Copy No.________

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

DEFENCE R&D DÉFENSE

# Review of Architectures for Simulation of Virtual Naval Platforms

*Doug Perrault*

**Defence R&D Canada**

Technical Memorandum

DRDC Atlantic TM 2003-193

September 2003

Canada

This page intentionally left blank.

# Review of Architectures for Simulation of Virtual Naval Platforms

Doug Perrault

**Defence R&D Canada – Atlantic**

Technical Memorandum

DRDC Atlantic TM 2003-193

September 2003

Author

Doug Perrault

Approved by

Neil Pegg

Section Head Warship Performance

Approved for release by

Kirk Foster

DRP Chair

## Abstract

In order to meet the rapidly changing requirements for military action, Canada and many of its allies have recognized the utility of modeling and simulation for simulation-based acquisition, training, and operational/tactical experimentation. Canada is looking to High Level Architecture (HLA) to provide the flexibility to model a wide variety of situations, including maritime operations. The Virtual Maritime System Architecture (VMSA) is a viable implementation of the HLA. The initial role of the Virtual Naval Platform (VNP) Group at DRDC Atlantic in either the HLA or the VMSA is to provide motion information for the platform(s) being simulated. This document examines both the HLA and the VMSA architectures in order to determine how best to create motion federates to act in the variety of proposed simulations. After a thorough investigation of the relevant documents describing each architecture, it was determined that the VNP group can develop motion federates for either architecture, but that there are differences in philosophy between the two that will affect the implementation of the motion federate.



## Résumé

Pour satisfaire aux exigences en matière d'action militaire, lesquelles changent à un rythme accéléré, le Canada et un grand nombre de ses alliés ont reconnu l'utilité de la modélisation et de la simulation pour l'acquisition, l'entraînement et l'expérimentation opérationnelle/tactique basés sur la simulation. Le Canada envisage la possibilité d'utiliser l'architecture de haut niveau (HLA) pour offrir la souplesse nécessaire à la modélisation d'une grande variété de situations, notamment des opérations maritimes. L'architecture des systèmes maritimes virtuels (VMSA) constitue une mise en application viable de la HLA. Le rôle initial du Groupe des plates-formes navales virtuelles (PNV) de RDDC Atlantique, en ce qui concerne soit la HLA, soit la VMSA, consiste à fournir des données sur le mouvement pour les plates-formes simulées. Dans le présent document, on examine tant l'architecture HLA que l'architecture VMSA dans le but de déterminer la meilleure façon de créer des fédérés du mouvement utilisables dans les diverses simulations proposées. Après avoir étudié en profondeur les documents pertinents décrivant chaque architecture, on a déterminé que le groupe PNV peut élaborer des fédérés du mouvement pour l'une ou l'autre des architectures, mais qu'il existe entre les deux architectures des différences de principe qui auront une incidence sur la mise en application du fédéré du mouvement.

This page intentionally left blank.

# Executive Summary


Introduction

The Canadian Department of National Defence is looking to use a simulation infrastructure called the High Level Architecture (HLA) to provide integrated, combined, joint simulation environments for the development of tactics and doctrine as well as for training and systems acquisition. In order to implement the HLA, it is necessary to first examine the requirements for development of components of a simulation.

The Virtual Maritime System Architecture (VMSA) is a viable implementation of the HLA. By choice, some of the freedoms of the HLA are traded for more stringent time management and guaranteed repeatability of the initial conditions. The added constraints also mean that the VMSA provides more specific instruction to simulation developers than the HLA alone.

Initially, the key role of the Virtual Naval Platform (VNP) Group at DRDC Atlantic within either the HLA or the VMSA is to provide motion information for the platform(s) being simulated. This report examines both the HLA and the VMSA to determine the relevance of the differences between them in terms of platform motion model development.

Results

The motion federate may be as simple as a single simulation providing position information, or it may be as complex as a sub-federation that provides detailed information about the motion of the platform with varying damage conditions and in wind and waves. The trade-off, as usual, will be between fidelity and bandwidth: higher fidelity will require a much more computationally intensive simulation and more communication bandwidth.

Significance

The Virtual Naval Platform Group is able to build models of ships and seaways for simulations in either a High Level Architecture or Virtual Maritime System Architecture using a library of time domain (and frequency domain) seaway and platform motion objects called ShipMo3D. This capability will add a level of fidelity to a wide range of simulations for developing future force concepts and capabilities, applying systems engineering and advanced acquisition concepts, optimizing integrated weapons systems, as well as facilitating human-in-the-loop testing/training and gathering information for human behaviour representation. All of these tasks are important for increasing the science and technology capacity that will enable the Canadian Forces to meet the challenges of evolving global security issues in a manner that makes the CF readily available to participate in combined and joint operations.

Future Work

The Virtual Naval Platform Group of DRDC Atlantic's Warship Performance Section should build motion federates for various platforms (surface ships and submarines; and even submersibles and unmanned underwater vehicles). The seaway can be modeled internally to the motion federate as required by the Virtual Maritime System Architecture, or it may be modeled as a separate federate. The motion federate or sub-federation can be used in an integrated distributed simulation, or it may be used off-line for engineering analysis.

As a demonstration project, the Virtual Naval Platform Group will simulate a simplified replenishment at sea scenario.

Douglas Perrault; 2003; Review of Architectures for Simulation of Virtual Naval Platforms; DRDC Atlantic TM 2003-193; Defence R&D Canada – Atlantic.

# Sommaire

Introduction

Le ministère canadien de la Défense nationale étudie la possibilité d'utiliser une infrastructure de simulation appelée architecture de haut niveau (HLA) pour offrir des environnements de simulation intégrés, combinés et conjoints aux fins de l'élaboration de tactiques et de doctrines, ainsi qu'aux fins de l'entraînement et de l'acquisition des systèmes. Pour mettre en application l'architecture HLA, il faut d'abord examiner les exigences en ce qui concerne l'élaboration des éléments d'une simulation.

L'architecture des systèmes maritimes virtuels (VMSA) constitue une mise en application viable de la HLA. Par préférence, certaines des libertés offertes par la HLA sont abandonnées en échange d'une gestion plus rigoureuse du temps et d'une répétabilité garantie des conditions initiales. Les nouvelles contraintes signifient également que la VMSA fournit des instructions plus spécifiques aux responsables de l'élaboration de la simulation que la HLA seule.

Le rôle clé initial du Groupe des plates-formes navales virtuelles (PNV) de RDDC Atlantique, en ce qui concerne soit la HLA, soit la VMSA, consiste à fournir des données sur le mouvement pour les plates-formes simulées. Dans le présent rapport, on examine tant l'architecture HLA que l'architecture VMSA dans le but de déterminer la pertinence des différences entre les deux architectures du point de vue de l'élaboration du modèle de mouvement des plates-formes.

Résultats

Le fédéré du mouvement peut être d'une grande simplicité, consistant par exemple en une simple simulation qui fournit des données de position, ou il peut être d'une grande complexité, consistant par exemple en une sous-fédération qui fournit des données détaillées sur le mouvement de la plate-forme dans différentes conditions de dommages et en présence de vent et de vagues. Le compromis, comme c'est normalement le cas, devra être fait entre la fidélité et la largeur de bande : une plus grande fidélité signifie une simulation nécessitant beaucoup plus de ressources de calcul et une plus grande largeur de bande de communication.

