

23rd ICCRTS

Model Based Systems Engineering Process for Complex Command and Control Systems

Topic

Topic 8: Methodology, Experimentation, Analysis, Assessment and Metrics

Authors

Dr. R. Oosthuizen
(CSIR, South Africa; ROosthuizen@csir.co.za)

J.P. Venter
(CSIR, South Africa; JPVenter@csir.co.za)

C. Serfontein
(CSIR, South Africa; CSerfontein@csir.co.za)

Point of Contact
Dr. Rudolph Oosthuizen
(CSIR, South Africa; roosthuizen@csir.co.za)

Name of Organization: Council for Scientific and Industrial Research
PO Box 395
Pretoria
0001
info@csir.co.za

Model Based Systems Engineering Process for Complex Command and Control Systems

"How can I tell what I think 'til I see what I say?"

E.M. Forster

Abstract

Systems Engineering for complex and sociotechnical systems is becoming increasingly difficult due to growing complexity in the problem and solution space. This resulted to the development of Model Based Systems Engineering (MBSE), which applies modelling supported by an architecture framework, modelling language in a software tool. Modelling is applied to capture and represent the mental models of the systems' stakeholders during the requirements analysis and concept development stages. The software tool enables capturing model information in a database through various diagrams and views. This data centric approach ensures consistency and traceability of the defined model elements and their relationships. Since Command and Control (C2) systems can be viewed as complex sociotechnical systems, MBSE should be useful for capturing requirements to support development of concept solutions. The MBSE method presented in this paper assists capturing of requirements for a C2 system as well as identification of the main logical block elements and their required functionality. This approach is demonstrated through development a concept C2 system solution suited to the complex operational environment of Special Operations.

1 Introduction

The basic objective of Systems Engineering is to solve problems by bringing systems into being (Stensson 2010, Walden et al. 2015). The systems approach, through systems thinking, aims to understand the part in the context of the whole, while interacting with and adapting to the environment. Systems engineering consists of interdisciplinary activities to ensure that the stakeholders' needs are met in a cost-effective and timely manner. However, today's problems tend to be complex and ill-defined with a wider impact than the system of interest.

In essence, the Systems Engineering process is an iterative method for discovery, learning and continuous improvement to gain insight on requirements and emergent properties (Walden et al. 2015). The input to the Systems Engineering process is a need from a customer or stakeholder which must be analysed to discover requirements for defining the purpose, objective and high-level functions of the solution system (Buede 2000, Ramos et al. 2012, Walden et al. 2015, Oliver et al. 2009). The operational context and expected scenarios are defined from the initial high-level requirements to derive the required roles, tasks and functions of the system (Stanton et al. 2012).

Holt & Perry (2008) list the three evils of Systems Engineering as complexity, communication, and understanding. Level of system complexity depends on the number of system elements and their interaction. An improper understanding of the problem and user needs leads to inaccurate requirements and the improper application of Systems Engineering. Communication problems between engineers, the development team and the stakeholders lead to interpretations of the meaning of requirements and associated models. This can be further exacerbated by poor communication between the design team and manufacturing teams. The demands on Systems Engineering to produce systems capable of effective operation within complex environments are ever-increasing. Modelling is one way to address the complexities and enable effective understanding and communication.

Model Based Systems Engineering (MBSE) as a method, provides a solution that consists of a process and supporting tools to address the complexity. Various forms of models and constructs can be used to capture and represent the information and knowledge on problems. A suitable model, derived from the stakeholders' mental models, is used to absorb and interpret important information (De Weck et al. 2011, Sterman 1994).

In Systems Engineering, models are constructed through schematics and network diagrams of the system. Modelling can assist system design by understanding the relationship between the system as a whole and its parts to derive emergent properties (Buede 2000, Ramos et al. 2012, Maria 1997, Ramos et al. 2011). Models also tend to be more useful than text-based Systems Engineering documents to develop system concepts and requirements as they support experimentation with knowledge on the problem and develop an understanding of the implications of different solutions (Estefan 2007).

MBSE employs modelling languages, such as Systems Modelling Language (SysML) or Unified Modelling Language (UML). The language utilises a graphical notation, enhanced with relevant parameters, attributes and qualifying information. These languages are used to model complex systems and their architectures through diagrams that capture, analyse, and specify behaviour, structure, requirements, relationships, and capabilities of a system through various consistent viewpoints. The different viewpoints of the model may be used to experiment with knowledge on the problem and to develop an understanding of the implications of different solutions (Ramos et al. 2012, Buede 2000, Oosthuizen and Venter 2016). Other advantages of employing MBSE in the planning include management of problem complexity, reuse of information elements and supporting of systemic thinking (Ramos et al. 2012, Walden et al. 2015).

This paper reports on practical experience gained through implementing a MBSE process for conceptual modelling of a C2 system. Firstly, Systems Engineering and modelling literature are discussed to define a MBSE process before relating it to practical experience. The process is demonstrated by developing a concept C2 solution tailored for special operations. Special operations tend to operate in a more complex environment than conventional forces with unique requirements.

2 Command and Control Concepts

C2 is a collective term to address multiple aspects of command and control of military operations. Command is defined as the authority vested in or delegated to an individual of the armed forces for the direction, coordination, and control of military forces (Brown 1996, MoD 2017). This represents the creative expression of a human will for articulating and delivering the intent necessary to accomplish a mission. The chain of command network describes the flow of delegated authority with its mandate to generate intent. Command emanates from the individual who possesses the most information on the status of mission (Young 2017).

“Control” is defined as the authority exercised by a commander over part of the activities of subordinate organisations that encompasses the responsibility for implementing orders or directives. Control includes the structures and processes to manage risk through setting boundaries, managing resources, and adjusting action to achieve the Commander’s intent. The purpose of control is to ensure that the stated intent is achieved through adjustment of required activities (Brown 1996, Young 2017).

The purpose of a C2 system is to direct, align and coordinate the intent and activities of the force across multiple domains. The C2 system has to be agile to adapt in order to meet the requirements of a changing environment, context and mission (Oosthuizen and Pretorius 2013, MoD 2017).

A C2 capability is further defined as a dynamic and adaptive (complex) socio-technical system to design and execute joint action (MoD 2017). The complex socio-technical system encompasses behaviours, tools and techniques, enabled by the interactions between people, structures, technology and processes. C2 provides the focus for individuals and organisations to integrate and maximise their resources and activities to achieve desired outcomes (MoD 2017). The specific C2 solution is characterised by the degree of network decentralisation required for the type of environment and mission. The appropriateness of a specific C2 solution is influenced by the following three environmental variables (Young 2017):

1. Environment Rate of Change. Unusual mission types lead to task uncertainty. Also, a faster rate of change during the mission requires a greater degree of situation awareness decentralisation and update rate.
2. Degree of Element Connectivity. Multiple independent elements in an environment are simpler to analyse and understand than a complex environment with multiple and highly connected elements. Tasks in a complex environment are harder to decompose, requiring a greater degree of decentralisation of control.
3. Degree of Stakeholder Interest. Stakeholders impose standards, constraints or expectations on the conduct of the activities. The degree of interest has an inverse relationship with the degree of centralisation required in command.

The chosen C2 approach has to be adapted to the applied operational environment. The contemporary Special Operations operational environment is complex. The environment is characterised by a global system of systems, comprising of numerous interacting variables that create intertwined national, political, economic, social, spiritual, cultural, and military interests, challenges, and threats. The main factors affecting the complex environment for Special Operations include the following (Ott 2002, Madden et al. 2016, Johansen 2015, Votel 2016):

1. Globalization. Globalization, as shaped by advances in technology (internet, cellular phones, satellite communications), have exponentially increased the rate of change, innovation, and commerce. Technologies allow instantaneous information transmission around the world. International terrorist organizations can operate effectively from remote regions, while being

supported (finance, training, and recruitment), from anywhere around the globe. Critical events are also globally reported by the media as they occur, in near real-time.

2. Urbanization. An increasing percentage of the world's population migrates to urban areas. Belligerents increasingly seek the sanctuary of urban areas to hide among the population. This mitigates the technological advantage of weapons and sensors to increase the threat of collateral damage from strikes.
3. Geopolitical Trends. More political power and military capabilities are becoming available to non-state actors. This leads to complex regional security developments and threats emanating from territories of weak and failing states. These threats cannot be addressed in isolation.
4. Military-technical Trends. Military technologies continue to rapidly advance, leading to emergence of new military competition arenas. Therefore, operators may be surprised by the capabilities of a new belligerent. Adversaries have access to advanced weapon systems and cyber capabilities that provide an expanding variety of coercive options at their disposal.
5. Remote Operational Areas. Rugged, mountainous terrain, caves, harsh climates may limit the capabilities of weapons or operational support. Terrorist organizations tend to train, sustain, plan, and operate in these remote and unstable regions. To operate in remote areas, the Special Operations personnel have to make contact with the indigenous population and leaders, gain their acceptance and confidence to gather intelligence in support of operations.
6. Mode of Conflict. The mode of conflict is changing in response to the environment. Low-intensity and asymmetric conflicts are unfolding in 'less developed' parts of the world between regular soldiers on one side fighting guerrillas, terrorists, and even civilians. These conflicts tend to be brutal, bloody, and ugly without a reliance on high-technology weapons of modern armed forces. The aim is to destabilize a regime with outside help. Irregular threats typically thrive on the fringes of state controlled territories and in the ungoverned spaces.
7. Duration of Operations. Fiscal and political sustainability of a Special Operations makes them more attractive. The small-footprint approach of Special Operations allows cost-imposing strategies that force opponents to spend disproportionate resources to defend against friendly capabilities. They support a deeper understanding of the environment to predict flashpoints of instability and reduce operational and strategic blind spots.

