Systems Engineering, Architecture Frameworks and Modelling & Simulation

Dr. Tommy Nordqvist
Front End Strategy AB
SE-582 21 Linköping
Sweden
+46 13 24 93 51
tommy.nordqvist@frontend.se

Ms. Anna Edberg
Front End Strategy AB
SE-582 21 Linköping
Sweden
+46 13 24 93 51
anna.edberg@frontend.se

Mr. Håkan Byström
Front End Strategy AB
SE-582 21 Linköping
Sweden
+46 13 24 93 51
hakan.bystrom@frontend.se

Mrs. Marie-Louise Gustafsson
Swedish Defence Material Administration
SMART-Lab
SE-115 88 Stockholm
Sweden
+46 8 7826870
marie-louise.gustafsson@smart-lab.se

Keywords:
Systems Engineering, Architectural Frameworks, Modelling & Simulation (M&S)

ABSTRACT: *The international standard for systems engineering ISO/IEC 15288 "Systems Engineering Life Cycle Processes" receives more and more attention; probably due to an increasing need for a common framework within the systems engineering community. For example; NATO has decided that ISO/IEC 15288 shall be the framework for Systems Life Cycle Management (SLCM). Architectural Frameworks is sometimes regarded as the necessary communication tool of system design when stakeholders communicate what the systems are, the purpose they fulfil, and what the systems elements are. Department of Defence (DoD), NATO, Ministry of Defence (MOD) and the Swedish Armed Forces to mention a few, all develop Architectural Frameworks. Modelling & Simulation (M&S) is the "silver bullet" that will make systems engineering more efficient with respect to delivering the right systems; within budget and time limits. The present paper will give a brief introduction to ISO/IEC 15288, Architectural Frameworks and M&S. It will also present an example on how ISO/IEC 15288, Architectural Frameworks and M&S can be integrated and the benefits of that integration, especially how M&S can be used to develop and execute system designs. Finally this paper gives a very short presentation of some ideas concerning tool support. The presented work was performed under contract from the Swedish Defence Material Administration.*

1 ISO/IEC 15288 Systems Engineering Life Cycle Processes

1.1 Introduction

The international standard ISO/IEC 15288 ”Systems Engineering Life Cycle Processes” [2] describes processes appropriate for systems engineering and offers stakeholders a common process framework that improves the communication and cooperation during a systems entire life cycle.

The processes in ISO/IEC 15288 can be used by organisations, projects, acquisition managers and/or system suppliers to create life cycle models appropriate to the products and services. For example; an organisation can use the processes to develop a common framework supported by methods, procedures, tools and trained personnel. Based on the framework, a project can decide which processes to use to secure the delivery of the intended products and services. Acquisition managers and system suppliers can use the framework to reach an agreement concerning the processes used by the respective party and the necessary demands on the processes. NATO has decided to use ISO/IEC 15288 as the framework for Systems Life Cycle Management (SLCM) [1] and the Swedish Defence Material Administration (FMV) to use ISO/IEC 15288 as the framework for systems engineering.

The ISO/IEC 15288 processes applies for all levels in a systems hierarchy and selected processes can be used throughout a systems entire life cycle to guide and perform the activities necessary in the life cycle stages.

1.2 The Processes

The Systems Engineering Life Cycle Processes are divided into four groups; there are Enterprise Processes, Agreement Processes, Project Processes and Technical Processes. Figure 1 illustrates the groups with the included processes. Each group will be described very briefly in the following sections.

The **Enterprise Processes** consists of five processes. With the *Enterprise Environment Management Process* the policies and procedures necessary for an effective systems life cycle management are defined and retained. The *Investment Management Process* is used to initiate and finance the projects needed to fulfil the organisational goals. To guarantee that efficient life cycle processes are available, the *System Life Cycle Processes Management Process* is used.

The *Resource Management Process* is used to guarantee that resources (trained personnel, material etc.) are available to the project in an effective way. The *Quality Management Process* is used to guarantee that products and services fulfil the required quality and the stakeholders’ requirements.

The **Agreement Processes** consists of two processes; the *Acquisition Process* is used to acquire products or services and the *Supply Process* to deliver them.

