Defence Research and Development Canada | Recherche et développement pour la défense Canada

DEFENCE | DÉFENSE

A Vision for M&S Processes, Tools, and Standards for CapDEM

An Update by the Integrated Synthetic Environment Work Stream of CapDEM

Richard G. Brown and Fawzi Hassaïne

Defence R&D Canada – Ottawa

TECHNICAL MEMORANDUM

DRDC Ottawa TM 2004-133

April 2004

Canada

A Vision for M&S Processes, Tools, and Standards for CapDEM

An Update by the Integrated Synthetic Environment Work Stream of CapDEM

Richard G. Brown
Defence R&D Canada – Ottawa

Fawzi Hassaïne
Defence R&D Canada – Ottawa

Defence R&D Canada – Ottawa

Technical Memorandum

DRDC Ottawa TM 2004-133

April 2004

Abstract

The Department of National Defence (DND) wants to greatly reduce the life-cycle costs and procurement times for new systems and to ensure that new systems will interoperate with existing ones. To do so, the concept of *capability engineering* has emerged. It will help to quantify what capability is required, evaluate options to achieve it, procure the best system, and manage the system throughout its life cycle.

The Collaborative Capability Definition, Engineering and Management (CapDEM) Technology Demonstration Project (TDP) is developing the processes and tools to enable capability engineering within DND. The processes are expected to rely heavily on Modelling and Simulation (M&S) and so an Integrated Synthetic Environment (ISE) work stream was created. Its mandate is to identify the processes, tools, and standards that are necessary to support various M&S activities within CapDEM.

This report discusses the M&S requirements of capability engineering; the need for a comprehensive M&S Development and Execution Process (MASDEP) and an integrated software development methodology; the need for a common modelling language to be used throughout; current M&S standards, including several under development; and a vision of the processes, tools, and standards to be developed.

Résumé

Le Ministère de la Défense Nationale (MDN) s’intéresse à la réduction du coût ainsi que du temps nécessaires impartis à l’acquisition de nouveaux systèmes, et pour assurer un fonctionnement avec les systèmes existants. Dans ce contexte, un nouveau concept a émergé, appelé *Ingénierie des Capacités*. Ce dernier aidera à évaluer les capacités, à juger entre de multiples options pour achever une capacité, à l’acquisition du meilleur système, ainsi que de sa gestion durant son cycle de vie.

Le Programme de Démonstration de Technologies (PDT), Définition, Ingénierie, et Gestion des Capacités (DIGCap), voit à développer les processus et outils pour achever l’ingénierie des capacités au sein du MDN. Ces processus reposent fortement sur l’usage de la Modélisation et de la Simulation (M&S), ce qui a conduit à la création d’une équipe pour les Environnements Synthétiques Intégrés (ESI). Celle-ci s’est défini comme mandat l’identification des processus, outils, ainsi que les standards nécessaires au support des multiples activités de M&S qui ont lieu au sein de DIGCap.

Ce rapport examine les exigences de M&S pour l’ingénierie des capacités, et le besoin d’un Processus pour le Développement et l’Exécution pour la M&S (MASDEP). Ce rapport aborde aussi la notion d’une méthodologie intégrée de développement de logiciels, du besoin d’un langage de modélisation commun, de la nécessité des standards, incluant ceux encore en développement, et enfin d’une vision globale incluant les processus, outils et standards à être développés.

Executive Summary

The Department of National Defence (DND) wants to greatly reduce the life-cycle costs of military systems and their procurement times. In addition, it wishes to ensure that all of its systems will complement each other and work together efficiently. To achieve these goals, a new planning and procurement approach is sought—one that focuses on the development of military capabilities and not just purchasing another new system. The envisaged approach is referred to as *capability engineering*.

In the fall of 2002, the Future Forces Synthetic Environments Section (FFSE) at Defence Research and Development Canada (DRDC) Ottawa was awarded the Collaborative Capability Definition, Engineering and Management (CapDEM) Technology Demonstration Project (TDP). Its mandate is to deliver a prototype infrastructure (process and tools) that will enable capability engineering within DND.

The infrastructure is expected to rely heavily on Modelling and Simulation (M&S) to visualize concepts, analyze options, develop and test systems, support maintenance and training, etc. As a result, the Integrated Synthetic Environment (ISE) work stream was created. Its main objective is develop the process, tools and standards that are necessary to support the M&S requirements of capability engineering.

This report discusses the M&S requirements of capability engineering and how specific simulation requirements must be determined by CapDEM's overarching "systems of systems" Capability Engineering Process (CEP). Given the requirements, a comprehensive M&S Development and Execution Process (MASDEP) is required. The MASDEP for CapDEM will be similar in spirit to the well-known High Level Architecture (HLA) Federation Development and Execution Process (FEDEP), but it will interface with the CEP; address software development issues in order to ensure that models and software have long-term reusability; and include verification, validation, and accreditation (VV&A), and other "overlay" processes.

The MASDEP will need a modelling language with well-defined semantics to ensure that models and systems can be represented in an unambiguous manner. At present, the Unified Modeling Language (UML) is the most likely technique to be adopted, given that it is an open standard and it now has extensions for systems engineering. The MASDEP must also be compatible with other M&S standards (reviewed) but not be tied to any one because so many tend to be short-lived.

The MASDEP for CapDEM will likely be based on the Synthetic Environment Development and Exploitation Process (SEDEP) that was developed as part of the European Co-operation for the Long-term In Defence (EUCLID) Research and Technology Program (RTP) 11.13. Once the SEDEP tool suite has been evaluated and other VV&A studies have been completed, the ISE work stream will design and implement a MASDEP for CapDEM.

Brown, R.G., Hassaïne, F. 2004. A Vision for M&S Processes, Tools, and Standards for CapDEM. DRDC Ottawa TM 2004-133. Defence R&D Canada – Ottawa.

This page intentionally left blank.

Sommaire

Le Ministère de la Défense Nationale (MDN) s'intéresse à la réduction des coûts encourus par le maintien des systèmes militaires durant leur cycle de vie, ainsi que du temps nécessaire pour leur acquisition. Le MDN veut également s'assurer que ces systèmes se complémentent, et fonctionnent ensemble de manière efficace. Pour arriver à cette fin, une nouvelle approche pour la planification et l'acquisition est recherchée. Cette dernière devrait mettre l'accent sur le développement de capacités militaires, et non pas seulement sur l'achat de nouveaux systèmes. Cette approche est appelée *Ingénierie des Capacités*.

Durant l'automne 2002, la section des Environnements Synthétiques des Futures Forces (FFSE), à Recherche et Développement pour la Défense Canada (RDDC) s'est vu attribuée le Programme de Démonstration de Technologies (PDT), Définition, Ingénierie, et Gestion des Capacités (DIGCap). Son mandat est de délivrer une infrastructure prototype en terme de processus et outils, et qui permettra l'ingénierie des capacités au sein du MDN.

Cette infrastructure devrait reposer grandement sur la Modélisation et la Simulation (M&S) afin de visualiser les concepts, d'analyser les options, de développer et de tester les systèmes, d'assurer leur maintenance ainsi que l'instruction des opérateurs, etc. Cela a abouti à la création de l'équipe Environnements Synthétiques Intégrés (ESI). Celle-ci s'est vue assigné comme premier objectif le développement de processus, outils, et standards nécessaires au support des exigences de M&S pour l'ingénierie des capacités.

Ce rapport examine les exigences de M&S pour l'ingénierie des capacités, et s'intéresse aux exigences spécifiques de simulation. Ces dernières sont déterminées par le Processus d'Ingénierie des Capacités (PEC) de DIGCap, et sa vision globale en termes de « Systèmes des Systèmes ». Étant donné ces exigences de simulation, un Processus pour le Développement et l'Exécution pour la M&S (MASDEP) est requis. Ce dernier est similaire, dans l'esprit, au processus FEDEP (Federation Development and Execution Process) de HLA (High Level Architecture), et s'interfacera au processus PEC. Entre autres tâches, le MASDEP s'occupera des questions de développement de logiciels afin d'assurer que modèles et logiciels ont été conçus pour être réutilisables, et pour inclure des processus de recouvrement (overlay) comme celui de la vérification, validation et accréditation (VV&A).

Il est impératif d'identifier un langage de modélisation à être utilisé par le MASDEP, et qui aura une sémantique bien définie, afin de représenter les modèles de manière non équivoques. UML (Unified Modeling Language) représente actuellement le candidat le plus susceptible d'être adopté, étant donné qu'il est reconnu comme étant un standard, et qu'il bénéficie d'extension pour l'ingénierie des systèmes. D'un autre côté, le MASDEP doit pouvoir être compatible avec d'autres standards de M&S, mais sans être forcément lié à aucun en particulier, considérant la courte durée de vie de certains standards.

Le MASDEP pour DIGCap sera plus vraisemblablement basé sur le SEDEP (Synthetic Environment Development and Exploitation Process), qui est le résultat du Programme de Recherche et Technologie (RTP) 11.13 européen EUCLID (European Co-operation for the Long-term In Defence). Une fois les outils de SEDEP évalués, et les études de VV&A complétées, l'équipe ESI concevra et implémentera un processus MASDEP pour DIGCap.

Brown, R.G., Hassaïne, F. 2004. A Vision for M&S Processes, Tools, and Standards for CapDEM. DRDC Ottawa TM 2004-133. R & D pour la défense Canada – Ottawa.

Table of Contents

Abstract i

Résumé ii

Executive Summary iii

Sommaire v

Table of Contents vii

List of Figures ix

1. Introduction 1
   1.1 CapDEM TDP 1
   1.2 Relationship to the Collaborative Engineering Process 2
   1.3 Report Organization 3

2. Modelling and Simulation Processes for Capability Engineering 5
   2.1 The M&S Requirements of Capability Engineering 5
   2.2 The High Level Architecture 6
   2.3 MASDEP Definition 7
   2.4 Example MASDEPs 8
   2.5 Benefits and Limitations of MASDEPs 11
   2.6 MASDEP-Related Processes 11
      2.6.1 FEDEP Checklists 12
      2.6.2 Verification Validation and Accreditation “Overlays” 12
      2.6.3 Configuration Management 14
      2.6.4 Security 14

3. Software Development Considerations 17
   3.1 Software Development Methodologies 17
   3.2 Relationship of SDMs to MASDEPs 18

4. Unified Modeling Language 21

4.1 UML for Software Engineering 21
4.2 UML V2.0 for Software and Systems Engineering 26
4.3 Systems Modeling Language 27
4.4 Significance of UML for the ISE Work Stream 28

5. Other Relevant Standards 29
5.1 Model Driven Architecture 29
5.2 Simulation Reference Markup Language 31
5.3 Base Object Model 31
5.4 eXtensible Modeling and Simulation Framework 31
5.5 eXtensible Markup Language Metadata Interface 32
5.6 Significance for the ISE Work Stream 32

6. A Vision for the MASDEP, Tools, and Standards 35
6.1 M&S-Related Issues 35
6.2 A Vision for the Process, Tools, and Standards 36
6.3 The Road Ahead 38

7. Conclusions 41

References 43

List of Abbreviations 45

List of Figures

Figure 1. Relationship between the CEP and ISE work streams. 2

Figure 2. The DMSO HLA FEDEP.. 8

Figure 3. IEEE 1516.3 FEDEP 9

Figure 4. EUCLID RTP 11.13 SEDEP (High level steps only) 10

Figure 5. The HLA FEDEP with the VV&A 7-Step Process Model Overlay.......................... 13

Figure 6. Use Case Diagram.. 23

Figure 7. Class Diagram 23

Figure 8. State Diagram... 23

Figure 9. Activity Diagram... 24

Figure 10. Sequence Diagram 24

Figure 11. Component Diagram 25

Figure 12. Deployment Diagram 25

Figure 13. The OMG Model Driven Architecture.. 30

This page intentionally left blank.

1. Introduction

1.1 CapDEM TDP

The Department of National Defence (DND) wants to greatly reduce the life-cycle costs of military systems and the length of time that it takes to procure them. In addition, it wishes to ensure that all of its systems will complement each other and work together efficiently. To achieve these goals, major changes are required in the procurement process, which now focuses on the procurement of systems with insufficient planning regarding how they will interact with others already in place.

