Second International Symposium on Engineering Systems
MIT, Cambridge, Massachusetts, June 15-17, 2009

Architecting C4I Systems

Lean Weng YEOH
Defence Science and Technology Agency
71 Science Park Drive
Singapore 118253
yleanwen@dsta.gov.sg

Ming Chun NG
Defence Science and Technology Agency
1 Depot Road, #22-F03
Singapore 109679
nmingchu@dsta.gov.sg

Abstract. *Defence Science & Technology Agency (DSTA) has adopted the systems engineering methodology for developing C4I systems. The Systems Engineering approach ensures the alignment of the systems solution to business needs and processes of the users. It improves the processes and methodology through experimenting and harnessing new technologies such as Business Process Management and configuration management. Systems are baselined and fielded for operational validation and verification. Operational feedback are collected, analysed and synthesised so as to fill the operational gaps and to meet the future operational challenges. This paper addresses the key issues for designing and developing C4I systems. When architecting the C4I System-of-Systems, the systems architect has to synthesise the C4I systems with other present and future operational systems that meet the network-centric warfare concept. The paper also discusses the architecture principles that ensure all C4I systems are integrated and harmonised.*

1. Introduction

The Command, Control, Communications, Computers and Intelligence (C4I) systems provide battlefield information for the commanders to make decisions, and to control the military forces to accomplish missions. Defence Science & Technology Agency (DSTA) leverages state-of-the-art info-communications technology to design C4I systems that bring better situational awareness to the commanders. The C4I systems would provide comprehensive information to the commanders in a timely fashion and enable the commanders to disseminate orders expeditiously to the troops on the ground. This will enable the ground troops to execute their missions effectively. The earlier generation of C4I systems were largely designed in a stovepipe manner to serve a specific mission. As military operations are increasingly conducted in a network-centric manner spanning peacetime to hot war, the methodology for architecting and developing C4I systems had to evolve to meet the system of systems capability. To meet the complex operational requirements today, C4I systems must be able to inter-operate with other weapon systems as part of a larger complex system, or System-of-Systems (SoS). C4I SoS has to work coherently to deliver operational capabilities that are greater than the sum of what each component system can provide.

A Systems Architecting (SA) Process was developed to guide developers in designing robust, coherent, enduring and cost-effective C4I systems that would provide network-centric warfare capabilities. Essentially, SA enables the construction of an enterprise-level architecture for the various systems to inter-operate in an integrated and coherent manner in order to achieve synergistic operational capabilities. A component system could be a C4I system, weapon system, logistic system or IT system. Architecting these component systems to operate coherently and to deliver the intended SoS capabilities requires a balanced application of science and art. Rechtin suggested that SA is both a science and an art for designing and building effective and efficient SoS (Rechtin, 1997). It is a science because it uses architectural tools such as the Enterprise Architecture Framework to capture the various perspectives of the interconnecting systems for the stakeholders to share a common understanding of the SoS. The tools and techniques would also address the global integration, consistency and integrity of the SoS design. As an art, SA also aims to balance the local needs of the stakeholders of the component systems in the interest of a global optimum. More often than not, it is challenging to architect an optimal SoS through analytical or mathematical derivations alone. Instead, the SoS solution is often derived through a combination of analytical exercises and intellectual discussions and engagements with the key decision-makers and stakeholders. By leveraging holistic experiences of the leading domain experts, such as senior commanders and experienced large-scale systems engineering practitioners, the systems architects could understand the utility and effectiveness of new military operational concepts. They would also be able to develop new operational ideas. Through the collaborative approach in developing the SoS architecture, it would help to garner greater buy-in from the users and stakeholders. This close co-ordination among the stakeholders would ensure systems interoperability and encourage sharing of critical resources.

2. Systems Architecting Process

The SA Process is a six-step process designed to guide the systems architects in developing the SoS architecture from the systems conceptualisation stage to the systems operationalisation.

Figure 1 illustrates the SA Process. The first four steps focus on understanding the issue and developing an SoS architecture to address the issue. From framing the issue to developing the architecture, it typically takes about three to six months to complete. The fifth step is to implement the SoS. The time needed for design, development, installation, and verification of the SoS could span two to three years. The sixth step is to validate and certify that the SoS has addressed the identified issue. The SA process advocates *iterations* throughout the architecting process. This is especially important when the development of the SoS takes several years. By then, some of the operational requirements and boundaries may have changed. This may require the SoS architecture to be regularly re-examined for adaptation and to stay relevant.

