

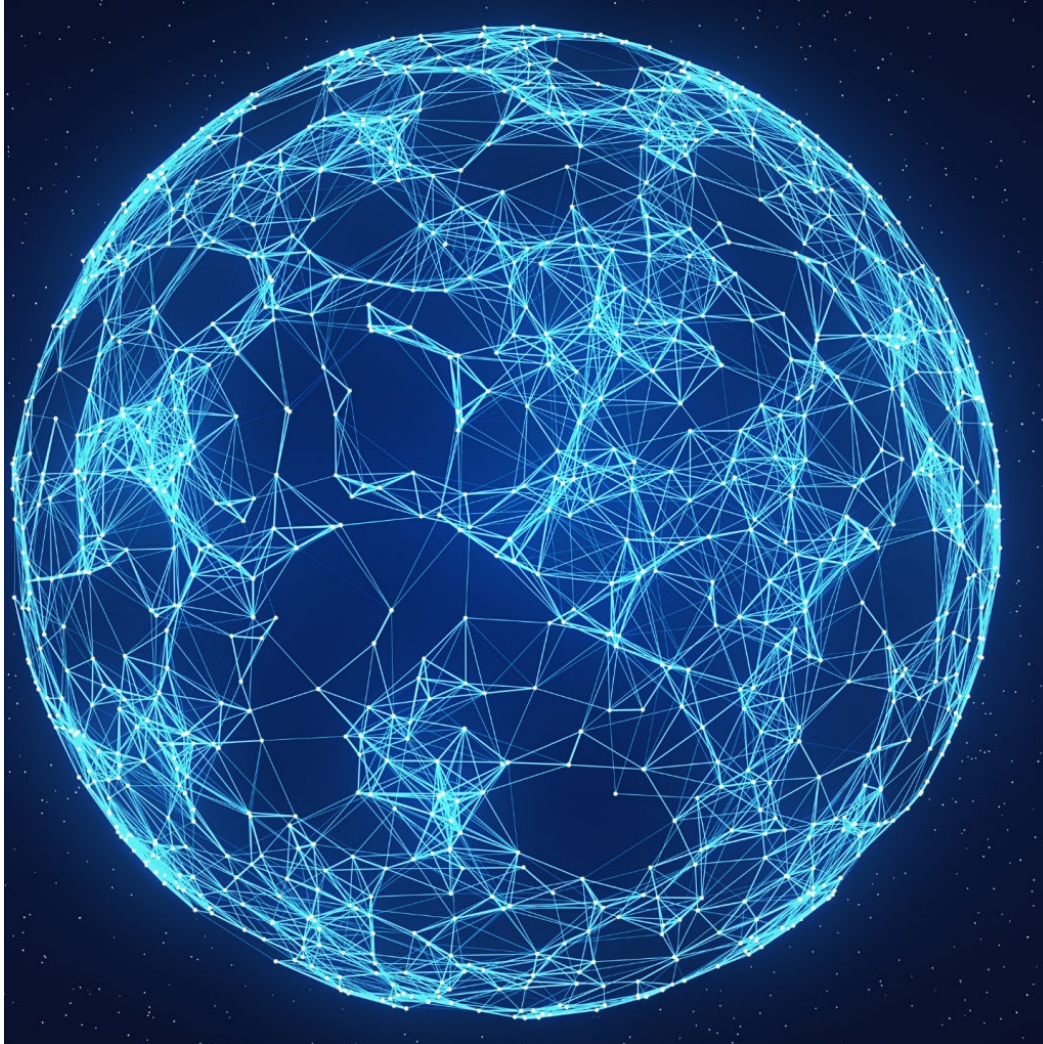
Managing Software Obsolescence: Strategies and Cost Impacts in Cost Estimation

Dr Sanathan Rajagopal

Contents

- I. Company Overview
- II. Software- Definition
- III. Software Obsolescence –Definitions
- IV. Types of Software Obsolescence
- V. Software Obsolescence Management Level
- VI. Software Obsolescence Resolution approach
- VII. Software Obsolescence Complexity Matrix
- VIII. Software Obsolescence key cost drivers
- IX. Estimate the cost of Software Obsolescence Resolution Cost
- X. Case Studies
- XI. Conclusions and Questions

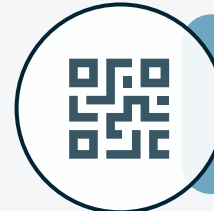
Company Overview



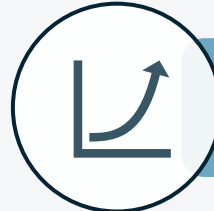
Established in 2020



**105 employees, rising to
~250 by the end of 2028**



**Sirius Digital Services and
Sirius Inc established to
diversify capability**



**Turnover more than trebled
in the past 2 years**

Scope of Services

P3M

- P3M Design, implementation & Strategic Advice
- P3M Maturity Assessments
- Change Management and transition support
- PMO Improvement initiatives
- End to end project management & project leadership
- Bid Management
- Collaboration leadership

