ENGINEERS AUSTRALIA
SETE
SYSTEMS ENGINEERING TEST AND EVALUATION CONFERENCE 2018
ICC SYDNEY
30 APRIL - 2 MAY 2018
www.sete2018.com.au

ISBN for SETE2018: 978-1-925627-15-2

A framework to support model-based management of Capability Programs

Duane Jusaitis
Shoal Engineering Pty Ltd
duane.jusaitis@shoalgroup.com

Stephen Cook
Shoal Engineering Pty Ltd
stephen.cook@shoalgroup.com

ABSTRACT

In 2016, following the First Principles Review, Defence introduced the Capability Life Cycle (CLC) and the Program Management construct to develop and manage Defence Capabilities for the Joint Force of the Future. Conventional wisdom holds that to realise the potential of the Joint Force and the capabilities that support it, careful analysis of the capabilities and their integration and interoperability is required. Programs are a step towards more formally managing the complex capabilities and their integration and interoperability demands and are in addition to the Project/Product level management previously undertaken.

Programs have been formed by combining existing Projects and Products under a single banner. Program Managers are formulating an understanding of how to coordinate the various Project and sustainment activities. The Joint Capability Narrative, the Joint Capability Needs Statement, the Program Strategy, and the Program Integrating Operational Concept are elements that provide top-down driven understanding of what the capability must achieve, where it needs to operate, and with whom it needs to operate. The central theme of all these elements can be portrayed in a set of scenarios covering the operation and support of the capability.

Demands on the Defence budget put pressure on Programs to provide robust Capability-based business cases when seeking approval for new acquisition or upgrade Projects. The business case should include new Project justification, e.g. the capability is not effective in the new threat environment, and why the proposed acquisition or upgrade best meets the overall capability needs. This justification can be derived from analysis of the current and proposed capabilities in Program-level scenarios against an agreed set of Program-level Measures of Effectiveness.

This paper discusses the development of a modelling framework to support simulation and analysis of Program-level capability scenarios in order to produce low-fidelity effectiveness predictions in support of the ongoing management of Program Capability effectiveness over a lifetime covering many decades. Further, it explores a method for the integration of constituent Project models into the overarching Program-level model to facilitate more detailed analysis as Project design and measurement information evolves through the CLC.

INTRODUCTION

In 2016, following the First Principles Review (Defence 2015), Defence introduced an updated Capability Life Cycle (CLC) and the Program Management construct to develop and manage Defence Capabilities for the Joint Force. Conventional wisdom holds that to realise the potential of the Joint Force and the capabilities that support it, careful analysis of the capabilities and their integration and interoperability is required. Programs are a step towards more formally managing and assessing the complex capabilities and their integration and interoperability demands and are in addition to the Project/Product level management previously undertaken. Defence has created around 40 Programs each of which is "a group of related Projects, Products, and activities that are managed in a coordinated way to optimize the capability outcome within allocated resources" (Defence 2017b).

Cook and Unewisse (2017, 2018) identified these Programs as enduring Systems of Systems (SoS) and concluded that the management and systems engineering (SE) of these Programs will require a SoS Engineering (SoSE) approach. The recommended approach has been tailored to the Australian environment to support the CLC approach and Smart Buyer Framework demands in a manner that does not require large numbers of resources to deliver. The approach proposes a Program team to provide oversight and SoS evolution guidance to constituent Projects and Products at key decision points in their life cycle.

Program Managers are formulating an understanding of how to coordinate the various Projects and Sustainment activities. The Joint Capability Narrative, the Joint Capability Needs Statement, the Program Strategy, and the Program Integrating Operational Concept (PIOC) are artefacts that provide top-down driven understanding of what the capability must achieve, where it needs to operate, and with whom it needs to operate. The central theme of all these elements can be portrayed in a set of Program-level scenarios covering the operation and support of the capability.

Hallett, Psalios, Jusaitis, and Cook (2018) describe and propose a model-based approach to Program-level scenario analysis, they claim that "the key to implementing this approach is to adopt Model-Based Systems Engineering (MBSE) utilising a tool that allows both descriptive and analytic models to be developed and links between separate Program and Project models to be made".

