# Cost of Software Obsolescence Resolution

S Rajagopal- QinetiQ Fellow
Estimating Manager- Cyber, Information and Training

**13th June 2018**
**ICEAA Professional Development & Training Workshop**

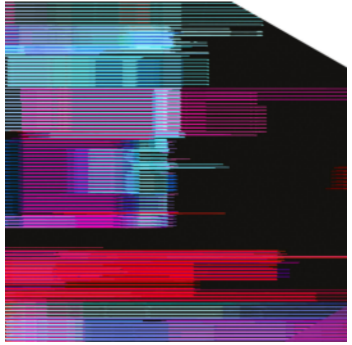
# QinetiQ Overview

## Air
Providing support across every aspect of air, built on 70 years of experience.

## Weapons
Creating safe and effective military components and testing them before and during service.

## Robotics & Autonomy
Developing robotics to keep military personnel and first responders away from danger.

## Land
Making significant scientific and technological innovations in order to evolve physical protection systems.



## Cyber
Protecting and improving cyber resources to overcome challenges and threats.

## Training
Delivering realistic scenarios via simulation and virtual reality to prepare people for every eventuality.

## Maritime
Designing and testing innovative naval systems and components.

## Space
Building, launching and operating complex space infrastructure.

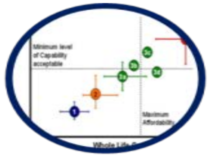
## Advisory Services
Acting as an impartial partner for customers looking to procure services and develop capabilities.

# QinetiQ, International Business, Advisory Services

Requirements & Acceptance

WLC Modelling

Options Analysis


COEIA

Through Life Support

Benefits Analysis

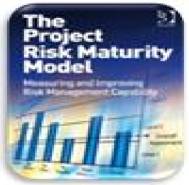
Strategy Devt / Wargaming

Technology Management

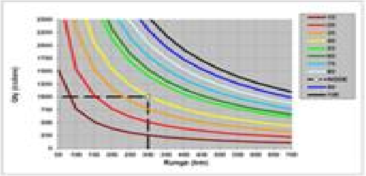
Legacy & Market Surveys

Risk Management

FACET
(Family of Advanced Cost Estimating Tools )



RMM
(Risk Maturity Model)


Joint Utility Model

EVC
(Economic Value Chains)

Industry Standard Toolsets

- Over 150 highly skilled and experienced subject matter experts
- 70%+ Professional Accreditation
- ACostE, SCAF, APM, Prince2, MSP, ICEAA
- Based across 5 UK Sites
- Deployed internationally
- Average experience of 10 years
- Over 40% PhD / MSc qualified

# Agenda

| | |
|---|---|
| 1 | Research Aim |
| 2 | Definitions |
| 3 | Introductions |
| 4 | Software Obsolescence Cost Analysis Framework |
| 5 | Software Obsolescence Resolution Cost Optimisation Model |
| 6 | Summary |
| 7 | Conclusion |

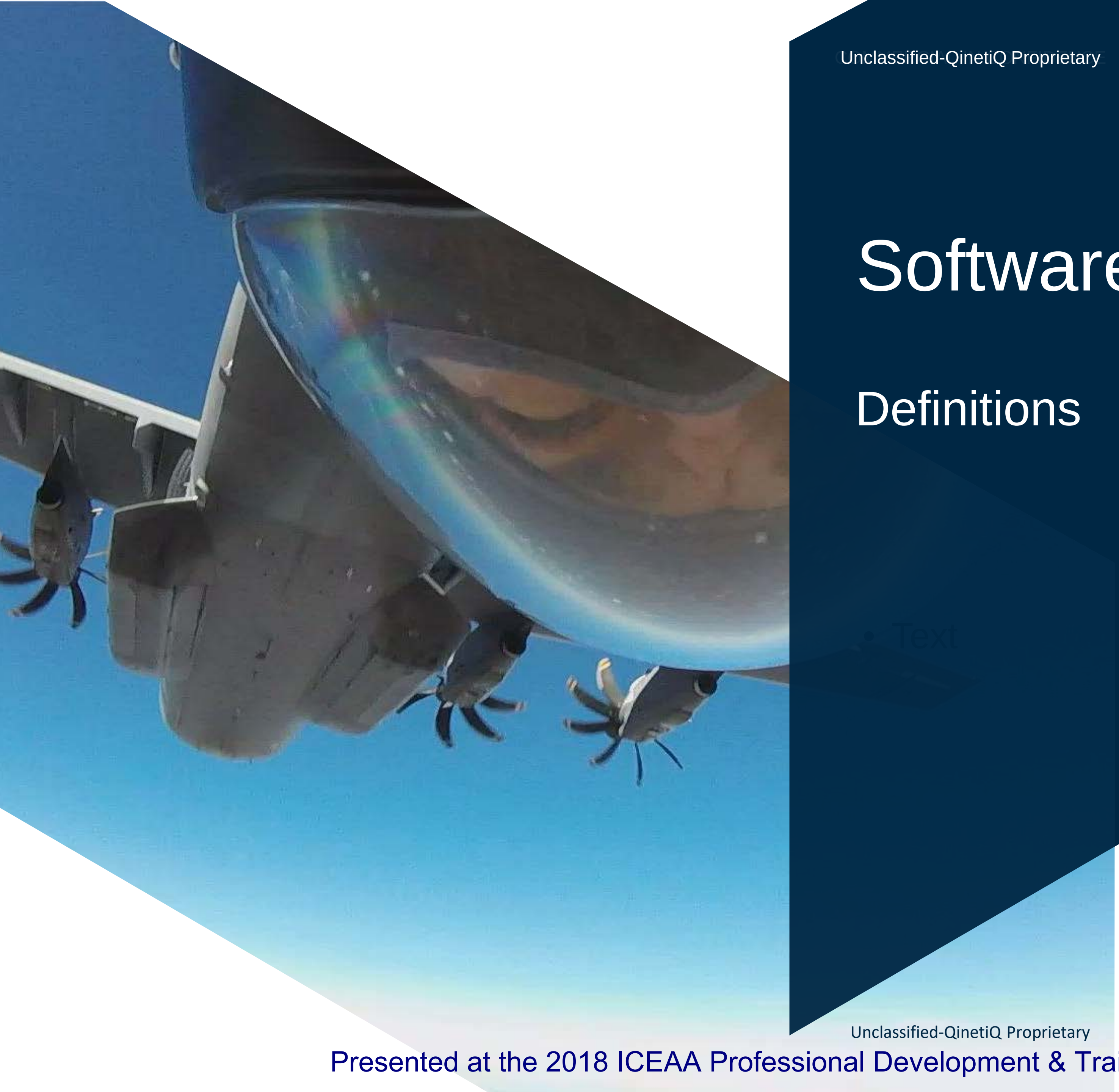
# Software Obsolescence

## Research Aim

- Text

# Research Aim

“To develop a cost analysis framework to estimate the cost of Software Obsolescence Resolution of a bespoke real-time software in defence and aerospace”