The implemented C2 system requires the ability to collect, distribute and transform data into intelligence to execute rapid decision-making. Decisions have to support the ability to direct forces, across multiple domains and missions. Effective C2 depends on the decisions for action based on an understanding of the situation. Understanding is defined as the perception and interpretation of a particular situation in order to provide the context, insight and foresight required for effective decision-making. Understanding enables decision makers to map emerging patterns and identify new opportunities and threats from available information using data analytics and visualisation (MoD 2017).

Standard and traditional Systems Engineering methods may not be adequate to support development of the systems required for these complex environments. These complexities may lead to vague and integrated requirements for the C2 system. The Systems Engineering fraternity is moving towards MBSE to cope with development of systems for complex problems (Walden et al. 2015).

3 Model Based Systems Engineering for Concept Development

3.1 Model Based Systems Engineering

Modelling is an iterative process to develop, use and update models through a standard, rigorous, structured methodology to obtain insight into a system's behaviour. A model is defined as an explicit and incomplete representation or idealized abstraction of reality, or a selected part thereof, to aid its description and understanding (Ramos et al. 2012, Maria 1997). The model must achieve a balance between realism and simplicity, to enable understanding and simulation (Maria 1997, Gau Pagnanelli et al. 2012).

In Systems Engineering, models are used to describe the structure, behaviour, operation and characteristics associated with a system as well as selected interactions with its operational environment. The environment includes enabling systems and other systems (Buede 2000, Hitchins 2008). The act of modelling and the resulting models themselves support insight into a problem as a basis for making decisions (Maria 1997, Harrison et al. 2007, Buede 2000). Models are utilized to experiment with knowledge on the problem and to develop an understanding of the implications of different solutions. This support improved solution development decisions by clarifying requirements with visual tools (Walden et al. 2015, Ramos et al. 2012).

Models tend to be more useful than text-based documents to develop system concepts and requirements as common notations depict many system characteristics and attributes in a consistent way (Buede 2000,

Ramos et al. 2012). The main advantages of modelling for problem solving in Systems Engineering can be summarised as the following (Buede 2000, Ramos et al. 2012, Maria 1997, Walden et al. 2015):

1. Manage Complexity. Modelling help to address complexity in system analysis and solution design in the following ways:
 - a) Enable system understanding through visual analysis of multiple perspectives on the problem.
 - b) Guide identifying the elements and interactions that cause complexity.
 - c) Improve impact analysis to identifying potential consequences of a solution.
 - d) Discover causes and effects using model traceability.
2. Capture the Problem in Systemic Terms. Stakeholders check their own thinking (mental models) and communicate concepts to others to confirm system needs and behaviour. Models support improved system development decisions through clarifying requirements.
3. Capture, Analyse, Share and Manage Information. The benefits of having a modelling approach with a standard language are improved communication among stakeholders, ability to manage system complexity, product quality, knowledge capture and ability to teach and learn Systems Engineering fundamentals. Models support experimentation as well as qualitative and quantitative analysis of knowledge on the problem.
4. Reuse and Automation. Modelling enables reuse of existing information and knowledge in new projects, saving time and money. Modelling also facilitates automation during problem analysis by performing repeatable tasks with scripts.

However, modelling of a complex and sociotechnical systems still remains difficult as it has to present the structure and behaviour of human work in the system. Behaviour is caused by dynamic interaction between the humans, operators, system elements and the environment. Modelling of system interfaces at multiple levels of abstraction may aid in understanding complex systems and their interactions (Bahill & Szidarovszky 2009, Piaszczyk 2011, Oosthuizen and Pretorius 2014).

MBSE is defined as the formalized application of modelling to support system requirements, design, analysis, verification and validation activities throughout the system life cycle phases (Gau Pagnanelli et al. 2012). MBSE focuses on the application of information rich models to complement traditional Systems Engineering methodologies (Walden et al. 2015, Estefan 2007). Any MBSE implementation should address at least the following elements (Meyer 2014, Tschirner et al. 2015, Gau Pagnanelli et al. 2012, Karban et al 2012):

1. Process. The process provides the logical sequence of tasks performed at various levels of detail and aggregation to achieve a particular objective. Existing MBSE processes and need to be tailored per specific implementation environment. Basic steps of a MBSE process should include information (requirement) capturing, model generation, validation and implementation. The process needs not be the same as for traditional Systems Engineering standards.
2. Method. The method consists of the set of techniques for performing a task as listed in the process. Each method may also be a process in itself.
3. Toolset. The tools are the instruments that enhance the efficiency of the tasks in a specific method. The tools only facilitate how the tasks are performed and provide the repository of the information captured in the model.

MBSE is not only about developing a set of diagrams to be pasted into text based reports. Drawing these diagrams using standard modelling notations alone will not improve the models. The meta-model for MBSE (Holt and Perry 2017), as presented in Figure 1, indicates the relationships between the system being modelled and the application of a modelling tool within an architecture framework. This meta-model provides the basis for development of a MBSE methodology suited to a specific application domain.

Before modelling a system through a number of views, as per the middle “Goal” column of Figure 1, support structures need to be in place. Firstly, an architecture is required to define the views for the model. The modelling method applies an architectural framework. Any suitable architecture framework (e.g. on Service Oriented Architecture, NAF, MoDAF or DODAF) for the problem and solution space may be applied by the MBSE method. In all cases some level of tailoring will be required. The architectural framework consists of various viewpoints required to describe a system and its behaviour as defined by a standard ontology. Different views on the system model are used to analyse the system at different levels (Kossiakov et al. 2011, Ryan et al. 2014).

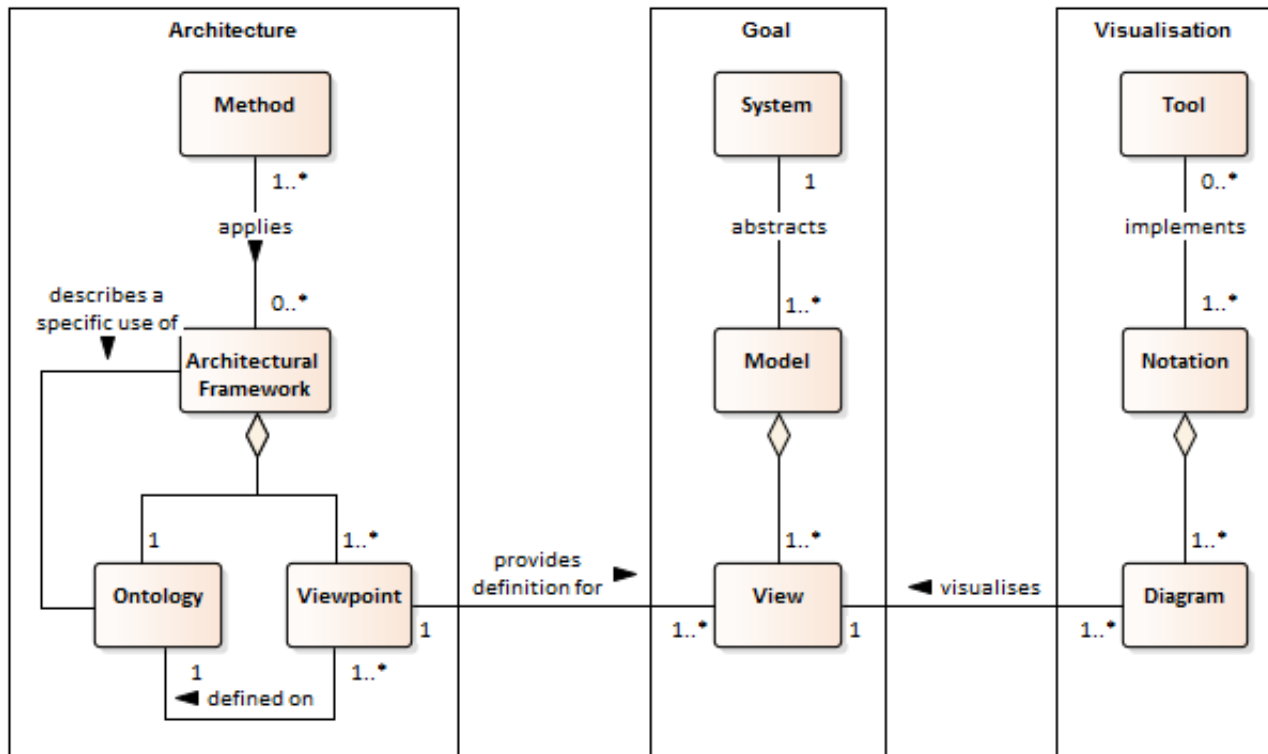


Figure 1: Model Based Systems Engineering Meta Model

The various views of the system are visualised through diagrams. To embrace the power of MBSE, the modellers should advance beyond text based tools. These text-based tools, such as Visio™ and Powerpoint only provide pictures for text based documents. They fail to manage the relationships between elements in complex diagrams. Dedicated data centric tools for MBSE should be implemented. These tools implement accepted modelling notations, such as SysML and the UML, to develop consistent system views through diagrams. SysML is used to model complex systems through diagrams of system structure, parameters, attributes, requirements, behaviour and relationships (Friedenthal et al. 2012, Hause 2014).

