Delivering SE in practice

www.incose.org/symp2022

Duncan's background

Professor Duncan Kemp, CEng FIET, INCOSE Fellow

DE&S Fellow for Systems Engineering
Internal Technical Support Team Leader
Engineering Group
Abbey Wood South, BS34 8JH
Tel: +44 (0)7966 146 724

Defence Equipment & Support

1984 – Joined MoD as Student Engineer
1989 – Graduate Engineer

... Support TL
... Technical Discipline Lead for SE
2019 – Senior Fellow for SE
2022 – Digital Engineering Implementation TL

And …
- Chair of the INCOSE System Safety working group
- Published 20 peer reviewed technical papers, including several on System Safety
- Presented DE&S Maritime Safety Refresher, MOD 1* Boot camp
- Guest lectured at Birmingham, Loughborough and Bristol Universities, MIT, USMA West Point
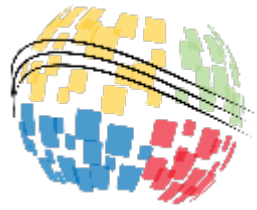- Visiting Professor for Systems Thinking at Loughborough University
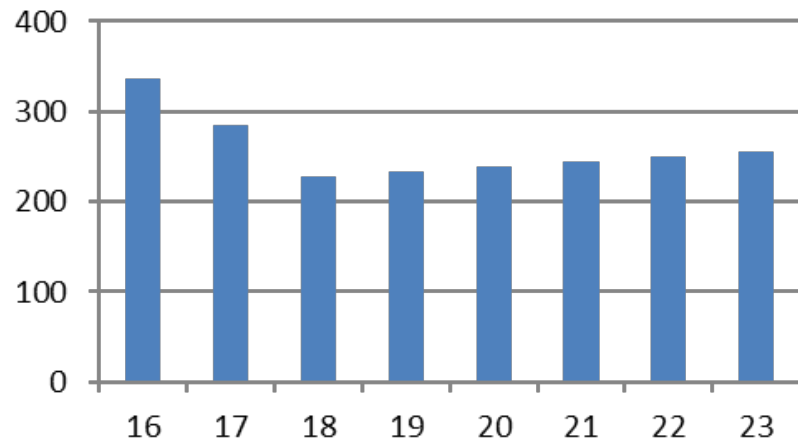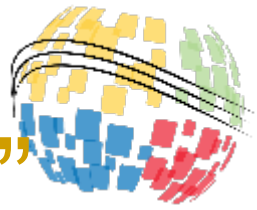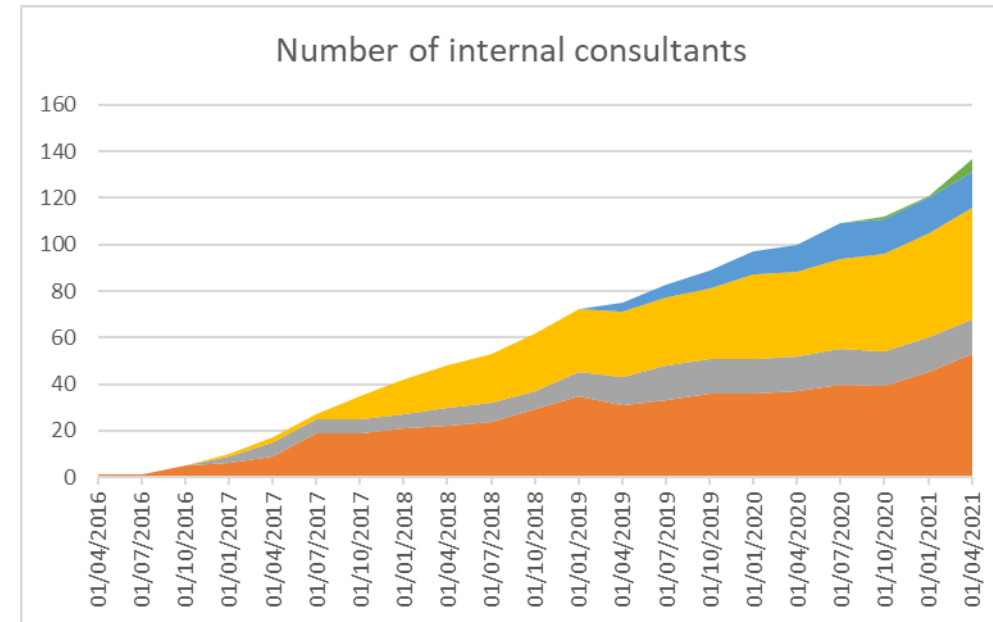- INCOSE Fellow

Presentation overview

- Background to the Internal Technical Support Team
- The right operating model for SE delivery
- Planning SE task delivery
- Challenges with core SE processes
- Selecting the right SE
- Delivering, learning and growing
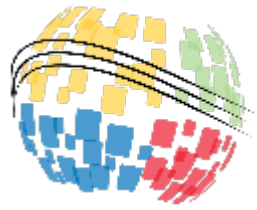
In-house staff	In house tasks
Contracted staff	Contracted tasks

"As much as possible, as fast as possible"

"Can you build an internal technical support capability"