# Software Obsolescence

## Definitions

- Text

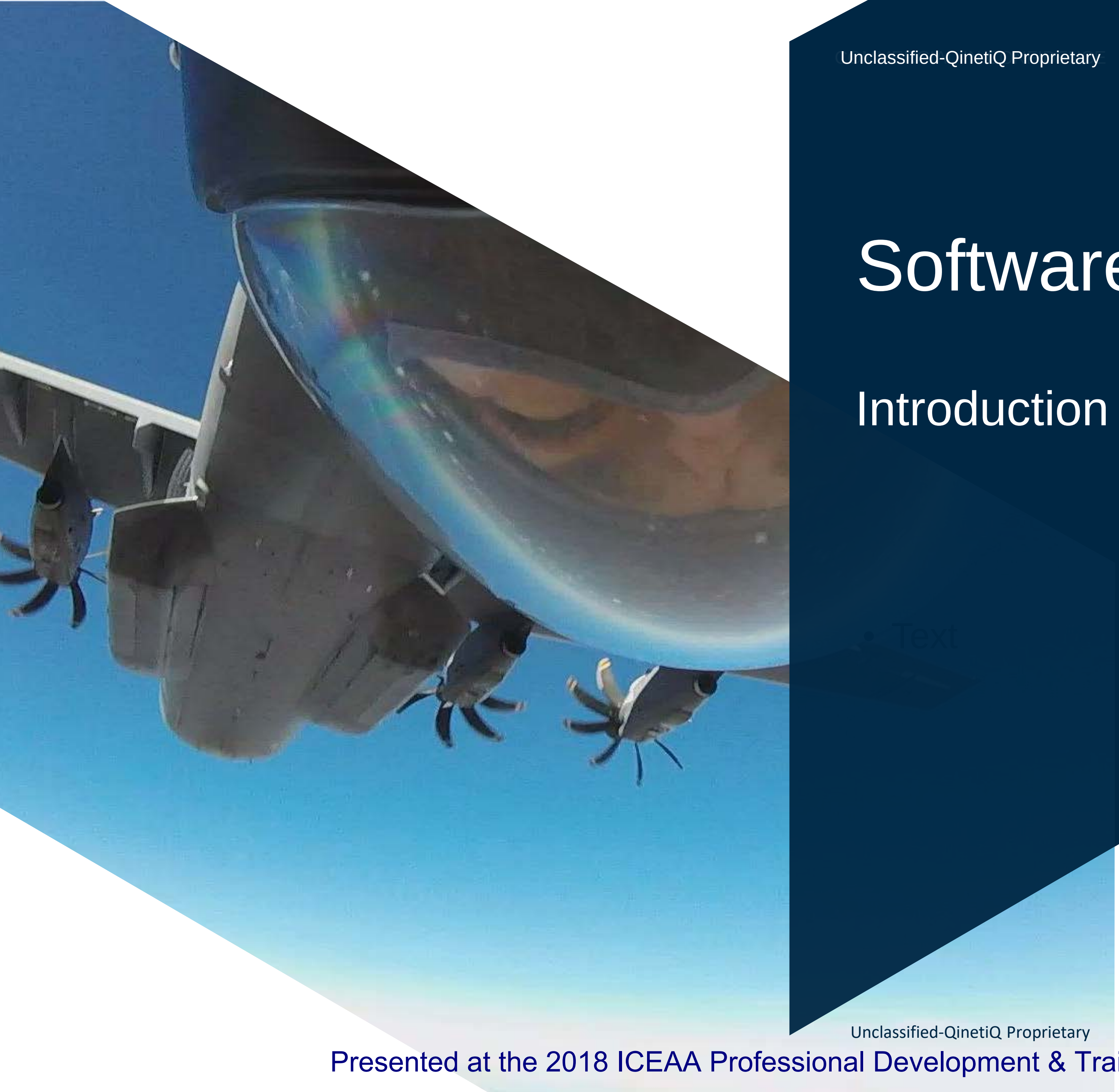
# Software Obsolescence Definitions

Software Obsolescence is defined as “ what happens when the original and authorised third party ceases to provide support with regular update, upgrade, fixes or due to the changes in target or operating environment, systems or hardware which makes the software unusable”

-S Rajagopal et al; (2014)

# Software Obsolescence vs Software Maintenance

| Software Maintenance | Software Obsolescence |
|---|---|
| Bug fixes | Replacement of entire application if need be to a new one |
| To address fault/Failures, security patches etc. | To address the issues with the application in totality |
| Maintenance is the review of the stored files to ensure they are still useable | Solves unavailability of fixes, licenses, permission and upgrades |
| Software maintenance takes care of the current versions to ensure that its up and running and meeting the requirements | Software Obsolescence management looks forward the industry standards and other software to continue supportability of the software |
| Maintenance deals with the upgrading the software to enhance capability | Obsolescence management deals with enforced changes in the environment |



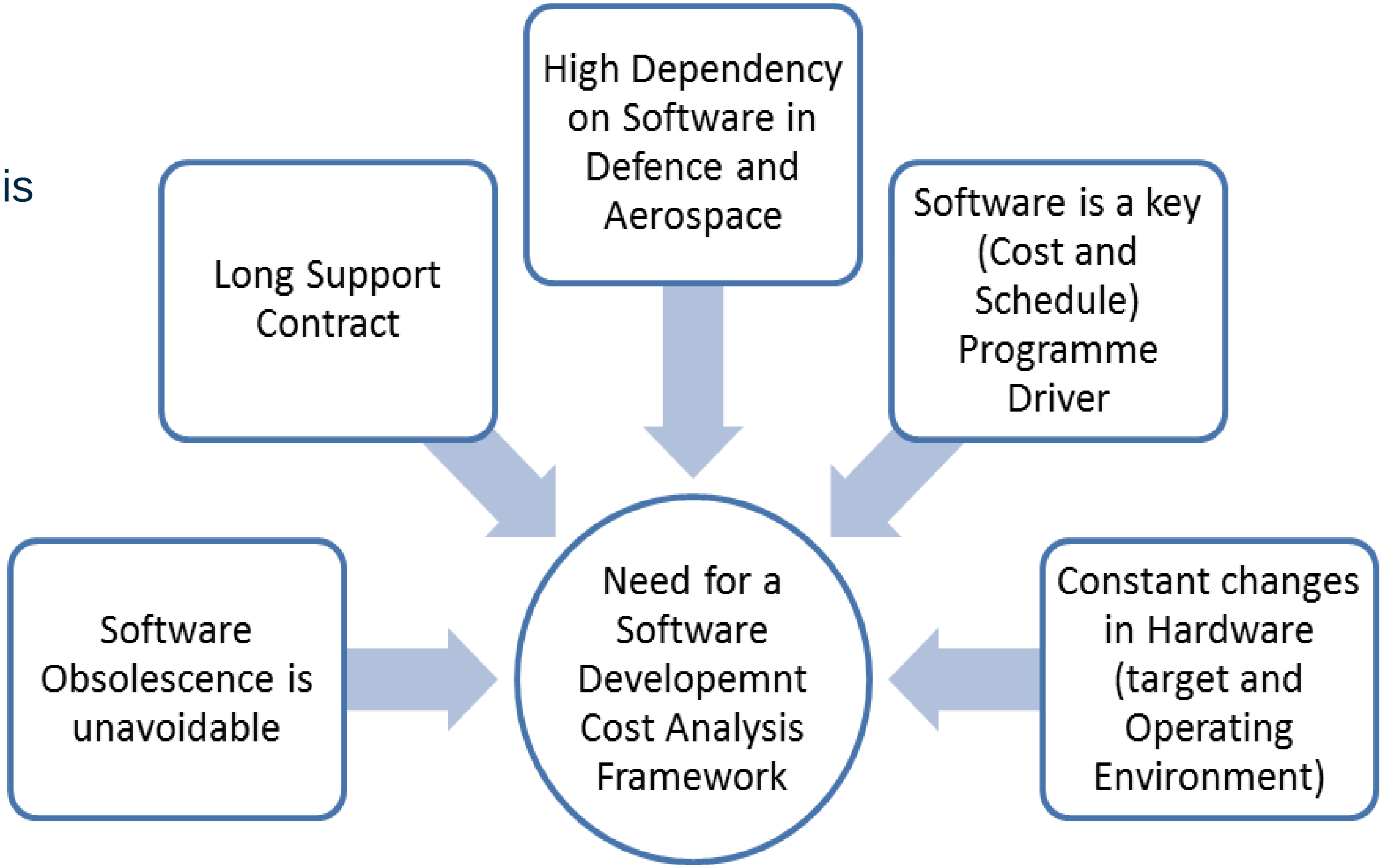
# Software Obsolescence

## Introduction

- Text

# Introduction

The need for a Software Obsolescence Cost Analysis Framework



High Dependency on Software in Defence and Aerospace

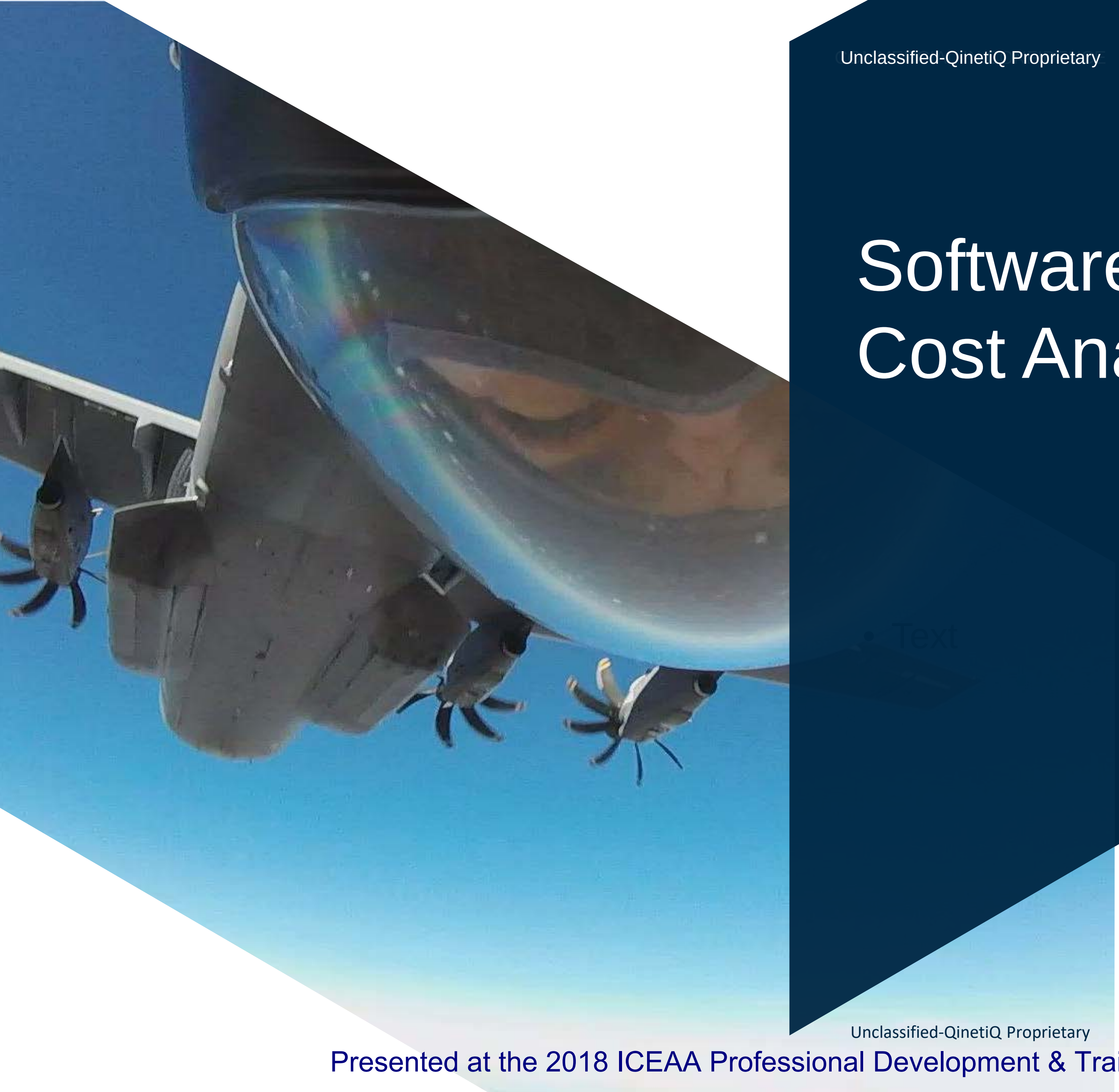
Long Support Contract

Software is a key (Cost and Schedule) Programme Driver

Software Obsolescence is unavoidable

Need for a Software Developemnt Cost Analysis Framework

Constant changes in Hardware (target and Operating Environment)