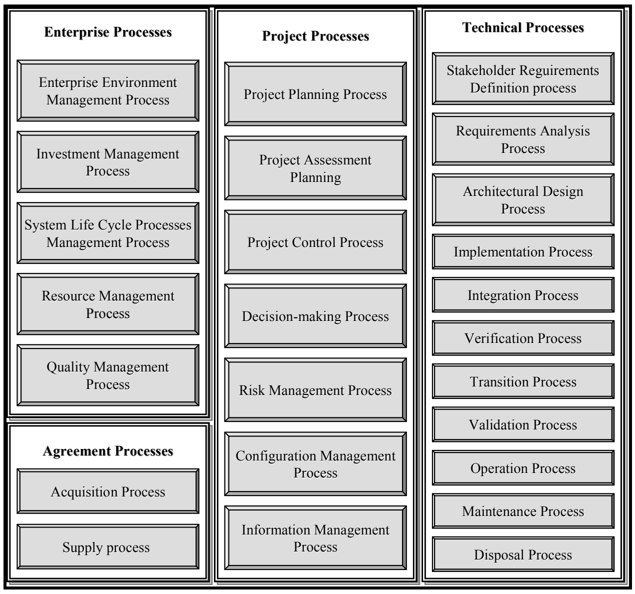

Figure 1. *Systems Engineering Life Cycle Processes divided in Enterprise Processes, Agreement Processes, Project Processes and Technical Processes.*

The **Project Processes** consists of seven processes. The *Project Planning Process* is used to develop and communicate effective and useful project plans. The *Project Assessment Process* is used to determine that the project progress as planed. The *Project Control Process* is used to manage the project according to the project plan, budget and defined goals. The *Decision-making Process* is used when it is necessary to choose between different implementations of a project or parts of a project. The *Risk Management Process* is used to diminish the effects of unforeseen events that could have a negative influence on a projects quality, budget or time schedule. The *Configuration Management Process* is used to secure the integrity of all results and that the results are available to all concerned parties. The *Information Management Process* is used to guarantee that all stakeholders receive the right information, at the right time and with the right quality.

The **Technical Processes** consists of eleven processes. The *Stakeholder Requirements Definition Process* is used to elicit the stakeholders' service requirements. The *Requirements Analysis Process* is used to develop the stakeholders' requirements further so that they can be used as a basis in the development of a system that delivers the services needed. The *Architectural Design Process* is used to design a system that complies with the requirements. The *Implementation Process* produces the specified system elements and the *Integration Process* is used to integrate the system elements according to the architectural design. The *Verification Process* verifies that the system elements are produced and integrated according to the design. The *Transition Process* is used to install the verified system together with the necessary support systems. The *Validation Process* is used to guarantee that the services provided by the system are according to the agreement. The purpose of the *Operation Process* is to use the system in order to deliver its services. The *Maintenance Process* is used to guarantee that the system deliver the correct services as long as the system is in operation. The *Disposal Process* is used when the system is to be terminated.

The interactions between the process groups within and between organisations are illustrated in Figure 2.

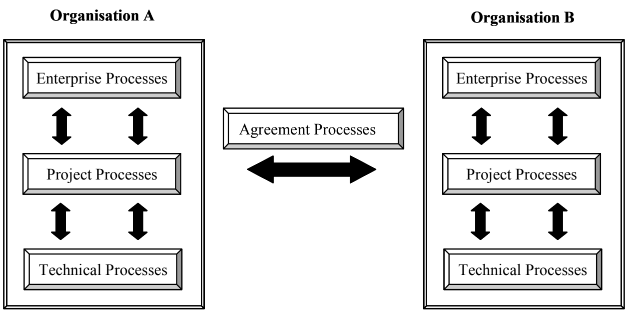

Figure 2. *Interaction within and between organisations. The Enterprise, Project and Technical Processes are used within an organisation while the Agreement Processes are used between organisations.*

1.3 The Life Cycle of a System of Interest

Every System of Interest (SOI) has its own life cycle. A life cycle can be decomposed into a set of stages; an example of a life cycle, from ISO/IEC 15288 [2], is shown in Figure 3.

