	🧩	🔌
Pre-project	Terms of Reference		
Feasibility	Feasibility Asssessment	Outline Business case	
		Outline Solution	
		Outline Plan	
	Feasibility Prototype		
Foundations	Business Foundations	PRL	
		BAD	
		Business Testing Strategy	Ja
		Business case	Ja
	Solution Foundations	SAD	
		Technical Impl. Standrds	
		Technical testing Strategy	
		Solutionn Prototype	
	Management Foundations	Management Plan	
		Delivery Plan	
		Oper. Readiness Checklist	
		Risk Log	

Algemene inleiding Atern

Fase	🎁	🧩	🔌
Exploration	Evolving Solution		
	Business Analysis Model		
	Design Model		
	Solution Review Records		
	Business Testing Suite		Ja
	Technical Testing Suite		
	Timebox Plan		
	Timebox Review Record		
	Delivery Control Suite		Ja
	Deployment Plan		
	Benefits Realisation Plan		Ja
Engineering	Deployable Solution		
Deployment	Deployed Solution		
	Support Documentation		
	Trained users		Ja
	Business User Document.		Ja
	Project Review Report		
Post-project	Benefits Realisation Rep.		Ja

Algemene inleiding Atern



people make the difference

Forwarde reasoning or learning from the past