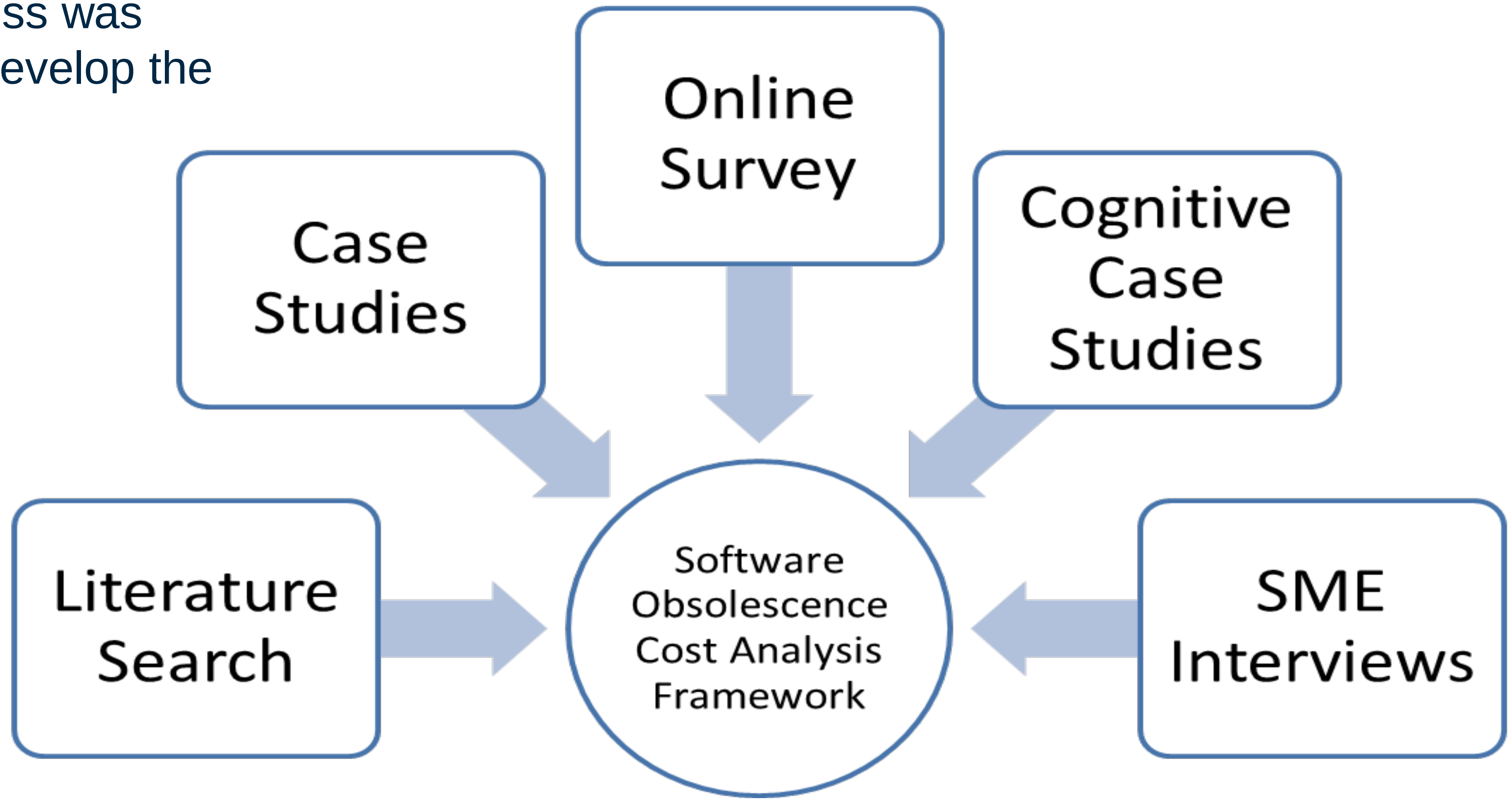


# Software Obsolescence Cost Analysis Framework

- Text

# Software Obsolescence Cost Analysis Framework

Following process was undertaken to develop the Framework



Online Survey

Case Studies

Cognitive Case Studies

Literature Search

Software Obsolescence Cost Analysis Framework

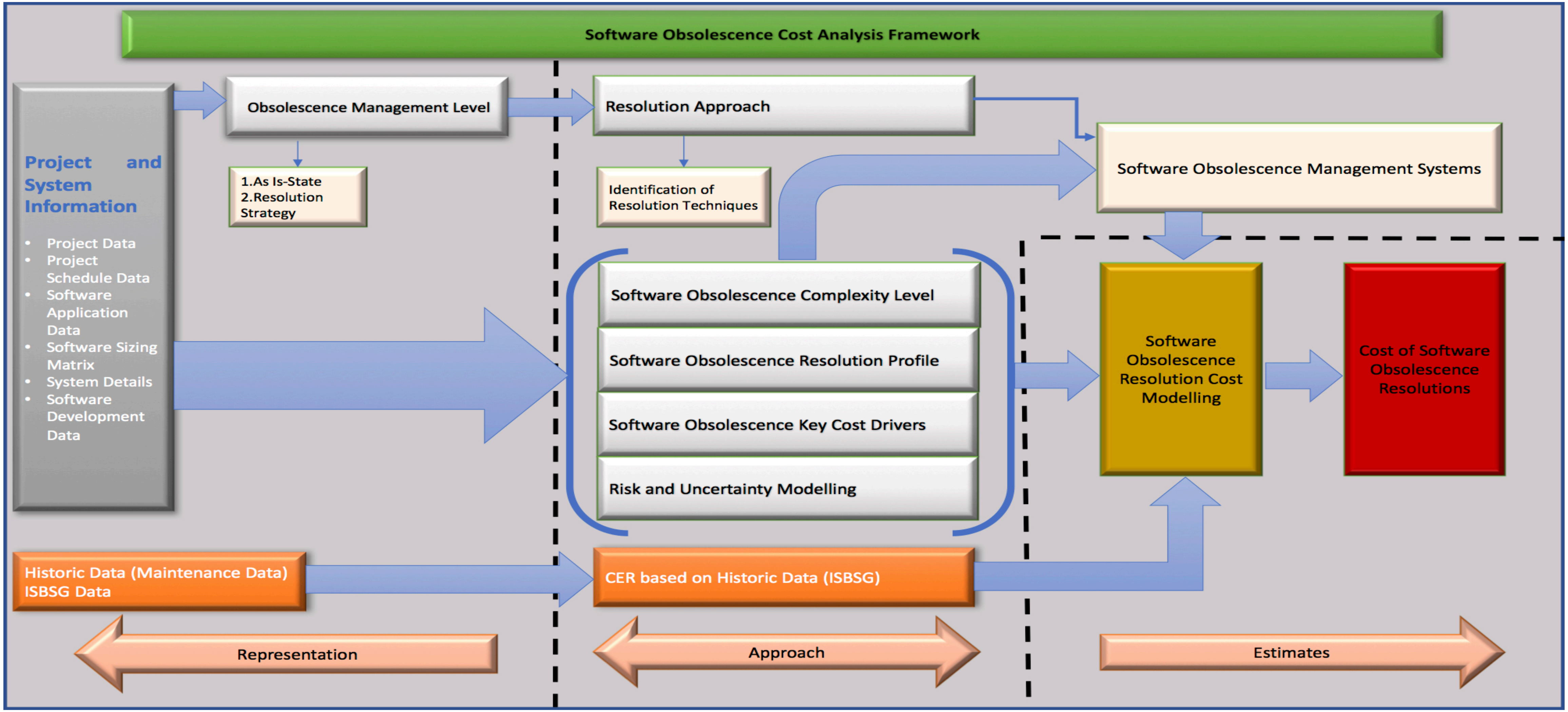
SME Interviews

# Software Obsolescence Cost Analysis Framework

## The framework has the following attributes

- This framework is in its final iterations.
- This framework's foundation is based on the Literature Searches, Case Studies, Online Survey results, SME Interviews and Cognitive Case Studies.
- This framework has several attributes that can be mapped across from and to, to the software estimating principals.
- This framework looks at the Cost Risk and Uncertainty which is at its development stage.

