Copy No._____

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

DEFENCE R&D DÉFENSE

Horizon 3 Torpedo Launcher Plug-in

User Guide and Technical Description

Tania E. Wentzell
DRDC Atlantic

Lisa Goold
Innovation Science Pty Ltd.

Defence R&D Canada – Atlantic

Technical Memorandum
DRDC Atlantic TM 2005-209
December 2005

Canada

Horizon 3 Torpedo Launcher Plug-in

User Guide and Technical Description

Tania E. Wentzell
DRDC Atlantic

Lisa Goold
Innovation Science Pty Ltd

Defence R&D Canada – Atlantic

Technical Memorandum

DRDC Atlantic TM 2005-209

December 2005

Author

Tania E. Wentzell and Lisa Goold

Approved by

A/SH/MICS for.

J. S. Kennedy

Head, Maritime Information and Combat Systems

Approved for release by

Kirk Foster

DRP Chair

The information contained herein has been derived and determined through best practices and adherence to the highest levels of ethical, scientific, and engineering investigative principles. The reported results, their interpretation, and any opinions expressed therein, remain those of the authors and do not represent, or otherwise reflect, any official opinion of position of DND or the Government of Canada.

Abstract

This paper describes the torpedo launcher plug-in that has been added to the Horizon 3 Command and Control (C2) framework. Horizon is a central component of the distributed experimentation environment used by the Virtual Combat System (VCS) Group at Defence Research and Development Canada – Atlantic (DRDC Atlantic). This plug-in provides an interface for the Horizon operator to prepare a torpedo for launch and to keep track of weapon inventory. It sends a launch interaction to another component of the distributed simulation environment which in turn creates and controls the torpedo entity. This plug-in was developed to support the requirements of a Canadian Forces-wide exercise called War-in-a-Box (WIB) that aims to connect platform simulators and computer simulations distributed across the country using the High Level Architecture (HLA). This launcher plug-in also increases the capabilities of the VCS Group's simulation environment for future demonstrations and experiments.

Résumé

Ce document décrit le module d'extension de tube lance-torpilles, qui a été ajouté au cadre de commandement et de contrôle (C2) d'Horizon 3. Horizon est un élément central de l'environnement d'expérimentation réparti utilisé par le groupe VCS (Système de combat virtuel) à Recherche et développement pour la défense Canada – Atlantique (RDDC Atlantique). Le module d'extension fournit une interface à l'opérateur d'Horizon pour préparer le lancement d'une torpille et contrôler les stocks d'armes. Il établit une interaction de lancement avec un autre élément de l'environnement de simulation réparti, lequel crée et contrôle la torpille. Ce module a été développé afin de répondre aux besoins d'un exercice pour l'ensemble des Forces canadiennes appelé WIB (War-in-a-Box), qui vise à relier des simulateurs de plates-formes et des simulations informatiques répartis à l'échelle du pays en utilisant l'architecture HLA (High-Level Architecture). En outre, le module accroît les capacités de l'environnement de simulation du groupe VCS aux fins de futures démonstrations et expériences.

This page intentionally left blank.

Executive Summary

Report title

Wentzell, T.E. and Goold, Lisa; 2005; Horizon 3 Torpedo Launcher Plug-in: User Guide and Technical Description; DRDC Atlantic TM 2005-209; Defence R&D Canada – Atlantic; Unclassified.

Background

The Horizon 3 Command and Control (C2) framework [1] is a key component of the distributed experimentation environment used by the Virtual Combat System (VCS) Group at Defence Research and Development Canada – Atlantic (DRDC Atlantic). Horizon provides a display for entities and sensor tracks as well as tools for managing these tracks.

The VCS Group's experimentation environment [2] uses the Virtual Maritime Systems Architecture (VMSA) [3], an application and extension of the High Level Architecture (HLA). While the VMSA architecture supports weapon modelling, the group's work to date has focused more on sensor modelling.

Principal results

This plug-in extends the simulation capabilities that can be offered by the VCS Group for demonstration and experimentation purposes to include support for hostile attacks using passive/active acoustic homing torpedoes. It enhances an existing C2 framework by including an interface for preparing a torpedo launch. The launcher sends a request into the HLA world for another pre-determined, capable federate to create, launch and control the torpedo.

Significance of results

DRDC Atlantic is participating in a Canadian Forces-wide exercise called War-in-a-Box (WIB) that aims to connect both platform simulators and computer simulations distributed across the country using HLA. DRDC Atlantic is contributing a red force submarine to the exercise, which will be simulated through a VMSA federation using various VMSA federates. Prior to the development of this plug-in, there was no existing VMSA implementation which would allow the submarine to launch torpedoes – a requirement of a hostile, enemy sub. With this plug-in, an operator on the simulated submarine can now decide where and when to launch a torpedo.

Future work

Future enhancements to this plug-in may include modification of the interface to support additional types of torpedoes and missiles, including those that require operator control post-launch. As well, allowing an operator to launch weapons from platforms other than the ownship could prove interesting, at least from a research point of view.

Sommaire

Titre du rapport

Wentzell, T.E. and Goold, Lisa; 2005; Horizon 3 Torpedo Launcher Plug-in: User Guide and Technical Description [Guide d'utilisation et description technique du logiciel module de tube lance-torpilles d'Horizon 3]; RDDC Atlantique TM 2005-209; R & D pour la défense Canada – Atlantique; non classifié.