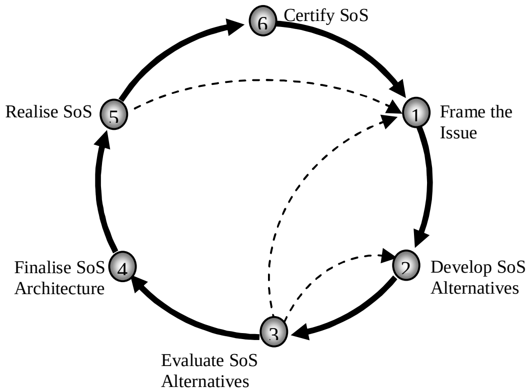

Figure 1: Systems Architecting Process

The SA Process exhibits *recursive* characteristic that is able to better manage the SoS complexity. As the SoS complexity increases, SA child processes could be spun off to handle the needs at four different levels such as product, system, capability and enterprise.

Figure 2 shows the relationships of the parent and child SA processes. The recursive characteristic provides the flexibility for the systems architect and stakeholders to evolve system requirements, conduct experiment with different products, and explore different approaches to the solution more effectively and efficiently. When an SA child process is completed, the capabilities are then consolidated at the parent level for eventual integration at the SoS level. For example,in an Integrated Air Defence programme, the architecting process would be spun off into several child processes to architect the command and control system, weapon systems, communications system and surveillance system. The resulting architecture at the system level would then be consolidated at the capability level to deliver integrated air defence architecture.

The following paragraphs will describe the details of the six-step systems architecting methodology.

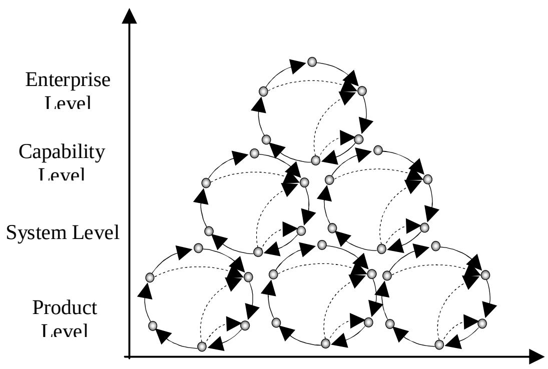

Figure 2: Multi-Level Application of the SA Process

Step 1 Frame the Issue. This step could influence the number of iterations of the subsequent steps before a satisfactory SoS architecture is reached. It aims to identify and articulate the higher intent and the system needs. It will uncover the underlying assumptions, constraints and limitations to establish a comprehensive, unambiguous and an accurate representation of the issue. In order to develop a deeper understanding of the issue, all the stakeholders are brought together to examine and evolve the strategic, operational and technical perspectives of the issue.

A set of techniques such as the Stakeholder Analysis, Systems Thinking, and Systems Decomposition are useful in assisting the systems architects in carrying out this step. The Stakeholder Analysis is used to identify all the individuals or groups who have the potential to affect certain activities in a network-centric operation and get them involved in the business analysis. Interviewing, conducting focus groups discussion, surveys and performing structured walkthrough with the stakeholders are some of the useful avenues to let the stakeholders air and register their concerns with the systems architects. Architectural diagrams, Use Case or Activity Diagram can also be used to capture the current processes and gaps in the individual component systems to help uncover constraints and limitations of the operating environment. The systems architect also applies Systems Thinking to construct the big picture overview of the issue. This enables him to adopt a top-down approach to define the various related issues in order to enhance the understanding of the higher intent.

Using the above techniques, capability gaps can be identified. These gaps will serve as input for the designers to develop the SoS requirements. A set of Measures of Effectiveness (MOEs) and its corresponding Measures of Performance (MOPs) that define how well the SoS would perform to meet the operational needs are also established at this stage for later use.

Step 2 Develop SoS Alternatives. The second step of the architecting process is to generate a broad range of possible architectural solutions to address the capability gaps. The emphasis is on the exploration of the solution space and to consider solutions involving any combination of doctrine, organisation, personnel, training, systems and facilities. Some techniques and tools such as Modelling and Simulation (M&S) and Functional and Requirement Analyses are useful to the systems architects for designing different architectural solutions.