# Software Obsolescence Cost Analysis Framework


Software Obsolescence Cost Analysis Framework
Project and System Information
Project Data
Project Schedule Data
Software Application Data
Software Sizing Matrix
System Details
Software Development Data
Obsolescence Management Level
1.As Is-State
2.Resolution Strategy
Resolution Approach
Identification of Resolution Techniques
Software Obsolescence Management Systems
Software Obsolescence Complexity Level
Software Obsolescence Resolution Profile
Software Obsolescence Key Cost Drivers
Risk and Uncertainty Modelling
Software Obsolescence Resolution Cost Modelling
Cost of Software Obsolescence Resolutions
Historic Data (Maintenance Data)
ISBSG Data
CER based on Historic Data (ISBSG)
Representation
Approach
Estimates

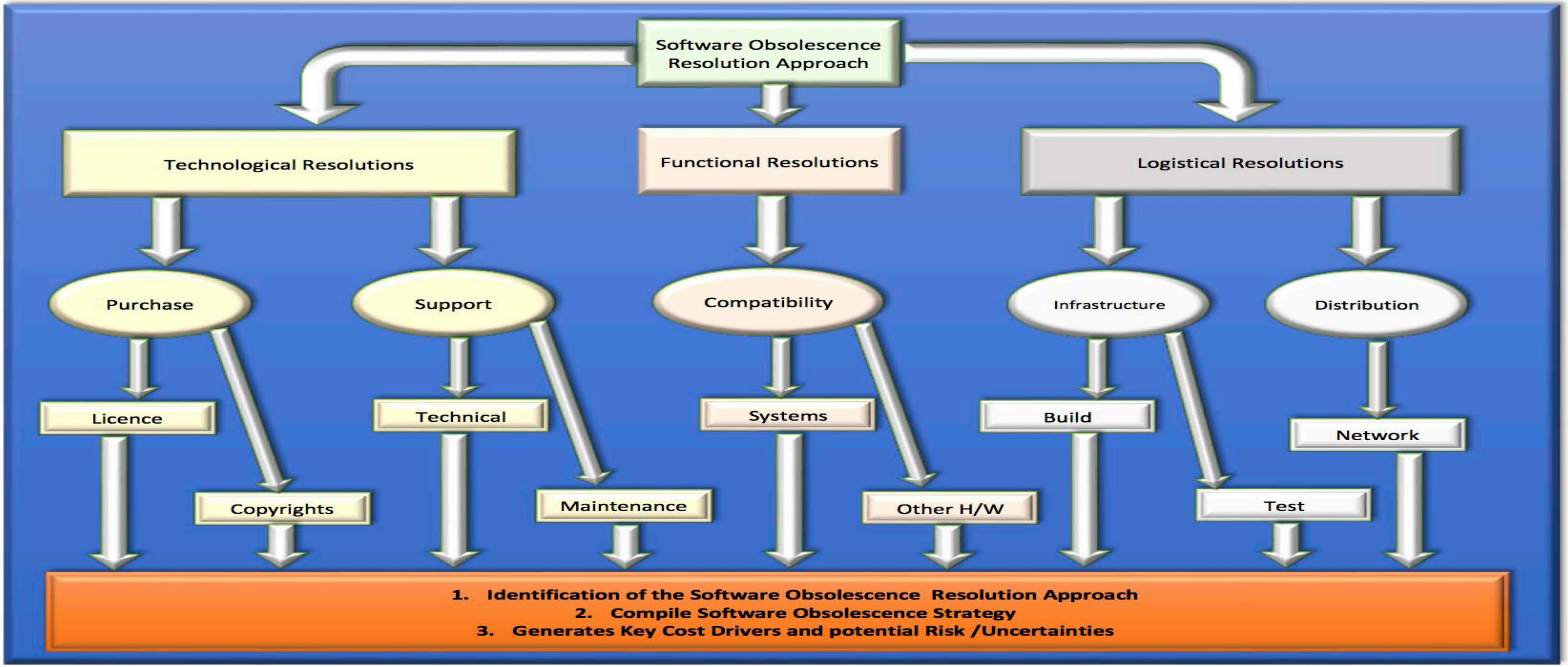
# Software Obsolescence Resolution Approach

- Software Obsolescence Resolution Approach helps to tease out the resolution strategy.
- The resolution approach will help in identification of the best possible resolution techniques.
  - Resolution techniques are determined based on the Project and Systems Parameters and Software Management Level
  - Resolution techniques are identified for individual software component rather than software program as a whole.
- Software Obsolescence Resolution Approach helps the project team to compile an appropriate software obsolescence strategy.

# Software Obsolescence Resolution Approach

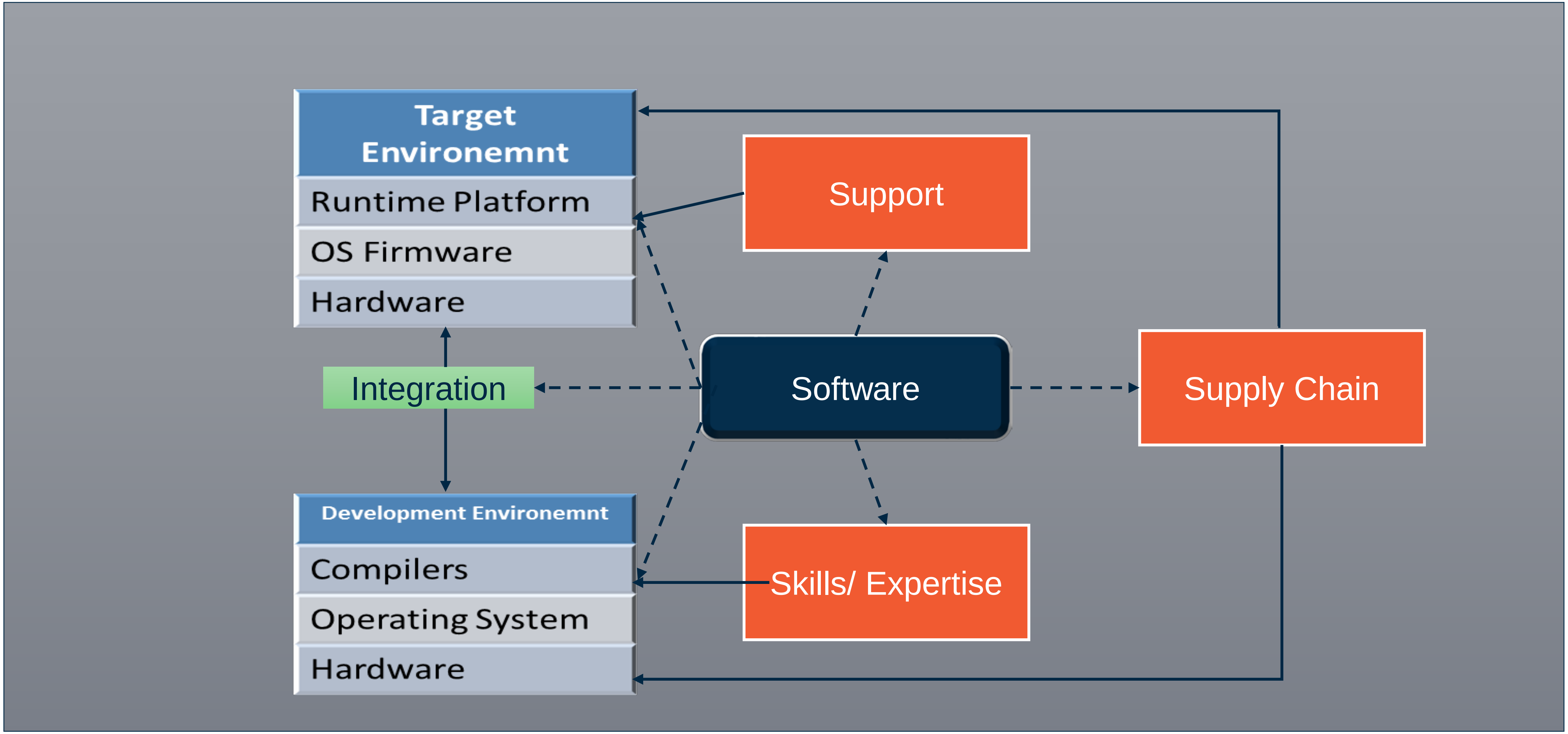
- Software Obsolescence Resolution Approach will help to identify the key cost drivers.
- Software Obsolescence Resolution Approach will help to identify the key risk/ uncertainties around the selection of appropriate resolution approach.
- Three major types of resolution approach are identified
  - Technical Resolutions
  - Logistical Resolutions
  - Functional Resolutions