Contexte

Le cadre de commandement et de contrôle (C2) d'Horizon 3 constitue un élément clé de l'environnement d'expérimentation réparti utilisé par le groupe VCS (Système de combat virtuel) à Recherche et développement pour la défense Canada – Atlantique (RDDC Atlantique). Horizon permet l'affichage d'entités et de trajectoires détectées, et offre des outils pour la gestion de ces trajectoires.

L'environnement d'expérimentation du groupe VCS utilise l'architecture VMSA (Virtual Maritime Systems Architecture), qui est une application et une extension de l'architecture HLA (High Level Architecture). Bien que l'architecture VMSA permette la modélisation d'armes, jusqu'à maintenant les travaux du groupe ont porté surtout sur la modélisation de capteurs.

Principaux résultats

Ce module d'extension augmente les capacités de simulation offertes par le groupe VCS à des fins de démonstration et d'expérimentation en incluant des outils d'appui pour des attaques au moyen de torpilles à tête chercheuse acoustique passive/active. Il améliore un cadre C2 existant en fournissant une interface pour la préparation du lancement d'une torpille. Le lanceur envoie dans l'environnement HLA une demande à une autre composante logicielle prédéterminée ayant les capacités requises afin que cette composante crée, lance et contrôle la torpille.

Portée des résultats

RDDC Atlantique participe à un exercice pour l'ensemble des Forces canadiennes appelé WIB (War-in-a-Box), qui vise à relier des simulateurs de plates-formes et des simulations informatiques répartis à l'échelle du pays en utilisant l'architecture HLA. RDDC Atlantique fournit aux fins de cet exercice un sous-marin des forces ROUGES, qui sera simulé au moyen d'un logiciel fédérateur VMSA utilisant diverses composantes VMSA. Avant le développement du module d'extension précité, il n'existait aucune application VMSA permettant au sous-marin de lancer des torpilles – ce qui est nécessaire pour un sous-marin ennemi. Avec ce nouveau module, un opérateur situé à bord du sous-marin simulé peut maintenant décider où et quand lancer une torpille.

Recherches futures

De futurs perfectionnements du module d'extension pourront comprendre la modification de l'interface en vue de permettre le lancement d'autres types de torpilles et de missiles, y compris ceux qui doivent être contrôlés par l'opérateur après le lancement. En outre, il pourrait être intéressant de permettre à un opérateur de lancer des armes à partir d'autres plates-formes que le navire observateur, du moins du point de vue de la recherche.

This page intentionally left blank.

Table of contents

List of figures

List of tables

This page intentionally left blank.

1. Introduction

The Horizon 3 Command and Control (C2) [1] framework is a key component of the distributed experimentation environment used by the Virtual Combat System (VCS) Group at Defence Research and Development Canada – Atlantic (DRDC Atlantic). Horizon provides a display for entities and sensor tracks as well as tools for managing those tracks (e.g. association, fusion, deletion, labelling, filtering, etc.). It is well-suited for use within a research environment as it can be quickly modified to incorporate new ideas or test new concepts using its plug-in architecture.

The VCS Group's experimentation environment [2] uses the Virtual Maritime Systems Architecture (VMSA) [3], an application and extension of the High Level Architecture (HLA). The VMSA Federation Object Model (FOM) defines the information and format of information that can be shared amongst various simulation components (federates) in the environment.

A VMSA plug-in to Horizon allows Horizon to read in and interpret entity and sensor track data created by other VMSA federates (otherwise known as data 'subscription') and communicate information about tracks that are fused within Horizon back out to the VMSA world for other federates to use (data 'publishing').

DRDC Atlantic is participating in a Canadian Forces-wide exercise called War-in-a-Box (WIB) that aims to connect both simulators (e.g., the Operational Flight Tactical Trainer (OFTT)) and computer simulations (e.g., the Joint Semi-Automated Forces (JSAF) and Joint Conflict and Tactical Simulation (JCATS) constructive simulations) distributed across the country using HLA. DRDC Atlantic is contributing a red force submarine to the exercise, which will be simulated through a VMSA federation composed of various VMSA federates (motion, helm, passive sonar, Target Motion Analysis (TMA), Electronic Support Measures (ESM), and Horizon) and hosted at the CFMWC.

The WIB team has prescribed the details of the scenario that will be used during this exercise. The scenario calls for the red force submarine to launch torpedoes at target platforms which will be generated by JSAF. To date, the development of the simulation environment at DRDC Atlantic has focused on sensor capabilities and not attack capabilities; thus, there is no VSMA simulation component to model the path or detonation of a torpedo. However, JSAF does include such a capability. It was decided that while the decision to launch the torpedo would rest with a controller on the simulated red submarine, JSAF could be used to actually create and control the torpedo and determine its success on the submarine's behalf. To achieve this, development of an interface from which to launch the torpedo was required.

Within Horizon, the operator has a plan view of its location and sensor contacts – necessary information for launching a torpedo. By placing the torpedo launching interface within Horizon, the operator can interactively choose the target locations in the plan display.

