



Calhoun: The NPS Institutional Archive
DSpace Repository

Reports and Technical Reports

All Technical Reports Collection

2013-09-30

Virtual Battlespace Scenario Encoding for Reuse, Phase I Report

Blais, Curtis L.; Eady, Michael; David Parkes, David

Monterey, California. Naval Postgraduate School

<http://hdl.handle.net/10945/69740>

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the United States.

Downloaded from NPS Archive: Calhoun



Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

<http://www.nps.edu/library>

NPS-MV-13-001



**NAVAL
POSTGRADUATE
SCHOOL**

MONTEREY, CALIFORNIA

VIRTUAL BATTLESPACE SCENARIO ENCODING FOR REUSE:

PHASE 1 REPORT

by

Curtis Blais, Michael Eady, and David Parkes

30 September 2013

Approved for public release; distribution is unlimited

Prepared for: Commander, Marine Corps Systems Command (DC SIAT)
2200 Lester Road, Quantico, VA 22134

REPORT DOCUMENTATION PAGE			<i>Form Approved</i> OMB No. 0704-0188		
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.					
1. REPORT DATE (DD-MM-YYYY) 30-09-2013		2. REPORT TYPE Technical Report		3. DATES COVERED (From-To) 1-30 Sep 2013	
4. TITLE AND SUBTITLE Virtual Battlespace Scenario Encoding for Reuse, Phase 1			5a. CONTRACT NUMBER		
			5b. GRANT NUMBER		
			5c. PROGRAM ELEMENT NUMBER N6327313WX50042		
			5d. PROJECT NUMBER		
			5e. TASK NUMBER		
6. AUTHOR(S) Curtis L. Blais, Michael Eady, and David Parkes			5f. WORK UNIT NUMBER RVGQ5		
			8. PERFORMING ORGANIZATION REPORT NUMBER NPS-MV-13-001		
			10. SPONSOR/MONITOR'S ACRONYM(S) MCSC		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) AND ADDRESS(ES) NPS MOVES Institute 700 Dyer Road RM265 Monterey, CA 93943-5000			11. SPONSOR/MONITOR'S REPORT NUMBER(S)		
			9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) Commander, Marine Corps Systems Command (DC SIAT) ATTN: Mike O'Neal 2200 Lester Road Quantico, VA 22134		
12. DISTRIBUTION / AVAILABILITY STATEMENT Distribution Statement A. Approved for public release; distribution is unlimited.					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT The United States Army and United States Marine Corps employ the Virtual Battlespace 2 (VBS2) commercial game for small unit training. Each service has significant investment in training scenarios constructed using VBS2 tools and conforming to the vendor's particular data formats. To move toward improved interoperability, to gain greater fiscal flexibility in addressing the statutory intent for open competition and affordability, and to protect the investment made in models, terrain, and other elements of training scenarios that are separate and distinct from the virtual and gaming environments in which the simulation executes, open standards need to be applied in place of proprietary commercial off-the-shelf architectures. The Extensible Markup Language (XML) is a wide-spread approach to describing data format and content to support efficient data processing and data interchange across systems. The purpose of this report is to describe initial application of XML technologies to the representation of VBS2 scenario data, demonstrating feasibility for the capture and exchange of VBS2 scenario data. The report provides a plan of action for a follow-on phase of the effort to expand the representation and for use with other XML-based standards, such as the Military Scenario Definition Language (MSDL), to create opportunities for broader interchange of scenario data across a variety of combat simulations.					
15. SUBJECT TERMS Modeling, simulation, scenario, reuse, interoperability, VBS2, XML					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 99	19a. NAME OF RESPONSIBLE PERSON Curtis Blais
a. REPORT Unclassified	b. ABSTRACT Unclassified	c. THIS PAGE Unclassified			

THIS PAGE INTENTIONALLY LEFT BLANK

**NAVAL POSTGRADUATE SCHOOL
Monterey, California 93943-5000**

Ronald A. Route
President

Douglas A. Hensler
Provost

The report entitled "*Virtual Battlespace Scenario Encoding for Reuse: Phase 1*" was prepared for and funded by the Marine Corps Systems Command, Modeling and Simulation Organization.

Further distribution of all or part of this report is authorized.

This report was prepared by:

Curtis L. Blais
Research Associate, MOVES

Capt Michael Eady, USMC
Student, MOVES

LTC David Parkes, USA
Student, MOVES

Reviewed by:

CDR Joe Sullivan, Director
MOVES Institute

Released by:

Jeffrey D. Paduan
Dean of Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The United States Army and United States Marine Corps employ the Virtual Battlespace 2 (VBS2) commercial game for small unit training. Each service has significant investment in training scenarios constructed using VBS2 tools and conforming to the vendor's particular data formats. To move toward improved interoperability, to gain greater fiscal flexibility in addressing the statutory intent for open competition and affordability, and to protect the investment made in models, terrain, and other elements of training scenarios that are separate and distinct from the virtual and gaming environments in which the simulation executes, open standards need to be applied in place of proprietary commercial off-the-shelf architectures. In the current (and foreseeable) environment of constrained budgets, it is ever more critical that the services protect and enhance their investments in simulation systems used for training and other purposes. Expanding capabilities for open scenario interchange will improve scenario reuse while creating greater opportunities for simulation data interchange and open competition for future gaming capabilities.

The Extensible Markup Language (XML) is a wide-spread approach to describing data format and content to support efficient data processing and data interchange across systems. This report describes initial application of XML technologies to the representation of VBS2 scenario data, demonstrating feasibility for the capture and exchange of VBS2 scenario data. The report provides a plan of action for a follow-on phase of the effort to expand the representation and for use with other XML-based standards, such as the Military Scenario Definition Language (MSDL), to create opportunities for broader interchange of scenario data across a variety of combat simulations.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I. INTRODUCTION.....	1
A. BACKGROUND	1
B. SCOPE	2
C. APPROACH	2
D. ORGANIZATION OF THIS DOCUMENT	3
II. MOTIVATION AND RELATED WORK	5
A. INTRODUCTION.....	5
B. REUSE POLICY AND COST SAVINGS	5
C. REUSE TECHNICAL INITIATIVES.....	6
1. Military Scenario Definition Language	6
2. Joint Training Data Services (JTDS)	8
3. Rapid Data Generation.....	9
4. Rapid Scenario Generation	11
5. Interoperability Frameworks	13
D. SUMMARY	14
III. VBS2 SCENARIO DATA	15
A. INTRODUCTION.....	15
B. VBS2 DATA STRUCTURES	15
C. PRELIMINARY XML FORMAT FOR SELECTED VBS2 DATA STRUCTURES.....	19
D. SUMMARY	20
IV. PRELIMINARY XML SCHEMA.....	21
A. INTRODUCTION.....	21
B. SELECTED VBS2 DATA STRUCTURES AND ASSOCIATED SCHEMA DESIGN	21
C. SUMMARY	22
V. TRANSFORMING XML CONTENT TO VBS2 DATA STRUCTURES	25
A. INTRODUCTION.....	25
B. XSLT DESIGN.....	25
C. EXAMPLE DATA TRANSLATIONS.....	27
D. VERIFICATION OF CORRECT DATA TRANSLATION	28
E. SUMMARY	29
VI. CONCLUSIONS, LESSONS LEARNED, AND NEXT STEPS.....	31
A. CONCLUSIONS	31
B. LESSONS LEARNED.....	31
C. PLAN OF ACTION FOR PHASE 2	32
APPENDIX A. VBS2 OBJECT CLASSES FOR A NOTIONAL SCENARIO..	35
APPENDIX B. VBS2 OBJECT CLASSES	67
APPENDIX C. PRELIMINARY XML SCHEMA DESCRIBING A REPRESENTATION OF VBS2 SCENARIO OBJECT CLASSES	75

**APPENDIX D. XSLT FOR TRANSFORMING AN XML REPRESENTATION
OF VBS2 UNIT OBJECT CLASS TO VBS2 SCENARIO INPUT DATA
FORMAT..... 79
LIST OF REFERENCES..... 81
INITIAL DISTRIBUTION LIST 83**

LIST OF FIGURES

Figure 1.	Top-level structure of the MSDL XML data model. From (Military Scenario Definition Language Product Development Group. SISO-STD-007-2008, 2008).....	7
Figure 2.	Conceptual view of the RDG Common Data Production Environment data discovery capabilities. From (Scrudder et al., 2011).	10
Figure 3.	RSG project scenario generation and execution process. After (Blais, Dodds, Pearman, & Baez, 2009).	12
Figure 4.	Scenario construction in RSG (left) and execution in Pythagoras (right). From (Blais et al., 2009).	12
Figure 5.	Rifleman's first-person view after being created in the VBS2 OME. The 'TESTTANK' model is present as a point of reference for comparison to later figures in this document.....	17
Figure 6.	Basic properties of a rifleman created in VBS2 OME.....	18
Figure 7.	Advanced properties of a rifleman created in VBS2 OME.	18
Figure 8.	XML structure for VBS2 unit class.	23
Figure 9.	XML structure for VBS2 scenario object classes.	24
Figure 10.	Data transformations between VBS2 scenario data and XML-based data models through an XML representation of the VBS2 data.....	26
Figure 11.	Rifleman's first-person view after translation through XML and XSLT, subsequently pasted directly into the .biedi initialization file and previewed in VBS2.	28
Figure 12.	Basic properties of a rifleman loaded through XML and XSLT.	29
Figure 13.	Advanced properties of a rifleman loaded through XML and XSLT.	29

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

Table 1.	VBS2 Empty Vehicle Class	67
Table 2.	VBS2 Group Class	68
Table 3.	VBS2 IED Class	69
Table 4.	VBS2 Intel Class	70
Table 5.	VBS2 Marker Class	71
Table 6.	VBS2 Object Class	71
Table 7.	VBS2 Trigger Class	72
Table 8.	VBS2 Unit Class	73
Table 9.	VBS2 Waypoint Class	74

THIS PAGE INTENTIONALLY LEFT BLANK

ACRONYMS AND ABBREVIATIONS

AI	Artificial Intelligence
BBP	Better Buying Power
BOM	Base Object Model
C2	Command and Control
CCTT	Close Combat Tactical Trainer
CDPE	Common Data Production Environment
COA	Course of Action
DDMS	DoD Discovery Metadata Specification
DIS	Distributed Interactive Simulation
DoD	Department of Defense
DOE	Design of Experiment
DVTE	Deployable Virtual Training Environment
HLA	High Level Architecture
IC-ISM	Intelligence Community Information Security Markup
IED	Improvised Explosive Device
ITE	Integrated Training Environment
IWARS	Infantry Warfare Simulation
JBUS	Joint Simulation Bus
JDS	Joint Data Services
JLVC	Joint Live, Virtual, Constructive
JTDS	Joint Training Data Services
LVC-G	Live, Virtual, Constructive and Gaming
LVC-IA	Live, Virtual, Constructive Integrating Architecture
M&S	Modeling and Simulation
MANA	Map-Aware Non-uniform Automata
MATREX	Modeling Architecture for Technology, Research and Experimentation
MC	Mission Command
MCSC	Marine Corps Systems Command
MOVES	Modeling, Virtual Environments, and Simulation
MSDL	Military Scenario Definition Language
NIPR	Non-classified Internet Protocol Router
NPS	Naval Postgraduate School
OBS	Order of Battle Service
OE	Operating Environment
OME	Offline Mission Editor
OOB	Order of Battle
PoR	Program of Record
RSG	Rapid Scenario Generation
SAF	Semi-Automated Forces
SIPR	Secret Internet Protocol Router
SISO	Simulation Interoperability Standards Organization
TADSS	Training Aids, Devices, Simulators and Simulations
TENA	Test and Training Enabling Architecture

TSP	Training Support Package
USA	United States Army
USMC	United States Marine Corps
VBS2	Virtual Battle Space
VTK	Virtual Training Kit
XML	Extensible Markup Language
XSLT	Extensible Stylesheet Language Transformations

I. INTRODUCTION

A. BACKGROUND

The United States Army and United States Marine Corps employ the Virtual Battlespace 2 (VBS2) commercial game for small unit training. Each service has significant investment in training scenarios constructed using VBS2 tools and conforming to the vendor's particular data formats. To move toward improved interoperability, to gain greater fiscal flexibility in addressing the statutory intent for open competition and affordability, and to protect the investment made in models, terrain, and other elements of training scenarios that are separate and distinct from the virtual and gaming environments in which the simulation executes, open standards need to be applied in place of proprietary commercial off-the-shelf architectures. In the current (and foreseeable) environment of constrained budgets, it is ever more critical that the services protect and enhance their investments in simulation systems used for training and other purposes. Expanding capabilities for open scenario interchange will improve scenario reuse while creating greater opportunities for simulation data interchange and open competition for future gaming capabilities.

The Marine Corps Systems Command has funded the Naval Postgraduate School's Modeling, Virtual Environments, and Simulation (MOVES) Institute to conduct an investigation of the application of the Extensible Markup Language (XML) for capture and exchange of VBS2 scenario data with other simulations.

XML is a wide-spread approach to describing data format and content to support efficient data processing and data interchange across systems. This report describes initial application of XML technologies to the representation of VBS2 scenario data, demonstrating feasibility for the capture and exchange of VBS2 scenario data. The report also provides a plan of action for a follow-on phase of the effort to expand the representation and for use with other XML-based standards, such as the Military Scenario Definition Language (MSDL). MSDL is an international XML standard for describing military scenarios, but has been shown to require model extensions for broader application (Beris & Whittington, 2008). While MSDL is introduced briefly in this report, a more detailed evaluation will be conducted in the subsequent phase to identify

and recommend extensions needed in MSDL to accommodate the set of data used in VBS2 scenarios.

B. SCOPE

This is the first phase of a two-phase project to develop and apply an XML data structure for capturing VBS2 scenario content. The work involves systems analysis and data modeling using XML technologies. Phase 1 will focus on learning XML, XML schemas, and the Extensible Stylesheet Language for Transformations (XSLT), in addition to gaining familiarization with VBS2 data structures and MSDL. The work will produce an initial XML schema describing VBS2 scenario data content. This project is a principal component of thesis research by two Naval Postgraduate School (NPS) graduate students (one Marine, one Army) in the Modeling, Virtual Environments, and Simulation (MOVES) curriculum, with assistance from research faculty for technical instruction, consultation, software development, and laboratory support. The students will address specific and cross-cutting issues of their respective services to maximize benefits to scenario interchange and resource reuse across both services, while also investigating acquisition policy issues for future affordability and open competition.

C. APPROACH

Phase 1 of this project involves the following tasks:

- (1) Investigate the use of XML, XML schemas, and the Extensible Stylesheet Language Translations (XSLT) for representing, storing, and manipulating VBS2 scenario data content. Provide instruction and technical support in XML and related technologies applicable to data modeling and data transformation.
- (2) Examine VBS2 data files and program content to identify information relating to the description of scenarios. Include in the analysis VBS2 representation of force structures, performance parameters (weapons, vehicles, etc.), terrain, weather, behaviors, and other aspects of the simulated battlespace. Prepare a preliminary (Phase 1) XML schema structure for capturing VBS2 data.

- (3) Evaluate application of existing standard XML formats such as MSDL to capture portions of the scenario data, identifying any extensions needed to more fully capture VBS2 content.
- (4) Prepare a plan of action for Phase 2 efforts to complete the detailed XML specification of VBS2 data content and to develop software (parsing tool) for transforming VBS2 data to/from the specified XML structure.
- (5) Perform project management and financial tracking. Provide periodic progress and status reports to the sponsor and other designated personnel/organizations.

The present report constitutes the deliverable for the Phase 1 effort, documenting accomplishments and products of the above tasks. The content of this document represents a quick-look in framing the effort, forming the foundation for follow-on work in Phase 2. The Phase 2 effort will expand on the work and products started in Phase 1, to include development of software to read and transform VBS2 content into XML files for archival storage and for interchange with other systems (i.e., across VBS2-specific data structures, the VBS2-specialized XML, and the extended MSDL structure). Phase 2 work will also include evaluation of force structure data from mission command systems to partially populate scenario initialization files, following recent advancements in the Rapid Data Generation program.

D. ORGANIZATION OF THIS DOCUMENT

Chapter I of this report provides background information and identifies the scope and technical approach for the work. Chapter II provides motivation for the project and identifies related efforts occurring in the DoD Enterprise. Chapter III examines VBS2 data structures and provides preliminary XML structures to capture selected portions of data from the VBS2 files. Chapter IV documents a preliminary XML schema for selected VBS2 data structures. Chapter V provides a preliminary XSLT document for transforming data from XML format into the VBS2 text format for selected structures identified earlier. Chapter VI gives conclusions and lessons learned, and lays out a plan for Phase 2 of the project. Appendixes provide supporting material for the narrative contained in the body of the document.