Number of internal consultants
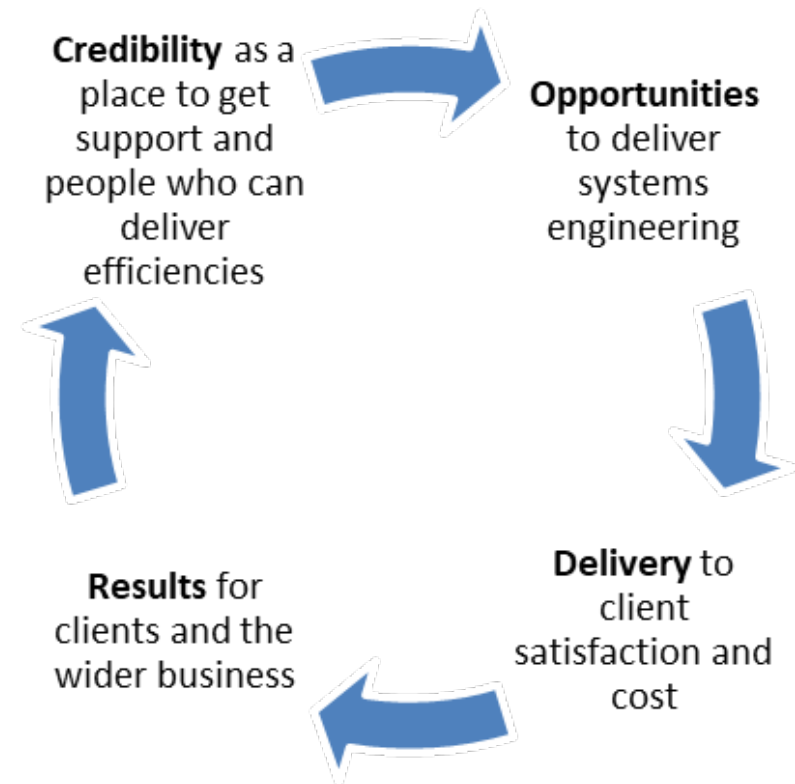
What do our stakeholder want?

A team to reduce the cost of expensive support contractors

A team to provide systems engineering support to projects

Systems Engineering Internal Technical Support Team

A team to grow the capacity and capability of DE&S engineering

A team to provide interesting and varied work to team members
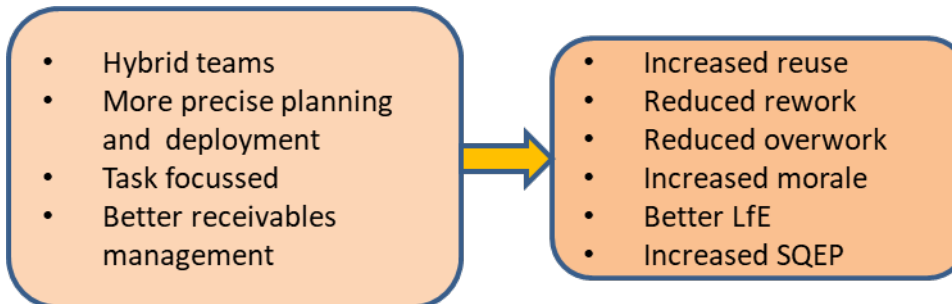
Credibility as a place to get support and people who can deliver efficiencies

Opportunities to deliver systems engineering

Delivery to client satisfaction and cost

Results for clients and the wider business
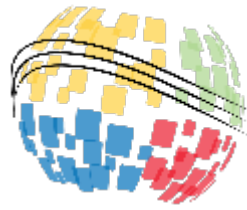
Fix scope, fixed price delivery

2 days L5
20 days L4
60 days L3
———
£44,160

Hybrid teams
- Hybrid teams
- More precise planning and deployment
- Task focussed
- Better receivables management

→

- Increased reuse
- Reduced rework
- Reduced overwork
- Increased morale
- Better LfE
- Increased SQEP

Team income and expenditure		
	Income	Expenditure
Value of tasks	£x,xxx,xxx	
Team cost		£x,xxx,xxx
Net income	£xx,xxx	
% recovery	101%	
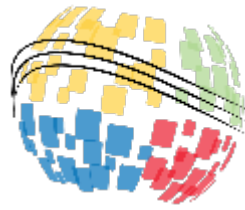
Simple operating model, clear golden thread

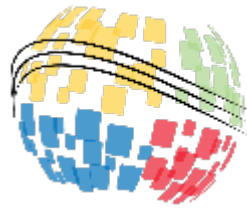| Prioritisation | • Client reviews portfolio priorities
• Client agrees next tranche of work priorities |
Agree Scope and Cost	• Task Lead Generates POAP (receivables, activities, deliverables, benefits) • Client accept POAP, confirm funding • Finance transfer
Task Leads Deliver	• Deliver Tasks, booking to internal code • Provide regular feedback to Client
Accept and Close	• Deliverables Accepted – Tasking form signed off • Client feedback requested / LFE captured

Savings delivered
Costs Recovered
Satisfied Customers
Happy Team

Individual utilisation target
Task delivery (CPI)
Customer satisfaction
Support wider team

Give it go and take risks

80% solution for 20% effort

Clients over process

Doing over documenting

Doing over talking about it

Accelerator not brake

Planning delivery

How are we going to deliver this?

Who is this for? Who will sign off the work?

What led to this need?

What will we achieve for you?

What do we need from you? What have we assumed will happen?

What will you be able to achieve once we have delivered this?

Who is going to do the work?

What are we going to do for you?

What are we going to physically deliver you?

How much will this cost you, and how much have we saved you?

