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

DEFENCE R&D DÉFENSE

Evolution of the Synthetic Environment in the FFSE Section

Survey of Current Usage of Synthetic Environment within FFSE and Recommendations for the Future

Nacer Adbellaoui, Paul Hubbard and Oliver Schoenborn

Defence R&D Canada – Ottawa
TECHNICAL MEMORANDUM
DRDC Ottawa TM 2005-141
November 2005

Canada

Evolution of the Synthetic Environment in the FFSE Section

Survey of Current Usage of Synthetic Environments within FFSE and Recommendations for the Future

Nacer Abdellaoui and Paul Hubbard
Defence R&D Canada – Ottawa

Oliver Schoenborn
Greenley & Associates

Defence R&D Canada – Ottawa

Technical Memorandum

DRDC Ottawa TM 2005-141

November 2005

Abstract

As part of a project on the Exploitation of Synthetic Environments (Project #13jb), this document summarizes a data-gathering effort to measure the current synthetic environment usage in the Future Forces Synthetic Environment section. Eleven simulation instances, ranging from a two-week, human-in-the-loop rehearsal to a 30-second stand-alone computation, are described and measured in term such as accessibility, complexity, agility and stability.

The problems that limit the exploitability of the current simulation network are also discussed, and various solutions are proposed. Some of these solutions will be implemented within the 13jb project.

Résumé

Ce document décrit conceptuellement le projet « Exploitation du Réseau de Simulation », FFSE – ARP 13jb, en particulier la tâche *Spécification/Implémentation du logiciel outil*. Ce document introduit d’abord l’organisation et le projet, établit l’état actuel du « Réseau de Simulation » à travers la description d’une douzaine de systèmes de simulation disponibles, discute des problèmes limitant l’exploitabilité du réseau de simulation, propose différentes sortes de solutions et élabore les solutions qui seront développées dans la première phase du projet.

This page intentionally left blank.

Executive summary

The Synthetic Environment Technology Group of the Future Forces Synthetic Environment (FFSE) section of DRDC-Ottawa uses a variety of modeling and simulation tools to create virtual experiments and exercises for DND, to assist in concept development, and to accept or reject hypotheses underlying military tactics and doctrines.

The simulation tools and experiments are quite diverse in scale, architecture, fidelity and purpose. They are also, in general, poorly documented, difficult to set up and, as a consequence, sometimes difficult to access and use. This document is a result of project on the Exploitation of Synthetic Environments (Project #13jb), which seeks to make the science and technology products at FFSE more accessible, usable, less time intensive, and more comprehensive, by identifying the major bottlenecks that interfere with the exploitability of simulation components.

As a first step, eleven simulation tools and experiments are identified that represent a reasonable sampling of FFSE usage. These range from a two-week rehearsal of a live experiment to a stand-alone computation that takes only thirty seconds to execute. By interviewing principal subject matter experts and researching available documentation, the eleven instances have been characterized and are briefly described.

The bottlenecks interfering with exploitability have been identified as network set-up, accessibility, stability and agility. Network set-up problems are determined to be predominantly administrative barriers that can be solved by establishing a stream-lined, documented process for connecting geographically distributed sites.

Accessibility problems stem from the complexity and diversity of the simulation instances within the section in combination with a relative lack of knowledge of their functionality and use, e.g. start-up processes. It is proposed that accessibility problems be addressed first by a documentation task that includes web access to a central repository, and second, by a reconfigurable GUI portal from which documentation can be accessed and, whenever possible, associated software can be triggered. Scalable

Stability refers to the unpredictable level of operational functionality at any given time or in any simulation instance. It is proposed that, along with specific improvements to certain components, the section move to an open architecture design for all simulation instances, so that unstable components can be replaced or upgraded as needed.

Agility refers to the lack of compatibility between simulation components and the lack of adaptability of individual components. Solutions to agility problems caused by scalability are proposed, as well as interoperability solutions such as federation gateways or interface mapping tools.

A parallel task underway in the 13jb project is seeking to compare several tools for computer-generated forces and synthetic environments that, along with the recommendations herein, will help define the future evolution of the synthetic environment.

Abdellaoui, N., Hubbard, P. and Schoenborn, O. 2005. *Evolution of the Synthetic Environment in the FFSE Section*. DRDC Ottawa TM 2005-141. Defence R&D Canada – Ottawa.

Sommaire

Ce document décrit conceptuellement le système à développer dans le cadre du projet ARP 13jb « Exploitation du Réseau de Simulation », en particulier la tâche « Spécification et Implémentation d'un Logiciel Outil ».

Le groupe d'environnement synthétique de la section Environnement Synthétique pour les Forces Futures (ESFF) de RDDC Ottawa emploie une variété d'outils de modélisation et de simulation pour créer des expériences et des exercices virtuels pour DND, dans le but d'aider au développement de concept, d'évaluer la validité des hypothèses affectant la tactique et les doctrines militaires, et pour la formation du personnel militaire.

Les outils et les expériences de simulation sont à tous les niveaux très variables: au niveau de l'échelle de la simulation, de l'architecture, de la fidélité ainsi que des objectifs. Ces simulations sont, généralement, mal documentées, difficiles à installer ou à faire fonctionner et, par conséquent, difficile à utiliser et à exploiter.

Le but du projet d'exploitation du réseau de simulation est d'améliorer l'accessibilité et l'usage du système, ainsi que la réduction de l'effort requis pour exploiter les ressources disponibles et l'élargissement de la gamme des services offerts, en identifiant les obstacles majeurs qui limitent l'exploitabilité des ressources de ces différentes composantes de simulation.

Ayant interrogé les experts des différents outils, exercices et expériences, onze instances de simulations ont été identifiées et caractérisées. Ces instances sont décrites brièvement, pour donner une vue d'ensemble des ressources disponibles dans le réseau de simulation.

Les problèmes d'exploitabilité ont été identifiés comme suit: installation de réseau, accessibilité, stabilité et agilité. Les problèmes d'installation de réseau sont principalement administratifs et peuvent être résolues en établissant un processus documenté pour interconnecter les différents lieus géographiquement séparés.

Les problèmes d'accessibilité proviennent de la complexité et de la diversité des exemples de simulation dans la section en combinaison avec un manque relatif de la connaissance de leur fonctionnalité et utilisation, e.g. processus de mise en train. On lui propose que des problèmes d'accessibilité soient adressés d'abord par une documentation chargent qui inclut l'accès d'enchaînement à un dépôt central, et en second lieu, par un portail reconfigurable de GUI duquel la documentation peut être consultée et, autant que possible, le logiciel associé peut être déclenché.

La stabilité se rapporte à l'imprévisibilité de la fonctionnalité opérationnelle de la simulation. Ce qui est proposé, en plus des améliorations spécifiques de certains composants, est la migration à une conception d'architecture ouverte pour les différentes simulations, de sorte que des composants instables puissent être remplacés ou améliorés si nécessaires.

L'agilité se résume en manque de compatibilité entre les composants de simulation ainsi le manque d'adaptabilité de ces différents composants. Les solutions sont proposées aux problèmes d'agilité causés par la taille de la simulation (en termes de nombre d’entités), ainsi que ceux causés par le manque d'interopérabilité. Ces solutions sont principalement création de portails entre fédérations et/ou développement d’outils d’interface.

Une tâche parallèle est en cours dans le même projet 13jb visant à comparer plusieurs outils de forces générées par ordinateur (CGF) en environnements synthétiques qui, avec les recommandations ci-dessus, aidera à définir la future évolution de l'environnement synthétique.

Abdellaoui, N., Hubbard, P. and Schoenborn, O. 2005. *Evolution of the Synthetic Environment in the FFSE Section*. DRDC Ottawa TM 2005-141. R & D pour la défense Canada – Ottawa.

Table of contents

List of figures

List of tables

This page intentionally left blank.

1. Document Overview

This document is targeted at project team members, managers, and system users/administrators. It assumes some familiarity with concepts inherent to synthetic environments, such as what is a simulation, what is a network, what is HLA, and what are some of the technologies required to support modeling and simulation.

Background information on the project is presented in the Section 2. Such information includes what is the "business" of the organisation, how it relates to other organisations relevant to the current project, as well as what is the motivation for this task.

