Systems Engineering – The Hard Way

A R Edmondson, BEng (Hon), MSc, CEng, FIET[a]
B Twomey, BEng (Hon), CEng. FIET, FIMarEST, MIMechE[b]

[a]*BAE SYSTEMS Maritime - Submarines, UK*
[b]*Rolls Royce, UK*
*Corresponding authors' e-mail: andrew.edmondson@baesystems.com: bernard.twomey@rolls-royce.com

SYNOPSIS

Ship designers, builders, owners, insurers and class societies are becoming ever more aware of the complex interactions of the various systems found on all types of marine vessels. Therefore a design process that acknowledges these demands and assesses the risks posed, and manages them becomes ever more important. This paper seeks to explore some of the, sometimes apparently, conflicting requirements that are placed on designs of new marine platforms and looks at methods that enable these elements to be expressed, understood and managed in the context of an integrated ship design.

The demands placed on new vessels include a range of requirements that move away from being solely based around the traditional functional requirements; including the ideas of designing for ease of shipbuilders, operators and maintainers; and now acknowledging the need of a through life safety case, cyber security case, and full obsolescence planning. This becomes ever more complex when consideration is given to how these through life elements are practically managed, with a range of methods, none of which are without their own challenges.

It is important to note as these demands are discussed that often a 'solution' in the truest sense does not exist and the management of risk becomes a balance between the expected risk, the practicable solution, along with the potential compromises to both programmes and cost.

While these demands place huge constraints and drive complexity into design processes, the issues can, and regularly have, been further exacerbated when some of these, or other requirements, are introduced into the design or build phases of projects. Introduction of design drivers should not be undertaken lightly or without expected, and accepted, increases in required resources, both financial and calendrical.

Keywords: Systems Engineering; Requirements; Functional; Transverse; Implicit; Explicit; Through-life;

1 Introduction

When purchasing, specifying, or designing any new marine platform the aspect that is always sought is a clear and unambiguous set of 'requirements.' While in the commercial world the functional, or capability, requirement is clear, as the purpose is singular and the business case has to be evident, or the programme doesn't progress beyond the initial consideration; in the military sphere, even the full functional intent can have limited definition. Predicting the function to be fulfilled, or the capability to be deployed, by a naval platform in 10 years, for entry into service, or up to 50 years for the late life of the end of class platforms, remains understandably challenging.

An example of this difficulty becomes clear when vessels such as HMS Hermes [Figure 1] are considered. HMS Hermes was originally laid down, to be HMS Elephant, in 1944 to a design developed in

---

Authors' Biographies

Andrew Edmondson is a Senior Technologist with BAE Systems Maritime - Submarines in Barrow-in-Furness. He obtained his BEng in Electrical and Electronic Engineering at UMIST and his MSc in Marine Technology at the University of Newcastle-upon-Tyne. He has been involved throughout the design process for electrical power systems of a variety of submarines and surface ships and has led teams throughout concept and detailed design phases. Andrew is a Chartered Engineer and a Fellow of the IET.

Bernard Twomey is currently working for Rolls-Royce (Marine) with responsibility for Regulatory Development within the maritime sector and is undertaking a PhD at York University. He obtained his BEng in Electro-Mechanical Power Systems from Loughborough University after spending 13 years in the Merchant Navy. He then spent 23 years working for Lloyd's Register as the Global Head of Electrotechnical Systems. Bernard is a Chartered Engineer and Fellow of the IET, IMarEST and MIMechE.

the closing years of the Second World War. The ship was finally commissioned in 1953 and remained in service with the Royal navy until 1984, and then remained in service with the Indian Navy until 2017, some 83 years after being laid down. In that time the world, and the requirements of both navies that used her in service had changed tremendously. The requirements she was designed for in the 1940's were very different from the conflict she found herself in around the Falkland Islands. The aircraft she was expected to carry at the end of her life were immeasurably different from those available in 1944. It is this rapidly changing outlook that makes setting requirements such a challenging process.

Fig 1: HMS Hermes

Despite not always understanding the functional requirements or the capability requirement the industry has to acknowledge the broad aims of such a platform, and develop designs to meet the needs of the end user (I.E. Royal Navy, Royal Australian Navy, etc.), which again are often different to those of the customer (I.E. UK MoD DE&S, US DoD). When the detailed requirements sought by other parties such as insurers or class societies are thrown into the mix, the picture becomes ever more complex, which can result in very different aims for the requirements, and a very different context for their expression.

Being able to marry the abundance of requirements that are faced by any prospective design project is a complex and daunting issue. For those that consider the entire platform life the issues become even greater, the needs, or desires, of the maintainer can be in direct opposition to those specified elsewhere, the ability to upgrade the platform either through evolved need through life, or due to obsolescence adds further complication.