THIS PAGE INTENTIONALLY LEFT BLANK

II. MOTIVATION AND RELATED WORK

A. INTRODUCTION

There is growing concern in DoD over shrinking budgets for development and sustainment of systems, including simulation systems. Simulations play a role, to varying degrees of depth, in wide-ranging activities, such as training, education, analysis, experimentation, test and evaluation, mission planning, and mission rehearsal. Scenarios provide the operational context for these activities. That context can include geographic locale, time of the operation, date of the operation, environmental conditions, terrain, forces and force structure, equipment and supplies, control measures, planned actions, and other aspects (to include performance parameters such as the endurance of an aircraft, firing range of a weapon, detection range of a sensor, movement speed for a vehicle, etc.). For any simulation, such conditions are defined in varying levels of detail depending on the design of that simulation. In federations of simulations, it is desirable for the models to be initialized with the same information, but the ability to do so can also vary widely depending on the representational capabilities and functionality of the systems involved. Moving forward, to achieve cost savings and practical efficiencies, the enterprise needs to evolve toward “develop once, use everywhere” capabilities, which includes the description or specification of military scenarios. In this chapter, we examine current policy regarding software and data reuse, and introduce some technical initiatives attempting to address such policies.

B. REUSE POLICY AND COST SAVINGS

Reuse is simply “the practice of using again, in whole or part, existing M&S tools, data, or services” (Department of Defense, 2007). The benefits of scenario and data reuse in the M&S community are many. First, the reuse and modification of scenarios over time improves the quality of scenarios and validates the original scenario. Secondly, units utilizing reused scenarios are able to more quickly setup training events, spending less time on development of scenarios, and more time actually training individuals. Other benefits include improvements in cost effectiveness and commonality in training when reused over a large population.

Official DoD policy indicates that reuse of M&S assets is encouraged (Department of Defense, 2007). Supporting this policy is the implementation of best practices to strengthen the Defense Department's buying power and improve industry productivity, wrapped into an initiative called Better Buying Power (BBP), established by the Under Secretary of Defense for Acquisition, Technology, and Logistics (AT&L) in 2010. The BBP, and its successor the BBP 2.0, calls for new M&S acquisitions programs to enforce open architectures and promote effective competition in order to maximize the government's investment with the intent of facilitating reuse (The Under Secretary of Defense (Acquisitions, Technology, and Logistics), 2013). Further, the DoD's Acquisition M&S Master Plan specifically notes that reuse is highly desirable at any point where the asset being reused can meet a requirement more cost-effectively (Department of Defense, 2006).

C. REUSE TECHNICAL INITIATIVES

Reflecting the concern for savings in cost and time of development of new systems in DoD M&S, several efforts conducted over the past few years have been performed. Some of these efforts are described in the following subsections.

1. Military Scenario Definition Language

The Military Scenario Definition Language (MSDL) is an international standard developed by the Simulation Interoperability Standards Organization (SISO) for describing military scenario information to enable simulations and command and control (C2) systems to share data content. The MSDL standard consists of a human-readable specification and an Extensible Markup Language (XML) Schema document governing the structure and content of valid MSDL documents. The top-level structure of MSDL is shown in Figure 1.

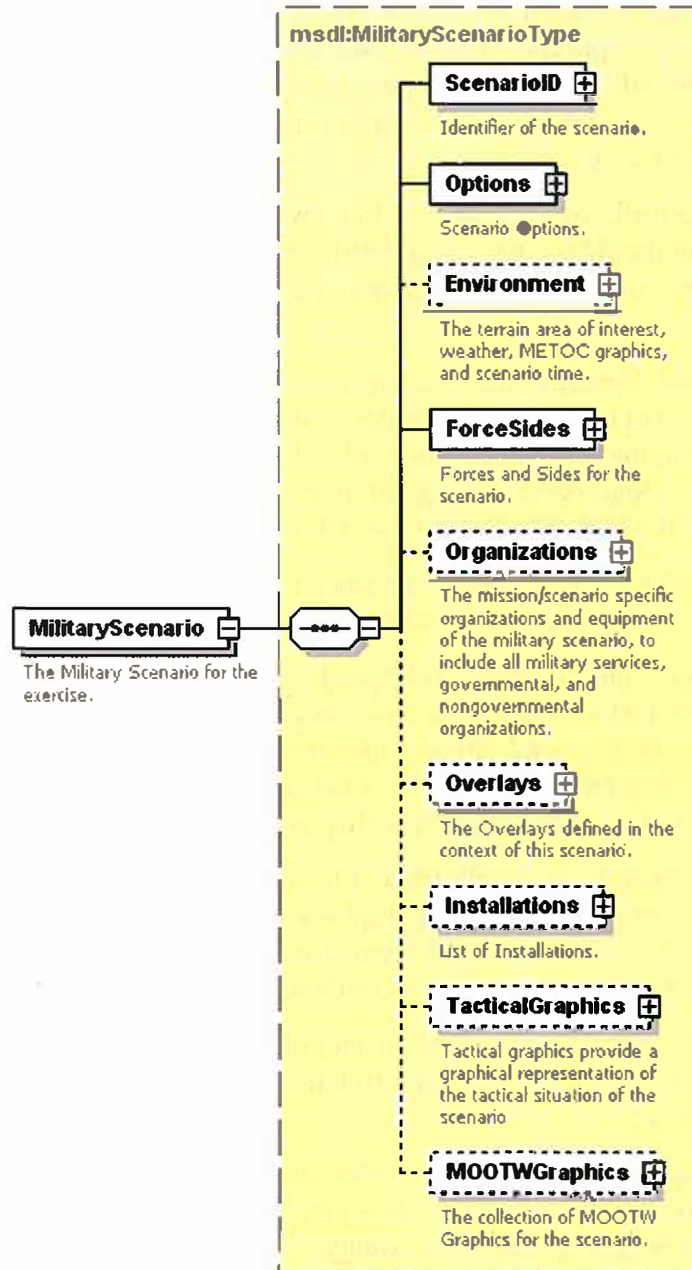


Figure 1. Top-level structure of the MSDL XML data model. From (Military Scenario Definition Language Product Development Group. SISO-STD-007-2008, 2008).

Principal components of the structure are described below:

- **ScenarioID**: Provides metadata describing the scenario, using the **modelIdentificationType** data structure defined in the Base Object Model (BOM) international standard (SISO Base Object Model Product Development Group, 2006). Information in this element includes name (required), type (required), version (required), modificationDate (required), securityClassification (required),

releaseRestriction (zero or more), purpose (optional), applicationDomain (optional), description (required), useLimitation (optional), useHistory (zero or more), keyword (zero or more), poc (one or more Point of Contact(s)), reference (zero or more), other (optional), glyph (optional), and a location for general extension of the structure.

- **Options:** Identifies the MSDL version, level of detail by echelon of organizations described in the XML file, and identifies standards for symbology (e.g., MIL-STD-2525C) and coordinate system (e.g., Military Grid Reference System, Geodetic Coordinates).
- **Environment:** Identifies the scenario start date and time, defines the bounds of the geographic area of interest, describes weather conditions (using structures from the Joint Command Control Information Exchange Data Model such as atmosphere, cloud cover, precipitation, etc.), and meteorological information (leveraging the MIL-STD-2525C specification).
- **ForceSides:** Defines the forces and sides represented in the scenario, including allegiance and associations between the forces and sides.
- **Organizations:** Defines units and equipment (entity-level) represented in the scenario. Unit data includes the force/side, command, and support relationships, communication nets, and initial disposition (position, orientation, speed, formation). Equipment information relates individual items to owning unit or force/side, and also provides initial disposition of the entity.
- **Overlays:** Organize the intelligence information described by control measures and held in Installation, TacticalGraphics, and MOOTWGraphics (Military Operations Other Than War) element structures. Overlay types include operations, fire support, modified combined obstacles, and logistics.
- **Installations:** Identifies installations included in the scenario, providing owning unit of the data/graphic, affiliation (relative to the data owner), and associations to other overlays.
- **TacticalGraphics:** Provides information on C2 graphics used in the scenario, including the owning unit of the data/graphic, affiliation, and associated overlays. Six types of tactical graphic symbology sets are defined: point symbols, line symbols, area symbols, boundary symbols, Nuclear/Biological/Chemical (NBC) symbols, and task symbols.
- **MOOTWGraphics:** Provides information on MOOTW graphics, including the owning unit of the data/graphic, affiliation, and associated overlays. MOOTW symbology sets cover such events as violent activities and various kinds of operations (recruitment, demonstrations, psychological operations, etc.).

2. Joint Training Data Services (JTDS)

The Joint Training Data Services (JTDS) is an initiative to define authoritative data sources and data preparation techniques to support storage, access, and utilization of

information in C2 systems and simulation systems (Chambers, 2012). JTDS is an online web-enabled system, available worldwide 24/7 via the Non-classified Internet Protocol Router (NIPR) and Secret Internet Protocol Router (SIPR) networks. It is a fast single repository for data (200+ million records) which possesses intuitive drag and drop scenario building tools. It provides an integrated rule set for Joint Live, Virtual, and Constructive (JLVC) data model error checking. JTDS also provides output as a standardized documented XML file, providing a common data set for JLVC initiation. The data produced for scenario events can be archived for reuse. Initial efforts have focused on the representation and exchange of order of battle information (e.g., force structures). Current efforts are focusing on representation and exchange of terrain and environmental data.

3. Rapid Data Generation

McGroder & Lashlee (2011) describe a DoD M&S Enterprise Data Strategy involving data production, discovery, and sharing in accordance with principles of the DoD Net-Centric Data Sharing Strategy (Department of Defense, 2004a). The Rapid Data Generation (RDG) project is working to address the long-standing challenges of preparation and reuse of data across the DoD M&S community (Scudder et al., 2011): "...the RDG project represents an implementation of the Net-Centric Data and Services Strategies for the M&S Enterprise, and is the DoD M&S Steering Committee's flagship project to implement the DoD M&S Enterprise Data Strategy" (p. 5). A principal element of the RDG solution is the Common Data Production Environment (CDPE), shown in Figure 2.

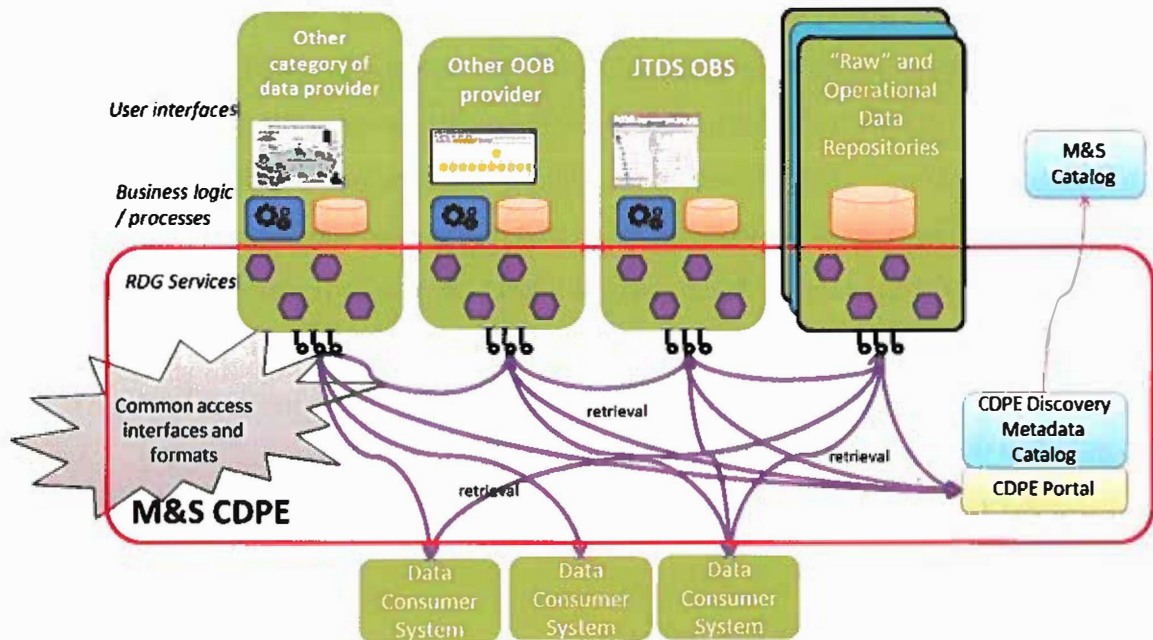


Figure 2. Conceptual view of the RDG Common Data Production Environment data discovery capabilities. From (Scudder et al., 2011).

Operational requirements of the CDPE include the following (Scudder et al., 2011):

- “provide users a capability to discover data to meet M&S requirements—both existing datasets built by M&S data integrators and data from Authoritative Data Sources;
- provide users a capability to view metadata and instance data with data-specific viewers;
- provide users a capability to retrieve data in intermediate M&S data exchange formats;
- provide users a capability to manipulate/fuse/correlate data; and
- provide access to these capabilities through a web-based CDPE portal and/or web service (SOA) interfaces (to allow integration and use of CDPE capabilities in user-developed applications).”

Initial efforts in the RDG project focused on Red and Blue order of battle (OOB). The project is now working on data and processes for geospatial, atmosphere, space, and ocean data. Future areas include Logistics and Command and Control (C2).

4. Rapid Scenario Generation

The U.S. Army Training and Doctrine Command (TRADOC) Analysis Center, Monterey (TRAC-Monterey), together with the Naval Postgraduate School, performed a study of the applicability of MSDL to support scenario data sharing across scenario generation tools, the Infantry Warfighting Simulation (IWARS), and agent-based models such as the Map-Aware Non-uniform Automata (MANA) and Pythagoras. The Rapid Scenario Generation project (Alt et al., 2007) developed a prototype scenario generation tool and demonstrated the use of MSDL to store and exchange common scenario data across multiple simulations, while also employing extensibility mechanisms in the language to store scenario content that was unique to each system. In this manner, the MSDL structure was used for archival storage of scenario data as well as for data interchange across different simulations. An overview of the RSG scenario generation and execution process is shown in Figure 3 and described below:

- (1) Develop course of action sketch using the RSG tool.
- (2) Save the scenario data to various formats for simulation tool-specific editing and execution.
- (3) From the RSG tool, launch the Design of Experiment (DOE) Tool Editor and import the scenario data. Select desired factors and factor levels to generate the design. Launch the DOE Tool Executor to generate scenario files with the requested design points. Save out the scenario files for the simulation-specific engine. Execute the selected simulation to conduct the experiment and generate reports.

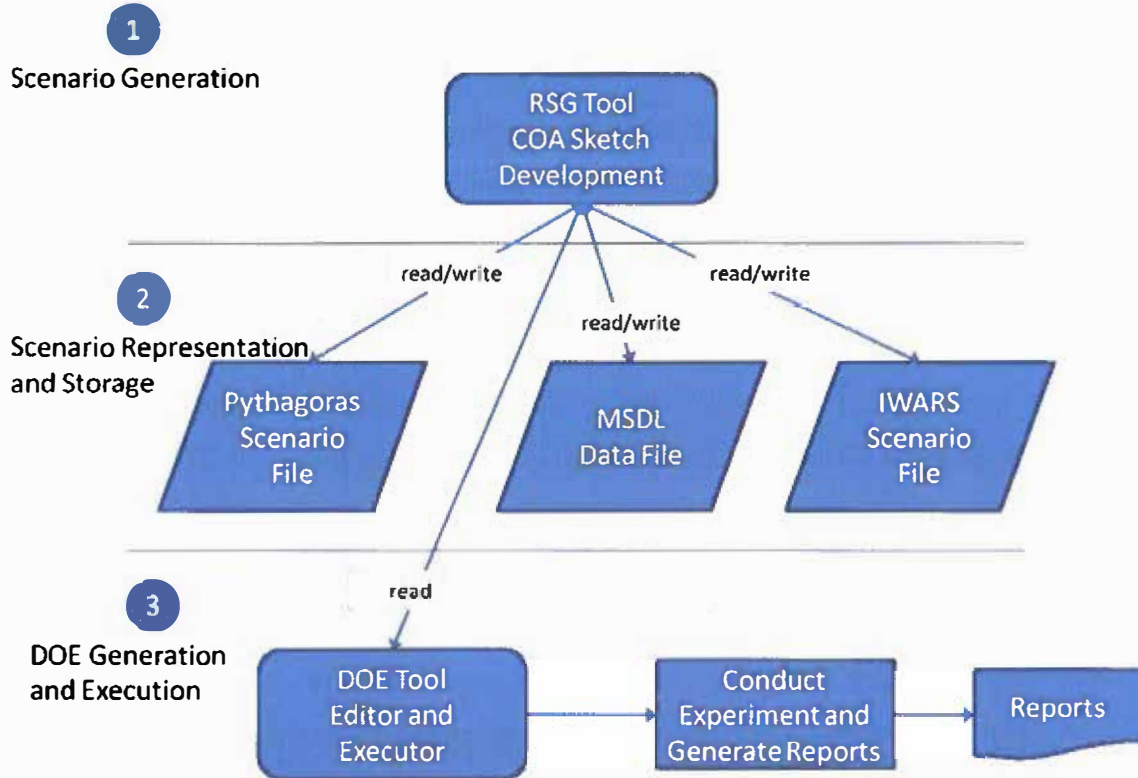


Figure 3. RSG project scenario generation and execution process. After (Blais, Dodds, Pearman, & Baez, 2009).

One example of the data translation through MSDL is shown in Figure 4, where a scenario was first constructed using the RSG tool, saved to MSDL, converted to Pythagoras scenario data format, and loaded into Pythagoras for execution.

