Software Estimating – is the problem solved ?

05/2020

QINETIQ

The QinetiQ Team

Sanathanan Rajagopal – QinetiQ Fellow

Estimating Manager- Cyber and Information

- ICEAA Certified Cost Estimator / Analyst with the Parametric Specialism (ISPA)
- Chairman and member of the board of the Society for Cost Analysis and Forecasting (SCAF)
- Fellow of Defence Equipment and Support, MoD, UK
- Recipient of the ICEAA Technical Achievement of the year 2017 award
- Member of EPSRC Centre of Innovative Manufacturing (Through Life Engineering Services, Cranfield University)

Publications:
- More than 30 publications in National, International, Scientific Journal papers
- Supported in the development of modules for ICEAA Software CEBoK
- Contributor to a study undertaken by Institute for Defence Analysis (US DoD) sponsored by Defence Logistics Agency (DLA, DoD) on Obsolescence titled " A Research and Development Investment Portfolio for Diminishing Manufacture Sources and Material shortages"
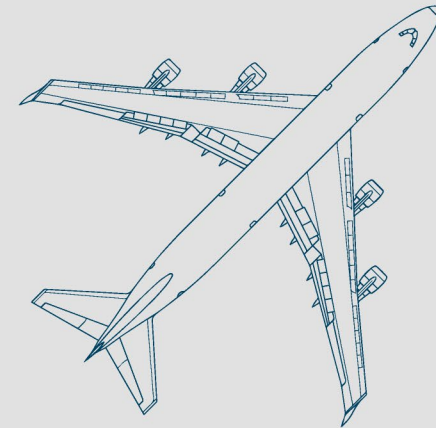
5

5 fundamental mobile phone technologies including Touch Screen and Liquid Crystal Displays (LCD) developed by QinetiQ experts

40

40 organisations, including the Royal Navy involved in 6 weeks of operations during the Unmanned Warrior Exercise

50+

including 50+ unmanned vehicles operating in the air, land and sea

12

COMMERCIAL IN CONFIDENCE

12 Empire Test Pilot School students have become astronauts including Major Tim Peake

85+
locations woldwide

1,300+
patents (including 300+ pending)

We are QinetiQ

£833m
FY2018 revenue

6,000+
people with unique science and engineering expertise

3secs

Every 3 seconds a Boeing aircraft takes off or lands that has been tested in QinetiQ's low speed Wind Tunnel

1,850km

1,850km of the TANAP pipeline will be protected by OptaSense®

16

Our Ocean Basin in Gosport, UK contains enough water to fill 16 Olympic swimming pools

QINETIQ

Agenda

Disclaimer

"This is my personal view and does not represent views of my company"

COMMERCIAL IN CONFIDENCE

Software Estimation

Why is it different to Hardware

QINETIQ

Introduction

Programming a computer does requires intelligence. Indeed, it requires so much intelligence that nobody really does it very well. Sure, some programmers are better than others, but we all bump and crash around like overgrown infants. Why? Because programming computers is by far the hardest intellectual task that human beings have ever tried to do. Ever.

-G.M. Weinberg, 1998

Software Engineering

"Software is the entire set of programs, procedures and related documentation associated with a system"

"Software engineering is the application of science and mathematics by which the capabilities of computer equipment are made useful to man via computer programs, procedure, and associated documentation"

Software Engineering Economics

Why is it different to Hardware ?

- Software engineering is a new discipline compared to civil/mechanical/electrical

- Software weigh nothing

- Software is invisible

- It always increases

- Software is intangible

- Complexity

- Conformity

- Software is malleable – can be shaped to do anything

Software Estimating
Myths, Perception and Facts

- Text

QINETIQ

Why we think it is difficult to estimate Software Cost

- Software is difficult to understand

- Software estimation is a black art

- Software costs are estimated under systems cost

- Software cannot be measured in any form, therefore difficult to estimate the cost