All this cacophony of disparate and often contradictory requirements is further complicated by the need to retain a whole platform understanding to retain a design authority throughout the total life of the platform. This need for design authority is expressed in many ways, perhaps the two most visible and currently topical are the development and maintenance of a safety case and cyber-security (or information assurance) case. This paper seeks to explore the range of requirements that can be presented to the platform project team with discussion regarding how these can be considered until the end of life of the last platform of a class, with the differences between individual vessels of the class understood, documented and traced until, potentially, the end of the disposal cycle.

2 Requirements – Explicit Functional

2.1 Initial

As was discussed in the previous section, those requirements that define the purpose of a platform are always those first sought out, for multiple reasons. For the customers, it is an expression of the capability that they desire, it is the basis for the justification of the expense, and it is the documentation of those things that the project will be judged upon. For the designer it is the expression of those things that start to drive the design. Most commonly these requirements are high level:

Inherently safe design[2]
Payload
Speed
Endurance
Operating Area
Cost
Required by Date

It may also include estimates for physical elements of the solution such as:

Displacement,
Length,
Keel Depth,
Beam,
Complement,

especially, where these are to be constrained for reasons of infrastructure that 'currently exists and' where there is a desire to continue its use.

As the design develops these aspects drive many of the key decisions regarding any platform, even with the cursory list above, this can be seen with the acknowledgement of any one of the requirements there is an impact on the ability to deliver the others, at face value this becomes a circular argument with no sense that it can be easily solved, the requirements fight against each other in a circular fashion.

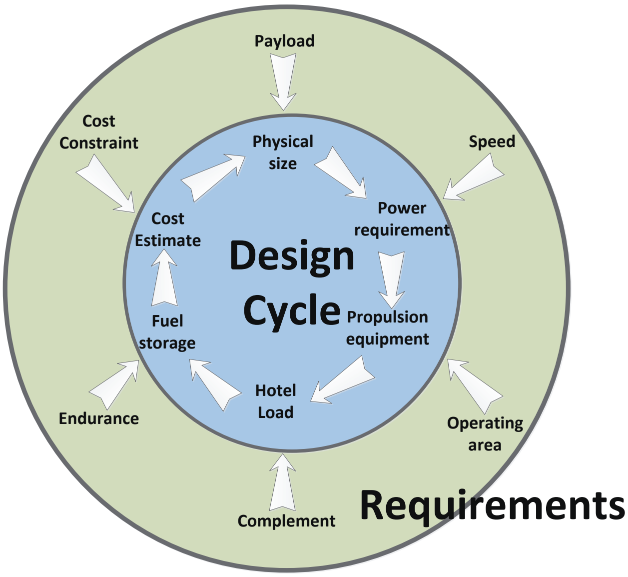

Fig 2: Circle of influence due to design requirements?

[2] An inherently safe design is one which prevents a specific harm occurring by using an approach, design or arrangement that ensures that the harm cannot happen.

 http://doi.org/10.24868/issn.2515-818X.2018.009

As can been seen in Figure 2 if we acknowledge the requirements and progress through a design cycle of:

- a prospective physical size governed by the required payload;
- assess the propulsion power requirements given by the required speed;
- allowing the choice of propulsion equipment, acknowledging the operating environment;
- define the platform loading, impacted by the complement size;
- assess the fuel storage requirements, governed by the required endurance;
- which allows a basic cost estimate, tempered by what is affordable;
- which leads back to a need to reassess physical size for the elements already calculated;

At which point even if the external requirement's remain constant there is a need for multiple passes of the cycle of iteration until a solution can be defined, if such a design meeting all the requirements is possible at all.

2.2 Growth

It is due to the abundance, variety, and the interaction between fundamental requirements that the actual design process is better represented as a design spiral, as shown in a simplistic fashion in Figure 3. The design then matures in a gradual fashion until it concludes with, in an ideal world, the optimum design, although more probably an acceptable design with an acceptable balance between the fulfilments of the requirement set.