During solutions exploration, the systems architects need to understand the inherent behaviours of the SoS alternatives. This allows the systems architects to evaluate on the potentiality of the SoS alternatives to address the identified capability gaps. However, it is usually very expensive to build the physical systems and to conduct field tests just for study or evaluation purposes, especially when weapon platforms are involved. Modelling and Simulation (M&S) is often used to enable the systems architects to understand the architectural behaviours of the SoS at the early design stage at a relatively low cost and with a shorter period of time. A model is a simplified representation of a system in a real world. Simulation is the manipulation of the model over time and space. Using M&S allows the systems architects to easily tune the model's parameters, which represent the properties of the real systems, and re-run the simulation to learn about the SoS' behaviours. By varying the model's parameters, the systems architects could generate different SoS alternatives to suit different degrees of operational needs such as high system responsiveness versus extreme system security.

Functional Analysis decomposes all facets of SoS capability and functionality requirements developed from Step 1 into a hierarchy of functions. At the top of the hierarchy is the enterprise level function. The enterprise level function is then successively partitioned into different sets of lower level functions or sub-functions. Further partitioning of the sub-functions may be required if its usefulness is warranted. Each set of sub-functions use some of the inputs and produces some of the outputs of the parent function. Eventually, a hierarchy of operational functions of the SoS is obtained. Requirement Analysis, in turn, groups the functions or sub-functions logically into different component systems that would eventually make up the SoS. The desired performance specifications for each component system could be specified where necessary. By varying the grouping of the functions into different component systems, different SoS alternatives could be generated.

Step 3 Evaluate SoS Alternatives. The third step is to evaluate the set of SoS alternatives based on performance, robustness and cost. The aim of the evaluation is to find the most flexible and scalable architecture capable of adapting to future requirements at a reasonable cost. At this stage, software models would need to be developed to represent each SoS alternative. Some of the software models that may have been built in the earlier step can be reused here. Separately, a set of test and evaluation parameters is also established to facilitate the incorporation of test criteria into the models. Some recommended methods and tools for the evaluation of SoS alternatives include M&S, Operational Analysis (OA) and Red Teaming.

OA could be utilised to provide quantitative assessment of the design of an SoS alternative. It uses a combination of mathematical and scientific methods such as mathematical models, probability and statistics, and algorithms to derive an optimal or near-optimal solution to complex problems. OA models such as combat models, resource/logistics models and cost models could be formulated to generate the optimal solution for addressing the identified issues.

Red Teaming is used for analysing systems from a devil advocate’s perspective. The ability to analyse possible courses of action from an adversary’s point of view helps the systems architects to identify the architectural vulnerabilities and rule out specific the SoS alternative. Alternatively, they may tweak or re-design the SoS alternative to mitigate or eliminate the vulnerabilities.

During the evaluation process, new insights from the analysis may emerge that may result in the need to reframe the issue or to refine the SoS design. It may be necessary to go through several iterations before a satisfactory architectural solution is derived.

Step 4 Finalise SoS Architecture. The output of SA is an endorsed SoS architecture. This SoS architecture is described in terms of architectural views in accordance with the EA framework and governance guidelines. The documentation will facilitate the promulgation, communication, master planning and realisation of the SoS architecture.

The finalised SoS Architecture will facilitate the formulation of the various master plans that will chart the milestones for capability build-up, resource planning and competency development.

Step 5 Realise SoS. The SoS is then built based on the endorsed architecture. Given the size and complexity of a typical SoS, its implementation is usually broken up into several component systems for better management. There will be different programme teams responsible for the acquisition and development of various component systems. Where necessary, a Technical Working Group or Programme Steering Committee may be formed to provide management guidance to the programme teams. The needs for comprehensive planning and close collaborations among the various programme teams should be emphasised. While each of the programme teams is likely to have its implementation schedule and priorities, there will be a master schedule to keep a tab on the progress of each component system so that component systems integration and capability demonstrations could be carried out at scheduled milestones. It will also serve to track the progress SoS integration.

During the SoS realisation stage, any deviation of the SoS architecture will need to be raised at appropriate governance forums for endorsement. Since the realisation of SoS may take several years, it is possible that the changing external environment may invalidate the assumptions made during the architecting process. This may result in the need to re-examine the SoS architecture's relevance to the new environment.