The behavioural diagrams represent the parts of a situation and their causal interactions (Friedenthal et al. 2012, Hause 2014). The system's behaviour is modelled with use case, activity, sequence and state machine diagrams. The use-case diagram provides a high-level description of the system functionality. The flow of data and control between activities are captured in an activity diagram. A sequence diagram represents the interaction between collaborating parts of a system. Requirements diagrams ensure that hierarchies and the derivation, satisfaction, verification and refinement relationships are clear. System parameter constraints and other parameters such as performance, reliability and physical properties are captured in the parametric diagram (Hause 2014).

Structural diagrams represent the parts of a situation with their logical relationships (Friedenthal et al. 2012, Hause 2014). Block definition diagrams describe the system structure through hierarchy, relationships and classification. Internal block diagrams describe the internal structure with parts, ports, and connectors, of a system. The blocks in the block diagrams can represent any level of the system hierarchy to describe a system as a collection of parts and connections between them that enable communication and other forms of interaction. Ports provide access to the internal structure of a block for use when the object is used within the context of a larger structure. The requirement views specify desired structural and behavioural properties as derived from stakeholder needs. Parametric views provide the critical engineering parameters of the system for evaluating performance, reliability and physical characteristics (Friedenthal et al. 2012, Hause 2014).

Implementing this meta-model of MBSE will assist Systems Engineers (modellers) to organize and present information to support development of tangible model based requirements to develop solution options. Depending on the modelling tool, the information captured in models can be exported in textual form for review and contractual purposes. MBSE implemented through a data-centric tool enables Systems Engineers to manage the relationships between requirements and the model elements. The effect of changes in requirements can be traced to the model elements they affect. The next section will propose a modelling process for C2 Systems.

3.2 Modelling Process for Command and Control Systems

As seen from the discussion above, MBSE is a modern approach to designing and developing complex systems. In principle, the process follows a top-down application of models instead of a document-based text for specifying, designing, integrating, validating, and operating a system. MBSE employs a process to develop and increase the detail in models using a concurrent and incremental process to support communication between stakeholders (Estefan 2007, Walden et al. 2015, Oosthuizen and Pretorius 2015).

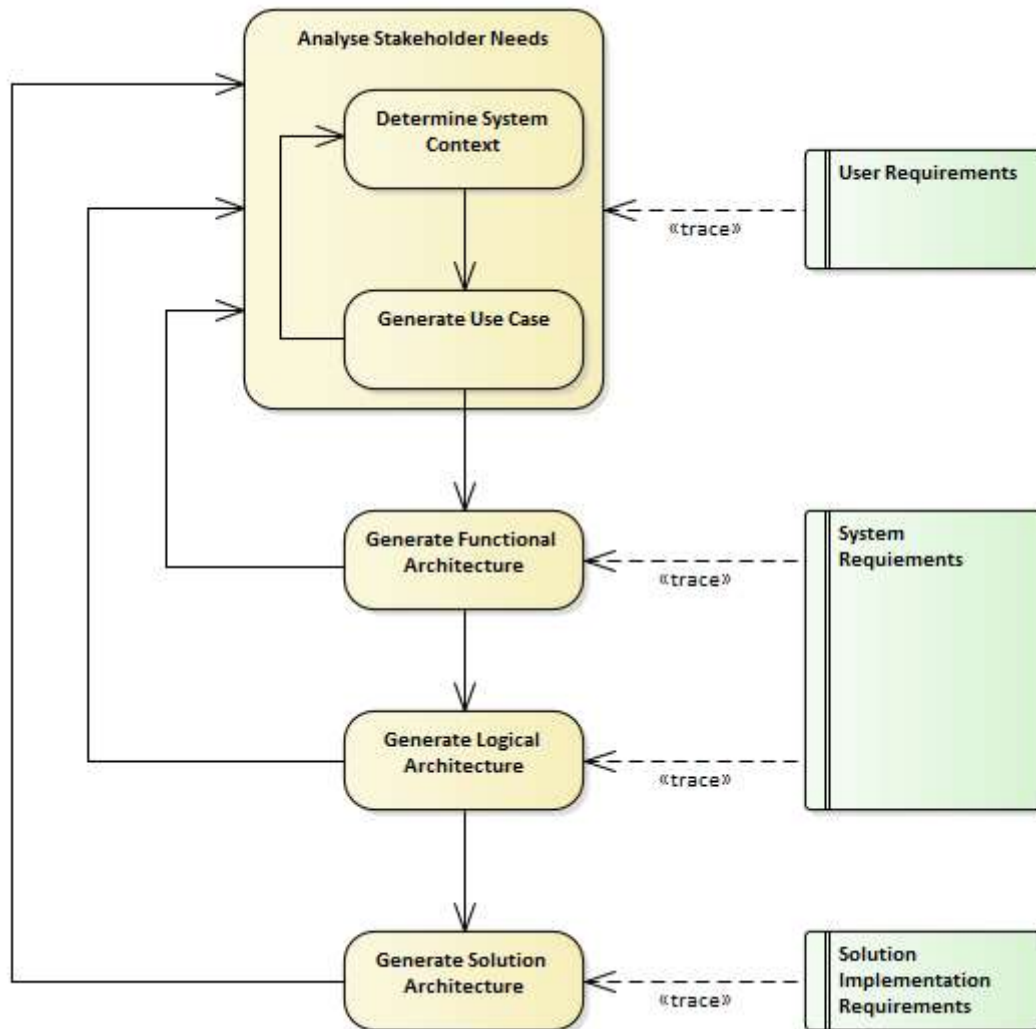


Figure 2: Basic MBSE Modelling Process

The basic MBSE process in Figure 2 for modelling of C2 systems is loosely based on MoDAF. The method is adapted and simplified from the INCOSE Handbook and other sources to include the following activities (Walden et al. 2015, Ryan et al. 2014, Hause 2014):

1. **Analyse Stakeholder Needs.** This activity captures the limitations and potential improvement areas of the “as-is” systems to support developing the “to-be” solution. Existing text-based user ConOps and other requirement documentation may provide a useful input to modelling stakeholder needs.
 - a) **Determine System Context.** The first step is to define the boundaries of the system with the interfaces to external systems or the operational environment. The objective of the system context diagram is to focus attention on external factors and events that should be considered in developing a complete set of systems requirements and constraints.
 - b) **Generate Use Case.** The high level system function is captured from the user requirements in the Use View, which consists of a number of Use Case Diagrams in a hierarchical structure of Packages. This activity defines the system requirements to support the mission requirements. The system is modelled as a black box that interacts with the external systems and users. From the Use Cases the required structure and behaviour of the system are derived through an iterative process.

2. Generate Functional Architecture. The system functions are modelled in the form of Activity Diagrams to define the relationships between the actives derived from the Use Cases. The verbs in the description and naming of the Use Cases should be used to define the activities. Modelling of the functional architecture enables development of functional system requirements.
3. Generate Logical Architecture. The system is decomposed and partitioned into logical elements, which interact, that satisfy the system requirements. The logical elements are derived from nouns in the Use Cases. The logical elements of the system are modelled with Block Diagrams to define the relationships between them. The scenarios are also modelled using swimlanes, which combine activity diagrams with the block definition diagrams. The swimlanes assign activities to the logical block elements as well as presenting a logical flow of activities. Modelling of the logical architecture enables development of non-functional system requirements.
4. Generate Solution Implementation. This step describes the relationships among the physical system elements or nodes that define the distribution of resources. The blocks identified in the in the logical architecture are instantiated through properties of the block diagrams. Internal block diagrams are used to model the interfaces between the system elements along with information elements exchanged. The elements in the logical architecture may be instantiated for multiple possible solution implementations.

The solution architecture is the output of this modelling process and will be used to derive the system's functional, interface, data, and performance requirements. A number of solutions may be generated for input to an option selection process. Depending on the contractual guidelines, the requirements may be exported in a text-based or a repository of the model.