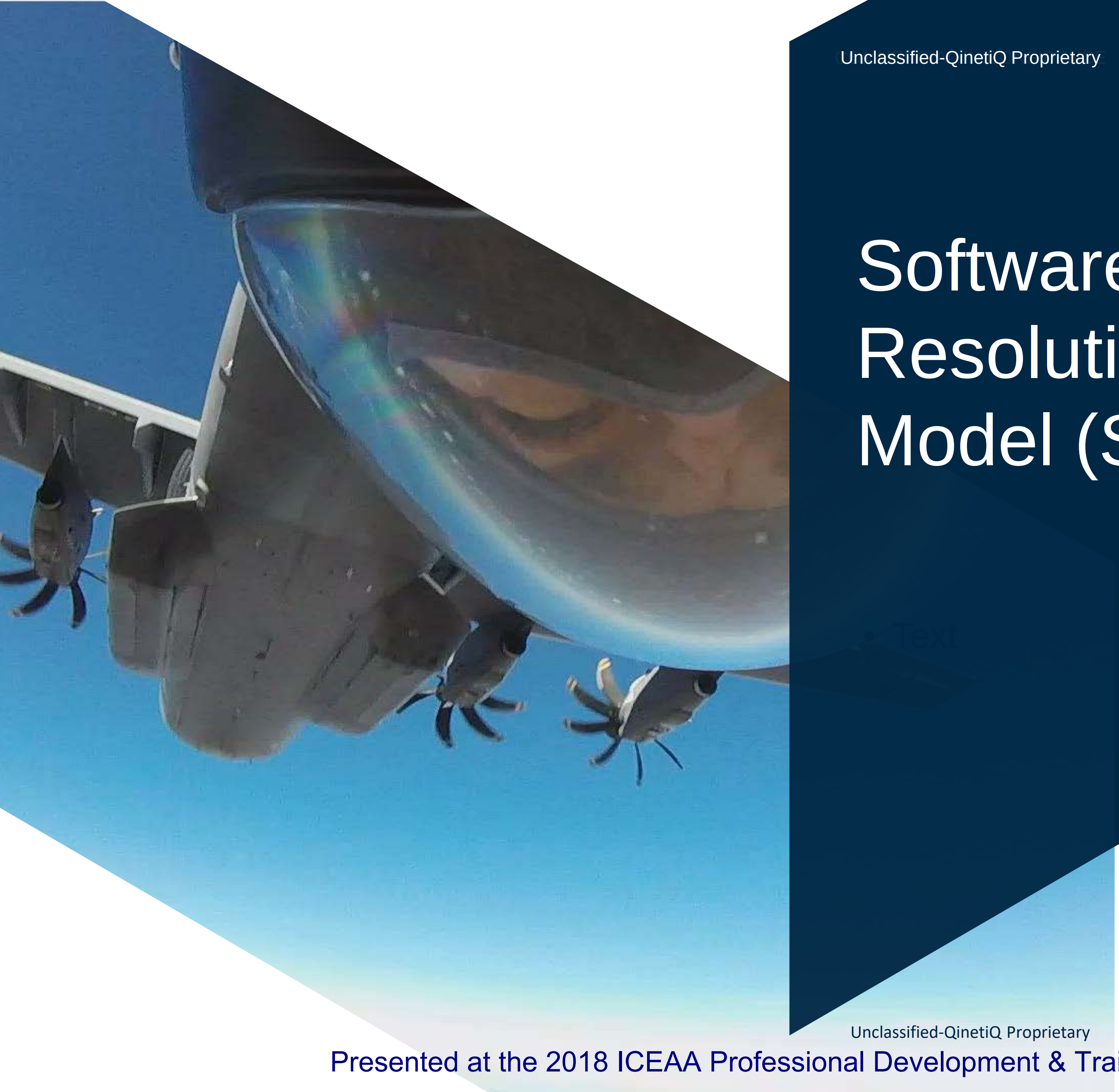
# Software Obsolescence Resolution Approach (Adapted from Bartel et al)



1. Identification of the Software Obsolescence Resolution Approach
2. Compile Software Obsolescence Strategy
3. Generates Key Cost Drivers and potential Risk /Uncertainties

# Software Obsolescence Management





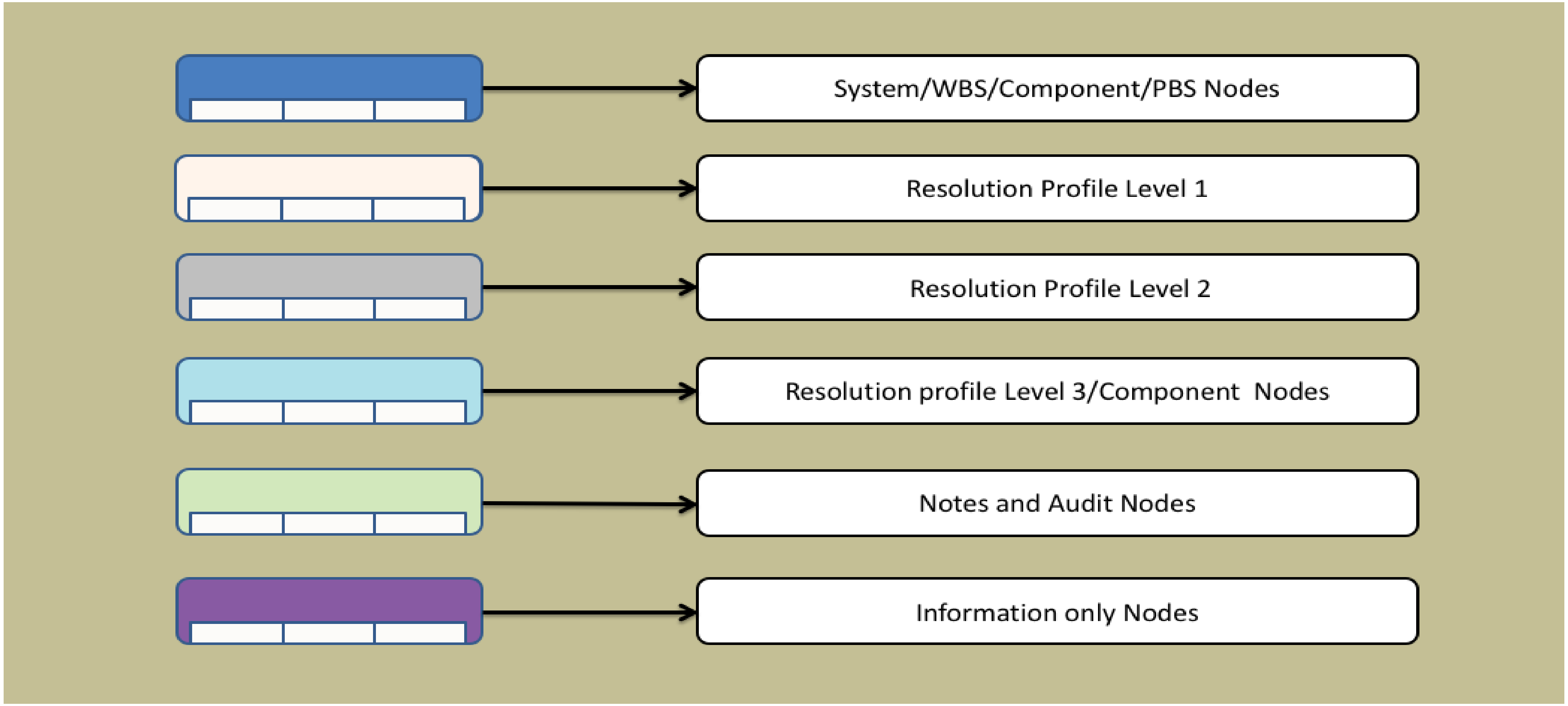
# Software Obsolescence Resolution Cost Optimisation Model (SRCOM)

- Text

# Software Obsolescence Resolution Cost Optimisation Model (SRCOM)

- This model will recommend the best resolution approach based on the cost
- Optimisation is undertaken using MatLab Optimisation toolbox
- This model is at a very early stages of its development and
- This methodology involve diagrammatical representation of the cost
- Diagram represents decompositions of cost from output node through successive levels to individual inputs nodes
- This decomposition is the focus of the diagram and technique
- The diagram will be converted into a set of inputs and calculations
- Each nodes have inputs (Min, ML and Max)

# Software Obsolescence Resolution Cost Optimisation Model - Nodes



System/WBS/Component/PBS Nodes

Resolution Profile Level 1

Resolution Profile Level 2

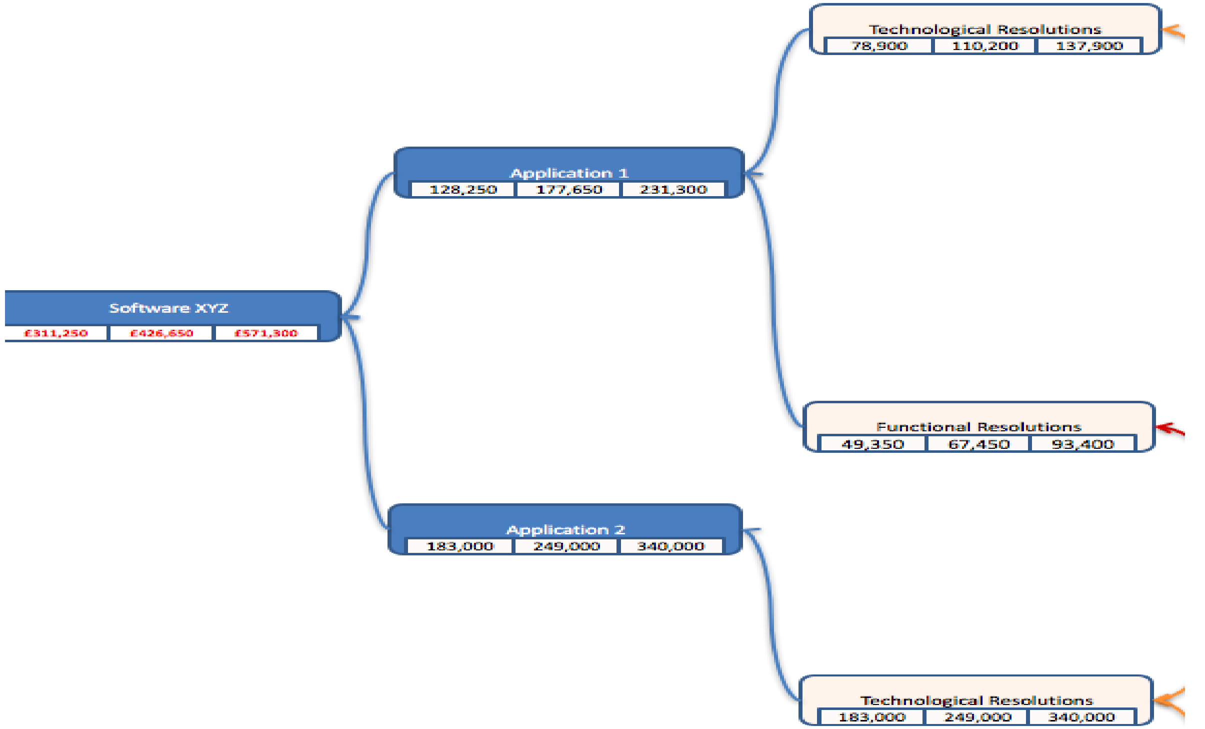
Resolution profile Level 3/Component  Nodes