Importance

Le Groupe des plates-formes navales virtuelles est capable d'élaborer des modèles de navires et de voies maritimes pour fins de simulation, soit avec une architecture de haut niveau, soit avec une architecture des systèmes maritimes virtuels, en se servant d'une bibliothèque d'objets de mouvements de plates-formes et de voies maritimes du domaine temporel (et du domaine fréquentiel) appelée ShipMo3D. Cette capacité permettra d'élever le niveau de fidélité d'une vaste gamme de simulations visant à élaborer les concepts et les capacités de force futurs, à appliquer les principes de conception des systèmes et

d'acquisition évoluée, à optimiser les systèmes d'armes intégrés, ainsi qu'à faciliter l'essai/entraînement à intervention humaine et à recueillir les données destinées à représenter le comportement humain. Toutes ces activités sont importantes en vue d'accroître la capacité en matière de science et technologie qui permettra aux Forces canadiennes de relever les défis posés par les questions en évolution relatives à la sécurité mondiale, d'une manière qui les préparera bien à participer à des opérations combinées et conjointes.

Travaux futurs

Le Groupe des plates-formes navales virtuelles de la Section de l'évaluation de la performance des navires de combat de RDDC Atlantique devrait élaborer des fédérés du mouvement pour diverses plates-formes (navires de surface et sous-marins, et même submersibles et véhicules sous-marins téléguidés). La modélisation de la voie maritime peut être faite dans le fédéré du mouvement, tel que requis par l'architecture des systèmes maritimes virtuels, ou elle peut être faite dans un fédéré distinct. Le fédéré ou la sous-fédération du mouvement peut être utilisé dans une simulation distribuée intégrée ou être utilisé indépendamment pour l'analyse technique.

Comme projet de démonstration, le Groupe des plates-formes navales virtuelles simulera un scénario simplifié de ravitaillement en mer.

Douglas Perrault; 2003; *Review of Architectures for Simulation of Virtual Naval Platforms* (Étude des architectures pour la simulation des plates-formes navales virtuelles); RDDC Atlantique TM 2003-193; R & D pour la défense Canada – Atlantique.

# Table of Contents

# List of Figures



# List of tables

# 1. Introduction

Recognizing the current world situation as being radically different than it was during the cold war, NATO nations, including Canada, are making major changes to the way they conduct operations. There have been and continue to be major changes in the way wars are fought, demanding large-scale changes in military doctrine and in operational and organizational structures, such that the very nature of war is fundamentally different. Many view these changes as a Revolution in Military Affairs (RMA) [1]. The current risks demand actions by forces that are well equipped, yet flexible and rapidly deployable. To achieve the implementation of innovative technologies and rapid development of tactics and doctrine needed to adjust to the ever-changing face of war requires adequate tools. One major part of RMA is the process of concept development and experimentation (CDE), which is intended to facilitate rapid modifications in doctrine and tactics, taking advantage of advances in technology as they occur. A major tool for carrying out CDE is modeling and simulation (M&S), which will allow testing of concepts, tactics and equipment, as well as training of personnel in an effort to promote rapid deployment and flexibility.

In response to the RMA, the US Department of Defense (DoD) formulated an M&S Master Plan (MSMP) in which they stated their vision for M&S [2]:

> Defense modeling and simulation will provide readily available, operationally valid environments for use by the DoD Components:
>
> 1. To train jointly, develop doctrine and tactics, formulate operational plans, and assess warfighting situations.
> 2. To support technology assessment, system upgrade, prototype and full-scale development, and force structuring.
>
> Furthermore, common use of these environments will promote a closer interaction between the operations and acquisition communities in carrying out their respective responsibilities. To allow maximum utility and flexibility, these modeling and simulation environments will be constructed from affordable, reusable components interoperating through an open systems architecture.

Following the American lead, NATO also developed their own M&S Master Plan (MSMP) [3], wherein they stated their vision for M&S:

> Modelling and simulation will provide a readily available, flexible and cost-effective means to enhance NATO operations dramatically in the application areas of defence planning, training, exercises, support to operations, research, technology development and armaments acquisition. This goal will be supported by a NATO-wide co-operative effort that promotes interoperability, reuse and affordability.

Canada typically participates in combined joint actions; that is, as a member of a coalition ("combined" with other nations) able to provide integrated air, land, and naval capabilities (joint effort). To be effective, the Canadian Forces must maintain the technological capability to operate across the wide range of conflicts occurring today, and to adapt rapidly to the evolving spectrum of conflict. Canada must have effective, robust equipment that is interoperable with the equipment of our allies, especially the US. This will require the right mix of capabilities and the flexibility to change/add to the mix in real time.

The concept paper Creating the CF of 2020 [4] identifies CDE and M&S as two critical tools for meeting the objectives outlined in the DND/CF: "*Given the uncertain nature of the future security environment, CDE and the integrated use of M&S are valuable tools for forging the best, flexible posture for the future with the resources available.*" [4]

Modelling and simulation will have significant effect in many areas of military interest: *"Advanced techniques in M&S encompass nearly the complete range of defence activities, including concept exploration, capability determination, acquisition, training, mission rehearsal, sustainability analysis, life-cycle/O&M/disposal cost determinations, etc."* [5] In particular, modeling and simulation will play key roles in:

- Acquiring new equipment (simulation-based acquisition – SBA), and in life-cycle management. Current economic conditions demand that acquisition of equipment be accomplished in a cost-effective manner with a minimum of risk.
- Training personnel. This area will continue to improve as technology advances to provide more realistic simulations.
- Operational support; facilitating what-if scenarios and rapid analysis of ongoing situations.

In his paper [6], LCdr Tunniclife recognized the cost savings to be made in these areas and made recommendations concerning the way ahead for modelling and simulation in the Canadian Navy. He also noted the utility of taking advantage of the expertise of different agencies, e.g. the physics-based modeling capabilities of Defence Research and Development Canada (DRDC), to maximize the potential benefits.

In its 2002 Technology Investment Strategy [7], Defence Research and Development Canada identified Simulation and Modelling for Acquisition, Requirements, Rehearsal and Training (SMARRT) as one of the 21 key activities necessary to develop the science and technology capacity that will enable the Canadian Forces to meet the challenges of the evolving global security environment and to operate in a coordinated and seamless manner with our allies. The investment strategy takes into account the recommendations of Defence Strategy 2020 [8] and the approach of Strategic Capability Planning [9] to meet the challenges of RMA including CDE.

In order to achieve the necessary standard of adaptability, flexibility, reuse, and interoperability called for, there needs to be an enabling infrastructure to facilitate integrated, distributed simulations.

Section 2 will describe software architectures that can provide the necessary infrastructure. These architectures – the High Level Architecture (HLA) and the Virtual Maritime System Architecture (VMSA) – define the features necessary to provide integrated distributed simulations. Section 3 is a short discussion of what the Virtual Naval Platform Group is capable of doing, and what the current intentions are. These include making use of seaway models [10] and time domain ship motion models [11] to provide platform motion information. Section 4 presents the recommendations arrived at after reviewing the pertinent documentation, and relevant conclusions are made in section 5.