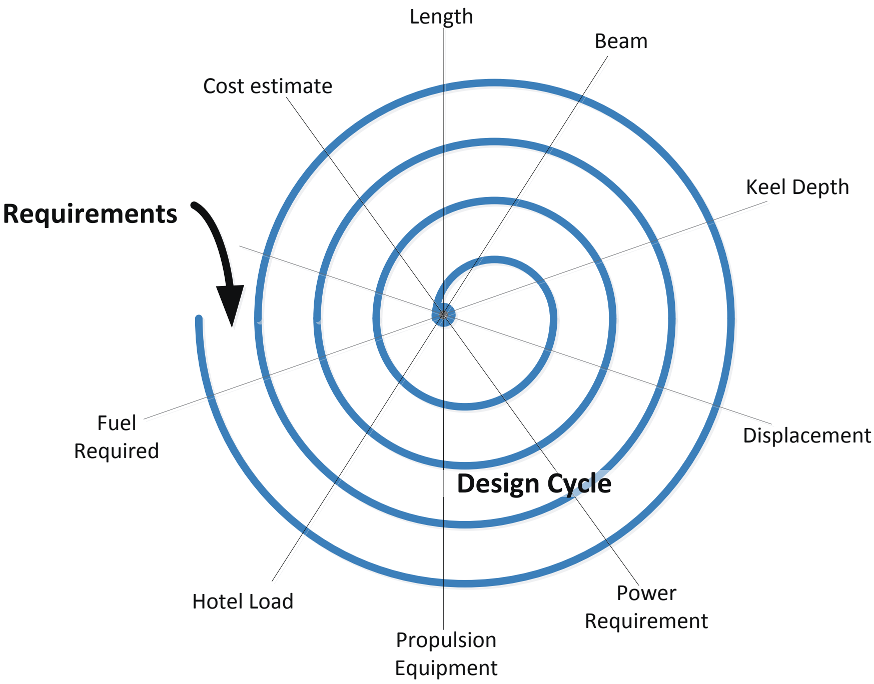

Fig 3: Design Spiral

With both the cycles and spiral patterns shown it can be seen how it is possible to progress towards an acceptable design given the appropriate requirements upon which to progress. The reality of many platform design programmes however is not as simple. It is unlikely that throughout a design phase the requirement set remains fixed. When the impacts of some requirements are seen and their overbearing influence on the design is highlighted it is far from unusual to see requirement change. In a similar manner requirement 'creep' is a real phenomenon experienced in many programmes. Most predominantly this occurs upon the introduction of 'new' customer personnel, either through the customer team enlarging to monitor the ever growing design or even if the customer representative changes throughout the design cycle. This allows the 'what if…' 'can you just…' 'a bit more…' discussions to creep into the design and, in most cases unwittingly, delay and enlarge the spiral development required. These changes can have a significant impact on cost as the 'change' could have an impact on the safety case which then needs to be re-evaluated with the relevant stakeholders.

 http://doi.org/10.24868/issn.2515-818X.2018.009

There will always be changes to the requirements as the reality of implementing them becomes known, or as limitations caused by the requirements, or limitations in the requested capability are exposed and the design spiral allows those requirement changes to be accommodated. Resistance should however be applied to change to ensure that only 'required changes' are adopted and 'change for the sake of change' is prevented.

The inevitability of change has to be accepted, but resisted, with perhaps the optimum way to manage this being that of the identified platform "guiding mind." The prime responsibility 'guiding mind' must rest with an identified person, or small team responsible for the activities of how the capability is to be realised, and the known constraints, advantages, and advances; the detail changes because of perceived limitations in one area can be prevented if the compensation elsewhere is more fully understood and managed by the 'guiding mind'.

2.3 Through life Change

The fact that the functional requirements can change through life is an additional concern, not so much for the original designer, as any facilities to aide such changes will have been specified during the design, but for the owner, user, and modification team.

When modifying the platform in any sense there is the danger of inadvertently impacting existing functionality. The iterations between the design elements, to meet the original requirements, which lead to the need for spiral development also means that there were significant interaction between the different elements of the platform design, which when modified have a resultant effect that may not be appreciated at the time of change.

These interfaces, and dependencies could have been captured within the original design providing this was requested as part of a documented deliverable set, however there will almost certainly be intricacies within that were either not recorded fully, or that were never fully appreciated at the time of design. The potential for significant change suggests that from the outset there needs to be consideration of a party to remain as the authority regarding the design, which have the design knowledge, and the knowledge of the intent of the guiding minds, and can sustain this capability through the platform life. While this does not remove the dangers associated with change entirely, it does begin to minimise the risk.

Through life changes however small, will still require the 'safety case' to be reviewed to ensure the platform remains as an inherently safe design, failure to adequately record the initial design intent could mean reverse engineering the system under consideration to ensure the change does not result in a key hazard from arising. The same principle applies to any other 'through life cases' that exist, such as information assurance.

3 Requirements Transverse

In addition to those functional requirements that express the capability in a tangible sense there are also the abundance on requirements that pervade through the entirety of a platform without being able to be tied to a particular capability. These are unlikely to be fully understood at the requirements definition phases, and their impact may not be understood for a further time still.