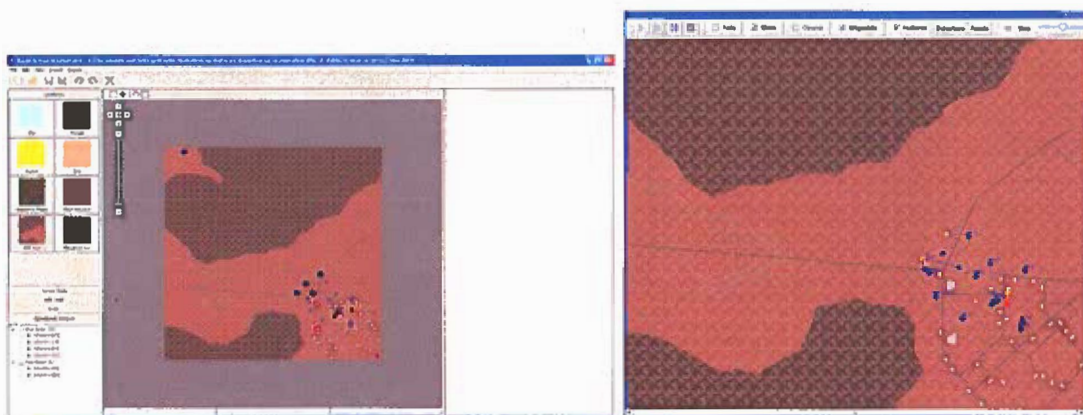


Figure 4. Scenario construction in RSG (left) and execution in Pythagoras (right). From (Blais et al., 2009).

5. Interoperability Frameworks

Interoperability—“the ability of systems, units, or forces to provide data, information, materiel, and services to and accept the same from other systems, units, or forces and to use the data, information, materiel, and services so exchanged to enable them to operate effectively together” (Department of Defense, 2004b, p. 15)—is a different challenge than software and data reuse, but it has the common concern of being able to share information across multiple systems. DoD and the component services have developed several interoperability frameworks over the past 20+ years, such as Distributed Interactive Simulation (DIS), High Level Architecture (HLA), Test and Training Enabling Architecture (TENA), Close Combat Tactical Trainer (CCTT), Modeling Architecture for Technology, Research, and Experimentation (MATREX), Joint Simulation Bus (JBUS), and the Live-Virtual-Constructive Integrating Architecture (LVC-IA). It is beyond the scope of this initial study to examine the initialization capabilities and challenges across these frameworks, but suffice to say that data consistency across systems linked within these frameworks remains a significant and costly effort. The JDS efforts discussed previously are motivated by these challenges, together with interactions between M&S systems and real-world C2 systems.

The Live, Virtual, Constructive – Integrating Architecture (LVC-IA) is the U.S. Army’s Program of Record (PoR) to provide protocols, standards, and interfaces for interoperability of currently dissimilar Training Aides, Devices, Simulations, and Simulators (TADSS) supporting Live, Virtual, Constructive, and Gaming (LVC-G) training enablers so that they can simultaneously stimulate Mission Command (MC) systems. LVC-IA is the materiel solution to connect TADSS and move toward an Integrated Training Environment (ITE), a system of systems that combines or technically connects support tools and selected TADSS in a persistent and consistent manner. The ITE leverages MC systems to meet the commander’s training objectives within the appropriate Operational Environment (OE) and is capable of supporting individual and multi-echelon collective training within all of the Army’s Training Domains (Myers, 2012). This is an important step for Army Gaming, because it recognizes the interoperability requirement for small unit training and larger constructive exercises. Linking “connected” VBS2 squads and platoons to MC systems enables a commander to

fully integrate multi-echelon training. The LVC-IA program has integrated VBS2 in its increment 1 upgrade.

D. SUMMARY

This chapter provided additional background information, discussing the need for greater information sharing across systems and preservation of investments in scenario development. Several projects related to these needs were presented to provide a foundation for the technical approach and objectives of the current effort. Additional initiatives relevant to this project will be examined during the Phase 2 effort.

III. VBS2 SCENARIO DATA

A. INTRODUCTION

This chapter provides a description of major components of VBS2 scenario data files. Preliminary XML structures capturing the content of specific excerpts from the VBS2 data files are proposed to show the fundamental viability of the approach.

B. VBS2 DATA STRUCTURES

VBS2 scenario data are stored in several files that are combined for loading into the simulation prior to execution. There are three files required for VBS2 scenarios to be run locally (not in a networked training mission utilizing a dedicated server). These files are as follows (Bohemia Interactive, 2012):

- **mission.sqm:** describes which units in a scenario are playable by the user (as opposed to artificial intelligence controlled) , and defines groups of players or units. This file is essential in starting a networked scenario. VBS2 parses this file first, meaning that units and groups are the first elements of the scenario to be created during initialization.
- **mission.sqf:** a script file which executes after all of the units and groups have been initialized. This file will create vehicles, waypoints, markers, and any other control measures or editor objects built into the scenario.
- **mission.biedi:** describes ALL editor objects in the scenario. This file is essentially the combination of the mission.sqm and mission.sqf files, and when edited directly outside of the VBS2 program, will initialize additional editor objects not saved through the VBS2 Offline Mission Editor (OME), as long as they are correctly formatted.

All of the above listed files are text files, readable with any open source text editor. While VBS2 help manuals indicate that directly editing these files should never be attempted due to the possibility of corrupting the mission, our research indicates that when done correctly, there is no adverse effect in editing the files directly, particularly the .biedi file, which is of the most interest to us, as it consolidates all of the editor objects in the scenario. For running networked training missions, VBS2 wraps the three mission files into a compressed format ending in “.pbo,” which when unwrapped by the

game engine during initialization, expands back into a file tree structure. The .pbo files used for networked and training scenarios are generally encoded and cannot be directly edited using a text editor in the same manner as the mission files can.

For example, the following code snippet was generated by VBS2 and saved into a .biedi file through the OME native to VBS2. It represents a single rifleman in the simulation, labeled as “_unit_0”.

```
class _unit_0
{
    objectType="unit";
    layer="classes";
    height=2.7128906e-005;
    class Arguments
    {
        NAME="rifleman";
        URN="rifleman1";
        DESCRIPTION="rifleman";
        SIDE="WEST";
        FILTER_BY_FORCE="";
        FILTER="US";
        TYPE="vbs2_us_mc_rifleman_D_m16a2";
        BASE_TYPE="";
        EDIT_UNIT="";
        NEW_UNIT="";
        DELETE_UNIT="";
        IS_EDIT="";
        COMBATMODE="YELLOW";
        BEHAVIOUR="SAFE";
        PLAYER="true";
        ADVANCED="";
        BG1="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG2="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG3="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG5="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG6="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG7="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG8="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG9="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG10="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG11="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG12="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG13="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG14="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        HEALTH="1.00000";
        AMMO="1.00000";
        PROBOFPRES="1.00000";
        TRAINING="0.50000";
        EXPERIENCE="0.20000";
        ENDURANCE="0.50000";
    }
}
```



```

LEADERSHIP="0.20000";
FLEECHANCE="0.75000";
HEIGHT="1.82000";
BMI="0.00000";
AZIMUT="0.00000";
POSITION="[4918.57861, 6790.73486, 640.72897]";
COLOR="[0.600000023841858,0.600000023841858,1]";
SUB_TEAM="";
DISPLAY_NAME="(P) (rifleman) rifleman1";
DISPLAY_NAME_TREE="(P) (rifleman) rifleman1";
PARENT_UNIT="";
LOADOUT="";
};
};

```

This excerpt declares an instance of a “unit” class named `_unit_0` and assigns values to several attributes, such as `NAME`, `TYPE`, `SIDE`, and `POSITION`. Figures 5, 6, and 7 illustrate the rifleman’s first-person view of the battlefield when loaded into the simulation, and the rifleman’s attributes in the OME when the scenario is loaded, respectively. For reference, the full `.biedi` file containing all of the objects and units in our test scenario is provided in Appendix A.

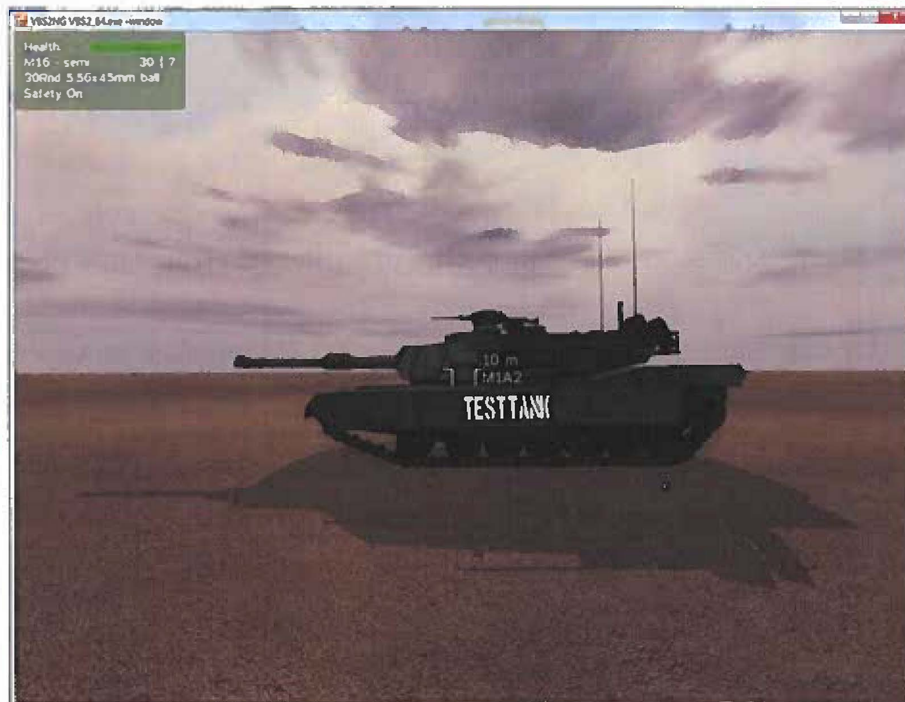


Figure 5. Rifleman’s first-person view after being created in the VBS2 OME. The ‘TESTTANK’ model is present as a point of reference for comparison to later figures in this document.

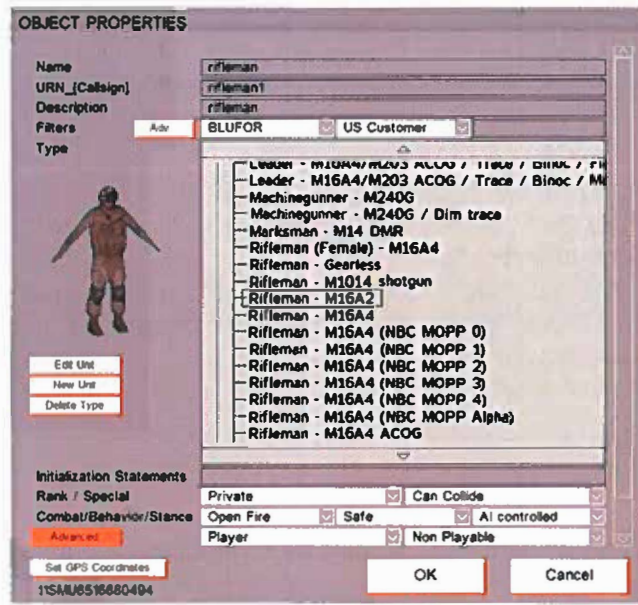


Figure 6. Basic properties of a rifleman created in VBS2 OME.

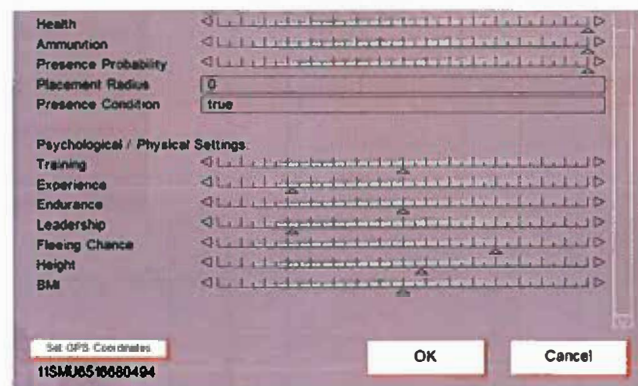


Figure 7. Advanced properties of a rifleman created in VBS2 OME.

By examining the above excerpt and the full file provided in Appendix A, a number of object classes (indicated by the objectClass values) and VBS2 arguments of those classes can be identified. The full set of objects that can be defined in a scenario are listed in the user interface (OME) and include: Artillery, ArtilleryStrike, Boomerang, Camera, ChemLight, ControlLink, CREWLink, EmptyVehicle, FleePoint, GlintObject, Group, IED (improvised explosive device), Intel, LightBeacon, LightSource, LookAt, Marker, MeasureDistance, Mine, Object, Record, Sandstorm, Script, SoundSource, Surrender, Trigger, Unit, VBS2Fires, VBS2StrikeAircraft, Vehicle, VerbalCommand, and Waypoint. A preliminary spreadsheet identifying the argument lists for several of these objects is provided in Appendix B. The spreadsheet data helps to identify common

subsets of arguments across the various object classes to assist in further defining the XML structure in the next phase of the project.

C. PRELIMINARY XML FORMAT FOR SELECTED VBS2 DATA STRUCTURES

Using the example excerpt from the previous section, the following provides an XML structure that captures the relevant structure and content from the VBS2 data:

```
<?xml version="1.0" encoding="UTF-8"?>
<vbs:VBS2Scenario xmlns:vbs="http://www.nps.edu/XML/schemas/scenarios/VBS2"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.nps.edu/XML/schemas/scenarios/VBS2 VBS2Classes.xsd">
  <Classes>
    <PrefixClass vbs:name="_prefix_0" vbs:objectType="prefix">
      <Version>8</Version>
      <AddOns/>
      <Layers/>
    </PrefixClass>
    <UnitClass vbs:name="_unit_0" vbs:objectType="unit" height="2.7128906e-005"
      layer="classes">
      <Name>rifleman</Name>
      <URN>rifleman1</URN>
      <Description>rifleman</Description>
      <Side>WEST</Side>
      <FilterByForce/>
      <Filter>US</Filter>
      <Type>vbs2_us_mc_rifleman_D_m16a2</Type>
      <BaseType/>
      <EditUnit/>
      <NewUnit/>
      <DeleteUnit/>
      <IsEdit/>
      <Init/>
      <CombatMode>YELLOW</CombatMode>
      <Behaviour>SAFE</Behaviour>
      <Player>true</Player>
      <Advanced/>
      <Backgrounds>
        <Background id="BG1"
          file="\vbs2\ui\Data\NewEditor\slider_back_rad.paa"/>
        <Background id="BG2"
          file="\vbs2\ui\Data\NewEditor\slider_back_rad.paa"/>
        <Background id="BG3"
          file="\vbs2\ui\Data\NewEditor\slider_back_rad.paa"/>
        <Background id="BG5"
          file="\vbs2\ui\Data\NewEditor\slider_back_rad.paa"/>
        <Background id="BG6"
          file="\vbs2\ui\Data\NewEditor\slider_back_rad.paa"/>
        <Background id="BG7"
          file="\vbs2\ui\Data\NewEditor\slider_back_rad.paa"/>
        <Background id="BG8"
          file="\vbs2\ui\Data\NewEditor\slider_back_rad.paa"/>
        <Background id="BG9"
          file="\vbs2\ui\Data\NewEditor\slider_back_rad.paa"/>
        <Background id="BG10"
          file="\vbs2\ui\Data\NewEditor\slider_back_rad.paa"/>
        <Background id="BG11"
          file="\vbs2\ui\Data\NewEditor\slider_back_rad.paa"/>
        <Background id="BG12"
          file="\vbs2\ui\Data\NewEditor\slider_back_rad.paa"/>
      </Backgrounds>
    </UnitClass>
  </Classes>
</vbs:VBS2Scenario>
```

```

        file="\vbs2\ui\Data\NewEditor\slider_back_rad.paa"/>
    <Background id="BG13"
        file="\vbs2\ui\Data\NewEditor\slider_back_rad.paa"/>
    <Background id="BG14"
        file="\vbs2\ui\Data\NewEditor\slider_back_rad.paa"/>
</Backgrounds>
<Health>1.00000</Health>
<Ammo>1.00000</Ammo>
<ProbabilityOfPresence>1.00000</ProbabilityOfPresence>
<Training>0.50000</Training>
<Experience>0.20000</Experience>
<Endurance>0.50000</Endurance>
<Leadership>0.20000</Leadership>
<FleeChance>0.75000</FleeChance>
<Height>1.82000</Height>
<BMI>0.00000</BMI>
<Azimuth>89.66441</Azimuth>
<Position posX="4909.20947" posY="6738.80030" posZ="640.76577"/>
<Color hue="0.600000023841858" saturation="0.600000023841858"
    lightness="1"/>
<SubTeam/>
<DisplayName> (P) (rifleman) rifleman1</DisplayName>
<DisplayNameTree> (P) (rifleman) rifleman1</DisplayNameTree>
<ParentUnit/>
<LoadOut/>
</UnitClass>
    <PostfixClass vbs:name="_postfix_0" vbs:objectType="postfix"/>
</Classes>
</vbs:VBS2Scenario>

```

D. SUMMARY

As shown in this chapter, XML structures can be created to capture and store the content of VBS2 scenario data files. In the next chapter, we build on this initial work by specifying an XML schema for selected portions of the VBS2 scenario data content to demonstrate how the data structure and content can be more formally described for use by other systems.