Section 3 describes the current status synthetic environments, i.e. the current set of simulation components and practices that support the "business". Section 3 therefore establishes the baseline state of the simulation network infrastructure. It describes briefly each simulation component in use at FFSE and tabulates various characteristics, such as the architecture type. It does not discuss exploitability.

In section 4 the document lists essential problems inherent in the current status, i.e. the deficiencies that interfere with its degree of exploitability. Whereas section 3 baselines the system components, Section 4 baselines usage problems.

Section 5 groups these problems into a small number of categories and proposes some high-level solutions for each category. It also states which problems will *not* be addressed in phase I of this project.

Section 6 gives a conceptual description of each solution that will be addressed in the first phase of this project. The next step for most of these solutions will be specification document that describes more precisely what capabilities and features the solution should provide.

2. Background

DRDC services primarily the needs of the Department of National Defence (DND). It also services, to a lesser but increasing extent, those of our public security partners as well as industry and academia. The Synthetic Environment Technologies group of the Future Forces Synthetic Environment (FFSE) section of DRDC Ottawa (abbreviated simply "FFSE" for the remainder of this document), uses a variety of simulation tools to create experiments and exercises. These experiments and exercises can be used to accept or reject the hypotheses underlying concept development. Some of the simulation tools used are HLA-based, some are not. Some exercises involve a multiplicity of collaborating organizations and simulation components, both hardware and software.

Modeling and Simulation (M&S) can make substantial contributions to the stimulation and dissemination of concepts and to the effectiveness of experimentation, validation and development:

Stimulation

The challenge for M&S is to stimulate the imagination of new concepts by providing concept developers, through interaction with simulations, with virtual experiences of potential future scenarios.

Dissemination

Participants, ranging from junior officers to senior mentors, leave the experiment with a much deeper understanding of the concept than they could get from reading a document or discussing it at a briefing. Often great insights are provided by the players themselves; they take ownership and become the best spokesmen for the concept.

Effectiveness

To be effective, simulations must represent all of the domains referenced by the concepts being developed and support the expression of new concepts, experiments, tactics and scenarios. Simulations must be easy to use, understand, verify and validate. The easier, cheaper and trustful a simulation is to use, the more likely it will be to help effect transformation.

The fundamental challenge is to make M&S efficient enough to extend its utilization to all phases of concept evolution, from development to validation, i.e., make M&S a tool that no concept developer can work without.

FFSE is always in need of better tools to continually improve its service to DND and wants to attract as many of DND clients as possible. The goal is to make the FFSE services more accessible, usable, less time intensive and more comprehensive. The goal of this project is to develop a toolset that will allow users to start the various simulation systems available at FFSE and facilitate their interoperation, i.e. to improve the exploitability of the "Simulation Network". It aims to achieve significant deficiencies reduction in a new design while simultaneously improving readiness, increasing system flexibility, reducing latency time and meeting customers' needs.

Of central concern is the role of the human. In particular, issues abound regarding how to:

1. Reduce the heavy need for personnel during system build,

2. Apply automation as a means to off-load developers,

3. Support developers with advanced technology, and

4. Provide real-time monitoring and trouble shooting of runtime issues.

3. Current System

The FFSE project portfolio currently comprises several simulation tools, exercises and experiments . The different simulations accessible from the Synthetic Environment lab are independent of each other; each simulation, experiment or exercise can be a complete standalone event, having its own setup, integration, execution, and teardown. The sampling of usage at FFSE comprises 2 tools, 6 experiments and 3 exercises that are considered to form the “current system”. The first subsection shows and discusses the data acquired for each, and the remaining subsections describe the individual simulation instances that form the “current system”.

3.1 Characteristics of Simulation Instances

Table 1 is a summary of the architectural characteristics of each system, whereas Table 2 tabulates data relating to “size” characteristics of each system.

Table 1. *Sampling of usage of synthetic environments*

	SYSTEM	TYPE	COMMS STANDARD	STRIVE REQD	MULTI FEDN	DONE
1.	UAV RTB	Tool	HLA	Y	n	Y
2.	MALO phase I	Tool	TCP/IP	n	n	n
3.	UAV CSE	Exp	HLA	Y	n	n
4.	APSD	Exp	DIS	n	Y	Y
5.	MARSIE	Exp	HLA	Y	n	n
6.	MATLAB@Ott (Aerosonde Control)	Exp	HLA	Y	Mak + CAE	Y
7.	RAVEN	Exp	TCP/IP	Y	n	n
8.	MUX	Exp	HLA	Y	n	Y
5.9.	SE ALIX	Ex	HLA	Y	n	Y
6.10.	JSMARTS	Ex	HLA	Y	NTS + CAE	Y
11.	JSMARTS2	Ex	HLA	Y	NTS + CAE	n

Exp=experiment, Ex=exercise

In Table 1, each instance is categorized as a tool, experiment (Exp) or exercise (Ex). In practice, there seems to be little real distinction between an experiment and an exercise, but an exercise tends to be a larger scale experiment that must include operators. The communication architecture of each system is either TCP/IP, in which case the simulation software subunits exchange data via straight TCP/IP communication, or HLA, in which case the subunits communicate via an HLA-compliant RTI, or DIS, in which case the DIS infrastructure is used.

The experiments and exercises (abbreviated EXs for the remainder of this document) make use of one or more of the listed tools, and in most cases of many other tools (subunits), such as STK, MetaVR, etc. Though most EXs require STRIVE, it is clear that they have very little in common otherwise: some use DIS, some use HLA, some straight TCP/IP (e.g. in #0, a UAV GCS connects directly to the CAE RTI via TCP). Some involve only one federation; some involve two federations, connected through a gateway mechanism. For the latter, the federations are heterogeneous (different RTIs) but all involve STRIVE as one of the federations.

Table 1 also shows that some simulation components are still under development; the JSMARTS2 is in its definition stage, whereas for RAVEN and MALO phase I, the ETC is early Fall 2005. Note that system 4 of Table 1 (APSD) provides an HLA gateway to connect it to an RTI as a federate. However, given that it uses JSAF in the background, the federate is really a proxy for the JSAF-based federation. The system has not been tested in federate mode.

In Table 2, each simulation network experiment and exercises (all items of Table 1 except for the UAV RTB and MALO Phase I which are considered tools) was categorized in terms of size in a set of dimensions. These dimensions are as follows:

1. *Creation cost*: The cost to create the system, up to the time when it is first demonstrated.

2. *Run cost*: The cost to re-run the EX, after the demo.

3. *Setup time*: The time it would take the setup the EX, after the first demonstration.

4. *Duration*: How long the EX lasts. One experiment may involve many runs.

5. *Num entities*: How many simulation components are involved; this includes federates as well as individual components that provide a particular and essential “aspect” of the system, such as a GCS, an emulator for a piece of hardware, a visualization application, a voice communications app, etc.

6. *Num CPUs*: How many “standard” CPUs are involved in one experiment, a CPU being defined as being roughly 2 GHz, 2GB RAM.

7. *Num operators*: How many people take part in the experiment, either as operator, user, student, tester, etc.

8. *Stability*: How often the EX, or any of its components, had to be restarted

9. *Vis fid*: The visual fidelity, not to be confused with visual quality. The visual quality of all systems is high given modern PC systems. However, how close the visuals are to the real system is a measure of fidelity. E.g. if a real sensor display is low resolution with much noise, then a high-resolution crisp virtual sensor display in simulation is considered low fidelity, since it will wrongly predict the performance of the operators in the experiments.

10. *Phys fid*: The realism of the motion and state changes in the simulation

11. *Complexity*: The SAF complexity or behaviour that can be described by the experiment framework. The number of parameters that can be set by the users or experimenters was used as a measure of complexity, as well as how many doctrine concepts can be encoded in the scenarios.

12. *Net bw*: The network bandwidth. This tends to be high for systems that involve video streaming, but low otherwise. Note that video streaming is taken into account only if it is essential to the experiment.

13. *Agility*: The degree of adaptability and interoperability of the system with other simulation components not used in the EX, or how easily could the system accommodate new federates etc.