Notes and Audit Nodes

Information only Nodes

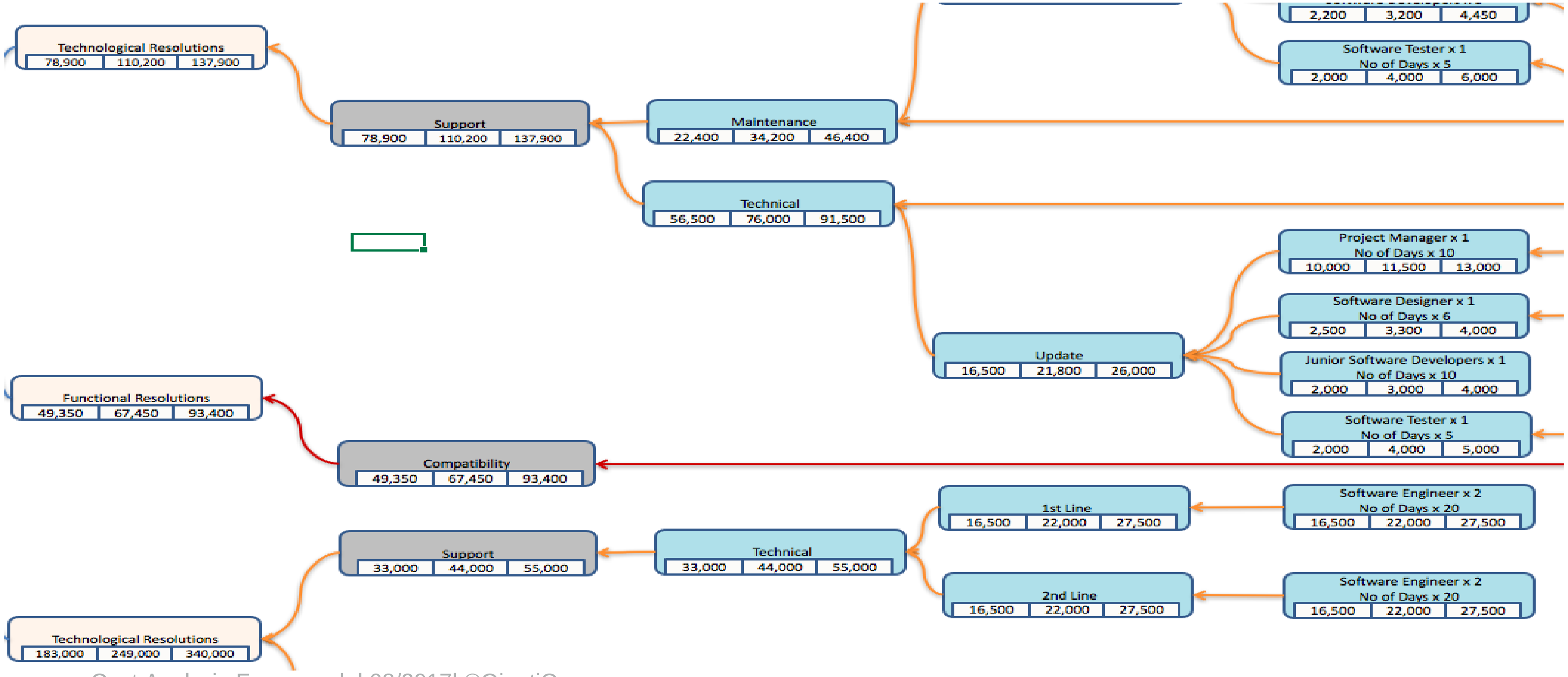
# Software Obsolescence Resolution Cost Optimisation Model - Nodes

- Software has several components and each component undergoes different obsolescence resolution profiles which can be diagrammatically represented using above methodology.
- Diagram represents decomposition of cost from output node (on the left) through successive levels to individual input nodes (on the right);
- This decomposition is the focus of the diagram and technique.

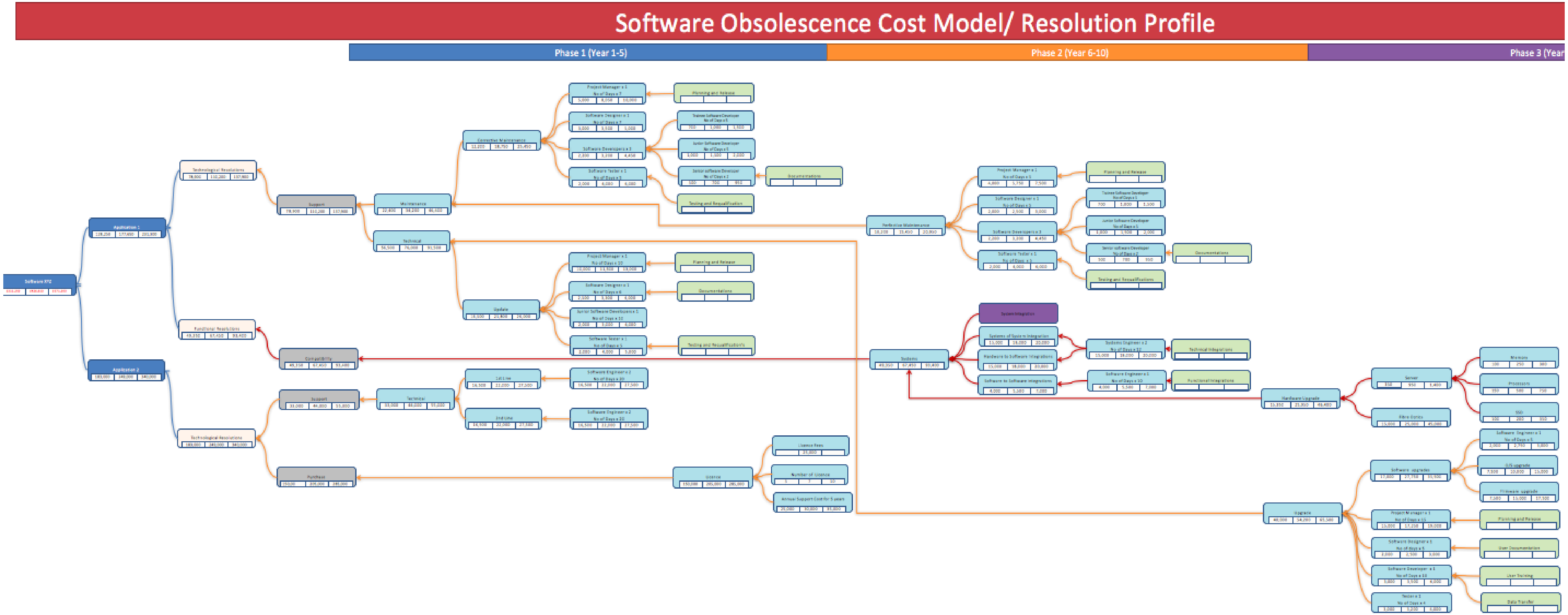


# Software Obsolescence Resolution Cost Optimisation Model - Nodes

- Input values are entered on the right of the diagram and aggregate through the model to give the vignette cost.
- Each node has a specific formula for combining inputs, which are defined in the diagram.
- Module nodes allow repeated use of common elements, such as Software Engineers rates.