IV. PRELIMINARY XML SCHEMA

A. INTRODUCTION

This chapter uses the preliminary XML structures proposed in the previous chapter for selected VBS2 scenario data files to specify an XML schema that describes the respective VBS2 structure and content. This process opens up the VBS2 data descriptions for human and machine understandability.

B. SELECTED VBS2 DATA STRUCTURES AND ASSOCIATED SCHEMA DESIGN

Using the example class and associated XML structure from the previous chapter, we can specify the structure and content of this portion of the VBS2 scenario data with the XML Schema language, as follows (the structure is illustrated in Figure 8):

```
<xs:complexType name="UnitClassType">
  <xs:sequence>
    <xs:element name="Name" type="xs:string"/>
    <xs:element name="URN" type="xs:string"/>
    <xs:element name="Description" type="xs:string"/>
    <xs:element name="Side" type="xs:string"/>
    <xs:element name="FilterByForce" type="xs:string"/>
    <xs:element name="Filter" type="xs:string"/>
    <xs:element name="Type" type="xs:string"/>
    <xs:element name="BaseType" type="xs:string"/>
    <xs:element name="EditUnit" type="xs:string"/>
    <xs:element name="NewUnit" type="xs:string"/>
    <xs:element name="DeleteUnit" type="xs:string"/>
    <xs:element name="IsEdit" type="xs:string"/>
    <xs:element name="Init" type="xs:string" minOccurs="0"/>
    <xs:element name="CombatMode" type="xs:string"/>
    <xs:element name="Behaviour" type="xs:string"/>
    <xs:element name="Player" type="xs:boolean"/>
    <xs:element name="Advanced" type="xs:string"/>
    <xs:element name="Backgrounds" type="BackgroundsType"
minOccurs="0"/>
    <xs:element name="Health" type="xs:decimal"/>
    <xs:element name="Ammo" type="xs:decimal"/>
    <xs:element name="ProbabilityOfPresence" type="xs:decimal"/>
    <xs:element name="Training" type="xs:decimal"/>
    <xs:element name="Experience" type="xs:decimal"/>
    <xs:element name="Endurance" type="xs:decimal"/>
    <xs:element name="Leadership" type="xs:decimal"/>
    <xs:element name="FleeChance" type="xs:decimal"/>
    <xs:element name="Height" type="xs:decimal"/>
    <xs:element name="BMI" type="xs:decimal"/>
    <xs:element name="Azimuth" type="xs:decimal"/>
    <xs:element name="Position" type="PositionType"/>
    <xs:element name="Color" type="ColorType"/>
    <xs:element name="SubTeam" type="xs:string"/>
```

```

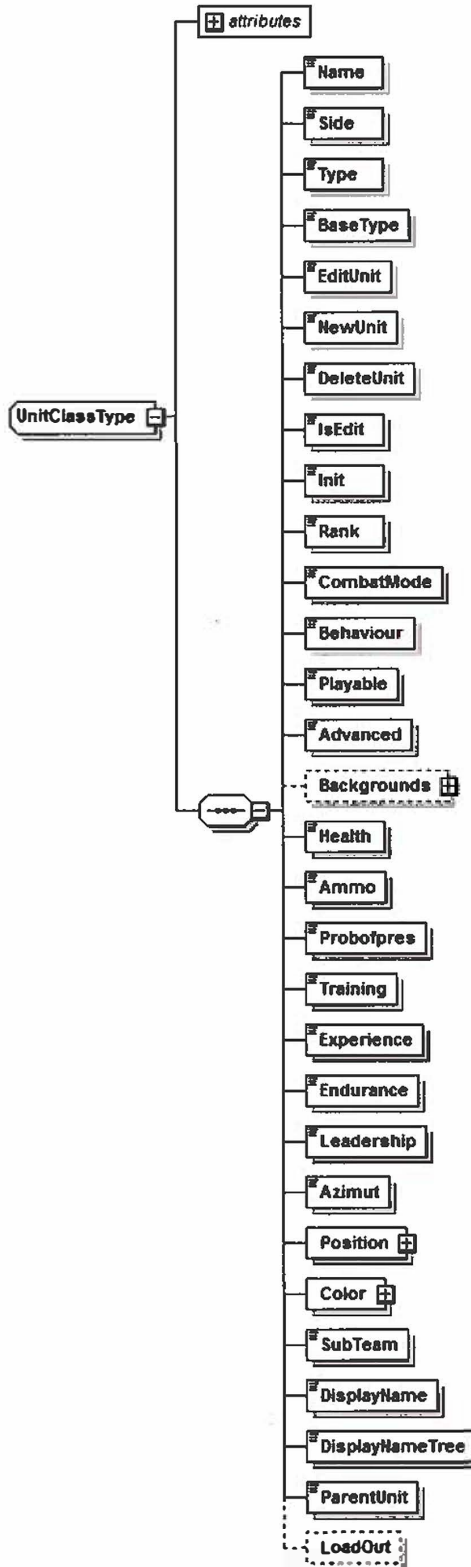
        <xs:element name="DisplayName" type="xs:string"/>
        <xs:element name="DisplayNameTree" type="xs:string"/>
        <xs:element name="ParentUnit" type="xs:string"/>
        <xs:element name="LoadOut" type="LoadOutType" minOccurs="0"/>
    </xs:sequence>
    <xs:attributeGroup ref="commonClassAttributes"/>
</xs:complexType>

```

This captures the class attributes and arguments for the VBS2 unit object as an XML element structure with associated attributes and child elements. Figure 9 illustrates the XML structure represented by this schema (note: only a few of the child elements of the Classes element have been defined at this stage of the project). The full XML schema for this structure is provided in Appendix C. This schema is incomplete at this time, representing just the work performed for the Phase 1 effort. The schema will be refined and completed during the Phase 2 effort.

C. SUMMARY

This chapter has shown that an XML schema can be developed to describe the structure of the various object classes that can appear in a VBS2 scenario. The schema developed here is very preliminary; many of the data structures need to be further defined and greater precision entered for the declaration of valid content of the data (ranges of values, string enumerations, string patterns, etc.). In the next chapter, we develop an initial XSLT file to show how the XML representation of a selected excerpt from the VBS2 scenario file can be transformed back to the VBS2 format.



Generated by XMLSpy

www.altova.com

Figure 8. XML structure for VBS2 unit class.

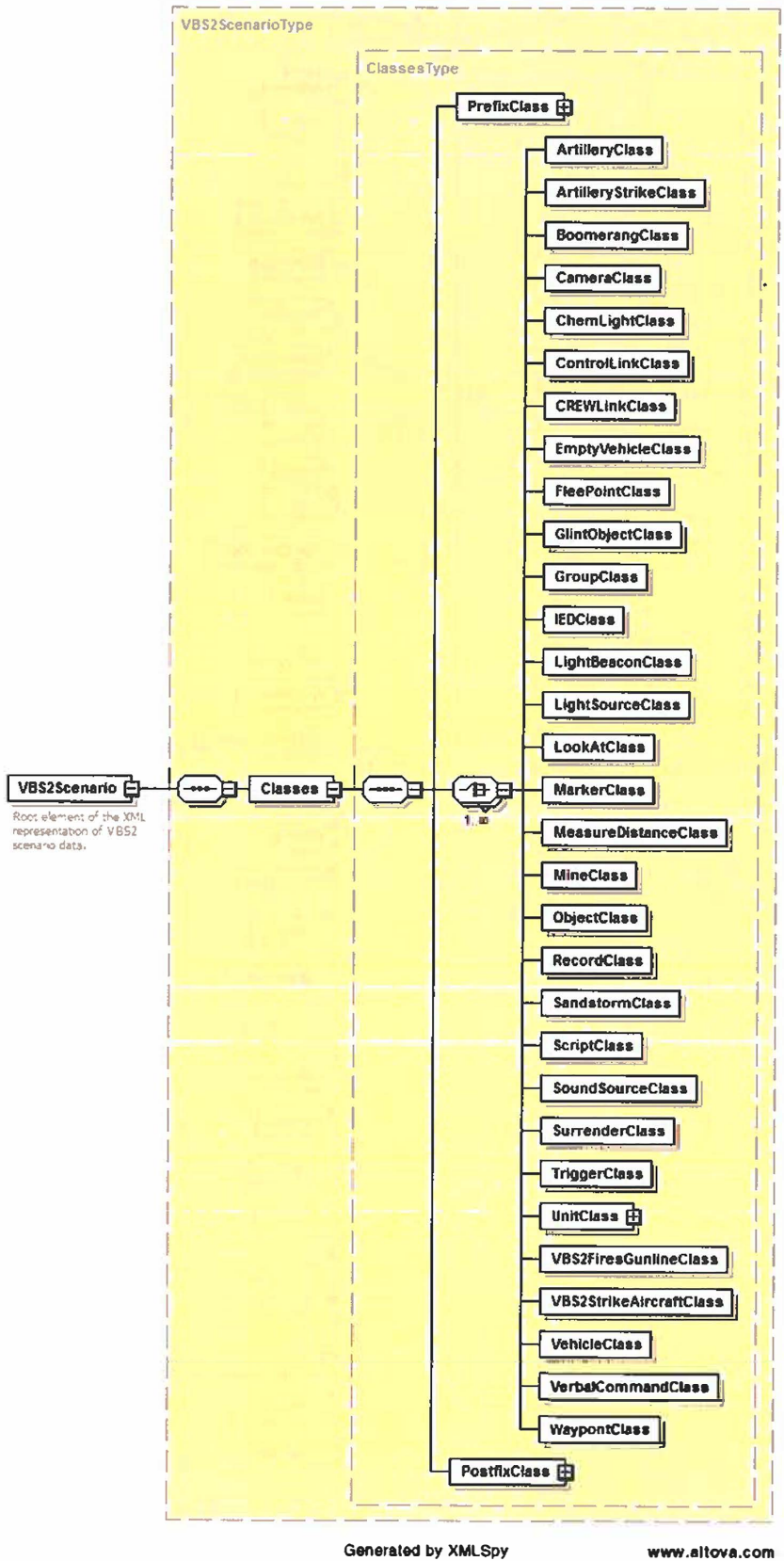


Figure 9. XML structure for VBS2 scenario object classes.

V. TRANSFORMING XML CONTENT TO VBS2 DATA STRUCTURES

A. INTRODUCTION

This chapter demonstrates the use of XML as a means to capture VBS2 data in a way that allows other software to read, validate, process, and write out the data using standard XML tools and techniques. We assume the existence of software (to be developed in Phase 2 of this project) that can read VBS2 data files and generate XML files that conform to the XML schema described in the previous chapter. Based on the XML representation of VBS2 data presented in the previous chapters, we develop an Extensible Stylesheet Language Transformations (XSLT) (World Wide Web Consortium (W3C), 1999) document that reads from the XML document and reproduces an excerpt of a VBS2 file that is verified to be correct by showing that the scenario data still loads correctly into the VBS2 software.

B. XSLT DESIGN

XSLT is an XML language that provides instructions for reading data from an XML file and writing out a text file. The output file can be another XML format, Hyper-Text Markup Language (HTML), or other text-based format. In our case, we want to read data from the XML representation of VBS2 scenario data and write out a result file containing that data but in the VBS2 format, as seen in the earlier example and provided in Appendix A. This process is shown in Figure 10. Once the VBS2 data content is available in XML, transformations can be readily developed to interchange information with other XML-based data models, such as MSDL and IWARS, as well as with real-world data such as the Order of Battle Services (OBS) and National Information Exchange Model (NIEM). Follow-on work will also consider how metadata standards, such as the DoD Discovery Metadata Specification (DDMS) and the Intelligence Community Information Security Markup (IC-ISM), can be integrated into the XML structure to enable the scenario data to become discoverable across the DoD data enterprise.

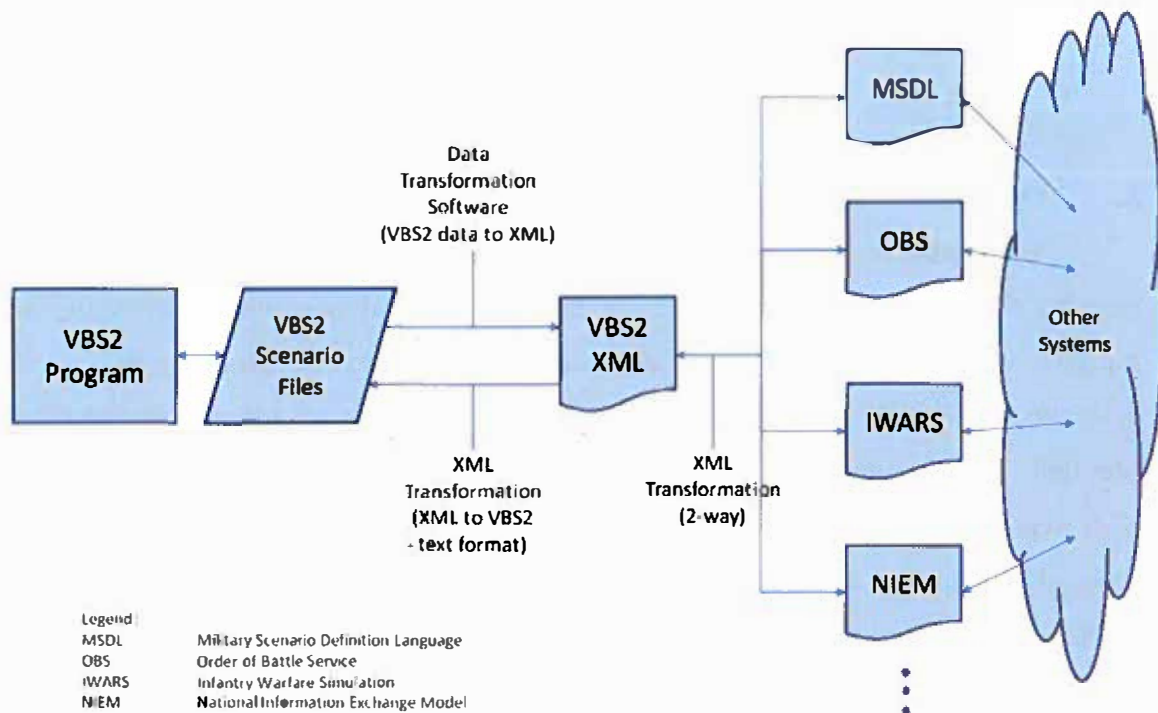


Figure 10. Data transformations between VBS2 scenario data and XML-based data models through an XML representation of the VBS2 data.

For example, the following XSLT excerpt reads data from selected elements in the XML structure (`xsl:value-of` statements) and writes out the data in VBS2 format, as seen in section III.B:

```
<xsl:text>class </xsl:text>
<xsl:value-of select="@vbs:name"/>
{
  objectType="<xsl:value-of select="@vbs:objectType"/>";
  layer="<xsl:value-of select="@layer"/>";
  height="<xsl:value-of select="@height"/>";
  class Arguments {
    NAME="<xsl:value-of select="Name"/>";
    URN="<xsl:value-of select="URN"/>";
    DESCRIPTION="<xsl:value-of select="Description"/>";
  }
}
```

This XSLT excerpt generates the following output from the source XML file:

```
class _unit_0
{
  objectType="unit";
  layer="classes";
  height=0.000027128906;
  class Arguments {
    NAME="rifleman";
    URN="rifleman1";
    DESCRIPTION="rifleman";
  }
}
```

The full XSLT document to transform the XML representation of the VBS2 unit object class is provided in Appendix D.

C. EXAMPLE DATA TRANSLATIONS

Application of the XSLT above to the earlier example XML file results in the following VBS2 data structure defining a unit:

```
class _unit_0
{
    objectType="unit";
    layer="classes";
    height=0.000027128906;
    class Arguments {
        NAME="rifleman";
        URN="rifleman1";
        DESCRIPTION="rifleman";
        SIDE="WEST";
        FILTER_BY_FORCE="";
        FILTER="US";
        TYPE="vbs2_us_mc_rifleman_D_m16a2";
        BASE_TYPE="";
        EDIT_UNIT="";
        NEW_UNIT="";
        DELETE_UNIT="";
        IS_EDIT="";

        RANK="";
        COMBATMODE="YELLOW";
        BEHAVIOUR="SAFE";
        PLAYER="true";
        ADVANCED="";
        BG1="vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG2="vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG3="vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG5="vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG6="vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG7="vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG8="vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG9="vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG10="vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG11="vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG12="vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG13="vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG14="vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        HEALTH="1";
        AMMO="1";
        PROBOFPRES="1";
        TRAINING="0.5";
        EXPERIENCE="0.2";
        ENDURANCE="0.5";
        LEADERSHIP="0.2";
        FLEECHANCE="0.75";
        HEIGHT="1.82";
        BMI="0";
    }
}
```

```

AZIMUT="89.66441";
POSITION="[4909.20947, 6738.8003, 640.76577]";
COLOR="[0.600000023841858,0.600000023841858,1]";
SUB_TEAM="";
DISPLAY_NAME=" (P) (rifleman) rifleman1";
DISPLAY_NAME_TREE=" (P) (rifleman) rifleman1";
PARENT_UNIT="";
LOADOUT="";
};
};

```

It is asserted that this content from the data transformation is equivalent (other than insignificant white space characters) to the respective portion of the source VBS2 data file, as will be assessed in the next section.