This page intentionally left blank.

# 2. Architectures for Distributed Simulation

Simulations can take the form of live simulations where real people use real equipment to practice tactics and implement doctrine in realistic conditions. Modelling and simulation can also mean virtual simulations with humans-in-the-loop in a synthetic environment, with real or simulated equipment. It can also involve constructive simulations with simulated people, simulated equipment; and with or without human interaction.

At its root, computer modelling involves physics-based, computer models of single devices. These are the basic building blocks. These blocks can be combined to build system models to assess the given system's performance. Systems can also be combined to form larger systems or units where tactical effectiveness is the issue to be explored (e.g., a ship). Several of the larger systems may be used in conjunction with one another to provide an ability to investigate operational capabilities of the group (e.g., a task group). The building process can be continued to form an integrated simulation of a battle space.

In order to accomplish such high-end simulations there needs to be an agreed-upon structure and means of coordination/communication. At the level of a task force, or even at the level of a ship, the simulation will be too much for a single computing platform. Distributed simulation will be required.

## 2.1 High Level Architecture

High Level Architecture (HLA) is a framework to facilitate the design and implementation of integrated distributed simulations. Initially the US Department of Defense (DoD) Defense Modeling and Simulation Office (DMSO) developed HLA. It has since been approved as an open standard through the Institute of Electrical and Electronic Engineers: IEEE Standard 1516-2000 [12 - 14].

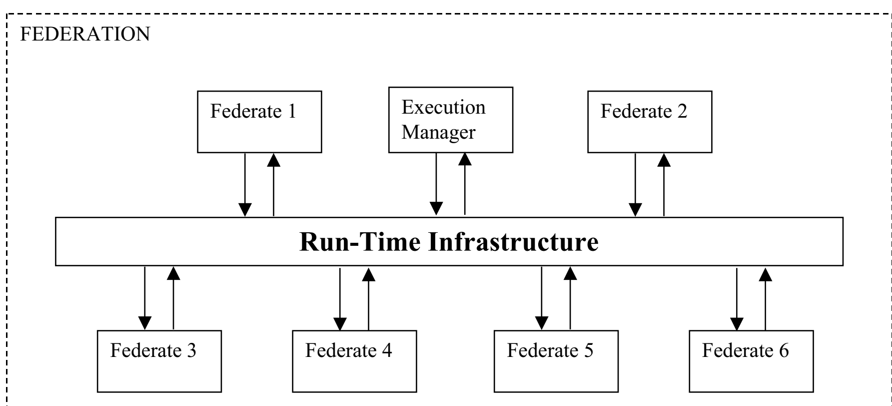


**Figure 1. Generic HLA Federation**

As envisioned by the HLA, an integrated simulation consists of two or more simulation models acting as *federates* linked over a network to form a *federation* (see Figure 1). Each of the federates may be in a different physical location, but they all interact in a common virtual environment. The federates interact by exchanging information via the Run-Time Infrastructure (RTI), software that provides the necessary services for such interaction. Types of federates that can be integrated into the simulation include passive data loggers, computer models (time-stepped, event-stepped, real-time, etc.), and interfaces to real systems (hardware–in-the-loop and man-in-the-loop).

HLA embodies a set of rules [12], a specification for federate interaction (the Run-Time Infrastructure, or RTI [13]), and some standard templates for component objects (Object Model Templates, or OMT [14]).

The rules describe the mandatory characteristics of federates and federations to ensure reuse and interoperability of the federates. There are five rules for the federation and five relating to the federates themselves:

A. Federation Rules [12]:

1. *Federations shall have an HLA Federation Object Model (FOM), documented in accordance with the HLA OMT.* The FOM will document all data to be exchanged in accordance with the HLA, including what data is to be exchanged and the minimum conditions for the exchange. The FOM is essential to federation definition. HLA is domain independent, therefore formal agreements for information exchange are important to ensure RTI initialization, as well as interoperability and reuse of federates.

2. *In a federation, all simulation-associated object instance representation shall be in the federates, not in the RTI.* Within a federation, several federates may be instances of the same object model (e.g. three ships may use the helm model, with each ship having its own version (instance) of the basic model). Federates own the federate-associated instance attributes (each helm instance uses its own parameters, independent of the other helm instances). The RTI may, however, own federation FOM-associated instance attributes, which it may use, but it does not change the instance attribute's data. Separation of federate-specific functionality from general-purpose infrastructure is key to the HLA concept.

3. *During a federation execution, all exchange of FOM data among joined federates shall occur via the RTI.* Federates shall identify to the RTI all information they require/provide, and instance attribute and interaction data corresponding to changing states of object instances. The RTI coordinates and synchronizes data exchange, ensuring delivery of the data and a common view of shared data, while the federates are responsible to ensure the right data is available at the right time. This will facilitate a coherent data exchange.

4. *During a federation execution, joined federates shall interact with the RTI in accordance with the HLA interface specification* [13]. This allows for independent development of the RTI and the federates.

5. *During a federation execution, an instance attribute shall be owned by at most one joined federate at any given time.* Ownership may be transferred dynamically during federation execution, but this constraint aids in ensuring coherent data across the federation. The ability to transfer ownership means enhanced flexibility for users.

B. Federate Rules [12]:

1. *Federates shall have an HLA Simulation Object Model (SOM), documented in accordance with the HLA OMT.* It will define the object classes, the class attributes, and the interaction classes of the federate that can be made public. Documenting federate capabilities is important to enable interoperability and reuse.

2. *Federates shall be able to update and/or reflect any instance attributes and send and/or receive interactions, as specified in their SOM.* The flexibility inherent in HLA allows federates to make internal object representations and interactions available for external use, but the capabilities for external interaction must be documented in the SOM. These capabilities shall include the obligation to export updated values of instance attributes calculated internally, and the obligation to be able to exercise interactions represented externally. These are really the details of federation – federate interaction, and are fundamental to ensure reusability.

3. *Federates shall be able to transfer and/or accept ownership of instance attributes dynamically during federation execution, as specified in their SOM.* The instance attributes must, however, be documented in the SOM. This rule is meant to enhance reusability.

4. *Federates shall be able to vary the conditions (e.g. thresholds) under which they provide updates on instance attributes, as specified in their SOM.* The conditions for update of instance attributes shall be documented in the SOM. Again this rule is intended to enhance reusability.

5. *Federates shall be able to manage local time in a way that will allow them to coordinate data exchange with other members of a federation.* For a given federation, the federates shall adhere to the time management approach of the federation. Therefore, federation designers must identify a time management approach as part of design implementation. This will enhance interoperability.

The infrastructure specification describes the services provided by the RTI and the interface to it. HLA is characterized by an implicit-invocation style of data exchange. Federates make data available and the RTI ensures delivery to the other, interested federates. Publish means intend to produce (send) data. Subscribe means intend to consume (receive) data. Federates declare their intentions to do one or both. The declarations are used by the RTI to transfer data received in accordance with the subscription prior to delivery. The RTI also uses declarations to inform federates of usefulness of published data (allows federate to cease producing data not being used). These services manage the producer-consumer relationships in terms of interactions and object classes.