Receivable → Activity → Deliverable → Benefit

Double loop delivery

Issue

Benefits realisation

Problem understanding

Solution selection

Use

Requirements

Acceptance

Design

Integration

Manufacture

Pyramid (top right):
- Purpose
- Operational concept
- Whole capability design
- Sub-system requirements and interfaces

Flowchart (center):

What is the risk the activity is managing

Is that a risk for you
- Yes → Decide the level of detail needed to manage the risk → Do the work!
- No → Record the decision and move on
- Not sure → Discuss with engineering manager

Pyramid (bottom right):
- Purpose
- Engineering Approach
- Whole systems engineering approach
- Individual SE tasks, products and interfaces

Problem understanding and solution selection
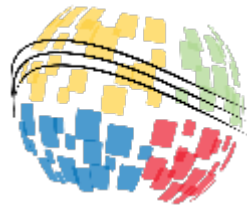
Phase at which error is detected and fixed	Cost to Fix
Requirements	x1 (reference)
Design	x3 to x8
Build	x7 to x16
Test	x21 to x78
Operations	x29 to x1615, mean x250

The Systems Praxis Framework, a joint project of the International Council on Systems Engineering and the International Society for the Systems Sciences

INTEGRATIVE SYSTEMS SCIENCE
Identifying, exploring, and understanding patterns of complexity through contributions from

Scientific Disciplines e.g., Physics, Neuroscience

Humanistic Disciplines e.g., Psychology, Culture, Rhetoric

Foundations
Meta-theories of Methodology, Ontology, Epistemology, Axiology, Praxiology (theory of effective action), Teleology, Semiotics and Semiosis, Categories, etc.
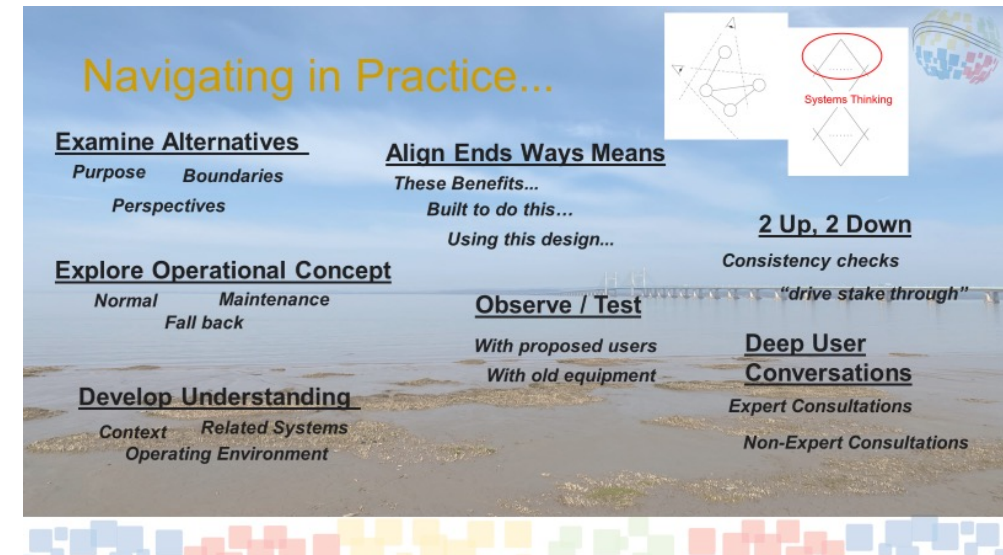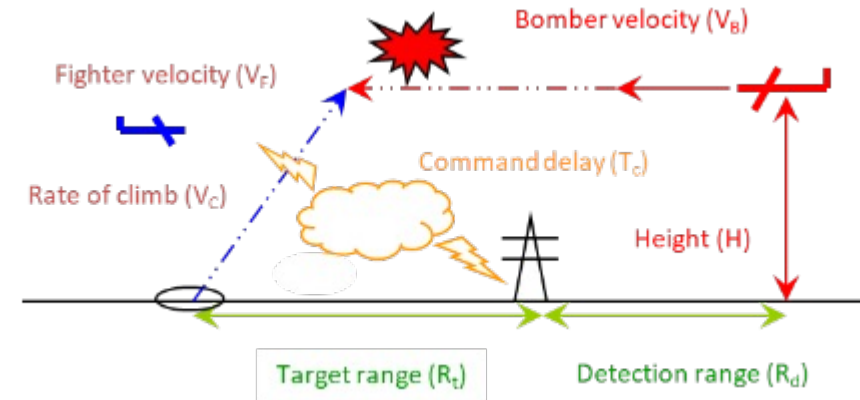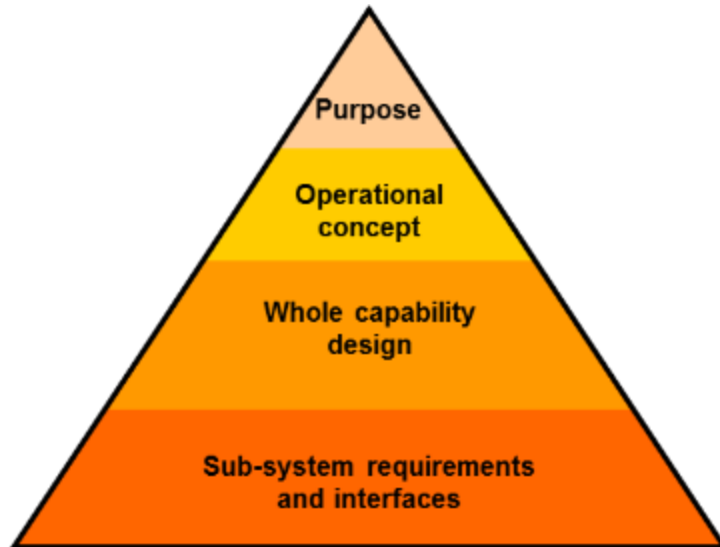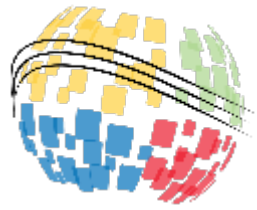
Theories
General Systems Theory, Systems Pathology, Complexity, Anticipatory Systems, Cybernetics, Autopoiesis, Living Systems, Science of Generic Design, Organization Theory, etc.