D. VERIFICATION OF CORRECT DATA TRANSLATION

To confirm accuracy and completeness of the XSLT result file, the excerpt shown above was copied and pasted into the original VBS2 scenario file, replacing the original definition of the unit class named “_unit_0.” This excerpt was pasted directly into the VBS2 .biedi file previously generated by the VBS2 OME with outputs displayed in Figures 5-7. Resultant figures 11-13 show that VBS2 software successfully loaded the same model and its associated position, orientation, and advanced properties into the simulation.

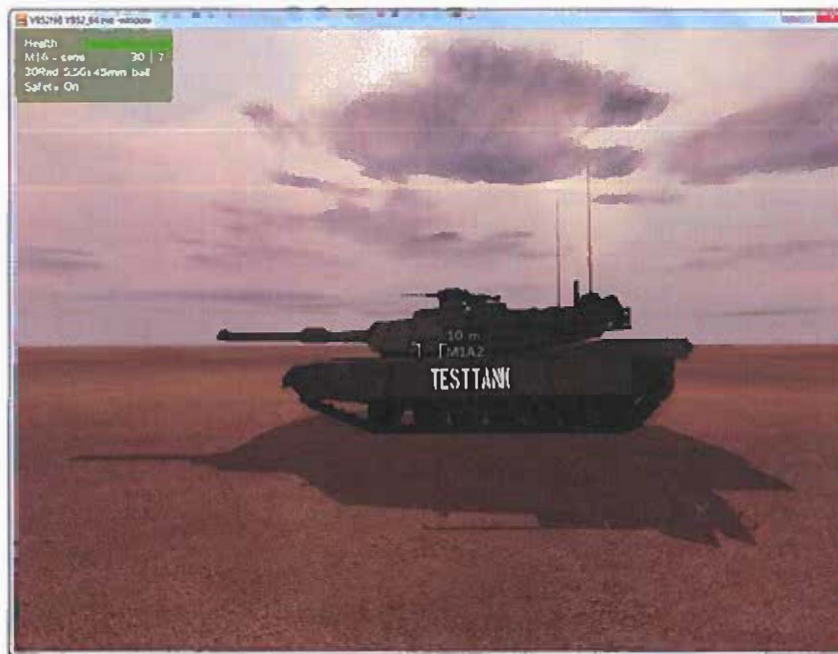


Figure 11. Rifleman’s first-person view after translation through XML and XSLT, subsequently pasted directly into the .biedi initialization file and previewed in VBS2.

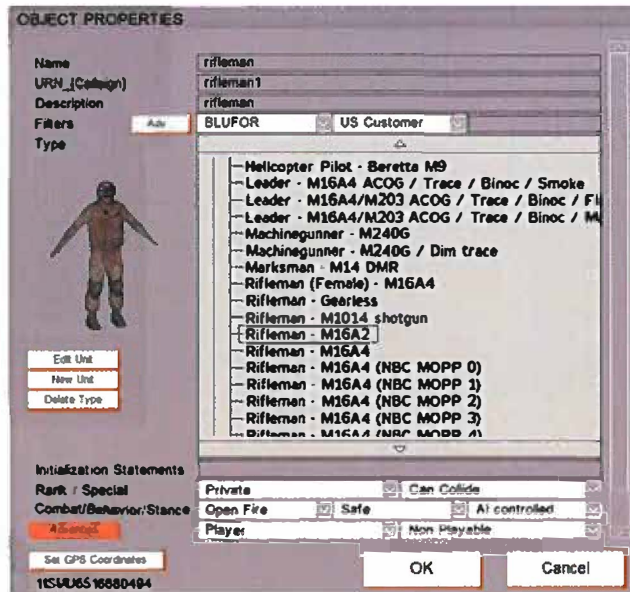


Figure 12. Basic properties of a rifleman loaded through XML and XSLT.

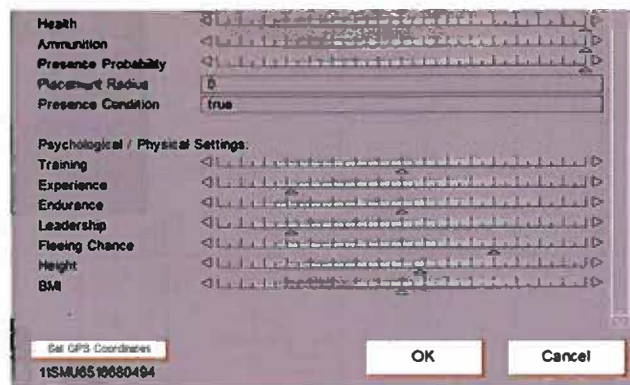


Figure 13. Advanced properties of a rifleman loaded through XML and XSLT.

E. SUMMARY

This chapter demonstrated the ability to write XML transformation logic to convert from an XML representation of VBS2 data back into the textual format of VBS2 data files. It solidified the feasibility of utilizing XSLT to transform XML files back into a .biedi file format, with no loss of functionality in the simulation runtime environment.

THIS PAGE INTENTIONALLY LEFT BLANK

VI. CONCLUSIONS, LESSONS LEARNED, AND NEXT STEPS

A. CONCLUSIONS

The Phase 1 project successfully demonstrated design of XML data structures to capture an important portion of the VBS2 scenario data content, and showed the capability to transform content from the XML model back to the VBS2 scenario data formats.

B. LESSONS LEARNED

In conducting this initial look at creating an XML representation of VBS2, the project team verified that training scenarios developed on U.S. Army licensed VBS2 platforms can be executed in USMC- licensed versions of VBS2. A training support package (TSP) developed in VBS2 Version 1.4 and licensed to the U.S. Army was downloaded from the U.S. Army MilGaming portal (TCM Gaming, 2010). The TSP files were then loaded onto a USMC Deployable Virtual Training Environment (DVTE) laptop running VBS2 Virtual Training Kit (VTK) Version 1.4. The scenario performed correctly in all observable aspects when run locally, which was the extent of the test. Models and terrain data loaded correctly, and interaction of the user-playable models with the scenario's artificial intelligence (AI) driven models was as expected.

Due to the late start of the Phase 1 effort, the project team was able to perform only a very cursory investigation into storage of data from VBS2 scenario files into MSDL data structures. There is potential for the usual interchange of unit/entity name, location, and orientation/disposition, together with use of the general extensibility capability in MSDL as demonstrated by the RSG project (see II.C.4) to store portions of VBS2 data that do not associate readily with elements in the MSDL data structure. More research is needed in Phase 2 to determine how best to employ MSDL, whether directly (transferring VBS2 scenario data content directly into MSDL structures), as part of the XML representation of VBS2 scenario data (employing MSDL XML data structures in the XML representation through reference to the MSDL namespace and schema), or through transformation, as implied in the data flow shown in Chapter V (Figure 10).

The Phase 1 work to date has raised a number of questions regarding VBS2 data. While not prohibiting the Phase 2 effort, finding answers to these questions will assist us

in making the XML data structure and content as accurate as possible. Representative questions are listed below (note: there is no significance in the ordering of these questions); the team will add to this list as questions arise during the project and will remove questions as they are answered.

- Q1: What are the dimensions of color (red-green-blue, hue-saturation-value, hue-saturation-lightness, etc.) in the COLOR attribute in VBS2 class definitions?
- Q2: How is the COLOR attribute given its values in the VBS2 user interface?
- Q3: Where are weapon, sensor, and vehicle performance characteristics provided in the input data (e.g., vehicle speed, weapon ranges, firing rates, probabilities of hit/kill, etc.)?
- Q4: Are users able to set/modify weapon, sensor, and vehicle performance characteristics (if so, using what user interface)?
- Q5: Are all decimal values in the VBS2 data represented as fixed point decimal or floating point values?
- Q6: Can we obtain VBS2 source code for parsing VBS2 scenario input files? Being able to work with that code would reduce the level of effort to develop and test software to parse VBS2 scenario input files to produce XML representations of the data.

C. PLAN OF ACTION FOR PHASE 2

The content of this document represents a quick-look in framing the effort, forming the foundation for follow-on work in Phase 2. The Phase 2 effort will expand on the work and products started in Phase 1, to include development of software to read and transform VBS2 content into XML files for archival storage and for interchange with other systems (i.e., across VBS2-specific data structures, the VBS2-specialized XML, and the extended MSDL structure). Phase 2 work will also include evaluation of force structure data from mission command systems to partially populate scenario initialization files, following recent advancements in the Rapid Data Generation program. The following tasks need to be performed in Phase 2 of the project:

- Investigate and identify the VBS2 scenario object class attributes and arguments not completed in Phase 1, looking for commonality and difference with the objects examined in Phase 1.
- Examine the use of metadata standards for discovery (i.e., DDMS) and security markings (i.e., IC-ISM), as well as related standards for the M&S community, such as the M&S Community of Interest Discovery Metadata Specification (MSC-DMS). Determine a strategy for employing these in the XML representation of VBS2 scenario data.
- Examine the use of other scenario data models, such as MSDL and OBS, for use in the XML representation of VBS2 scenario data. Determine a strategy for employing these in the XML representation of VBS2 scenario data through direct use of data constructs or through XML data transformations.
- Investigate other areas of VBS2 data (e.g., weapon/vehicle/sensor performance characteristics, behaviors, terrain, environment, etc.). Determine a strategy for including these constructs in the XML representation of VBS2 scenario data.
- Re-design, rework, and refine the preliminary XML schema started in Phase 1 according to the findings of the above activities.
- Develop software to parse VBS2 data files into the XML representation, if possible (i.e., if access to the source code can be obtained) by modifying existing VBS2 source code that currently exists to read VBS2 data files (i.e., reuse the code to read the VBS2 data files, then add logic to write out the XML representation of that data in conformance to the XML schema).
- Demonstrate successful round-trip data transformation from VBS2 scenario data files to XML and back to VBS2 format.
- Demonstrate interchange of VBS2 scenario data with other data formats through the intermediate XML representation of VBS2 data.

In addition to the above technical tasks, the project team will provide periodic reporting of technical progress and project financial status during performance of the Phase 2 effort.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. VBS2 OBJECT CLASSES FOR A NOTIONAL SCENARIO

The following is a VBS2 .biedi scenario file containing data describing a number of example VBS2 object classes; namely: group, IED, intel, marker, object, trigger, unit, vehicle, and waypoint (as well as prefix and postfix classes).

```
class _prefix_0
{
    objectType="prefix";
    class Arguments
    {
        VERSION="8";
    };
    layers[]=
    {
        "classes"
    };
    DeletedMapObjects[]={};
    TerrainModifs[]={0,0,0,0,0};
};
class _unit_0
{
    objectType="unit";
    layer="classes";
    height=2.2929687e-005;
    class Arguments
    {
        NAME="rifleman";
        URN="rifleman1";
        DESCRIPTION="rifleman";
        SIDE="west";
        FILTER_BY_FORCE="";
        FILTER="US";
        TYPE="vbs2_us_mc_rifleman_D_m16a2";
        BASE_TYPE="";
        EDIT_UNIT="";
        NEW_UNIT="";
        DELETE_UNIT="";
        IS_EDIT="";
        COMBATMODE="YELLOW";
        BEHAVIOUR="SAFE";
        PLAYER="true";
        ADVANCED="";
        IS_ADVANCED="true";
        BG1="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG2="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG3="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG5="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG6="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG7="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG8="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
    };
};
```

```

        BG9="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG10="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG11="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG12="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG13="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        BG14="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        HEALTH="1.00000";
        AMMO="1.00000";
        PROBOFPRES="1.00000";
        TRAINING="0.50000";
        EXPERIENCE="0.20000";
        ENDURANCE="0.50000";
        LEADERSHIP="0.20000";
        FLEECHANCE="0.75000";
        HEIGHT="1.82000";
        BMI="0.00000";
        AZIMUT="89.66441";
        POSITION="{4909.20947, 6738.80030, 640.76577}";
        COLOR="{0.600000023841858,0.600000023841858,1}";
        SUB_TEAM="";
        DISPLAY_NAME="(P) (rifleman) rifleman1";
        DISPLAY_NAME_TREE="(P) (rifleman) rifleman1";
        PARENT_UNIT="";
        LOADOUT="";
    };
};
class _group_0
{
    objectType="group";
    layer="classes";
    height=-0.0014379101;
    class Arguments
    {
        NAME="TalibanGroup";
        DISPLAY_NAME="TalibanGroup";
        DISPLAY_NAME_TREE="TalibanGroup";
        POSITION="{5061.01269, 6751.86133, 639.80917}";
        SIDE="east";
        SELECTED_SIDE="east";
        FILTER2="US";
        CATEGORY="vbs2_af_taliban";
        TYPE_GROUP="rifleteam";
        GROUP_MEMBERS_NAME="";
        GROUPFRAME="";
        GROUP_MEMBERS2="vbs2_af_taliban_ak74gla";

        PREVIEW_PIC2="\VBS2\people\afg\afg_man\data\ico\preview_afg_fight
er1_ca.paa";
        CREATE_GROUP="";
        SIDE_NEW="west";
        CATEGORY_NEW="West";
        CATEGORY_DELETE="";
        TYPE_NEW="West";
        TYPE_DELETE="";
        GROUP_MEMBERS="";
        FILTER="US";
        AVAILBLE_MEMBERS="";
    }
}

```

```

MEMBER_UP="";
MEMBER_DOWN="";
ADD_MEMBER="";
REMOVE_MEMBER="";
COPY_MEMBER="";
EDIT_UNIT2="";
NEW_UNIT2="";
IS_EDIT="";
};
};
class _unit_1
{
  objectType="unit";
  layer="classes";
  height=-0.0014379101;
  class Arguments
  {
    SIDE="EAST";
    TYPE="vbs2_af_taliban_ak74gla";
    RANK="Corporal";
    AZIMUT="0.00000";
    POSITION="[5061.01269, 6751.86133, 639.80917]";
    COLOR="[1,0.5,0.5]";
    LEADER="";
    DISPLAY_NAME="Taliban - AK-74 / GP-25";
    DISPLAY_NAME_TREE="Taliban - AK-74 / GP-25";
    PARENT="_group_0";
    PARENT_UNIT="";
  };
};
class _unit_2
{
  objectType="unit";
  layer="classes";
  height=-0.0014345313;
  class Arguments
  {
    SIDE="EAST";
    TYPE="vbs2_af_taliban_rpk74";
    RANK="Private";
    AZIMUT="0.00000";
    POSITION="[5065.01269, 6750.86133, 639.78421]";
    COLOR="[1,0.5,0.5]";
    LEADER="_unit_1";
    DISPLAY_NAME="Taliban - RPK-74";
    DISPLAY_NAME_TREE="Taliban - RPK-74";
    PARENT="_group_0";
    PARENT_UNIT="";
  };
};
class _unit_3
{
  objectType="unit";
  layer="classes";
  height=-0.0014395899;
  class Arguments
  {

```

```

        SIDE="EAST";
        TYPE="vbs2_af_taliban_akm";
        RANK="Private";
        AZIMUT="0.00000";
        POSITION="[5057.01269, 6750.86133, 639.83639]";
        COLOR="[1,0.5,0.5]";
        LEADER="_unit_1";
        DISPLAY_NAME="Taliban - AKM";
        DISPLAY_NAME_TREE="Taliban - AKM";
        PARENT="_group_0";
        PARENT_UNIT="";
    };
};
class _unit_4
{
    objectType="unit";
    layer="classes";
    height=-0.0014363282;
    class Arguments
    {
        SIDE="EAST";
        TYPE="vbs2_af_taliban_akm";
        RANK="Private";
        AZIMUT="0.00000";
        POSITION="[5069.01269, 6749.86133, 639.75955]";
        COLOR="[1,0.5,0.5]";
        LEADER="_unit_1";
        DISPLAY_NAME="Taliban - AKM";
        DISPLAY_NAME_TREE="Taliban - AKM";
        PARENT="_group_0";
        PARENT_UNIT="";
    };
};
class _marker_0
{
    objectType="marker";
    layer="classes";
    height=-0.0014379101;
    class Arguments
    {
        POSITION="[5061.01269, 6751.86133, 639.80917]";
        NAME="m1";
        LASTNAME="m1";
        MARKER_TYPE="TacticalMarker";
        COLOR="ColorRed";
        PARENT="_group_0";
        PARENT_UNIT="_unit_1";
        DISPLAY_NAME="Marker m1";
        SIDE="F2";
        UNIT="I22";
        SIZE="M2";

        PREVIEW_SIDE="\vbs2\ui\tacticmarkers\data\Frames\Hostile_Surface";

        PREVIEW_UNIT="\vbs2\ui\tacticmarkers\data\Icons\H_Infantry";
    };
};