Table 2 confirms that large numbers of entities are typically associated with large number of CPUs, however small numbers of entities can also use substantial compute power, if the entities are themselves complex and compute-intensive. One correlation that seems to emerge is that setup time for an experiment tends to require twice as much time as the duration of the experiment.

Further analysis of the data may reveal more, as seen from figures 1 to 3, which plot the data from Table 2 in three separate charts. The first figure aggregates “resource” dimensions, i.e. those dimensions that relate to how many resources are required to run an experiment. The second figure aggregates “simulation size” dimensions, and the third figure, “modelling size” dimensions. The column of data was scaled to fit between 0 and 15. Columns that use HML (High Medium Low) used the scale 1 (low) to 5 (high), with cells that have two values using the average (so L-M would be converted to 2, and M-H to 4). The values were then multiplied by 2 to give values between 2 and 10. Cells for which no data was available were left empty. Figure 1 shows that the resource usage dimensions were well chosen as they each follow the same tendency. This is also true for the simulation size dimensions. This is less obvious for the modelling dimensions; however a closer inspection reveals that two humps are roughly followed by all simulation instances.

Note that this data was not obtained from a controlled experiment, therefore establishing correlations between the various dimensions is approximate at best, and should be used more as a snapshot of the current “system of simulation components at FFSE”.

Table 2. *Size data for available Simulation Network Experiments and Exercises*

SYSTEM	CREATION COST	RUN COST	SETUP TIME	DURATION	NUM ENTITIES	NUM CPU	NUM OPERATORS	STABILITY	VIS FID	PHYS FID	COMPLEXITY	NET BW	AGILITY
UAV CSE	100k + 200k	18 ph	2 hrs	1.5 hrs	24	10	5+1	L - M	L - M	M	M	n/a	n/a
APSD	$100k	9 pw	4 days	1 wk	300	5-6	7-10	H	H	H	M - H	L	M
MARSIE	$150k	$50k	1 wk	3-4 days	100	15	4-5	H	L	L	L - M	L	L
MATLAB@Ott (Aerosonde Control)		MATLAB	10 min	n/a	1	1	1	H	L (2D plot)	H	H	L	n/a (not yet HLA compliant)
RAVEN	$100k	$50k	2 wks	3-4 days	100	10	4-5	n/a	L	H	M - H	M - H	n/a
MUX	$36k + 2 pw	$5k + STRIVE	1 wk	40-50 days	3-4	1	0	H	n/a	n/a	M	n/a	L
SE ALIX	$1M	$200k	1 month	2 wks	20	20	10	M	L	L	L	H	L - M
JSMARTS	35 pw	15 pw	2 wks	1 wk	10	10	10	L	L	L	M	H	M
JSMARTS2	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a

Exp=experiment, Ex=exercise

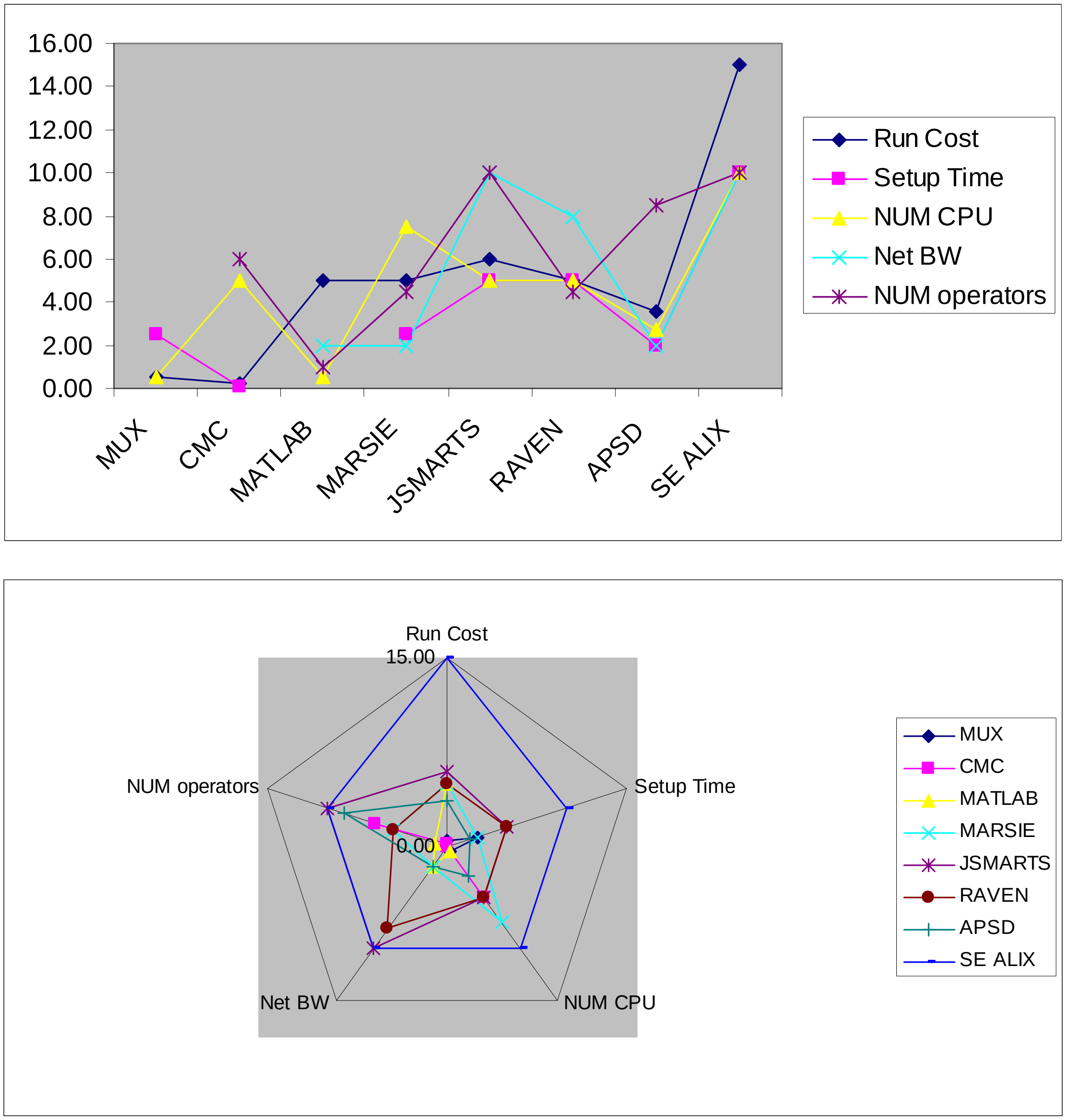
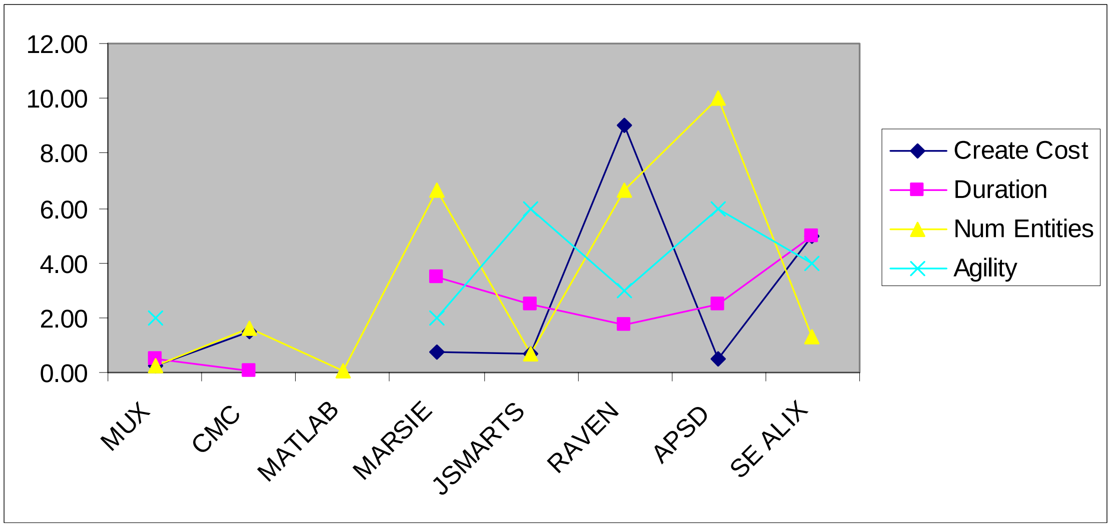