- Software is added as a Risk

- Systems are driven by software

- Increasing resource will increase the productivity

- Software does not get obsolete
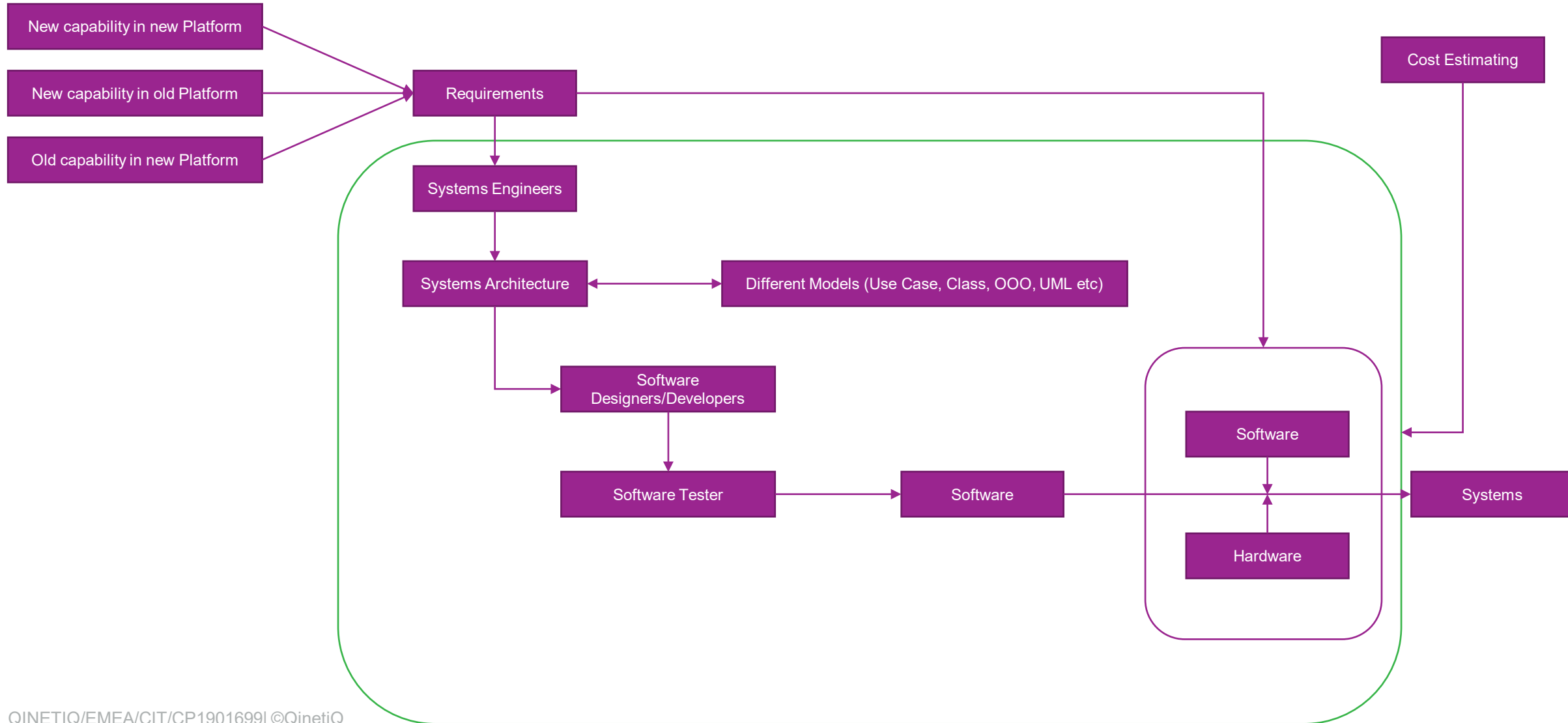
Software Engineering Problems (Software Crisis)

- Text

QINETIQ

Software Engineering Problems

- First used in 1968 NATO Conference on Software engineering

- Unreliable

- Delivered Late

- Prohibitive in terms of modification costs

- Impossible to maintain

- Performing at an inadequate level

- Inability to estimate with accuracy the cost, resource and schedule required for a software project

- Exceeding budget cost

" *Surprise – it is still with us today in some form or the other, however each of these complaints can be traced to the inability to define the requirements*"

Software Estimating

New capability in new Platform

New capability in old Platform

Old capability in new Platform

Cost Estimating

Requirements

Systems Engineers

Systems Architecture

Different Models (Use Case, Class, OOO, UML etc)

Software Designers/Developers

Software Tester

Software

Software

Hardware

Systems

COMMERCIAL IN CONFIDENCE

Software Sizing

- Text

QINETIQ

Software Sizing

- It is the quantification of effort required

- One of the key cost drivers

- Various methods are used to determine software sizes

  - Function Points
  - SLOC
  - IFPUG
  - COSMIC
  - Object Points etc

- Other key cost drivers are complexity and dependencies

Software Development Life Cycle

- The complete lifetime of a software system form initial conception through to final obsolescence.