An alternative planning and procurement process is sought, that is, one that focuses on the development of military capabilities—the latter being defined (in this report) as the ability to achieve some goal with a certain level of performance. The vision for the process is to start with a rigorous analysis of what level or degree of a desirable capability is required and what systems are capable of providing it. For instance, a capability might be defined as the ability to identify all ships off the east coast of Canada. Possible systems that can provide such a capability are upgraded maritime patrol aircraft, unmanned aerial vehicles, satellites, etc. Once candidate systems are identified, they can be analyzed and compared in order to identify the preferred solution.

The envisaged planning and procurement process will first quantify capability needs and use the results to compare the efficacy of competing systems. The results will then be used to help procure a real system. A spiral *capability engineering* process will be used within a "system of systems" (SoS) construct to continually verify and refine assumptions that were made in previous iterations. The process will also help to ensure that whatever system is delivered meets expectations regarding its performance, life-cycle costs, interoperability with existing systems, etc.

As part of Defence Research and Development Canada (DRDC) efforts to implement a capability engineering process, the Future Forces Synthetic Environments Section (FFSE) at DRDC – Ottawa was awarded the Collaborative Capability Definition, Engineering and Management (CapDEM) Technology Demonstration Project (TDP) in the fall of 2003. Its main objective is to deliver a prototype infrastructure (process and tools) that will enable capability engineering within DND.

The infrastructure is expected to rely heavily on Modelling and Simulation (M&S) and so an Integrated Synthetic Environment (ISE) work stream was created within CapDEM. Its main objective is to develop the process, tools and standards that are necessary to support the M&S requirements of capability engineering. A vision for the outcome of the work stream is the subject of this report.

1.2 Relationship to the Collaborative Engineering Process

To fully appreciate the need for the ISE work stream and its objectives, its linkage with another CapDEM work stream needs to be understood.

The ISE work stream is directly linked to the Capability Engineering Process (CEP), which is responsible for the overarching and iterative SoS engineering process in CapDEM. The relationship between the CEP and the ISE work stream is depicted in Figure 1.

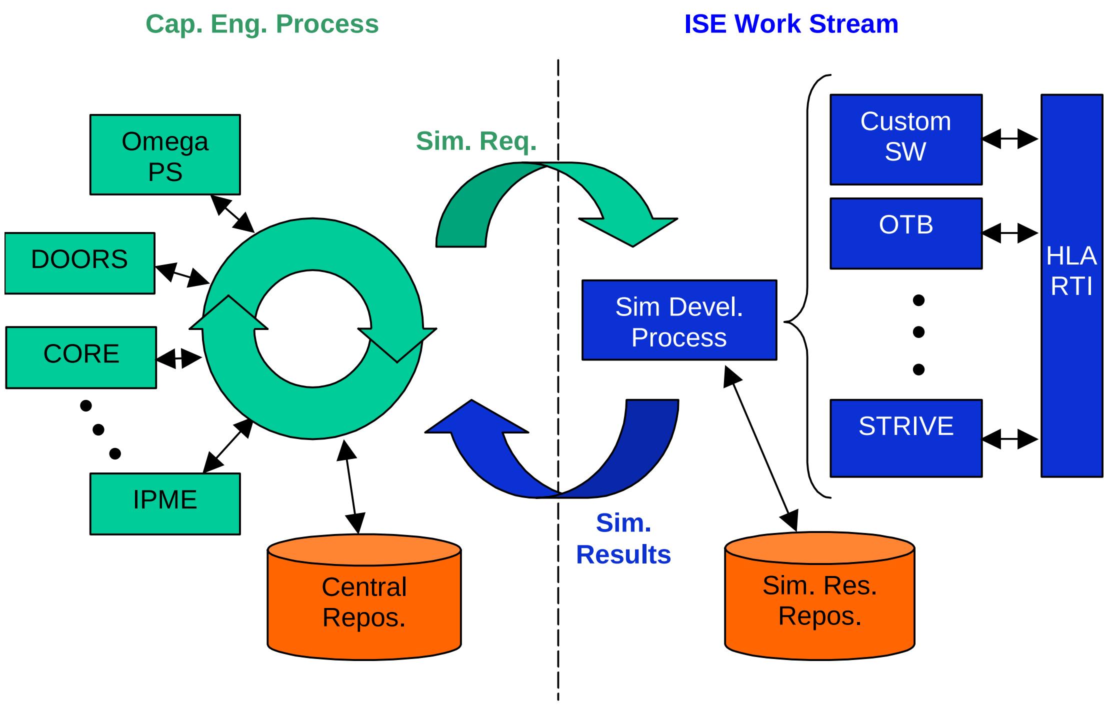

Figure 1. Relationship between the CEP and ISE work streams.

The CEP's SoS engineering process has its own set of software tools, such as CORE and IPME, that have built-in M&S capabilities; however, the tools are limited in ways. For instance, they are not designed to incorporate large terrain databases or very realistic three-dimensional (3D) computer graphics, nor do they include facilities for interoperating with any other simulation or live system using recognized standards (discussed in later sections). In addition, the persons using the CEP tools are not expected to be specialists in simulation technology (a specialty in its own right), or to have significant software development experience, which is necessary for building large simulations.

To remove these limitations, when the M&S capabilities of the CEP tools need to be exceeded, M&S requirements for arbitrarily complex and distributed simulations can be generated and passed to the ISE work stream. Once the simulation is built and executed using applications such as OTB or custom software, the results are returned to the CEP.

1.3 Report Organization

The remainder of this report is organized as follows.

- Chapter 2 describes the need for a simulation development process that will define how M&S should be performed as part of the overarching CEP and how the two processes need to be linked.

- Chapter 3 describes the role of software (SW) development in the M&S process and how an appropriate software development environment (SDE) can enable model reusability, traceable simulation results, and other highly desirable features.

- Chapter 4 describes the need for a common nomenclature for describing models, simulation systems, software structure, etc. The Unified Modeling Language (UML), a software design language, is described as a potential solution.

- Chapter 5 describes other relevant standards that will be key in future M&S efforts, even though some of them are still being developed by standardization organizations.

- Chapter 6 describes the vision for the ISE work stream based on the processes, software tools, and standards described in previous chapters.

- Chapter 7 concludes the report.

This page intentionally left blank.

2. Modelling and Simulation Processes for Capability Engineering

Section 1 indicated how the ISE work stream is developing a M&S capability that will greatly enhance that of the CEP SoS engineering process. The ISE M&S capability will be based on existing processes, tools, and standards that are found to be appropriate for CapDEM in terms of their relevance, availability, cost, etc. This section of the report focuses on M&S development processes that have been standardized by various organizations and widely adopted in the US and Europe.

2.1 The M&S Requirements of Capability Engineering

Capability engineering has a number of requirements that are directly linked to M&S technology. These requirements include:

- **Visualization:** People better appreciate the benefits and limitations of a concept when they have the opportunity to visualize it through the use of animations. Since real prototypes are expensive to build and test, animations based on high- (or low-) fidelity computer models can be used instead at much lower cost. Further, scenarios that cannot be tested for real (because of the risks to property or persons involved) can be safely tested in simulated environments.

- **Options analysis:** Capability engineering will require many options to be tested. M&S readily supports such analysis in a relatively short period of time. While costs and time may limit live trials to a very few experiments, computer-based models can usually be run repeatedly to test various options and with statistically meaningful results.

- **Scientific rigour:** To date, many capabilities are developed based on the opinions of experienced personnel. While past experience is often invaluable to future planning, capability engineering should also enable decisions to be based on scientific principles that can prove (within model accuracy) that decisions made result in optimized systems. M&S is invaluable for performing such analysis, especially when systems are large and complex, and their development has to be coordinated with other systems already in place.

- **Traceability:** Capability engineering will require traceable models, decisions, designs, etc. If a concept is developed using M&S and proper configuration management is used to capture the models used on a regular basis, the end results become traceable back to the original design and requirements. Assumptions can be verified along the way and corrective action taken as necessary.

- **Improved transition:** Capability engineering will likely be used on occasion throughout the entire life cycle of a system, especially those that are upgraded periodically or are used to help define future system requirements. Since the transition from one version of a system to another should be as smooth as possible, M&S can help by providing models that are reusable and extensible. Variations on a single model can also support related activities such as training, maintenance, upgrade planning, etc.

The above benefits indicate that M&S can have a positive impact on how a capability is planned and implemented. To do so, however, M&S must be performed in a proper way using appropriate standards. The key is a guiding M&S development and execution process, which is supported by an appropriate tool suite and standards.

The next section discusses a well-known approach to M&S development. While not originally intended to support capability engineering, it provides excellent background material for later discussions.

2.2 The High Level Architecture

The United States Department of Defense (DoD) High Level Architecture (HLA) [1] for M&S was developed under supervision of the Defense Modeling and Simulation Office (DMSO). The main objectives of the HLA were to increase interoperability among simulations and to promote reuse of simulations and their components.

After going through several development iterations, the HLA was defined by a series of documents that specified:

- Basic concepts and a set of rules that defined general principles to be met.

- The Federation Development and Execution Process (FEDEP), which is a six-step process for developing distributed simulations.[1] Note that a federation is a set of federates (individual simulations, service applications, and interfaces to live systems) that has been assembled to meet the needs of some simulations.

- The Federate Interface Specification, which specifies the interface between federates and a run-time infrastructure (RTI), the latter being the software that manages data distribution between federates and provides various services to them, such as simulation time management.

- The HLA Object Model Template (OMT) Specification, which defines the format for specifying a federation object model (FOM). A FOM specifies the data that will be exchanged between federates at run-time. Every federate in a federation must use the same FOM, but every federation may have a different FOM.

Although the HLA documents collectively specify the necessary technical foundation for developing distributed, interoperable simulations, they do not define how to develop federations in sufficient detail for everyday use.

[1] The version of the FEDEP that was developed by the Institute of Electrical and Electronics Engineers has seven steps; it is discussed later in the report.

In order to provide more practical guidance to the HLA user community, a more detailed process is required to lead and supervise the development and execution of HLA simulations. As with the HLA, it should cover all aspects of the process, starting from the analysis of user requirements and ending with the analysis and reporting of simulation data.

Another key aspect of the more detailed process that has not been elaborated in previous documents or standards, including the HLA, is the methodology used to develop and implement the models that are at the heart of simulation applications. The authors of this report believe that defining such a methodology is key to achieving simulation interoperability and reuse, just as much as the issues that were addressed by the HLA. This belief is based on observations that HLA simulations are frequently not compatible, reusable, or extensible despite being “HLA compliant”. A root cause is the underlying models were developed for a very specific purpose and future reuse and extensibility were not considered when the models were developed.

Since alternative processes to the HLA FEDEP will be discussed in subsequent sections, a general term is required to refer to any process that deals with simulation development, regardless of whether or not it is distributed. In this report, the term Modelling and Simulation Development and Execution Process (MASDEP) will be used.

The following section presents the definition of a MASDEP, followed by examples of such a process. One particular MASDEP use case will be elaborated, and this process will be shown to serve as a common framework to articulate other processes. The latter have been designed in the form of overlay processes to the elaborated MASDEP, and address related issues such as verification, validation and accreditation (VV&A), configuration management (CM), security, etc.

2.3 MASDEP Definition

A MASDEP is a process that defines all of the steps for the development and execution of a simulation. It can be seen as a systems engineering process model that defines a framework for simulation development, distributed or otherwise. To be applicable to the broadest range of simulation development projects, a MASDEP should not be dedicated to the HLA or any other standard; on the other hand, a MASDEP can be dedicated in order to elaborate on issues specific to a particular standard, such as the HLA.

A MASDEP aims at driving simulation developers in their tasks, from understanding user requirements and translating them into a more formalized representation, through the development of the (mathematical) model(s) representing the simulated object/scenario, its execution, and the collection and analysis of simulation data.