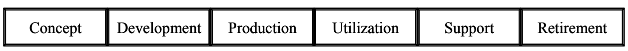

Figure 3. *Life cycle stages from ISO/IEC 15288.*

The life cycle is partitioned into stages where each stage represents one essential period of a systems life time and has a distinct purpose. The partitioning is practical since it is easier to work in small, understandable and timely steps.

The **concept stage** starts with a decision to fill a capability gap and ends with a requirements specification. One or more system alternatives that fill the capability gap are developed and evaluated regarding needs, risks and cost benefits.

In the **development stage** a SOI that meets the user requirements is developed. It is important that the SOI is developed with the proceeding stages in mind so that the system can be produced, utilized, supported and retired.

In the **production stage** the product is produced and tested. If there is a need for supporting systems, these are produced as well.

In the **utilization stage** the product is operated at the intended operational site where the required services are delivered.

Most systems need support and this is done in the **support stage** where logistics, maintenance and support are provided.

The **retirement stage** begins when a SOI is taken out of service. This stage provides for the removal of a SOI and its related services.

The partitioning in stages also helps address uncertainties and risk associated with cost, schedule, general objectives and decision making. The transition from one stage to the next is done through decision gates with entry/exit criteria. At every decision gate the project need to revise if the project should execute the next stage, continue the current stage, go back to a preceding stage, terminate the project etc.

1.4 The Processes and the Life Cycle of a SOI

Most of the processes presented in section 1.2 are used to some extent during every stage of a systems life cycle. Figure 4 roughly illustrates to what extent the process groups, here with focus on the technical processes, are used in the different stages.

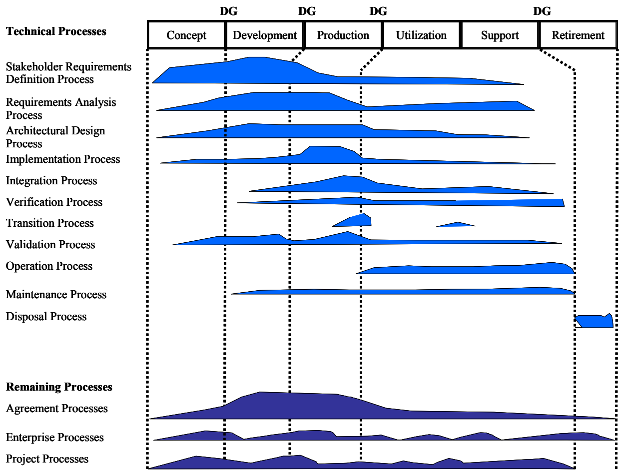

Figure 4. *An example of the processes in relation to the life cycle presented by ISO/IEC 15288.*

2 Architectural Frameworks

2.1 Introduction

The efforts within the area of architectural frameworks that we know of (C4ISRAF, DoDAF, NAF, FEAF, ATTCIS, SwAF and MODAF) all strives to create a common framework for descriptions of system architectures. [4, 5, 6, 7, 8, 9, 10]

Two of the key reasons for developing architectural frameworks can be found in the following paragraphs:

"The Defence Science Board and other major studies have concluded that one of the key means for ensuring interoperable and cost effective military systems is to establish comprehensive architectural guidance for all of DoD." [3]

"The committee believes that the absence of enterprise architecture has been a major contributor to the problems faced by the implementers of the [FBI's] Trilogy program. That is, the lack of architecture to guide the planning of an information and communication infrastructure has resulted in improvisation that has virtually no chance of resulting in a well-ordered infrastructure for the enterprise to build upon. In fact, merely providing parts (e.g., computers and accessories, piece-part applications, and so on) is like buying brick, mortar, and lumber and expecting a builder to produce a functional building without benefit of building codes, blueprints, or an understanding of how people will use the building." [4]

The primary aim with Architectural Frameworks [5, 6, 7, 8, 9, 10], according to our interpretation, is to support development and documentation of system architectures, we would like to call it "intellectual interoperability"; in other words to provide the means for communication between stakeholders when creating systems.

Architectural frameworks make it possible to describe and communicate system architectures within and between organisations, both national and international, providing that, the same, or similar architectural frameworks are used. Without a common language it is difficult to communicate.

Architectural frameworks provide guidelines regarding content and how to describe architectures; they do not provide any guidance concerning how to design or implement a specific architecture or how to develop or acquire systems.

