Metodologie AGILE nello sviluppo di medical device software

Marco Prattichizzo
Campus di Ingegneria, Cesena
21 Novembre 2019

Overview

- AGILE
- Regulations
- AGILE in Medical Device Software
- Waterfall --> Agile
- Conclusion

Introductions First

Laurea Magistrale in Ingegneria Biomedica (Cesena)
Luglio 2012

Tesi “Caratterizzazione Quantitativa delle Curve Tempo-Concentrazion nella TC perfusionale polmonare»

Mirada Medical (Oxford, UK)
Febbraio 2014

Internship
Test Engineer
Analyst

Vision RT (Basingstoke, UK)
Maggio 2019

Quality Engineer

Optellum (Oxford, UK)
Novembre 2019

Analyst/Test Engineer

What is AGILE?

What is AGILE?

A framework of principles and practices used by collaborative teams to rapidly and frequently deliver customer valued software that satisfies business needs

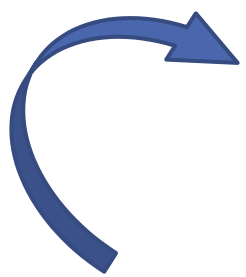
AGILE Manifesto

https://agilemanifesto.org/

Roles and Responsibility

Product Owner

- Defines the features of the product
- Manages product backlog in priority order
- Represents all stakeholders' needs
- Accepts or rejects the product



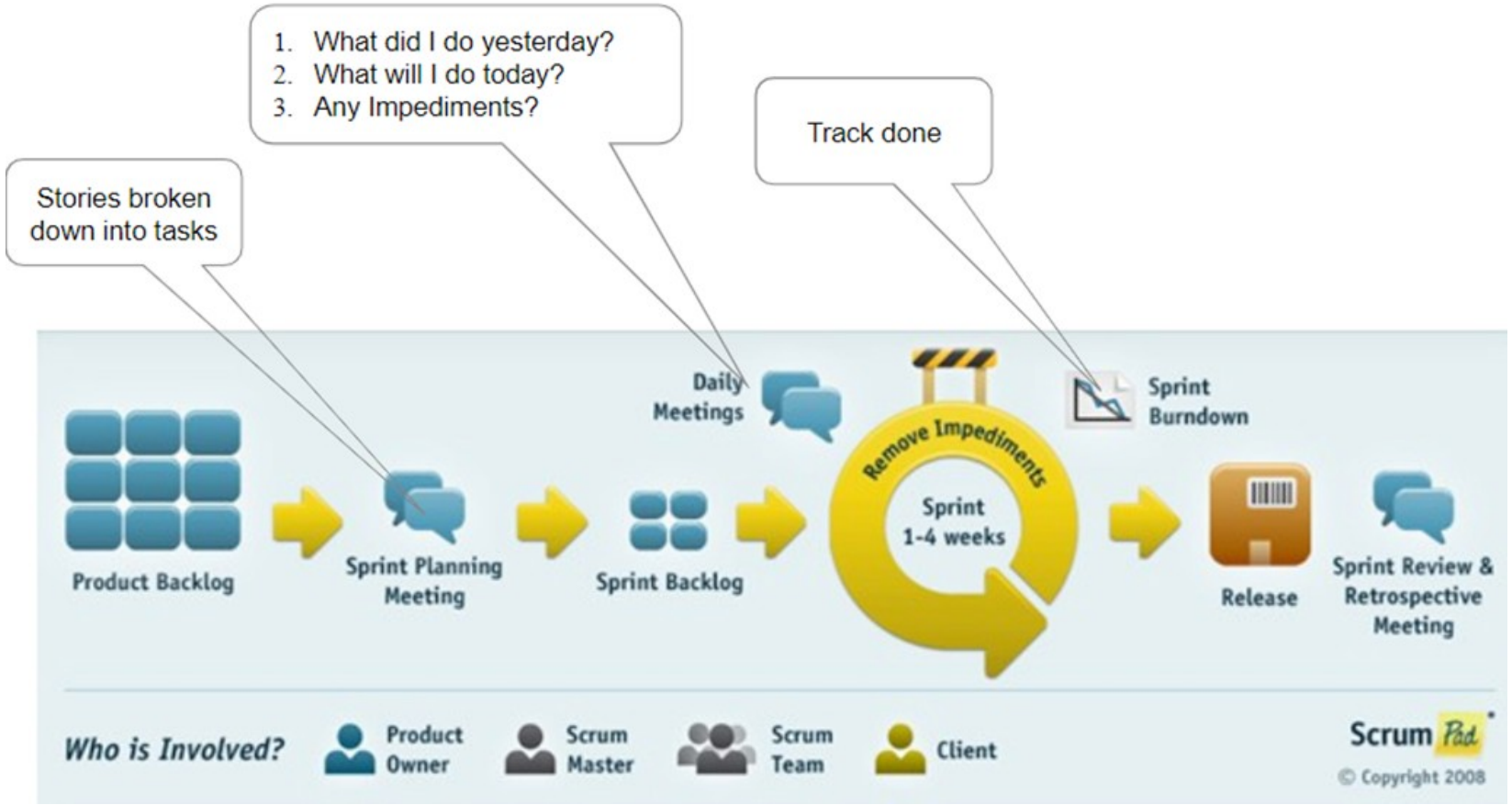
Scrum Master

- Remove impediments to allow the the team to make progress at constant/desired pace
- Facilitates communication/interaction among all parties
- Promotes scrum values and practices

Development Team

- Cross functional and 5-9 in size
- Performs the actual development work
- Decides on how works gets done
- Responsible for managing the sprint backlog and meeting the self-defined target