A MASDEP should definitely not be confused with a software development methodology (SDM) such as the Rational Unified Process (RUP) [2], which is marketed by IBM Corp. However, a SDM should be a key part of a MASDEP because it will define many important lower-level process details. These include:

- how software should be designed and developed from model specifications
- how individual simulation applications should be tested
- how groups of simulation applications should be integrated and tested
- how software (and the models they are based on) should be under configuration control

2.4 Example MASDEPs

Several MASDEPs have been designed. The most popular is certainly the HLA FEDEP [3], which was developed by DMSO. The process, shown in Figure 2, is composed of six generic steps; each step is then tailored to meet the needs of individual simulations.

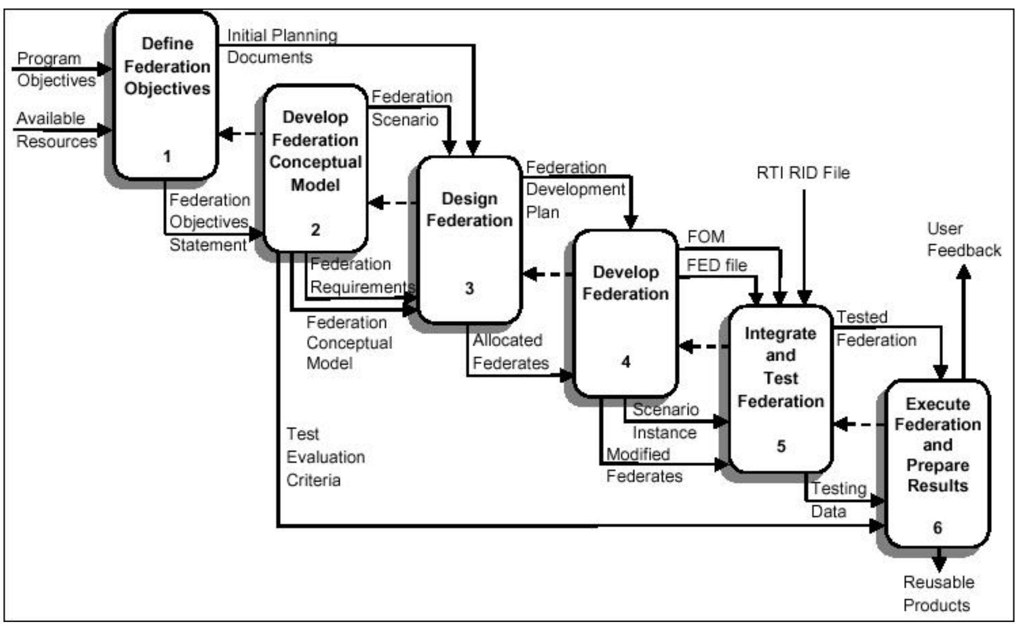

Figure 2. The DMSO HLA FEDEP

The Institute of Electrical and Electronics Engineers (IEEE) also established its own version of the HLA as IEEE Standard 1516 [4]. Its version of the FEDEP was established as a *standard of recommended practice* (IEEE 1516.3) in March 2003 [5]. Figure 3 presents the seven-step process. The main difference between the two FEDEPs is the IEEE's addition of a new final step, called "*Analyze Data and Evaluate Results*".

Another initiative, the European Co-operation for the Long-term In Defence (EUCLID) Research and Technology Program (RTP) 11.13, completed the development of its own MASDEP in late 2003. In order to promote the use of Synthetic Environments (SEs), the participating nations and industries attempted to overcome significant limitations of the FEDEP (discussed in section 2.5) by developing the Synthetic Environment Development Environment (SEDE), which consists of a MASDEP and a supporting tool suite. The MASDEP was called the Synthetic Environment Development and Exploitation Process (SEDEP) [6]; it is shown in Figure 4.

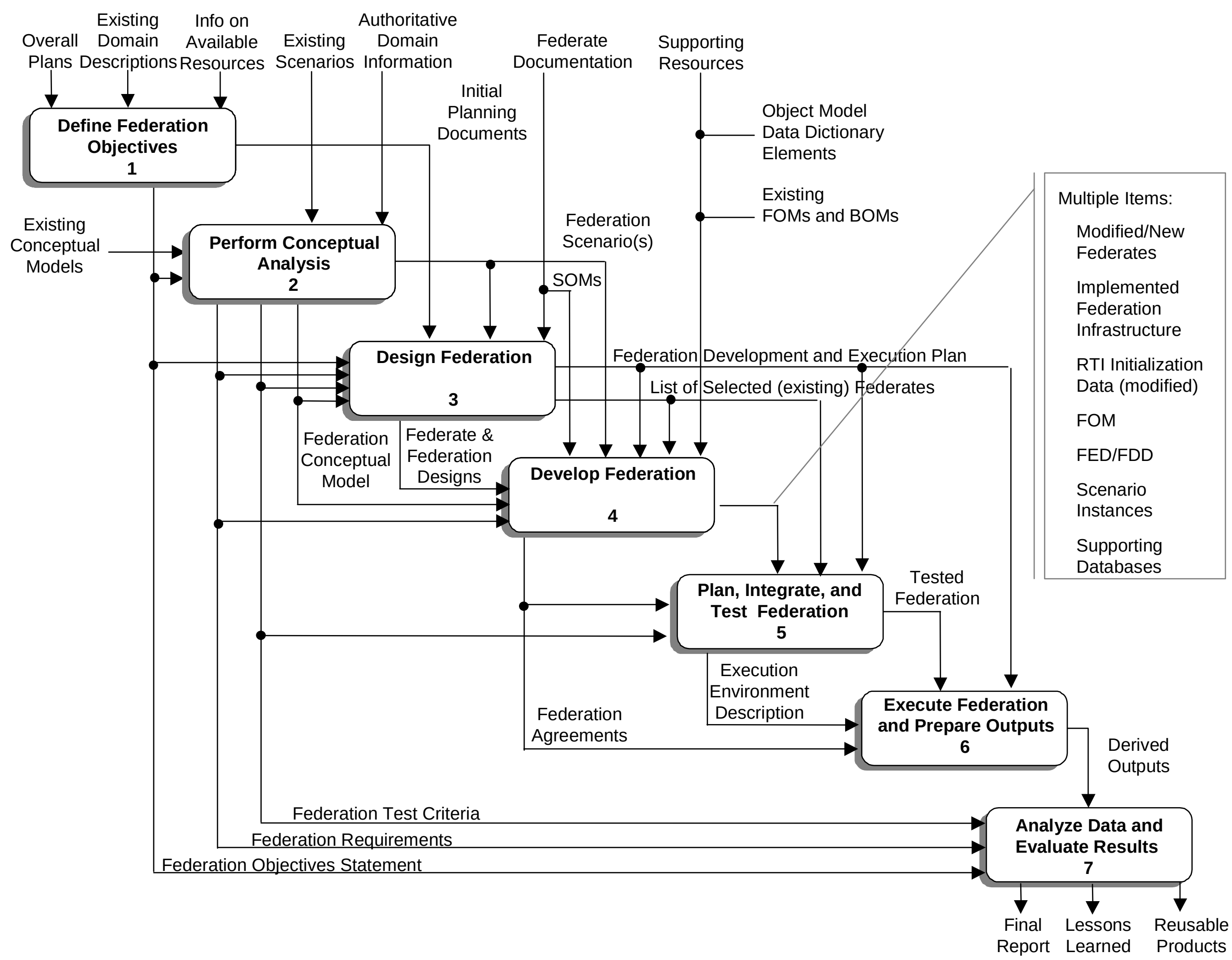

Figure 3. IEEE 1516.3 FEDEP

The key features of the SEDEP are as follows:

- The SEDEP provides data formats that make SE-relevant information more accessible to an SE practitioner. In particular, it defines the inputs and outputs for each step and activity of the process, and it explicitly identifies the different library components in use by the different steps and activities.

- When compared to the DMSO FEDEP, it introduces two new steps dedicated to:
    a. Supporting the users in determining the suitability of a SE in solving their problem and estimating project parameters such as cost, duration, risk, etc.
    b. Analysing the outputs from a simulation's execution, and evaluating the results.

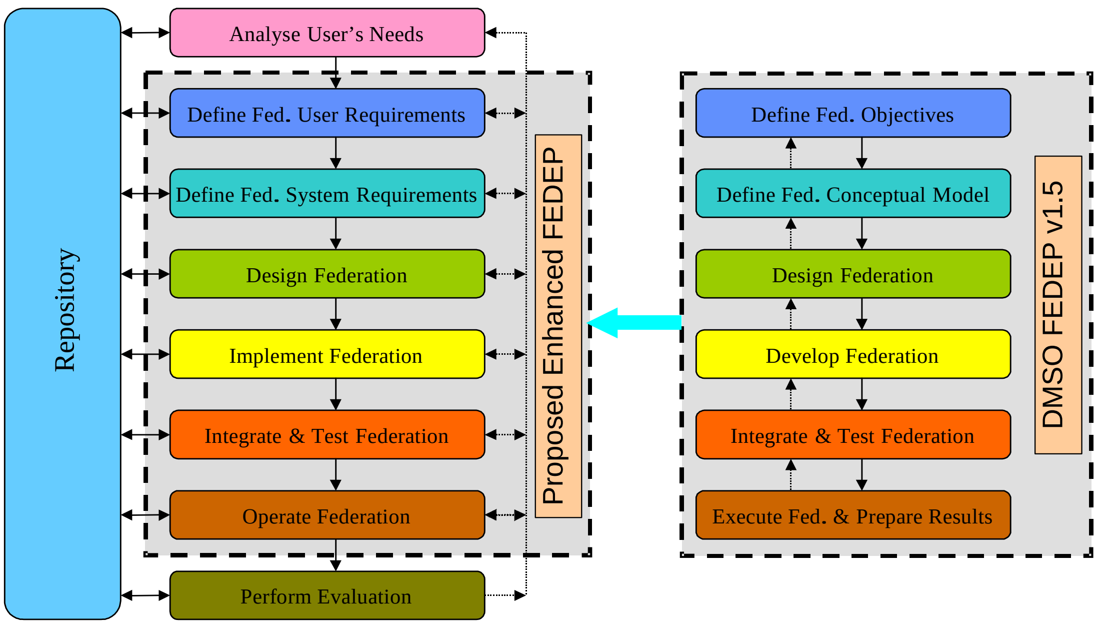

Figure 4. EUCLID RTP 11.13 SEDEP (High level steps only)

- It extends the FEDEP's overlay representation to provide guidance on many other activities that span across all the SEDEP steps. In particular, it incorporates relevant activities from Capability Maturity Models [7] to promote good systems engineering practices.

- It generalises the wording and definitions so as not to be limited to HLA technology.

- It supports the composition of federates from components, as well as the reuse of federations, federates and their components.

- It specifies the use of a repository for enabling support tools to share data. The repository is also used as a source for the tools; SE developers simply download whatever tools they require.

2.5 Benefits and Limitations of MASDEPs

The main benefit acquired from the use of a MASDEP is the order that it brings to the development of what is typically a large and complex, distributed software system. Ideally, such a process provides all the guidance required by federation developers using a systems engineering process. It should promote the use of spiral development involving phases of requirements analysis, conceptual modelling, design, software development, integration, execution and results analysis.

In addition, a MASDEP like the IEEE 1516.3 FEDEP is a recognized standard rather than a commercial product. Hence, it is developed and continually improved based on feedback and new ideas from experts with a broad range of civilian and military experience.

Finally, the widespread adoption of the DMSO and IEEE versions of the FEDEP has proven that organizations consider their use worthwhile. With time, the relatively new SEDEP may also be widely adopted (depending on how quickly a number of intellectual property issues can be addressed).

Still, MASDEPs have their weaknesses. Limitations include:

- They are fairly generic processes; details are not provided on how to execute them.
- They offer a conceptual evolutionary sequence without specifying mechanisms for transitioning from one step to another.
- They may not address reusability issues (for simulation components). To be more specific, the FEDEP does not but the SEDEP does to a limited extent.
- Configuration management is not addressed.
- Finally, no guidance is provided for any underlying VV&A effort.