Design ideas for the look and functionality of the plug-in interface were obtained through conversations with Subject Matter Experts (SMEs) at DRDC Atlantic, review of JSAF's torpedo modelling capability and requirements [4], and the weapon launching interfaces in two commercial games, Dangerous Waters [5] and 688i Hunter/Killer [6]. This initial version of the launcher plug-in was designed specifically to complement JSAF's capabilities. JSAF, which can also be used as a stand-alone simulation, allows the operator, prior to the torpedo launch, to enter a launch bearing and range from ownship at which to activate the torpedo's sensors. These attributes have been reflected in the Horizon plug-in implementation. All other details regarding the torpedo entity will be controlled automatically within JSAF.

Despite JSAF's influence on the design of the launcher, the plug-in is not tied to JSAF in any way. Any other model (federate) with similar torpedo modelling capabilities could be used in its place – this is the beauty of HLA. For this reason, this document does not discuss what happens once the launch interaction is sent, since this is specific to the destination federate. It discusses everything up to and including the operator clicking the 'fire' button. The document is divided into two main sections; a guide for the operator using the plug-in, and a technical description for the scenario developer or programmer configuring the plug-in and interpreting the VMSA interaction it publishes.

2. Guide to Using and Understanding the Plug-in

2.1 Purpose

This Horizon plug-in provides the operator with an interface for initializing, loading and triggering the launch of torpedoes from the ownship. It can also be used to monitor weapon inventory. This plug-in is intended to for use within an HLA environment where a separate simulation component has the capability to actually create, launch and control the torpedo according the specifications defined by the launcher plug-in.

2.2 Assumptions

Some assumptions were made to simplify the initial version of a torpedo launcher plug-in. These assumptions are as follows:

1. The amount of time to load a torpedo, regardless of the storage rack, launch tube and torpedo type, is constant;
2. The time to flood and equalize a tube is independent of torpedo type;
3. Any torpedo can be launched from any tube;
4. All tubes are in the nose of the sub and have the same launch position relative to the host platform's position; torpedoes of a particular type have equal capability and probability of reaching a given target regardless of the launch tube;
5. There is no requirement to unload a loaded torpedo;
6. Tubes are not automatically reloaded. They must be reloaded after firing if they are to be used again; and
7. Torpedoes will be launched one at a time. However, they can be launched one directly after another by selecting multiple 'Fire' buttons in sequence.

2.3 Initialization Data

Prior to using the plug-in within the Horizon environment, the following information must be pre-set by the scenario designer:

1. Number and type of pre-loaded torpedoes;
2. Number and type of torpedoes stored in the racks;
3. Minimum range from ownship for sensor activation for each torpedo type;
4. Maximum range from ownship for sensor activation for each torpedo type;
5. Default minimum range from ownship for sensor activation for a general torpedo (to be used if #3 is not set);
6. Default maximum range from ownship for sensor activation for a general torpedo (to be used if #4 is not set);
7. Time to flood a tube, drain a tube, and equalize a tube;
8. Time to open/close the muzzle door; and
9. Time to load a torpedo from the rack into a tube.

Details on modifying the information in the initialization file are provided in the technical section (section 3) of this document.

2.4 Launch Preparation

In order to activate the torpedo launcher plug-in, the map display in Horizon must be centred on ownship. (To achieve this, simply click on the 'Ownship' button in the Map Controls menu on the right-hand side of the Horizon screen).

Figure 1 shows the interface for loading a torpedo into a tube and defining the path that the torpedo is meant to follow. It consists of the following elements, as indicated:

1. Drop-down list of all available torpedo types[1];
2. Type of torpedo loaded in the tube;
3. Number of the torpedo tube;
4. Button to **F**lood the tube;
5. Button to **E**qualize the pressure in the tube;
6. Button to open the **M**uzzle door;
7. Button to FIRE the loaded torpedo;
8. Bearing path for the torpedo to follow when launched;
9. Range from ownship at which the torpedo's sensors will be activated (i.e., the expected range of the target);
10. Button to open the weapon inventory table; and
11. Button to move back to the previous process.

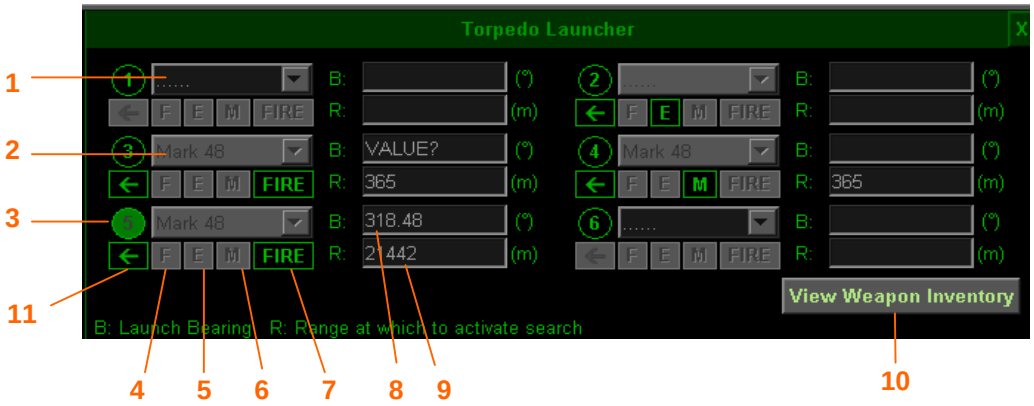

Figure 1. *Torpedo Launcher Plug-in Interface*

[1] Torpedo types are 'available' if there is at least 1 torpedo of that type stored in the racks. If all torpedoes of a particular type are already loaded in tubes, that type will not appear in the list, as it is assumed the torpedo can be launched from the tube it is already in and would not be moved.