Figure 1. *Spider diagram for resource usage of simulation instances*



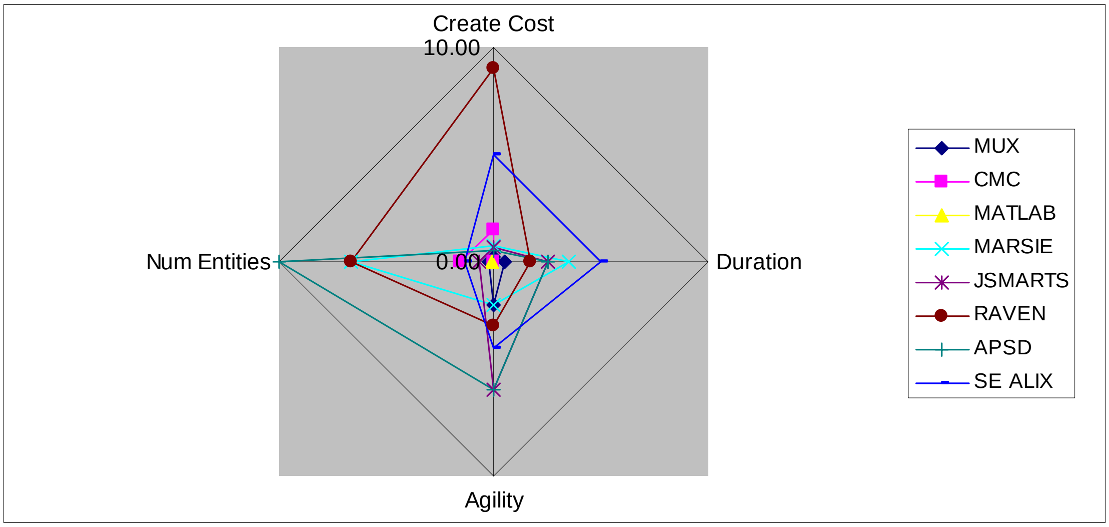


Figure 2. *Spider diagram for simulation size characteristics of simulation instances*

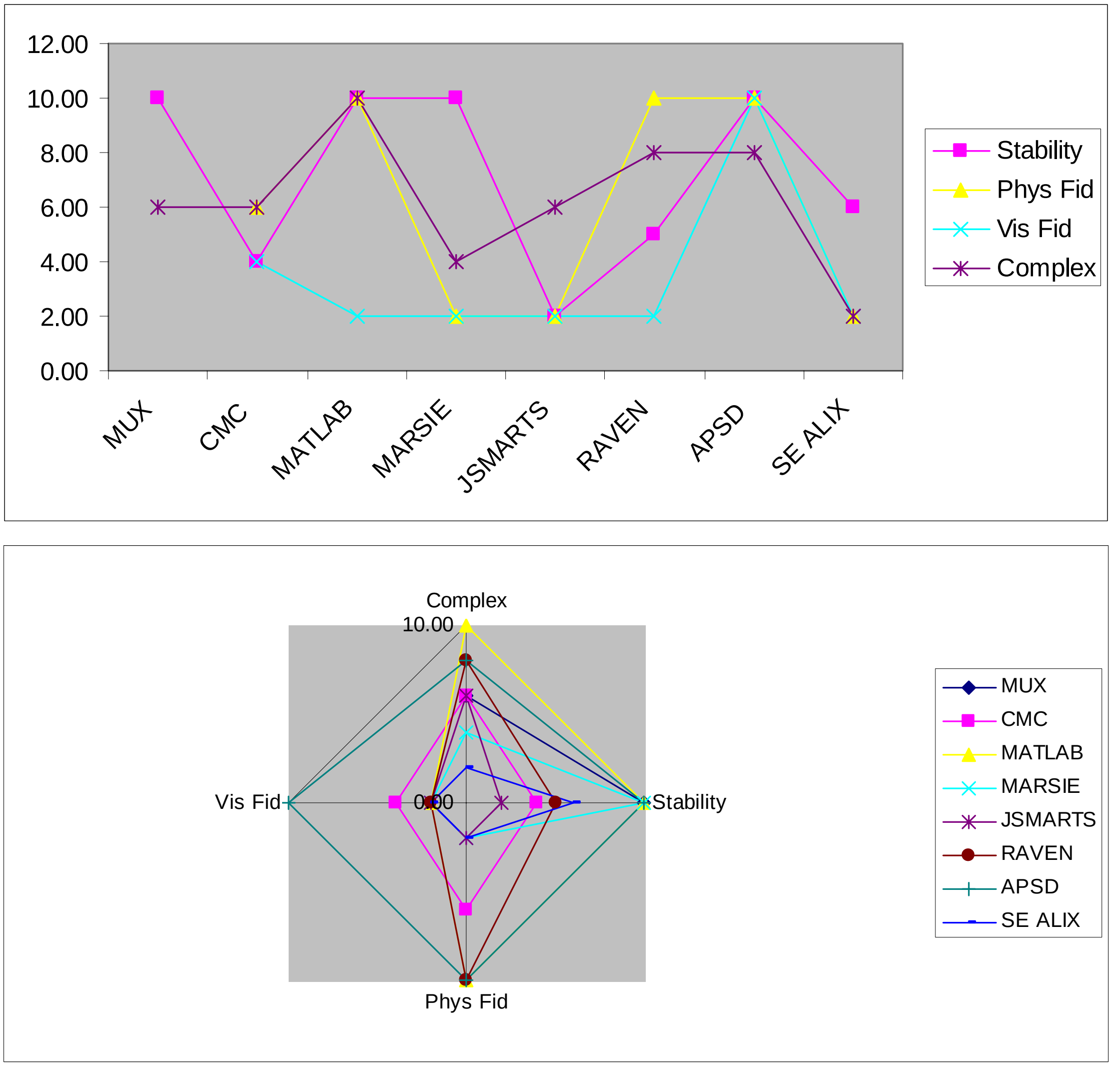


Figure 3. *Spider diagram for modelling size characteristics of simulation instances*

3.2 UAV RTB

The objective of this project was for the contractor, CAE Inc., to provide CFEC and FFSE with an immediate simulation capability for Uninhabited Aerial Vehicles (UAVs) and ground control stations. This simulation capability is referred to as the UAV Research Test Bed (RTB).

The main components of the UAV RTB are a ground control station (GCS), a Communications system and a synthetic environment (SE). The GCS is CDL System’s

Vehicle Control Station (VCS), adapted to work within synthetic environments. It is a dual operator (2 PCs) control station, one for air vehicle operator (AVO) and one for mission payload operator (MPO). It is interfaced to an SE via serial port or UDP. It provides various displays and application components, such as mission planner, moving map display, video streaming, overlay editor, etc.

The SE, developed by CAE, includes the UAV air vehicle simulation and payload simulation (sensors, etc.) as well as a tactical environment within which the simulated UAV operates. A Tactical Scenario Control Station (TSCS) provides the capability to create, modify, and save SE scenarios in a Repository (database), by way of a GUI and two high-resolution LCD panels.

The Communications system allows for voice communication between operators and within the SE.

The STRIVE™ framework is used to provide interoperability between real-time simulation components. Two PCs are used for image generation, one for sensor simulation, one for the TSCS, two for the GCS, and one for the communications system.

3.3 MALO

The Maritime Air Littoral Operations (MALO) Technology Demonstration Program (TDP) is a project undertaken by the Future Forces Synthetic Environment (FFSE) at Defence Research and Development Canada (DRDC) Ottawa in support of Defence Science and Technology Agency DSTA's Client Group 3 R&D Program. MALO TDP is to develop a Modeling and Simulation (M&S) based experimental environment to support the evaluation of Maritime Air operational tactics.