Scrum


1. What did I do yesterday?
2. What will I do today?
3. Any Impediments?
Track done
Stories broken down into tasks
Daily Meetings
Sprint Burndown
Remove Impediments
Sprint 1-4 weeks
Product Backlog
Sprint Planning Meeting
Sprint Backlog
Release
Sprint Review & Retrospective Meeting
Who is Involved?
Product Owner
Scrum Master
Scrum Team
Client
Scrum Pad
© Copyright 2008

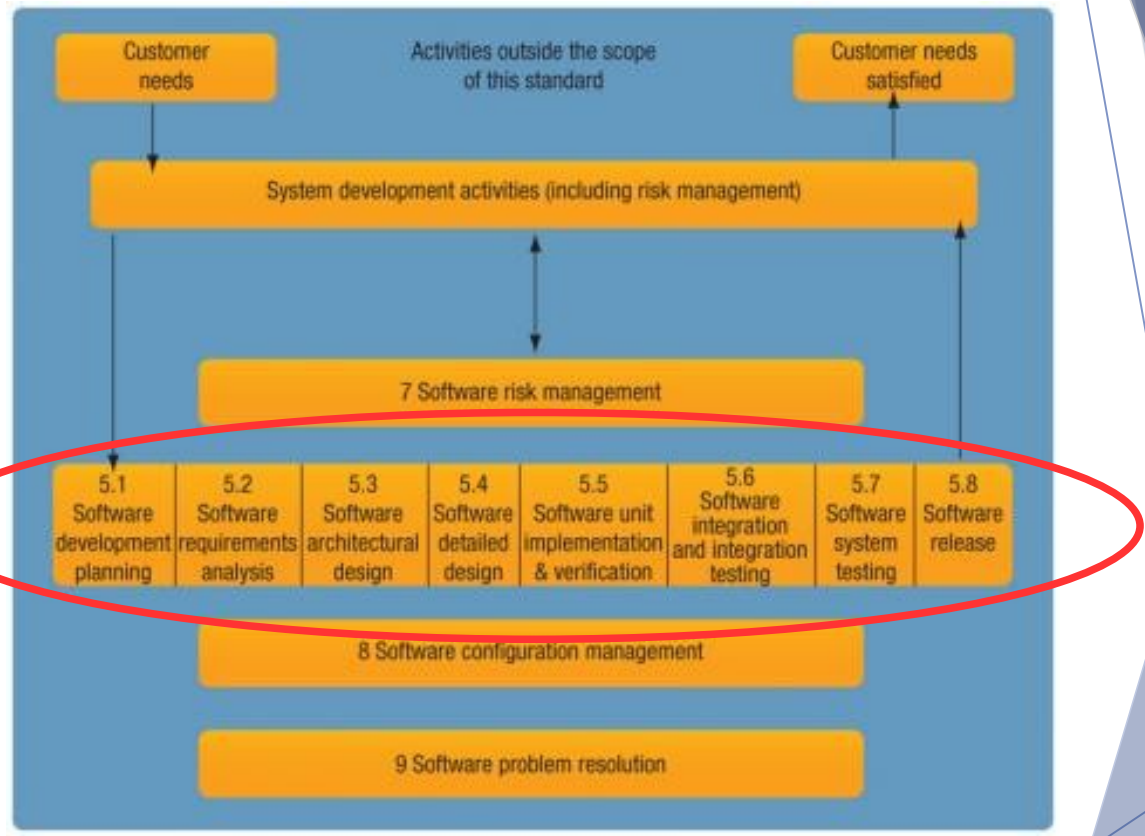
Regulations

- EU: MDR (MDD)
- USA: FDA 21CFR820
- IEC 62304 Medical Device Software – Software Life Cycle Processes
- ISO 13485 Medical devices -Quality Management Systems - Requirements for Regulatory Purposes
- ISO 14971 Medical Devices – Application of Risk Management to Medical Devices

…

IEC 62304

1. Scope
2. Normative References
3. Terms & Definitions
4. General Requirements
5. Development Process
6. Maintenance Process
7. Risk Management
8. Configuration Mgmt
9. Problem Resolution