2.4.1 Set Launch Bearing and Sensor Activation Range

Before the operator can launch a torpedo, the bearing (B) and sensor activation range (R) must be set and fall within the predefined values. Both a minimum sensor activation range and a maximum range may be defined by the scenario developer for a given torpedo type. All ranges are measured with respect to ownship. A minimum sensor activation range may be set to prevent the torpedo from searching for a target too soon and accidentally seeking its ownship. A maximum range is set to prevent the operator from having an unrealistic expectation of the range the torpedo is capable of reaching. If a torpedo type has not yet been selected from the drop-down box, the default values will be used to restrict the entries to valid entries only. Values outside of these extremes are considered invalid and if entered, the range box will be reset to the appropriate extreme value (minimum or maximum range).

There are two ways to set the bearing and range:

1. Click in the bearing and range boxes and manually enter the information using the keyboard. Note that the units are shown to the right of the boxes, and will typically be in degrees and metres respectively. Alternatively,

2. Click in either the bearing or range box to activate the tool for visually setting the two values for the corresponding torpedo tube. A line will appear in Horizon's plan view extending from ownship out to the maximum range of the chosen torpedo type (or to the default maximum range if a torpedo type has not yet been chosen). The line can be rotated to achieve the desired bearing. A sliding ball on the line can be moved along the line to choose the desired activation range. The slider will not slide closer to ownship than specified by the minimum activation range parameter. The current tube number is shown at the end of the line. This method is illustrated in Figure 2. Note that in this initial version of the plug-in, it is not possible to select a specific target track to launch the torpedo at. While this would be ideal for the operator, it was not well suited for the implementation of the WIB exercise. Such an implementation requires both the launch-requesting software (Horizon) and the torpedo-modelling software (JSAF) to maintain the same target tracks. In the case of WIB, all submarine sensors are modelled within VMSA and tracks are displayed in Horizon; these tracks are entirely unknown to JSAF.

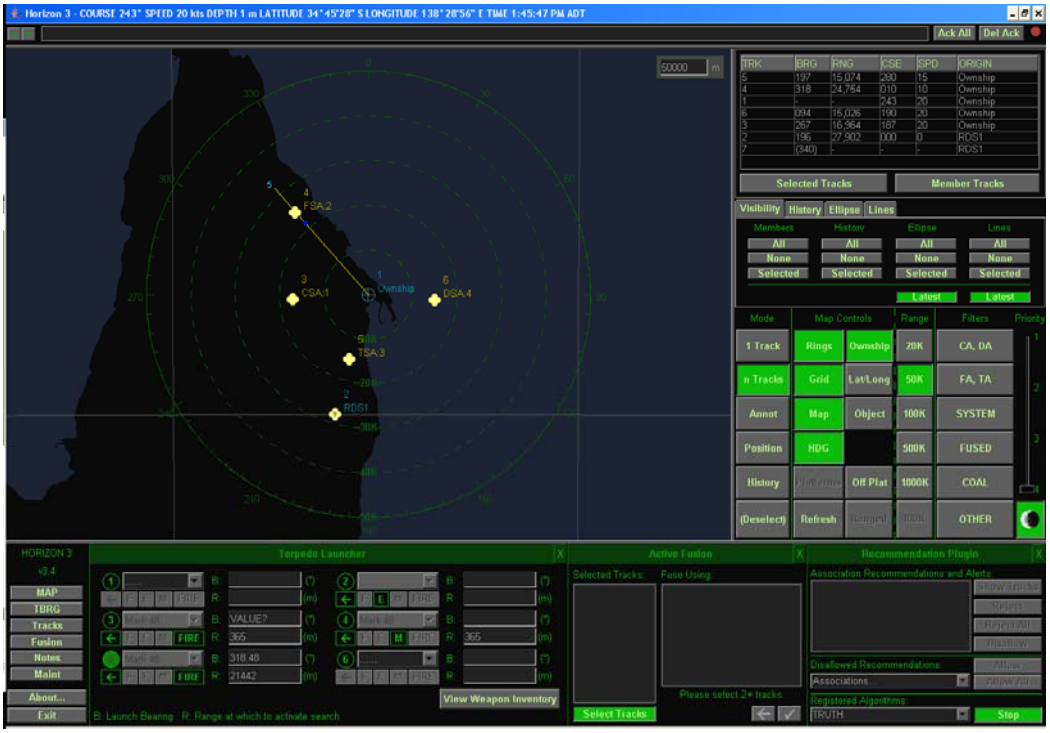

Figure 2. *Interactive Launch Plan Designation*

2.4.2 Load a Torpedo

To load a torpedo into an empty tube, select the torpedo from the corresponding drop-down list. (BE WARNED: Once the torpedo loading process is underway, it cannot be stopped, and while there is no requirement to further prepare the torpedo for launching, the tube cannot be unloaded without firing it.) The name of the chosen torpedo will appear next to the tube number, and the tube number flashes until the torpedo is completely loaded. There is no restriction on loading more than one tube at a time.