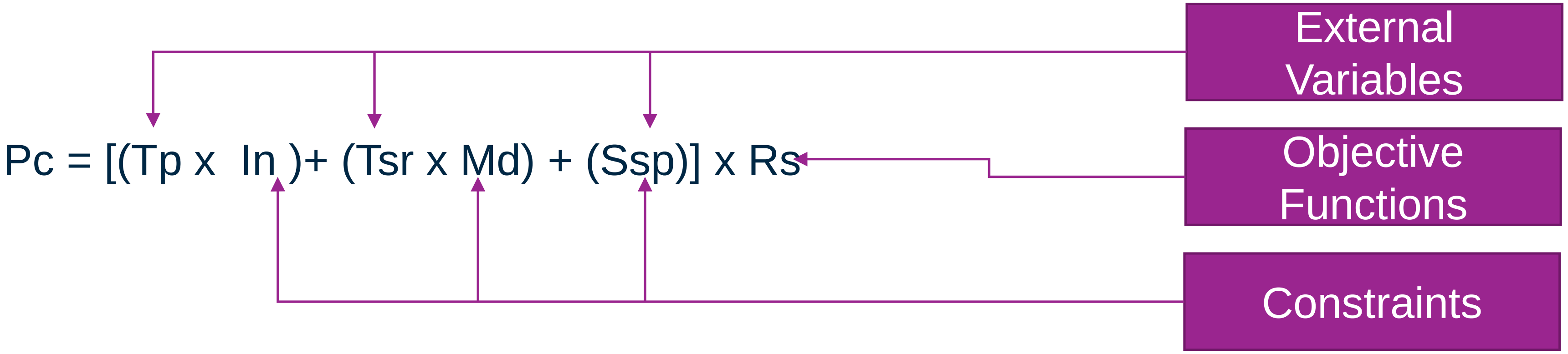
# Software Obsolescence Resolution Cost Optimisation Model – Example



# Optimisation Options

- In order to undertake Optimisation, it is necessary to identify the following
  - External Variables
  - Objective Functions
  - Constraints
- It is also necessary to develop a formula that uses the above variables, constraints and Objective functions in order to generate optimisation of
  - Cost
  - Resolution Strategy

# Optimisation Formula



Pc = [(Tp x  In )+ (Tsr x Md) + (Ssp)] x Rs

External Variables

Objective Functions

Constraints

Tp = Type of Platform

In = Level of Integration

Tsr = Testing and Requalification

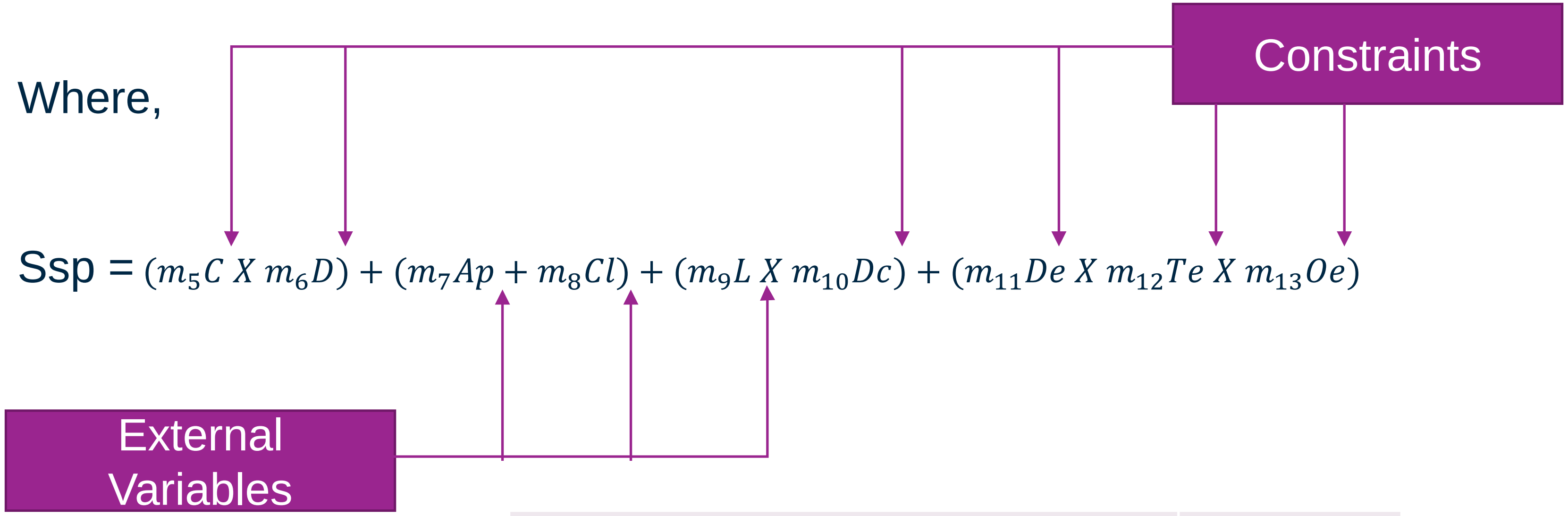
Md = Level of Modification

Ssp = Software System Parameter

Rs = Resolution Strategy Cost

# Optimisation Formula

Where,



$$Ssp = (m_5C\ X\ m_6D) + (m_7Ap + m_8Cl) + (m_9L\ X\ m_{10}Dc) + (m_{11}De\ X\ m_{12}Te\ X\ m_{13}Oe)$$

C = Software Complexity

D = Software Dependency

Ap = Number of Applications

Cl = Software Contract Length

L = Software Language

Dc = Software Development Cycle

De =Software Development environment

Te = Target Environment

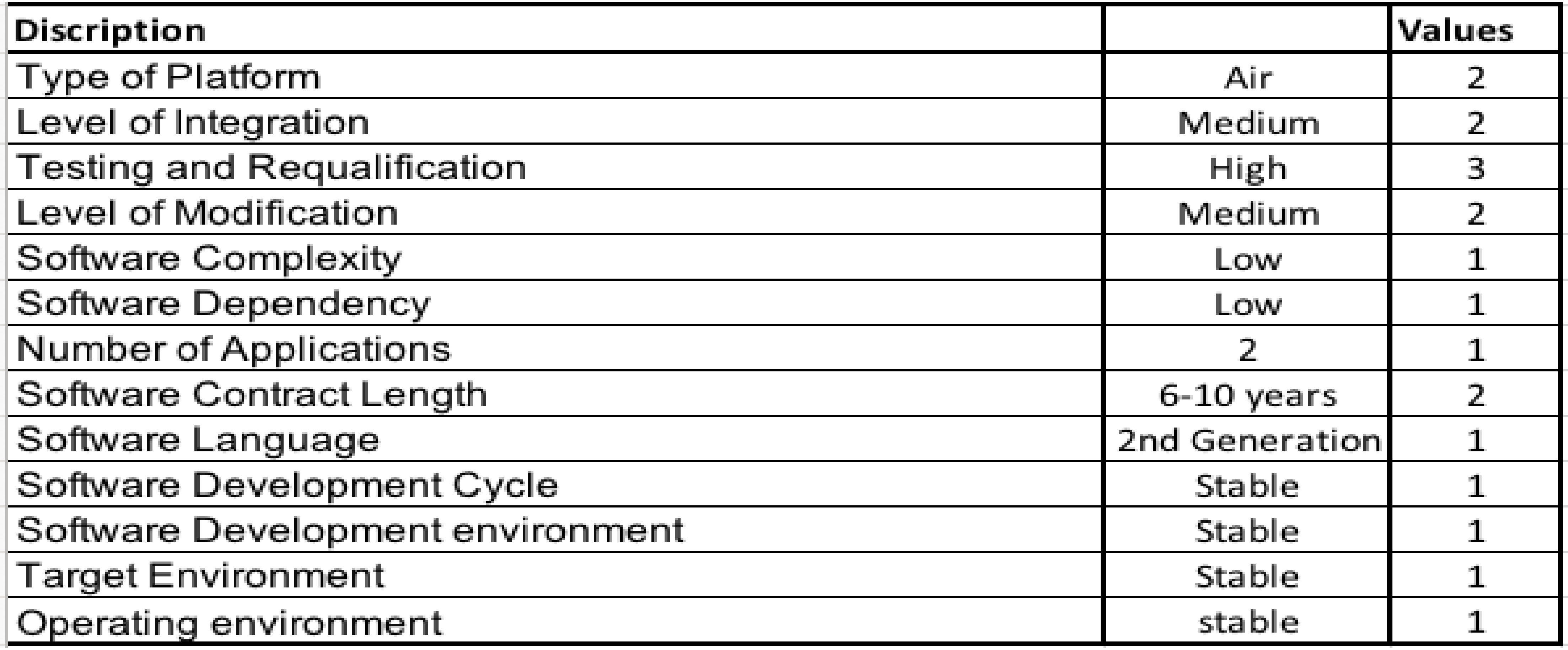
Oe = Operating environment

# Optimisation Formula

$$P = [(m_1 Tp\ X\ m_2 In) + (m_3 Tr\ X\ m_4 M) + \{(m_5 C\ X\ m_6 D) + (m_7 Ap + m_8 Cl) + (m_9 L\ X\ m_{10} Dc) + (m_{11} De\ X\ m_{12} Te\ X\ m_{13} Oe)\}]\ X\ R$$