IEC 62304 V-Model

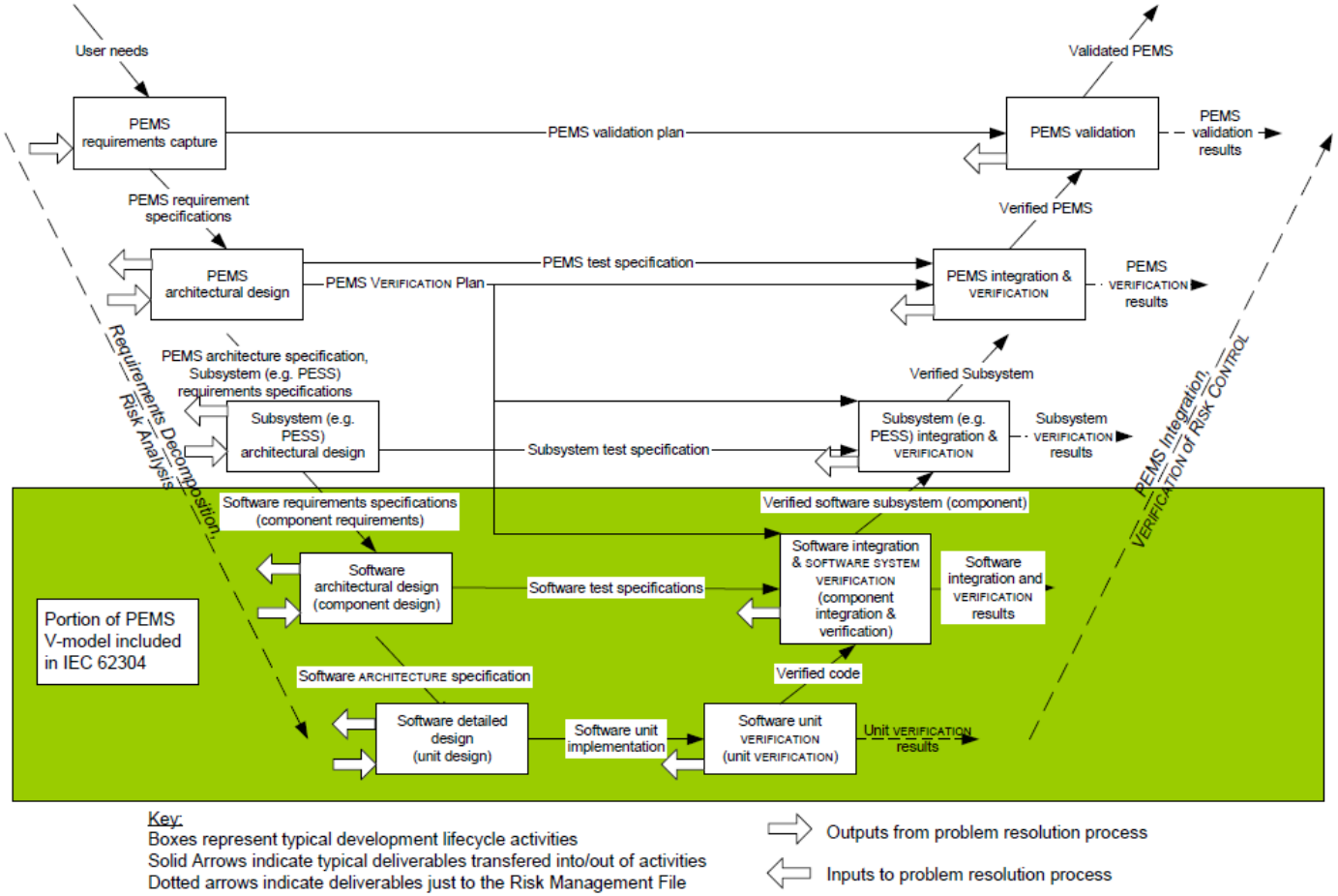
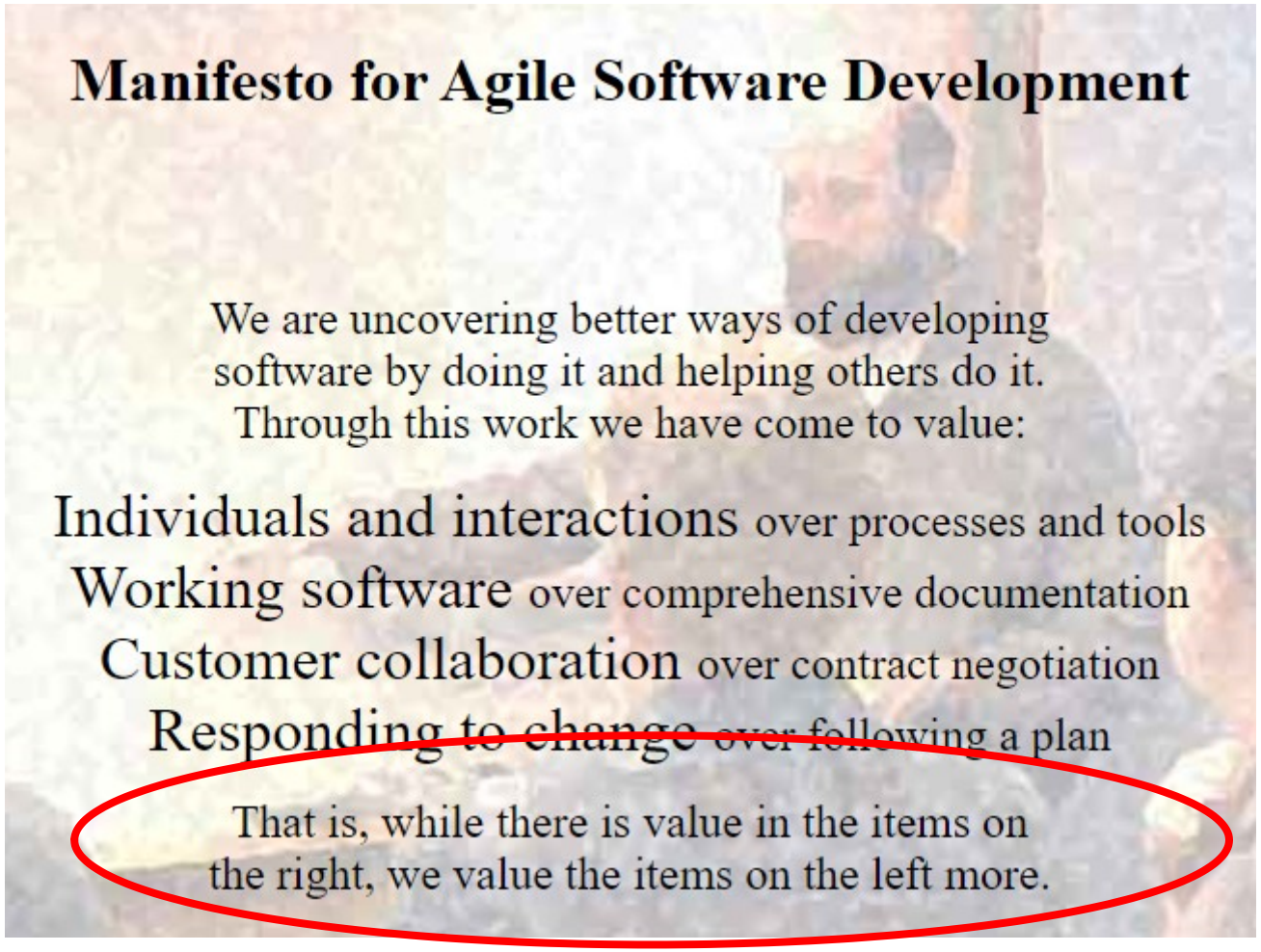