This paper explores the practical underpinnings of the Hallett et al. (2018) methodology and discusses the development of a model framework to support simulation and analysis of Program capability scenarios in order to produce low fidelity effectiveness predictions in support of the ongoing management of Capability effectiveness over a lifetime covering many decades. To facilitate more detailed analysis based on evolving Project design and measurement information, it also explores the integration of constituent Project models with the overarching Program-level model. The paper then ends by proposing a metamodel to support such model integration and simulation.

FRAMEWORK GOALS

Defence has for many years been undertaking Capability Design activities on Projects utilising the Systems Engineering approach defined in the Capability Definition Documents Guide (Defence 2017a). This approach is implemented by Shoal Engineering in a model-based environment, utilising the Whole of System Analytical Framework (WSAF) developed by the Defence Science and Technology Group (DST Group) (Robinson et al. 2010). The Capability Design and associated model achieved through this methodology are descriptive in nature and form the basis upon which the Operational Concept Document (OCD), Function and Performance Specification (FPS), and Test and Evaluation Master Plan (TEMP) documents for Projects are generated and maintained.

Programs are starting to adopt a model-based approach to manage Program-level Capability and its integration and interoperability aspects. The pedigree of the WSAF framework makes it a sensible place to start when considering how to approach effective model-based management of Capability Programs. Naturally, the emphasis may be different between Program and Project levels, but much can be learnt from Project-level MBSE experiences and successful techniques can be evolved to suit Program needs.

The Program-level model development undertaken by Shoal has been guided by the approach laid out in the WSAF framework that has been tailored through the addition of Program-level terminology, updates to better capture required Program-level information, and artefacts such as the PIOC (French & Heard 2018). This evolved methodology was adopted for the Program-level model development undertaken as part of this study.

Defence Project capability design models developed to-date have been largely descriptive in nature. This study hypothesised that capability feasibility and effectiveness, in the current and predicted future environment, could be predicted through the simulation of Program-level scenarios and the analysis of scenario variants, and used to support capability management decisions. Further, that this Program design and simulation could be better supported and enhanced by leveraging the information and models available at the Project / Product level through integration with the Program-level model.

The above hypothesis formed the basis for the objective of this study; to work towards a metamodel that could be used as the semantic basis to support Program capability design modelling and information management across the CLC, scenario simulation, and integration of Project design models with the Program model.

MODEL DESIGN AND DEVELOPMENT

To explore the information required for early-stage capability simulation, and for Project and Program model integration, an executable demonstration model was developed in the Vitech GENESYS MBSE tool.

The approach taken was to:

1. Select a suitable system and scenario, and identify scenario variants for analysis;
2. Determine a set of suitable Critical Operational Issues (COIs) and derive Measures of Effectiveness (MOEs) that could provide a means of assessment for scenario variants;
3. Build the demonstration model based on the chosen scenario and develop it with the data and constructs required to produce a useful simulation that accommodates chosen variants;
4. Specify a set of key performance parameters that are germane to capability performance and collectable from constituent Project models;
5. Develop a way for the demonstration model to record and report simulation results for the comparison and analyses required to inform an assessment of scenario effectiveness against the defined MOEs.

To gain useful insights, the demonstration model needed to be representative of the systems that are commonly encountered in Defence. A suitable campaign that closely met these requirements was the attempted invasion of Britain by Germany, specifically the activities of the two respective air forces, the Royal Air Force (RAF) and Luftwaffe, in World War 2, now known as the Battle of Britain. The RAF element of this campaign was centred around, and provides a suitable example of, a SoS commonly known as the "Dowding System" after its Chief Systems Architect - Air Chief Marshall Sir Hugh Dowding (Imperial War Museums n.d.).

The Dowding System has been classified as the first “integrated air defence system” (Omps 2014) and is considered to be an early example of an “information system” (Checkland & Holwell 1998). It is a “rich source of general lessons relevant to the development, implementation and use of any information system” (Checkland & Holwell 1998), and there is a good deal of information and analysis of both the system and the scenario in the public domain (Bungay 2000, Holland 2010). The system, illustrated in Figure 1, is a perfect example of a Directed SoS (SEBoK 2017) clearly designed and managed to fulfil a specific purpose with the constituent systems subordinated to the SoS; although able to operate independently, the constituent systems normal operational mode in the defence of the UK was subordinated and controlled through the Air Defence Network.