3.1 Explicit

There are the explicit requirements that can be numerically expressed, instances could be *'survive a shock impulse of a given magnitude, have an external noise of less than a given decibel figure, have a statistical availability number in each operating year'*. Such requirements do not have an immediate drive on the design yet pervade through each and every system, often becoming a much bigger influence on system designs than some of the functional requirements.

3.2 Implicit

There are also those requirements that pervade through the entirety of the system design, and yet while often stated as a design aim, are often not specific either, such as ensuring a safe, or secure platform. The reality of these types of requirements is an enormity of design and verification effort. Examples of the type of process that must be considered can be seen within regulations for the nuclear industry [Ref 1]. The onus on demonstrating compliance with such a requirement is always on the platform provider and often the effort that this requires is unappreciated by those setting the targets. Many of engineering practices that such regulations invoke are, by their very nature, good engineering practices that are inherent in good engineering designed solutions, but the act of documenting them in a full and traceable manner is labour intensive and can be very expensive as demonstration of compliance has to be held and demonstrable at any point throughout the life of the platform.

One of the principles brought out is that of designing for 'Defence in Depth.' In the simplest expression possible, this is designing to ensure a single failure cannot cascade into a catastrophic event. Expressed as a 'bow tie' diagram [Figure 4], it becomes clear that there may be many threats that could lead to a hazardous event, but there should also be many safeguards the prevent it occurring. The right hand side of the bow tie, is a representation of how, if a hazardous event does occur, the consequence is mitigated, or limited, for which again there should be multiple options.

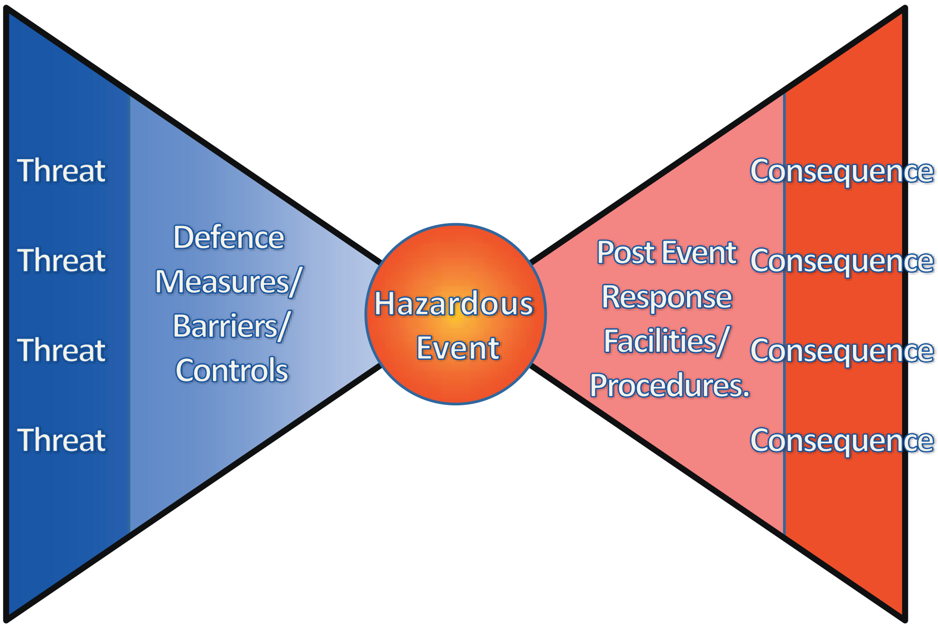

Fig 4: Bow Tie Diagram

Defence in depth			
Level	**Objective**	**Defence/Barrier**	**Guidance**
Level 1	Prevention of abnormal operation and failures by design	Conservative design, high quality in construction, maintenance and operation in accordance with appropriate safety criteria, engineering practices and defined quality levels.	Compliance with specified standards.
Level 2	Prevention and control of abnormal operation and detection of failures	Control, limiting and protection systems, other surveillance features and operating procedures to prevent or minimise damage from failures.	Compliance with specified standards and outcome from the risk analysis of the system under consideration..
Level 3	Control of faults within the design basis to protect against escalation to an accident	Engineered safety features, multiple barriers and accident or fault control procedures	Compliance with specified standards, outcome of risk analysis and requirements from the customer.
Level 4	Control of severe ship or infrastructure conditions, in which the design basis may be exceeded, including protecting against further fault escalation and mitigation of the consequences of severe accidents	Additional measures and procedures to protect against or mitigate fault progression and for accident management.	Examples include fire, flooding or grounding of the vessel. Total loss of electrical power.
Level 5	Mitigation of accident consequences through emergency responses	Emergency control and on- and off-site emergency response (e.g. salvage, fire-fighting tugs, etc).	Example: Vessel/platform out of operation (not under command) and drifting towards a main shipping lane, terrorist attack, earthquake, flooding event. This can be considered by the builder but ultimate responsibility lies with the client and operator/

Table 1: Defence in depth

The analysis to derive the required levels of 'Defence in Depth' and the mitigation against a hazardous event occurring should not be underestimated and changes to the requirements would require the initial results to be re-evaluated.