```

```

PREVIEW_SIZE="\vbs2\ui\tacticmarkers\data\Modifiers\S_Team";

PREVIEW_MODIFIER="\vbs2\ui\tacticmarkers\data\SubRoles\Blanc";
};
class DynamicArguments
{
    name="TacticalMarker";
    class Params
    {
        class CUSTOM_MARKER
        {
            type="config";
            subtype="CfgMarkers,name";
            description="Custom Marker";
            w=0.34;
            idc=5001;
            onChanged="[_map] call compile preprocessFile
""\vbs2\editor\Data\Scripts\marker\customMarkers\markerUIcustomLBChange
d.sqf""";
        };
        class CUSTOM_DELETE
        {
            type="button";
            x=0.56;
            y=0.11;
            w=0.1;
            onInit="_control_CUSTOM_DELETE ctrlSetText
(localize ""STR_CA_DELETE ""); _control_CUSTOM_DELETE
ctrlSetEventHandler [""ButtonClick"",preProcessFile
""\vbs2\editor\Data\Scripts\marker\customMarkers\markerUIcustomDelete.s
qf""]";
        };
        class CUSTOM_NAME
        {
            type="text";
            description="Save As";
            w=0.34;
            idc=5002;
        };
        class CUSTOM_SAVE
        {
            type="button";
            x=0.56;
            y=0.14;
            w=0.1;
            onInit="_control_CUSTOM_SAVE ctrlSetText
(localize ""STR_DISP_ARCMAP_SAVE ""); _control_CUSTOM_SAVE
ctrlSetEventHandler [""ButtonClick"",preProcessFile
""\vbs2\editor\Data\Scripts\marker\customMarkers\markerUIcustomSave.sqf
""]";
        };
        class SIDE
        {
            type="configEx";
            subtype="cfgTacticalSymbols.Sides,name";
            description="Side Icon";
        };
    };
};

```

```

        default="f1";
        w=0.25;
        idc=5026;
        onInit="_sel = __control_SIDE lbData lbCurSel
_control_SIDE; lbSort _control_SIDE; _was = _sel; for ' i' from 0 to
(lbSize _control_SIDE) -1 do {if (_control_SIDE lbData _i == _sel) then
{ _was = _i; _control_SIDE lbSetCurSel _i}};['addMRK_SIDE', ctrlText
_control_NAME != '', 0, 'lb', _control_SIDE, _was] call compile
preProcessFile '\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";
        onChanged="_this call
_func_COC_Marker_DlgChanged";
    };
    class SIDE_POSITION
    {
        type="number";
        subtype="range(-0.05, 0.05)";
        description="Side Icon";
        default="0";
        hidden=1;
        w=0.15000001;
        onChanged="[_this, _control_PREVIEW_SIDE, ""X"" ]
call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
    };
    class SIDE_DOWN
    {
        type="number";
        subtype="range(-0.05, 0.05)";

        description="str_editor_edobj_param_placeholder";
        default="0";
        hidden=1;
        x=0.36250001;
        w=0.15000001;
        onChanged="[_this, _control_PREVIEW_SIDE, ""Y"" ]
call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
    };
    class SIDE_SCALE
    {
        type="number";
        subtype="range(0.5, 1.5)";
        description="Scale";
        default="1";
        x=0.47;
        y=0.17;
        w=0.15000001;
        idc=5027;

        onChanged="[_this, _control_PREVIEW_SIDE, _control_SIDE_POSITION, _c
ontrol_SIDE_DOWN] call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerScale.sqf""");
    };
    class UNIT
    {
        type="configEx";
        subtype="cfgTacticalSymbols.Units,name";

```



```

        description="Unit Icon";
        default="I1";
        w=0.25;
        idc=5028;
        onInit="_sel = _control_UNIT lbData lbCurSel
_control_UNIT; lbSort _control_UNIT; _was = _sel; for 'i' from 0 to
(lbSize _control_UNIT) -1 do {if (_control_UNIT lbData i == _sel) then
{ _was = i; _control_UNIT lbSetCurSel i}}; ['addMRK_UNIT', ctrlText
_control_NAME != '', 0, 'lb', _control_UNIT, _was] call compile
preProcessFile '\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";
        onChanged="_this call
_func_COC_Marker_DlgChanged";
    };
    class UNIT_POSITION
    {
        type="number";
        subtype="range(-0.05, 0.05)";

        description="str_editor_edobj_param_placeholder";
        default="0";
        hidden=1;
        w=0.15000001;
        onChanged="[_this, _control_PREVIEW_UNIT, "X"]";
    call compile preprocessfile
    ("'\vbs2\editor\Data\Scripts\marker\layerPosition.sqf");
    };
    class UNIT_DOWN
    {
        type="number";
        subtype="range(-0.05, 0.05)";

        description="str_editor_edobj_param_placeholder";
        default="0";
        hidden=1;
        x=0.36250001;
        y=0.259999999;
        w=0.15000001;
        onChanged="[_this, _control_PREVIEW_UNIT, "Y"]";
    call compile preprocessfile
    ("'\vbs2\editor\Data\Scripts\marker\layerPosition.sqf");
    };
    class UNIT_SCALE
    {
        type="number";
        subtype="range(0.5, 1.5)";
        description="Scale";
        default="1";
        x=0.47;
        y=0.2;
        w=0.15000001;
        idc=5029;
        onInit="['addMRK_UNIT_SCALE', ctrlText
_control_NAME != '', 0, 'sld', _control_UNIT_SCALE] call compile
preProcessFile '\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";

        onChanged="[_this, _control_PREVIEW_UNIT, _control_UNIT_POSITION, c

```

```

ontrol_UNIT_DOWN] call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerScale.sqf""");
};
class SIZE
{
    type="configEx";
    subtype="cfgTacticalSymbols.Size,name";
    description="Size Icon";
    w=0.25;
    idc=5030;
    onInit=["'addMRK_SIZE', ctrlText _control_NAME
!= '', 0, 'lb', _control_SIZE] call compile preprocessfile
'\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";
    onChanged="_this call
_func_COC_Marker_DlgChanged";
};
class SIZE_POSITION
{
    type="number";
    subtype="range(-0.05, 0.05)";

    description="str_editor_edobj_param_placeholder";
    default="0";
    hidden=1;
    w=0.15000001;
    onChanged=["_this, _control_PREVIEW_SIZE, ""X""]
call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
};
class SIZE_DOWN
{
    type="number";
    subtype="range(-0.05, 0.05)";

    description="str_editor_edobj_param_placeholder";
    default="0";
    hidden=1;
    x=0.36250001;
    y=0.31999999;
    w=0.15000001;
    onChanged=["_this, _control_PREVIEW_SIZE, ""Y""]
call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
};
class SIZE_SCALE
{
    type="number";
    subtype="range(0.5, 1.5)";
    description="Scale";
    default="1";
    x=0.47;
    y=0.23;
    w=0.15000001;
    idc=5031;
    onInit=["'addMRK_SIZE_SCALE', ctrlText
_control_NAME != '', 0, 'sld', _control_SIZE_SCALE] call compile
preProcessFile '\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";
};

```

```

    onChanged="[_this,_control_PREVIEW_SIZE,_control_SIZE_POSITION,_c
ontrol_SIZE_DOWN] call compile preprocessfile
(""\vbs2\editor\Data\Scripts\marker\layerScale.sqf""");
    };
    class MODIFIER
    {
        type="configEx";
        subtype="cfgTacticalSymbols.Modifiers,name";
        description="Modifier Icon";
        default="S0";
        w=0.25;
        idc=5032;
        onInit="_sel = _control_MODIFIER lbData
lbCurSel _control_MODIFIER; lbSort _control_MODIFIER; _was = _sel; for
'i' from 0 to (lbSize _control_MODIFIER) -1 do {if (_control_MODIFIER
lbData_i == _sel) then {_was = _i; _control_MODIFIER lbSetCurSel
_i}};['addMRK_MODIFIER', ctrlText _control_NAME != '', 0, 'lb',
_control_MODIFIER,_was] call compile preProcessFile
'\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";
        onChanged="_this call
_func_COC_Marker_DlgChanged";
    };
    class MODIFIER_POSITION
    {
        type="number";
        subtype="range(-0.05, 0.05)";

        description="str_editor_edobj_param_placeholder";
        default="0";
        hidden=1;
        w=0.15000001;

        onChanged="[_this,_control_PREVIEW_MODIFIER,""X""] call compile
preprocessfile
(""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
    };
    class MODIFIER_DOWN
    {
        type="number";
        subtype="range(-0.05, 0.05)";

        description="str_editor_edobj_param_placeholder";
        default="0";
        hidden=1;
        x=0.36250001;
        y=0.38;
        w=0.15000001;

        onChanged="[_this,_control_PREVIEW_MODIFIER,""Y""] call compile
preprocessfile
(""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
    };
    class MODIFIER_SCALE
    {
        type="number";
        subtype="range(0.5, 1.5)";

```

```

        description="Scale";
        default="1";
        x=0.47;
        y=0.25999999;
        w=0.15000001;
        idc=5033;
        onInit=["'addMRK_MODIFIER_SCALE', ctrlText
_control_NAME != '', 0, 'sld', _control_MODIFIER_SCALE] call compile
preProcessFile '\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";

        onChanged=["_this,_control_PREVIEW_MODIFIER,_control_MODIFIER_POS
ITION,_control_MODIFIER_DOWN] call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerScale.sqf""");
    };
    class PREVIEW_SIDE
    {
        type="picture";
        description="Preview";
    };
    class PREVIEW_UNIT
    {
        type="picture";

description="str_editor_edobj_param_placeholder";
        x=0.20999999;
        y=0.28999999;
        w=0.15000001;
        h=0.2;
    };
    class PREVIEW_SIZE
    {
        type="picture";

description="str_editor_edobj_param_placeholder";
        x=0.20999999;
        y=0.28999999;
        w=0.15000001;
        h=0.2;
    };
    class PREVIEW_MODIFIER
    {
        type="picture";

description="str_editor_edobj_param_placeholder";
        x=0.20999999;
        y=0.28999999;
        w=0.15000001;
        h=0.2;
    };
};
create[]=
{
    "%VARIABLE_NAME =
([""%VARIABLE_NAME"", %NAME, %TEXT, %MARKER_TYPE, """"", %COLOR, [%A,
%B], %ANGLE, %POSITION, %CONDITION, %PREVIEW_SIDE, %PREVIEW_UNIT, %PREVIEW_SI
ZE, %PREVIEW_MODIFIER, [%SIDE_POSITION, %SIDE_DOWN], %SIDE_SCALE, [%SIZE_POS
ITION, %SIZE_DOWN], %SIZE_SCALE, [%UNIT_POSITION, %UNIT_DOWN], %UNIT_SCALE, [

```

```

%MODIFIER_POSITION,%MODIFIER_DOWN],%MODIFIER_SCALE,%AUTOSCALE] +
[%PARENT_UNIT]) call _func_COC_Marker_Tactical_Create;"
};
update[]=
{
    "[%NAME,%TEXT,%MARKER_TYPE,""",%COLOR,[%A,
%B],%ANGLE,%POSITION,%CONDITION,%PREVIEW_SIDE,%PREVIEW_UNIT,%PREVIEW_SI
ZE,%PREVIEW_MODIFIER,[%SIDE_POSITION,%SIDE_DOWN],%SIDE_SCALE,[%SIZE_POS
ITION,%SIZE_DOWN],%SIZE_SCALE,[%UNIT_POSITION,%UNIT_DOWN],%UNIT_SCALE,[
%MODIFIER_POSITION,%MODIFIER_DOWN],%MODIFIER_SCALE,""%PARENT"","%VARIA
BLE_NAME"",%AUTOSCALE] call _func_COC_Marker_Tactical_Update;"
};
execDrawMap="false";
};
};
class _waypoint_0
{
    objectType="waypoint";
    layer="classes";
    height=9.6093754e-006;
    class Arguments
    {
        VECTOR_DIRECTION="";
        POSITION="[4991.55029, 6793.38965, 640.25416]";
        AZIMUT="0.00000";
        TYPE="MOVE";
        DESCRIPTION="MOVEwaypoint";
        DISPLAY_NAME="MOVE (MOVEwaypoint)";
        DISPLAY_NAME_TREE="MOVE (MOVEwaypoint)";
        SCRIPT="hint str (1 + 1)";
        AVRS_ACTION="";
        NAME="";
    };
};
class _group_1
{
    objectType="group";
    layer="classes";
    height=9.6875001e-006;
    class Arguments
    {
        DISPLAY_NAME="B 1-1-A-2";
        DISPLAY_NAME_TREE="B 1-1-A-2";
        POSITION="[4919.25684, 6737.82324, 640.71583]";
        SIDE="WEST";
    };
};
class _vehicle_1
{
    objectType="vehicle";
    layer="classes";
    height=2.7929686e-006;
    class Arguments
    {
        NAME="testEmptyVehicle";
        URN="testEmptyVehicle";
        URN_DIS="testEmptyVehicle";
    };
};

```

```

SUB_TEAM="";
SIDE="all";
FILTER_BY_FORCE="";
FILTER="US";
TYPE="VBS2_US_ARMY_M1A2_D_X";
EDIT_UNIT="";
NEW_UNIT="";
DELETE_UNIT="";
IS_EDIT="";
HAS_CREW="false";
ADVANCED="";
BG1="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
FUEL="1.00000";
BG2="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
HEALTH="1.00000";
BG3="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
AMMO="1.00000";
BG4="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
PROBOFPRES="1.00000";
AZIMUT="-0.00172";
VECTORDIR="[-0.00003,0.99999,-0.00435]";
VECTORUP="[0.00616,0.00435,0.99997]";
POSITION="[4971.83398, 6741.41211, 640.40314]";
COLOR="[0.5,0.5,0]";
DISPLAY_NAME="(testEmptyVehicle) testEmptyVehicle";
DISPLAY_NAME_TREE="(testEmptyVehicle) testEmptyVehicle";
};
};
class _marker_2
{
    objectType="marker";
    layer="classes";
    height=-4.2382812e-006;
    class Arguments
    {
        POSITION="[5007.68164, 6744.40332, 640.16534]";
        NAME="testMarker";
        LASTNAME="testMarker";
        TEXT="testMarker";
        MARKER_TYPE="TacticalMarker";
        COLOR="ColorBlue";
        DISPLAY_NAME="Marker testMarker (testMarker)";

        HIDDENS=["""_hidden_0""","""_hidden_1""","""_hidden_2""","""_hidden_3""",
        ""_hidden_4""","""_hidden_5""","""_hidden_6""","""_hidden_7"""];
        CUSTOM_MARKER="new";
        CUSTOM_DELETE="";
        CUSTOM_SAVE="";
        SIDE_POSITION="-0.00000";
        SIDE_DOWN="-0.00000";
        SIDE_SCALE="1.00000";
        UNIT_POSITION="-0.00000";
        UNIT_DOWN="-0.00000";
        UNIT_SCALE="1.00000";
        SIZE="M1";
        SIZE_POSITION="-0.00000";
        SIZE_DOWN="-0.00000";
    }
}

```

```

SIZE_SCALE="1.00000";
MODIFIER_POSITION="-0.00000";
MODIFIER_DOWN="-0.00000";
MODIFIER_SCALE="1.00000";

PREVIEW_SIDE="\vbs2\ui\tacticmarkers\data\Frames\Friend_Units";
PREVIEW_UNIT="\vbs2\ui\tacticmarkers\data\Icons\Blanc";
PREVIEW_SIZE="\vbs2\ui\tacticmarkers\data\Modifiers\Blanc";

PREVIEW_MODIFIER="\vbs2\ui\tacticmarkers\data\SubRoles\Blanc";
);
class DynamicArguments
{
    name="TacticalMarker";
    class Params
    {
        class CUSTOM_MARKER
        {
            type="config";
            subtype="CfgMarkers,name";
            description="Custom Marker";
            w=0.34;
            idc=5001;
            onChanged="[_map] call compile preprocessFile
""\vbs2\editor\Data\Scripts\marker\customMarkers\markerUIcustomLBChange
d.sqf""";
        };
        class CUSTOM_DELETE
        {
            type="button";
            x=0.56;
            y=0.11;
            w=0.1;
            onInit="_control_CUSTOM_DELETE ctrlSetText
(localize ""STR_CA_DELETE ""); _control_CUSTOM_DELETE
ctrlSetEventHandler [""ButtonClick"",preProcessFile
""\vbs2\editor\Data\Scripts\marker\customMarkers\markerUIcustomDelete.s
qf""]";
        };
        class CUSTOM_NAME
        {
            type="text";
            description="Save As";
            w=0.34;
            idc=5002;
        };
        class CUSTOM_SAVE
        {
            type="button";
            x=0.56;
            y=0.14;
            w=0.1;
            onInit="_control_CUSTOM_SAVE ctrlSetText
(localize ""STR_DISP_ARCMAP_SAVE ""); _control_CUSTOM_SAVE
ctrlSetEventHandler [""ButtonClick"",preProcessFile
""\vbs2\editor\Data\Scripts\marker\customMarkers\markerUIcustomSave.sqf
""]";
        };
    };
};