OA / OR

- Soft methods & problem structuring
- Historical trends & data analysis
- Modelling, simulation & wargaming
- R&D, concepting, experimentation
- Capability planning
- Acquisition

Systems Engineering

- Product, service & enterprise level
- Tailorable & scalable lifecycle solutions
- Requirements, Architecting
- Systems Analysis
- Integration
- Verification
- Validation & Testing
- Sustainment
- Disposal & Replacement

Cost Analysis

- Whole life cost models
 - Should-/Could-cost
 - Cost-Benefit
 - Value-For-Money
- Cost estimation
 - Acquisition
 - Through-life
 - All domains, platforms & DLODS
- Investment Appraisals
- Risk Analysis
- Programme Mgt
- Business Cases

Approvals

- Business Case support
- 5 Case Model
- HMT Approvals
- Cost-benefit, cost-effectiveness and Option selection
- Value for Money cases
- Financial estimation
- Risk management
- Socio-economic benefits

Our Clients & Partners – trade references on request



Delivering clarity from complexity



Software



Software is defined as programmes, procedures, rules, data, and documentation associated with programmable aspects of system hardware and infrastructure (BS EN 62402:2007, 2007).

In simple terms, software is a set of programs, supported by a manual, which will make the functionality of system or hardware simpler and easier to use.

Software Obsolescence

- Software obsolescence happens when the original developer and authorised third-party cease to provide support with regular updates, upgrades, and fixes or due to changes in the target environment, systems, and hardware, which makes software unusable.

- S Rajagopal et al, 2015



Types of Software Obsolescence (Sandborn, 2007)

Software gets obsolete due to one of the following causes

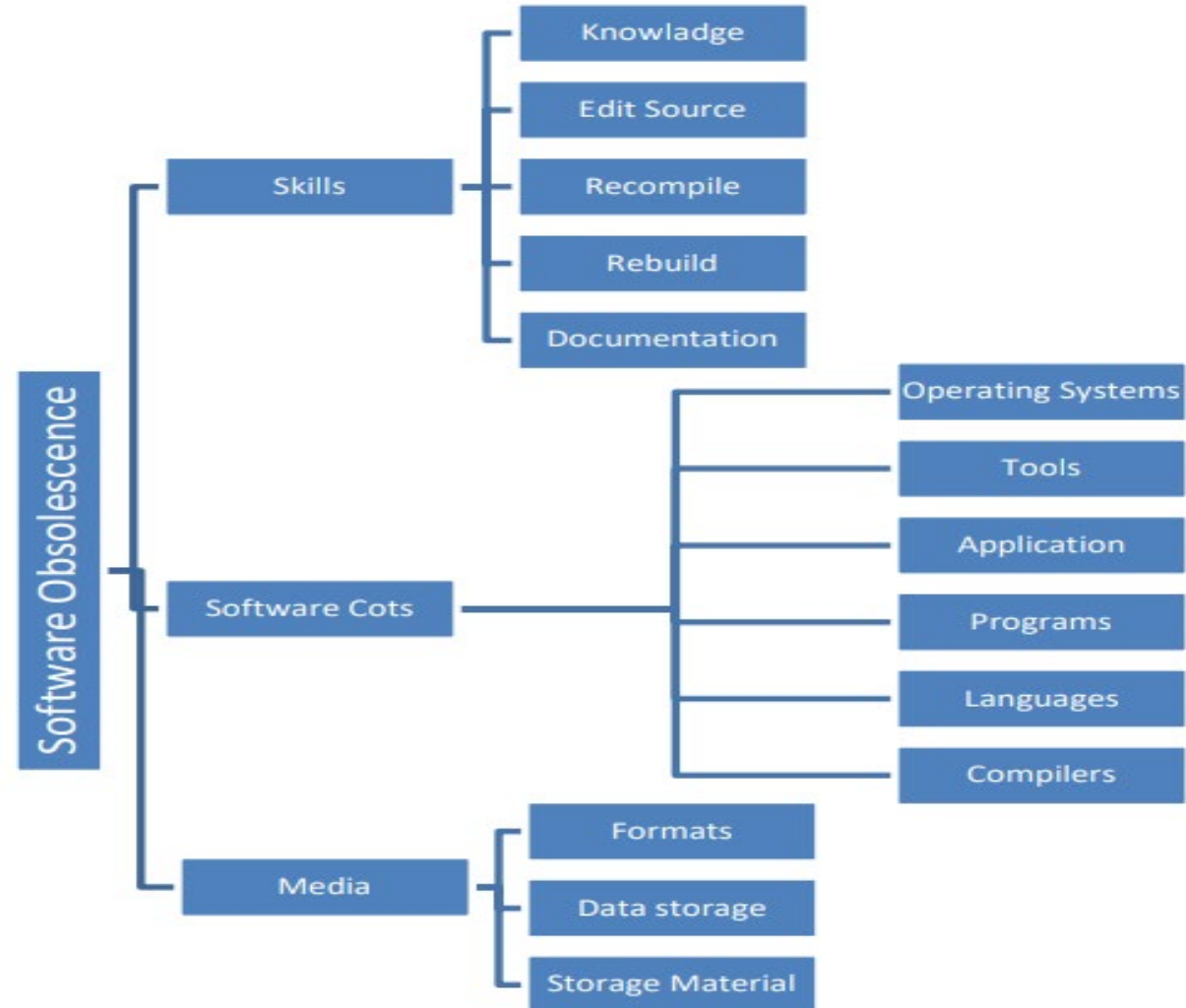
Functional Obsolescence: If there are changes to the hardware, system, or other software in the same system. This will behave like a ripple effect as changes in one software or hardware will affect the next one and so on until it affects the entire system or capability.