- Different Models

  - Code and fix (before software engineering)
  - Waterfall (Various derivatives available)
  - Agile
  - Prototyping
    - Incremental
    - Iterative
    - Evolutionary
    - Spiral

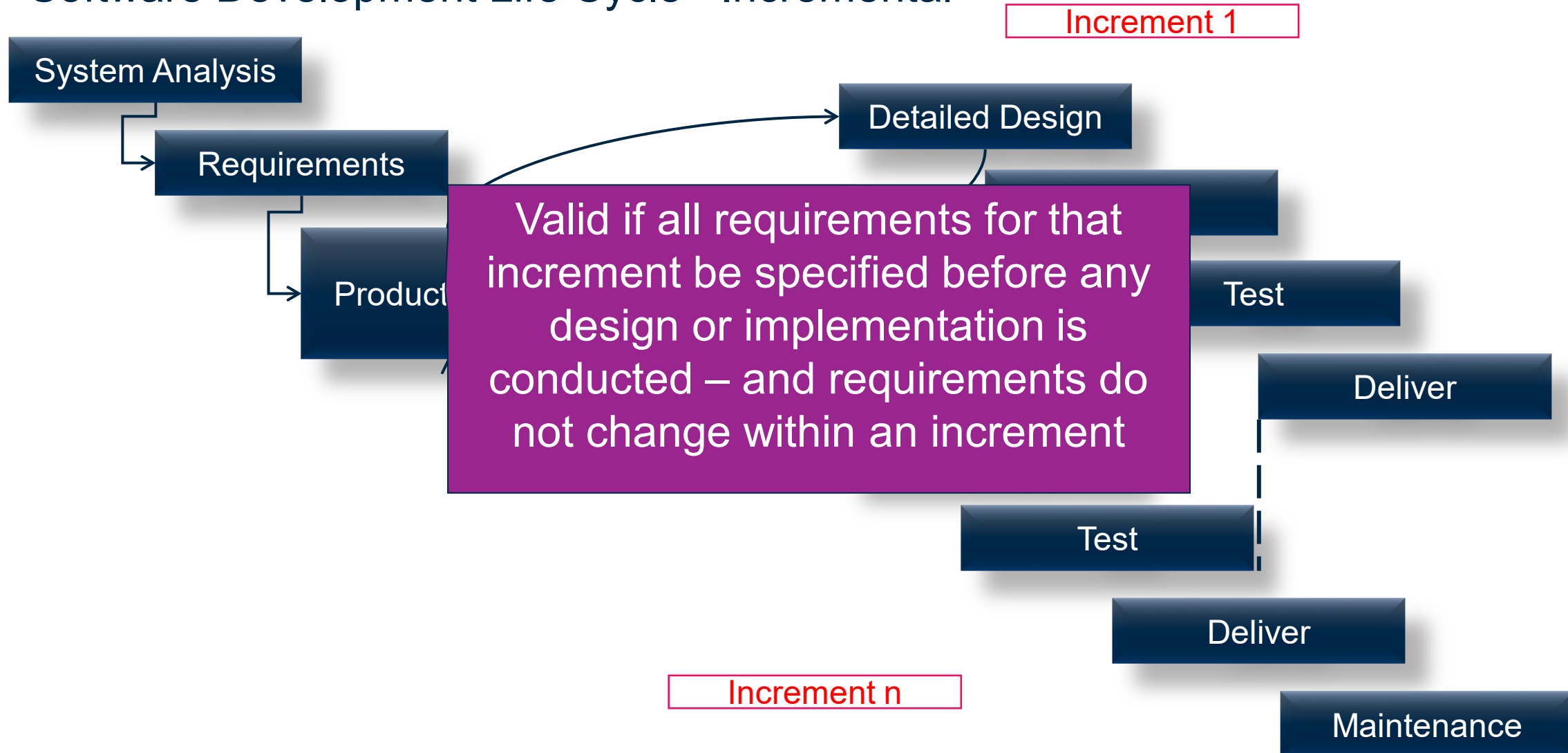QINETIQ

Software Life Cycle

QINETIQ
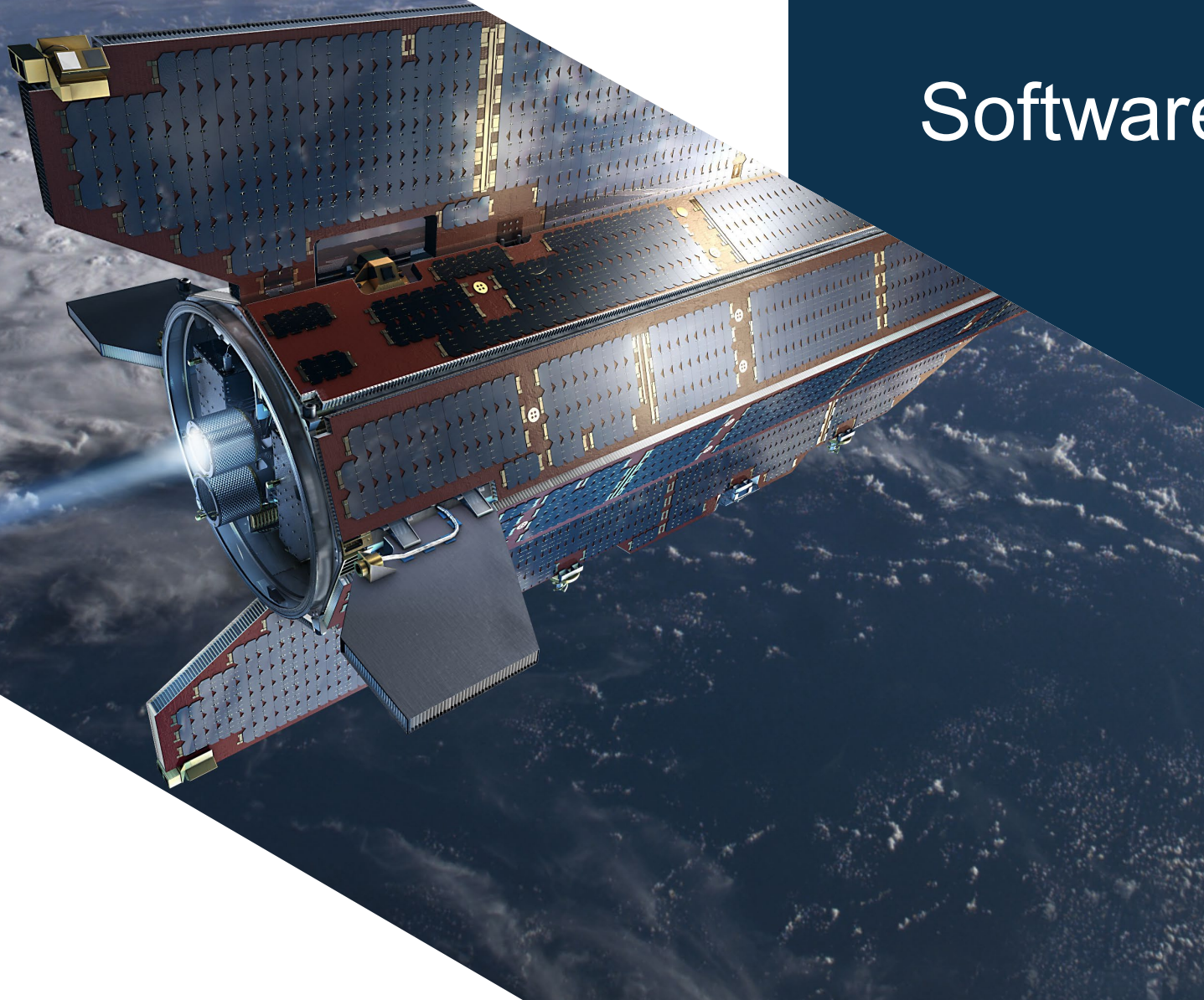
Software Development Activities

- Planning
- Implementation
- Testing and Documenting
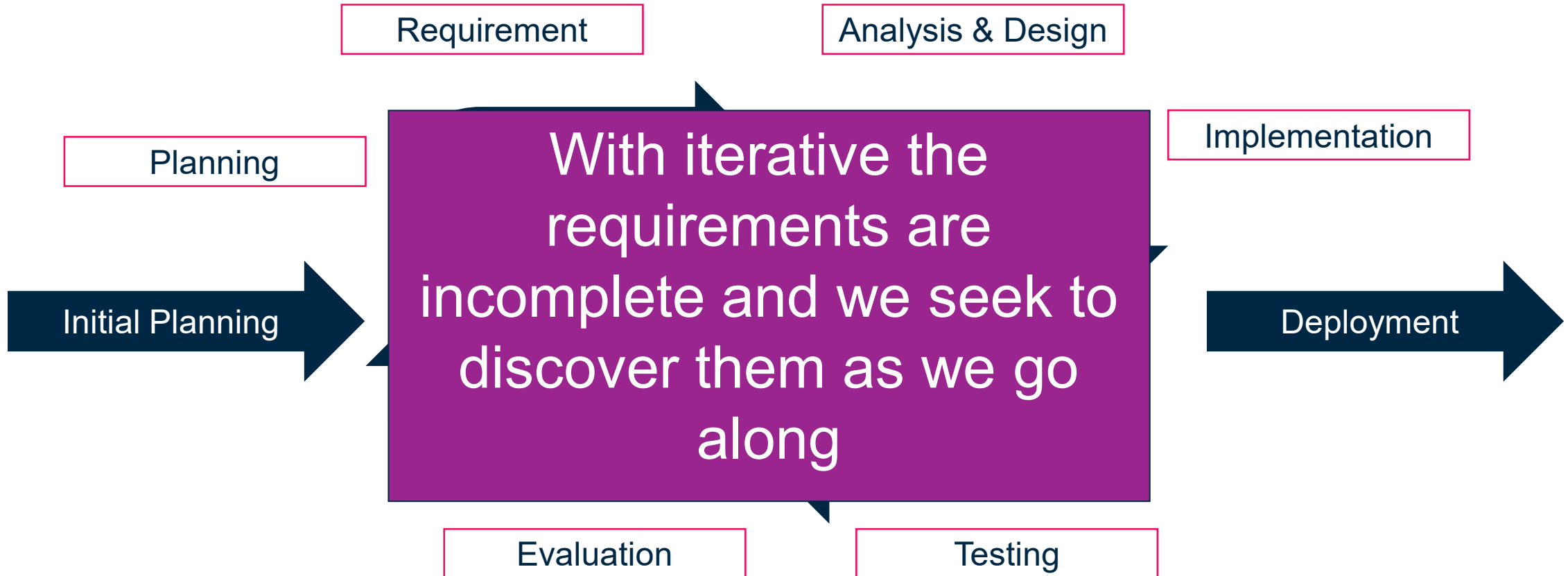- Deployment and
- Maintenance

Software Development Life Cycle –Waterfall

Requirements

Valid if all requirements can be specified before any design or implementation is conducted – and requirements do not change

Verification

Maintenance

Software Development Life Cycle –Incremental

Increment 1

System Analysis

Requirements

Detailed Design

Product

Test

Deliver

Valid if all requirements for that increment be specified before any design or implementation is conducted – and requirements do not change within an increment

Test

Deliver

Increment n

Maintenance

Software Development Life Cycle –Iterative

Requirement

Analysis & Design

Planning

Implementation

Initial Planning

With iterative the requirements are incomplete and we seek to discover them as we go along

Deployment

Evaluation

Testing

Software Development Life Cycle –Iterative

Start

Stop

Requirements Gathering and Refinement

Engineer Product

Quick Design

Refining Prototype

Build Prototype

Customer Evaluation of Prototype

Typically With evolutionary the requirements are never complete we stop when the money / time runs out