2.6 MASDEP-Related Processes

As just discussed, MASDEPs have their own limitations concerning some important issues like software reuse, VV&A, configuration management, etc. As a result, many studies have tried to overcome these limitations by proposing other processes that would be linked to the MASDEP but generally have their own control flow. They are usually conceptualized as a “process overlay” that has functional blocks that match those in the MASDEP one-for-one; however, their functionality is usually dedicated to a separate but complementary activity.

The benefit resulting from having an overlay process is that simulation process developers have the ability to tailor the overlay to meet other process requirements, such as VV&A. The following subsections elaborate on some common overlays.

2.6.1 FEDEP Checklists

The FEDEP checklist [8] has been prescribed by DMSO as an accompanying overlay routine/process for the FEDEP. It covers all steps of the FEDEP by giving specific details that one must consider during a federation development. It was developed based on federation developers' cumulative experience, and, in general, it represents management and technical decisions as well as tasks that are acted upon by FEDEP participants. The checklists require that the appropriate participants be identified.

A specific application domain can also be part of this process, by ensuring that each step of the checklist defines domain-specific procedures to be applied. An example in the domain of VV&A is discussed next.

2.6.2 Verification Validation and Accreditation "Overlays"

Before VV&A overlays are explored, definitions related to VV&A are presented to ensure that subsequent discussions are understood as intended.

Verification and validation (V&V) may be defined as a way of measuring simulation credibility, whereas accreditation is the acceptance by the accreditation authority of a simulation for use in a particular context. More precisely, in the context of M&S, verification refers to the means by which a determination is made of how well a given set of requirements is captured throughout the design, implementation, testing, and documentation of a simulation. Validation measures how adequately the implementation conforms to the targeted model, or in other words, how close it is to the "real world".

Clearly, VV&A issues must be addressed when a simulation is being developed to ensure a high quality and successful simulation; thus, a MASDEP should address VV&A issues since it is providing guidance to the development process. Somewhat surprisingly, the DMSO and IEEE FEDEPs, which are the most widely adopted MASDEPs, do not.

To address this deficiency, numerous researchers have suggested means of incorporating VV&A in the FEDEP. The Boeing company's approach, proposed by V. Dobey and R. Lewis [9], uses an overlay to the IEEE 1516.3 (FEDEP) seven-step process, as illustrated in Figure 5.

The alignment of the VV&A process over the FEDEP allows:

- the VV&A and the FEDEP to progress together
- the processes to end up with the same objectives

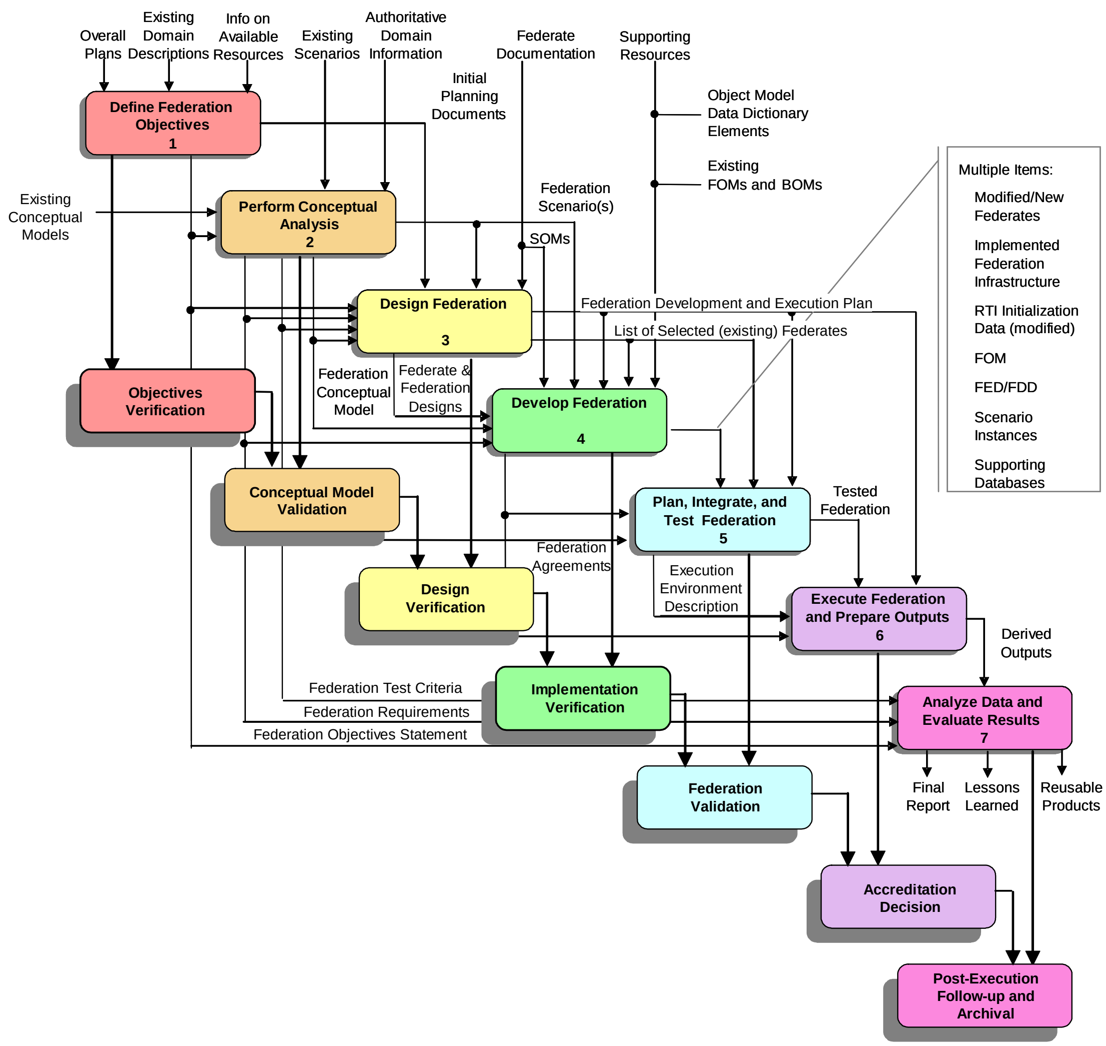

Figure 5. The HLA FEDEP with the VV&A 7-Step Process Model Overlay[2]

- for reusability as both processes share many development tools and data

Dobey and Lewis also give special attention to accreditation. It is no longer seen as a "one step event" at the end of the development process that evaluates system adequacy for a specific purpose. Instead, the study promotes an accreditation effort that would start at the early stages by looking at the federation objectives. It would accept the federation if it met its intended initial purpose and reject it otherwise.

Another VV&A process of interest to the ISE work stream is the risk-based user-focused VV&A process being developed by the DND Synthetic Environment Working Group (SEWG) and the DND Synthetic Environment Coordination Office

[2] This figure is an expansion of the detailed diagram of the IEEE 1516.3™ FEDEP process.

(SECO) [10]. Its applicability to a MASDEP for CapDEM will be determined in the future as part of on-going ISE work stream efforts.

2.6.3 Configuration Management

Although CM is included in some VV&A processes, it can also be implemented as a separate overlay.

CM is usually responsible for performing version control of software and management of all aspects of baseline product. The latter includes strict management of M&S requirements, including additions, refinements, and other changes that are inevitable during the development and life cycle of a simulation. CM also includes information about the supporting environment in term of used software, hardware, and data.

Another major step where CM is important is during the development of the conceptual model for the simulation, since changes in the requirements need to be reflected in the conceptual model. Unfortunately, last minute requirement changes may be (and often are) implemented in simulation software without the conceptual model being updated. As a result, the conceptual model and software become unsynchronized which impacts their reusability. The traceability of user requirements through to simulation results is also adversely affected (that is, lost). The end product could then not match the intended model, nor its underlying requirements.

2.6.4 Security

Due to operational requirements, simulation data may need to be exchanged between classified and unclassified security domains. A study by Filsinger and Lubbes [11] proposes the following approach for dealing with security considerations in such simulations:

- HLA must allow processing of Multi-Level Security (MLS) data among federates that are operated by users who do not have the appropriate security clearances for all aspects of the simulation.

- Information must be prevented from leaking from high- to low-level security domains.

- HLA must support processes that are capable of protecting information within a security classification and support processes that can be trusted to downgrade (sanitize) data.

- HLA must support security mechanisms to allow object ownership and object attributes to be safely read or updated by any simulation with a federation.

- HLA must support enforcement of mandatory confidentiality, integrity, and need-to-know policies.

- Simulations must be reusable at different security levels at different times in different federations.

Although security issues for the SEDEP and the FEDEP have received attention in previous studies, the issues have not been resolved to the extent described above. EUCLID RTP 11.13 studied security issues related to the SEDEP, while DMSO proposed a separate overlay process called the Federation Security Process (HLA FSP) for the FEDEP [12]. The latter refers explicitly to security and certification processes of the US DoD and the National Security Telecommunications and Information Systems Security Committee (NSTISSC). However, HLA federation developers outside of the US may be able to replace these processes by comparable ones from their own country.

This page intentionally left blank.

3. Software Development Considerations

Previous sections presented the usefulness of a MASDEP and some current examples. Since M&S relies very heavily on specialized software, one might expect MASDEPs to address at least some SW development issues. For the most part, such is not the case.

SW development is often considered a low-level activity that does not need to be considered in any detail in MASDEPs. Not only is SW development not described in detail in existing MASDEPs, its existence as an activity is usually only implied by higher-level process steps that have labels such as “Develop Federation”. In contrast, the authors of this report believe that Software Development Methodologies (SDMs) should be a key part of any MASDEP and described in detail.

The remainder of this section describes SDM background information, the benefits of relating SDMs to MASDEPs, and the need for a common SW development and systems engineering modelling language.

3.1 Software Development Methodologies

According to Bauer [13], “[Software engineering is] the establishment and use of sound engineering principles in order to obtain economical software that is reliable and works efficiently on real machines.” And according to Pressman [13], “The foundation for software engineering is the process layer”, where the process layer is a methodology that defines what activities are performed during software development and in what order.

If one were to define “simulation engineering” as the “establishment and use of sound engineering principles in order to obtain economical *simulations* that are reliable and work efficiently on real machines” and “the foundation for simulation engineering is the MASDEP”, there are obvious parallels between software and simulation development. Both transform user requirements into usable software-based systems that are economical and reliable, and both are based on a process. Considering how much research has gone into studying and improving software development processes over the years, and the parallels between SW and simulation development, the ISE work stream is investigating SDMs in order to help formulate an appropriate MASDEP for CapDEM. As well, since model reuse and extensibility is so closely tied to the manner in which the matching SW is developed, the authors believe that an SDM should be incorporated into the MASDEP.

A recent study has indicated that many SDMs exist in the literature. Some of the well-known methods are as follows:

- The Waterfall Model. Most people believe this model promotes a one-way flow of effort from user requirements through to software delivery. Although the original research did not promote a strict one-directional flow, the term is still used to denote such an approach. It is not popular because each step cannot accommodate changes that require earlier steps to be revisited. Since last minute changes invariably occur in the real world, the model is not very useful.

- The Prototyping Model. This approach is intended for situations when the requirements cannot be well defined at the early stages of a project. This model promotes the early development and delivery of a working system with limited functionality. The application is intended to be evaluated by the users and then refined and enhanced in an iterative fashion.

- The Rapid Application Development (RAD) Model. This model promotes the rapid and linear (non-iterative) development of applications based on SW components.

- The (Boehm) Spiral Model. This method is an iterative process that develops SW in a series of incremental releases. Unlike the Prototyping Model, it assumes that user requirements are well known at the beginning of a project; it is also more systematic in managing the changes from one iteration to another.

Ultimately, every SDM produces software from a set of user requirements and so what happens in between is often ignored by higher-level processes. Presumably, this is why SW development is not addressed in a typical MASDEP.

3.2 Relationship of SDMs to MASDEPs

A MASDEP that is devoted to the HLA is focused on the creation of a federation based on user requirements. Ideally, an entire federation could be developed without the need for any new software. To do so would require a repository of federates that could be configured as required. The various configurations would have to address simulation requirements such as:

- what terrain database to use

- what computer generated force (CGF) entities should appear, their movements and their behaviour characteristics

- what FOM to use, assuming the federates were FOM-agile (meaning that they can work with multiple FOMs)

In addition, configuration settings would have to specify infrastructure details such as the exercise number, federate identifiers, network addresses, RTI settings, etc.