Technological Obsolescence: This happens when vendor stops supporting the products or unavailability of the software in market etc.

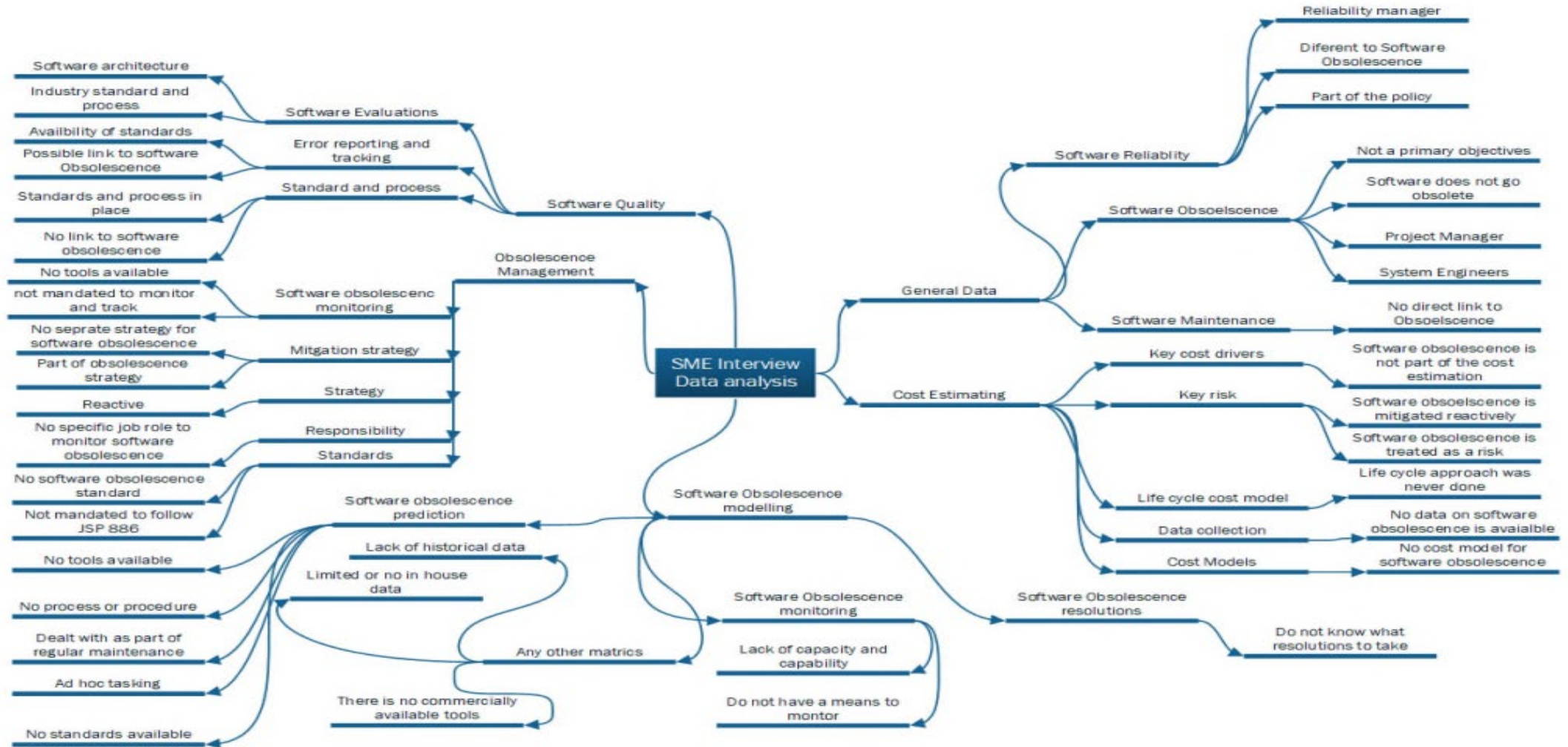
Logistical Obsolescence: This happens when the media or the hard drive for example does not support the software. For example, some of the latest software may not function on old processors