If these processes and methods are utilised from the outset then the actual establishment and embedment of these multiple levels of design are easily incorporated. Extra effort is required to document the process and show that all threats and possible consequences are understood, and this level of documentation, or the 'burden of proof' should be managed when assessing the potential consequence. For example, if we apply this process to the design of a small unmanned survey vehicle, the largest consequence could be loss of the craft itself, at a cost in the region of tens of thousands of pounds, at which the level of proof required would be incredibly minimal; If we apply the same process to a military ship, or cruise liner, the consequence could be major loss of life, or major environmental disaster, in which case the burden of proof would be expected to be much greater.

You will note that at 'Level 5' Mitigation of accident consequence through emergency response' cannot be solved by an individual manufacturer, this will require the various stakeholders, including potentially governments, to help manage this risk, but ultimate responsibility lies with the client and operator.

3.3 Emerging

The idea of adopting measures like that described in the previous section for management of requirements such as safety are nothing new, indeed have been employed on naval platforms for multiple decades, however as the awareness of hazards grows there are ever emerging new requirements. The threat of insecure software is one that has emerged over the last ten years with many well documented instances where corrupt or maliciously modified software has caused hazardous events, and even catastrophic consequences. When these new hazards, or requirements (which are normally a response to a hazard), emerge consideration has to be given to how these can be managed also. In many senses, a similar philosophy to that described above is equally as valid, with adjustment made for the hazard that is being 'protected' against. In a similar manner to the guidance for safety in nuclear installations, there exists guidance for security [Ref 2]. The Security Plan Principal given is that of 'Secure by Design' and mimics the early discussed 'Defence in Depth' albeit in a pyramid style diagram [Figure 5].

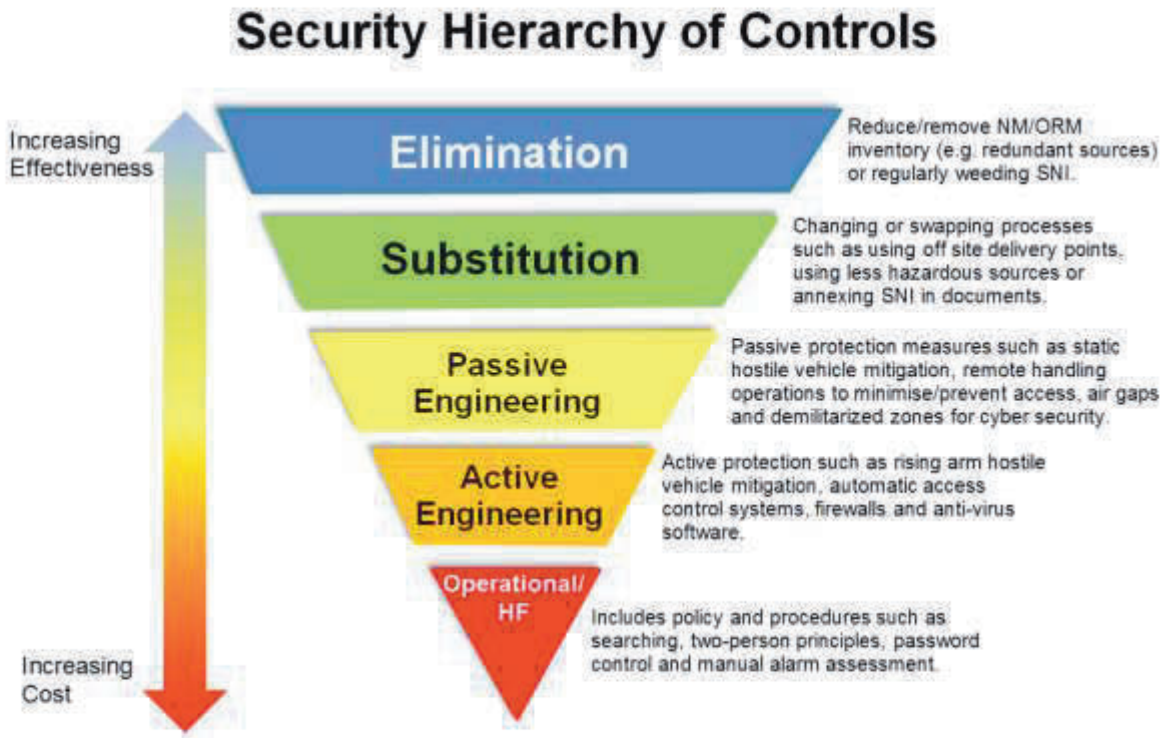

Fig 5: Security Hierarchy of Controls

The philosophy of this analysis and mitigation process follows some common steps:

- Identify the hazard
- Reduce or remove the hazard
- Prevent the hazard causing an incident
- Prevent an incident cascading
- Respond appropriately in a catastrophic event

As with all such philosophies the further down the list we step the more invasive and costly the resolution becomes. The final steps, that are generally a manual intervention, are procedural to reduce the impact of an event, when the other measures have proved insufficient and the hazard has materialised, and then cascaded.

While this may at first seem to be an over the top analysis, this is ultimately also reflected in the hierarchy of risk analysis championed by organisation such as the UK Health and Safety Executive [Ref 3]. The variable in this, as was discussed earlier, is the burden of proof that is determined as appropriate for the level of potential consequence.

4 Expression of adherence (Documented Burden of Proof)

4.1 *Interfaces and Integration*

As was discussed earlier, many of the principles associated with the requirements, such as inherent safe design, are general engineering principles that will by the nature of being developed by a suitably qualified

 http://doi.org/10.24868/issn.2515-818X.2018.009

and experienced design team be embedded within the design itself. The difficulty and variability comes from the need to document such processes and embedded 'goodness.' The traditional V diagram for requirements tracking and verification is tried and tested for demonstration of functional requirements where the successful implementation of a function can be tested or demonstrated in a 'witnessable' fashion. Where requirements cannot be proved by inspection or test, the traditional V diagram approach becomes less useful. It is here where, for instance, tracing things such as interfaces across the platform becomes more important. Many of the non-witnessable requirements are those that require a detailed understanding of the integration of the systems embedded within a platform, and more often than not also require knowledge of the principles of operation for the platforms systems.

It is in agreeing to the satisfaction of the customer that these requirements have been met that has the potential to trigger huge variability in the level of effort, and therefore the cost expended. It becomes important from the outset, for the benefit of both the customer and the design team, to have a clear understanding of who is going to approve the sign off of the requirements and what their expectations are. With short duration projects this can be fairly simple, however with naval platforms whose programmes extend for decades then this becomes ever more difficult and potentially enables another change of scope, while never modifying any of the fundamental requirements. Changes such as these may also be introduced by outside parties, such as regulators, classification societies, or insurance companies; where both the customer and design may not have ultimate control of the effort that the must expend.

4.2 Compromises

In proving the system is meeting the requirements to the satisfaction of the customer there has to be an element of realism regarding what can be proved. There will be requirements, either explicit or driven in by the need for safety, that dictate how the system will perform when the platform is taken beyond its normal operating envelope. To demonstrate these on the physical product puts the platform and people at unacceptable risk, by purposely moving into areas of abnormal, or emergency, operation.

Compromise must be reached as to how these things should be demonstrated. Some elements of the functionality may be demonstrable as an exercise, such as life craft drills, prove the ability to evacuated passengers and crew of many commercial surface vessels, but some require much more integration of systems. For these integrated functions it may be more appropriate to run 'desktop' simulations of the scenario with an independent, but suitably knowledgeable, oversight assessing where the system interactions will occur and the potential success, or failure, of each scenario. The additional benefit of work such as this is to try and reveal where any inter-system cliff edges actually manifest themselves.

4.3 Risk

The difference between those requirements that can be physically be demonstrated and those that are proven by examination, or analysis; in many ways is the expression of risk, however this view also misses a fundamental limitation. In the same way the a UK car MoT test proves that a car was fit to be on a UK road at the time of test, the sign off of requirements witnessed, shows that the system performed as expected on the day of test. The ongoing ability of the system to perform as designed, and as previously demonstrated, becomes a through life maintenance and management issue, for the remainder of the life of the platform. If this risk is to be continually minimised, the platform must be managed appropriately through life, with the continued guidance and oversight of a recognised technical authority.

5 Managing Competing or conflicting Requirements

5.1 Requirements Engineering

The IREB definition of Requirements Engineering (Pohl & Rupp 2011) [Ref 4] says:

> *Requirements Engineering is a systematic and disciplined approach to the specification and management of requirements with the following goals:-*
>
> - *Knowing the relevant requirements, achieving a consensus amongst the stakeholders, documenting and managing them systematically.*
> - *Understanding and documenting the stakeholders desires and needs*
> - *Specifying and managing requirements to minimize the risk of delivering a system that does not meet the requirements.*

 http://doi.org/10.24868/issn.2515-818X.2018.009

5.2 Definition of Requirements Conflict

Conflicting requirements is a problem that occurs when a requirement is inconsistent with another requirement [Ref 5]. Consistency between requirements requires no two or more requirements contradict each other . In requirements engineering, the term conflict involves interference, interdependency or inconsistency between requirements .