The current configuration for Phase I of the project requires only one high-end PC (extended RAM, video capability and dual processor high-end CPU). The software includes AGI's Satellite Tool Kit and some FFSE code. One operator is required.

In future phases, a 3-operator workstation is envisioned, from which simulators or federates at other labs in different cities can be remotely configured and simulated in parallel. Therefore, higher fidelity, complexity and extensibility are envisioned in the future.

3.4 UAV CSE

This crew-selection experiment (CSE) measures the correlation between current Canadian Forces job components (e.g., tasks and knowledge) and the performance of job incumbents in three emerging jobs (e.g., Vehicle Operator, Payload Operator, and Mission Commander) through a synthetic environment. Each experimental run involves five operators (e.g., two White Cells, one Strive Operator, and two Performance Assessors) plus one Canadian Forces participant, collaborating to operate a UAV RTB in one of nine randomly selected scenarios. All nine scenarios vary by only a single parameter: the location of a terrorist vessel. Each participant assumes the role of Vehicle Operator, Payload Operator, and finally as Mission Commander during a total of three experimental runs. Two experimental staff members, acting as White Cells hold the variation in performance in the two other positions while the participant performs in each of the three emerging positions during the scenario. This work sample test allows the participant to command and control up to six UAVs and their payloads to investigate and identify a terrorist vessel among 15 fishing vessels. A CP140 (Aurora) and two CF18s (Hornets) support the interception and destruction of the terrorist vessel as per requests from the Mission Commander.

A substantial amount of project effort (75%) was spent in the human factors engineering of the system, contracted out to UAV CSE; 10% was spent on equipment, and the remainder on software (about $70k worth). The system consists of a scenario run by STRIVE, a PC acting as a mission manager and logger, four consoles, and a GCS. The PC is the central node, and all entities (20 or so) run on the UAV RTB rack. The system therefore requires 10 PCs and 6 humans to run, a couple of hours to setup, and 30 minutes for each scenario.

Figure 4 shows the system architecture for UAV CSE. In this figure, VO=Vehicle Operator, PO=Payload Operator, STP=Shared Tactical Plot, MC=Mission Commander, MM=Mission Manager, CO=Communications Officer. VO, PO, CO and MC refer to the people at those stations.

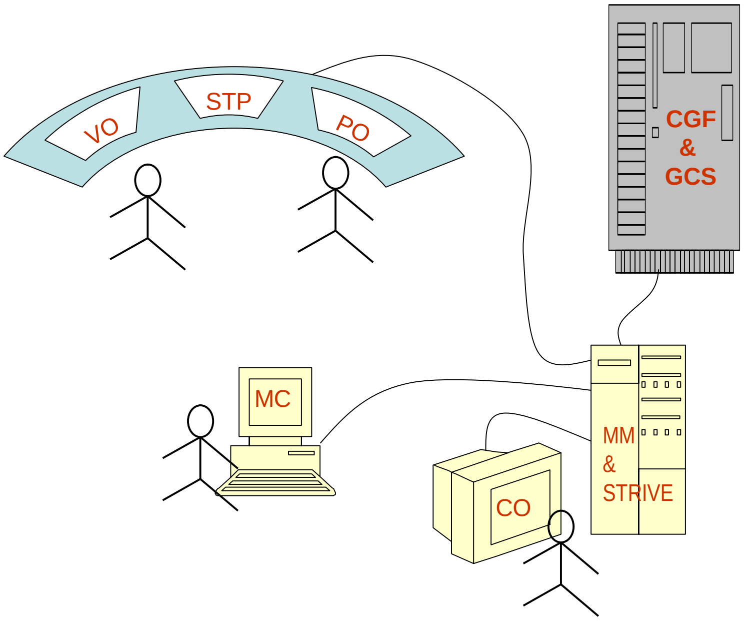


Figure 4. *System architecture diagram for UAV CSE*

3.5 APSD

Airfield Perimeter Surveillance and Defence (APSD) is an integrated capability that provides threat management ability to CF planners, trainers, and security managers. This experiment developed a Uninhabited Aerial System (UAS) synthetic environment simulator to allow a commander to investigate the ability of a mini-UAS to enhance his or her capability to provide this threat management. The UASs that can be simulated include rotary and fixed wing platforms that weigh less than 25 lbs and can fly for 1 to 4 hours with typical EO/IR sensors.

The Linux-based Joint Semi-Autonomous Forces (JSAF) application generated the simulation's entity level platforms such as infantrymen, tanks, sensors, etc. These interacted and were task organized into units for a given mission and could be controlled in group or individually. A Windows-based MetaVR application was used to build the virtual world and construct EO and IR airborne sensor images. The sensor models were validated with real UAV data. They included atmospheric effects, aircraft motion, sensor noise, resolution, contrast and video transmission effects to provide realistic images for operator interpretation.

3.6 MARSIE

This experiment consisted of simulating the Maritime Incursion Scenario – Canada Portion in a synthetic environment, as a precursor to the live Maritime Sensor Integration Experiment (MarSIE) trial. The purpose was to evaluate sensors and recognized maritime pictures (RMP), develop a data analysis plan for MarSIE, and test various aspects of the MarSIE concept before implementation.

The system involved 9 PCs running a large part of the UAV RTB, The Common Operating Environment (COE), and the CAE RTI, with 4-5 human operators. The large number of entities (100) was achieved by slowing down the update rate for the slow-moving ships, but not for the fast-moving UAVs.

3.7 MATLAB@Ott

This experiment was developed at FFSE in spring 2003 to show how MatLab could be used as a simulation component within an HLA-based federation. It consists of a MatLab-based simulation of one UAV with extremely high-fidelity physics, a state machine describing the system state, a PID controller, and a sequence script describing the scenario. The visual display is very basic, a 2D plot of the UAV trajectory on the surface of the Earth. The system description has many hundreds of parameters and several levels of detail. The system has been designed to run as a federate in an HLA federation. However, this capability will be implemented only in the fall 2005, and should allow it to connect to a STRIVE-based simulation.

The simulation runs on one PC and takes approximately 30 seconds to execute.

3.8 RAVEN

The objective of the Remote Aerial Vehicles for ENvironmental monitoring (RAVEN) project is to develop a Beyond-line-of-sight Mission Management System (BMMS) capable of operating a UAV for civilian applications. The project-specific target applications are maritime surveillance and environmental monitoring in a harsh environment.

The RAVEN project is lead by the Memorial University of Newfoundland. FFSE's contribution to the project is a distributed synthetic environment for development of autonomous algorithms for maritime UAVs. As such, the UAV RTB is being distributed, with ground control station and autonomous algorithms at Memorial University in St. John's, Newfoundland, and the synthetic environment at FFSE in Ottawa.

3.9 MUX

The Multi-UAV Experiment (MUX) consisted of using STRIVE and the CAE RTI to run a multi-UAV scenario 1500 times, varying certain parameters between runs (specifically, the number of UAVs, which of two search patterns to use, and the incidence of smuggler vessels), to obtain statistical data on incidence (number of targets detected and identified, time between detections, etc), for analysis by tactics-development personnel.

The experiment cost $36k and 2 weeks of FFSE labour. The experiment requires 40-50 days to run but involves 1500 operator-free scenarios, all different from one another. An operator is required only to set up the scenario parameters that will be simulated, and the system will run all possible combinations of parameters. In contrast to most other STRIVE-based simulations, only one PC was used.

3.10 SE ALIX

This exercise involved seven Canadian Forces crews operating a UAV RTB for approximately two weeks in June 2004, as a synthetic environment experiment complementing the Canadian Forces Experimentation Centre's (CFEC) Atlantic Littoral Intelligence Surveillance Reconnaissance Experiment (ALIX) project. Abbreviated versions of the three UAV missions planned for ALIX were rehearsed in SE ALIX.

SE ALIX involved 20 computers at any given time, in five rooms distributed over the DRDC site and connected via JSimNet. Eight of those CPUs were used for data distribution to the dozen visualization components used. Some scenarios made use of a slightly different set of machines, such that about 30 computers were used overall. Over 15 software applications were required, such as CAE STRIVE, Medallion-S and RAP, CDL VLC and GCS, C2PC, web browsers, COE, etc, operated by approximately 10 people. The SAF complexity was low since human operators were used instead of rule-based scenarios.