Figure C.2 – Software as part of the V-model

AGILE VS Medical Device



Won't do

- Doesn't allow developers to hack code any way they want without docs
- Won't increase development productivity overnight
- Won't let product managers get everything they won't

Will do

- Give clear visibility on progress
- Show where the problems are and drives continuous improvement
- Lets product owners decide on what is most important and adapt ideas
- Deliver highest values features

"Is not a euphemism for no documentation, it is a disciplined project process that requires planning and communication"

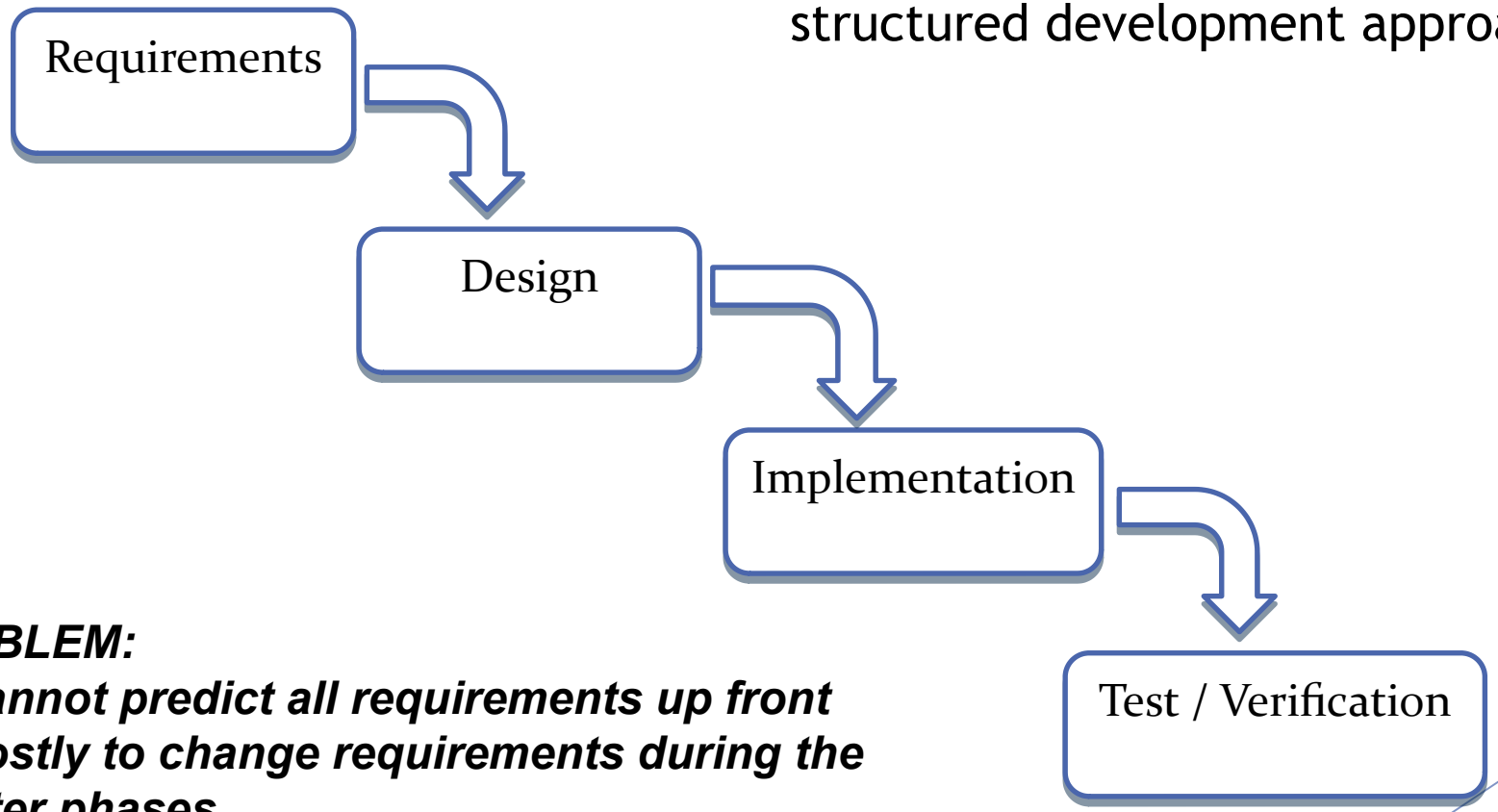
Mirada Medical – From Waterfall to AGILE

- Small engineering team 20-25 people
- Strong academic and scientific culture
- Highly experienced in medical imaging
- Skepticism of AGILE

- 2-3 product lines with 3-9 month releases
- Minimal tooling and devops support
- Primarily manual testing

Waterfall

Waterfall became popular as a more structured development approach



PROBLEM:

- ***Cannot predict all requirements up front***
- ***Costly to change requirements during the later phases***

Development Driven Development

Step 1 – Cross functional Design session

Step 2 – Rapid Prototyping / Development

Step 3 - Requirements, tests, code

Key Points:

- Help enable early user feedback
- Proof of concept to reduce technical uncertainty
- Improved communication – more effective concept reviews
- Better alignment between reqs, test and product
- Does not mean skipping reqs and design
  - Requirements analysis evolves and refines design
  - Test makes the design more robust

What Which Why How?