Kim et al. [Ref 7] gives a good definition of requirements conflict as:

> *"The interactions and dependencies between requirements that can lead to negative or undesired operation of the system"*

An example of a conflict in non-functional requirements can be the gap between performance and security; when the client wants certain functionality to be satisfied in minimal time (e.g. calculate something and display it on screen), as well as the use of a secure protocol for data transfer and access control.

The causes of requirements conflict are well documented [Ref 8], but the challenge is how to avoid these at all costs.

5.3 Managing Requirements

There are a number of tools that can help reduce the risk of managing the requirements and avoiding unnecessary requirements drift. Model Based Systems Engineering is the formalized application of modelling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycle phases [Ref 9].

The challenge is the consistent use of the MBSE tool by all relevant stakeholders as a failure to do so will not provide the benefits that can be achieved by the use of the tool. The model shown in Fig 6 shows a strategic view, operational view and systems view that are inherently linked.

If the strategic objective is to deliver a military capability then having a full understanding of the operational requirements is vitally important, the challenge then is to consider the systems view as what may be required by the operational aspects may not be possible within the constraints of the strategic objectives.

The use of MBSE can help ensure the rationale behind requirements are clearly understood by all relevant stakeholders and the MBSE process/tool can be used to evaluate any changes to the requirements.

 http://doi.org/10.24868/issn.2515-818X.2018.009

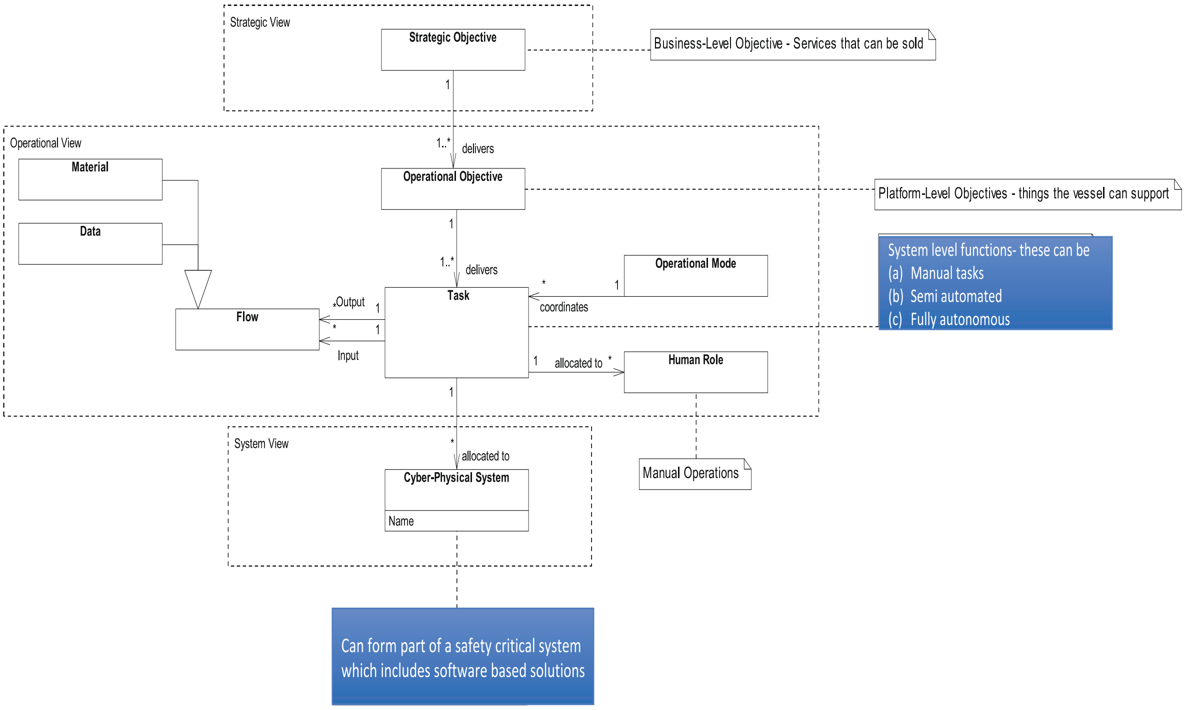

Fig 6: Example of MBSE

6 Through-Life Considerations

To maintain the safety argument someone has to remain the owner of these safety/security/cases through life with the understanding of how they map against the original requirements. Without this the ship/platform owner becomes totally responsible, even if they are not knowledgeable, they are ultimately responsible for maintaining the safety and security of the ship/platform.