3.11 JSMARTS

Joint Simulation, Modeling for Material Acquisition, Requirements, Training and Support (JSMARTS) was a 1 week exercise involving the simulation of one UAV and a CH146 Griffon Helicopter connecting DRDC's RTB lab and Carleton University's Network Tactical Training System in a synthetic environment with CGF provided by CAE Inc.. Technical support was also provided by CAE, Inc.. The purpose was primarily to investigate how rapidly a joint simulation involving Government, Industry and Academia could be setup, using an HLA framework. The secondary purpose was human performance measurement in the scenarios simulated.

The system consists of a UAV RTB running at FFSE, Carleton U.'s CH146 Helicopter simulator, and four other PCs, for a total of over 14 PCs. It required about 10 people to operate and coordinate, and it involved two dozen simulation components (5 federates, 2 RTIs, STRIVE, CDL, Genesa, HLA radio, logging, STAGE, Stealth, Vega, Helisim, etc). The visual fidelity of the system was evaluated as low due to the much-better-than-life quality of the visuals. The stability and physics fidelity were low due to STRIVE. One strength of STRIVE was its ability to represent complex doctrines. The agility of the system was low to medium, given that standard HLA was used (medium), that the CAE FOM was required (low), and that Genesa (FOM adapter for STRIVE to other RTIs) helped bridge with other FOMs.

3.12 JSMARTS2

The phase 2 of JSMARTS is in progress and will involve a more complex exercise, more entities, more participants from Industry and DND and academia. Further details will be available in the fall 2005.

4. Problems with Current System

This section will outline a variety of problems inherent in the current "system", by which we mean the current set of usage outlined in Table 1.

All tools and EXs lack documentation in a form that is appropriate for system use, e.g. the UAV RTB tool has some online documentation, but lacks user manuals, for instance. What documentation does exist is not easily accessible, and its existence is sometimes not widely known. ALIX, JSMARTS and APSD each have technical reports, but all lack some essential elements for the purpose of system re-use.

All EXs are essentially one-time efforts: re-using them is not possible or extremely difficult for various reasons, e.g., for JSMARTS, the barrier is the level of effort required to setup the network access to maintain the security requirements of FFSE and DND. For others, there is not a single package/distribution; rather they are a collection of simulation components that were accessible during a certain window of time and are now inaccessible or have become obsolete and must be replaced. Many of these were used as a proof of concept and are improved or replaced at the next round.

The following is a non prioritized, non-comprehensive list of observed problems or issues:

1. Setup of the EXs that require STRIVE (i.e. #6 to 11 of Table 1), network occupies about 50% of the setup time when external federations are involved: network administration is centralized due to security requirements. For instance, external machines that participate must be properly firewalled and proxied, their IP address and required ports must be added to the network, all done centrally.

2. Of all the systems, only the UAV RTB is documented as to setup and start-up procedures. All others involve tacit knowledge, in many cases acquired over a year ago.

3. There is no central access point to the simulation network resources to create federations. All systems, whether they are tools or EXs, use different combinations of architectural network elements: HLA, TCP, DIS. These are not interoperable without human intervention. If an EX is DIS-based, and it should be used with an HLA federation, a gateway must be created manually. If two components were created with different FOMs, the two have to be mapped manually.

4. Many of the EXs that use CAE Inc.'s STRIVE^{TM} (i.e. #6 to 11 of Table 1) have run into problems with implementation and delays. Documentation on FFSE experiences with STRIVE can be found in a technical note [6]. Additionally, the following comments specific to STRIVE are noted:

   - Publishing extra data from an entity requires modifying some header files and recompiling the associated DLLs. This is a common occurrence, with each exercise requiring modifications to different STRIVE DLLs. This has lead to multiple versions of STRIVE existing in FFSE, on separate hard drives. Loading a component from a scenario created in one exercise into a scenario created with a different version of STRIVE typically leads to crashes and other undesirable behaviour. This makes re-use difficult and laborious and certainly not user-friendly. This also requires that each “system” be on a separate hard drive, requiring hard drive swapping for each EX but also impeding a centralized access point to all EXs.

- FOMs are incompatible: FOM bridges must be created for each EX.

- CAE-RTI ambassador is hard-coded, so no extensions to it are possible (not a “factory” design, fat base interface doesn’t scale) and it can not be adapted to a new FOM, etc.

- A STRIVE federation can not communicate with another federation without extensive customization/patching.

- It is difficult to compose some entities. For example, the EO entity that was easily added to a UAV cannot be added to a CP140 without coding and rebuilding the software.

- The licenses expire without notice, and the system does not notify the user if it cannot start due to an expired license. Each license is valid for three months, making this problem frequent.

- Current simulations using STRIVE are often restricted to simulating up to a hundred entities. This is insufficient to represent military support forces, where the order of magnitude is 10000 entities. Current experience has shown federations of more than 30-40 entities are difficult to simulate in STRIVE. Adding more entities causes the simulation to fall behind and can no longer function in real-time.

5. It would be difficult to work with potential experimentation partners such as PSEPC, RCMP, other DRDC offices, industry, and allied nations; as the capability of “remotely” executing the experiments does not currently exist.

6. As a federation grows, the constituent simulations and models impose more constraints on the integration of new simulations, thereby increasing the cost of adding them. Dividing a federation into several “sub-federations” is currently not possible without implementing an ad hoc solution specific to each case.

7. With the recent focus on asymmetric warfare, FFSE increasingly needs to represent the civilian population and traffic in major city areas since it serves as the primary cover for opponents. This requires significant compute power.

8. “Open architecture” for federation interaction is desired so that federations that don’t yet exist may be accommodated.

5. Solutions

For the purpose of identifying solutions to the previous section's long list of problems, it is useful to categorize the issues. Thus, the problems are classified into four groups:

1. Network setup: problems that arise from requirements to setup network access or interconnectivity of simulation components,
2. Accessibility: problems that arise from the lack of accessibility of the simulation components, which can be due to insufficient accessibility to the documentation, or the technology itself,
3. Stability: problems that arise from the lack of stability of the simulation components that rely on STRIVE, i.e. the unpredictable level of operational functionality supported at any given time or in any given simulation
4. Agility: problems that arise from the lack of adaptability of a component of the system to other components, or lack of interoperability with other components.

Each of the following subsections presents and addresses one category of problems with associated solution. The solution is discussed only at a high level, i.e. at the level of "task" rather than specification.

5.1 Network setup

Problems arise as a consequence of having to setup network access for, or interconnectivity between simulation components, across pre-defined security boundaries or physical geography. For instance, in the JSMARTS project, the University of Carleton simulation components required computers to be setup with proper security level, giving them access to the DRDC simulation network (JSimNet), and opening the appropriate ports in the JSimNet firewall to allow network traffic between the simulation components of the JSMARRTS system.

For any simulation that involves simulation components living outside the security perimeter of the DRDC network, this is a step that can require administrative effort in order to set up links, configure firewalls and satisfy security requirements.

However, this hurdle is mostly administrative and due primarily to the fact that linking simulation components across network boundaries is a relatively recent activity. The process for connecting the FFSE network to external sites has been documented [9], and this process is constantly being improved as requirements evolve. It is recommended that instances of the network setup be documented for re-use in other circumstances.

5.2 Accessibility

There are two main causes for accessibility problems, namely documentation, and centralization:

1. Documentation: Some of the tools and EXs have *some* documentation, but this documentation is either difficult to access (e.g. there is no way of finding out what documentation is available), incomplete for purposes of system exploitation, or in need of validation or update.

   A solution to this aspect of the accessibility problem would entail:

   a. Documenting each simulation component listed in Table 1
   b. Creating a centralized repository for storage
   c. Defining an access method that supports decentralized access (e.g. via web browser)
   d. Defining accessibility constraints (who, when, etc)

2. Centralization: there is no centralized "portal" from which the different simulation tools and EXs can be configured and used. Such a portal is in fact unlikely to be possible, since few of the tools consist of one "program" which can be "pointed and clicked" for configuration and use, and many EXs involve starting up components in separate geographic locations.