Software Lifecycle Model Usage

COMMERCIAL IN CONFIDENCE
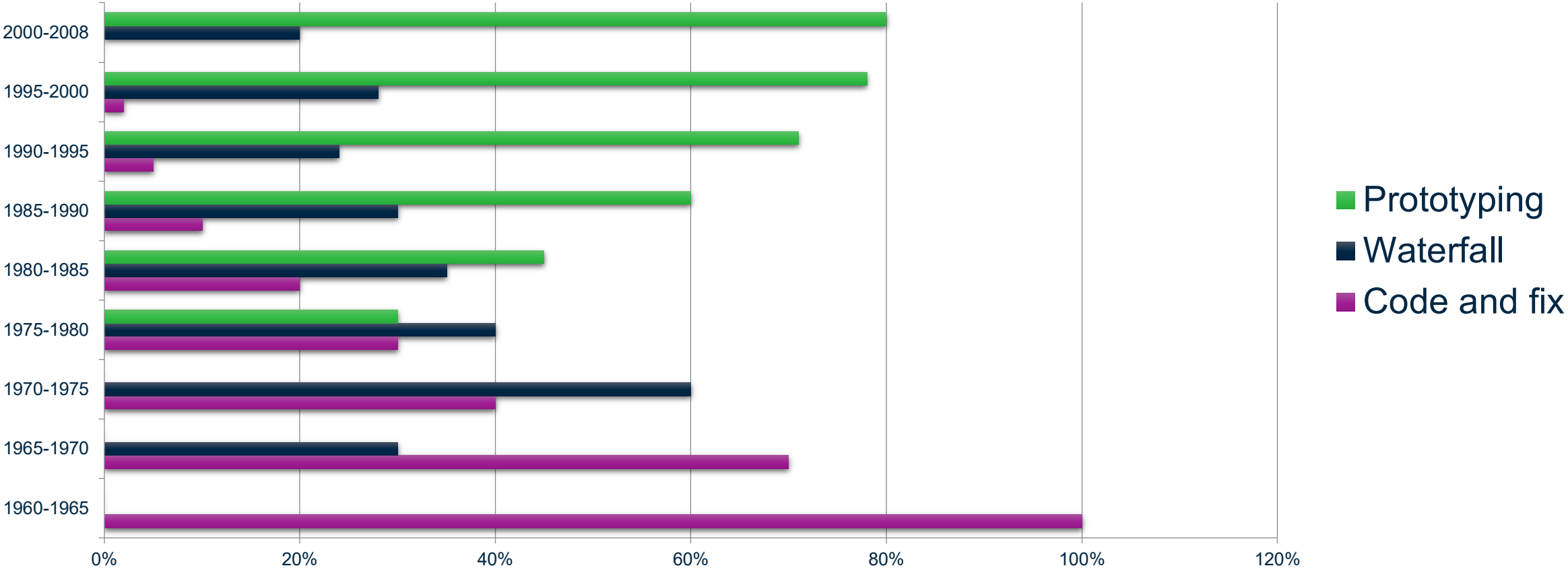
Where is the problem then?

- Text

QINETIQ

Software Estimating – Where is the problem ?

New capability in new Platform

New capability in old Platform

Old capability in new Platform

Cost Estimating

Requirements

Systems Engineers

Systems Architecture ← → Different Models (Use Case, Class, OOO, UML etc)