Step 6 Certify SoS. Verification, validation and certification of the SoS are essential activities during this process. The SoS will be evaluated and validated for its capability and performance with respect to the master plan. Verification is a quality management process to ensure that a system complies with specifications and should be conducted throughout the systems development phase. Whereas, validation is used to establish a level of confidence that a system accomplishes its intended mission capabilities and addresses user needs. When the SoS is successfully verified and validated with respect to the master plan, the systems architects can proceed to certify the SoS with the customers and stakeholders.

The different capability build-up plans for various component systems will lead to several SoS capability milestones. Since SoS consists of multiple component systems, end-to-end testing is generally costly and time consuming especially when weapon platforms are involved. Therefore, M&S developed in an SoS Integration Lab would serve as valuable surrogates to emulate end-to-end SoS validation. With the aid of M&S, key test points can be selected to validate the SoS performance. The process of verification and validation may lead to new insights or discovery of unexpected emergent behaviour. It is therefore important to perform regular monitoring of SoS operations to look out for emergent behaviour that may warrant adaptation to the architecture. If changes in external environment resulted in deficiencies in the SoS architecture, it may be necessary to re-examine the relevance of the SoS architecture and reinitiate the whole architecting process.

3. Enabling Architecting C4I Systems

As armed forces adopt network-centric approach to conduct military operations, C4I systems have to be designed based on SoS concept where the C4I systems of the different stakeholders are integrated horizontally so that all the command posts are interoperable by design. The network-centric warfare concept marks a departure from the traditional command-centric approach in which the stakeholders operate more independently in a missions. Thus, when the network-centric warfare concept was introduced, our stakeholders had to align quickly with this transformation. They were faced with a critical need to articulate the transformed operational requirements to support this new concept of operations.

In order to have better clarity about their needs, the stakeholders and technologists were brought together through meetings, forums, working groups and focus groups to share and discuss their suggestions and proposals, voice their concerns, and to brainstorm new ideas. The key stakeholders were identified and grouped according to their seniority and professional domains for better management. The senior commanders were first consulted to understand the strategy of conducting future military operations with a system of fully networked fighting forces. This vision helped to provide guidance for discussions at the lower levels. Some stakeholders were also grouped into different domain groups like air base operations, logistics or human resource to address specific operational needs that would be required to support network-centric operations. The Enterprise Architecture Framework (EAF) was used during this frame-the-issue stage to translate the senior commanders' operational vision into specific systems requirements. Based on the needs gathered from different levels and domains, the technologists were able quickly produce an initial hierarchical structure of system components and functions by using the System Decomposition technique. The EAF's architectural views were used to capture the operational, system and technical perspectives of the desired SoS for the purpose of promulgation and communication to achieve synchronous understanding among the stakeholders and technologists, as well as for guiding the SoS development at the later stage.

<u>EAF's Architectural Views</u>. There are five types of architectural views in the EAF designed to describe architecture from different viewpoints. They are the *Strategic Views, Service Views, Operational Views, Technical Views* and *System Views.* The Strategic View describes the current capabilities, gaps, dependencies, and options available to close the gaps. It also describes the desired end-state, the systems capabilities required and the strategy or approach to undertake to acquire the capabilities in order to reach the end-state. The Service View defines the business, technical and non-functional services required to provide a capability. The Operational View describes the operation processes and activities, the information exchanged among operational units and the outcomes of the missions. The Technical View describes the standards and the products used for the system implementation. The System View describes the interfaces among systems, physical deployment of systems, system functions, the information exchanged by these functions, the mapping of the functions over the operational nodes, and the data model at the system level. The System View helps to identify the new system functions to be built and those to be enhanced or reused.

Indeed, when acquiring or developing C4I capability, the EAF advocates the reuse of software components that were previously built and tested for operational deployment and trial wherever possible. If re-use is not possible, the project team will develop the new component. The project team would then contribute the components to the Common Repository for further re-use by other project teams. The objective is to reduce the systems development time so that new capabilities could be fielded rapidly. The Common Repository and Service-Oriented Architecture (SOA) were put in place to allow us to build upon the intellectual capital to develop and field our C4I systems more rapidly. The Business Process Management System (BPMS) is adopted to model users' workflow in order to further reduce the time-to-fielding and to provide operational agility by complementing SOA.

Common Repository. The common repository keeps the systems business application and technical component services that developers could draw upon to rapidly assemble and deploy C4I systems. As the repository applications and services are thoroughly tested for operational deployment, the assembled C4I systems would achieve a high degree of assured quality for operational trial and deployment. The common repository is an enterprise asset that must be properly maintained, continually expanded in the number of re-useable components and evolved through a rigorous quality management process. New applications and services should only be developed when operational gaps are identified and they must be sufficiently tested for performance and reliability before being released into the pool of reusable software.