As seen from the connectors in Figure 2, this is an iterative process. Each cycle will improve the completeness and accuracy of the models. The views may also be used to support development of the Concept of Operations (ConOps) for the C2 system. Implementing this process with a software data-centric MBSE tool will ensure that all model elements and their identified are managed to maintain consistency and traceability. This process is applicable to most architecture frameworks.

Aspects not addressed yet in this process include requirements and sequence diagrams; they are required at a lower level where design of the solution is implemented. Other steps not included in the process discussed in this paper include selecting the preferred architecture as well as validate and verification of the proposed solution system. This framework is the basis for developing the process implemented in specifying a Special Operations C2 system.

4 Modelling Command and Control Concepts for Special Operations

This section implements the process in Figure 2 to define a concept solution C2 system for Special Operations. This was done in the MBSE tool Enterprise Architect™ with SysML to generate a number of views on the model. Only a limited number of output diagrams of the models will be provided for illustrative purposes. The information provided in this section will support the demonstration of the modelling methodology in the next section to define a generic C2 system that may be required for Special Operations. The information provided in this section will support the demonstration of the modelling methodology in the next section to define a generic C2 system that may be required for Special Operations.

4.1 Special Operations Background

Each application of the approach for a specific operational environment will result in a different C2 concept solution. The development of a C2 system tailored for Special Operations has to support of the missions, with potentially mission-critical adverse consequences. The key characteristics of Special Operations, which distinguish them from more conventional operations, include the following (Johansen 2015, Kiras 2015, Votel 2016, Brown 1996, Eaton et al. 2017):

1. Capabilities. Special Operations personnel implement different (specialist) capabilities and skills (e.g. hostage rescue) than the ordinary army, navy or air force without incurring high costs and risks of failure.
2. Agility and Flexibility. Special Operations personnel can rapidly seize opportunities and adapt to unforeseen requirements and operational challenges. It enables them to integrate with partners to develop strategic opportunities to counter irregular and hybrid approaches employed by adversaries. They are more flexible to apply force directly (surgical strike) as well as indirectly (special warfare). They can operate isolated (without regular and sustained support) in harsh environments with limited or constrained infrastructure.

3. Scale of Operations. Special Operations tend to be unorthodox, high-risk small-unit clandestine covert or overt operations. Considering the relatively small size of Special Operations teams, as well as the fact that the chain of command is short, teams can be deployed in a much shorter time than a larger conventional force.
4. Persistent and Long Term Operations. Irregular and low intensity conflicts require long term (months and years) and sustained operations to reverse the will of the adversary and root causes of confrontation. This places a demand on continuity and knowledge management at tactical, operational or strategic level. It allows for deep understanding of the human domain, which is necessary to identify and influence relevant actors for acceptable outcomes.
5. Support. Operations are normally conducted outside the national borders, great distances from established operational or support bases behind OPFOR lines. These operations demand specialist support (overt and covert) and cannot be conducted by any conventional force. This requires tailor-made capabilities, protection and logistical support. Special Operations may require clandestine techniques in order to ensure low visibility, while concepts such as stealth, deception, and the indirect approach are always applied. Reach-back to headquarters, access to and maintenance of secure communication channels are challenging for Special Operations.
6. Operational Tempo. Special Operation units can operate at a high tempo, due to their size and flexibility. Special Operations personnel can act early, persistently, and precisely to create the required decision space and strategic options to provide sustainable outcomes.
7. Tools of National Policy. Special Operations personnel are often deployed to implement foreign policy, which increase the degree of sensitivity and risk. Special Operations tactical units can be employed and led from the highest levels of military or national command to achieve specific political or strategic goals.
8. Low Level Decision Making. Although operations are authorised at the highest level, decisions that could have a strategic outcome are often made at the lowest tactical level. Mission command is actively encouraged in the units, and participative decision-making based on individual expertise is the order of the day.
9. Joint, Interagency, Intergovernmental and Multinational Operations. Contemporary Special Operations tend to be Joint, Interagency, Intergovernmental and Multinational (JIIM) and networked with shared objectives and common goals. Objectives have to be achieved by working through, and with indigenous or surrogate forces. Personnel face challenges in coordinating, deconflicting, and leveraging Special Operations activities with other operational participants. Special Operations assets are deployed to collect information that may not be acquired through other means. Information-sharing with external role player may erode trust.

These operational characteristics listed above will influence the ConOps to be developed for Special Operations. The solution system has to cope with diverse, autonomous, complex and high tempo operations. Due to its variety in application, Special Operations may also be controlled from tactical, operational or strategic level. At the lower tactical level the external relationships are reduced with linear hierarchies dominating and command provided in a directive manner. At the operational and strategic levels interfacing with external partners and other actors will be more common. These may not be under the same command structure and will require collaboration and influencing with a less direct approach. Also, commanders at the various C2 levels will be empowered to different degrees in different circumstances (MoD 2017).

Special Operations ConOps will require a C2 system that is modular, since the same user may face different challenges on different missions. The C2 system has to support making sense of the situation to implement control and coordination of activities (Rantakokko et al. 2010, Alberts 2011). Managing the elements and interfaces of a modular system may become a complex problem. The proposed MBSE process will assist systems engineers to derive specific requirements for the expected complexity and diverging situations.

4.2 Context Analysis

The system context diagram, as seen in Figure 3, defines the boundary between the system and its environment, showing the entities that it interacts with. This diagram is a high level view (black box) of system of interest. The small blue blocks on the system of interest and the external systems are ports to facilitate flow of information over the interfaces. The arrow heads indicate the flow of information. The external environment mostly consists of systems and stakeholders external to the system. It also provides focus on the actual system of interest that is to be developed. This diagram bounds the solution system to be designed.

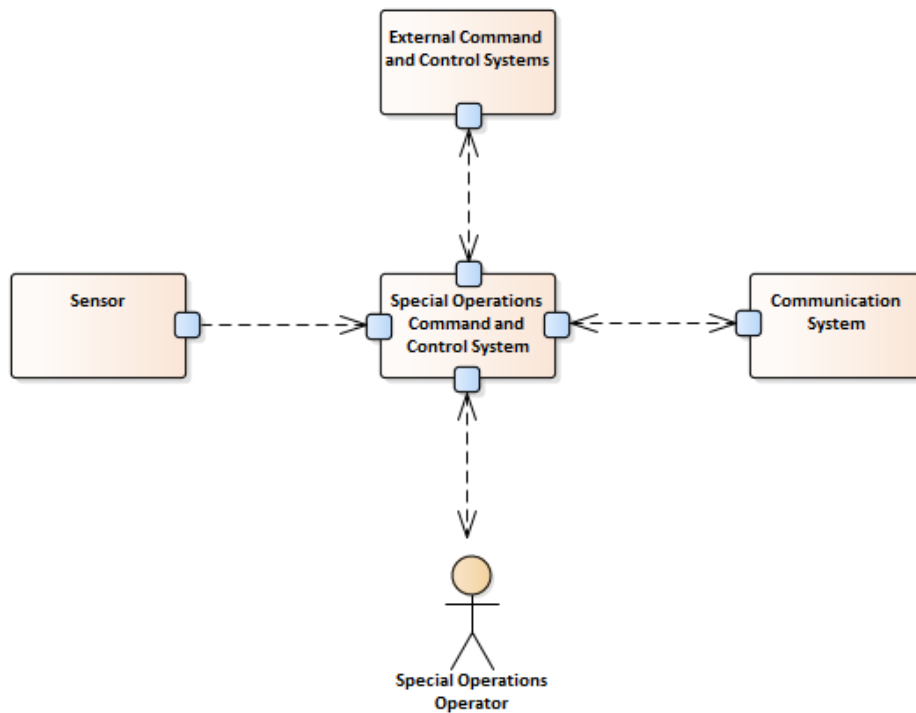


Figure 3: Context Diagram

The C2 system will focus on the interfaces (e.g. displays) and information processing elements that are to be deployed with the operators and their commanders. The communication system is seen as external to the C2 system. Communications is an enabler of the C2 system. It is often controlled by external units and used by others for different purposes. The C2 system will also be required to interface to external sensors deployed with the Special Operations operators. For the Special Operations C2 system to support implementation of Collaborative or Edge C2 approaches, effective communication interfaces and interoperability will be required.

4.3 Use Cases

Use Cases are used to capture and represent the high level system functions and uses of the Special Operations C2 System. The Use Cases also identify different scenarios to derive the functionality required in the system. The Use Cases are structured into packages, as seen in Figure 4 below, to manage a hierarchy of different modelling levels. The arrows between the packages depict the dependencies between them. The sources for the Use Cases include various published doctrine documents and interactions with Subject Matter Experts and other stakeholders. The Use Cases also forms the basis for developing a ConOps.