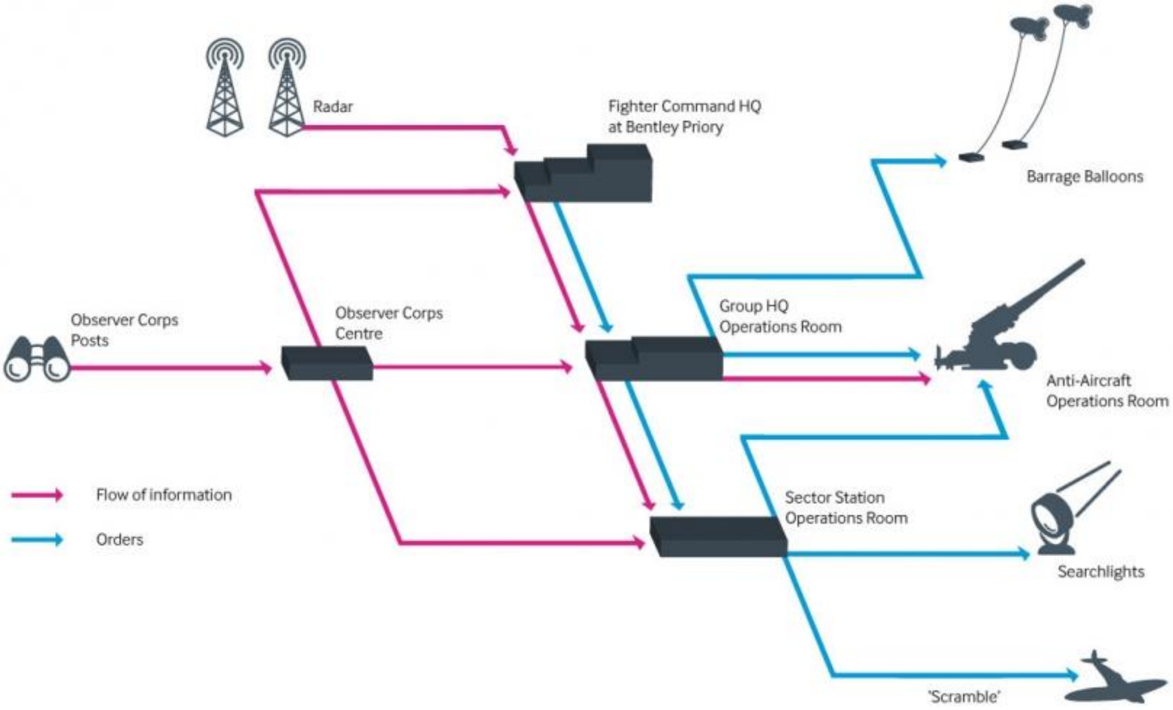

Figure 1. The Dowding System (Imperial War Museums n.d.).

The system comprised Fighter Command HQ with four group HQs, multiple sector HQ stations per group, and multiple airfields and fighter squadrons per station. All HQ’s at each level had an operations room with a large map of the UK for monitoring the tracks of incoming raids and the intercepting fighter squadrons. Fighter Command HQ at the top had a Filter Room in which all incoming sensor information was combined and analysed to form a clear operational picture for distribution to Groups and Sectors. Each group was essentially an instance of the Dowding System including a range of sub-systems and people.

The scenario selected for simulation comprised a single bombing raid, consisting of 100 bombers from German-occupied France, targeting London. This scenario is sufficiently complex to exercise all Dowding System elements while being simple enough to model within the modest study resources.

The dynamic targeting process of ADDP 3.14 (ADF Warfare Centre 2009) was chosen as a starting point for model development because it is well matched to the scenario. The process’s Find, Fix, Track, Target, Engage, Assess (F2T2EA) pattern was used to develop the top-level functional flow. The US DoD Joint Targeting process (Defense 2013) was then used as a starting point for functional decomposition.

Based on historical information gathered on the Dowding System and the Battle of Britain, the generic flows were then tailored to the scenario, cutting activities that were not relevant and adding activities that were specific to the scenario. The GENESYS simulation feature was first used to ensure logically correct functional flow, further iteration developed the model to better reflect the historical details of the system recorded in the literature (Checkland & Holwell 1998, Dowding 1941, Hamilton 1969, Holland 2010, Imperial War Museums n.d., Johnson 2006, Omps 2014, Royal Air Force 2017, Trueman 2015, Williams 2005). Figure 2 shows the top-level functional flow.