Some of the stakeholders that benefit from architectural descriptions are:

- System investments decision-makers.
- The ones defining capability requirements.
- Interoperability managers.
- The ones acquiring systems.
- System suppliers.
- System developers and producers.

Architectures are used to identify capability needs, relate needs to systems development and integration, attain IT interoperability and supportability, and manage IT investments.

2.2 Architecture Frameworks: The MODAF Example

MODAF [8] is chosen to exemplify an architectural framework since it is the latest architectural framework released and since the next version of NATO's Architectural Framework will be based upon work within MODAF. In our opinion, the basic principals and thoughts described in MODAF also conform to other architectural frameworks.

MODAF has been designed to meet the specific business and operational needs of the Ministry of Defence and defines a representation that enables the stakeholders to focus on specific areas of interests; whilst retaining the big picture. MODAF provides the means to model, understand, analyse and specify capabilities, business processes and technical systems in the enterprise.

In MODAF the problem space is split into manageable pieces, defined as views, which enables decision-makers to manage complexity. Each view has a particular purpose and the views are categorised under six viewpoints, shown in Figure 5.

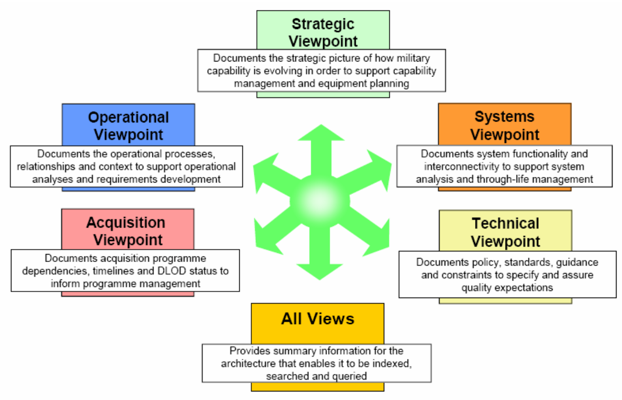

Figure 5. *MODAF six viewpoints [8].*

The six viewpoints depicted in Figure 5 consist of 38 views and each view represents a particular aspect of a system. Some of the views provide high level information about the system; whilst others describe specific aspects or relationships between different aspects of the system. Users select the views that represent their needs most effectively for a given architecture; not every view is appropriate or necessary.

The Viewpoints

The **Strategic Viewpoint** consists of six views. The *Capability Vision* describes the capability vision over time and how goals and strategies shall be reached in terms of capabilities; the *Capability Taxonomy* is used to structure and decompose capabilities; the *Capability Phasing* shows the available military capability at different points of time during specific time periods; the *Capability Clusters* is used to describe the relationships between capabilities; the *Capability to Systems Deployment Mapping* shows the deployment of systems in organisations, including their relations, that satisfies the capability need for a particular period of time; the *Capability Function to Operational Activity Mapping* is used to describe the mapping between the capability elements and the operational activities supported by those capabilities.

The **Operational Viewpoint** consists of seven views. The *High Level Operating Concept* is a high level description of the system divided into three views; one that graphically describes the system context, one that supplement the graphics with textual descriptions and one that provides detail of the operational performance attributes associated with the scenarios. The three views constituting the High Level Operating Concept are shown in Figure 6. The *Operational Node Connectivity Description* graphically describes operational nodes (or organisations) and the need for information exchange between those nodes.

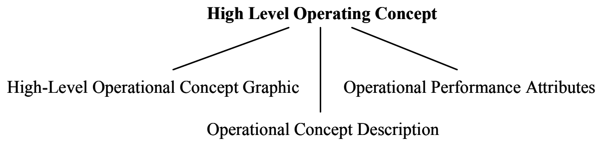

Figure 6. *The three views constituting the High Level Operating Concept.*

The *Operational Information Exchange Matrix* describes the information exchange in terms of information necessary, with whom, why the information is necessary and how the exchange must occur. The *Organisational Relationships Chart* is used to describe the command structure. The *Operational Activity Model* describes the operational activities necessary, according to the scenario(s) developed in the High Level Operating Concept views. The *Operational Activity Sequence & Timing Description* consists of three views; the *Operational Rules Model* specifies the rules or constraints on an enterprise or business, the *Operational State Transition Diagram* describes how an operational node or activity responds to different events by changing its state and the *Operational Event Trace Description* is used to describe the information exchange over time in a particular scenario. The three views constituting the Operational Activity Sequence & Timing Description are shown in Figure 7. The *Logical Data Model* shows the structure and relationships between operational data elements.