The OMT is a structured means of describing the information exchanged between the federates. A simulation object model (SOM) catalogues the capabilities of a given federate; it provides a specification of the types of information an individual federate will provide or receive. A standard form of SOM facilitates the ready determination of the suitability of the federate to participate in the federation, since the required inputs and expected outputs can be examined prior to use of the federate. A federation object model (FOM) documents all objects and interactions in the federation; it provides a specification for standardized data exchange among federates at runtime. Data include enumeration of all pertinent object and interaction classes, and the attributes or parameters that characterize these classes. An HLA FOM establishes the “information model contract” that is necessary (but not sufficient) to achieve interoperability among the federates. The OMT facilitate interoperability among simulations and reuse of simulation components by providing a common basis for FOM and SOM.

These standards specify the necessary structure for a distributed simulation, but the implementation of the standards is left up to the application developer. The FOM is really application specific. It is where the details of the simulation scenario are defined. It will make use of an established RTI and several appropriate SOM. In general, each SOM should not have to be significantly altered to be used in a new or modified federation. For a given application, it may be possible to use an existing FOM, but it will be necessary to verify whether the FOM fits the simulation scenario, or if it needs to be modified, or even rejected. In some cases, the RTI may also have to be modified to add functionality that will accommodate the simulation. The goal again is to be able to use the basic building blocks (physics-based computer models) to build a variety of simulations.

## 2.2 Virtual Maritime System Architecture

The Virtual Maritime System Architecture (VMSA) [15], formerly known as the Virtual Ship Architecture (VSA), is an implementation of the original HLA intended to facilitate repeatability of initial conditions primarily for the sake of Monte Carlo simulations and statistical analysis. It is developed in the context of maritime operations, but facilitates joint air, land, (sea) surface, and subsurface operations. The guarantee of repeatable initial conditions requires additional constraints on joining and resigning from the federation, as well as more structured time-management procedures. The HLA allows federates to join or resign during the simulation. The VMSA requires that all federations be joined prior to the simulation start. Also, the HLA allows that a federate may choose to be time-constrained (advance of logical time is constrained by other federates), or time-regulating (advance of logical time regulates other federates), or both, or neither. Instructions are processed in either a Time-Stamp Order (TSO) or Receive Order (RO). In contrast, the VMSA requires that all federates be both time-constrained and time-regulating and that instructions be processed strictly in TSO for repeatable results.

Although it requires changes that reduce HLA capabilities in order to provide the new functionality, VMSA should be viewed as an implementation of the HLA, having more specific architecture. VMSA adds seven general rules and an execution management rule to ensure interoperability of the federates in the Virtual Maritime System.

A. Seven general rules [15]:

1. *When calling the AttributeValues, sendInteraction and deleteObjectInstance RTI methods, all federates shall provide the time for which these methods are valid in the tag, regardless of whether these methods are being sent Time Stamp Ordered (TSO) or Receive Ordered (RO).* The receiving (subscribing) federate will know whether the data is valid, even if the data is received RO.

2. *All federates shall respond to the provideAttributeValueUpdate callback. The response shall be in the form of an updateAttributeValues message sent RO, with the tag containing the time for which the values are valid.* If the federate is time regulating, it will send a (TSO) updateAttributesValues message containing the requested information as soon as possible. This may allow the receiving federate to make use of the data from sending federates having large timesteps, therefore not providing a TSO update for a significantly long period of time. The receiving federate can either wait for a TSO update or act on the values in the RO message.

3. *When an attribute is updated, an interaction is sent, or an object is deleted by a federate (either RO or TSO), the time for which the variables are valid is to be encoded into the tag of the message as a human readable string.* This rule provides the standard for the time tag of rule 1.

4. *When providing an update of kinematic attributes, which may be used in dead-reckoning algorithms, the update shall provide all the attributes up to the highest order time derivative required.* This rule removes the potential for lack of determinism in the dead-reckoning algorithm due to mismatch of initial conditions and/or lower order parameter values. The updates will take one of the following forms:

   Position
   Position, Velocity
   Position, Velocity, Acceleration
   Orientation
   Orientation, OrientationRate

Position, Orientation
Position, Orientation, OrientationRate
Position, Velocity, Orientation
Position, Velocity, Orientation, OrientationRate
Position, Velocity, Acceleration, Orientation
Position, Velocity, Acceleration, Orientation, OrientationRate

5. *All data shall be sent to the RTI in Big Endian format.* This rule specifies the byte ordering configuration for data communication and storage.

6. *Sending multi-element data shall be accomplished by concatenating the byte representations of the individual data together.* The ordering of data shall be in accordance with the VMS-FOM. This rule provides the standard for data interpretation.

7. *All enumerators are to be represented as 32 bits, in Big Endian format.* This is a further specification not present in the HLA OMT.

B. Execution management rule

*All object instances registered by a federate during a simulation must be deleted from the RTI at the end of the loop.* This is a cleanup rule to ensure the simulation is properly set up for multiple loops.

The developers of VMSA have chosen to impose additional constraints on the HLA such that there is one common VMS-FOM, which federates are required to comply with. The key items that constitute the VMSA are:

1. A description of the types of federates that make up the Virtual Maritime System [15];
2. The VMS-FOM [15];
3. A standard for coordinate usage [16];
4. The concept for execution management [17];
5. The concept for treatment of environmental effects;
6. The concept for treatment of propagation effects in sensors; and
7. The concept for treatment of interaction between active and passive sensors.

The use of a common FOM is very useful for guiding the development of the federates and the RTI. In the HLA, however, the rules were deliberately defined in such a manner as to allow federation developers the freedom to design a FOM that is suitable for the particular application, i.e., the FOM itself can be application-specific rather than common. The VMS-FOM adds rules to ensure repeatability and greater control of timing at the expense of that freedom. The risk involved with a common FOM is that applications will be shoehorned into the FOM instead of letting the FOM fit the application, which could be considered more desirable. Even in VMSA, it is not necessary that the common FOM be used; this has been a choice made in the interests of development in a common structural frame. If new functionality is required, the VMS-FOM will need to be modified.

A further requirement of the VMSA is that each federate deal with environmental effects (e.g., sea state, wind, etc.) internally, i.e., there is no environment federate. This is an arbitrary choice made by the developers of VMSA, as there is no technical reason preventing the use of an environment federate. Note also that an environment federate is possible for other HLA FOM, and could even be used with a modified VMS-FOM. The Virtual Combat System group at DRDC Atlantic is currently developing such a federate.

Figure 2 shows a typical VMSA federation with one or more virtual maritime systems. See Table 1 and Table 2 for the current implementation of the VMS-FOM Class table and Interaction table respectively. In the VMSA, every object class is a sub-class of the HLAObjectRoot and every interaction class is a sub class of the HLAInteractionRoot. The highlighted items in these tables are those objects and interactions that the motion federate of a virtual naval platform will affect or be affected by.