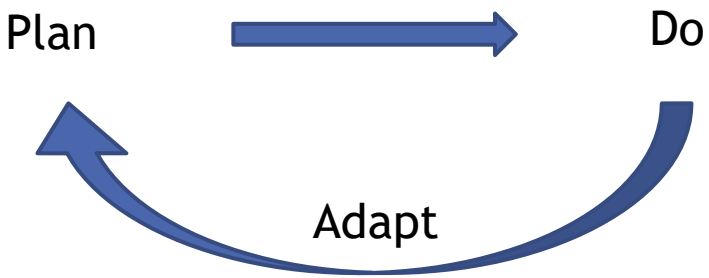
- How do we implement AGILE?
- How does AGILE relate to Mirada development process (QMS)?
- Why AGILE?
- How does AGILE fit with medical device regulated development?
- Which AGILE flavor?

Challenges

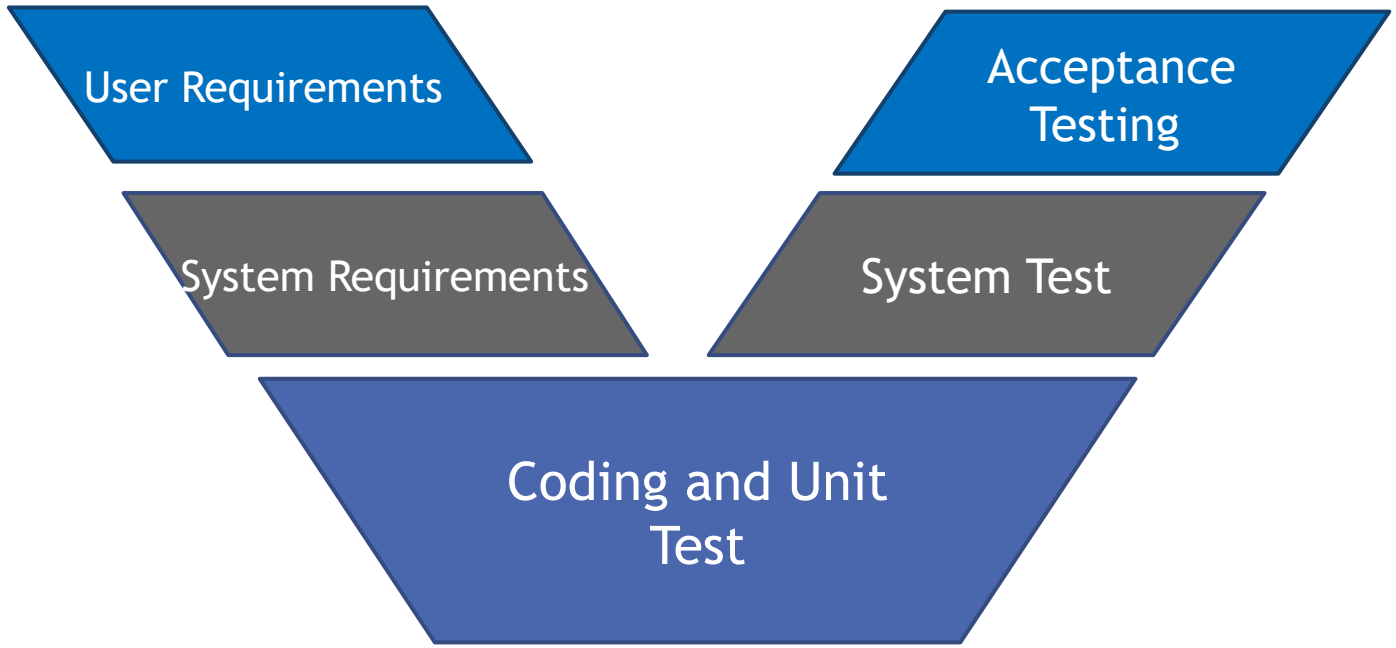
- Volume of documentation
- High overhead for each release
- Not ready to release until tested
  - Releasing is costly
- Tying development to requirements is hard
- Risk management
- Do we want scope scope flexibility?
  - Scope change is an overhead
- There is a minimum scope to make a release
  - Cost to customer to update
- Why sprints?
  - Sprint could be harmful for small teams, they just add barriers