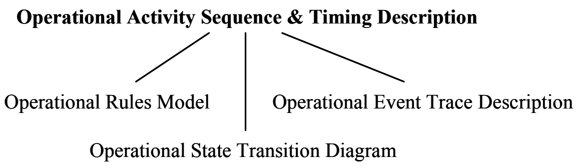

Figure 7. *The three views constituting the Operational Activity Sequence & Timing Description.*

The **Systems Viewpoint** consists of 11 views. The *System Interface Description* is used to describe systems and the interfaces between them. It also shows the nodes at which the systems are located. The *Systems Communications Description* consists of three views, the *System Port Specification* specifies system ports and protocols used by those ports when communicating with other systems, the *System To System Port Connectivity* defines the protocol stacks used when connecting two ports and the *System Connectivity Clusters* is used to define how individual connections are grouped into logical connections between nodes. The three views constituting the Systems Communications Description are shown in Figure 8.

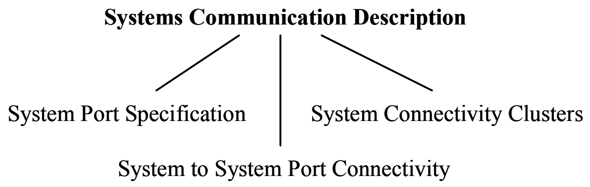

Figure 8. *The three views constituting the Systems Communications Description.*

The *Systems-Systems Matrix* is used to describe the interface characteristics (described in the System Interface Description) in detail. The *Systems Functionality Description* describes functional hierarchies, system functions and data flow between systems. The *Operational Activity to Systems Functionality Traceability Matrix* shows the relations between operational activities and the system functions that supports them. The *Systems Data Exchange Matrix* specifies the characteristics of the system data exchanged. The *Systems Performance Parameters Matrix* is used for specifying the quantitative characteristics of systems and hardware/software elements, their interfaces and functions. The *Systems Evolution Description* is used to plan the systems or the architectures evolution over time. The *Systems Technology Forecast* defines the development of the supporting technologies over time. The *Systems Functionality Sequence & Timing Description* consists of three views; the *System Rules Model* describes the rules compulsory for the architecture or its systems, the *Systems State Transition Description* graphically describes how a system and its subsystems reacts to various events; the *Systems Event-Trace Description* is used to describe the data elements that are exchanged between particular systems. The *Physical Schema* defines the structure of various data types used by the system.

The **Technical Viewpoint** consists of two views; the *Technical Standard Profile* presents the standards and conventions constraining both the architecture it self and its realisation, and the *Technical Standards Forecast* describes the expected changes of the standards and conventions presented in the Technical Standard Profile.

The **Acquisition Viewpoint** consists of two views; the *Systems of Systems Acquisition Clusters* shows how multiple acquisition projects relate to each other from an organisational point of view, and the *Systems of Systems Acquisition Programmes* provides an overview of the time-line and dependencies between a program and individual projects.

The **All Views** also consists of two views; the *Overview and Summary Information* provides a brief description of the architecture including for example purpose and views to be developed for the system in question, and the *Integrated Dictionary* is used to define terms used within the architecture.

NATO is currently working on their architectural framework version 3.0 that will use the result of the work with MODAF, possibly complemented with a Service Viewpoint. The Service Viewpoint aims at describing the services provided by a system [5]. The Swedish Armed Forces are also conducting work similar to NATO [10].

3 Modelling & Simulation

Before we speak further about modelling and simulation (M&S) we will define the terms model and simulation. The following definitions are from [11].

Model
A "model" is a mathematical, logical, physical, or procedural representation of some real or ideal system.

Simulation
A "simulation" is the implementation of a model in executable form or the execution of a model over time.

According to our interpretation of the definitions above, a simulation can be anything from an exercise to a complex model in an advanced computerised environment.