   A solution could however address some aspects of setup and use by bringing together in one location the set of available simulation tools and EXs, their documentation, a technical point of contact and links to as many of their components as possible.

In order to continue to explore this problem, it is recommended that for each resource, a list be compiled of the three most critical issues causing it to be inaccessible. From these, a one GUI front-end could be designed, e.g. a front-end from which all documentation can be browsed and searched, with a procedure to follow and list of entities to contact for network setup, and a procedure to follow with links to the components that are available via mouse click. Other capabilities, which however would require a significant amount of effort, are a "FOM adaptor", and an "Automatic Federation Configuration". Note that currently, there is no a "library of federates", making the specification of such capability difficult.

5.3 Stability

Based on the observed problems listed in Section 4, it appears that the root cause of current stability problems in most of the usage is related the SAF tool that is currently the most used within the section, namely CAE Inc.'s STRIVE software. STRIVE was used to quickly stand up a simulation capability for initial works for FFSE clients and it is recognized that other tools, if used to this extent in the section, may also lead to stability problems. As a result, it is of interest to investigate the causes of the stability in FFSE's usage of STRIVE which are due to two main factors:

1. Old version of STRIVE are in use at FFSE: The CGF in use at FFSE is Version 1.8. Substantial changes, bug fixes and improvements have taken place from 1.8 to 2.0.
2. Technical Support: some techniques mentioned by STRIVE system experts at CAE should have been employed earlier.

A solution to this problem would involve one or more of the following:

1. Upgrading the STRIVE system. This would require upgrading most EXs developed at FFSE to the new version and may not be practical or even possible. Most likely, copies of the old STRIVE should be kept alive in case one of the EXs is required in its original form. New EXs would directly make use of the new STRIVE. Some entities used in the old version, and modified at FFSE, would likely require re-implementing in the new version as the XML export feature is not available in v1.8. The STRIVE will be upgraded to 2.0 in the coming months, independently of this SNE project's Software Tool Specification/Implementation task. Therefore, this item will not be pursued.
2. Tech support. Making the users of the STRIVE framework aware of the scope of technical support, and making the names of technical support contacts easily accessible, and the generation of bug reports clear and easy to access. This was identified as a knowledge problem, so it should be dealt with as part of the documentation.
3. Replacing STRIVE with a different CGF/GSM framework. This is being explored in a parallel effort in 13jb so it will not be pursued in this project.
4. Customize STRIVE. Custom code could be added to FFSE's new STRIVE to give better error messaging when licenses expire, or when other errors occur. It is unlikely that CAE will be able to do this, so this item will not be pursued.
5. Determining rigorously what lead to the instabilities seen in the past and creating a process that will avoid them, with enforcement or control mechanism (who is allowed to make modifications to DLLs, when, how, what kind of testing required, etc).
6. Determining if there is a way of integrating the virtual EO functionality to STRIVE in such a way that the EO can be attached to any entity.

In the future, it is essential that FFSE have access to multiple CGF and SAF tools, terrain services, image generators, etc., and not be dependent on a single tool. An open architecture, where component systems can be decomposed and unstable components replaced, is the longer term solution.

5.4 Agility

The agility problems relate to the interoperability of simulation tools and to the adaptation of a simulation component to work in a different simulation framework or scalability in a large simulation.

The scalability of a SAF tool to larger simulations could take one or both of the two different routes:

1. Buy more compute power: the STRIVE system requires significant computational power. A current project [2] in the US will require 26 computers to simulate 1000 entities with STRIVE. This option can only be addressed on a project basis, as required, so it will not be pursued.
2. Better simulation techniques:

    a. Use of banding: vessels that are slow can be updated at a lower frequency, thus decreasing the required compute power. This can be added to the documentation.
    b. Turning off entities when no longer in use. This can also be added to the documentation.
    c. Source-level interest management (SLIM) can be used to improve the number of federates that can interact, by filtering what data is generated *by a given federate*. This decreases the amount of data that must pass through the RTI, thus improving scalability (see reference 3 in the Bibliography). It is unlikely that this technique could applied to federates used within a STRIVE simulation without having to modify STRIVE. The resources for such a task are not currently available so this solution will not be pursued.

The electro-optical sensor visualization problem could involve any of the following:

1. Find or create (or contract out the work to create) an open-source CGF framework
2. Work with CAE to integrate their visualization capability into STRIVE 2.0. This has the disadvantage of maintaining the dependency of FFSE on STRIVE.

The interoperability of STRIVE with other simulation frameworks would require FOM adaptation and/or inter-federation gateways:

1. STRIVE uses a unique CAE FOM, within which all federates must be able to publish and subscribe attributes. Most other simulation frameworks use different FOMs, e.g. the RPR FOM 1.0 or 2.0. A solution would be to create a user interface to map FOMs, or to create an expert system to automatically map FOMs, or purchase either or both. The authors do not know of any expert system to solve this problem at the time of writing, though a good mapping tool is available in Virtual Technologies Corporation's Interdaptor™ or CAE's Genesa (other FOM to CAE FOM), as well VR-Exchange™ from MAK Technologies.
2. A "federation gateway" encompasses a variety of techniques to link two federations, i.e. two simulation frameworks. Ideally, the gateway does not require any changes to the RTI software or, worse, to the HLA standard. More often, precluding such changes greatly decreases the number of simulation "tasks" that can be carried out within and between the different federations. The solutions to this problem are:
    a. Replace or complement STRIVE with a tool that provides such capabilities, such as CAE's Genesa or Virtual Technologies Corporation's Interdaptor™;
    b. Simply document a set of specifications that such a system should meet, for future purchases/plans; this is currently being done in FFSE Task 149;
    c. There has been some talk that moving to JSAF (away from STRIVE) would be desirable in the long term due to the popularity of that HLA-compliant framework in the US DOD. The answer to this will likely emerge from FFSE Task 149.

6. Selected Solutions for Project 13jb

In section 5 lists several solutions for each category of problem. In this section, this list is further refined and recommendations are made for action within the 13jb project.

6.1 Documentation

A solution to the problem of lack of documentation would consist of:

1. Documenting each simulation component listed in Table 1:
    a. For tools, desirable types of documentation are tutorials and examples, user documentation, system documentation, and list of one or more local "expert user" available for technical support. For STRIVE in particular, documenting how to obtain technical support (5.3.2) and how to improve simulation performance (5.4.2 a and b)
    b. For EXs, the documentation should have links to the final project report which should include the results, system architecture, communications protocols, simulation components used (with their version numbers, cost, and provider), the system requirements, and the setup and start-up processes
    c. Versioning each document such that different versions can be sequenced and the data of release easily inferred from the version number
    d. Develop a library of federates currently available within the section with associated FOMs, RTI compatibility and component tools.
    e. Instances of the network setup for re-use in other circumstances.
2. Creating a centralized repository for storage:
    a. Defining an access method that supports decentralized access (e.g. via web browser)
    b. Defining accessibility constraints (who, when, etc)
    c. Define a storage location that supports such access
3. Assign a person to be responsible for:
    a. enforcing accessibility constraints
    b. adding upgrading and removing documentation
    c. creating regular backups
    d. communicating the access method and contents of the repository to potential users (via a billboard posting, a yearly seminar, a quarterly email, etc)

6.2 Stability and Agility

The solutions actionable within project 13jb are all particular to STRIVE, which is in the process of being upgraded to v2.0. Therefore they are addressed in one section, and should all relate to v2.0 of the STRIVE rather than the current outdated v1.8:

1. Determine rigorously the causes of the STRIVE instabilities seen in the past and create a process that will avoid them:

a. Apply the bug report process already in place at FFSE for the most unstable exercises, on the current v1.8 system if time is pressing, or on v2.0 if possible.
b. If using v1.8, determine if this is a v1.8 bug or a problem with how the scenario was constructed. Verify that the problem is absent in v2.0, and if not, document it and make available via GUI portal.
c. If instability is caused by wrongly modifying some of the DLLs, define enforcement or control mechanism: who is allowed to make modifications to DLLs, when, how, what kind of testing required, etc.