Representations
Models, Dynamics, Networks, Cellular Automata, Life Cycles, Queues, Graphs, Rich Pictures, Narratives, Games and Dramas, Agent-based Simulations, etc.

Pragmatic Disciplines e.g., Accounting, Design, Law

Formal Disciplines e.g., Math, Logic, Computation

practice informs theory

SYSTEMS THINKING
Appreciative and reflective practice using 'systems-paradigm' concepts, principles, patterns, etc.

theory informs practice

SYSTEMS APPROACHES TO PRACTICE
Addressing complex problems/opportunities using methods, tools, frameworks, practice patterns, etc.

direct input from disciplines

measured and specified data, metrics, etc.

Pragmatic, Pluralist, or Critical multi-methodology uses heuristics, prototyping, model unfolding, boundary critiques, etc., to understand assumptions, contexts, and constraints, including complexity from stakeholder values and valuations; chooses appropriate mix of 'hard', 'soft', and custom methods; sees systems as networks, societies of agents, organisms, ecosystems, rhizomes, discourses, machines, etc.

'Hard' methods are suited to solving well-defined problems with reliable data, clear optimization goals, and at most objective complexity; use machine metaphor and realist/functionalist foundations.

'Soft' methods are suited to structuring problems involving incomplete data, unclear goals, perspective and role complexity, etc.; use learning system metaphor and constructivist/interpretivist foundations.

input from experience and legacy practices

solicited local values, knowledge, etc.

http://www.systemspraxis.org

Outcomes   Actions

Creative Commons Attribution 3.0 License
© 2012 International Federation for Systems Research

Navigating in Practice...

Examine Alternatives
Purpose   Boundaries
Perspectives

Align Ends Ways Means
These Benefits...
Built to do this...
Using this design...

Explore Operational Concept
Normal   Maintenance
Fall back

2 Up, 2 Down
Consistency checks
"drive stake through"

Observe / Test
With proposed users
With old equipment

Deep User Conversations
Expert Consultations
Non-Expert Consultations

Develop Understanding
Context   Related Systems
Operating Environment

Systems Thinking

See "3.1.2 Death Rays, databases, and double diamonds"

Requirements

TUSBAT

TEPID OIL

Function	Performance	Threshold	Objective	Acceptance

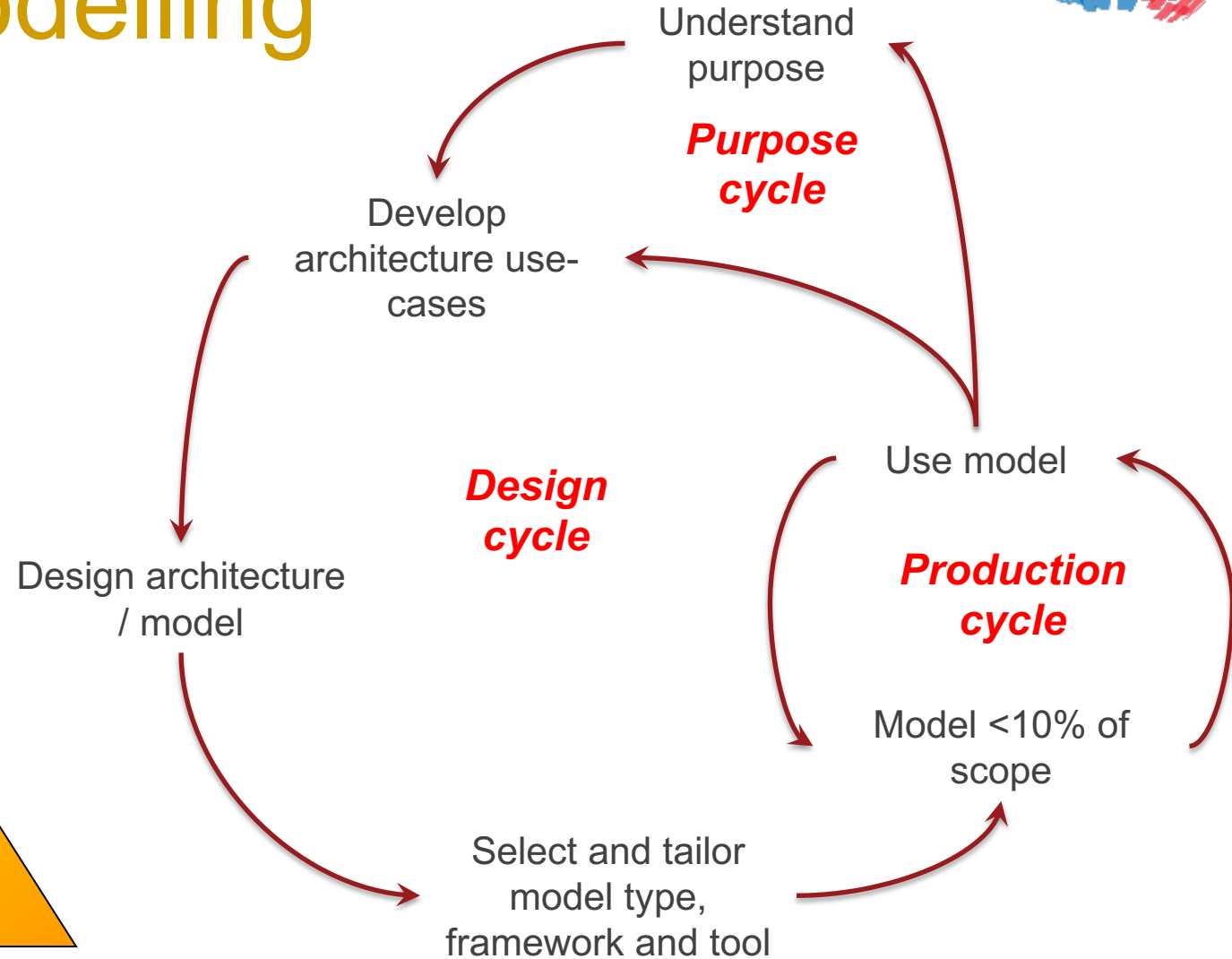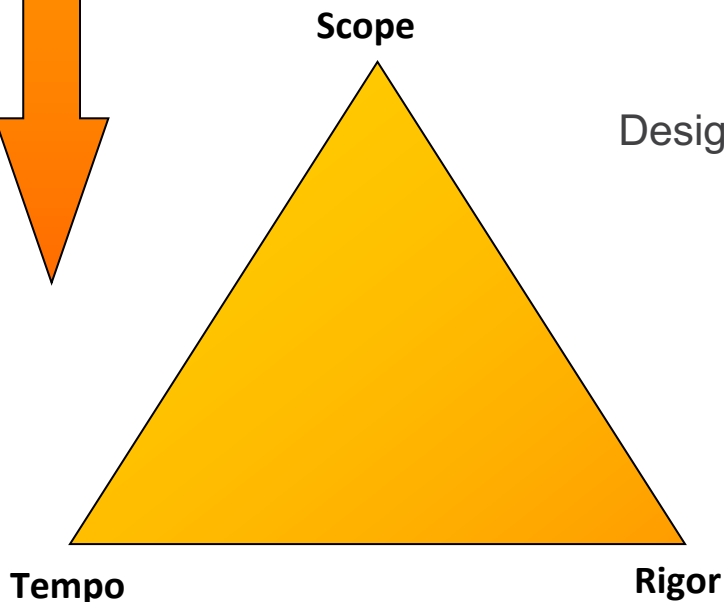# Architecting and modelling
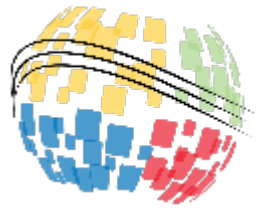
Expert judgement