In Table 1, P means publish (i.e., the class will output its attributes for other classes to use), S means subscribe (i.e., the class needs inputs from other classes), N means neither. The CompositeEntity class indicates the type of platform and requires specification of the (operational) environment subclass (Air, SeaSurface, SubSurface, or Land) to completely define the platform. Every platform has an instance of the motion federate associated with it. Each platform federate instance will publish (P) and subscribe (S) to the appropriate data. Surface ships, submarines, and submersible vehicles are the most common platforms of interest at DRDC Atlantic. The ComponentEntity class defines the types of systems found on the platform. Of particular interest to the Warship Performance Section at DRDC Atlantic are the NavigationDirectionSystem (Helm) of the CommandAndControlSystem component, since this federate provides the command input to the motion federate, and the NavigationReportingSystem subclasses AltitudeReportingSystem, CourseReportingSystem, and KinematicReportingSystem, since these objects represent the output from the motion federate.

In the interaction table (Table 2), I means initiate (similar to publish, the interaction outputs parameters), R means react (similar to subscribe, the interaction takes parameters for inputs), and N means neither. The motion federate must deal with the (highlighted) ExectutionManagement interaction subclasses, as well as the SetPropulsionSystemAttribute and AckSetPropulsionSystemAttribute interactions.

## 2.3 Run-Time Infrastructure

The run-time infrastructure (RTI) provides the following services to facilitate the interaction between the federates:

- Federation management – required
- Declaration management – required
- Object management – required
- Ownership management – optional
- Time management – optional
- Data distribution management – optional

In addition, the RTI provides support services. The interface document also defines:

- Management Object Model (MOM)
- Programming language mappings –> API

Currently, most commonly, both HLA and VMSA use RTI 1.3NG (Next Generation).

## 2.4 Federates

The major advantage of the HLA and VMSA is that computer models can be developed in isolation from the federation, and even legacy code can be utilized with a (hopefully) minimum of rework. Since the RTI and the federates are developed independently of each other, it is the job of the FOM to manage the interaction of the federates over the RTI. It is clear that the RTI must have sufficient capability to support the required interactions. This capability is defined in the RTI specification, but the implementation of the specification is left to the developer of the RTI. Each federate must also be capable of taking in and putting out information in a suitable format for use in the federation. This capability must be clearly defined in the federate's SOM, but again, the implementation is left to the developer. The implementation of an appropriate FOM can therefore be seen to depend on the particular RTI used and the federates to be included. The circularity of the dependencies may lead to a somewhat iterative process, depending on whether or not an existing FOM is going to be used as the basis of the simulation.