Types of Software Obsolescence (Javier Romero Rojo, 2011)



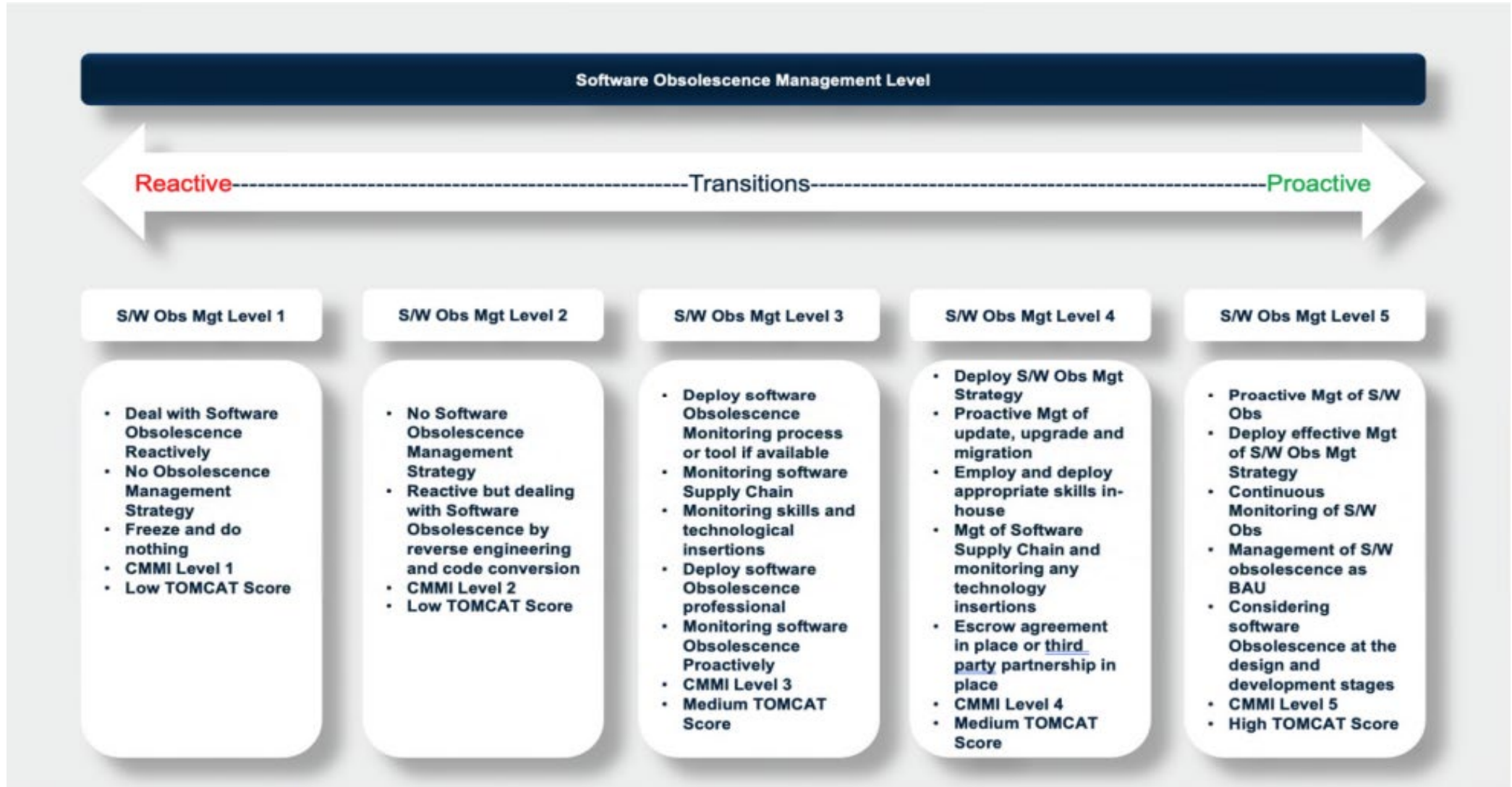
Data Analysis



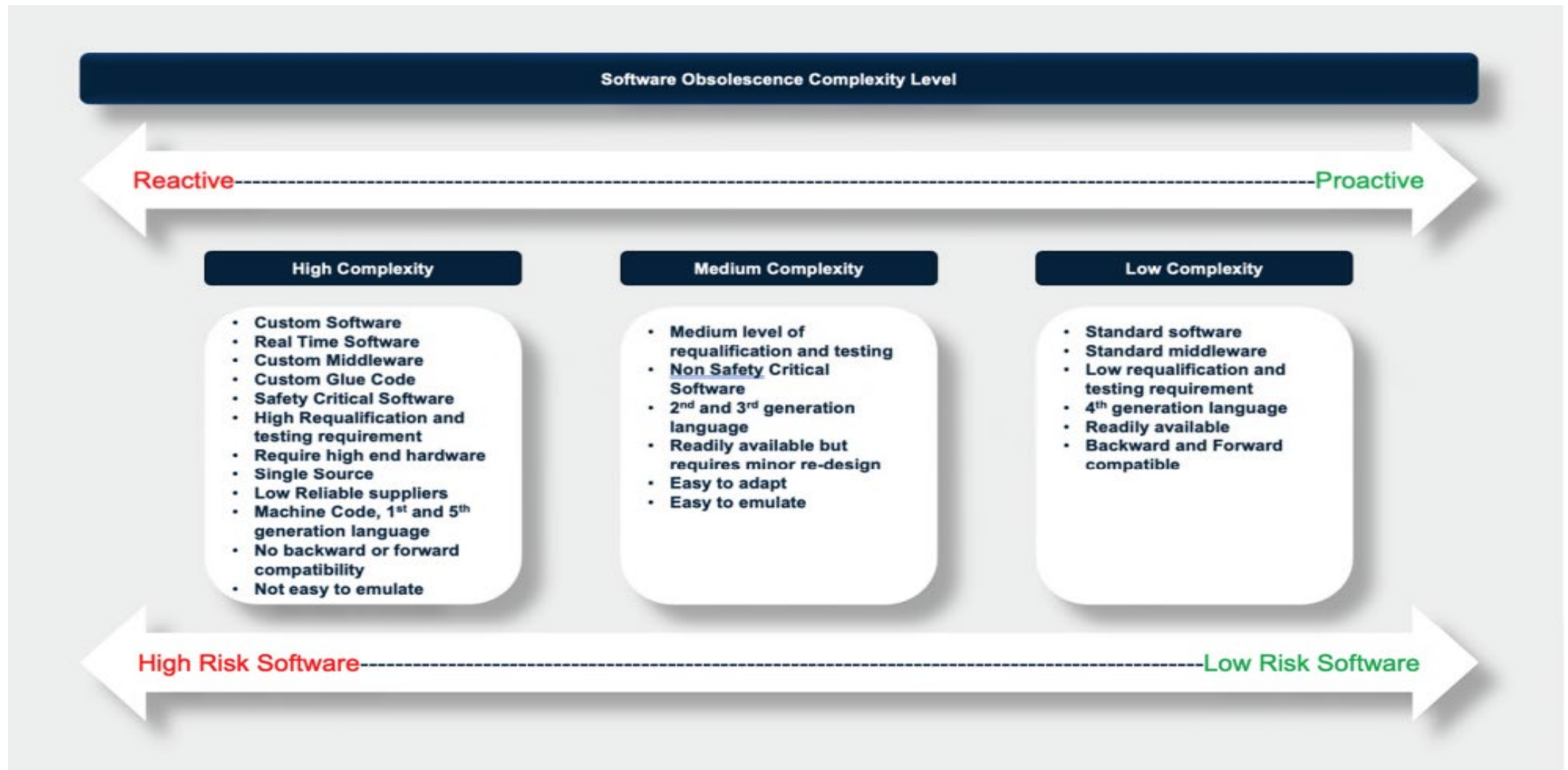
What are you estimating in Software Obsolescence ?

- Need to identify how to manage software obsolescence ?
 - Software Obsolescence Management Plan
- Need to identify how risky/Complex/probability of Software Obsolescence ?
 - Software Obsolescence Complexity Matrix
- We now need to identify appropriate resolution strategy
 - Software Obsolescence Resolution Strategy
- We then need to identify the key cost drivers
 - Software Obsolescence Key cost Drivers