Pencil + paper calcs

Static models

Dynamic simulations

Person/kit in the loop

Increasing time, cost, precision, rigor

- Fidelity?
- Validation?
- Confidence?
- Assumptions?

Scope

Tempo

Rigor

Understand purpose

Purpose cycle

Develop architecture use-cases

Design cycle

Use model

Design architecture / model

Production cycle

Model <10% of scope

Select and tailor model type, framework and tool

Design and Manufacture

Force

↕

Capability

Platform

↕

Radio

Antenna
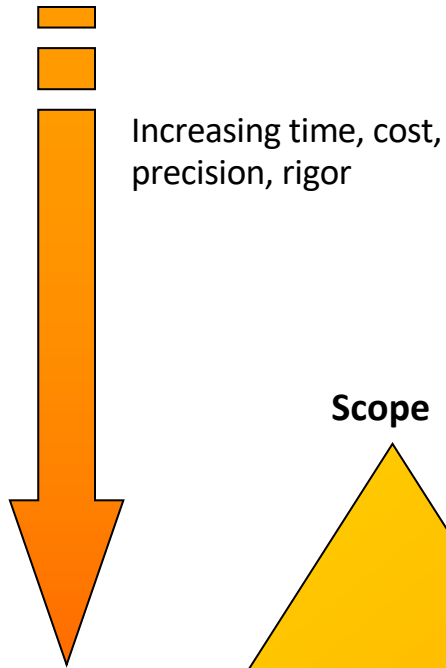
	Loose	Tight
Closed	No rights to change	Off the shelf
Open	Plug and play	Outsourced design

Commercially

Technical coupling
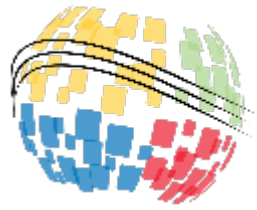
Integration and acceptance
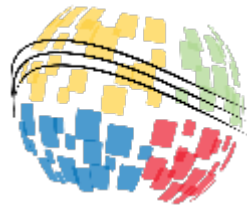

Requirements → Design
Acceptance
Integration → Design
Manufacture
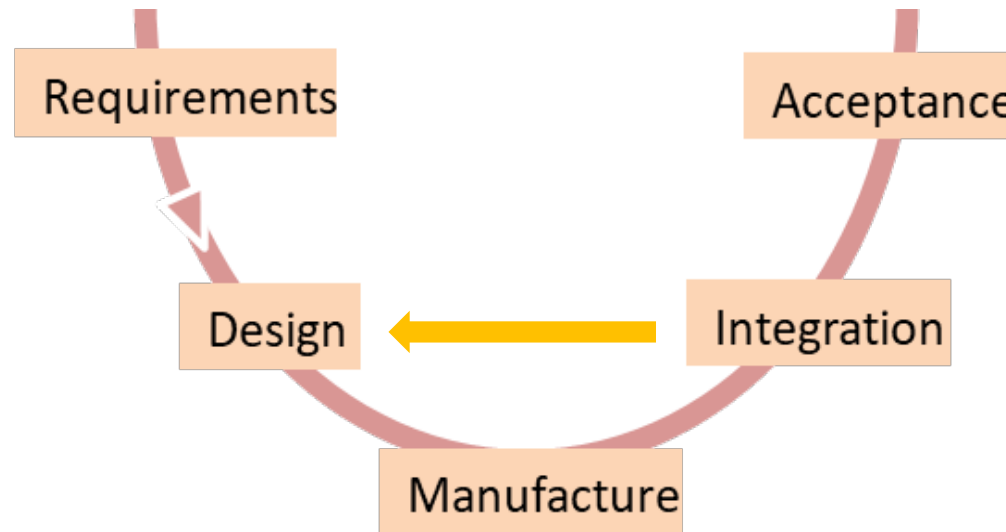
- FATS
- HATS
- SATS
- SIFATS