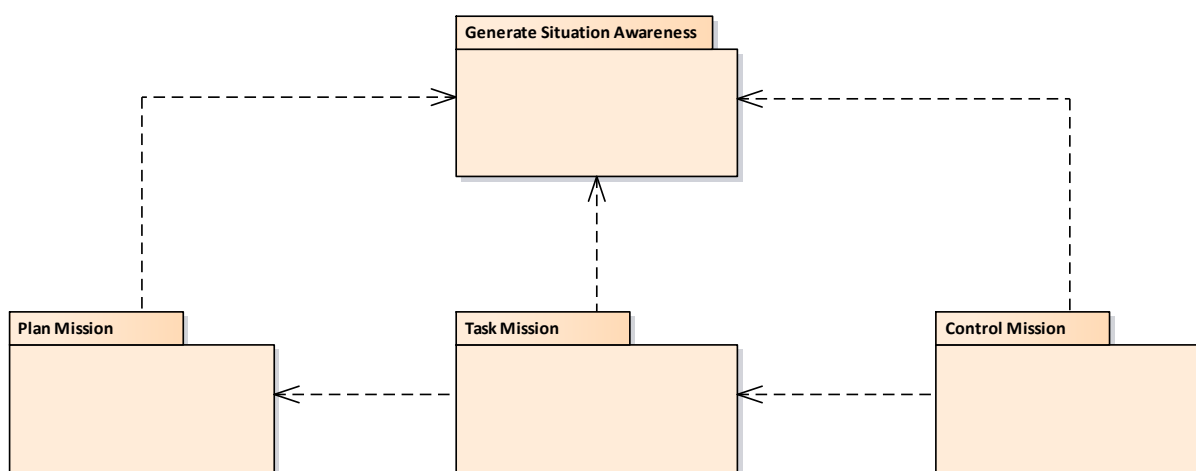


Figure 4: Use Case Packages

An abbreviated Use Case for using the Special Operations C2 system for generating situation awareness information by a tactical level user is given in Figure 5. Situation awareness supports commanders and operators to understand the operational situation in support of decisions. Situation awareness support quick decision making for C2. Situation awareness needs to support changing and adapting of plans at short notice according to new information received or a change in objectives, even though operations are executed with limited information. Information consists of information objects that have been analysed and context added. Information is at a higher abstraction level (processed) than data objects.

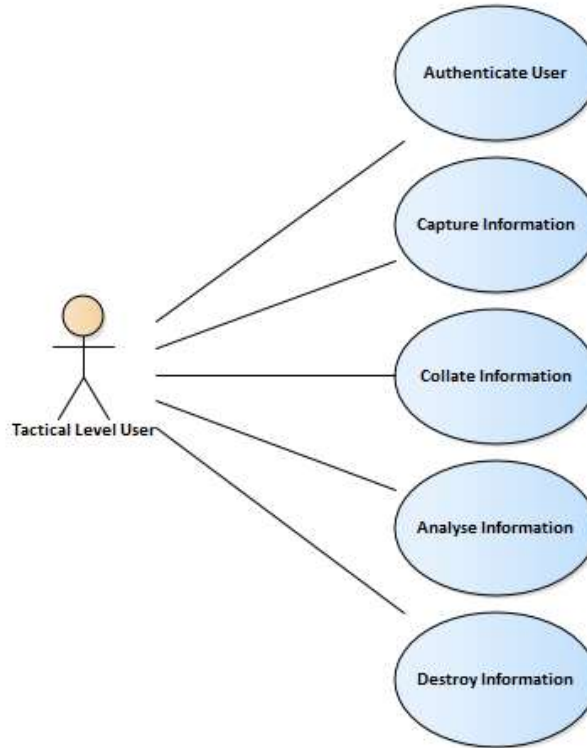


Figure 5: Process Information Use Case

The system needs to implement role based authentication to protect the security and integrity of information. This allows only certain users to access information with defined access (need-to-know) rights. Information about the operational environment that is relevant to the mission is captured in the system by information gathering resources that observe events. The typical information to be captured in the system may include images, video, own position reporting, meteorological information, geographic (topographical) information and social media.

Information collation is also required to receive, process, and displaying current information about friendly and adversary force activities or capabilities. Information is displayed over (overlay) terrain to produce a common operational picture. Data is processed to add value and make it usable as “information”. This involves correct appraisal and understanding of the changing political and military environment. Data from different sensors needs to be fused to provide the most applicable and accurate information for a given use. Finally, Special Operations operators need to be able to destroy all the information in the system in cases of emergency to prevent classified information to fall into the hands of opposing forces.

Similar Use Cases can be generated to address strategic or operational level users and uses of the proposed system. Different Use Cases may also be generated for specific scenarios. The aim is to identify all the uses and actors of the system before initiating the next level of analysis.

4.4 Functional Architecture

The functions to be performed by the C2 are derived from the verbs in the Use Cases. The functional logical architecture of the system is captured in a number of Activity Diagrams to define the relationships between the activities (functions). The hierarchy of selected functional activities is shown in Figure 6.

Another view of the functional architecture is provided in Figure 7 that consists of the same elements (activities) but with different relationships (control and information flows) between them. First, the information in the C2 system is captured through sensing (integrated) of elements in the operational environment and

visual observation of operators, including position and description of an element of interest. Additional information are collected and collated with GIS data to establish the links between various bits of information.

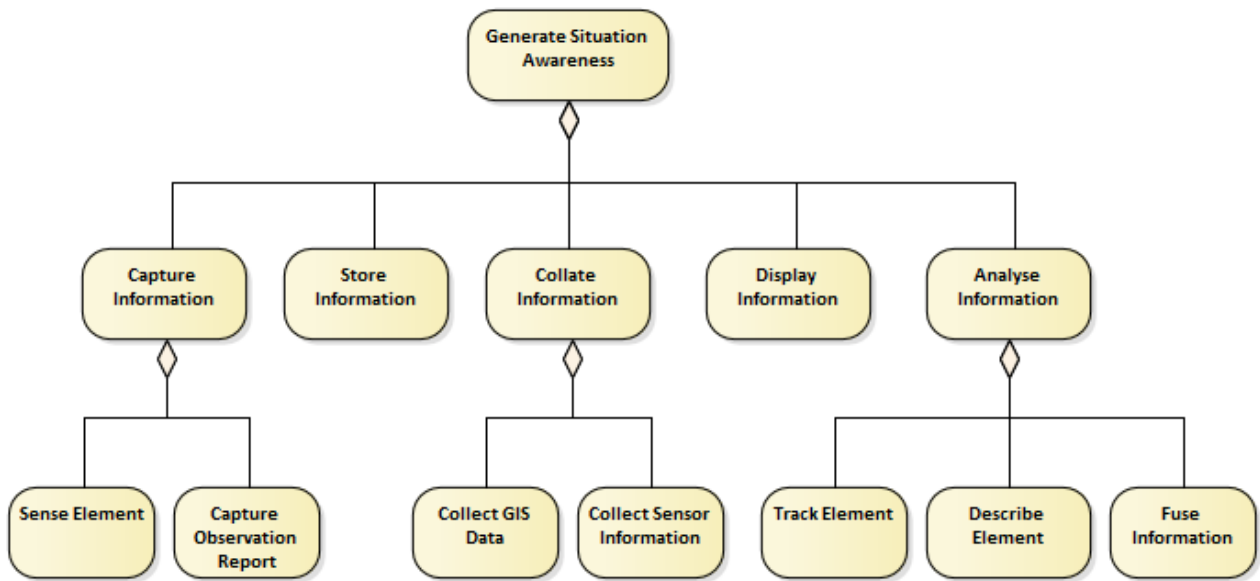


Figure 6: Functional Architecture Hierarchy

The information is stored for display and analysis. Selected information is to be displayed for the operators to provide the commander with a complete picture of the battlespace and the ability to order, respond, and coordinate actions. Another key aspect is to analysis information to improve the value and usability of the captured information. Again, ports are used to define the interfaces between the activities.