2.4.3 Prepare to Fire Torpedo

Before launching a torpedo, the following steps must be completed in sequence:

1. Click the F button to flood the tube;
2. Click the E button to equalize the pressure; and
3. Click the M button to open the muzzle door.

Flooding the tube, equalizing the pressure, and opening the door all require a predefined amount of time (as defined in the initialization file). Clicking the

F , E , or M button results in the button flashing until the appropriate amount of time has passed.

The availability of the buttons will indicate the stage of the launch process.

2.4.4 Fire Torpedo

To launch a torpedo, the FIRE button must be clicked twice. After the first click, it will turn red: FIRE . This color change is meant to warn the operator that the next time the button is clicked, the torpedo will be launched.

2.4.5 Prepare Tube for Reloading

Before loading another torpedo into the tube that a torpedo has been launched from, the steps above must be completed in reverse, as indicated here:

1. Click the M button to close the muzzle door,
2. Click the E button to equalize the pressure,
3. Click the F button to drain the tube.

Alternatively, if the scenario developer has set the 'auto reinitialize after fire' parameter in the plug-in's initialization file to true, these three processes will occur in sequence automatically after launching.

Following the completion of this process, the drop-down list of torpedoes available for loading becomes available again.

2.5 Interpreting the Launch Display

The availability of buttons can be used to infer the stage of the launch or post-launch process. For example, the following conclusions can be drawn from Figure 1:

1. Tube 1 is ready to be loaded;
2. A torpedo has been launched from tube 2 and the tube is now empty. The muzzle door has been closed, however the pressure has not been equalized following the launch;
3. Tube 3 is loaded with a Mark 48 torpedo. The tube is flooded, the pressure has been equalized, and the muzzle door is open. A bearing value must be entered before the FIRE button will respond to a click. The range value is currently set to its default value, which is the shortest range at which the torpedo can be safely activated;
4. Tube 4 is loaded with a Mark 48 torpedo. The tube is flooded and the pressure has been equalized;
5. Tube 5 is loaded with a Mark 48 torpedo. The tube is flooded, the pressure has been equalized, and the muzzle door is open. The bearing and range have been set and the FIRE button is ready to be clicked; and
6. Tube 6 is ready to be loaded.

2.6 Torpedo Inventory

By clicking on the View Weapon Inventory button, an inventory table appears in the launcher plug-in display window, indicating the number of torpedoes of each type already loaded in the tubes and the number of torpedoes of each type stored on the racks.

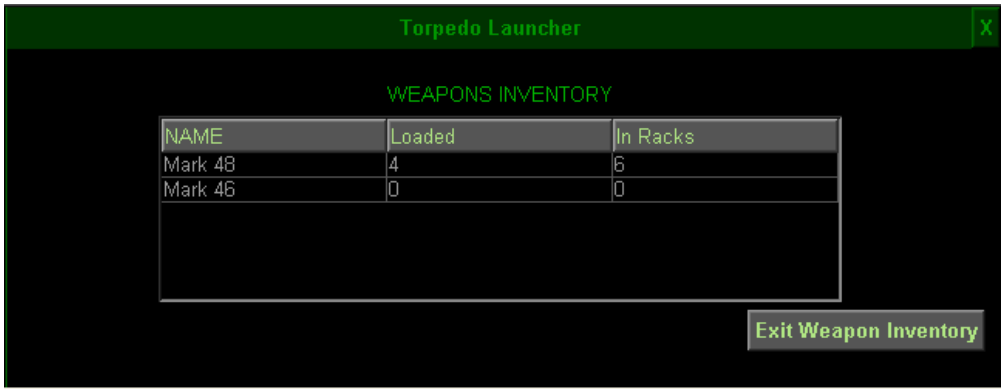

Figure 3. *Inventory Table*

Click on the Exit Weapon Inventory button to return to the main plug-in display.

3. Installing and Initializing the Torpedo Launcher Plug-in

3.1 Software Versions

This plug-in was designed to be compatible with Horizon 3 version 3.4 and VMSA FOM v3.3.5. Compatibility with previous versions of both Horizon and the VMSA FOM has not been explored or tested.

3.2 Plug-in Registration

To add a plug-in to Horizon 3 version 3.4, it must be registered in the plugin.ini initialization file. This file is used to tell Horizon which plug-ins should to be made available to the particular Horizon instance.

Add the following lines to the plugin.ini file:

```
[Torpedo Launcher]
class                                                    =
au.com.iscience.navy.horizon.torpedo.TorpedoLauncherPlugIn
```

The VMSA plug-in is likely already registered. However, check for the text below and if it is not found in the plugin.ini file, add it as well:

```
[Horizon VMSA Plugin]
class = au.com.iscience.navy.trackfusion.federate.VMSAPlugIn
number of instances = 1
```

3.3 GUI Setup

The Horizon display consists of a number of sockets that can display the Graphical User Interface (GUI) of various plug-ins. The 'main' plug-in socket typically displays a map or time-bearing chart which can be manipulated using other plug-ins in 'standard' sockets. In some cases, the size of a standard socket is not large enough to display all of the information pertinent to a particular plug-in. In this case, a 'wide' socket can be used. In the case of the launcher plug-in, the socket is twice the width of a standard socket.