Experiment and adapt

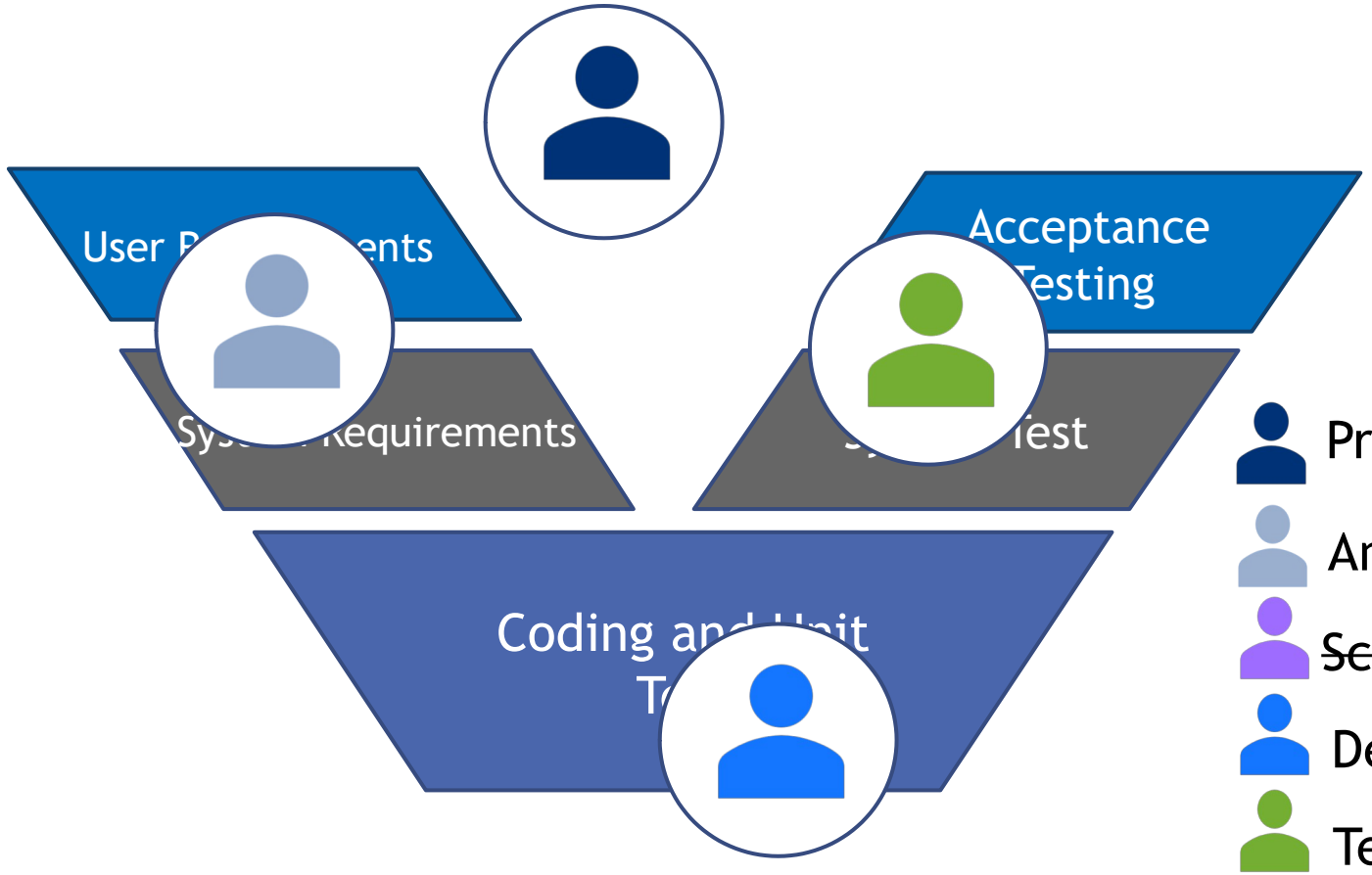
- Several workshops ran by the software developer manager involving all development team (analysts, testers, developers, architects ecc…)
- New updated processes not imposed but results of collaborative efforts
- Continuous improvement
- Retrospectives