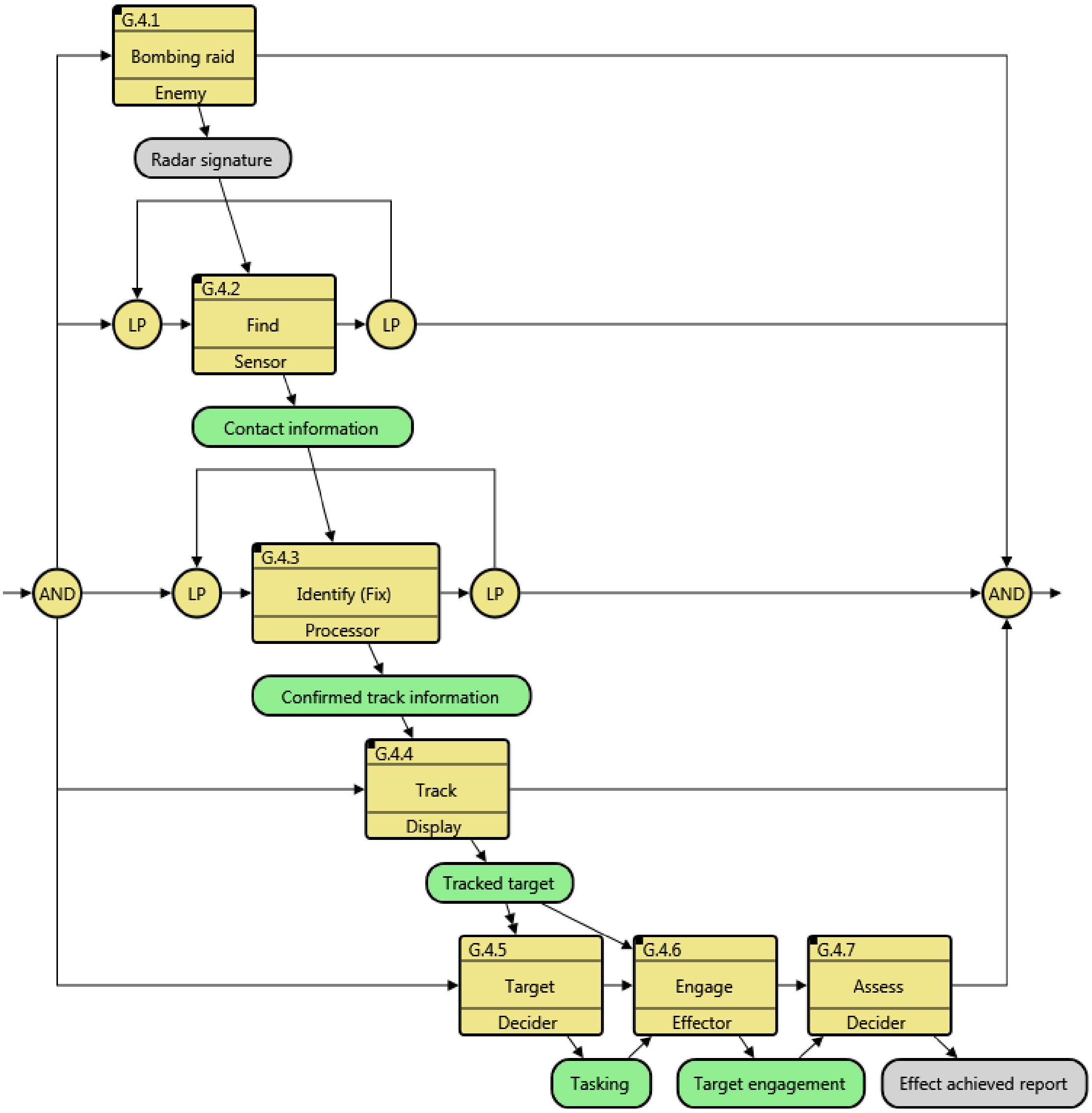

Figure 2. High-level dynamic targeting functional flow (green items represent triggers).

A scenario baseline was achieved by tuning the critical activity path and configuring model data inputs so that the simulation would closely represent the historical record. **Error! Reference source not found.** s hows the selected parameters, baseline values, and example variant values that were modelled for each parameter.