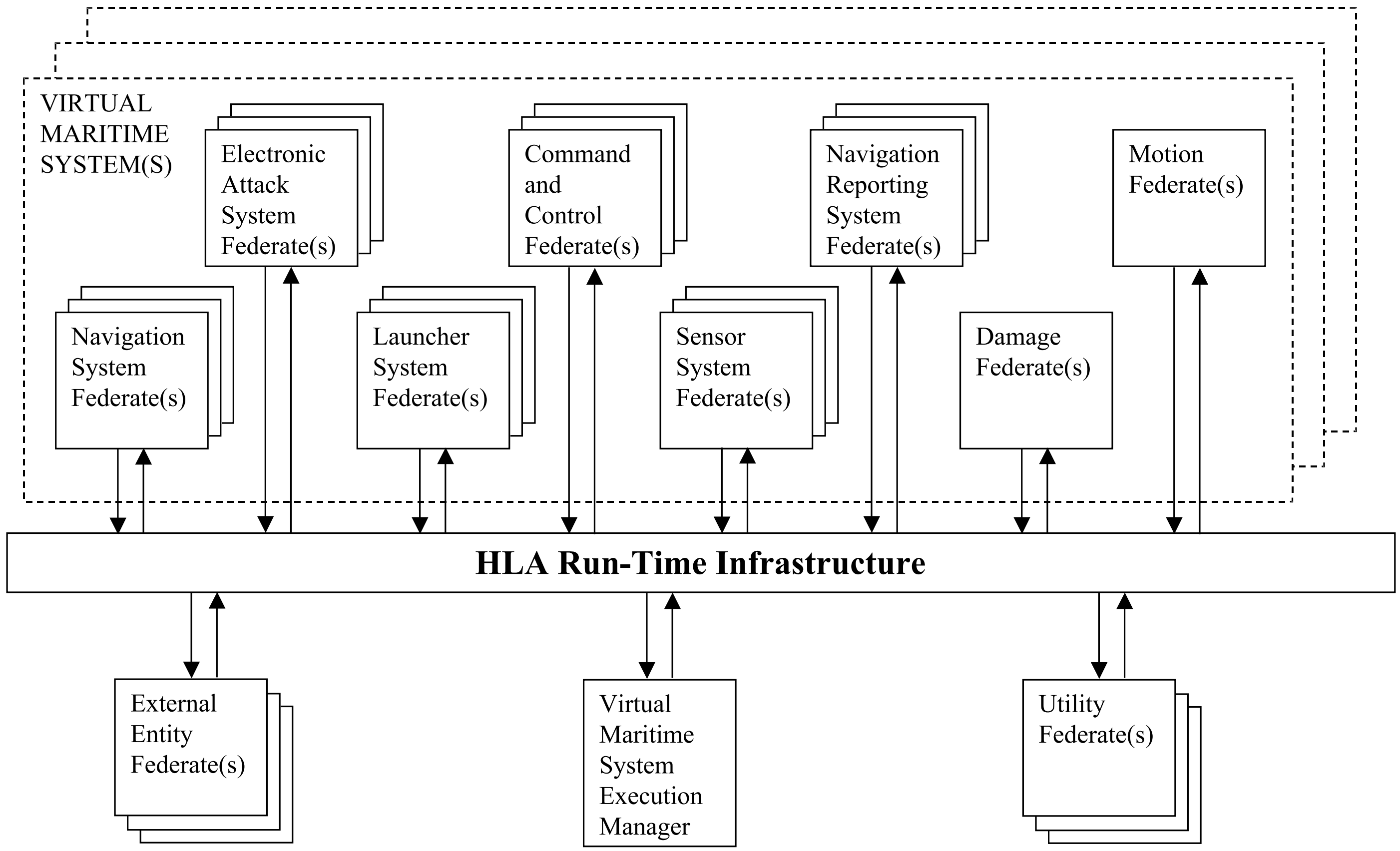


**Figure 2. Typical VMSA Federation [15]**

| | | | |
|---|---|---|---|
| HLAObject Root | [1] CompositeEntity (N) | [2] Air (P/S) | |
| | | [3] SeaSurface (P/S) | |
| | | [4] SubSurface (P/S) | |
| | | [51] Land (P/S) | |
| | [5] ComponentEntity (N) | [6] CommandAndControlSystem (P/S) | [36] CombatDirectionSystem (P/S) |
| | | | [37] NavigationDirectionSystem (P/S) |
| | | [7] EASystem (P/S) | |
| | | [8] LauncherSystem (N) | [33] EffectorLauncherSystem (P/S) |
| | | | [34] SensorLauncherSystem (P/S) |
| | | | [35] VTOLAircraftLauncherSystem (P/S) |
| | | [9] NavigationReportingSystem (N) | [38] AltitudeReportingSystem (P/S) |
| | | | [39] CourseReportingSystem (P/S) |
| | | | [40] KinematicReportingSystem (P/S) |
| | | [10] SensorSystem (N) | [11] ESSystem (P/S) |
| | | | [12] IRSystem (P/S) |
| | | | [13] IFFSystem (P/S) |
| | | | [14] RadarSystem (P/S) |
| | | | [15] SonarSystem (P/S) |
| | [16] NavigationReport (N) | [17] AltitudeReport (P/S) | |
| | | [18] CourseReport (P/S) | |
| | | [19] KinematicReport (P/S) | |
| | [20] SignalTask (N) | [41] ActiveTask (N) | [42] EOEATask (P/S) |
| | | | [43] RFEATask (P/S) |
| | | | [44] RadarTask (P/S) |
| | | | [45] ActiveSonarTask (P/S) |
| | | [46] PassiveTask (N) | [47] ESTask (P/S) |
| | | | [48] IRTask (P/S) |
| | | | [49] PassiveSonarTask (P/S) |
| | [21] Track (N) | [22] AbsoluteTrack (N) | [31] AbsoluteSystemTrack (P/S) |
| | | | [32] AbsoluteFusedTrack (P/S) |
| | | [23] RelativeTrack (N) | [24] RelativeSystemTrack (P/S) |
| | | | [25] RelativeFusedTrack (P/S) |
| | | | [26] RelativeESTrack (P/S) |
| | | | [27] RelativeIRTrack (P/S) |
| | | | [28] RelativeIFFTrack (P/S) |
| | | | [29] RelativeRadarTrack (P/S) |
| | | | [30] RelativeSonarTrack (P/S) |

**Table 1. Class Table Reconstructed From VMS-FOM OMT [15]**

| | | | |
|---|---|---|---|
| HLAInteraction Root | [1] ExecutionManagement (N) | [2] CreateCompositeEntity (I/R) | |
| | | [3] CreateComponentEntity (I/R) | |
| | | [4] SetScenarioDescription (I/R) | |
| | | [5] SetRandomNumberSeed (I/R) | |
| | | [6] ExecutionManagementError (I/R) | |
| | | [14] TerminateIteration (I/R) | |
| | | [15] TerminateFederation (IR) | |
| | [7] Launch (N) | [23] SelectEffector (I/R) | |
| | | [24] PowerUp (I/R) | |
| | | [25] AcknowledgeEffectorSelect (I/R) | |
| | | [26] EffectorReady (I/R) | |
| | | [27] InitiateEngagementPlan (I/R) | |
| | | [28] Engagement Plan (I/R) | |
| | | [29] ModifyEngagementPlan (I/R) | |
| | | [30] Fire (IR) | |
| | | [31] AcknowledgeFire (I/R) | |
| | [8] SensorDetection (N) | [32] ESDetection (I/R) | |
| | | [33] IRDetection (I/R) | |
| | | [34] IFFDetection (I/R) | |
| | | [35] RadarDetection (I/R) | |
| | | [36] SonarDetection (I/R) | |
| | [9] SystemControl (N) | [10] PropulsionSystemControl (N) | [11] SetPropulsionSystemAttribute (I/R) |
| | | | [12] AckSetPropulsionSystemAttribute (I/R) |
| | | [13] RadarSystemControl (N) | [21] SetRadarSystemAttribute (I/R) |
| | | | [22] AckSetRadarSystemAttribute (I/R) |
| | [16] TaskControl (N) | [37] RadarTaskControl (N) | [38] InitiateSearchTask (I/R) |
| | | | [39] InitiateTrackingTask (I/R) |
| | | | [40] InitiateIlluminationTask (I/R) |
| | | | [41] AckInitiateRadarTask (I/R) |
| | | | [42] DeleteRadarTask (I/R) |
| | | | [43] AckDeleteRadarTask (I/R) |
| | [17] TrackManagement (N) | [18] InitialTrack (I/R) | |
| | | [19] CorrectTrack (I/R) | |
| | | [20] DeleteTrack (I/R) | |
| | [44] Destroy (I/R) | | |

**Table 2. Interaction Table Reconstructed From VMS-FOM OMT [15]**

This page intentionally left blank.

# 3. Virtual Naval Platform Application

## 3.1 Virtual Combat Systems Work

The Virtual Combat Systems (VCS) Group at DRDC Atlantic has been using the VMSA to develop a federation for use in a Virtual Battle Experiment Canadian-1 (VBE CA-1). The experiment (which will involve multiple simulation runs) has two main goals. The first is to capture knowledge about the reasoning process operators use to make association choices[1]. The second goal is to ascertain the effects of using relative (as opposed to absolute) tracking data from other virtual (allied) platforms within the scenario to form an assessment of the tactical situation.

The VCS Group is also working on an environment federate within an experimental VMSA-FOM. It is possible that this federate could be incorporated in the real VMSA-FOM in the future.

## 3.2 Virtual Naval Platform Work

In the context of the integrated battle scene, the key responsibility of the Virtual Naval Platform (VNP) Group is to provide motion information of a fidelity suitable for the scenario. This may include motion in the presence of damage to the platform and consequent taking on of water.

The motion federate may be as simple as a single simulation providing position information, or it may be as complex as a sub-federation that provides detailed information about the motion of the platform with varying damage conditions and in wind and waves. The trade-off, as usual, will be between fidelity and bandwidth: higher fidelity will require a much more computationally intensive simulation and more communication bandwidth on the RTI.

The VMSA[2] federates applicable to VNP are the motion federate (see Table 3 and Table 4), the helm (navigation direction system) federate (see Table 5 and Table 6), and the damage federate. The motion federate is of course the prime concern, being the place where platform kinematics and dynamics are determined. Both the helm and the damage federates will provide input to the motion federate.

[1] The operator is presented with a time-bearing plot, which will have one or more tracks. When there are multiple tracks, they may represent one or more target vessels. It is the operator's task to decide which of these tracks belong to the same vessel and associate them. This association means that the associated tracks are now treated as one track from one vessel. The operator may see a hundred tracks on the screen when there are only 10 vessels, so the ideal solution would see the tracks associated into 10 groups (or less, since some targets might never be detected).

[2] The federates could be similar for the HLA; the federates in VMSA a possible implementation.