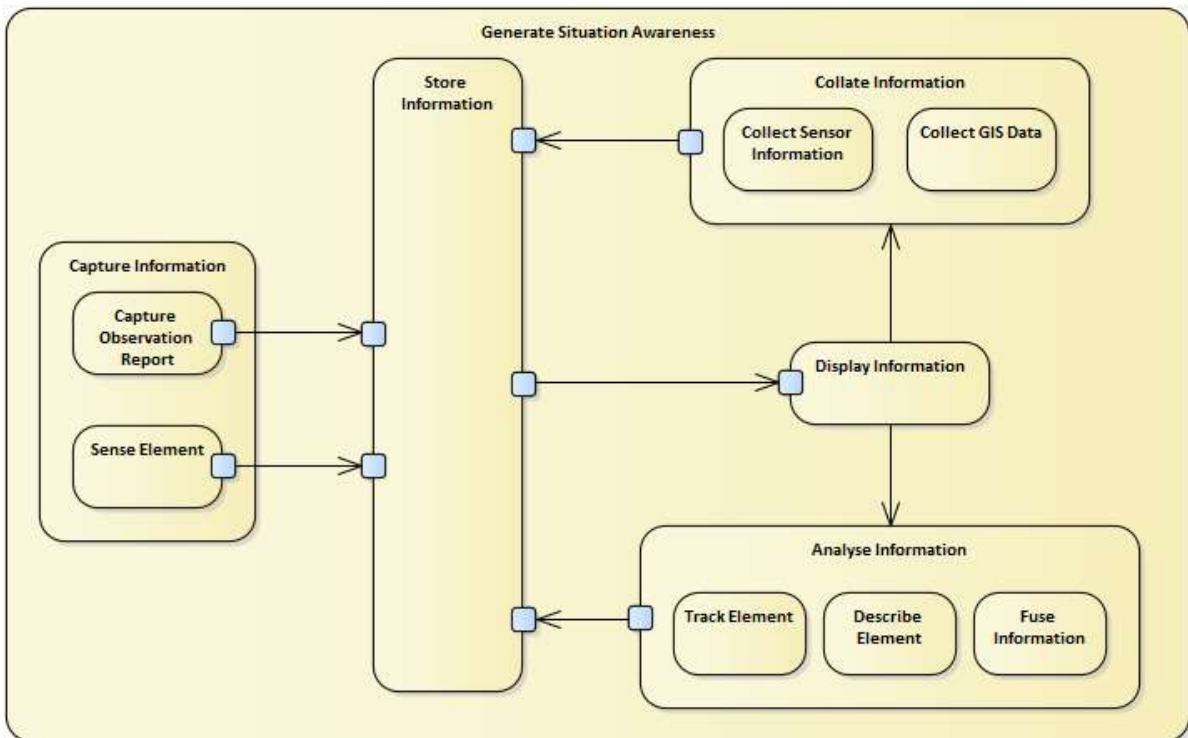


Figure 7: Functional Architecture

4.5 Logical Architecture

The C2 system logical architecture, as seen in Figure 8, consists of system elements derived from nouns in the Use Cases to provide the structure and tools (system elements). The system is decomposed and partitioned into logical elements that interact to satisfy the system requirements using blocks in a block.

Figure 9 provides interfaces between the logical elements of the C2 System from Figure 8. The diagram consists of the same elements (blocks) but with different relationships (interface) between them. The ports on the external boundary of the system of interest correspond to the ports defined in the context diagram. The ports define the interfaces for flow of items and information with the external systems and Situation Management System operators.

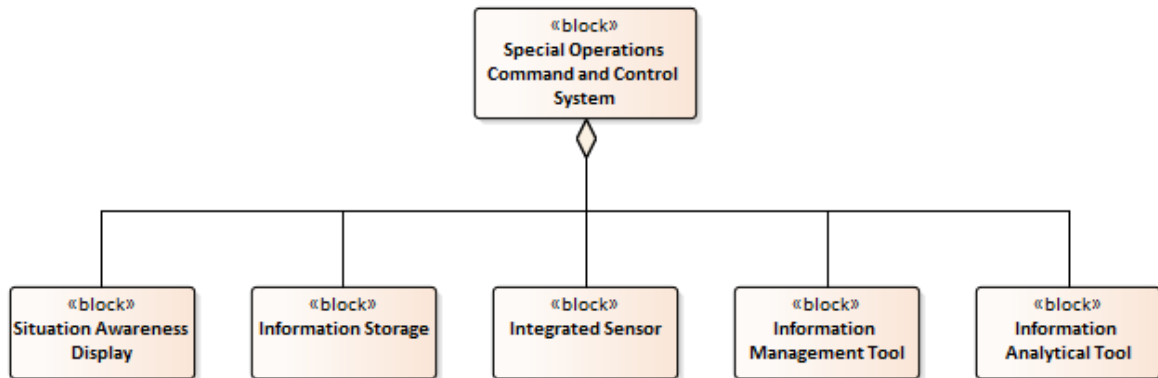


Figure 8: System Architecture Structure

The logical system elements defined in the logical architecture perform functions, as seen in Figure 10. Here the activities from Figure 6 and Figure 7 are assigned to the block elements from Figure 8 and Figure 9. The information captured in Special Operations C2 system is presented to operators on a display. This display enables operators to view the information and interact with it. The information displayed could include the common operation battlefield picture consisting of a map with own, friendly forces, threat forces and target positions. Additional information, that may be filtered as required, includes mission status, health status of personnel, electronic attack warning and video or photo imagery in real time. This information may be from collocated sensors, which may be integrated and into the C2 system.

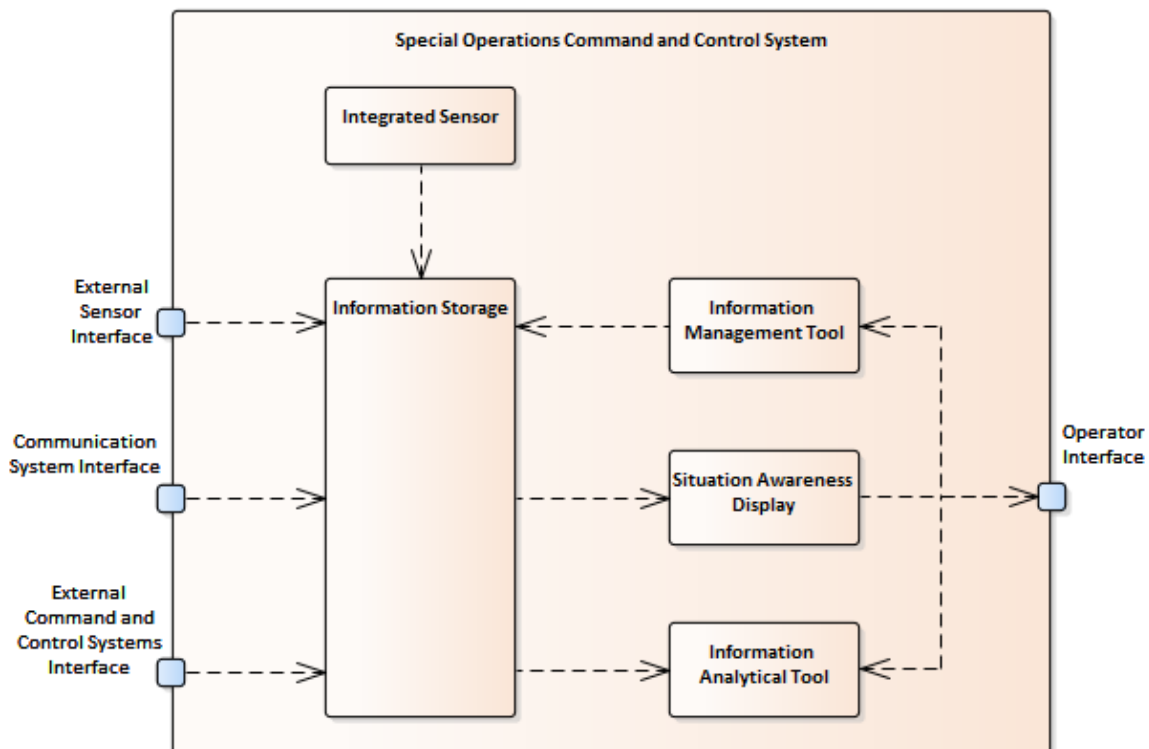


Figure 9: System Architecture with Interfaces

An Information Analytical Tool or application supports analysis and processing of captured information in support of the intelligence process (cycle) as well as developing situation awareness. The tool accesses the stored information through the situation awareness display to view, play back and process the information in the database. All information in the C2 systems is stored in a centralised database. The database requires a data meta-model and ontology suitable to serve all the Special Operations missions.