Table 1. Simulation parameter variants.

Parameter type	Performer	Baseline	Variant 1	Variant 2
Speed	Enemy raid: Formation speed	250 mph	500 mph	
Range	Sensor: Radar range	80 miles	100 miles	150 miles
Latency (delay)	Processor & Decider: sense to scramble time	360s	300s	600s
Latency (delay)	Effector (Fighter): scramble to intercept	840s	720s	960s
Variability	Effector (Fighter): attack pass time	30 – 180s		
Quantity	# Fighters	48	60	36
Duration	Combat time (limited to fighter fuel reserve)	1200	60	
Probability	Hit on bomber	0.25		
Probability	Kill bomber	0.15		
Probability	Hit on fighter	0.1		
Probability	No hit	0.5		

SCENARIO OPTION SIMULATION

To understand the effectiveness of a capability in its operational environment, and to align the feasibility study with SE practice, Critical Operational Issues (COIs) and Measures of Effectiveness (MOEs) were defined to determine how well the Dowding System worked; the impacts of changing the constituent elements, tactics, and threat/bomber performance; and the effects of damage to, or performance degradation of, constituent elements.

The following COIs and related MOEs were defined:

COI 1 – Does the Air Defence Network reduce the efficacy of enemy attacks on High Value or populated areas?

MOE 1 – Percentage reduction in bomber raid efficacy during a single multi-bomber raid.

COI 2 – Can the Air Defence Network effectiveness be sustained over an extended period?

MOE 2 – Length of time the Air Defence Network can maintain operational assets on or above critical levels.

COI 3 – Can the Air Defence Network be readily adapted to counter changes in enemy weapons and tactics?

MOE 3 – Length of time taken to adapt the Air Defence Network to changes in enemy tactics.

MOE 4 – Length of time taken to adapt the Air Defence network to changes in enemy weapons.

Four sections of the capability system were selected for sensitivity analysis to compare the effect of system variants against the baseline and thus understand how each variant performed against the first MOE. System variants were chosen that had potential to reduce civilian casualties and provide the fighters with a combat advantage, these were as follows:

1. **Improvement in response time** demonstrated through upgrade options for:
    - the sensor (extended radar range resulting in earlier warning);
    - the decider (sense-to-scramble response time); and
    - the effector / fighter (scramble-to-intercept response time).
2. **Improvement of fire power** demonstrated through:
    - increase in available effectors (number of fighters).

The functional flow contained some elements of probability surrounding the air battle between the fighters and the bombers. Thus, each system variant was simulated several times to provide an expected value with sufficiently low standard deviation that was used to calculate the percentage difference in performance from the baseline. For each variant, data points were loaded to configure the model using a spreadsheet update before each run of simulations.

To assess MOE 3 and 4, the same technique was applied to evaluate the impact of changes to the nature of the enemy threat by increasing the speed of the enemy formation to emulate the V-1 flying bomb. This allowed evaluation of how well the extant air defence capability responded to an evolved threat and clearly highlighted the need for capability upgrades. Varying the air defence capability in turn allowed comparison and identification of required changes and upgrades to best counter the more powerful threat and provided a foundation on which to base estimates for the length of time required to adapt to the changed enemy threat.

To assess how each capability variant affected attrition, measurements of the number of fighters and bombers remaining in combat were captured at two points in the scenario simulation: when the bombers reached their target, and when the combat ended due to the fighters returning to base to refuel.

Analysing the simulation results provided insight into the most effective variant. Table 2 shows results for MOE 1. It was apparent that sensor upgrades to extend the radar range provided the greatest improvement in fighter lead time, allowing the fighters a significantly larger portion of their full combat window before the bombers reached their target. In one case the entire combat window was exhausted before the bombers reached their target. Increasing the number of fighters provided the next-best option. Due to the higher speed of the enemy bomber variant, the fighters would only be able to attempt a single attack pass, and so the more effective variant was the increased number of fighters.

Table 2. Simulation results for selected Capability variants – MOE 1.

Capability Variation	MOE 1 score Baseline bomber speed	MOE 1 score Variant 1 Bomber speed
Baseline	32%	9%
Sensor range variant 1	49%	7%
Sensor range variant 2	58%	8%
Fighters variant 1	25%	5%
Fighters variant 2	45%	10%

METAMODEL DESIGN

Two approaches were considered for integrating constituent Project models with the Program model.