The reality, however, is that HLA federates are rarely so flexible that they can be configured for arbitrary simulations. A few of the major limitations are:

- The underlying models were originally designed for a limited and particular purpose, usually with little regard for future use due to the increased development costs.
- Many federates are designed to work with a single type of database, whether it be terrain, computer graphics, etc.
- Many federates are not FOM-agile because the increased development costs and complexity outweighed the increased flexibility.
- Most federates are designed to work with a particular RTI.

A potential solution to many of these issues is the development or adoption of new standards within an organization. While such an approach would likely reduce the number of internal compatibility problems, issues would arise when the organization attempted to participate in a simulation that involved any outside organizations that adopted different standards.

Another issue is that of constantly changing M&S technology and the resulting support necessary for legacy simulations. Today, HLA is considered the state-of-the-art technology, but in a few years time, it too will likely be updated or replaced with a new technology. As a result, today's federates will become obsolete and require updating or some form of "wrapping" (an interface that maps legacy software to a new interface standard).

The above issues indicate that the likelihood of a federate repository that can be configured for arbitrary federations is not likely to happen anytime soon. Furthermore, even if it could be developed, it would likely become obsolete very quickly. This observation is supported by past simulation standardization efforts. For instance, the Distributed Interactive Simulation (DIS) Standard [14] was intended to address many of the above issues, yet it was only partially successful because it did not foresee the network bandwidth issues that would arise, nor did it have sufficient Protocol Data Units (PDUs) to meet everyone's needs. In particular, the latter issue resulted in the development of "custom" PDUs, which undermined the standardization effort.

Conversely, the HLA is suffering from the opposite problem, that is, its intended openness. HLA-compliant federates are frequently unable to interoperate (not that the HLA was intended to enable universal interoperability), because federation developers fail to recognize that the openness of the standard is a double-edged sword. They are free to develop their own FOMs but when a federate is developed based on one in particular, it is rarely able to work with any other (unless a gateway or bridge is used). Further, developing a FOM-agile federate complicates the software so much that it is often not done, the result being federates that cannot readily interoperate.

What is needed to manage this process is a MASDEP that also addresses aspects of individual federate development to ensure that a federate is developed in such a way that its individual parts are reusable. For instance, the dynamics of an aircraft model that are based on inputs such as operator controls and environmental factors should be separated from control device drivers, environmental data calculations, networking software, data logging, etc.

Such an approach would enable the reuse of key components that are used to developer federates. The reuse would enable model extensions to support new functionality and adaptation of the model to new M&S standards. Depending on the tools available, the process of reusing key components in new federates could also be partially or fully automated.

To extend a MASDEP so that it addresses SW development issues is not a trivial problem; nor, however, is it insurmountable. More specific process details have to be specified to address issues such as SW design, testing, and configuration management, but many of these are already addressed in software engineering literature. Thus, a great deal of research on the subject is readily available.

To facilitate the development of such a detailed MASDEP, a common language is required for both SW and federation developers to ensure an unambiguous specification of federation and federate requirements as well as the underlying SW. In addition, standards are required to help ensure that simulation developers are using common processes and tools, which will help to reduce the number of problems that arise. The next two sections discuss the use of a common modelling language and standards in much more detail.

4. Unified Modeling Language

UML [15,16] is the leading industry-standard language for specifying, visualizing, constructing, and documenting the artefacts of software systems. UML foundations rely on Object-Oriented Analysis and Design (OOAD), and represent the accomplishment of:

- research performed by G. Booch, I. Jacobson and J. Rumbaugh at Rational Software Corporation
- later standardization efforts of many industries and academic centres under the auspices of the Object Management Group (OMG)[3]

The next section describes the most relevant characteristics of UML for software development, the latter being the reason why UML was created. Next, the efficacy of UML V2.0 for systems engineering, modelling, and design is justified, which leverages its use in software engineering. Finally, the significance of UML V2.0 for the CapDEM ISE work stream as part of a SoS iterative development process is addressed.

4.1 UML for Software Engineering

As software development became increasingly complex, the need for design tools led to the development of the UML in the late 1990s. The work was driven in part by the success of design tools in many other technical disciplines, such as Computer Aided Design (CAD) tools for mechanical engineering and building construction. Since design tools were used extensively and successfully in these industries, equivalent tools for software design were sought.

The tools that were developed reflected the commonality between software design and other technical activities. For example, software design:

- serves to define the architecture of the final product
- is essential for communication between projects teams
- is predictive, as it allows conclusions to be derived about the real targeted software system
- makes large complicated systems much more understandable by describing individual components separately

[3] The OMG is an international organization that promotes the theory and practice of object-oriented technology in software development.

- reduces overall project costs by helping to ensure that the system built meets the project requirements

- reduces the need to modify an operational system by making design flaws evident at an earlier stage, when they are also much cheaper to rectify

Furthermore, a software design model can provide a high degree of abstraction by emphasizing more important aspects, while ignoring irrelevant or less important details.

It is highly desirable that the user's model or architecture takes into consideration various aspects of software engineering. Important issues include reliability, robustness, usability, performance, evolvability, reusability, scalability, portability, etc.

Unfortunately, it is very hard for a software design to encompass all of these good software practices in one single view, much like a single blueprint cannot depict all of the mechanical, electrical, and architectural designs for a building. Therefore, UML uses an approach where the problem is attacked from different views, resulting in multiple design diagrams for a given architecture. The set of UML diagrams includes all of the following:

- Use case diagrams
- Class diagrams
- Behaviour diagrams, which consist of:
    - Statechart diagrams
    - Activity diagrams
    - Interaction diagrams, which include:
        - Sequence diagrams
        - Collaboration diagrams
- Implementation diagrams, which consist of:
    - Component diagrams
    - Deployment diagrams

Different diagram examples, all part of the design of a sample business activity application, are shown below:

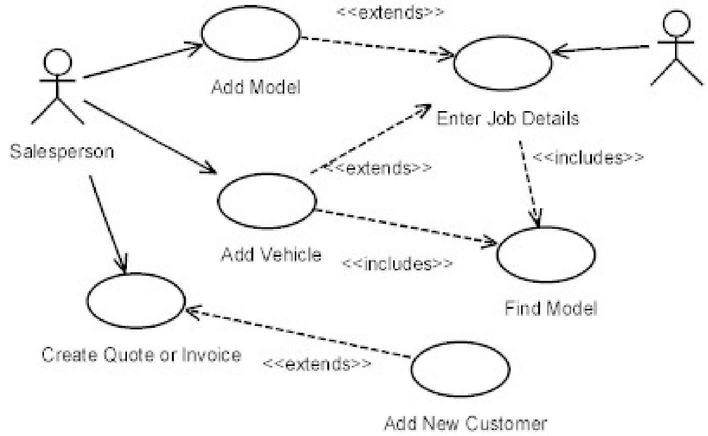

Figure 6. Use Case Diagram

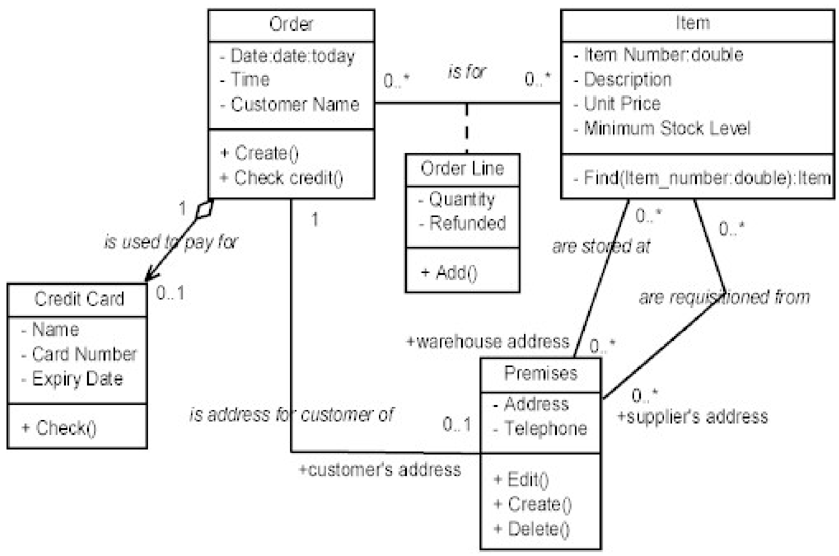

Figure 7. Class Diagram

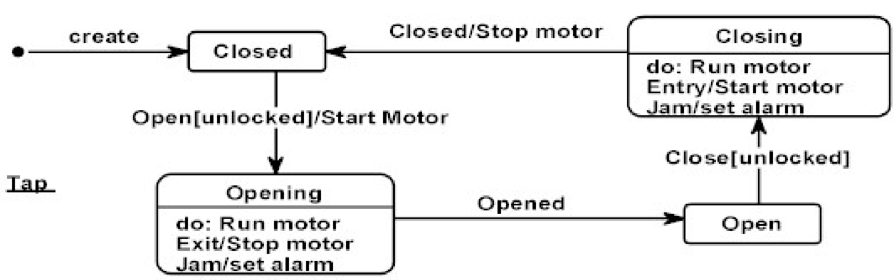

Figure 8. State Diagram

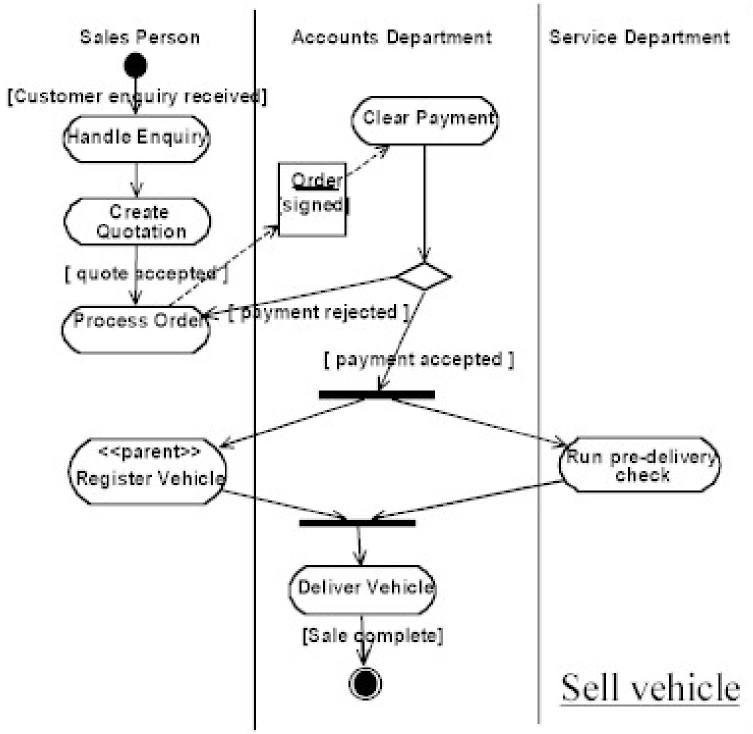

Figure 9. Activity Diagram

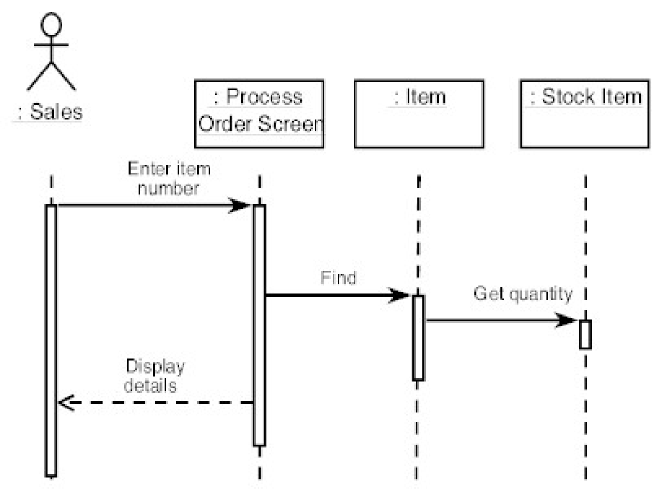

Figure 10. Sequence Diagram

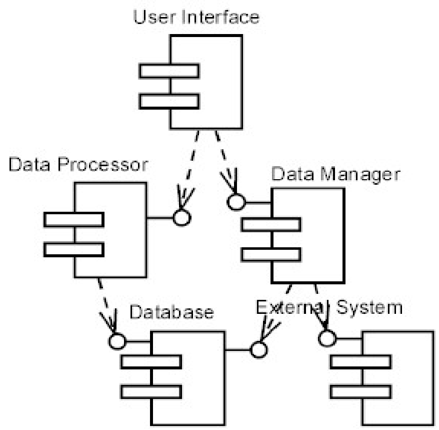

Figure 11. Component Diagram

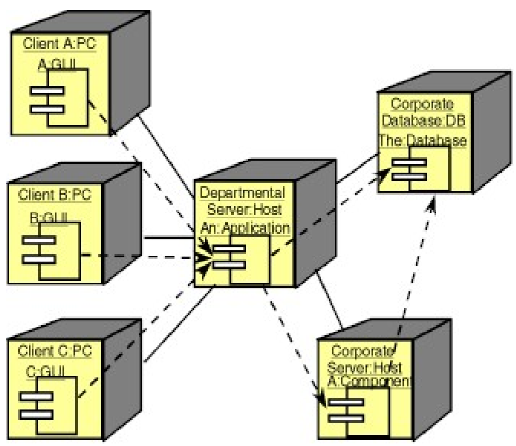

Figure 12. Deployment Diagram

These diagrams provide multiple perspectives of the analysed system and constitute the primary artefacts that a software system architect uses. The software architect makes use of only those diagrams that capture the relevant aspects of his design; the use of all of the diagrams for a single design is not necessary.

Various commercial tools support users in creating such diagrams as well as providing other powerful capabilities such as automatic code generation from the design in a target language (such as C/C++ or Java). Users can also customize the code generation by using code templates or by specifying explicit code segments, such as assembly code for performance or memory-critical parts of the system.

4.2 UML V2.0 for Software and Systems Engineering

The suitability of UML for systems engineering design has been questioned in recent studies [17,18]. As it was originally designed for software development, UML V1.x fails when dealing with systems engineering modelling. Identified gaps include:

- It is too low level because of the embedded C++ concepts in it.
- It has inadequate modelling capabilities for:
  - business process modelling, and
  - large-scale and general distributed systems.
- It does not address some non-functional aspects, such as quality-of-service specifications (QoS).
- It is too complex because it encompasses too many concepts, some of which overlap (for example, “Class”, “subsystem”, and “Component”).
- Its semantics are inadequately defined. In particular:
  - Some are vague or missing, such as those for inheritance and dynamic semantics.
  - It has an informal definition, making it unsuitable for code generation or executable models.
- It has no diagram interchange capability, the result being commercial-off-the-shelf (COTS) tools store UML diagrams in proprietary formats. As a result, the diagrams produced with one tool cannot be shared with tools from other vendors.

The OMG is currently working on the standardization of UML V2.0. They have identified the following list of items to be supported in the new version:

- Continuous time behaviour
- Decision trees (for example, support for trade studies)
- Geometric and spatial information (for example, installation drawings and 3D models)
- Hierarchical modelling of scenarios and behaviours
- Input/output flow (including data and mass/energy flow)
- Integration with other specialty engineering models (for example, mechanical, electrical, reliability, and safety models)

- Parametric relationships (for example, performance models)
- Performance, physical, and non-behavioural characteristics (such as accuracy, weight, safety, and reliability)
- Problem definition and causal analysis
- Properties (such as performance and physical characteristics) including probability distributions
- Physical interfaces and connections
- Requirements constructs
- System, subsystem, element, and component definitions and representations
- Terminology harmonization
- Verification and validation results

Clearly, UML V2.0 will offer significantly more power than its previous versions. Presumably, SW tools that support UML will quickly be updated to support the new standard. When they become available, they will be evaluated for use in the ISE work stream.

4.3 Systems Modeling Language

A UML-related initiative involving the OMG and the International Council on Systems Engineering (INCOSE) is working towards the elaboration of a "pure" systems engineering modelling language, called the Systems Modeling Language (SysML) [19]. It mainly uses UML V2.0 profiles to support systems engineering paradigms, and to provide requirements for the following capabilities:

- Specification
- Analysis
- Design
- Verification and validation

Intended applications include complex systems with both hardware and software components.

The SysML partners are also collaborating with the International Organization for Standardization (ISO) AP-233 working group to ensure that SysML is architecturally aligned with the evolving AP-233 standard for data interchange between systems engineering tools.

4.4 Significance of UML for the ISE Work Stream

UML V2.0 is expected to become very useful to the ISE work stream. It is hoped that it will enable:

- The convenient capture and depiction of user simulation requirements
- The design of conceptual models for simulations in unambiguous terms that can be directly transferred to a software implementation
- The development and execution of the entire simulation based on a UML model

The adoption of UML is expected to provide all steps of the MASDEP for CapDEM with a common language, enabling the inconsistent passage of data and information from one federation development step to another.

By using UML throughout, the ISE process will also be more likely to enable model reusability and composability. These capabilities will result from model, federate, and federation designs being readily available in a format that is well-suited to modification and extension. The designs will even support the automatic generation of source code, which should make the simulation development process much more efficient than it is without UML.

Other benefits can also result from the use of UML V2.0. One of the most important is the adoption of the Model Driven Architecture (MDA). This approach is one of the prominent standards being promoted by the OMG and supported by UML V2.0. MDA is explained in more detail in section 5.

Finally, if UML V2.0 becomes the de facto standard language for M&S for the capability engineering process being developed by CapDEM, its use will likely spread to supporting and collaborating organizations within and outside of DND. Given that some Canadian companies have regularly used UML for years[4], many obstacles for communicating and sharing technical ideas and designs will be reduced, if not eliminated altogether. Clearly, such improvements can only improve the efficiency with which systems are developed and procured—which is a key goal of the capability engineering process.

[4] One of the authors has personal experience with UML in a commercial setting. In the mid to late 1990s, he worked as a contractor on a multi-million dollar DND electronic warfare project that made extensive use of UML. The leading Canadian systems integration company for which he worked continues today to make extensive use of UML.

5. Other Relevant Standards

A look at the future technology standards in simulation, software development, and systems engineering[5] is mandatory in order to propose an evolutionary M&S process. These fast growing technologies must be considered prior to and during the design of any process or methodology.

This section presents some of the most prominent new standards in simulation as well as in related areas, such as software and systems engineering. Besides UML, which was discussed in the previous chapter, several relevant standards already exist and are being used in industry; examples to be described include the Model Driven Architecture, the eXtensible Modeling and Simulation Framework (XMSF), and the eXtensible Markup Language Metadata Interface (XMI). In addition, many new standards are being developed by organizations such as the Simulation Interoperability Standards Organization (SISO), the OMG, the IEEE, and some leading industries. In this category, the standards of interest are the Simulation Reference Markup Language (SRML) and the Base Object Model (BOM).

5.1 Model Driven Architecture

The OMG MDA [20] provides a mean for designing and building component-based systems that remain decoupled from the platform technologies used to implement the systems. As these technologies are subject to change, the design of the component-based system is preserved.

The MDA achieves the decoupling by separating the fundamental logic behind a component or system specification from the specifics of the particular language, platform, and middleware used to implement it. This approach enables the rapid development and delivery of new interoperability specifications that use new deployment technologies based on proven, tested models.

The MDA captures what a system does in a Platform Independent Model (PIM), and captures how a system is designed and implemented in a Platform Specific Model (PSM). The PIM can be complete and specific enough to enable early execution and testing of the application, independent of the design and implementation specifics.

Figure 13 depicts high-level aspects of the MDA. Central to it are many modelling techniques for capturing component designs (for instance, UML). The designs would be transformed in an automated fashion by implementation methodologies in the next layer, which make use of open standards and languages such as the Common Object Request Broker Architecture (CORBA) and Java. The resulting components would make use of services that are application independent, such as security and directory

[5] Software development and systems engineering best practices are being widely adopted in the simulation community.

services (depicted in the next layer). Resulting applications could be deployed in any industry, such as finance or telecom.

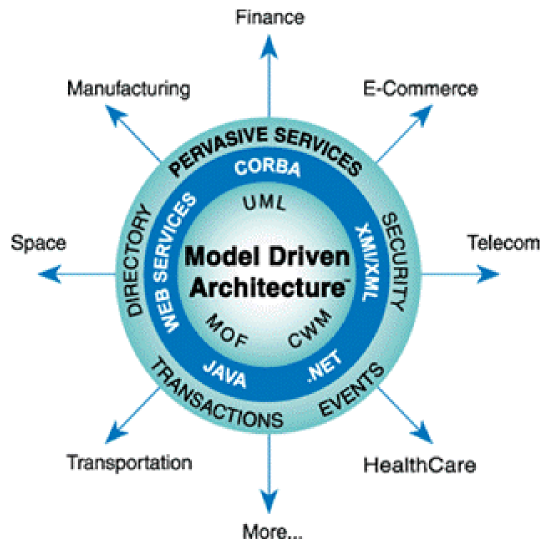

Figure 13. The OMG Model Driven Architecture

Since the simulation world can be considered just one of the many application domains that the MDA can support, it too can profit from the many MDA benefits. These are claimed to be:

- Reduced cost throughout the application/system life-cycle
- Reduced development time for new applications
- Improved application/system quality
- Increased return on technology investments
- Rapid inclusion of emerging technology into existing systems

Perhaps the benefit of most importance to the simulation community is that of long-term model reusability. Since simulation model logic should be independent of any standard that is used to make models interoperate and the MDA separates the two, the MDA readily supports the adoption of new simulation standards without requiring models to be modified. This assumes, of course, that the models were developed using MDA techniques such as UML.

5.2 Simulation Reference Markup Language

SRML [21] has been developed at Boeing Corp. as a means of adding behaviour to the eXtensible Markup Language (XML) [22]. The latter, a standard of the World Wide Web Consortium (W3C), is used to define, validate and share documents on the Web by specifying the organization, content, and presentation of document data. The applications encompass not only the content (words, pictures, etc.) but also the role this information might play.

To be more precise, XML is actually a “markup” language for documents containing structured information. It specifies neither semantics nor a tag set, as does the Hyper Text Markup Language (HTML). In fact, XML is really a meta-language for describing “markup” languages. It provides a facility for defining tags and the structural relationships between them.

SRML can be described as an enhanced XML that can specify the behaviour for distributed simulation models, and its run-time environment is the software that executes those models. The goal of SRML is to enable simulations (or other content) to be served, received, and processed in a standard fashion using Internet technologies and the World Wide Web.

5.3 Base Object Model

The BOM [23] is a concept initially intended to improve reusability and composability of models in HLA-based simulations. As such, BOMs are the basic building blocks to be used when developing an HLA simulation. At its core, a BOM represents a design pattern that identifies an interaction and/or a trigger related to one or more object classes and its attributes. A BOM can also encapsulate additional meta-data and information such as behaviour, which gives it the status of a component. In fact, BOMs are specifically identified in the FEDEP as a potential facilitator for providing reusable object model components that can be used for the rapid construction and modification of simulations and simulations spaces.

5.4 eXtensible Modeling and Simulation Framework

XMSF [24] is a recent technology designed by a group of US government, academic, and industry experts, under the leadership of the Naval Postgraduate School, George Mason University, and Science Applications International Corporation (SAIC). This initiative comes as one of the research avenues being explored as part of the US DoD institutional transformation.

XMSF aims at improving simulation capabilities by using Web-based technologies, since they are the only viable platform for running software and systems worldwide. XMSF is defined as a composable set of standards, profiles, and recommended practices for Web-based M&S. XMSF precepts are:

- Web-based technologies applied within an extensible framework will enable a new generation of M&S applications to emerge, develop, and interoperate.

- Support for operational systems is missing from current simulation frameworks; however, it is an essential requirement.