## 3.3 Motion Federate SOM

Currently the VMSA motion federate SOM is a "bare-bones" implementation where the only object information recorded is the type of composite entity of the platform (see Table 3), and a simple set of interactions (see Table 4).

| | | |
|---|---|---|
| HLAObject Root | [1] CompositeEntity | [2] Air |
| | | [3] SeaSurface |
| | | [4] SubSurface |
| | | [51] Land |

**Table 3 Motion Federate SOM Class Table**

| | | | |
|---|---|---|---|
| HLAInteraction Root | [1] ExecutionManagement | [2] CreateCompositeEntity | |
| | | [3] CreateComponentEntity | |
| | | [4] SetScenarioDescription | |
| | | [5] SetRandomNumberSeed | |
| | | [6] ExecutionManagementError | |
| | | [14] TerminateIteration | |
| | | [15] TerminateFederation | |
| | [9] SystemControl | [10] PropulsionSystemControl | [11] SetPropulsionSystemAttribute |
| | | | [12] AckSetPropulsionSystemAttribute |
| | [44] Destroy | | |

**Table 4 Motion Federate SOM Interaction Table**

## 3.4 Helm Federate SOM

As with the motion federate, the helm federate has a minimal implementation at present (see Table 5 and Table 6).

| | | | |
|---|---|---|---|
| HLAObject Root | [1] CompositeEntity | [2] Air | |
| | | [3] SeaSurface | |
| | | [4] SubSurface | |
| | [5] ComponentEntity | [6] CommandAndControlSystem | [39] NavigationDirectionSystem |

**Table 5 Helm Federate (Navigation Direction System Federate) SOM Class Table**

| | | | |
|---|---|---|---|
| HLAInteraction Root | [1] ExecutionManagement | [2] CreateCompositeEntity | |
| | | [3] CreateComponentEntity | |
| | | [4] SetScenarioDescription | |
| | | [5] SetRandomNumberSeed | |
| | | [6] ExecutionManagementError | |
| | | [11] TerminateIteration | |
| | | [12] TerminateFederation | |
| | [7] SystemControl | [8] PropulsionSystemControl | [9] SetPropulsionSystemAttribute |
| | | | [10] AckSetPropulsionSystemAttribute |

**Table 6 Helm Federate (Navigation Direction System Federate) SOM Interaction Table**

## 3.5 Factors Affecting Naval Platform Motion

In order to provide realistic information about the motion of a ship in a seaway, several factors both internal to the motion derivation and external must be taken into account. The factors internal to motion federate include:

- Nominal hydrodynamic coefficient/hydrostatic properties
- Seaway conditions
- Wind
- Interactions with other ships – proximity, wakes, etc.
- Damage condition
- Operational limits

The platform motion is also affected by information that is generated external to motion federate, such as:

- Helm commands:
- Launch of weapons
- Damage condition/event

These factors may be more or less relevant to any given simulation.

A VNP motion federate could be developed to work within VMSA or HLA, though the former may have more stringent requirements.

## 3.6 Virtual Naval Platform as a Sub-Federation

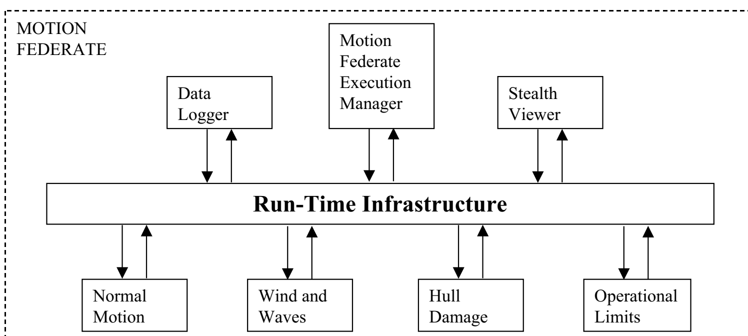


**Figure 3 . Motion Sub-Federation**

The sub-federation concept (see Figure 3) has a two-fold purpose: it allows support for larger simulations where the platform motion is only one federate among many; and it also facilitates a stand-alone simulation, where the objective is to determine platform motion in the face of sea conditions and/or damage states. Operational limits can also be defined as a separate federate based on model states (e.g., silent running) and simulation events (e.g., hull compromise event). This facilitates interactive changes to motion characteristics with changes in state and/or condition.

Alternatives to a sub-federation include multiple motion federates running concurrently, each representing a damage condition and running state, but only the federate representing the current state/condition is actively providing output. This approach will require more computing power, possibly one computing platform for each variation of the motion federate. Another possible solution is to script the changes in condition and or state. This approach inhibits reuse of the federate since the motion federate will be specific to the scripted series of events.

## 3.7 Way Ahead

The Virtual Naval Platform Group of DRDC Atlantic's Warship Performance Section will build motion federates for surface vessels using a library of time domain (and frequency domain) seaway [10] and platform motion [11] objects called ShipMo3D. The seaway can be modeled internally to the motion federate as required by VMSA, or it may be modeled as a separate federate. The motion federate or sub-federation can be used in an integrated distributed simulation, or it may be used off-line for engineering analysis (e.g. for fatigue analysis with specific damage conditions on a platform in a seaway).

Initially, the intent is to simulate a simplified replenishment at sea (RAS) scenario to look at tension on a cable strung between the ships in varying sea states (see Figure 4). The major simplification in early work will be to ignore seaway interactions between the ships, i.e., each ship will behave as if it were not in the presence of the other ship.

**Figure 4. VSHIP Visualizer Depiction of a Replenishment At Sea (RAS)**

# 4. Recommendations

The Virtual Platform Group at DRDC Atlantic should develop a motion federate complete with seaway effects as a model comprised of ShipMo3D time-domain objects. The use of the ShipMo3D library allows modular construction of the federate based on the requirements of the federation. To start, it is better to use the existing VMSA and the associated VMS-FOM, since there is more structure in place. To begin at the HLA level would mean developing a similar structure to be able to implement the motion federate. A federate based on a different HLA implementation can be developed at a later date. The VMS-FOM is also a good choice because the Virtual Combat Systems Group has chosen it for their simulation. Therefore the motion federate developed by the VNP Group will be immediately useful to them.

As well, DRDC Atlantic should develop the motion federate and a separate seaway federate, both consisting of time-domain objects from the ShipMo3D library of objects, where the federates interact as a sub-federation (see Figure 3). This configuration will also allow the seaway data to be used by other federates. Although a separate seaway federate is not currently envisioned in the VMS-FOM, it may be useful in the future or in other HLA implementations. It also adds flexibility to defining motion parameters for a given simulation; the seaway can be modified independently from the motion algorithms. Work on this type of motion simulation has already begun in the form of a simplified RAS scenario involving two ships in a seaway with a cable between them.

DRDC Atlantic should develop a new structural analysis federation for independent, off-line analysis (i.e. not as part of any operational/tactical simulation) of a ship in a seaway. The goal of the federation is to investigate the effects of damage on platform performance, especially in terms of seakeeping and manoeuvrability. This will allow simulation of many "what-if" scenarios, and could lead to a large database of ship behaviours that may be used in tactical and operational simulations.

This page intentionally left blank.

# 5. Conclusions

The US and NATO are committed to the use of HLA to provide integrated, combined, joint simulation environments for the development of tactics and doctrine as well as for training and systems acquisition. Canada is following suit and looking to HLA to provide the structure for a range of modeling and simulation tasks. Within the HLA there is the flexibility to model a wide variety of situations and scenarios, especially if the federation object model is not rigidly defined, but designed/adapted for the particular application.

The Australian VMSA is a viable implementation of the HLA. By choice, some of the freedoms of the HLA are traded for more stringent time management and guaranteed repeatability of the initial conditions. Typical use of the VMSA involves a standard FOM – the VMS-FOM. The VMSA provides more specific instruction to federation developers than the HLA alone.

The initial role of DRDC Atlantic's VNP group in either the HLA or the VMSA is to provide motion information for the platform(s) being simulated. In the VMSA, the motion federate will have to include any environmental interactions (wind and wave effects) integral to the federate. In other implementations of the HLA it may be possible to separate the platform motion from the environmental effects such that they each constitute a federate of their own. This separation may have added value for other federates that may wish to use the seaway information.