The XML-based GUI Manager (gui.xml) needs to be informed that the interface to support the new plug-in requires such a wide socket. To do this, add the following to the 'WideSkt1' layout defined in the gui.xml file:

```
<content name="Torpedo Launcher"/>
```

3.4 Torpedo Launcher Initialization File

The torpedo launcher initialization file (torpedolauncher.ini) allows the scenario developer to set initial conditions and defaults, define torpedo types, define the effect of user actions within the GUI interface, and more.

The initialization file, as shown in Annex A, includes a 'General' category (indicated by `[General]`) and a 'Defaults' category (indicated by `[Defaults]`). All other categories (i.e. all other phrases contained within [ ] brackets) are assumed by Horizon to define a type of torpedo.

3.4.1 General

The following parameters are found within the General category:

- **source platform** - The name of the source platform from which the torpedoes will be launched. The current implementation only supports "Ownship". Future implementations may utilise this parameter to allow the operator to launch torpedoes from other platforms.

- **destination** - The destination torpedo launcher subsystem, to which the torpedo launch command should be sent.

- **slaved to map instance** – The map instance with which the plug-in interacts. This parameter should be of the form `<pluginName>$<instanceNumber>`.

- **number of torpedo tubes** – The number of torpedo tubes displayed on the plug-in's GUI.

- **number of torpedo tube columns** – The number of columns of torpedo tubes displayed on the plug-in's GUI.

- **tube order** - The ordering of torpedo tubes in the GUI. Tubes are row-wise ordered. For example if there are two columns, 6 torpedo tubes and the tubes are ordered 1, 2, 3, 4, 5, 6, the first row would display tube 1 and then tube 2, the second row would display tube 3, then tube 4 and the third row would display tube 5 and then tube 6.

- **auto reinitialize after fire** - Indicates whether the torpedo tube should automatically reinitialize after firing a torpedo. If "true", after firing a torpedo, the muzzle door will be closed, the pressure equalized, and the tube drained. The corresponding buttons will flash to indicate what stage of the re-initialization process is taking place. If "false" the tube must be reinitialized manually.

- **time to flood tube** – The time that it takes to flood a torpedo tube (in milliseconds).

- **time to drain tube** – The time that it takes to drain a torpedo tube (in milliseconds).

- **time to equalize tube** – The time that it takes to equalize the pressure in a torpedo tube (in milliseconds).

- **time to open/close muzzle door** – The time that it takes to open/close the muzzle door (in milliseconds).

- **time to load torpedo** – The time that it takes to load a torpedo into a tube (in milliseconds).

3.4.2 Defaults

The following parameters are found within the Defaults category:

- **default minimum range to arm** – The default minimum range at which a torpedo's sensor system can be safely activated (in metres). Used if a torpedo category does not define a minimum range to arm.

- **default maximum range** – The default maximum travelling range of a torpedo (in metres). Used if a torpedo category does not define a maximum range.

3.4.3 Torpedo Categories

The name of the category indicates the torpedo type. For example, when creating a Mark 48 torpedo, an appropriate category name would be `[Mark 48]` or `[Mk-48]`. The phrase within the [ ] will be displayed in the drop-down list of torpedoes found in the GUI interface of the plug-in. The following parameters must be defined for each type of torpedo:

- **minimum range to arm** – The minimum range a torpedo of this type must be from the ownship before it's sensor system can be safely activated (in metres).

- **maximum range** – The maximum travelling range of a torpedo of this type (in metres).

- **number on racks** – The number of torpedoes of this type stored on the racks. This is the total number of torpedo of this type available.

- **loaded** - The tubes into which torpedoes of this type are loaded at simulation initialization time. This is a comma-delimited list of tube numbers (i.e., 1,2,5).

3.5 Configuration of Colours

The colours used for the launch bearing line, activation point marker and tube number within the map are configurable via the "Torpedo Launcher Layer" category of the Horizon framework's colors.ini file. Each color is defined using an 8-digit hexadecimal code, AARRGGBB, where AA represents the transparency of the color (the 'Alpha' code), and RR, GG, and BB represent the red, green, and blue components of the color. The "Torpedo Launcher Layer" colors category includes the following parameters:

- **bearing line** – The colour of the bearing line that is extended from ownship to indicate the fire bearing.

- **activation position** – The colour of the circular marker located on the bearing line that indicates the position at which the torpedo's sensors will be activated.

- **tube number** – The colour of the tube number text located at the end of the bearing line.

3.5.1 Example Torpedo Launcher Layer Colours category

```
[Torpedo Launcher Layer]
 bearing line = 0xffffff00
 activation position = 0xff0000ff
 tube number = 0xff00ffff
```

In this example, the bearing line is yellow, the activation position is blue, and the tube number is aqua.

4. The HLA Details

4.1 Implementation

Prior to the development of the launcher plug-in, DRDC Atlantic was already using a 'VMSA plug-in' with the Horizon software to allow the exchange of data between Horizon and a VMSA simulation. More specifically, the VMSA plug-in allows Horizon to subscribe to and display surface, subsurface and air entities, as well as track objects created and updated within a VMSA federation. It also allows Horizon to publish fused tracks back into the VMSA federation.