The first approach, *direct model integration*, involved the simulation of constituent Project scenarios as part of a Program scenario simulation. GENESYS provides a method of direct model integration through cross-project links, whereby a Program scenario model can directly link its entities to the entities in constituent Project models using any standard relationship. This approach to model interfacing seems to work best where a Program model treats constituent Project models as “black-boxes”, that is, the constituent model is self-contained and meets a neatly scoped requirement to provide a specified capability to the Program. However, this approach produces a single, large simulation, which requires all constituent models to be accessible by the Program model at run time. A modelling architecture of this type would also see simulation runtime increase with the size of the Program model and become unwieldy for larger models.

Considering the downsides of direct model integration another approach was proposed, *Measure Aggregation.* Measure Aggregation proposes that Project models reference Program-defined measures and are developed to report against them. Results from independently run Project simulations are recorded against these measures as outputs. Project measure results are then loaded into the Program-level model and used to inform Program scenario simulation. This results in a smaller Program-level model and provides a faster Program-level scenario simulation. Additionally, it obviates the requirement for run-time access to the constituent Project models.

Based on the Project measure aggregation approach, a preliminary metamodel structure was proposed as summarised in Figure 3.

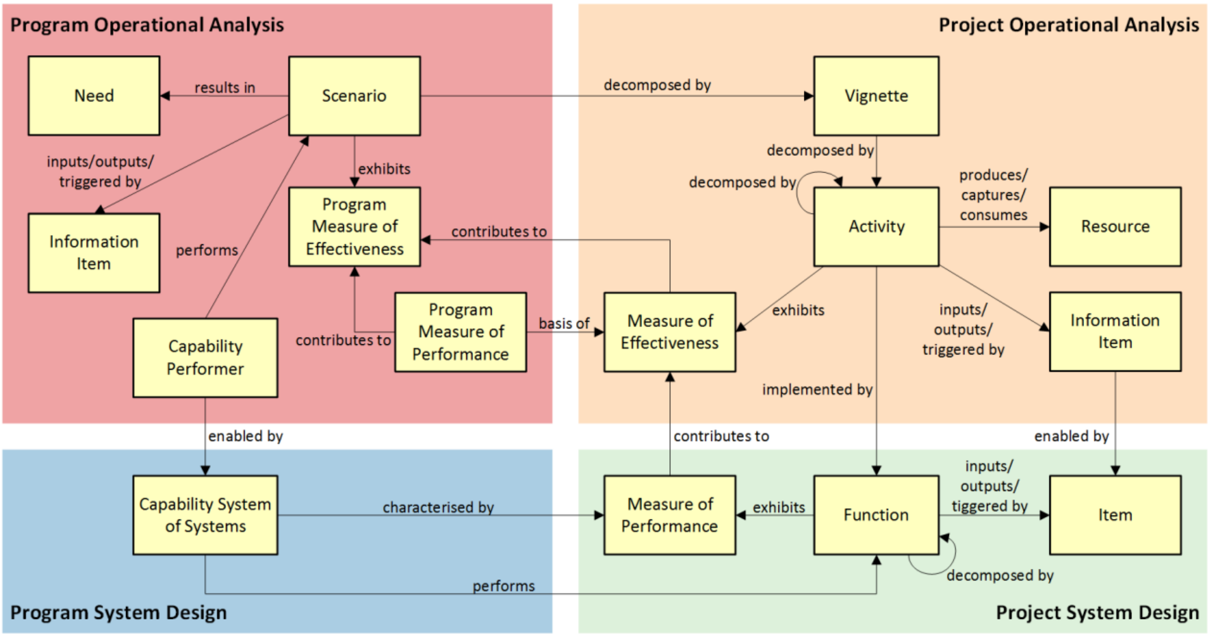

Figure 3. High-level Program metamodel to support model integration.

Figure 3 illustrates the relational links between the Program (left) and Project (right) models. In the demonstration model, the top-level activities are represented by the Scenario entity in the metamodel diagram, these Scenario activities are decomposed into Vignettes and Activities in Project-level models.

Of note is the structure of the measures; on the Program side, Program Measures of Effectiveness (PMOEs) and Program Measures of Performance (PMOPs) have been defined to differentiate from the Project-level equivalents of Measures of Effectiveness (MOEs) and Measures of Performance (MOPs).