```

```

    };
class SIDE
{
    type="configEx";
    subtype="cfgTacticalSymbols.Sides,name";
    description="Side Icon";
    default="f1";
    w=0.25;
    idc=5026;
    onInit="_sel = _control_SIDE lbData lbCurSel
_control_SIDE; lbSort _control_SIDE; _was = _sel; for 'i' from 0 to
(lbSize _control_SIDE) -1 do (if (_control_SIDE lbData i == _sel) then
{_was = i; _control_SIDE lbSetCurSel i});['addMRK_SIDE', ctrlText
_control_NAME != '', 0, 'lb', _control_SIDE, _was] call compile
preProcessFile '\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";
    onChanged="_this call
_func_COC_Marker_DlgChanged";
};
class SIDE_POSITION
{
    type="number";
    subtype="range(-0.05, 0.05)";
    description="Side Icon";
    default="0";
    hidden=1;
    w=0.15000001;
    onChanged="[_this, _control_PREVIEW_SIDE, ""X"""]
call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
};
class SIDE_DOWN
{
    type="number";
    subtype="range(-0.05, 0.05)";

    description="str_editor_edobj_param_placeholder";
    default="0";
    hidden=1;
    x=0.36250001;
    w=0.15000001;
    onChanged="[_this, _control_PREVIEW_SIDE, ""Y"""]
call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
};
class SIDE_SCALE
{
    type="number";
    subtype="range(0.5, 1.5)";
    description="Scale";
    default="1";
    x=0.47;
    y=0.17;
    w=0.15000001;
    idc=5027;

    onChanged="[_this, _control_PREVIEW_SIDE, _control_SIDE_POSITION, _c

```



```

ontrol_SIDE_DOWN] call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerScale.sqf""");
};
class UNIT
{
    type="configEx";
    subtype="cfgTacticalSymbols.Units,name";
    description="Unit Icon";
    default="I1";
    w=0.25;
    idc=5028;
    onInit=_sel = _control_UNIT lbData lbCurSel
_control_UNIT; lbSort _control_UNIT; _was = _sel; for '_i' from 0 to
(lbSize _control_UNIT) -1 do {if (_control_UNIT lbData _i == _sel) then
{_was = _i; _control_UNIT lbSetCurSel _i}};{'addMRK_UNIT', ctrlText
_control_NAME != '', 0, 'lb', _control_UNIT, _was] call compile
preProcessFile '\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";
    onChanged="_this call
_func_COC_Marker_DlgChanged";
};
class UNIT_POSITION
{
    type="number";
    subtype="range(-0.05, 0.05)";

    description="str_editor_edobj_param_placeholder";
    default="0";
    hidden=1;
    w=0.15000001;
    onChanged="[_this, _control_PREVIEW_UNIT, "X"]";
call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
};
class UNIT_DOWN
{
    type="number";
    subtype="range(-0.05, 0.05)";

    description="str_editor_edobj_param_placeholder";
    default="0";
    hidden=1;
    x=0.36250001;
    y=0.25999999;
    w=0.15000001;
    onChanged="[_this, _control_PREVIEW_UNIT, "Y"]";
call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
};
class UNIT_SCALE
{
    type="number";
    subtype="range(0.5, 1.5)";
    description="Scale";
    default="1";
    x=0.47;
    y=0.2;
    w=0.15000001;
};

```

```

        idc=5029;
        onInit=["'addMRK_UNIT_SCALE', ctrlText
_control_NAME != '', 0, 'sld', _control_UNIT_SCALE] call compile
preProcessFile '\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";

        onChanged=["_this,_control_PREVIEW_UNIT,_control_UNIT_POSITION,_c
ontrol_UNIT_DOWN] call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerScale.sqf""");
    };
    class SIZE
    {
        type="configEx";
        subtype="cfgTacticalSymbols.Size,name";
        description="Size Icon";
        w=0.25;
        idc=5030;
        onInit=["'addMRK_SIZE', ctrlText _control_NAME
!= '', 0, 'lb', _control_SIZE] call compile preProcessFile
'\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";
        onChanged="_this call
_func_COC_Marker_DlgChanged";
    };
    class SIZE_POSITION
    {
        type="number";
        subtype="range(-0.05, 0.05)";

        description="str_editor_edobj_param_placeholder";
        default="0";
        hidden=1;
        w=0.15000001;
        onChanged=["_this,_control_PREVIEW_SIZE, ""X""]
call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
    };
    class SIZE_DOWN
    {
        type="number";
        subtype="range(-0.05, 0.05)";

        description="str_editor_edobj_param_placeholder";
        default="0";
        hidden=1;
        x=0.36250001;
        y=0.31999999;
        w=0.15000001;
        onChanged=["_this,_control_PREVIEW_SIZE, ""Y""]
call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
    };
    class SIZE_SCALE
    {
        type="number";
        subtype="range(0.5, 1.5)";
        description="Scale";
        default="1";
        x=0.47;

```

```

        y=0.23;
        w=0.15000001;
        idc=5031;
        onInit=['addMRK_SIZE_SCALE', ctrlText
_control_NAME != '', 0, 'sld', _control_SIZE_SCALE] call compile
preProcessFile '\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';*

        onChanged='[_this, _control_PREVIEW_SIZE, _control_SIZE_POSITION, _c
ontrol_SIZE_DOWN] call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerScale.sqf""");
    };
    class MODIFIER
    {
        type="configEx";
        subtype="cfgTacticalSymbols.Modifiers,name";
        description="Modifier Icon";
        default="S0";
        w=0.25;
        idc=5032;
        onInit="_sel = _control_MODIFIER lbData
lbCurSel _control_MODIFIER; lbSort _control_MODIFIER; _was = _sel; for
'_i' from 0 to (lbSize _control_MODIFIER) -1 do {if (_control_MODIFIER
lbData _i == _sel) then {_was = _i; _control_MODIFIER lbSetCurSel
_i}}; ['addMRK_MODIFIER', ctrlText _control_NAME != '', 0, 'lb',
_control_MODIFIER, _was] call compile preProcessFile
'\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";
        onChanged="_this call
_func_COC_Marker_DlgChanged";
    };
    class MODIFIER_POSITION
    {
        type="number";
        subtype="range(-0.05, 0.05)";

        description="str_editor_edobj_param_placeholder";
        default="0";
        hidden=1;
        w=0.15000001;

        onChanged="[_this, _control_PREVIEW_MODIFIER, ""X"" ] call compile
preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
    };
    class MODIFIER_DOWN
    {
        type="number";
        subtype="range(-0.05, 0.05)";

        description="str_editor_edobj_param_placeholder";
        default="0";
        hidden=1;
        x=0.36250001;
        y=0.38;
        w=0.15000001;

        onChanged="[_this, _control_PREVIEW_MODIFIER, ""Y"" ] call compile

```

```

preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
};
class MODIFIER_SCALE
{
    type="number";
    subtype="range(0.5, 1.5)";
    description="Scale";
    default="1";
    x=0.47;
    y=0.259999999;
    w=0.15000001;
    idc=5033;
    onInit=["'addMRK_MODIFIER_SCALE', ctrlText
_control_NAME != '', 0, 'sld', _control_MODIFIER_SCALE] call compile
preProcessFile '\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";

    onChanged=["_this, _control_PREVIEW_MODIFIER, _control_MODIFIER_POS
ITION, _control_MODIFIER_DOWN] call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerScale.sqf""");
};
class PREVIEW_SIDE
{
    type="picture";
    description="Preview";
};
class PREVIEW_UNIT
{
    type="picture";

description="str_editor_edobj_param_placeholder";
    x=0.209999999;
    y=0.289999999;
    w=0.15000001;
    h=0.2;
};
class PREVIEW_SIZE
{
    type="picture";

description="str_editor_edobj_param_placeholder";
    x=0.209999999;
    y=0.289999999;
    w=0.15000001;
    h=0.2;
};
class PREVIEW_MODIFIER
{
    type="picture";

description="str_editor_edobj_param_placeholder";
    x=0.209999999;
    y=0.289999999;
    w=0.15000001;
    h=0.2;
};
};

```

```

        create[]=
        {
            "%VARIABLE_NAME =
            ([""%VARIABLE_NAME"", %NAME, %TEXT, %MARKER_TYPE, """", %COLOR, [%A,
            %B], %ANGLE, %POSITION, %CONDITION, %PREVIEW_SIDE, %PREVIEW_UNIT, %PREVIEW_SI
            ZE, %PREVIEW_MODIFIER, [%SIDE_POSITION, %SIDE_DOWN], %SIDE_SCALE, [%SIZE_POS
            ITION, %SIZE_DOWN], %SIZE_SCALE, [%UNIT_POSITION, %UNIT_DOWN], %UNIT_SCALE, [
            %MODIFIER_POSITION, %MODIFIER_DOWN], %MODIFIER_SCALE, %AUTOSCALE] +
            [%PARENT_UNIT]) call _func_COC_Marker_Tactical_Create;"
        };
        update[]=
        {
            "[%NAME, %TEXT, %MARKER_TYPE, """, %COLOR, [%A,
            %B], %ANGLE, %POSITION, %CONDITION, %PREVIEW_SIDE, %PREVIEW_UNIT, %PREVIEW_SI
            ZE, %PREVIEW_MODIFIER, [%SIDE_POSITION, %SIDE_DOWN], %SIDE_SCALE, [%SIZE_POS
            ITION, %SIZE_DOWN], %SIZE_SCALE, [%UNIT_POSITION, %UNIT_DOWN], %UNIT_SCALE, [
            %MODIFIER_POSITION, %MODIFIER_DOWN], %MODIFIER_SCALE, ""%PARENT"", ""%VARIA
            BLE_NAME"", %AUTOSCALE] call _func_COC_Marker_Tactical_Update;"
        };
        execDrawMap="false";
    };
};
class _trigger_0
{
    objectType="trigger";
    layer="classes";
    height=-1.1718751e-006;
    class Arguments
    {
        NAME="testTrigger";
        TEXT="testTrigger";
        POSITION="[4920.45166, 6668.51807, 641.04370]";
        ACTIVATION="WEST";
        ON_ACTIVATION="hint str 2";
        ON_DEACTIVATION="hint str 2";
        ADVANCED="";
        EFFRESOURCE="BIS";
        EFFOBJECT="Sphere";
        DISPLAY_NAME="testTrigger: testTrigger";
    };
};
class _object_0
{
    objectType="object";
    layer="classes";
    height=0.044618282;
    class Arguments
    {
        NAME="testObject";
        URN="testObject";
        SUB_TEAM="";
        SIDE="all";
        TYPE="Land_poleend";
        ADVANCED="";
        FUEL="1.00000";
        BG1="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        HEALTH="1.00000";
    };
};

```

```

        BG2="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        PROBOFPRES="1.00000";
        AZIMUT="0.00000";
        VECTORDIR="{0.00000,1.00000,-0.00398}";
        VECTORUP="{0.00000,0.00000,1.00000}";
        AMMO="1.00000";
        POSITION="{5017.24121, 6677.48438, 640.41913}";
        COLOR="{0.5,0.5,0}";
        DISPLAY_NAME="(testObject) testObject";
        DISPLAY_NAME_TREE="(testObject) testObject";
    };
};
class _ied_0
{
    objectType="ied";
    layer="classes";
    height=1.1914062e-006;
    class Arguments
    {
        NAME="testIED";
        POSITION="{5088.33984, 6675.08545, 639.89203}";
        TYPE="vbs2_ied_11";
        PREVIEW_PIC="\VBS2\weapons\ied\data\ico\preview_ied01_ca";
        AZIMUT="-0.00115";
        VECTORDIR="{ -0.00002, 0.99999, -0.00362}";
        VECTORUP="{0.00688, 0.00362, 0.99997}";
        CHARGE="vbs2_iedxhuge";
        EXPLOSION_TYPE="normal";
        BOOBY_TRAP="true";
        PROXIMITY="Proximity";
        PRESSURE_PLATE_WEIGHT="Everything";
        IS_HIDDEN="";
        ON_EXPLODE="hint str (1 + 2)";
        DISPLAY_NAME="(testIED) Mortars with Fuse";
    };
};
class _vehicle_0
{
    objectType="vehicle";
    layer="classes";
    height=9.6875001e-006;
    class Arguments
    {
        NAME="testVehicle";
        URN="testVehicle";
        URN_DIS="testVehicle";
        SUB_TEAM="";
        SIDE="all";
        FILTER_BY_FORCE="";
        FILTER="US";
        TYPE="VBS2_US_ARMY_M1A2_D_X";
        EDIT_UNIT="";
        NEW_UNIT="";
        DELETE_UNIT="";
        IS_EDIT="";
        ADVANCED="";
        BG1="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
    };
};

```

```

        FUEL="1.00000";
        BG2="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        HEALTH="1.00000";
        BG3="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        AMMO="1.00000";
        BG4="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        PROBOFPRES="1.00000";
        AZIMUT="-0.00115";
        VECTORDIR="[-0.00002,0.99999,-0.00398]";
        VECTORUP="[0.00543,0.00398,0.99998]";
        POSITION="[4919.25684, 6737.82324, 640.71583]";
        PARENT="_group_1";
        COLOR="[0.600000023841858,0.600000023841858,1]";
        DISPLAY_NAME="(testVehicle) testVehicle";
        DISPLAY_NAME_TREE="(testVehicle) testVehicle";
    };
};
class _unit_5
{
    objectType="unit";
    layer="classes";
    height=9.6875001e-006;
    class Arguments
    {
        DESCRIPTION="M1A2 Commander ";
        SIDE="WEST";
        TYPE="VBS2_US_ARMY_IFVCrew_W_M4";
        SPECIAL="CARGO";
        AZIMUT="-0.00115";
        POSITION="[4919.25684, 6737.82324, 640.71583]";
        COLOR="[0.600000023841858,0.600000023841858,1]";
        COMMANDER="_vehicle_0";
        DRIVER="";
        GUNNER="_vehicle_0";
        GUNNER_TURRET="[0,0]";
        LEADER="";
        DISPLAY_NAME="M1A2 Commander ";
        DISPLAY_NAME_TREE="M1A2 Commander ";
        PARENT="_group_1";
        PARENT_UNIT="";
    };
};
class _unit_6
{
    objectType="unit";
    layer="classes";
    height=9.6875001e-006;
    class Arguments
    {
        DESCRIPTION="M1A2 Driver ";
        SIDE="WEST";
        TYPE="VBS2_US_ARMY_IFVCrew_W_M4";
        SPECIAL="CARGO";
        AZIMUT="-0.00115";
        POSITION="[4919.25684, 6737.82324, 640.71583]";
        COLOR="[0.600000023841858,0.600000023841858,1]";
        COMMANDER="";
    };
};

```

```

        DRIVER="_vehicle_0";
        GUNNER="";
        LEADER="_unit_5";
        DISPLAY_NAME="M1A2 Driver ";
        DISPLAY_NAME_TREE="M1A2 Driver ";
        PARENT="_group_1";
        PARENT_UNIT="";
    };
};
class _unit_7
{
    objectType="unit";
    layer="classes";
    height=9.6875001e-006;
    class Arguments
    {
        DESCRIPTION="M1A2 Gunner ";
        SIDE="WEST";
        TYPE="VBS2_US_ARMY_IFVCrew_W_M4";
        SPECIAL="CARGO";
        AZIMUT="-0.00115";
        POSITION="[4919.25684, 6737.82324, 640.71583]";
        COLOR="[0.600000023841858,0.600000023841858,1]";
        COMMANDER="";
        DRIVER="";
        GUNNER="_vehicle_0";
        GUNNER_TURRET="[0]";
        LEADER="_unit_5";
        DISPLAY_NAME="M1A2 Gunner ";
        DISPLAY_NAME_TREE="M1A2 Gunner ";
        PARENT="_group_1";
        PARENT_UNIT="";
    };
};
class _unit_8
{
    objectType="unit";
    layer="classes";
    height=9.6875001e-006;
    class Arguments
    {
        DESCRIPTION="M1A2 Assistant Gunner ";
        SIDE="WEST";
        TYPE="VBS2_US_ARMY_IFVCrew_W_M4";
        SPECIAL="CARGO";
        AZIMUT="-0.00115";
        POSITION="[4919.25684, 6737.82324, 640.71583]";
        COLOR="[0.600000023841858,0.600000023841858,1]";
        COMMANDER="";
        DRIVER="";
        GUNNER="_vehicle_0";
        GUNNER_TURRET="[0,1]";
        LEADER="_unit_5";
        DISPLAY_NAME="M1A2 Assistant Gunner ";
        DISPLAY_NAME_TREE="M1A2 Assistant Gunner ";
        PARENT="_group_1";
        PARENT_UNIT="";
    };
};