This page intentionally left blank.

# 6. References


1. RMA Operational Working Group. (1999). Canadian Defence Beyond 2010: An RMA Concept Paper. Canadian Department of National Defence.

2. Under Secretary of Defense for Acquisition and Technology. (1995). Modeling and Simulation (M&S) Master Plan. (DoD 5000.59-P). United States Department of Defense.

3. North Atlantic Treaty Organization. (1998). NATO Modelling and Simulation (M&S) Master Plan. (AC/323 (SGMS)D/2 Version 1.0).

4. Symposium Working Group, Strategic Capability Planning Working Group. (2000). Creating the CF of 2020. Canadian Department of National Defence.

5. Symposium Working Group, Strategic Capability Planning Working Group. (2000). Modelling and Simulation: Enabling the Creation of Affordable, Effective 2020 Canadian Forces. Canadian Department of National Defence.

6. Tunnicliffe, LCdr. M. (2001). Technologies for workload and crewing reduction - Phase 1 Project report. (DCIEM TR 2001-109). October 2001.

7. DRDC. 2002. Technology Investment Strategy 2002.

8. Shaping the Future of the Canadian Forces: A Strategy for 2020. June 1999.

9. Strategic Capability Planning for the Canadian Forces. June 2000.

10. McTaggart, K. A. (2003). Modelling and Simulation of Seaways in Deep Water for Simulation of Ship Motion. (DRDC Atlantic TM 2003-190). Defence Research and Development Canada – Atlantic.

11. McTaggart, K. A. (2003). Hydrodynamic Forces and Motions in the Time Domain for an Unappended Ship Hull. (DRDC Atlantic TM 2003-104). Defence Research and Development Canada – Atlantic.

12. IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Framework and Rules. IEEE Std 1516-2000.

13. IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Federate Interface Specification. IEEE Std 1516.1-2000.

14. IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Object Model Template (OMT) Specification. IEEE Std 1516.2-2000.

15. Canney, S.A. (2002). Virtual Maritime System Architecture Description Document - Issue 2.00. Australian Defence Sciences & Technology Organisation (DSTO).

16. Best, J.P. (2002). Coordinate Usage in the Virtual Maritime System Architecture - Issue 2.10. Australian Defence Sciences & Technology Organisation (DSTO).

17. Cramp, A. (2002). Execution Management in the Virtual Maritime System Architecture - Issue 2.02. Australian Defence Sciences & Technology Organisation (DSTO).

# List of Acronyms

| | |
|---|---|
| ACD | Advanced Concept Development |
| API | Application Programmer Interface |
| CAS | Chief of Air Staff |
| CDE | Concept Development and Experimentation |
| CLS | Chief of Land Staff |
| CMS | Chief of Maritime Staff |
| DoD | (US) Department of Defense |
| DND | Department of National Defence |
| DRDC | Defence Research and Development Canada |
| DSTO | (Australian) Defence Science and Technology Organisation |
| ECDE | Environment Concept Development and Experimentation |
| FOM | Federation Object Model |
| HLA | High Level Architecture |
| IEEE | Institute of Electrical and Electronics Engineers |
| JE | Joint Experimentation |
| OMT | Object Model Template |
| M&S | Modelling and Simulation |
| MSMP | M&S Master Plan |
| MOM | Management Object Model |
| NATO | North Atlantic Treaty Organization |
| RBA | Revolution in Business Affairs |
| RO | Receive Order |

| | |
|---|---|
| RMA | Revolution in Military Affairs |
| RTI | Run-Time Infrastructure |
| SBA | Simulation-Based Acquisition |
| SOM | Simulation Object Model |
| TSO | Time-Stamp Order |
| VCS | Virtual Combat Systems |
| VMSA | Virtual Maritime System Architecture |
| VNP | Virtual Naval Platform |
| VSA | Virtual Ship Architecture (archaic) |
| WP | Warship Performance |

# Glossary

| Technical term | Explanation of term |
| --- | --- |
| Simulation Model Space | A general term describing the synthetic depiction of the real (or projected) world provided by a model, simulation or federation. |
| Model | A representation of a system, entity, phenomenon, or process. Software models consist of data and algorithms. |
| Simulation | The execution over time of the models representing one or more entities or processes. Includes human-in-the-loop simulations. |
| Interoperability | The ability of a model or simulation to provide services to, and accept services from, other models and simulations and to use these services to enable them to operate together. |
| Reuse | The use of resources for purposes beyond those for which they were originally developed. Reuse can occur within an organization, in different organizations, or in different application areas. |

This page intentionally left blank.

## 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. Centre sponsoring a contractor's report, or tasking agency, are entered in section 8.)<br><br>Defence R&D Canada - Atlantic<br>PO Box 1012<br>Dartmouth, NS, Canada B2Y 3Z7 | 2. SECURITY CLASSIFICATION (overall security classification of the document including special warning terms if applicable).<br><br>UNCLASSIFIED |

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

Review of Architectures for Simulation of Virtual Naval Platforms

4. AUTHORS (Last name, first name, middle initial. If military, show rank, e.g. Doe, Maj. John E.)

Perrault, Douglas E.

| | | |
|---|---|---|
| 5. DATE OF PUBLICATION (month and year of publication of document)<br><br>September 2003 | 6a. NO. OF PAGES (total containing information Include Annexes, Appendices, etc).<br><br>42 | 6b. NO. OF REFS (total cited in document)<br><br>17 |

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).

Technical Memorandum

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include address).
Defence R&D Canada - Atlantic
PO Box 1012
Dartmouth, NS, Canada B2Y 3Z7

| | |
|---|---|
| 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).<br><br>Project 11GK19 | 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.)<br><br>DRDC Atlantic TM 2003-193 | 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
(  ) Defence departments and defence contractors; further distribution only as approved
(  ) Defence departments and Canadian defence contractors; further distribution only as approved
(  ) Government departments and agencies; further distribution only as approved
(  ) 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).

Unlimited

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), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual).

In order to meet the rapidly changing requirements for military action, Canada and many of its allies have recognized the utility of modeling and simulation for simulation-based acquisition, training, and operational/tactical experimentation. Canada is looking to High Level Architecture (HLA) to provide the flexibility to model a wide variety of situations, including maritime operations. The Virtual Maritime System Architecture (VMSA) is a viable implementation of the HLA. The initial role of the Virtual Naval Platform (VNP) Group at DRDC Atlantic in either the HLA or the VMSA is to provide motion information for the platform(s) being simulated. This document examines both the HLA and the VMSA architectures in order to determine how best to create motion federates to act in the variety of proposed simulations. After a thorough investigation of the relevant documents describing each architecture, it was determined that the VNP group can develop motion federates for either architecture, but that there are differences in philosophy between the two that will affect the implementation of the motion federate.

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 not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title).

HLA

VMSA

Architecture

Modelling and Simulation

Federation Object Model

Motion Federate

Virtual Maritime System

Virtual Ship

Defence R&D Canada

R & D pour la défense Canada

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

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

DEFENCE R&D DÉFENSE

www.drdc-rddc.gc.ca