Service-Oriented Architecture (SOA). Similar to reusing software components from the common repository, SOA shortens the C4I systems development time through reusing existing services. Essentially, SOA is about exposing business logic as services to an external application or service. It serves as a development and integration platform for rapid development of new business application or service by assembling the necessary services. SOA is often implemented with the use of Web Services (WS), it utilises ubiquitous standards or protocols for data exchange and communication such as XML and Internet Protocol. The use of common standards and protocols eases the task of integrating heterogeneous systems in a network-centric environment. Figure 3 shows a multi-tier C4I systems architecture on which our C4I systems are built. The Business Logic Layer shows a simplified view of the SOA and the range of services that were incrementally built up over the years.

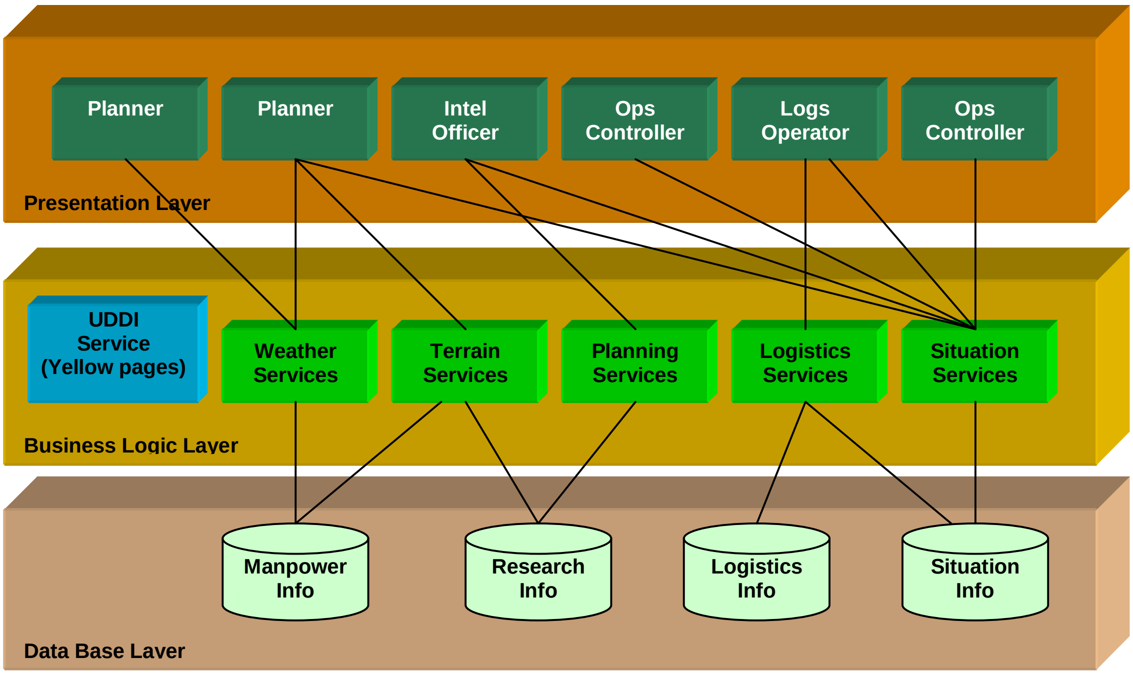

Figure 3: Multi-tier Architecture

Business Process Management System (BPMS). BPMS is the IT enabler for managing processes from process definition to process retirement. Most commercial BPMS solutions support rapid graphical modelling, development, testing, simulation, optimisation, deployment and monitoring of the processes. BPMS enables new operational capabilities that involve flowing of tasks among different parties to be developed and deployed faster as compared to traditional development through coding. When BPMS combined with SOA, they further shorten the development time of a new capability. Some process-oriented capabilities could be developed by system configuration or by assembling existing software components and services in the SOA. As BPMS externalises the business logic from the software code, the changes to the processes can also be implemented easily and deployed on the fly. This gives the much-needed operational agility to the stakeholders as they operate in a very dynamic environment. BPMS is an emergent yet highly promising technology that we are currently being explored for C4I systems development.

While the EAF is used to guide the design of C4I systems and the various technologies like SOA are used in rapid development of C4I capabilities, it is essential that proper governance and competency development frameworks are put in place to ensure that systems are built according to established standards and are interoperable by design.