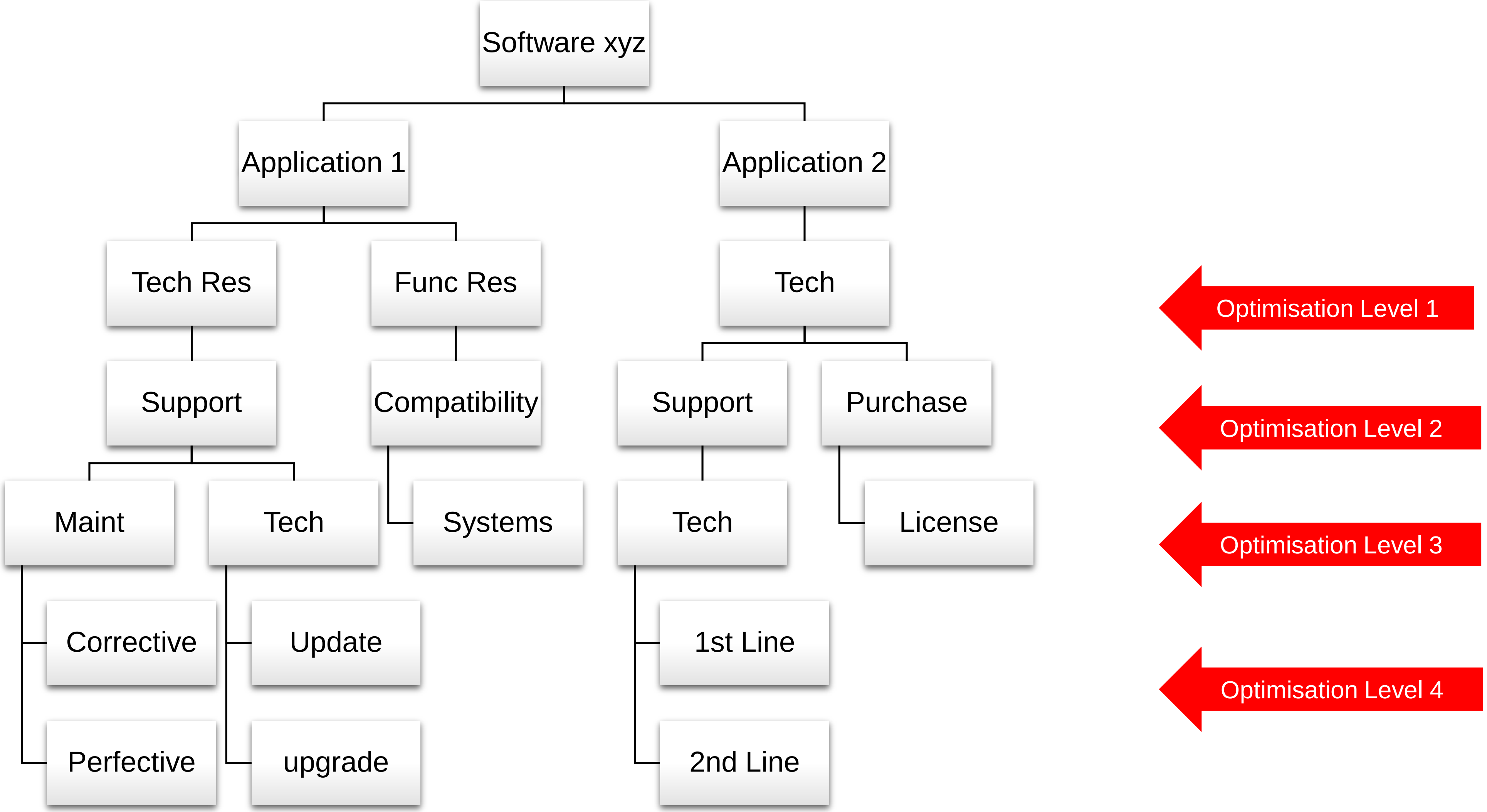
| Multiplier Identification number | Descriptions | Values |
|---|---|---|
| 1 | Communication and System | 2 |
| | Air | 2 |
| | Land | 1 |
| | Maritime | 1 |
| | Commercial System | 1 |
| 2,3,4,5,6 | High | 3 |
| | Medium | 2 |
| | Low | 1 |
| 7 | 10-50 Apps | 1 |
| | 51-100 Apps | 2 |
| | 101-500 | 3 |
| | 501 or More | 4 |
| 8 | 1-5 years | 1 |
| | 6-10 years | 2 |
| | 11 years or more | 3 |
| 9 | 1st Gen/Machine Code | 3 |
| | 2nd and 3rd Gen | 1 |
| | 4th Gen | 2 |
| | 5th Gen | 3 |
| 10 | Waterfall | 3 |
| | Spiral/Iterative | 2 |
| | Incremental | 1 |
| | Agile | 3 |
| 11,12,13 | Stable | 1 |
| | Unstable | 2 |

# Optimisation –Test Case

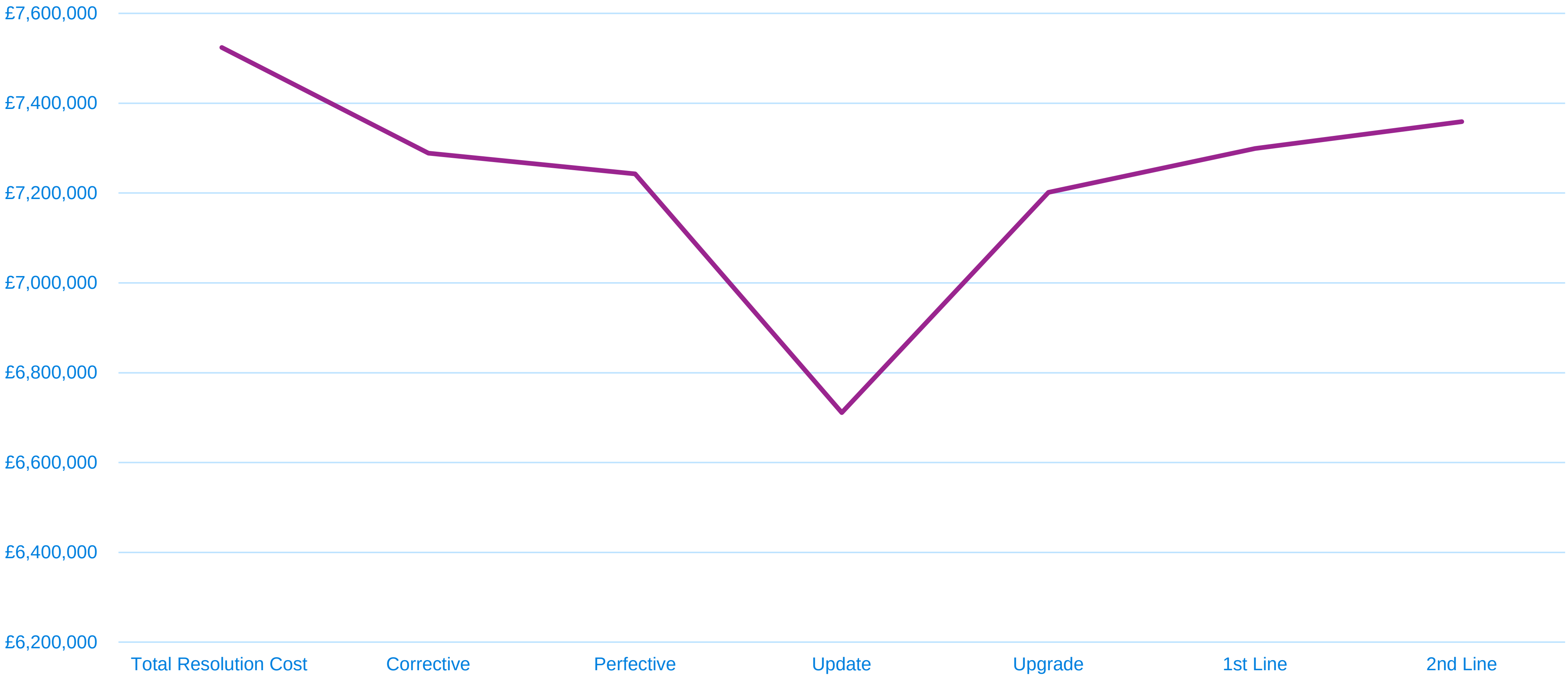


| Discription | | Values |
|---|---|---|
| Type of Platform | Air | 2 |
| Level of Integration | Medium | 2 |
| Testing and Requalification | High | 3 |
| Level of Modification | Medium | 2 |
| Software Complexity | Low | 1 |
| Software Dependency | Low | 1 |
| Number of Applications | 2 | 1 |
| Software Contract Length | 6-10 years | 2 |
| Software Language | 2nd Generation | 1 |
| Software Development Cycle | Stable | 1 |
| Software Development environment | Stable | 1 |
| Target Environment | Stable | 1 |
| Operating environment | stable | 1 |

# Optimisation –Test Case (Resolution Strategies)



# Optimisation –Results



# Summary

Software Obsolescence is a an emerging issues and it is important to understand how much SW/Obs is going to cost at a very early stages of development life cycle. In order to do so we need to

- Define what Software obsolescence is
- Understand the difference between Software Maintenance and Obsolescence
- Identify how Software Obsolescence is triggered
- Have a framework to manage software obsolescence proactively
- Identify the key Software Obsolescence Resolution approaches

# Conclusions

- Software plays an important role in defence. Almost every project in defence has software elements with various degrees of complexity and dependencies.
- In order to understand and see the bigger picture and challenges; software developers and the customers need to foresee the following issues that drive the whole life cost and should be in a position to develop innovative means to mitigate these issues by:
  - Anticipation of the Software Obsolescence at a very early stage of projects.
  - Understanding the technology insertion, technology update requirement.
  - Understanding the relationship between Software Maintenance and Software Obsolescence.
  - Anticipation of future capability integration to the existing platforms taking into account systems of systems, software to software and software to hardware integrations.
  - Formulation and evaluation of alternative architectural framework to inform the software designers that recognises the key market and cost drivers.

**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 ?