3.1 Why Modelling & Simulation

The complexity of systems increases and the time from concept to utilization becomes longer and longer. To handle this complexity and to shorten the time for development, tools are necessary. Modelling and simulation is one tool that can contribute to managing both the complexity and the time issue.

The developments of models (modelling) that are used in simulations are a tool to clarify and visualise ideas concerning a system. Modelling and simulation contributes to the understanding of the system as well as the business that the system intends to support.

Other examples of motives to why M&S is a valuable tool can be found in [12] where at least two important principles for systems development are pointed out; *Business Oriented Systems Development* and *Life Cycle Oriented Systems Development*.

Business Oriented Systems Development means that the development of systems always focus on the purpose of the business and how the business best should be operated, not on the product alone. Modelling and simulation makes the Business Oriented Systems Development possible in a concrete way. Both needs and current circumstances can be visualised by developing models and simulations of the business, scenarios and processes. The visualisation and elucidation provided by modelling and simulation result in a maintained focus during the systems entire life cycle. The communication and discussions improves between those with needs and requirements, and those who suggest and develop solutions.

Life Cycle Oriented Systems Development aims towards early foreseen of future requirements regarding the system in relation to development, production, utilization, support and retirement. Early and gradual implementation and integration of product models and simulations, defined iteratively during development, is supported by the principals of a life cycle oriented system development. In other words prototypes, with various levels of detail, are developed through modelling. The models of the products are then evaluated and tested regarding future life cycle stages through execution of the models (simulation).

The Federation Development and Execution Process (FEDEP) define one approach to development of models and execution of those models in simulations and in particular High Level Architecture (HLA) Federations. FEDEP can be recommended as a method for all development of models and executions of simulations.

4 ISO/IEC 15288, Architectural Frameworks and M&S

ISO/IEC 15288 is a process framework for systems engineering, architectural frameworks defines the information necessary regarding systems architecture and M&S is a tool that can benefit to systems engineering in general and to development of systems architectural descriptions in particular. The architectural descriptions are models, and by using the models in simulations they can for example be verified and validated. In Figure 9 the basic building blocks, SOI, ISO/IEC 15288, architectural frameworks (MODAF) and M&S, are illustrated. The foundation in this illustration is the systems life cycle; the life cycle model is taken from ISO/IEC 15288. [2]

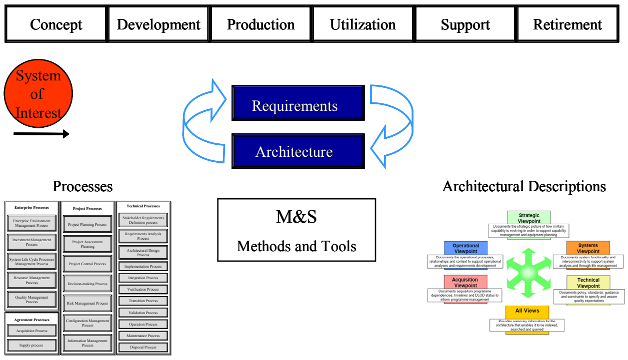

Figure 9. *The basic building blocks.*

Systems engineering is about how to manage a system through its entire life cycle. To support this, a couple of products, both informational and physical, need to be developed. The processes in ISO/IEC 15288 can be a valuable support in this work. One way of documenting the informational products that needs to be developed are architectural descriptions.

If we take a closer look on systems engineering and architectural frameworks, the essence in systems engineering is the interaction between requirements and solutions. In other words, during most of the systems life cycle, requirements are identified and put into solutions (architectures, physical products etc.). For example, preliminary requirements are developed during the concept stage, these requirements are analysed and serves as the basis for one or more potential solutions (concepts). These concepts are usually expressed in some kind of model (sketches, preliminary design suggestions, prototypes etc.).

If we play with the thought that MODAF (or any other architectural framework) represents our framework for the description of all solutions during a systems life cycle, then we will have a set of models describing the system for every life cycle stage.

To secure that these solution models describes what they are intended to and that this depiction is done in a correct way, different types of verification and validation (V&V) is demanded. One tool for V&V of models is simulation, which are execution of the models.

To exemplify the interplay between systems engineering, architectural frameworks and M&S we assume that we have a task to develop one or more concepts of a system.