Operations and maintenance
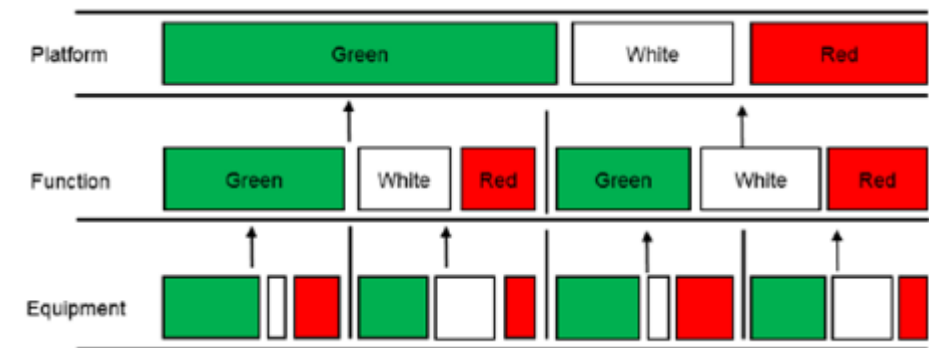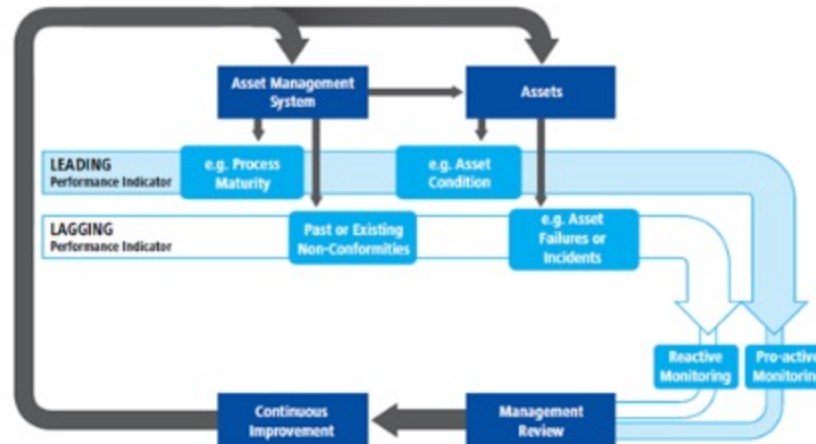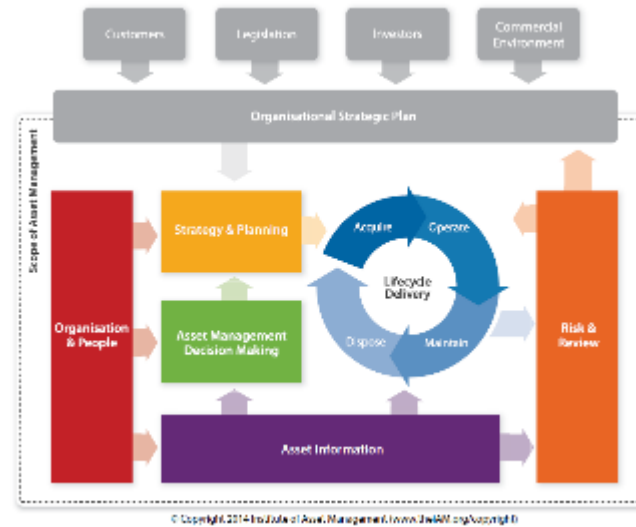
Selecting the right SE
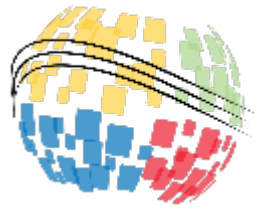
Technology