Micro AGILE Team Setup



Micro AGILE Team setup



Product Manager

Analyst

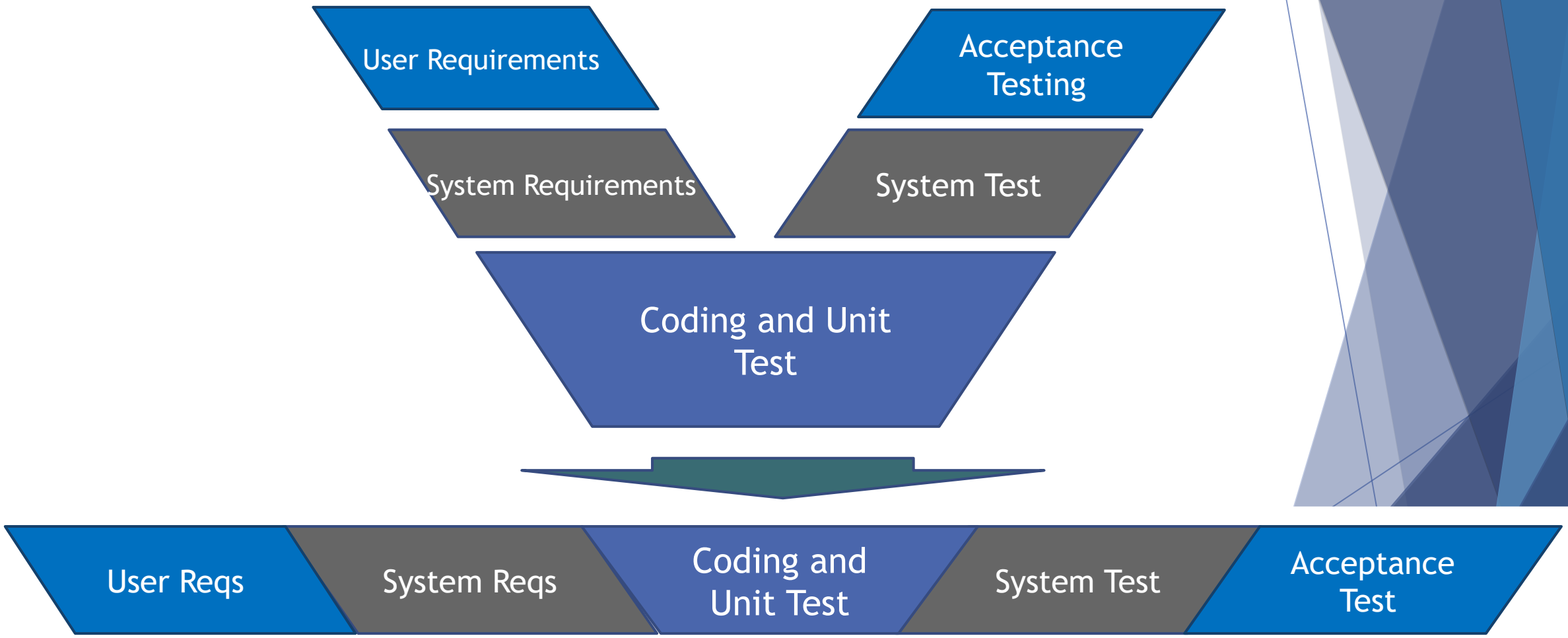
~~Scrum master~~

Developer

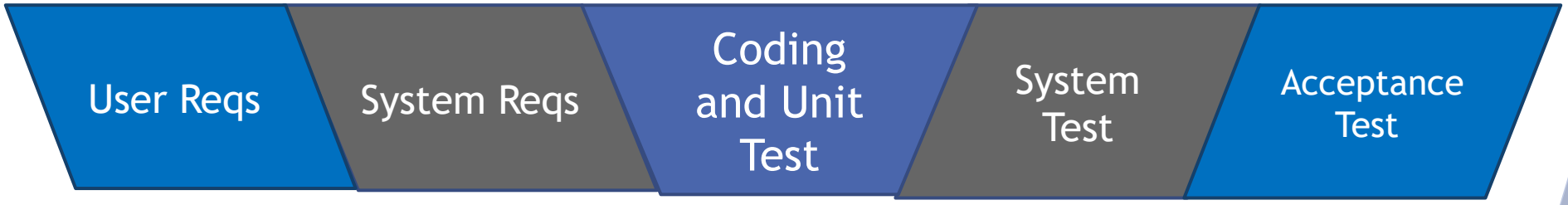
Tester

*Multi-skilled people

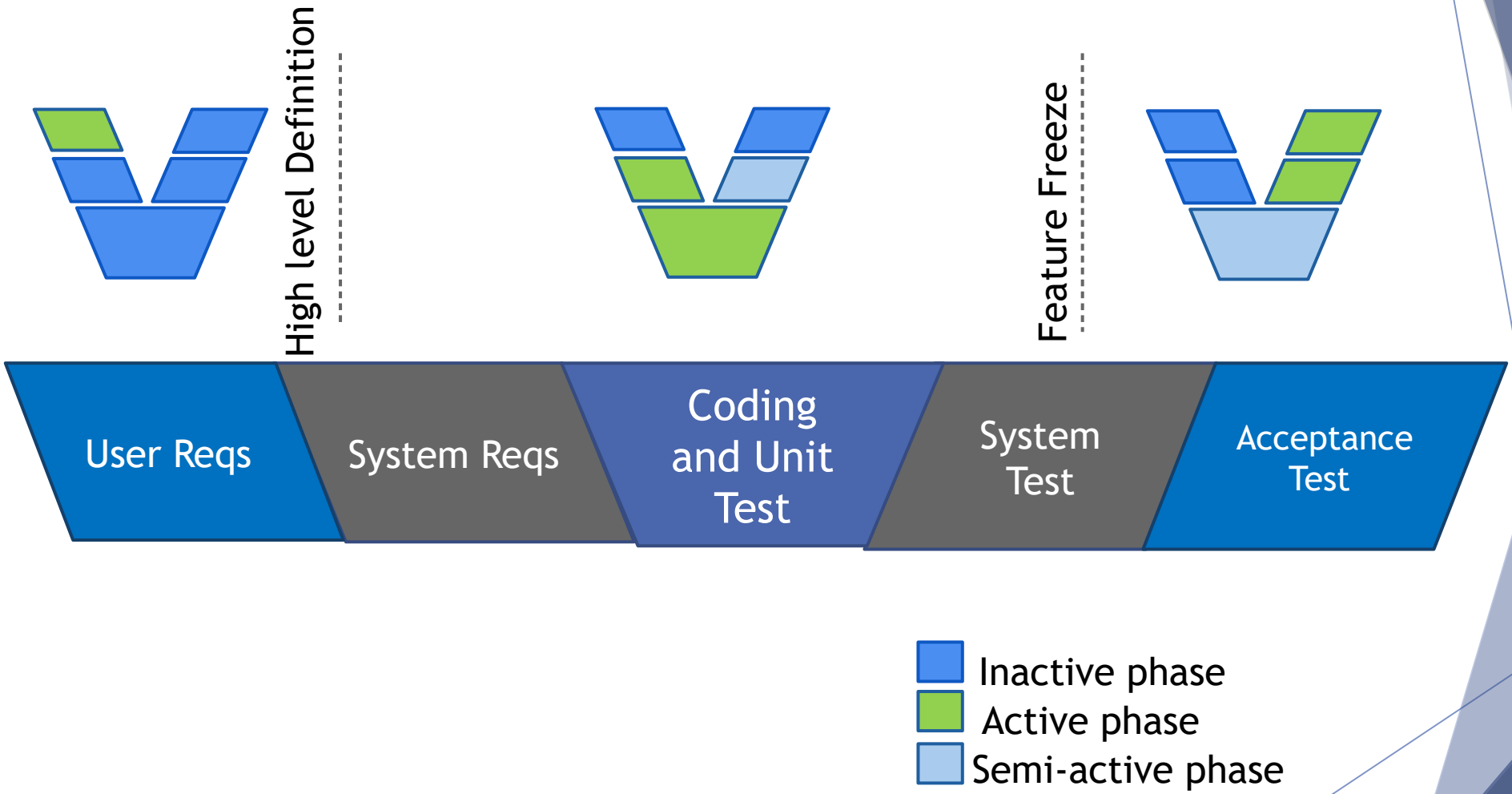
V-model


User Requirements
Acceptance Testing
System Requirements
System Test
Coding and Unit Test
User Reqs
System Reqs
Coding and Unit Test
System Test
Acceptance Test