Software Designers/Developers

Software Tester → Software

Software

Hardware

Systems
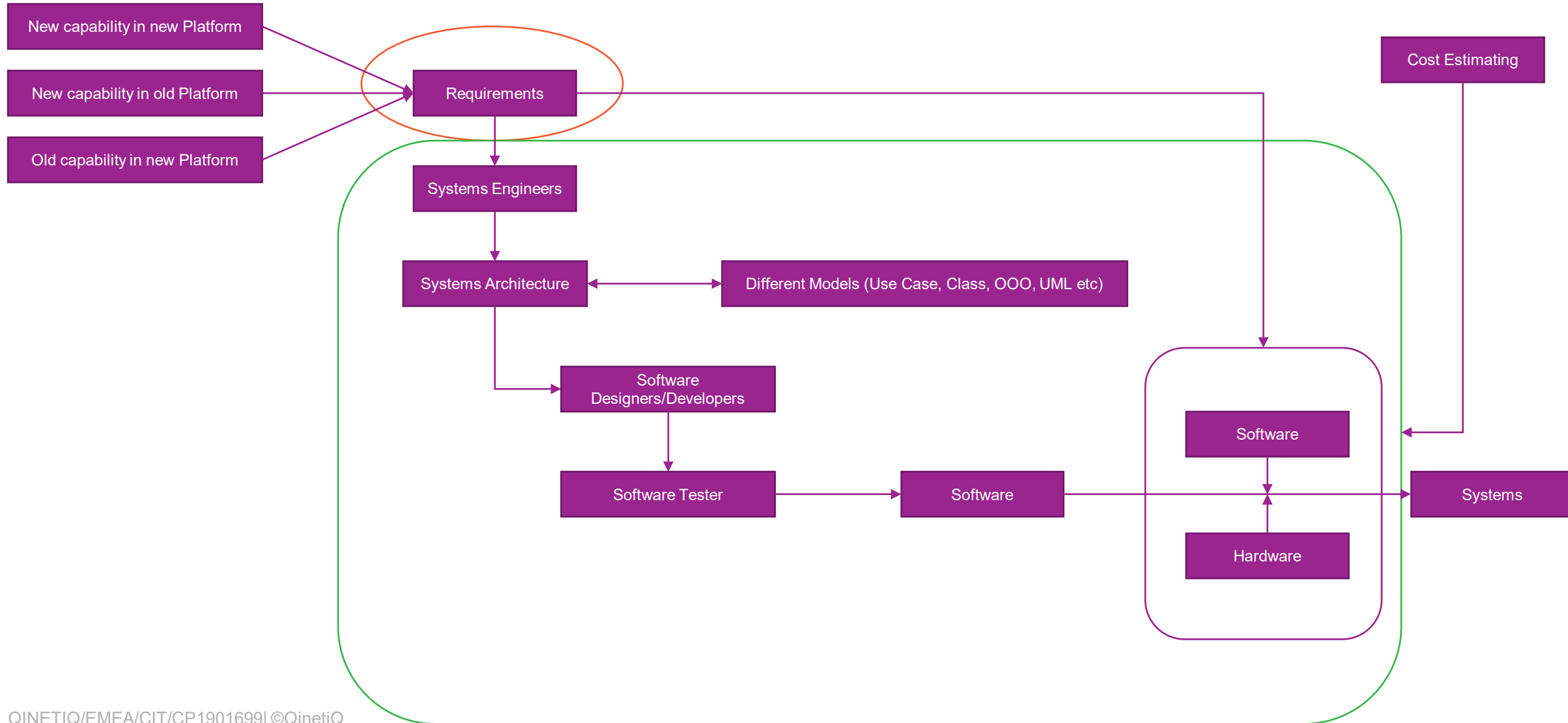
Software –Schedule Estimating Problems- (from software development Cost Estimating handbook)

More software projects have gone awry for lack of calendar time than for all other causes combined. Why is this cause of disaster so common?

- Our techniques of estimating are poorly developed. More seriously, they reflect an unvoiced assumption which is quite untrue, i.e., that all will go well

- Our estimating techniques fallaciously confuse effort with progress, hiding the assumption that men and months are interchangeable

- Because we are uncertain of our estimates, software managers often lack the courteous stubbornness of a chef. (Good cooking takes time. If you are made to wait, it is to serve better and to please the customer)

- Schedule progress is poorly monitored. Techniques proven routine in other engineering disciplines are considered radical innovations in software engineering

- When schedule slippage is recognised, the natural(and traditional) response is to add manpower. " like dousing a fire with gasoline, this makes matters worse, much worse. More fire requires more gasoline and thus begins a regenerative cycle that ends in disaster

Where is the problem

- The problem is not in the way that we estimate as we have

  – Approved sizing standard
  – Approved Software Development life cycle
  – A good understanding of how software works

Software Estimating issue is very similar to that of any hardware estimating

One of the main reasons the software projects fail is due to the uncertainty around the software project requirements

QinetiQ
Building 240
The Close
Bristol Business Park
Coldharbour Lane
Bristol BS16 1FJ
United Kingdom

Tel +44 (0)117 3172558
Mobile +44 (0)738 237 044
srajagopal@QinetiQ.com
www.QinetiQ.com

Sanathanan Rajagopal TMIET
QinetiQ Fellow
Estimating Manager –CiT

Thank you –Any Questions ?