Domain

π

Lifecycle stage

Attributes

Ben's background

2008 – MoD Electrical and Electronic Apprentice
2011 – Submarine Surveyor
2013 – Engineering Skills Development Manager for
      Apprentices
2015 – SSN Stealth Manager
2019 – ITS SE Consultant
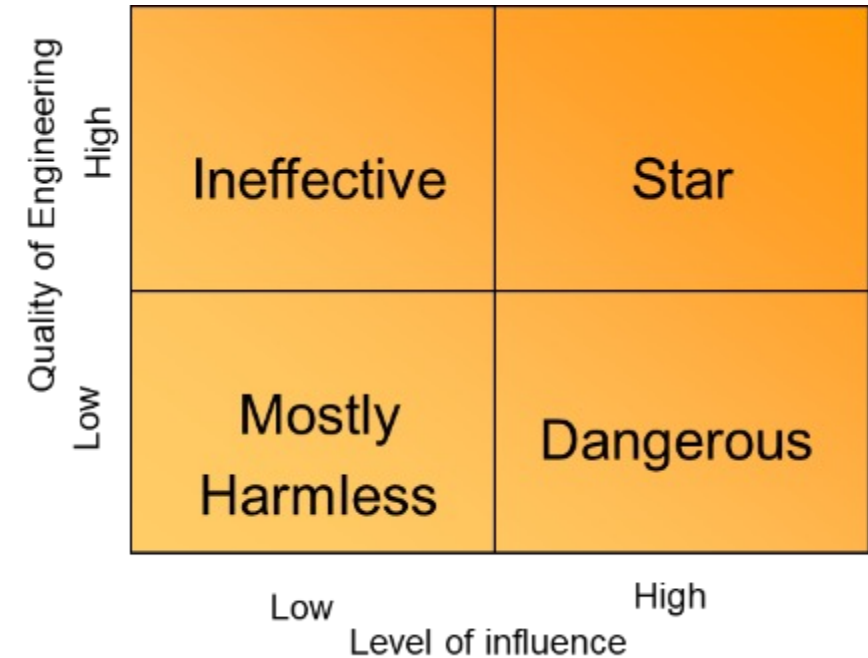
And …
- MSc Engineering Management
- ISO TC251 Asset Management committee
- BSI TC251 UK deputy chair
- Undertaking PhD
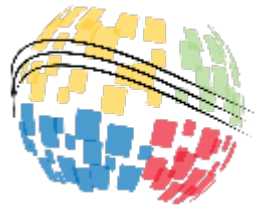
Ben Mogridge MSc CEng MIET MINCOSE
Internal Technical Support Systems Engineering Consultant
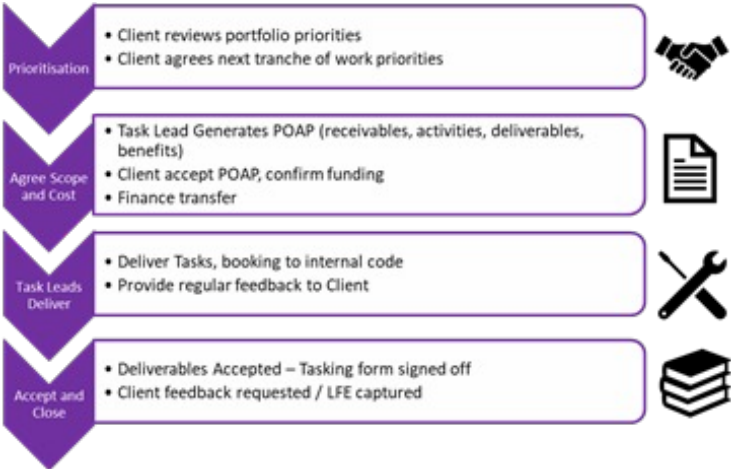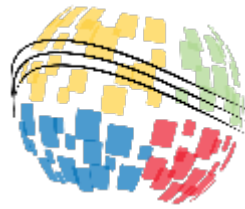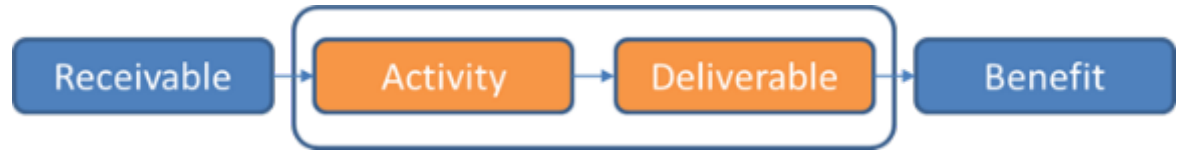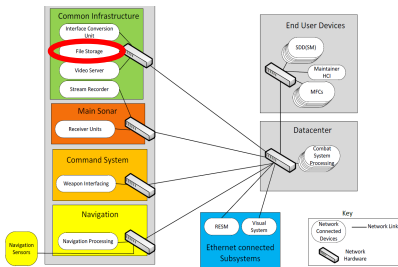Engineering Group
Abbey Wood South, BS34 8JH
Tel: +44 (0)7966 146 724

Defence Equipment & Support

An example task – SM Architecture Use Cases
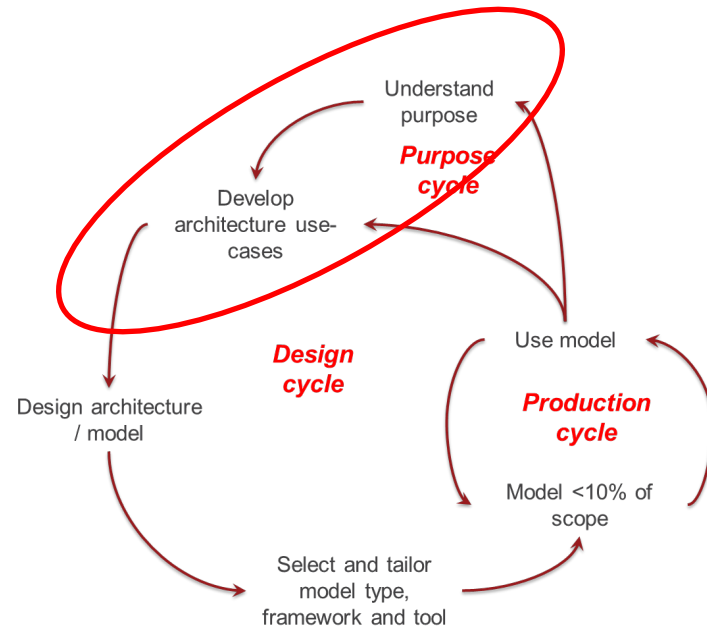
Technical – can I make it work?
Assurance – safety, performance and security.
Legal – can I share the information with third parties?
Commercial – can alternative suppliers use it?