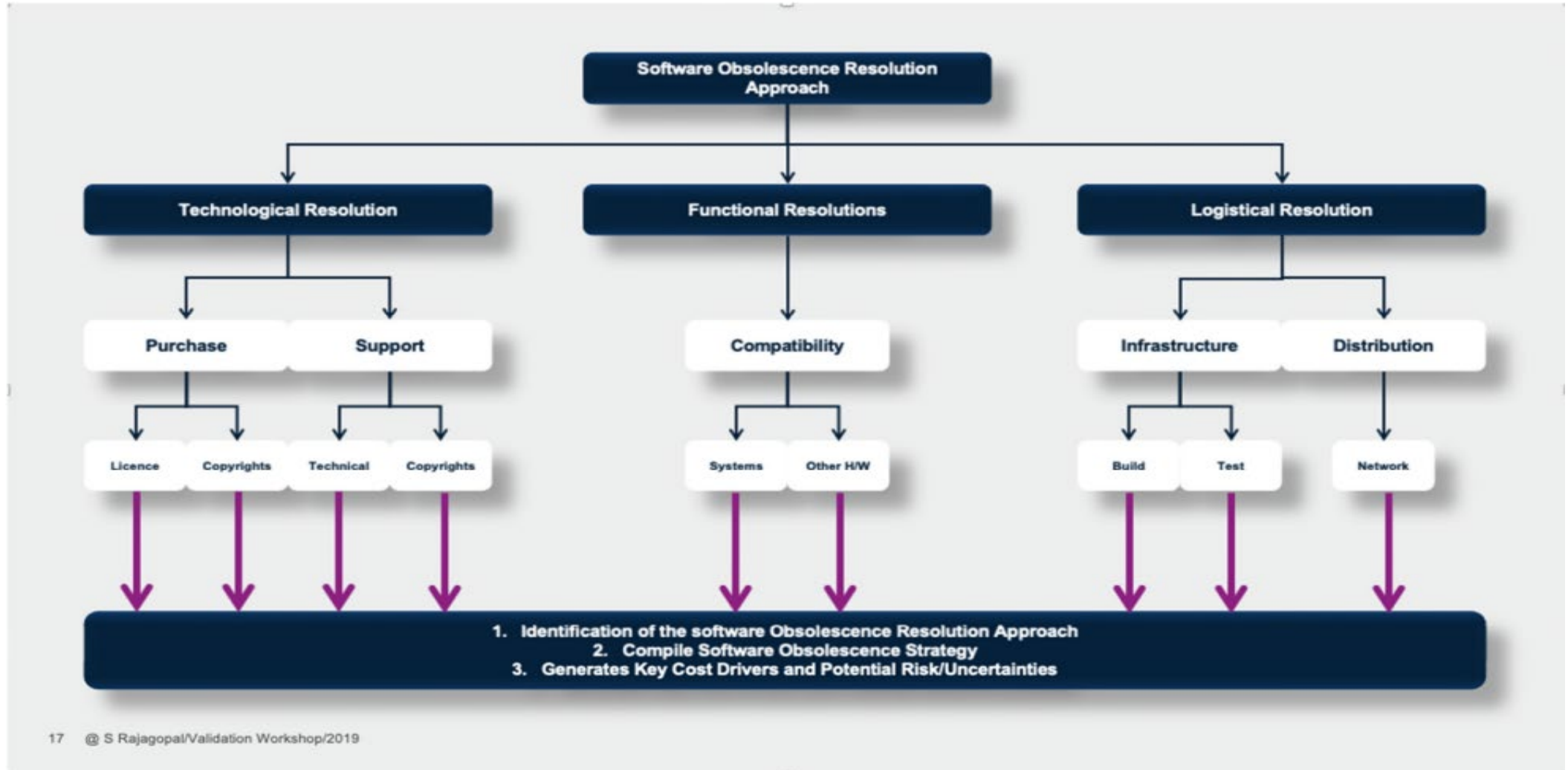
Software Obsolescence Management Level



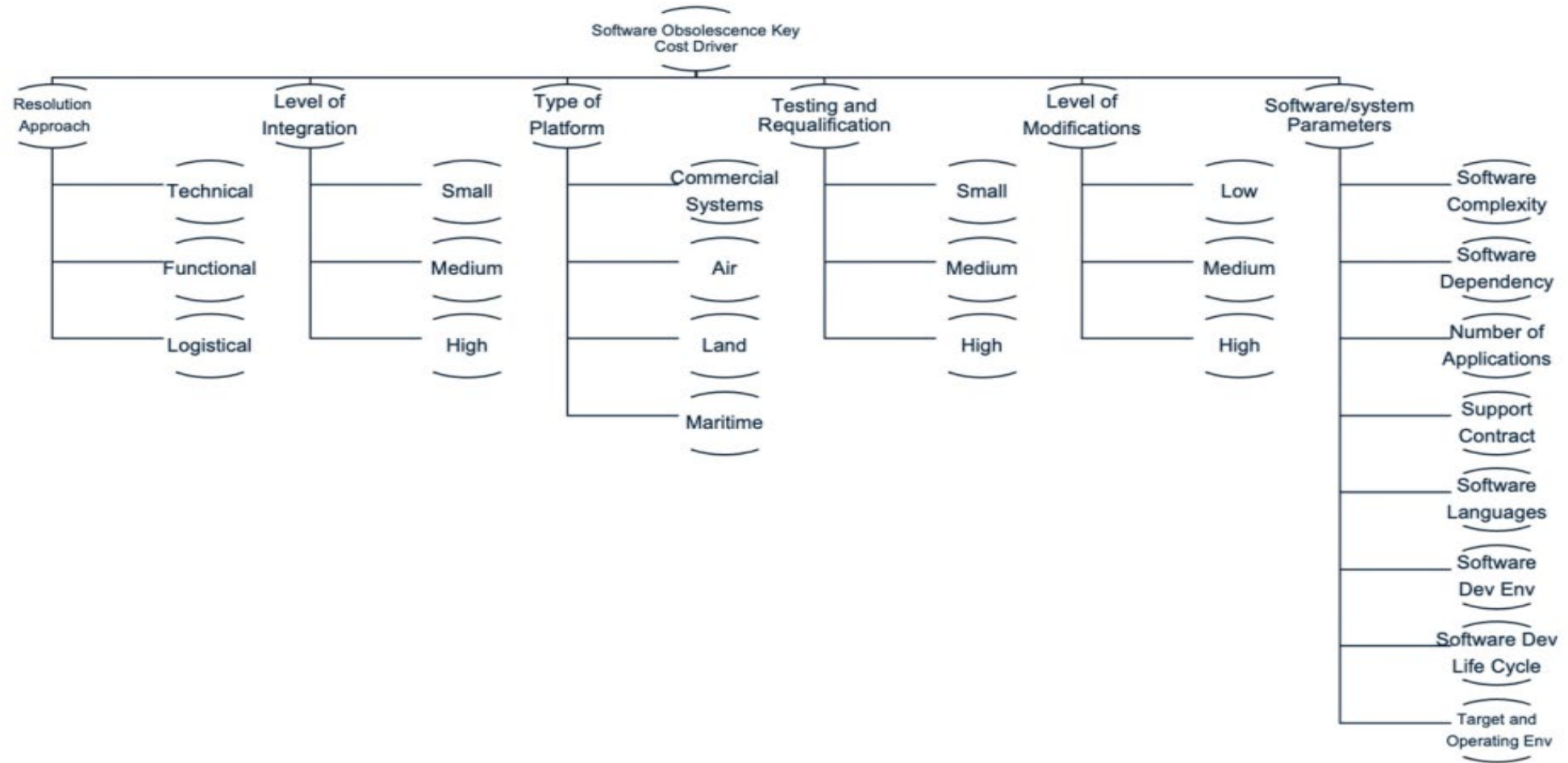
Software Obsolescence Complexity Matrix



Software Obsolescence Resolution Approach



Software Obsolescence Key Cost Drivers



Cost of Resolutions

Formula for calculating the resolution cost is shown below.

For each resolution identified, the cost of resolution is calculated as

$$RCost = \sum_{i=1}^n Rt_i \times Rr_i \times E_i$$

Where,

RCost = Resolution Cost (£),

n = Number of resources(FTE)

Rt = Resource type

Rr = Resource rate (£)

E = Effort (Months)

Total cost of resolution

Other key cost drivers and cost elements are added as a constant to the formula to get the total cost of software obsolescence. These constants were postulated as a result of literature reviews, interviews and case studies. These constants need to be matured and calibrated further over time as and when more actual data on software obsolescence is made available.

$$SOBCost = [(m_1Tp \times m_2In) + (m_3Tr \times m_4M)] \times m_5Ap \times RCost$$

Were,

SOBCost = Software Obsolescence Cost (£)