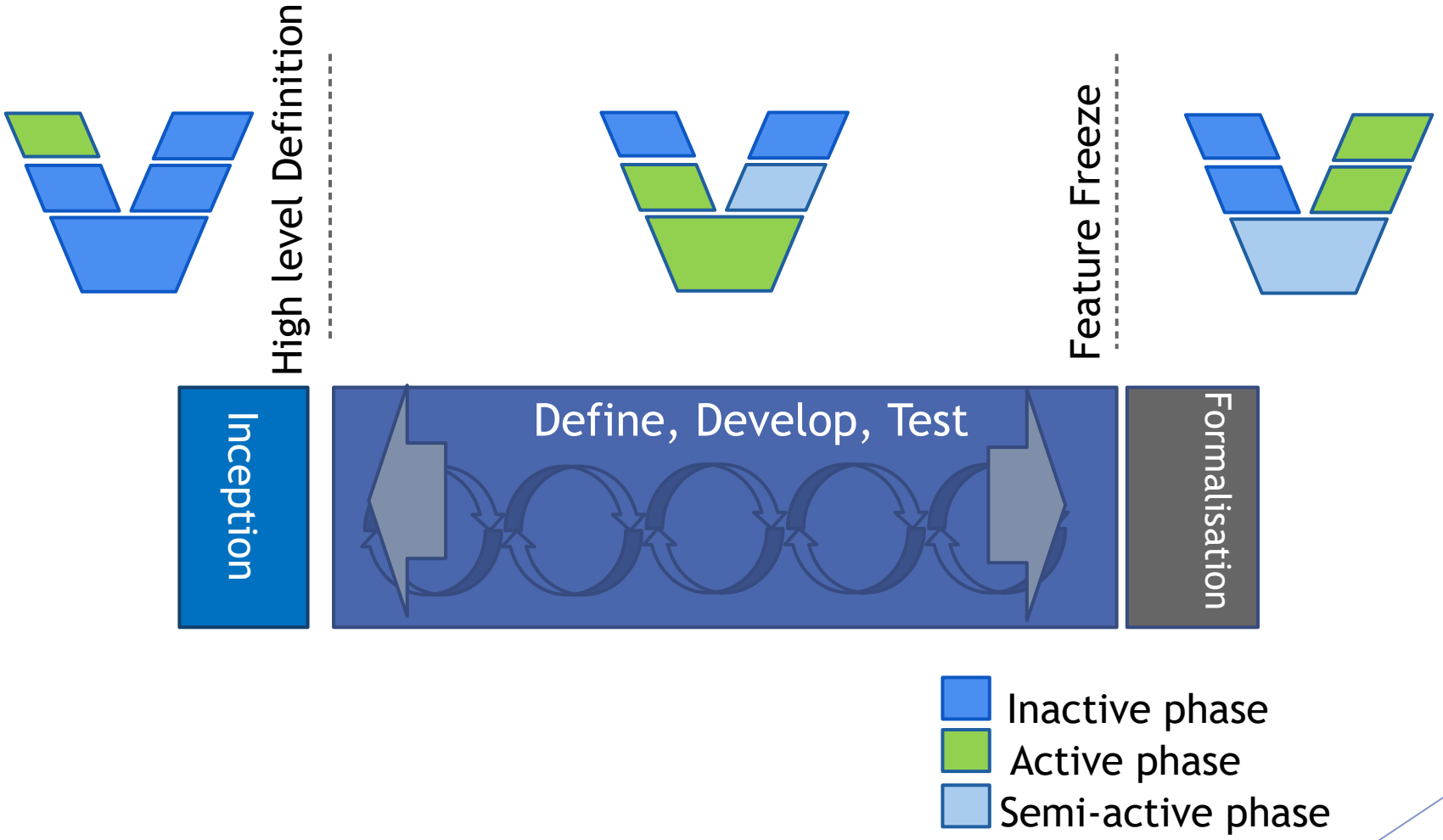
Agile in V-model (Wateragilefall)



Agile in V-model (Wateragilefall)



Agile in V-model (Wateragilefall)



Lesson Learned

- Micro AGILE team weaknesses
  - Higher impact from missing team member
  - Difficult to scale
  - Lack of cross team interaction – divergence, waste
  - Limited functional peer level support

Lesson Learned

- Project retrospectives helped identify pain points
- Piloted improvement idea (Kanban) on new major project
- Positive results:
  - Product backlog
  - Design sessions
  - Pair Programming
- Complaints about daily stand-ups (too many meetings) but:
  - Main goal of daily stand-ups is to increase day-to-day communication
  - Communication/collaboration already strong
- Development Driven approach encourages early feedback and improve communication
- You can't force adoption of AGILE and expect to get the right mindset
- Experiment and adapt (flexible QMS)

Lesson Learned (Challenges)

- Volume of documentation
  - Right tool, sum of the parts documentation
- High overhead for each release
- Not ready to release until tested
  - final INCREMENT to complete the final tasks needed to produce shippable software
- Tying development to requirements is hard
- Risk management
  - Included in *donness* criteria
- Do we want scope scope flexibility?
  - Change control
- There is a minimum scope to make a release
  - Backlog MVP line
- Why sprints?
  - Sprint could be harmful for small teams, they just add barriers

Vision RT

- Big development team ~100 people, 3 offices (2 UK, 1 Poland)
- Big AGILE team ~9 people
- 2 main product line with 6-12 month release cycle
- 3 week sprint
- Rigid structure
- Detailed AGILE process in QMS

But…

- Team under pressure, implement features as quickly as possible
- Frequent scope/feature changes (loose change control)
- Documentation tool difficult to use and not fit for AGILE
- Silos teams (developers, testers, product owners), little communication
- Backlog items done with no documentation
- Documentation (detailed design requirements) done all at the end of the project (just to meet regulatory requirements)
- >250 detailed design requirements created after implementation (with no trace to tests)!

There isn’t a magic formula for AGILE

- Process ownership
- Continuous improvement
- Clear definition and rigid application of *doness* criteria (including QMS requirements)
- Useful documentation
- Change management process
- Communication
- **Being** AGILE, not *doing* AGILE

Further reading

AAMI TIR45:2012, Guidance on the use of AGILE practices in the development of medical device software

Thank you

Questions?

Marco Prattichizzo
Analyst/Test Engineer

marco.prattichizzo@gmail.com
www.linkedin.com/in/marcoprattichizzo