- An extensible framework of XML-based languages can provide a bridge between forthcoming M&S requirements and open/commercial web standards, while continuing to support existing M&S technologies.

- Compatible and complementary technical approaches are now possible for model definition, simulation execution, network-based education, network scalability, and 2D/3D graphics.

- Web-based technology, software tools, and content production combined with the ubiquitous presence of the Internet provide the best business case from an enterprise-wide and worldwide perspective.

5.5 eXtensible Markup Language Metadata Interface

In the same product line as XML, SRML and XMSF, XMI [25] is another OMG standard for enhancing the exchange of objects in a distributed environment. XMI specifies an open information interchange model. It is intended to give developers that work with object technology the ability to exchange programming data in a standardized way. Thus, it will bring consistency and compatibility to applications being created in collaborative environments. By establishing an industry standard for storing and sharing object programming information, development teams can collaborate on applications while using tools from multiple vendors. The proposed standard will allow developers to leverage the Internet to exchange data between tools, applications, and repositories to create secure, distributed applications built in a team environment.

Moreover, the objective of XMI is to allow the exchange of objects from the OMG's Object Analysis and Design Facility (OADF). These objects are most frequently described using UML notation, or less commonly, with the Meta Objects Facility (MOF) [26], another OMG standard. As of yet, there is no industry standard for representing these notations and as a result, a number of proprietary formats have emerged. Each format is specific to the tools from a single vendor, which limits the ability to exchange data.

5.6 Significance for the ISE Work Stream

All the above-mentioned standards can be of benefit for the CapDEM ISE MASDEP. First, the MDA enhances the development and maintainability of a system during its life cycle, and provides an open framework for the integration of emerging technologies. As discussed, it also provides a key to long-term model reusability.

SRML, on the other hand, provides a means of exploiting the Internet by offering distributed processing capabilities and a means of exchanging simulation models. This constitutes a significant contribution to the *collaborative* aspect of the overarching CapDEM systems-of-systems engineering process.

The BOMs are to a simulation environment what components are to current software development environments. With the latter, a developer can build an entire application by interconnecting personally developed software components with those found in repositories or third party libraries. In a simulation environment, a developer will be able to construct federates more rapidly by reusing or modifying BOMs, that is, previously developed and tested models.

Like SRML, XMSF and XMI can also strengthen the distributed simulation capabilities of the ISE work stream by providing a framework for distributed simulations. By leveraging multiple web technologies and a common language for the exchange of programming objects (such as UML systems models), simulations can be developed, distributed, and executed using everyday tools, such as web browsers.

Although these technologies look promising, they will undoubtedly have limitations as well. For instance, while SRML and XMSF technologies may be very useful for building, distributing, and running simulations over the Internet, these approaches are not likely to be suitable for the most computationally intensive models or for very complex scenarios. Unfortunately, the bandwidth limitations and unpredictable performance of the Internet may limit the suitability of web-based simulations to the early stages of concept development, when the primary goals are the rapid testing, demonstration, and sharing of ideas. If this proves to be true, the technologies involved will still be valuable but not sufficient for all aspects of capability engineering.

The ISE work stream will continue to monitor the development of all such technologies so that it can incorporate the most appropriate into the MASDEP for CapDEM.

This page intentionally left blank.

6. A Vision for the MASDEP, Tools, and Standards

The ISE work stream needs to select and develop a MASDEP, tool suite, and standards that will support the M&S requirements of capability engineering. They must be compatible with others being used in CapDEM, especially those used by the overarching Capability Engineering Process (CEP).

This section considers a number of issues before presenting a vision for the process, tools, and standards, with emphasis on the MASDEP. The latter is the most important because once it is defined, an appropriate tool suite is often apparent. Standards, on the other hand, are the subject of current research and are expected to change significantly in the next few years; thus, it is too early to identify an appropriate set at this time.

6.1 M&S-Related Issues

The issues to be considered are based on the need of the MASDEP to support the M&S requirements of capability engineering (as discussed in Section 2). Other factors are derived from the requirement of the MASDEP to benefit M&S activities that would use a formal development process (since it would be used for such activities). The M&S-related issues that arise are:

- **The application domain:** M&S is used extensively in the fields of science, engineering and operations research. For the purposes of the ISE work stream, the type of capability being engineered is irrelevant because the process used to develop a simulation is, in general, application independent. Domain-specific details would only arise when a MASDEP was being tailored for a particular application.

- **The level of sophistication:** M&S models can range from small and simple to large and complex. M&S projects that use simple models are ubiquitous and their simplicity lends itself to informal development. For instance, changes can be made rapidly and many ideas tested quickly—provided there is little overhead in the process. Large, complicated simulations, on the other hand, usually require a coordinated effort to develop and execute. As a result, they benefit from a more structured development and run-time environment. Thus, the higher the level of sophistication, the more likely that an M&S effort will require a formal development process.

- **The intended audience:** At one extreme, individuals who are not computer specialists perform “everyday M&S” using general-purpose applications such as spreadsheets. Typically, they use a computer for common calculations such as estimating project costs or estimating a trend in data—both of which are very simple forms of M&S. At the other end of the spectrum, teams of highly trained computer and engineering specialists develop large, distributed, real-time simulations. In general, the former do not require a very formal process for their day-to-day M&S activities; the latter generally welcome, or at least recognize, the benefits that a formal process brings to a large development effort.

- **The simulation infrastructure:** Small simple models are frequently developed and run using general-purpose tools such as spreadsheets on common PCs. In general, large complex models require more computing resources and so they are developed for a distributed computing environment. The latter are usually much more complex and regularly involve multiple types of applications, multiple copies of applications, local- and wide-area networks, high-performance computers, etc. As a result, many computer specialists are required to develop and maintain the simulation infrastructure; therefore, their activities need to be recognized in a MASDEP.

Clearly, none of the above issues are "black-and-white" because each has a continuum of cases. However, they do indicate a trend, that is, a formal development process is more likely to benefit and to be used by technical specialists who develop and run complex distributed simulations. Relatively non-technical personnel who develop and run simple models on common PCs are not likely to want or use such a process because of the discipline it demands and the time that is lost to process overhead.

The above issues indicate that the MASDEP should target the development of relatively sophisticated simulations, and its developers should assume that it will be used by technical specialists. Note, however, that this does not preclude a comprehensive MASDEP from being tailored for less demanding simulations. Similarly, it could also be simplified for people who are not computer experts.

6.2 A Vision for the Process, Tools, and Standards

A key contribution of the ISE work stream to CapDEM will be the development of a MASDEP supported by tools and standards that take into consideration the issues just discussed. To be more specific, it must have the following properties:

- It will support the capability engineering requirements of CapDEM. Regarding the issues discussed in section 2, a MASDEP is not expected to require any special features to support visualization, nor will it require features directly related to options analysis. Both of these issues will be controlled and specified by the CEP and passed to the MASDEP. Similarly, the scientific rigour used in the simulations will be determined by the CEP and specified in the simulation requirements passed to the MADEP. MASDEP VV&A processes, however, will ensure that the models and simulations meet the specified standards. The MASDEP will also have to ensure that the models and simulation results are traceable back to the appropriate sources using configuration management and VV&A tools. By appropriate uses of configuration management and repositories, the MASDEP can also help transition the development of a capability from one phase to another.

- The MASDEP will interface with the CEP so that simulation requirements can be passed to the MASDEP with minimum effort. A convenient and robust means of doing so must be identified or developed, and it should be based on an open standard such as UML. It is important to note that simulation requirements may be much more than a text description because the requirements may make reference to models in repositories or any other type of computer-based resource. The requirements may also incorporate diagrams that depict configurations of systems to be modelled. To avoid ambiguity in the diagrams and text, both should be based on a modelling language that has well-defined semantics (such as UML).

- The MASDEP will interface with the CEP so that simulation results can be passed back to the CEP with minimum effort. The results are likely to be reports or data that can be readily accessed using general-purpose tools such as spreadsheets, word processors, and movie players (for replaying simulation results).

- The MASDEP must be practical to use. It must be understandable by the intended audience, that is, technical experts who develop and execute simulations.[6] It must have sufficient detail to provide meaningful guidance but still be flexible so that it can be tailored for specialized situations. It must be sufficient for complex M&S issues but also useful for developing simpler simulations. It must also be comprehensive to guide simulation development from the analysis of user requirements through to the return of simulation results.

- The MASDEP must utilize current and future open standards. Currently, many people and organizations (OMG, SISO, IEEE, etc.) are actively involved in the development of new M&S and related standards. Since some of the standards may compete with one another, decisions may have to be made regarding which standards to adopt. Further, since standards have a finite lifespan, the MASDEP should be as independent of standards as possible and refer to them in a generic fashion. For instance, it should be applicable to HLA simulations but not exclusively tied to it.

- The MASDEP must leverage previous M&S process work, such as the HLA and the SEDEP. The strengths and weaknesses of such processes need to be identified so that their best features can be integrated with any new process steps required.

- The MASDEP must have sufficient depth to address software development issues and related issues such as the Model Driven Architecture. Doing so will help to ensure that models developed are not tied to language-specific software implementations. As a result, the models will more likely be understandable, extensible and reusable in the long term. With an appropriate set of standards, it will also help to ensure that software is well documented and reusable, regardless of its function.

[6] Note that the process does not have to be meaningful to simulation participants, such as military personnel, who are interacting with the simulation applications as if they were real systems.

- The MASDEP must be supportable using tools and repositories. Ideally, the tools should be everyday COTS products, configured as necessary to meet the needs of CapDEM. However, if appropriate tools cannot be located, then the use of custom-developed tools will need to be considered. At a minimum, the tools must: (1) support a specified set of input and output standards that enable the tools to work together efficiently; and (2) enable individual tools to be replaced without compromising the flow of data through the suite. The tools must also support network-based repositories that store data and possibly applications that are required by a project.

- The MASDEP must make use of open standards to help ensure CapDEM simulation applications are interoperable with simulations developed elsewhere. Proprietary standards are to be avoided because they greatly reduce interoperability options, force future simulation development to be directed to the developers of the standards, and do not benefit from the breadth and depth of experience that goes into open standards.

Because the ISE work stream is attempting to adopt and integrate as much as possible from previous research and commercial efforts, the vision for the development of the MASDEP is to adopt and modify an existing process. The EUCLID RTP 11.13 SEDEP is the most promising at the time of this writing, although it is acknowledged that the SEDEP and its tool suite have not been evaluated in depth (due to the delays in the release of the RTP 11.13 DVDs).

Although the FEDEP has a longer history than the SEDEP and it has been adopted by the IEEE, the SEDEP is preferred because it describes each of its high-level steps in terms of several low-level steps. This additional information is considered essential in making a process practical to use. In addition, each of the SEDE tools has been designed to work together as a suite and they all make use of a resource repository (for software tools, models, etc.)

Although the SEDEP is very promising, the ISE work stream does have concerns about the number, quality, and usability of the SEDE tools. At the RTP 11.13 Final Demonstration in Paris (12-13 November 2003), the tool demonstrations indicated that some of the tools should have been combined with others. Further, it was often stated that some tools were “prototypes”, which suggested that the quality was not likely to be very high.

6.3 The Road Ahead

The ISE work stream is planning on evaluating the EUCLID RTP 11.13 SEDE as soon as it becomes available. Although the DVDs with the software are said to have been released, a copy has not yet been received. The evaluation of the process and tool suite is expected to take six months. However, this estimate is fairly uncertain because the number of tools to be released is unknown. The latter is a result of intellectual property issues that complicated their release.

One significant issue to be dealt with during the tool evaluation is the lack of support for the tools. The quality of support cannot be expected to be as good as with COTS products given that the project has ended, and that the tools were developed mostly by universities, government organizations, and companies who are no longer getting paid for their time.

In addition, all of the tools were designed to work with a centralized repository; consequently, one will have to be developed if the necessary software is not provided on the RTP 11.13 DVDs. This is another unknown in terms of time and complexity.

Besides the SEDE tool evaluation, a VV&A process study is underway. It is leveraging the efforts of many other groups within DND, such as SECO. As with the MASDEP, the ISE work stream intends on adopting and integrating as much as possible from other research and commercial efforts rather than developing new VV&A processes from scratch.