IT Governance. In DSTA, a technical working group was set up to ensure that the architecture and technical solutions of a system comply with the reference architecture or the approved SoS architecture. As part of the DSTA's Quality Management System, every C4I system project is required to seek the technical working group's endorsement for its solutions prior to the start of its development. The project manager is required to submit the solution in the form of system and technical views. The working group also scrutinizes the operational views to ensure the alignment of the system solution to business needs and processes of the users.

Competency Development. As technologies advance by leaps and bounds, it is important for an organisation to invest constantly in their human resource by upgrading the technologists' technical skills and developing new competencies. The effort includes identifying emerging competencies needed for future challenges and committing resources, time and effort to develop the technologists. The Organisation Capability Development Division works with the individual technologist to produce the annual learning plan that specifies his training needs, and the schedule and resources required to build up the skills and competency. A systems architect would need to be trained in diverse areas of technical domains as well as business competency areas. He should be competent in complex systems design, systems thinking and business analysis. The C4I Systems Architecting Guidebook and the C4I Development Guidebook and Competency Portal in DSTA's intranet are good sources of information for him to begin with. The guidebook embodies a wealth of architecting knowledge that was accumulated and compiled by a group of experienced systems architects who had been through many rounds of C4I systems developments. The portal serves as an online avenue for the C4I community to exchange views and share their experiences in C4I systems development.

4. Conclusion

The network-centric warfare concept elevates the need for systems interoperability to an unprecedented level. In the modern world where technologies advance at blazing speed, new threats emerge faster than ever before. This drives the demand for new counteracting capabilities to be deployed at an almost equal rate. Methodologies for architecting C4I systems need to be robust enough to bring systems to fielding in a shortest possible time. The search for alternatives to harness state-of-the-art technologies for rapid development and fielding of new operational capabilities is a never-ending business for the C4I systems architects.

References

[1] Cebrowski, Vadm A, Garstka, J, *Network-ceentric Warfare: Its Origin and Future*, Proceedings of the Naval Institute, 1998.

[2] Mo Jamshidi, *Systems of Systems Engineering – Principles and Applications*, CRC Press, 2009.

[3] Rechtin, E, *Systems Architecting: Creating and Building Complex Systems*, Prentice Hall, 1997.

[4] Rechtin, E, Maier, M.W, *The Art of Systems Architecting*, CRC Press, 2000.

[5] Yeoh, L.W, Chew, L.P, *System Engineering Approach to Development of C2I Systems in the Knowledge-Centric Age*, MIT ESD 2004 Symposium, 2004.

[6] Yeoh, L.W, Syn, H.B, Lam, C.V, *An Enterprise Framework for Developing Command and Control Systems*, 17th Annual International Symposium of the International Council on Systems Engineering, 2007.

[7] Yeoh, L.W, Teo, T.L, Lim, H.L, *Continuous Systems Development for Command, Control and Intelligence Systems*, 18th Annual International Symposium of the International Council on Systems Engineering, 2008.

Biography

Lean Weng Yeoh is Director C4I Development and DSTA Systems Architect. He is also concurrently, Deputy Director of TDSI and Adjunct Professor at the National University of Singapore (NUS). He received his Bachelor and MSc degrees from NUS in 1983 and 1987 respectively. He further obtained two Masters in 1990 and a PhD degree in Electrical Engineering in 1997 from the Naval Postgraduate School. He has extensive experience working on large-scale defence engineering systems. As a systems architect, he played a key role in developing the Enterprise Architecture for defence applications. He developed systems architecting methodology for masterplanning and defence transformation. He has published several papers on Enterprise Architecture, experimentation methodology and Integrated Communications Architecture. He is also Vice-President of the INCOSE Singapore Chapter, INCOSE Region VI Representative to Member Board and Chairman of Systems Engineering Technical Committee, Institution of Engineers, Singapore.

Ming Chun Ng is a Senior Engineer with DSTA. He received his Bachelor of Science (Hons) in Computer Science and Information Systems from National University of Singapore in 1997. He holds a Masters of Science in Computer Science and Information Systems from National University of Singapore in 1999. He researched in distributed systems and had published a paper on Adaptive Home-based Software Distributed Shared Memory System in International Conference on Parallel and Distributed Systems 2000. He is presently leading a C4I development team, and pursuing his interest in Systems Architecting and Business Process Management System.