PMOPs are the basis of the MOEs in the constituent Project model, but each can contribute to the PMOE, depending on the nature and context of the constituent Project model.

The concept for use of the performance characteristics is that the MOPs will inform the parameters of a set of operational activities for a specific simulation. Each of these operational activities will produce a result out of the simulation, the set of results are calculated to produce a Score MOE which is then compared against the threshold and objective attributes of relevant MOEs to establish the overall performance of the simulation. A similar process is then followed in the Program-level scenario simulation where the MOEs together with PMOPs contribute to the PMOEs.

CONCLUSIONS

The simulation demonstration was conducted as an example to test the concept. The data and numbers used to develop the model were derived from the historical literature and based on assumptions, so the results of the simulation may be imprecise. Yet the results strongly support the hypothesis that the simulation of Program scenarios based on low- to medium-fidelity information from constituent Project models can provide a suitable level of support to feasibility analysis, options analysis, and discovery of inconsistencies and gaps in functionality and capability.

Observations were made on how the Program capability responded under a number of variations. This information was used to rank options accordingly and could be used as an input to Program management decision making processes. The work also highlighted that Program simulations can be conducted in the absence of Project models, but that care needs to be taken with any assumptions made regarding product/solution performance in the scenario being simulated.

Another valuable finding from this research is that building an SoS (Program) model requires more than simply plugging constituent system models together. There needs to be a high-level functional flow model that is representative of the Program scenario that is traceable to constituent Project entities making use of Project data and functions. Work must be done to develop an overarching model that connects the models and directs the combined simulation according to the selected Program capability scenarios. This can be achieved by using an executable Program functional flow model that incorporates a method of integrating constituent Project model information.

While this study was in progress, a report commissioned by the Commonwealth of Australia and produced by Shoal and DST Group (Cook et al. 2017) provided a list of recommendations for successful integration of Projects and Products into Program capability. Of the recommendations, the class of model explored in this study would be able to address the following four:

1. Undertake Program capability SoSE design and analysis (including trade studies) for each stage of the Program capabilities evolution;
2. Establish and evolve Program SoSE assurance to provide convincing evidence that Programs will deliver the required capability outcomes for each Program capability stage;
3. Ensure that the Program-level is supported by model-based approaches, including the key artefacts, which should be austere, and where possible leverage existing databases and Project information. To facilitate this:
    a. Establish a common ontology and data model for capability development across the CLC;
    b. Ensure that information from key Program-level tools is consistent and can be exchanged between the tools and with external, analytical and executable modelling tools; and
    c. Iteratively develop model-based tools that facilitate the above and support SoSE approaches for Programs.
4. Implement targeted mission-engineering assessment of key SoS enablers (e.g. C4I Networks) and core SoS performance, with an intent to evolve towards an end-to-end mission engineering approach to inform evidence-based decision-making for Program SoSE. One aspect of this is the quantification of the value of any given constituent Product within a Program. (Cook et al. 2017)

This study has provided a clearer understanding of the information needs and the metamodel structure required to support simulation of Program scenarios, and it has demonstrated that Program scenario simulation can provide low-fidelity effectiveness predictions that are of value for Capability design and for the ongoing management of Capability effectiveness. Finally, it has provided a foundation on which to further investigate the integration of constituent Project models to support this.

REFERENCES

ADF Warfare Centre, *Australian Defence Doctrine Publication 3.14 Targeting*, Defence Publishing Services, 2009.

Bungay, S., *The most dangerous enemy: a history of the Battle of Britain,* London: Aurum, 2000.

Checkland, P. and Holwell, S., *Information, systems, and information systems: making sense of the field*. West Sussex, England: Wiley, 1998.

Cook, S. C. and Unewisse, M. H., "A SoS Approach for Engineering Capability Programs", *27th Annual INCOSE International Symposium*, Adelaide, Australia, 2017.

Cook, S. C. and Unewisse, M. H., "An Austere SoS Assurance Methodology", *Proceedings of the 28th Annual INCOSE International Symposium*. Washington DC, 2018.

Cook, S. C., Unewisse, M. H. & Flahive, A., *A SoSE Review of Program Capability Implementation*. Commonwealth of Australia, 2017.