Right Tools	All the gear, no idea	Efficient and Effective
Wrong Tools	Useless	Inefficient but Effective
	Wrong People	**Right People**

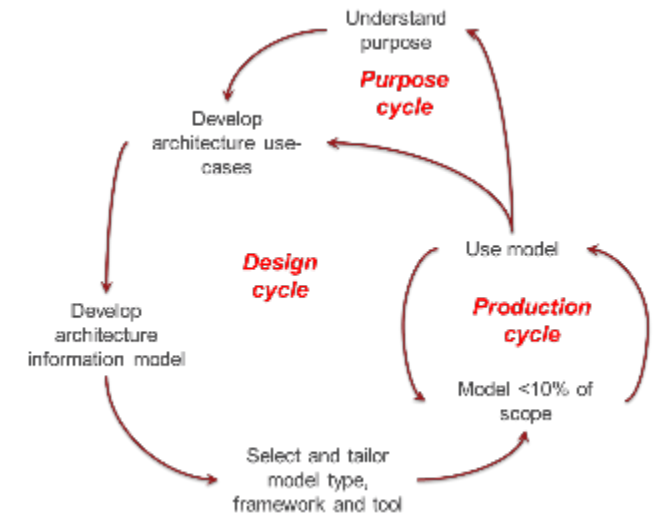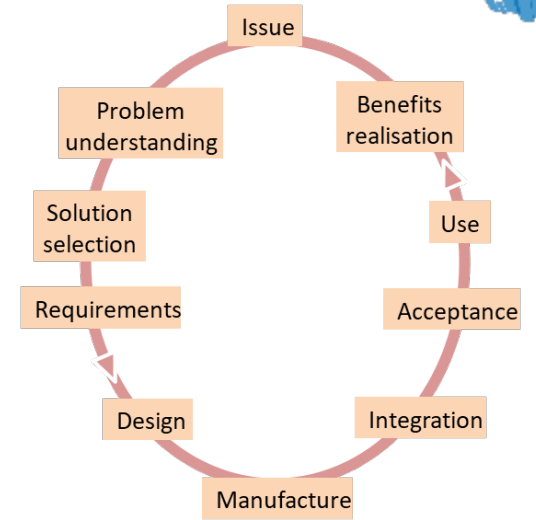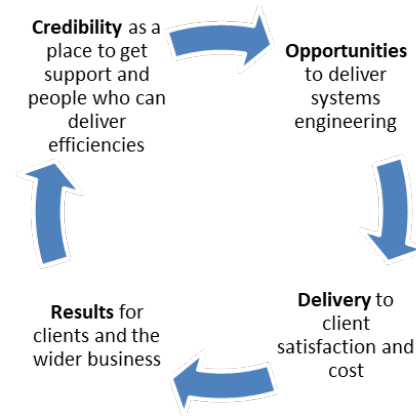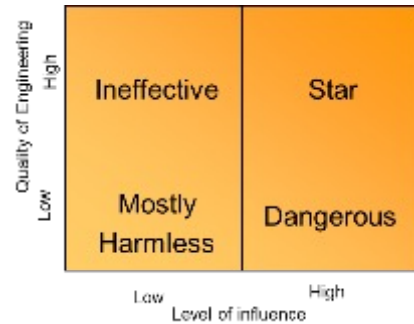Receivable → Activity → Deliverable → Benefit

18

Summary

A team to reduce the cost of expensive support contractors

A team to provide systems engineering support to projects

Systems Engineering Internal Technical Support Team

A team to grow the capacity and capability of DE&S engineering

A team to provide interesting and varied work to team members

Credibility as a place to get support and people who can deliver efficiencies

Opportunities to deliver systems engineering

Results for clients and the wider business

Delivery to client satisfaction and cost

Issue

Problem understanding

Benefits realisation

Solution selection

Use

Requirements

Acceptance

Design

Integration

Manufacture

Force

Capability

Platform

Radio

Antenna

	Ineffective	Star
High Quality of Engineering Low	Mostly Harmless	Dangerous

Low   Level of influence   High

Understand purpose

Purpose cycle

Develop architecture use-cases

Use model

Design cycle

Production cycle

Develop architecture information model

Model <10% of scope

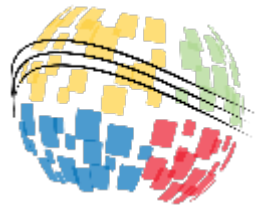Select and tailor model type, framework and tool

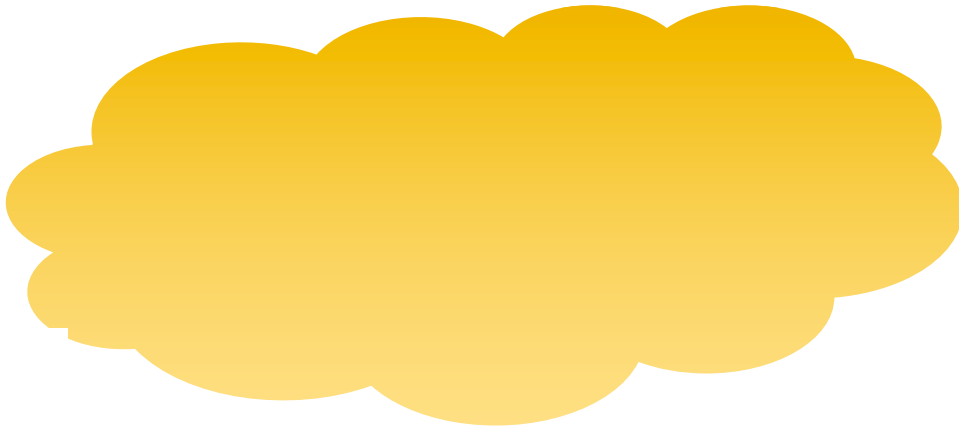32nd Annual **INCOSE**
international symposium

hybrid event

Detroit, MI, USA
June 25 - 30, 2022

www.incose.org/symp2022

Savings delivered
Costs Recovered
Satisfied Customers
Happy Team

Individual utilisation target
Task delivery (CPI)
Customer satisfaction
Support wider team