Based on a systems engineering perspective, according to ISO/IEC 15288, the Define Stakeholder Requirements Process is used to identify the stakeholders' needs. The needs are then developed further into a first version of requirements regarding the system. This is done using the Requirements Analysis Process. The requirements are then analysed in purpose to develop one or more possible solutions that meets the stakeholders' needs. This analysis and development is done using the Architectural Design Process.

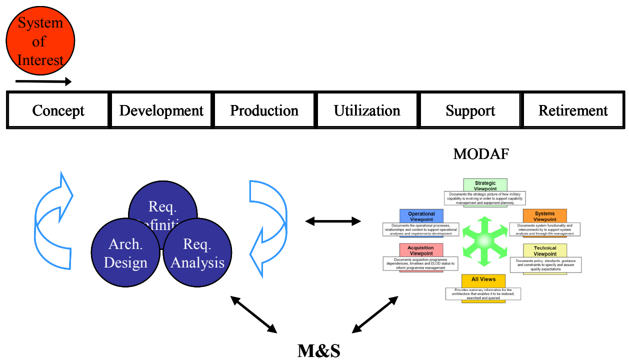

Figure 10. *The interplay between SOI, ISO/IEC 15288, MODAF and M&S.*

MODAF, or any other architectural framework, can support this work by providing suitable views to describe different aspects of the system. The operational viewpoint and its views can be used to describe the operational picture in which the system will operate. The systems viewpoint and its views can be used to describe the alternative concepts. Together with the traditional descriptions of user needs, system requirements and physical suggestions of system solutions (prototypes), the MODAF views constitute an important basis for systems development.

Modelling and simulation can contribute to system development in several ways. Simulation of the High Level Operating Concept models can contribute to a deeper understanding of the big picture and the requirements this puts on the System of Interest (SOI). Executable models of the High Level Operating Concept can also serve as a communication tool for different stakeholders to express their needs. An executable model depicting the big picture can also be used to try out alternative concepts; a prototype can for example be tested in this virtual operational environment to evaluate to what extent the suggested solutions fulfils the stakeholders needs. Other models within the operational viewpoint can be executed to verify and validate the models as well as increase the understanding; executable models can be developed to illustrate and analyse information exchange and activities.

The same goes for the systems viewpoint; executable models can be developed to illustrate and analyse interfaces and data flow between system elements and to specify data characteristics exchanged between the system elements.

5 Preliminary Ideas Concerning Tool Support

During the work with ISO/IEC 15288, architectural frameworks and M&S the ideas concerning a tool support started to evolve.

One tool that can be of help in development of executable models for the High Level Operating Concept is the Scenario and Activity Tool (SAT), illustrated in Figure 11 below.

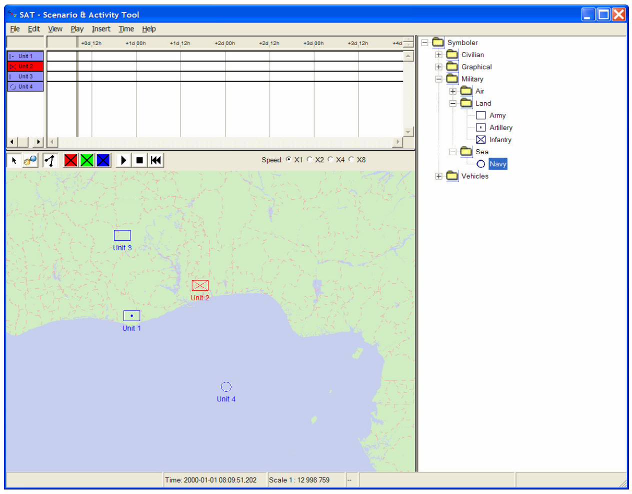

Figure 11. *Scenario and Activity Tool (SAT).*

SAT is used to model and visualise dynamic scenarios, and helps the user to build models consisting of objects and events. SAT can then visualise movement of objects, interaction between objects and when the objects are visual. SAT aims at describing scenarios quick and easy, the dynamic scenarios can then be used for further discussions, presentations and analysis.