$m_n = \text{Constant}$

$Tp = \text{Type of platform}$

$In = \text{Level of integration}$

$Tr = \text{Testing and requalifications}$

$M = \text{level of modifications}$

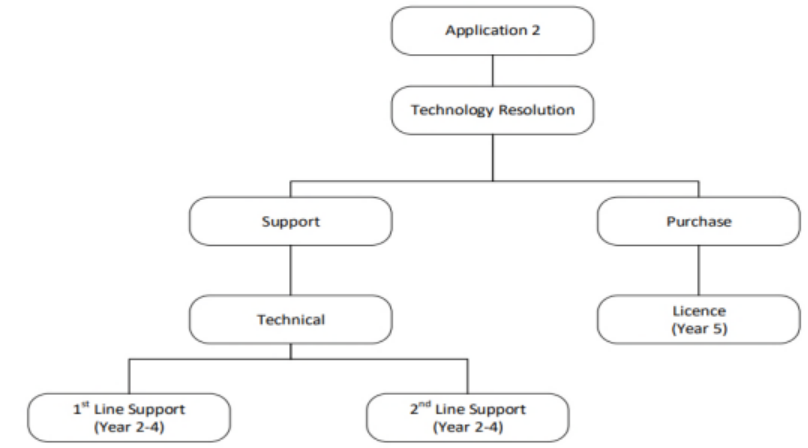
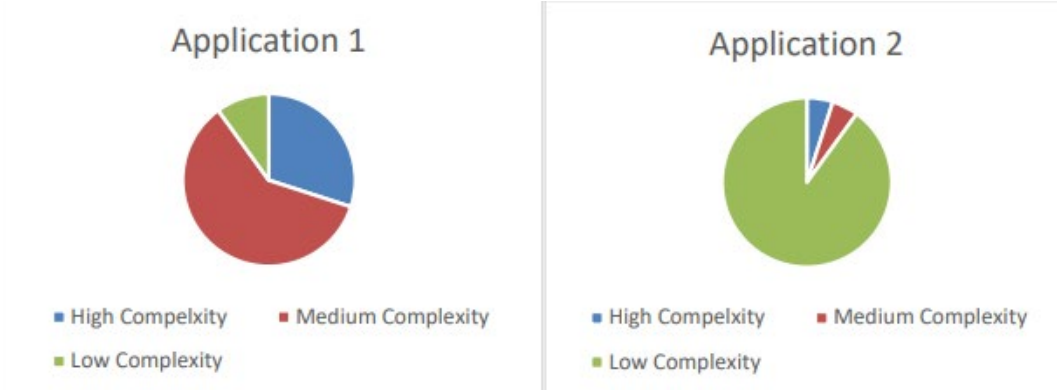
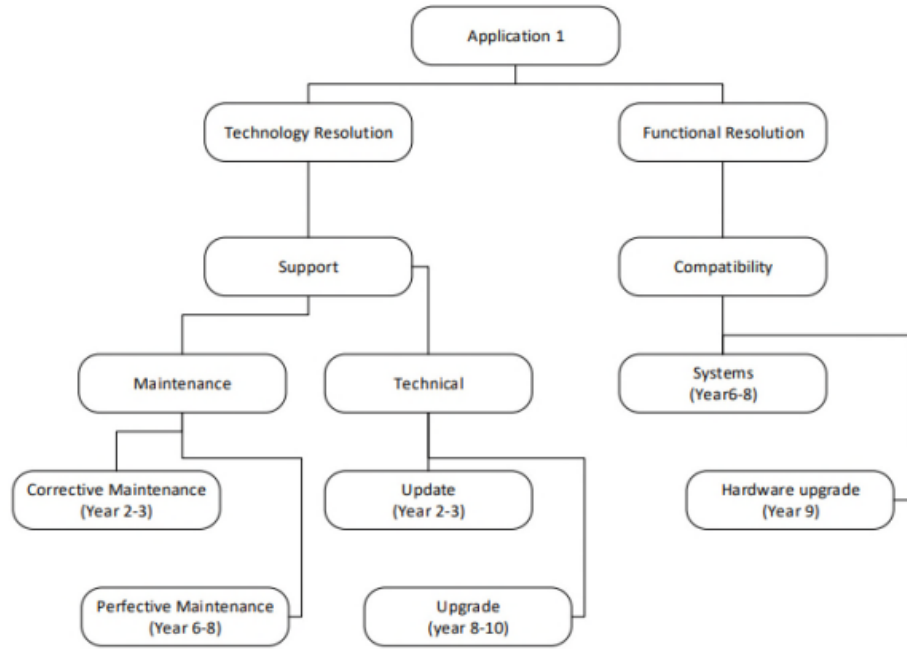
Case Study A (Air Platform)- Example

Data analysis software for a radar system

6 years + 4 years if life cycle

Descriptions	Inputs
Type of platform	Air
Level of integration	Medium
Testing and requalification	High
Level of modification	Medium
Number of applications	2

Case Study 1



Years	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6	Year 7	Year 8	Year 9	Year 10
Resolution Approach										
Functional Resolution										
Technological Resolution										
Logistical Resolution										

Summary of the findings of all case studies

Case Study	Contract Duration	Estimated Software Obsolescence Resolution Cost	Contractual/budgeted software obsolescence Cost	Difference
Project A	10	£7.5M	£6.2M	£1.3M
Project B	10	£5.8M	£5.1M	£0.7M
Project C	7	£18.5M	£20.1M	-£1.6M
Project D	7	£4.8M	£5.2M	-£0.4M
Project E	10	£3.1M	£2.9M	£0.2M

Conclusion

- Software obsolescence is inevitable
- Need to understand how we manage them and predict when software obsolescence will occur.
- Need to have a framework that will allow end to end management of the software obsolescence which will estimate the cost of each resolutions.
 - This was discussed today
- Further details of this research is available online. ([Development of a framework to estimate the software obsolescence resolution cost of custom build real-time software.](#))

LEGACY LOGIC

GOOD NEWS!
WE'RE SAVING MONEY
BY NOT UPGRADING
THE 2003 ACCOU-
NTING SYSTEM.



OF COURSE! WE
ONLY SPEND \$0.00
ANNUALLY ON
UPGRADES.

WHAT ABOUT THE TEN
CUSTOM PATCHES,
SECURITY WORKAROUNDS,
AND THE \$50,000
CONSULTANT WHO
SPEAKS COBOL?



I LABELLED THAT
'INNOVATION
OVERHEAD" AND
PUT IT IN
MARKETING'S
BUDGET

LEGACY
SOFTWARE
NEVER DIES
-IT JUST
REASSIGNED