The torpedo launcher plug-in requires additional communication with the VMSA environment, so the VMSA plug-in was modified to support this. The possibility of building VMSA communication directly into the launcher plug-in was considered, but quickly discounted as it would mean that there would then be two HLA plug-ins (acting as HLA federates) required to use the launcher. Coordination of the plug-ins could get out of hand if multiple plug-ins started exchanging data with components outside of Horizon, not to mention the fact that each plug-in would in this case require an RTI federate licence, which could be very expensive depending on the RTI being used. Instead, the existing VMSA plug-in was enhanced to include an additional VMSA interaction to support the launcher plug-in. So, the torpedo launcher plug-in provides the interface for the user to set up the launch and when the fire button is clicked this plug-in sends the information (internal to Horizon) to the VMSA plug-in, indicating what to publish to the VMSA environment.

4.2 VMSA Interactions

Table 1 describes the VMSA interaction that has been added to the VMSA plug-in in order to meet the requirements of the torpedo launcher. (Note that this same information is passed internal to Horizon from the launcher plug-in to the VMSA plug-in when the fire button is clicked, though in a different format.)

Table 1. *Relevant VMSA Interaction for Torpedo Launcher Plug-in*

INTERACTION	PARAMETERS	PARAMETER MEANING
ControlMessage. LaunchControl. Launch (P)	InitiatorObjectInstanceName	Identifies the object instance that has initiated the interaction.
	ControlMessageID	Provides an identifier of this message that is unique to the sender (e.g., 1, 2, 3, ...).
	Plan	The details of the launch plan. The format for this plan is agreed to by the initiating and receiving federates during federation design.
	EntityName	The name of the entity to launch. (This parameter is left blank to be assigned by the receiving federate).

The Plan parameter of the Launch interaction seems somewhat contrary to the spirit of HLA. HLA aims to have all data exchange formats specified in the FOM. In this case, however, the VMSA FOM includes this free format parameter which turns out to be quite convenient because it means that the information that must be passed out of Horizon to the VMSA federation can be custom-defined. Note that it is important that prior to federation run time, all subscribing federates are aware of the exact format of the Plan. The required data elements of the Plan agreed upon for the purposes of WIB are described in Table 2. This Plan information will be passed in XML format.

Table 2. *Required Data Elements within the Plan Parameter*

DATA ELEMENT	MEANING
WeaponType	The type of weapon being launched.
TubeNumber	The tube number from which the weapon is being launched.
LaunchBearing	The bearing at which to launch the torpedo, relative to the platform launching the torpedo.
WeaponActiveRange	Range at which to turn on the weapon's sensors.

Note that while that the TubeNumber has been included in this message, it does not currently have an effect on the performance of the torpedo. It is included to support future implementations of this plug-in that may wish to consider the exact launch location. Also for the purposes of future plug-in development, the VMSA plug-in subscribes to the VMSA ControlMessage.LaunchControl.Response.LaunchResponse interaction which includes Status and Outcome parameters. It forwards these to the Horizon event system as a custom event: au.com.iscience.navy.trackfusion.federate.launch.LaunchControlResponseEvent. Interested plug-ins can register a listener with Horizon's State Data module in order to receive launch response notifications.

4.2.1 Plan Parameter XML format

The initial implementation of the Plan Parameter has been defined in XML format as follows:

```
<?xml version=\"1.0\"?>
<launch>
      <weapontype>XXX</weapontype>
      <tubenumber>9</tubenumber>
      <launchbearing>999.99</launchbearing>
      <weaponactiverange>99999</weaponactiverange>
</launch>
```

5. Future Work

The initial plug-in implementation described in this document was chosen to meet the immediate requirements of the WIB demonstration. For WIB, passive/active homing torpedoes will be created and controlled by JSAF, so the available options in JSAF defined how the launcher plug-in interface was set up.

In the future, to allow other federates with different requirements or capabilities than JSAF to be able to receive and respond to meaningful launch request interactions from Horizon, the interface should be enhanced to manage more options. For example, the existing implementation assumes the use of a passive/active acoustic homing torpedo. However, there may be a desire to use other types of torpedoes, such as wire-guided torpedoes which would require operator control of the torpedo post-launch. Another example is to provide the option to select a target track rather than a bearing and sensor activation point. This would be a useful enhancement if another federate existed which could accept and deal with such information. The same interface could also be used for launching ship-to-ship missiles, since there is no requirement for the ownship to be a submarine. As well, in the current implementation, weapons can only be launched from the ownship. This could be modified to allow an operator to launch weapons from an alternative platform, such as an Unmanned Undersea Vehicle (UUV) or a third-party coalition platform.

6. References

1. Arnold, Ashley and Haddy, Michael, *Horizon 3 - Operator Guide, Release 2.0*, Innovation Science Pty Ltd, Australia, December 2001.

2. Wentzell, T.E. and Mark G. Hazen, *Virtual Maritime Experimentation*, DRDC Atlantic Technical Memorandum 2004-119, 2004.

3. Canney, Shane A., *Virtual Maritime Systems Architecture Description Document Issue 2.00*. Virtual Maritime System Document 00034, Defence Science and Technology Organization, Australia, 2002.

4. *Joint Semi-Automated Forces (JSAF) Software*, Multinational Experiment 3 (MNE3) version, obtained through Canadian Forces Maritime Warfare Center (CFMWFC), April 2005.

5. *Dangerous Waters User Manual (Section 10:Kilo Stations)*, Sonalysts Combat Simulations, 2004.