Currently a new tool (SATAFSIM) is developed based on the functionality in SAT. In SATAFSIM, it will be possible to simulate events, such as operational activities and operational information exchange. SATAFSIM will also be integrated with a tool that manage architectural framework, in this case System Architect®. The idea is that SATAFSIM can be used to develop the views defined in an architectural frameworks (e.g. MODAF) operational viewpoint. The information will then be transferred to System Architect where it will be stored. From System Architect the information can then be further developed.

6 References

[1] NATO: NATO System Life Cycle Stages and Processes, AAP 48 Draft 0.9 –Draft 2, November 2005.

[2] International Organization for Standardization: ISO/IEC 15288 Systems engineering — System life cycle processes, First edition (ISO/IEC 15288:2002(E)), Switzerland 2002.

[3] US Department of Defence: “Memorandum, Subject: DoD Architecture Coordination Council (ACC)”, 14 January 1997.

[4] National Research Council, “A Review of the FBI’s Trilogy Information Technology Modernization Program”, Computer Science and Telecommunications Board, National Academies Press, Washington, D.C., 2004

[5] Bailey, I., Kihlström, L-O., Hagenbo, M., Hozhabrafkan, F., Olaussen, S. M: “Extending the NATO Architectural Framework to Represent Service-Oriented Architectures”, to be presented at Civil & Militär beredskap (CIMI), Enköping 2006.

[6] NATO C3 Board (AC/322): NATO C3 System Architecture Framework (NAF) version 2, 2004.

[7] DoD Architecture Framework working group: “DoD Architecture Framework version 1.0”, USA 2004.

[8] Ministry of Defence: MOD Architectural Framework Technical Handbook version 1.0, UK 2005.

[9] Hagenbo, M: “Enterprise Architecture brief presented to Australian DOD 24th of March 2006”. 2006

[10] Swedish Defence Material Administration: “FM Arkitekturramverk, Swedish Armed Forces Architecture Framework” v. 016, 2001.

[11] National Research Council: “Modeling and Simulation in Manufacturing and Defence Systems Acquisition”, National Academy press, Washington, D.C. 2002.

[12] Nordqvist, T and Gustafsson, M-L: “Introduction to Simulation Based Acquistion – a Swedish perspective” (in Swedish eds.), Linköping 2006.

7 Author Biographies

Dr. TOMMY NORDQVIST is the managing director of Front End Strategy AB, a Swedish Consultancy Company focused on Business Analysis, Requirements Engineering, Systems Design, and M&S. Dr. Nordqvist has been a researcher at the Swedish Defence Research Agency where he conducted research in the field of Requirements Engineering, Systems Engineering and Decision Support Systems. He has also been a consultant at AeroTech Telub, a Swedish consultant company, where he, among other things, participated in the requirements engineering work aimed at developing a simulator for training of personnel in a C2 system for air defence. After AeroTech Telub, Dr. Nordqvist held a position as method expert, focused on Software Engineering, at the Swedish Defence Material Administration (FMV). Dr. Nordqvist has also participated in the work of developing the Federation Development and Execution Process (FEDEP). Dr. Nordqvist participates in the work at FMV to further develop their Systems Engineering processes.

ANNA EDBERG has a M.Sc. in Computer Science and works as a consultant at Front End Strategy AB. Currently she work with further development and adaptation of Simulation Based Acquisition according to Swedish circumstances, regarding both methods and tool support. She has also participated in development of a simulator, using FEDEP. Ms Edberg has previously worked in the county council of Östergötland with system development, and at Royal Melbourne Institute of Technology developing a tool support for the Prometheus Methodology; a methodology for developing intelligent agents.

HÅKAN BYSTRÖM, M.Sc. CS&E and Ret. Maj. Swedish Armed Forces (SwAF), works as a consultant at Front End Strategy AB. Mr Byström works with operational analysis, M&S and Simulation Based Acquisition. He has previously worked with architectural issues at the SwAF Headquarter. He has also worked with Network Centric Warfare and authorisation of IT systems during his time at the SwAF Headquarter,

MARIE-LOUISE GUSTAFSSON is a senior engineer and the project manager for SBA R&D Programme at the Swedish Defence Material Administration (FMV). Mrs. Gustafsson began her career working with air-to-air missile evaluations in support of acquisition in 1997. Mrs. Gustafsson has been involved in a wide variety of modelling, simulation, and analysis projects for different missile systems.