As we move towards greater integration and complexity along with the move towards delivery of the Operate/Maintain/Diagnose/Repair (OMDR) philosophy both for the present day and into the future, consideration has to be given to the training of the owner and maintainers to ensure they fully understand the requirements for maintaining the through life safety argument.

It is not realistic to expect the owners, operators and maintainers to fully understand the design intent of the system under consideration as this requires teams of specialists involved in the design and development of the safety argument. What we should expect is the owners, operators and maintainers fully understand the implications of their requirements and are willing to support their decisions technically and financially.

As an example, during the build process changes to requirements that are not fully evaluated could have a significant impact many years into the build cycle. A requirement change that states *'the software based solution is to be at a SIL 3 level'* is a simple request that can be managed effectively. The cost of meeting this requirement could run into £100,000's, or £1,000,000's when the contract is finally placed with the manufacturer, and this is when the debate starts to take place and reality kicks in and this is where 'trade-offs' are requested, such as change SIL 3 to SIL 2.

What is not fully appreciated is the change request which started in year 1, has been factored into several design decisions and all of these design decisions need to be re-evaluated, not just the change from SIL 3 to SIL 2 for that particular sub-system.

Most requirements are achievable, but there is a cost that needs to be accepted, not just at the new build stage but throughout the life of the platform. Maintaining the safety argument comes at a cost.

Reliance on OEM's to provide the support is unsustainable as OEM's can change hands throughout the life of the platform and this could end up as a significant security risk. If this happens what are the contingency plans as the IP resides with the OEM. In this changing environment not considering the through life operation of the safety argument could be extremely expensive.

7 Conclusion

The paper has tried to highlight some of the considerations that need to take place when requirements are being developed and the impact that changes can have on the overall safety argument. Often the full implications of requirements specified early in design stages are not fully appreciated until much later. Most changes can be accommodated and the management of risk becomes a balance between the expected risk, the practicable solution, along with the potential compromises to both programmes and cost.

There is not one simple solution to resolve these issues, but identification of technical authorities, and 'guiding minds' along with tools such as MBSE can be used effectively to understand the impact of a change to the specification, but all stakeholders need to understand the impact of that change at a system level and not just at a sub-system level. It also has to be recognised from the outset that the requirements management and the sustainability of the platform, both for function and non-functional performance is a through-life commitment that must be borne by somebody and identified early in the full product lifecycle.

ABBREVIATIONS

DE&S	Defence Equipment and Support
DoD	Department of Defence
IP	Intellectual Property
IREB	International Requirements Engineering Board
MBSE	Model Based Systems Engineering
MoD	Ministry of Defence
MoT	Ministry of Transport
OEM	Original Equipment Manufacturer
OMDR	Operate/Maintain/Diagnose/Repair
RN	Royal Navy
SIL	Safety Integrity Level

ACKNOWLEDGEMENTS

The authors would like to thank the various stakeholders for providing the 'cases' upon which the paper is based.

REFERENCES

1. Safety Assessment Principles for Nuclear Facilities, Office for Nuclear Regulation, 2014 Edition, Revision 0
2. Security Assessment Principles for the Civil nuclear Industry, Office for Nuclear Regulation, 2017 Edition, Version 0
3. Management of Risk when Planning Work: The right Priorities, Health and Safety Executive, HSE 11/11
4. Pohl and Rupp - Requirements Engineering Fundamentals ISBN 9781937538774

5. B. Schar, Requirements Engineering Process HERMES 5 and SCRUM, University of Applied Sciences and Arts, Northwestern Switzerland, 2015.
6. Aldekhail, Chikh, Ziani - (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 7, No. 10, 2016
7. M. Kim, S. Park, V. Sugumaran, and H. Yang, *Managing requirements conflicts in software product lines: A goal and scenario based approach*, Data Knowl. Eng., vol. 61, no. 3, pp. 417-432, Jun. 2007.
8. W. N. Robinson, S. D. Pawlowski, and V. Volkov, *Requirements Interaction Management*, ACM Comput Surv, vol. 35, no. 2, pp. 132-190, Jun. 2003.
9. L.E.Hart - INCOSE SE Vision 2020 (INCOSE-TP-2004-004-02, Sep 2007) https://www.incose.org/docs/default-source/delaware-valley/mbse-overview-incose-30-july-2015.pdf

DISCLAIMER

This paper represents the opinions and views of the authors, and does not necessarily represent those of BAE SYSTEMS plc, Rolls Royce plc, or any of their subsidiary undertakings.

 http://doi.org/10.24868/issn.2515-818X.2018.009