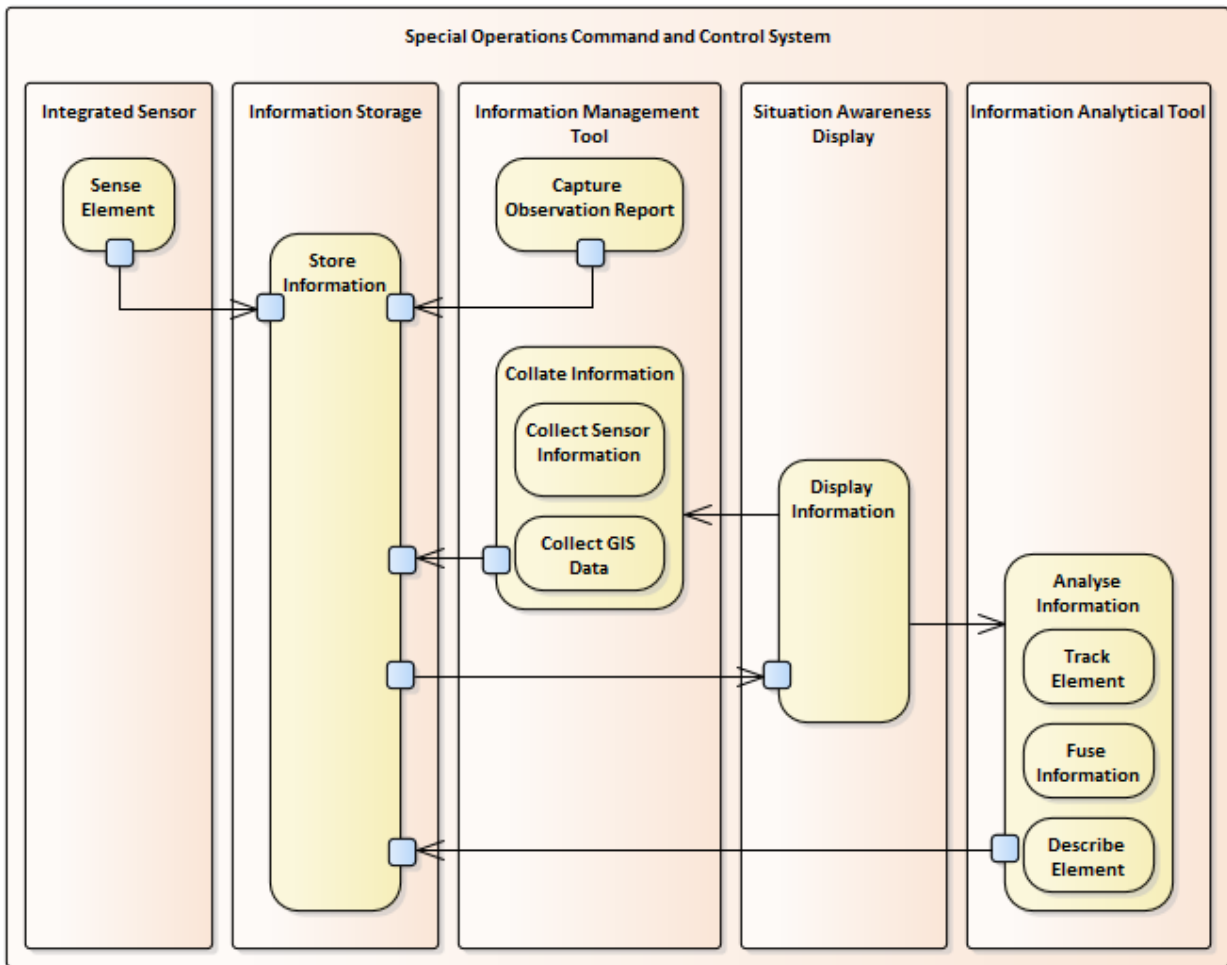


Figure 10: Logical Architecture in Swimlanes

4.6 Solution Implementation

The final step in the modelling process is to convert the logical and functional architectures into specific solution options for analysis and trade-off. The selected solution description will form the basis for a system requirement specification of the required C2 system to be procured. As seen in Figure 11, this step describes the relationships among the physical system elements or nodes that define the distribution of resources.

The blocks identified in the logical architecture are instantiated through properties of the block diagrams. The SysML blocks (UML classes) are instantiated as properties (UML objects) to model the different solutions. A block may be instantiated through multiple properties, as seen with the integrated sensor and the situation awareness display logical elements. The information analytical tool is not instantiated in the tactically deployed personal integrated C2 system as the operator may not have time for information analysis. This approach enables systems engineers to implement modularity in their system design.

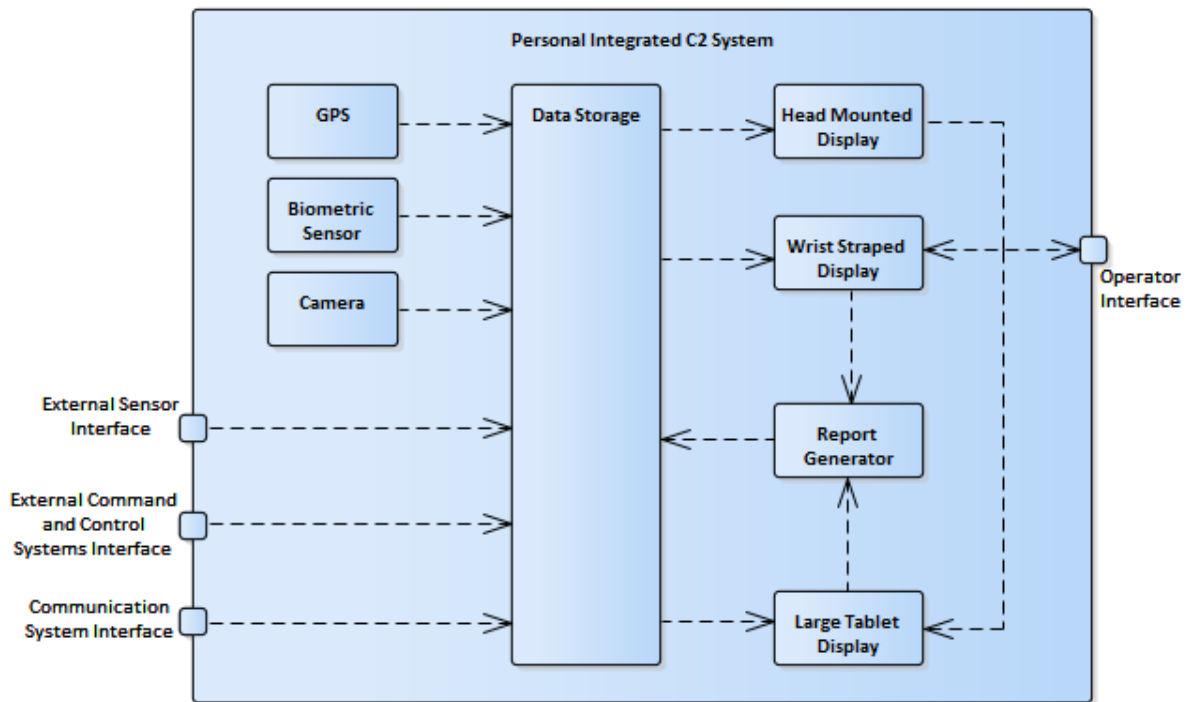


Figure 11: C2 System Solution Implementation

4.7 Traceability

One of the advantages of implementing MBSE with a software tool is the ability to maintain traceability between the various elements captured in the tool, as seen in Figure 12. This traceability enables the systems engineers to implement change management. A change in a Use Case can be traced to the implication in the system solution. Any of these blocks can be traced to user, system or implementation requirements.

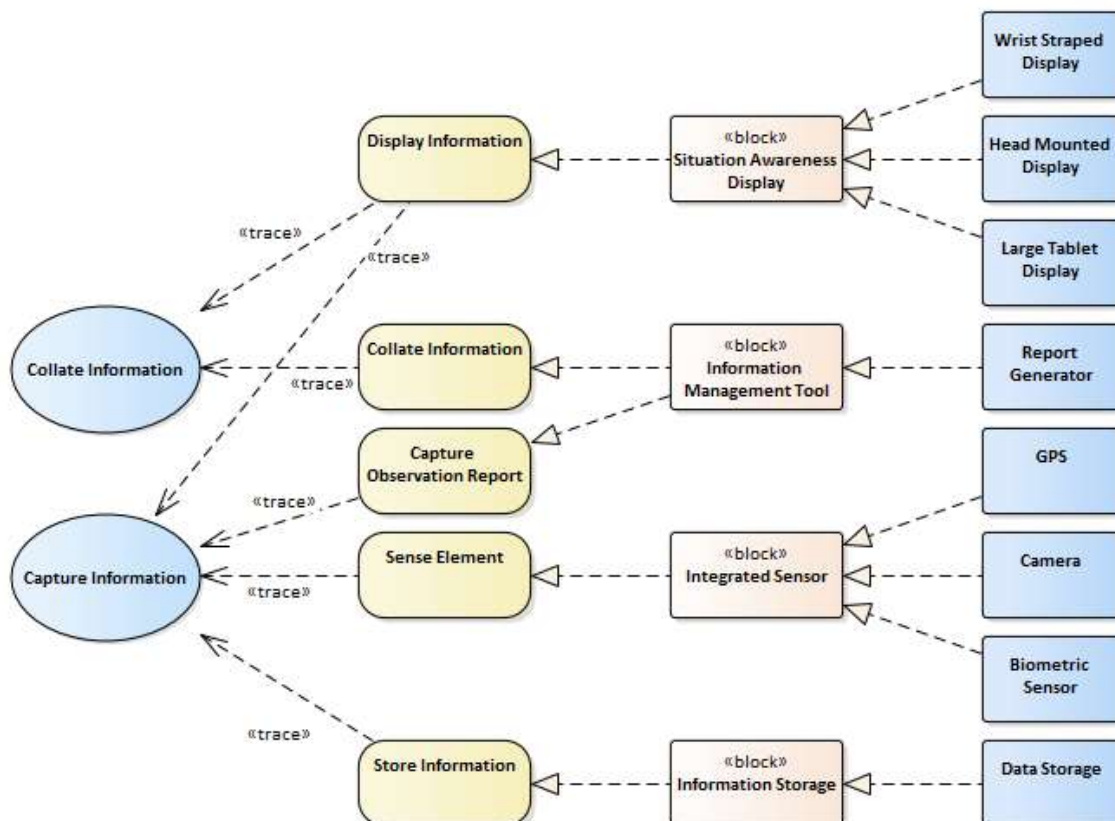


Figure 12: Element Traceability

4.8 Discussion

This section has demonstrated the MBSE based approach for development of C2 systems to be utilised in a complex environment. The process is architecture framework independent and can link into capability based systems modelling. The power of modelling is providing a visual representation of all the system elements with their relationships and interactions. A picture is worth a 1000 words. Models assist stakeholders to understand their requirements and system solution implications. Diagrams of the models still remain easier for use during reviews as complex relationships are easier to visualise and manage.