6. Hanscom, Valerie, et. al., *688(I) Silent/Hunter Manual*, Jane's Combat Simulations, 1998.

Annex A: torpedolauncher.ini file

Note: The numbers in this configuration file are not necessarily representative of the real world. They are place holders, included for clarity of the file format.

```
[General]
# The name of the source platform that the launcher is associated
# with.  This is the platform from which the torpedoes will be
# launched.
source platform = Ownship

# The destination torpedo launcher subsystem, to which torpedo launch
# command should be sent (for example, via VMSA).
destination = LaunchControl

# Interaction is made with the following map instance
slaved to map instance = Map Plot$1

# The number of torpedo tubes.
number torpedo tubes = 6

# The number of columns of torpedo tubes in the GUI.
number torpedo tube columns = 2

# The ordering of torpedo tubes in the GUI.  Tubes are rowwise
# ordered.
tube order = 1,2,3,4,5,6

# Indicates if the torpedo tube should automatically reinitialize
# after firing a torpedo.
auto reinitialize after fire = false

# The time that it takes to flood a tube (in milliseconds).
time to flood tube = 1000
# The time that it takes to drain a tube (in milliseconds).
time to drain tube = 15000
# The time that it takes to equalize the pressure in a tube (in
# milliseconds).
time to equalize tube = 1000
# The time that it takes to open/close the muzzle door (in
# milliseconds).
time to open/close muzzle door = 500
# The time that it takes to load a tube (in milliseconds).
time to load torpedo = 3000

[Defaults]
# The default minimum range a torpedo takes to arm (in metres).
default minimum range to arm = 365
# The default maximum range of a torpedo (in metres).
default maximum range = 32000
```

```
[Mark 48]
# The minimum range torpedoes of this type take to arm (in metres).
minimum range to arm = 365
# The maximum range of torpedoes of this type (in metres).
maximum range = 38000
# The number of torpedoes of this type stored on the racks.
number on racks = 0
# The tubes into which torpedoes of this type are loaded at
#  initialization  time.   This  is  a  comma-delimited  list  of  tube
numbers
# (i.e.  1,2,5)
loaded =
```

List of symbols/abbreviations/acronyms/initialisms

C2	Command and Control
CFMWC	Canadian Forces Maritime Warfare Center
ESM	Electronic Support Measures
FOM	Federation Object Model
GUI	Graphical User Interface
HLA	High Level Architecture
JCATS	Joint Conflict and Tactical Simulation
JSAF	Joint Semi-Automated Forces
MNE3	Multi-National Experiment 3
MSECO	Maritime Synthetic Environment Coordination Office
OFTT	Operational Flight Tactical Trainer
SME	Subject Matter Expert
TMA	Target Motion Analysis
UUV	Unmanned Undersea Vehicle
VCS	Virtual Combat System
VMSA	Virtual Maritime Systems Architecture
WIB	War-in-a-Box

Distribution list

LIST PART 1:

5 DRDC ATLANTIC LIBRARY FILE COPIES
1 T.E. WENTZELL
1 M.G. HAZEN
1 A.G. GILLIS
1 H/MICS

9 TOTAL LIST PART 1

LIST PART 2: DISTRIBUTED BY AUTHOR

1 LISA GOOLD, Innovation Science Pty Ltd, PO Box 333, Mitcham, South Australia 5062, Australia

1 TOTAL LIST PART 2

LIST PART 3: DISTRIBUTED BY DRDKIM 3

2 NDHQ/ ADMS&T/ DRDKIM 3
1 CFMWC/MSECO/P. Dietz, PO Box 99000, Halifax, NS B3K 5X5

3 TOTAL LIST PART 3

13 TOTAL COPIES REQUIRED

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.)

Defence R&D Canada – Atlantic
PO Box 1012
Dartmouth, NS, Canada B2Y 3Z7

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,R or U) in parentheses after the title).

Horizon 3 Torpedo Launcher Plug-in: User Guide and Technical Description

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

Tania E. Wentzell and Lisa Goold

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

December 2005

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

20

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

6

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).

TECHINICAL 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).

11bk

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 Atlantic TM 2005-209

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).

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).

This paper describes the torpedo launcher plug-in that has been added to the Horizon 3 Command and Control (C2) framework. Horizon is a central component of the distributed experimentation environment used by the Virtual Combat System (VCS) Group at Defence Research and Development Canada – Atlantic (DRDC Atlantic). This plug-in provides an interface for the Horizon operator to prepare a torpedo for launch and to keep track of weapon inventory. It sends a launch interaction to another component of the distributed simulation environment which in turn creates and controls the torpedo entity. The motivation behind the development of this plug-in is to support the requirements of a Canadian Forces-wide exercise called War-in-a-Box (WIB) that aims to connect platform simulators and computer simulations distributed across the country using the High Level Architecture (HLA). This launcher plug-in also increases the capabilities of the VCS Group's simulation environment for future demonstrations and experiments.

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).

Horizon
C2
Modelling & Simulation
War-in-a-Box

This page intentionally left blank.

Defence R&D Canada

Canada's leader in defence
and National Security
Science and Technology

R & D pour la défense Canada

Chef de file au Canada en matière
de science et de technologie pour
la défense et la sécurité nationale

www.drdc-rddc.gc.ca