Defence, *First Principles Review – Creating One Defence*, Commonwealth of Australia, 2015.

Defence, *Capability Definition Documents Guide*, Commonwealth of Australia, 2017a.

Defence*, Interim Capability Life Cycle Manual*, Department of Defence, 2017b.

Defense*, Joint Publication 3-60 Joint Targeting*, United States Department of Defense, 2013.

Dowding, H. C. T., "Battle of Britain in the Words of Air Chief Marshal Hugh Dowding", Retrieved 19 June 2017, from http://spitfiresite.com/2010/04/battle-of-britain-in-the-words-of-air-chief-marshal-hugh-dowding.html, 1941

French, C. and Heard, S., "Supporting capability integration and progress understanding of Programs and Projects through integrated information visualisation", *Proceedings of the Systems Engineering Test and Evaluation Conference 2018*, Sydney, 2018.

Hallett, J., Psalios, M., Jusaitis, D. and Cook, S. C., "The role of scenarios in model-based management of Capability Programs", *Proceedings of the Systems Engineering Test and Evaluation Conference 2018*, Sydney, 2018.

Hamilton, G., *Battle of Britain* [DVD], 1969.

Holland, J., *The Battle of Britain: five months that changed history May--October 1940*, London: Bantam Press, 2010.

Imperial War Museums, "What Was The 'Dowding System'?", *Imperial War Museums Website*, Retrieved 19 June 2017, from http://www.iwm.org.uk/history/what-was-the-dowding-system, n.d.

Johnson, B., *The Battle of Britain*, Granada Television, Retrieved June 2017, from https://www.youtube.com/watch?v=xf8mqaYg92I, 2006.

Omps, J., "An Overview of the Dowding System", Retrieved 19 June 2017, from http://beyourfinest.com/dowding-system-2/, 2014.

Robinson, K., Tramoundanis, D., Harvey, D., Jones, M. and Wilson, S., "Demonstrating Model-Based Systems Engineering for Specifying Complex Capability", *Proceedings of the Systems Engineering Test and Evaluation Conference 2010*, Adelaide, 2010.

Royal Air Force, “Battle of Britain in Detail”, Retrieved 19 June 2017, from https://www.raf.mod.uk/campaign/battle-of-britain-75th/the-battle/, 2017.

SEBoK, “Systems of Systems (SoS)”, *Systems Engineering Body of Knowledge* [Editorially controlled Wiki], Retrieved 19 June 2017, from http://sebokwiki.org/wiki/Systems_of_Systems_(SoS)#Types_of_SoS, 2017.

Trueman, C. N., “Group 11 Fighter Command”, *History Learning Site*, Retrieved 19 June 2017, from http://www.historylearningsite.co.uk/world-war-two/world-war-two-in-western-europe/battle-of-britain/group-11-fighter-command/, 2015.

Williams, A. G., “The Battle of Britain: Armament of the Competing Fighters”, Retrieved 19 June 2017, from http://www.quarryhs.co.uk/BoB.htm, 2005.

BIOGRAPHIES

Duane Jusaitis is an ACS-certified ICT professional with 14+ years of experience in IT development, management, and enterprise architecture across a range of sectors including Defence. He is an Enterprise Architect at Shoal where he applies his knowledge to a range of design and integration challenges and uses model-based systems engineering techniques to support conceptual design for better management of Defence capability and major systems. He holds a BIT (Hons) from the University of South Australia, and a MSc (Excellence) in Information Technology from the University of New South Wales Canberra where he conducted research into integration and execution of Program capability models.

Stephen Cook is a Systems Engineering Advisor with Shoal Engineering Pty Ltd where he applies his knowledge to a range of systems engineering management and research challenges. He is also the Professor of Defence Systems at the University of Adelaide where he works in the Entrepreneurship, Commercialisation and Innovation Centre undertaking research and teaching in system of systems engineering and complex project management. Until June 2014 he was the Professor of Systems Engineering at the University of South Australia where he led a number of research concentrations for over 15 years. Preceding this he accumulated 20 years of industrial R&D and SE experience spanning aerospace and defence communications systems. Prof Cook, PhD, is an INCOSE Fellow, a Fellow of Engineers Australia, and a Member of the Omega Alpha Association.

ISBN for SETE2018: 978-1-925627-15-2