At every step of the process it is possible to define and add requirement for the system. Each requirement can be associated or traced to a specific element. Even many-to-many relationships are possible between the requirements and system elements. If a list of requirements (e.g. user requirements) exist, it can be imported into the tool and linked to the various model elements.

Most MBSE tools enable modellers to add text descriptions to the model elements and the relationships. These tools also have automated applications to export diagrams with descriptions of the elements exported to text-based reports for formal reviews and acceptance by stakeholders. Reports may be in the form of a requirement specification, system description or ConOps that contain only the specific information required from the model.

Transportability of the model to the organisation developing and implementing the system is also important. Having access to the model elements (Use Cases, activities and logical blocks) and the traceability to requirements will assist in trade-off and design decisions.

5 Conclusion

This paper has proposed a MBSE process and demonstrated it by develop a C2 system for Special Operations. As Systems Engineering for the development of complex and sociotechnical systems is becoming increasingly difficult, MBSE provides an approach to design and implement effective and modular systems. Modelling that captures and represents the mental models of the systems' stakeholders during the requirements analysis and concept development stages helps understanding this complexity.

Using a model based (data-centric) software tool to model the system helps to manage the complexity and traceability between requirements and the proposed solution concept. Maintaining traceability between various system diagrams and views ensures consistency of the defined model elements and their relationships.

To demonstrate the MBSE method, a sociotechnical system in the form of C2 for Special Operations was modelled at a generic level. These views provide the structure for an in-depth analysis and development of an actual C2 system.

6 References

- Alberts, D.S., 2011. The agility advantage: a survival guide for complex enterprises and endeavors. Office of the Assistant Secretary of Defense (Networks and Information Integration) Washington DC.
- Alberts, D.S. and Hayes, R.E., 2007. Planning: complex endeavors. Assistant Secretary of Defense (C3I/Command Control Research Program) Washington DC.
- Bahill, A.T. and Szidarovszky, F., 2009. Comparison of dynamic system modeling methods. Systems Engineering, 12(3), pp.183-200.
- Brown, H.S., 1996. The Command and Control of Special Operations Forces. NAVAL POSTGRADUATE SCHOOL MONTEREY CA.
- Buede, D.M. 2000. Engineering Design of Systems - Models and Methods. John Wiley & Sons.
- De Weck, O.L., Roos, D, and Magee, C.L., 2011. Engineering systems: Meeting human needs in a complex technological world. MIT Press.
- Department of Defence, 2007. Joint Special Operations Task Force Operations, Joint Publication 3-05.1.
- Eaton, D., O'Mahony, A., Szayna, T.S. and Welser IV, W., 2017. Supporting Persistent and Networked Special Operations Forces (SOF) Operations. RAND Corporation.
- Estefan, J. 2007. Survey of Model-Based Systems Engineering (MBSE) Methodologies. Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA.

- Friedenthal, S., Moore, A. and Steiner, R., 2012. Integrating SysML into a systems development environment a practical guide to SysML. Elsevier, pp.523-556.
- Gau Pagnanelli, C.A., Sheeley, B.J. and Carson, R.S., 2012, July. Model-Based Systems Engineering in an Integrated Environment. In 22nd Annual INCOSE International Symposium. Rome, Italy.
- Harrison, J.R., Lin, Z., Carroll, G.R., & Carley, K.M. 2007. Simulation Modeling in Organizational and Management Research. *Academy of Management Review*, vol. 32, no. 4, pp. 1229-1245.
- Hause, M. C., 2014. SOS for SoS: A new paradigm for system of systems modeling. In *Aerospace Conference, 2014 IEEE* (pp. 1-12). IEEE.
- Hitchins, D. K. 2008. *Systems Engineering: A 21st Century Systems Methodology*. Wiley. com.
- Holt, J. and Perry, S., 2008. *SysML for systems engineering (Vol. 7)*. IET.
- Johansen, I., 2015. Special Operations Forces—a Weapon of Choice for Future Operations?. *International Military Operations in the 21st Century: Global Trends and the Future of Intervention*, pp.116-36.
- Karban, R., Zamparelli, M., Bauvir, B. and Chiozzi, G., 2012. Three years of MBSE for a large scientific programme: Report from the Trenches of Telescope Modeling. In *Proceeding 22nd Annual INCOSE International Symposium*.
- Kiras, J.D., 2015. A Theory of Special Operations:“These Ideas Are Dangerous”. *Special Operations Journal*, 1(2), pp.75-88.
- Kossiakoff, A., Sweet, W.N., Seymour, S. and Biemer, S.M., 2011. *Systems engineering principles and practice (Vol. 83)*. John Wiley & Sons.
- Madden, D., Hoffmann, D., Johnson, M., Krawchuk, F.T., Nardulli, B.R., Peters, J.E., Robinson, L. and Doll, A., 2016. *Toward Operational Art in Special Warfare*. RAND Corporation Arroyo Center Santa Monica United States.
- Maria, A., 1997. Introduction to Modeling and Simulation. In *Proceedings of the 29th conference on Winter simulation*. IEEE Computer Society, 7-13.
- Meyer, B., 2014, October. Generic Model-Based Systems Engineering Methodology. In *INCOSE International Symposium (Vol. 24, No. s1, pp. 207-222)*.
- Ministry of Defence 2017. *Future of Command and Control: Command and control in the contemporary and future operating environments*, Joint Concept Note 2/17
- O'Hara, P.M., 2007. *An analysis of Special Operations Command-South's distributive command and control concept (Doctoral dissertation, Monterey California. Naval Postgraduate School)*.
- Oliver D.W., Kelliher T.P., & Keegan Jr. J.G. 1997, *Engineering Complex Systems with Models and Objects*. McGraw-Hill.
- Oosthuizen, R. and Pretorius, L., 2013. Assessing command and control system vulnerabilities in underdeveloped, degraded and denied operational environments.
- Oosthuizen, R. and Pretorius, L., 2014. Modelling of command and control agility.
- Oosthuizen, R. and Pretorius, L., 2015, November. System dynamics modelling of situation awareness. In *Military Communications and Information Systems Conference (MilCIS), 2015* (pp. 1-6). IEEE.
- Oosthuizen, R. and Venter, C., 2016. Implementing UPDM to develop command and control systems.
- Ott, P.A., 2002. *Unconventional warfare in the contemporary operational environment: Transforming special forces*. Army Command And General Staff Coll Fort Leavenworth Ks School Of Advanced Military Studies.
- Piaszczyk, C., 2011. Model based systems engineering with Department of Defense Architectural Framework. *Systems Engineering*, 14(3), pp.305-326.
- Ramos, A.L., Ferreira, J.V., & Barceló, J. 2012. Model-Based Systems Engineering: An Emerging Approach for Modern Systems. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 42, no. 1, pp. 101-111.
- Rantakokko, J., Händel, P., Fredholm, M. and Marsten-Eklöf, F., 2010, September. User requirements for localization and tracking technology: A survey of mission-specific needs and constraints. In *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on* (pp. 1-9). IEEE.
- Ryan, J., Sarkani, S., & Mazzuchi, T. (2014). Leveraging variability modeling techniques for architecture trade studies and analysis. *Systems Engineering*, 17(1), 10-25.

- Stanton, N.A., Baber, C., & Harris, D. 2012. Modelling Command and Control: Event Analysis of Systemic Teamwork. Ashgate Publishing.
- Stensson, P. 2010. Thoughts about the Consequences of Inappropriate Application of Systems Engineering. In Proceedings of the 7th European Systems Engineering Conference, Stockholm, INCOSE.
- Sterman, J.D, 1994. Learning in and about Complex Systems. System Dynamics Review, vol. 10, no. 2-3. pp. 291-330.
- Tschirner, C., Dumitrescu, R., Bansmann, M. and Gausemeier, J., 2015, April. Tailoring Model-Based Systems Engineering concepts for industrial application. In Systems Conference (SysCon), 2015 9th Annual IEEE International (pp. 69-76). IEEE.
- Votel, J.L., 2016. Special Operations Forces Operating Concept. A Whitepaper to Guide Future Special Operations Force Development, Directorate of Force Management and Development, Concept Development and Integration Office.
- Walden, D.D., Roedler, G.J., Forsberg, K., Hamelin, R.D. and Shortell, T.M. eds., 2015. Systems engineering handbook: A guide for system life cycle processes and activities.
- Young, L., 2017. Future Joint Command and Control Concept.
- Holt, J. and Perry, S. 2017. 'Don't Panic! The Absolute Beginner's Guide to Model-Based Systems Engineering'. INCOSE UK Ltd.