Once the SEDE and VV&A studies have been completed, the ISE work stream will be prepared to propose and implement a MASDEP for CapDEM. The intention is to test it by applying it, in whole or in part, to a number of other projects within DRDC, specifically those that are considered "use cases" for CapDEM. The Multi-Mission Effects Vehicle (MMEV) TDP was being considered at one point although its advanced status will likely make it unsuitable; if so, another will be identified.

This page intentionally left blank.

7. Conclusions

Capability engineering will rely heavily on M&S technology. It will likely involve high-level models and visualization in the early stages and more detailed and sophisticated models as options are analyzed and the capability under study evolves into a system being procured. Over time, the models can be extended and modified to support subsequent activities such as training, maintenance, upgrades, etc.

Considering the sophistication of modern military systems, capability engineering cannot help but require complex M&S, resource repositories, reusable and extensible models, realistic synthetic environments (SEs), and other advanced technologies and processes. To manage the development and execution of such systems, a comprehensive process, set of tools, and various standards are required.

Existing processes, such as the HLA and SEDEP, are excellent starting points for the development of an appropriate process for CapDEM. A great deal of effort has gone into their development and much can be learned from them. However, they are not expected to be useful without some modification because the originators did not have to consider details such as an overarching CEP process, nor did they address key issues such as SW development and configuration management.

The MASDEP to be developed for CapDEM will leverage previous work and be based on current and future open standards in order to ensure compatibility with current and future software-based systems. The intention is to base all steps of the process on a common modelling language as well. UML is the best solution identified to date.

The MASDEP is likely to be based on the EUCLID RTP 11.13 SEDE, that is, the SEDEP and its tool suite. Future studies will identify its strengths and weaknesses as well as the best features to be incorporated into the CapDEM MASDEP. Although the tools are not expected to be as polished as COTS products or as well supported, hopefully some may prove to be very useful. The rest, at a minimum, will provide many valuable lessons learned.

The ISE work stream will also leverage the M&S standards development being conducted by organizations such as SISO, the OMG, etc. Many of the standards are in a state of flux but several web-based technologies are expected to become very useful in the next few years. The CapDEM MASDEP will make use of them whenever possible to ensure that M&S performed as part of capability engineering benefits from these efforts and meets internationally recognized standards.

This page intentionally left blank.

References

1. *The DMSO High Level Architecture (HLA)* (1996). Alexandria, Virginia: Defense Modeling and Simulation Office.

2. Kruchten, P. (2000). *The Rational Unified Process: An Introduction*. Toronto, Ontario: Addison-Wesley.

3. *High Level Architecture Federation Development and Execution Process (FEDEP) Model, Version 1.5* (1999). Alexandria, Virginia: Defense Modeling and Simulation Office.

4. *IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)-Framework and Rules* (2000). New York: Institute of Electrical and Electronics Engineers.

5. *IEEE Recommended Practice for High Level Architecture (HLA) Federation Development and Execution Process (FEDEP)* (2003). New York: Institute of Electrical and Electronics Engineers.

6. *SEDEP Version 1.0, SA-WE1.5-Volume 2* (2001). Euclid RTP 11.13-TT&S.

7. Paulk, M., Curtis, B., Chrissis, M. B., Weber, C. (1993). Capability Maturity Model for Software, Version 1.1. Technical Report CMU/SEI-93-TR-024, Software Engineering Institute. Pittsburgh, Pennsylvania: Carnegie Mellon University.

8. *High Level Architecture Federation Development and Execution Process (FEDEP) Checklists, Version 1.5* (1999). Alexandria, Virginia: Defense Modeling and Simulation Office.

9. Dobey, V. T. and Lewis, R. O. (2003). Verification, Validation, and Accreditation (VV&A) Process Overlay for the FEDEP, *SISO Spring Simulation Interoperability Workshop 2003*.

10. *Modelling and Simulation Verification, Validation and Accreditation (VV&A) Guidebook* (2003). Canadian Forces Synthetic Environment Coordination Office. www.drdc-rddc.dnd.ca/seco/documents/ - VVA_Guidebook_DND_SECO_May_2003_e.html. (April 2004)

11. Filsinger, J., and Lubbes, H. O. (1996). System Security Approach for the High Level Architecture (HLA). In *Proceedings of the 14th Workshop on Standards for Interoperability of Distributed Simulation*.

12. *High Level Architecture Federation Security Process, Version 1.2* (2001). Alexandria, Virginia: Defense Modeling and Simulation Office.

13. Pressman, R. S. (1997). *Software Engineering: A Practitioner's Approach*. Toronto, Ontario: McGraw-Hill.

14. *IEEE Standard for Distributed Interactive Simulation – Application Protocols, Standard 1278.1a-1998* (1998). New York: Institute of Electrical and Electronics Engineers.

15. Booch, G., Jacobson, I.., Rumbaugh, J. (1999). *The Unified Modeling Language User Guide*. Toronto, Ontario: Addison-Wesley.

16. *Unified Modeling Language (UML), Version 2.0, Draft Adopted Specification* (2003). The Object Management Group.

17. Franklin, S. J. (2002). Assessing the suitability of UML for Capturing and Communicating Systems Engineering Design Models. Vitech Corporation, www.vtcorp.com/infocenter/SuitabilityOfUMLForSE_2002.pdf.

18. *Requirements Analysis for Systems Engineering (SE) UML* (2003). The Object Management Group. www.omg.org/docs/syseng/02-11-01.pdf (April 2004).

19. *The Systems Modeling Language SysML*. www.sysml.org (April 2004).

20. *Model Driven Architecture* (2001). The Object Management Group. www.omg.org/mda (April 2004).

21. Reichenthal, S. W. (2002). Re-Introducing Web-Based Simulation. In *Proceedings of the 2002 Winter Simulation Conference*.

22. *eXtensible Markup Language (XML)* (2004). The World Wide Web Consortium (W3C). www.w3.org/XML (April 2004).

23. Gustavson, P. L., Hancock, J. P., McAuliffe, M. (1998). Base Object Models (BOMs): Reusable Component Objects for Federation Development. *SISO Fall Simulation Interoperability Workshop 1998*.

24. *The eXtensible Modeling and Simulation Framework (XMSF)*. The Modeling, Virtual Environments, and Simulation (MOVES) Institute, Naval Postgraduate School. www.movesinstitute.org/xmsf/xmsf.html (April 2004).

25. *OMG XML Metadata Interchange (XMI) Specification*. The Object Management Group. www.omg.org/technology/documents/formal/xmi.htm (April 2004).

26. *Meta-Object Facility (MOF), Version 1.4*. The Object Management Group. www.omg.org/technology/documents/formal/mof.htm (April 2004).

List of Abbreviations

BOM	Base Object Model
CAD	Computer Aided Design
CapDEM	Collaborative Capability Definition, Engineering and Management
CEP	Capability Engineering Process
CGF	Computer Generated Force
CM	Configuration Management
CORBA	Common Object Request Broker Architecture
COTS	Commercial Off-The-Shelf
DIS	Distributed Interactive Simulation
DMSO	Defense Modeling and Simulation Office
DND	Department of National Defence
DoD	Department of Defense
DRDC	Defence Research and Development Canada
EUCLID	European Co-Operation for the Long-term In Defence
FEDEP	Federation Development and Execution Process
FFSE	Future Forces Synthetic Environments
FOM	Federation Object Model
FSP	Federation Security Process
HLA	High Level Architecture
HTML	Hypertext Markup Language
IEEE	Institute of Electrical and Electronics Engineers
INCOSE	International Council on Systems Engineering
ISE	Integrated Synthetic Environment
ISO	International Organization for Standardization
M&S	Modelling and Simulation
MASDEP	Modelling and Simulation Development and Execution Process
MDA	Model Driven Architecture
MMEV	Multi-Mission Effects Vehicle
MOF	Meta Objects Facility

MLS	Multi-Level Security
NSTISSC	National Security Telecommunications and Information Systems Security Committee
OADF	Object Analysis and Design Facility
OMG	Object Management Group
OMT	Object Model Template
OOAD	Object Oriented Analysis and Design
PDU	Protocol Data Unit
PIM	Platform Independent Model
PSM	Platform Specific Model
QoS	Quality-of-Service
RAD	Rapid Application Development
RTI	Run-Time Infrastructure
RTP	Research and Technology Program
RUP	Rational Unified Process
SAIC	Science Applications International Corporation
SDM	Software Development Methodology
SEDE	Synthetic Environment Development Environment
SEDEP	Synthetic Environment Development and Exploitation Process
SDE	Software Development Environment
SE	Synthetic Environment
SECO	Synthetic Environment Coordination Office
SEWG	Synthetic Environment Working Group
SISO	Simulation Interoperability Standards Organization
SoS	Systems of Systems
SRML	Simulation Reference Markup Language
SW	Software
SysML	Systems Modeling Language
TDP	Technology Demonstration Project
UML	Unified Modeling Language
V&V	Verification and Validation
VV&A	Verification, Validation, and Accreditation

W3C	World Wide Web Consortium
XMI	XML Metadata Interface
XML	eXtensible Markup Language
XMSF	eXtensible Modeling and Simulation Framework
3D	Three-Dimensional

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Establishment sponsoring a contractor's report, or tasking agency, are entered in section 8.)

   Defence R&D Canada – Ottawa
   Department of National Defence
   Ottawa, Ontario, Canada K1A 0Z4

2. SECURITY CLASSIFICATION (overall security classification of the document, including special warning terms if applicable)

   UNCLASSIFIED

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C or U) in parentheses after the title.)

   A Vision for M&S Processes, Tools, and Standards for CapDEM: An Update by the Integrated Synthetic Environment Work Stream of CapDEM (U)

4. AUTHORS (Last name, first name, middle initial)

   Brown, Richard G.; Hassaine, Fawzi

5. DATE OF PUBLICATION (month and year of publication of document)

   April 2004

6a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc.)

   47

6b. NO. OF REFS (total cited in document)

   26

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

   DRDC OTTAWA TECHNICAL MEMORANDUM

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)

   DRDC OTTAWA

9a. PROJECT OR GRANT NO. (if appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant)

   15AS01

9b. CONTRACT NO. (if appropriate, the applicable number under which the document was written)

10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.)

   DRDC Ottawa TM 2004-133

10b. OTHER DOCUMENT NOS. (Any other numbers which may be assigned this document either by the originator or by the sponsor)

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

   (X) Unlimited distribution
   ( ) Distribution limited to defence departments and defence contractors; further distribution only as approved
   ( ) Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
   ( ) Distribution limited to government departments and agencies; further distribution only as approved
   ( ) Distribution limited to defence departments; further distribution only as approved
   ( ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected.)

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual).

(U) The Department of National Defence (DND) wants to greatly reduce the life-cycle costs and procurement times for new systems and to ensure that new systems will interoperate with existing ones. To do so, the concept of capability engineering has emerged. It will help to quantify what capability is required, evaluate options to achieve it, procure the best system, and manage the system throughout its life cycle.

(U) The Collaborative Capability Definition, Engineering and Management (CapDEM) Technology Demonstration Project (TDP) is developing the processes and tools to enable capability engineering within DND. The processes are expected to rely heavily on Modelling and Simulation (M&S) and so an Integrated Synthetic Environment (ISE) work stream was created. Its mandate is to identify the processes, tools, and standards that are necessary to support various M&S activities within CapDEM.

(U) This report discusses the M&S requirements of capability engineering; the need for a comprehensive M&S Development and Execution Process (MASDEP) and an integrated software development methodology; the need for a common modelling language to be used throughout; current M&S standards, including several under development; and a vision of the processes, tools, and standards to be developed.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

CAPABILITY ENGINEERING
MODELLING AND SIMULATION
DISTRIBUTED SIMULATION
M&S STANDARDS

Defence R&D Canada

Canada's leader in defence
and national security R&D

R & D pour la défense Canada

Chef de file au Canada en R & D
pour la défense et la sécurité nationale

DEFENCE R&D DÉFENSE

www.drdc-rddc.gc.ca