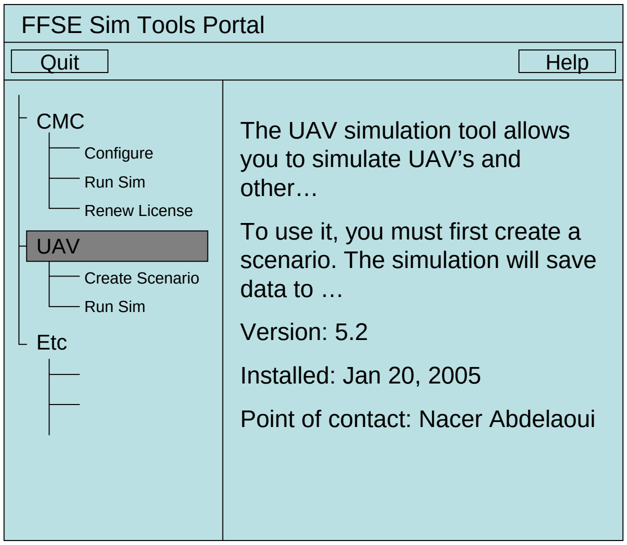
2. Determine if there is a way to integrate the virtual EO functionality, which is not a CAE product, into STRIVE in such a way that the EO can be attached to any entity. This requires interaction with CAE technical support.

3. Determine if Virtual Technologies Corporation's Interdaptor™ software or CAE Genesa is a viable complement to CAE's STRIVE, and evaluate Mak Technologies VR-Exchange software.

6.3 GUI front end

The GUI should display the set of simulation tools available, and a list of work tasks pertinent to each tool. Clicking on a tool should give high-level information about the tool such as Program name, purpose, version number, point of contact (expert), date installed, accessibility requirements, other requirement, etc.

An example of what such a GUI Front end might look like is provided in Figure 5.



The UAV simulation tool allows you to simulate UAV's and other…

To use it, you must first create a scenario. The simulation will save data to …

Version: 5.2

Installed: Jan 20, 2005

Point of contact: Nacer Abdelaoui

Figure 5. *Conceptual representation of GUI portal*

The tools and tasks displayed should be configured at run-time by reading a simple ASCII text file, e.g. an INI file or an XML file. An example content might be:

```
[UAV CSE]
config = “Configure” c:\...\uav_cse\config.exe
run = “Run Sim” c:\...\ uav_cse\run.exe
relic = “Renew License” c:\...\ uav_cse\emailLicReq.py

[UAV]
setup = “Create Scenario” …
run = …
```

where each section defines a tool that will be displayed in the GUI, and each attribute of the section defines a list of possible tasks specific to the tool for that section. Each task is a string that appears in the GUI, and a command to run. The command to run could be an executable, a Python script, etc. E.g. the `emailLicReq.py` script might open a dialog to get extra info from you and get your confirmation to proceed with sending the license renewal request. Clicking on a task should trigger the associated program or script.

Bibliography

1. J. Dingel et al., Bridging the HLA: Problems and Solutions, Sixth IEEE International Workshop on Distributed Simulation and Real Time Applications, Oct 2002.

2. M. Brunet, personal communication with CAE Inc. employee on use of STRIVE in the Aviation Training and Rehearsal Systems (ASTARS) program for the US Special Forces, July 2005.

3. Katherine Morse, An Adaptive, Distributed Algorithm for Interest Management, Ph.D. Dissertation, University California at Irvine, http://gamepipe.isi.edu/~zyda/Theses/Katherine.Morse.pdf, July 2005.

4. P. Hubbard, D. Persram and G. Fusina. (2004) ALIX Mission Rehearsal in a Synthetic Environment: Summary and Preliminary Analyses. (DRDC Ottawa TM 2004-200). Defence R&D Canada - Ottawa

5. B. Kim. (2005) JSMARTS Initiative: Advanced Distributed Simulation across the Government of Canada, Academia and Industry – Technical Description (DRDC Ottawa TM 2005-101). Defence R&D Canada - Ottawa.

6. P. Pace and B. Kim, Modelling and Simulation for the Use of a Mini-UAV for Airfield Perimeter Surveillance and Defence – Simulator Development, DRDC Ottawa Technical Memorandum in production, September, 2005.

7. FFSE Section UAV RTB Portal: Documentation stored on the STRIVE computer and accessible via web browser, June 2005.

8. A.L. Vallerand, B. Johnson, and P. Hubbard. (2005) FFSE's Experiences with STRIVE^{TM}. (DRDC Ottawa TN 2005-063). Defence R&D Canada - Ottawa. Internal publication

9. D. Skinner and A.L. Vallerand. (2005) Connections to the JSimNET: Connecting from Non-DRDC Sites. (DRDC Ottawa TN 2005-127). Defence R&D Canada - Ottawa. Internal publication

List of symbols/abbreviations/acronyms/initialisms

APSD	Airfield Perimeter Surveillance and Defence
AVO	Air Vhicle Operator
CFEC	Canadian Forces Experimentation Centre
DIS	Distributed Interactive Simulation
DLL	Dynamically Loaded Library
DND	Department of National Defence
DRDC	Defence Research & Development Canada
DSTA	Defence Science and Technology Agency
ETC	Estimated Time of Completion
EXs	Exercises and Experiments
FFSE	Future Forces Synthetic Environment
FOM	Federation Object Model
GCS	Ground Control Station
GSM	GUI for Simulation Management
GUI	Graphical User Interface
HLA	High-level Architecture
JSAF	Joint Semi-Automated Forces
MPO	Mission Payload Operator
MUN	Memorial University
NTS	Network Tactical Simulator (from Carleton University)

RCMP	Royal Canadian Mounted Police
RTI	Run-time Infrastructure
SE	Synthetic Environment
TCP/IP	Transport Control Protocol/Internet Protocol
TDP	Technology Demonstration Program
TSCS	Tactical Scenario Control Station
UAS	Uninhabited Aerial System
UAV	Unmanned Aerial Vehicle
UDP	Universal Datagram Packet
VCS	Vehicle Control Station

Glossary

TECHNICAL TERM	EXPLANATION OF TERM
Multi-federation	A simulation that involves more than one federation communicating. Each federation may or may not be HLA-based, however at least one federation is typically HLA-based
Interoperability	Ability to make separate federations work together. Other related terms: federation bridges and gateways, federation interoperability, inter-federation communication
Adaptability	Ability of a federation to integrate federates that use different FOMs or federates that are not HLA compliant. Other related terms: FOM agility, extendable FOM
Simulation Tool	A software tool for virtual representation of the behaviour of a system in both space and time.
Experiment	A scripted simulation that uses one or more tools, for the purposes of demonstrating a capability of the tool or combination of tools.
Exercise	A larger scale experiment that is based on smaller experiments having been successful. Typically crosses domain and geographical boundaries.
PID controller	Type of controller that uses position, accumulation and error to determine the forces that should be applied to a physical body (such as a plane rudder or a robot arm), to move the body from its current position and orientation to a new one, possibly within certain constraints (such as the amount of time or maximum torque available).

DOCUMENT CONTROL DATA

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

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

Defence R&D Canada – Ottawa
Ottawa ON K1A 0Z4

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

UNCLASSIFIED

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

Evolution of the Synthetic Environment in the FFSE Section (U)

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

Abdellaoui ,Nacer and Hubbard, Paul and Schoenborn, Oliver

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

November 2005

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

40

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

9

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 the address.)

DRDC Ottawa

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

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

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

DRDC Ottawa TM 2005-141

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

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

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

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

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

As part of a project on the Exploitation of Synthetic Environments (Project #13jb), this document summarizes a data-gathering effort to measure the current synthetic environment usage in the Future Forces Synthetic Environment section. Eleven simulation instances, ranging from a two-week, human-in-the-loop rehearsal to a 30-second stand-alone computation, are described and measured in term such as accessibility, complexity, agility and stability.
The problems that limit the exploitability of the current simulation network are also discussed, and various solutions are proposed. Some of these solutions will be implemented within the 13jb project.

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

FFSE Synthetic Environment ARP RTB MALO ALSP MARSIE RAVEN MUX ALIX JSMARTS Documentation Portal SE

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

DEFENCE R&D DÉFENSE

www.drdc-rddc.gc.ca