```



```

    };
};
class _marker_1
{
    objectType="marker";
    layer="classes";
    height=9.6875001e-006;
    class Arguments
    {
        POSITION="[4919.25684, 6737.82324, 640.71583]*";
        NAME="m2*";
        LASTNAME="m2*";
        MARKER_TYPE="TacticalMarker";
        COLOR="ColorBlue";
        PARENT="_group_1";
        PARENT_UNIT="_unit_5";
        DISPLAY_NAME="Marker m2*";
        SIDE="F1*";
        UNIT="I9*";
        SIZE="M2*";

        PREVIEW_SIDE="\vbs2\ui\tacticmarkers\data\Frames\Friend_Units";
        PREVIEW_UNIT="\vbs2\ui\tacticmarkers\data\Icons\Armour";

        PREVIEW_SIZE="\vbs2\ui\tacticmarkers\data\Modifiers\S_Team";

        PREVIEW_MODIFIER="\vbs2\ui\tacticmarkers\data\SubRoles\Blanc";
    };
    class DynamicArguments
    {
        name="TacticalMarker";
        class Params
        {
            class CUSTOM_MARKER
            {
                type="config";
                subtype="CfgMarkers,name";
                description="Custom Marker";
                w=0.34;
                idc=5001;
                onChanged="[_map] call compile preprocessFile
""\vbs2\editor\Data\Scripts\marker\customMarkers\markerUIcustomLBChange
d.sqf""";
            };
            class CUSTOM_DELETE
            {
                type="button";
                x=0.56;
                y=0.11;
                w=0.1;
                onInit="_control_CUSTOM_DELETE ctrlSetText
(localize ""STR_CA_DELETE""); _control_CUSTOM_DELETE
ctrlSetEventHandler [""ButtonClick"",preProcessFile
""\vbs2\editor\Data\Scripts\marker\customMarkers\markerUIcustomDelete.s
qf""]";
            };
            class CUSTOM_NAME

```

```

        type="text";
        description="Save As";
        w=0.34;
        idc=5002;
    };
class CUSTOM_SAVE
{
    type="button";
    x=0.56;
    y=0.14;
    w=0.1;
    onInit="_control_CUSTOM_SAVE ctrlSetText
(localize ""STR_DISP_ARCMAP_SAVE ""); _control_CUSTOM_SAVE
ctrlSetEventHandler ["ButtonClick",preProcessFile
""\vbs2\editor\Data\Scripts\marker\customMarkers\markerUIcustomSave.sqf
""]";
};
class SIDE
{
    type="configEx";
    subtype="cfgTacticalSymbols.Sides,name";
    description="Side Icon";
    default="f1";
    w=0.25;
    idc=5026;
    onInit="_sel = _control_SIDE lbData lbCurSel
_control_SIDE; lbSort _control_SIDE; _was = _sel; for 'i' from 0 to
(lbSize _control_SIDE) -1 do {if (_control_SIDE lbData_i == _sel) then
{_was = i;_control_SIDE lbSetCurSel_i}};['addMRK_SIDE', ctrlText
_control_NAME != '', 0, 'lb', _control_SIDE,_was] call compile
preProcessFile '\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";
    onChanged="_this call
_func_COC_Marker_DlgChanged";
};
class SIDE_POSITION
{
    type="number";
    subtype="range(-0.05, 0.05)";
    description="Side Icon";
    default="0";
    hidden=1;
    w=0.15000001;
    onChanged="[_this,_control_PREVIEW_SIDE, ""X""]
call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
};
class SIDE_DOWN
{
    type="number";
    subtype="range(-0.05, 0.05)";

description="str_editor_edobj_param_placeholder";
    default="0";
    hidden=1;
    x=0.36250001;
    w=0.15000001;
};

```

```

        onChanged="[_this,_control_PREVIEW_SIDE,""Y""]
call compile preprocessfile
(""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf"");
    };
    class SIDE_SCALE
    {
        type="number";
        subtype="range(0.5, 1.5)";
        description="Scale";
        default="1";
        x=0.47;
        y=0.17;
        w=0.15000001;
        idc=5027;

        onChanged="[_this,_control_PREVIEW_SIDE,_control_SIDE_POSITION,_c
ontrol_SIDE_DOWN] call compile preprocessfile
(""\vbs2\editor\Data\Scripts\marker\layerScale.sqf"");
    };
    class UNIT
    {
        type="configEx";
        subtype="cfgTacticalSymbols.Units,name";
        description="Unit Icon";
        default="I1";
        w=0.25;
        idc=5028;
        onInit="_sel = _control_UNIT lbData lbCurSel
_control_UNIT; lbSort _control_UNIT; _was = _sel; for 'i' from 0 to
(lbSize _control_UNIT) -1 do {if (_control_UNIT lbData _i == _sel) then
{_was = _i;_control_UNIT lbSetCurSel _i}};['addMRK_UNIT', ctrlText
_control_NAME != '', 0, 'lb', _control_UNIT,_was] call compile
preProcessFile '\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";
        onChanged="_this call
_func_COC_Marker_DlgChanged";
    };
    class UNIT_POSITION
    {
        type="number";
        subtype="range(-0.05, 0.05)";

        description="str_editor_edobj_param_placeholder";
        default="0";
        hidden=1;
        w=0.15000001;
        onChanged="[_this,_control_PREVIEW_UNIT,""X""]
call compile preprocessfile
(""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf"");
    };
    class UNIT_DOWN
    {
        type="number";
        subtype="range(-0.05, 0.05)";

        description="str_editor_edobj_param_placeholder";
        default="0";
        hidden=1;

```

```

        x=0.36250001;
        y=0.25999999;
        w=0.15000001;
        onChanged="[_this,_control_PREVIEW_UNIT,""Y""]
call compile preprocessfile
(""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf"");
};
class UNIT_SCALE
{
    type="number";
    subtype="range(0.5, 1.5)";
    description="Scale";
    default="1";
    x=0.47;
    y=0.2;
    w=0.15000001;
    idc=5029;
    onInit="['addMRK_UNIT_SCALE', ctrlText
_control_NAME != '', 0, 'sld', _control_UNIT_SCALE] call compile
preProcessFile '\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";

    onChanged="[_this,_control_PREVIEW_UNIT,_control_UNIT_POSITION,_c
ontrol_UNIT_DOWN] call compile preprocessfile
(""\vbs2\editor\Data\Scripts\marker\layerScale.sqf"");
};
class SIZE
(
    type="configEx";
    subtype="cfgTacticalSymbols.Size,name";
    description="Size Icon";
    w=0.25;
    idc=5030;
    onInit="['addMRK_SIZE', ctrlText _control_NAME
!= '', 0, 'lb', _control_SIZE] call compile preProcessFile
'\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";
    onChanged="_this call
_func_COC_Marker_DlgChanged";
);
class SIZE_POSITION
{
    type="number";
    subtype="range(-0.05, 0.05)";

    description="str_editor_edobj_param_placeholder";
    default="0";
    hidden=1;
    w=0.15000001;
    onChanged="[_this,_control_PREVIEW_SIZE,""X""]
call compile preprocessfile
(""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf"");
};
class SIZE_DOWN
{
    type="number";
    subtype="range(-0.05, 0.05)";

    description="str_editor_edobj_param_placeholder";

```

```

        default="0";
        hidden=1;
        x=0.36250001;
        y=0.31999999;
        w=0.15000001;
        onChanged="[_this,_control_PREVIEW_SIZE,""Y""]
call compile preprocessfile
(""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf"");
    };
    class SIZE_SCALE
    {
        type="number";
        subtype="range(0.5, 1.5)";
        description="Scale";
        default="1";
        x=0.47;
        y=0.23;
        w=0.15000001;
        idc=5031;
        onInit="['addMRK_SIZE_SCALE', ctrlText
_control_NAME != '', 0, 'sld', _control_SIZE_SCALE] call compile
preProcessFile '\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf'";

        onChanged="[_this,_control_PREVIEW_SIZE,_control_SIZE_POSITION,_c
ontrol_SIZE_DOWN] call compile preprocessfile
(""\vbs2\editor\Data\Scripts\marker\layerScale.sqf"");
    };
    class MODIFIER
    {
        type="configEx";
        subtype="cfgTacticalSymbols.Modifiers,name";
        description="Modifier Icon";
        default="S0";
        w=0.25;
        idc=5032;
        onInit="_sel = _control_MODIFIER lbData
lbCurSel _control_MODIFIER; lbSort _control_MODIFIER; _was = _sel; for
'_i' from 0 to (lbSize _control_MODIFIER) -1 do {if (_control_MODIFIER
lbData _i == _sel) then {_was = _i; _control_MODIFIER lbSetCurSel
_i}};['addMRK_MODIFIER', ctrlText _control_NAME != '', 0, 'lb',
_control_MODIFIER,_was] call compile preProcessFile
'\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf'";
        onChanged="_this call
_func_COC_Marker_DlgChanged";
    };
    class MODIFIER_POSITION
    {
        type="number";
        subtype="range(-0.05, 0.05)";

        description="str_editor_edobj_param_placeholder";
        default="0";
        hidden=1;
        w=0.15000001;

        onChanged="[_this,_control_PREVIEW_MODIFIER,""X""] call compile

```

```

preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
};
class MODIFIER_DOWN
{
    type="number";
    subtype="range(-0.05, 0.05)";

    description="str_editor_edobj_param_placeholder";
    default="0";
    hidden=1;
    x=0.36250001;
    y=0.38;
    w=0.15000001;

    onChanged="[_this,_control_PREVIEW_MODIFIER,""Y""] call compile
preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerPosition.sqf""");
};
class MODIFIER_SCALE
{
    type="number";
    subtype="range(0.5, 1.5)";
    description="Scale";
    default="1";
    x=0.47;
    y=0.25999999;
    w=0.15000001;
    idc=5033;
    onInit="{ 'addMRK_MODIFIER_SCALE', ctrlText
_control_NAME != '', 0, 'sld', _control_MODIFIER_SCALE] call compile
preProcessFile '\vbs2\editor\Data\Scripts\UI\saveLastSelection.sqf';";

    onChanged="[_this,_control_PREVIEW_MODIFIER,_control_MODIFIER_POS
ITION,_control_MODIFIER_DOWN] call compile preprocessfile
("""\vbs2\editor\Data\Scripts\marker\layerScale.sqf""");
};
class PREVIEW_SIDE
{
    type="picture";
    description="Preview";
};
class PREVIEW_UNIT
{
    type="picture";

description="str_editor_edobj_param_placeholder";
    x=0.20999999;
    y=0.28999999;
    w=0.15000001;
    h=0.2;
};
class PREVIEW_SIZE
{
    type="picture";

description="str_editor_edobj_param_placeholder";

```

```

        x=0.20999999;
        y=0.28999999;
        w=0.15000001;
        h=0.2;
    };
    class PREVIEW_MODIFIER
    {
        type="picture";

description="str_editor_edobj_param_placeholder";
        x=0.20999999;
        y=0.28999999;
        w=0.15000001;
        h=0.2;
    };
};
create[]=
{
    "%VARIABLE_NAME =
([""%VARIABLE_NAME"", %NAME, %TEXT, %MARKER_TYPE, """", %COLOR, [%A,
%B], %ANGLE, %POSITION, %CONDITION, %PREVIEW_SIDE, %PREVIEW_UNIT, %PREVIEW_SI
ZE, %PREVIEW_MODIFIER, [%SIDE_POSITION, %SIDE_DOWN], %SIDE_SCALE, [%SIZE_POS
ITION, %SIZE_DOWN], %SIZE_SCALE, [%UNIT_POSITION, %UNIT_DOWN], %UNIT_SCALE, [
%MODIFIER_POSITION, %MODIFIER_DOWN], %MODIFIER_SCALE, %AUTOSCALE] +
[%PARENT_UNIT]) call _func_COC_Marker_Tactical_Create;"
};
update[]=
{
    "[%NAME, %TEXT, %MARKER_TYPE, """", %COLOR, [%A,
%B], %ANGLE, %POSITION, %CONDITION, %PREVIEW_SIDE, %PREVIEW_UNIT, %PREVIEW_SI
ZE, %PREVIEW_MODIFIER, [%SIDE_POSITION, %SIDE_DOWN], %SIDE_SCALE, [%SIZE_POS
ITION, %SIZE_DOWN], %SIZE_SCALE, [%UNIT_POSITION, %UNIT_DOWN], %UNIT_SCALE, [
%MODIFIER_POSITION, %MODIFIER_DOWN], %MODIFIER_SCALE, ""%PARENT"", ""%VARIA
BLE_NAME"", %AUTOSCALE] call _func_COC_Marker_Tactical_Update;"
};
    execDrawMap="false";
};
};
class _intel_1
{
    objectType="intel";
    height=-640.72064;
    class Arguments
    {
        POSITION="{4912.23145, 6759.37500, 0.00000}";
        BG1="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        OVERCAST="0.80000";
        BG2="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        OVERCAST_WANTED="0.70000";
        BG3="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        FOG="0.00000";
        BG4="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        FOG_WANTED="0.40000";
        BG5="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        RAIN="0.40000";
        BG6="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        SNOW="0.00000";
    }
}

```

```

        BG7="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        WIND="7.40000";
        WIND_DIR="90";
        SEP1="";
        SETTIME="";
        YEAR="2013";
        MONTH="9";
        DAY="18";
        SEP5="";
        COORD_TYPE="MGRS";
        SEP4="";
        SETORIGIN="";
        RESETDECLINATION="";
        RESETCOORD="";
        EASTING="460257";
        NORTHING="3873755";
        ZONE="11";
        SEP6="";
        SEP2="";
        CIVIENEMY="East";
        SEP3="";
        DISPLAY_NAME="Intel";
        FOW_WEST="FOW_VISIBLE";
        FOW_EAST="FOW_VISIBLE";
    };
};
class _vehicle_2
{
    objectType="vehicle";
    height=8.7109374e-006;
    class Arguments
    {
        SUB_TEAM="";
        SIDE="all";
        FILTER_BY_FORCE="";
        FILTER="US";
        TYPE="VBS2_US_ARMY_M1A2_D_X";
        EDIT_UNIT="";
        NEW_UNIT="";
        DELETE_UNIT="";
        IS_EDIT="";
        HAS_CREW="false";
        ADVANCED="";
        BG1="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        FUEL="1.00000";
        BG2="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        HEALTH="1.00000";
        BG3="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        AMMO="1.00000";
        BG4="\vbs2\ui\Data\NewEditor\slider_back_rad.paa";
        PROBOFPRES="1.00000";
        AZIMUT="0.00057";
        VECTORDIR="[0.00001,1.00000,0.00088]";
        VECTORUP="[0.00964,-0.00088,0.99995]";
        POSITION="[4853.56690, 6719.75049, 641.14808]";
        COLOR="[0.5,0.5,0]";
        DISPLAY_NAME="M1A2";
    }
}

```



```

        DISPLAY_NAME_TREE="M1A2";
    };
class _ied_1
{
    objectType="ied";
    height=-4.0234377e-006;
    class Arguments
    {
        POSITION="[5599.10596, 7190.05762, 634.73962]";
        TYPE="vbs2_ied_11";
        PREVIEW_PIC="\VBS2\weapons\ied\data\ico\preview_ied01_ca";
        AZIMUT="-0.00115";
        VECTORDIR="[-0.00002,1.00000,-0.00264]";
        VECTORUP="[0.00829,0.00264,0.99996]";
        CHARGE="vbs2_iedxhuge";
        EXPLOSION_TYPE="normal";
        BOOBY_TRAP="true";
        PROXIMITY="Proximity";
        PRESSURE_PLATE_WEIGHT="Everything";
        IS_HIDDEN="";
        DISPLAY_NAME="Mortars with Fuse";
    };
};
class _postfix_0
{
    objectType="postfix";
    class Arguments
    {
    };
};
};

```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. VBS2 OBJECT CLASSES

The following tables identify several of the VBS2 object classes (listed alphabetically by object class name) and provide their descriptions (class attributes and arguments) as found in the notional scenario data file created for the Phase 1 effort. These individual tables are used in a spreadsheet to identify common arguments across the various classes.

Table 1. VBS2 Empty Vehicle Class

<i>Attribute-List</i>	Vehicle (empty)_Class
NAME	X
URN	X
URN_DIS	X
VECTORDIR	X
VECTORUP	X
SIDE	X
FILTER_BY_FORCE	X
FILTER	X
TYPE_DELETE	X
EDIT_UNIT	X
NEW_UNIT	X
DELETE_UNIT	X
IS_EDIT	X
HAS_CREW	X
ADVANCED	X
BACKGROUNDS(BG)	X (x4)
HEALTH	X
AMMO	X
FUEL	X
PROBOFPRES	X
AZIMUT	X
POSITION	X
COLOR	X
SUB_TEAM	X
DISPLAY_NAME_TREE	X
PARENT_UNIT	X

Table 2. VBS2 Group Class

<i>Attribute-List</i>	Group_Class
NAME	X
SIDE	X
SELECTED_SIDE	X
FILTER	X
FILTER2	X
CATEGORY	X
TYPE_GROUP	X
GROUP_MEMBERS_NAME	X
GROUPFRAME	X
GROUP_MEMBERS2	X
PREVIEW_PIC2	X
CREATE_GROUP	X
SIDE_NEW	X
CATEGORY_NEW	X
CATEGORY_DELETE	X
TYPE_NEW	X
TYPE_DELETE	X
GROUP_MEMBERS	X
AVAILABLE_MEMBERS	X
MEMBER_UP	X
MEMBER_DOWN	X
ADD_MEMBER	X
REMOVE_MEMBER	X
COPY_MEMBER	X
EDIT_UNIT1	X
NEW_UNIT2	X
IS_EDIT	X
POSITION	X
DISPLAY_NAME	X
DISPLAY_NAME_TREE	X

Table 3. VBS2 IED Class

<i>Attribute-List</i>	IED_Class
NAME	X
VECTORDIR	X
VECTORUP	X
CHARGE	X
ON_EXPLODE	X
EXPLOSION_TYPE	X
BOOBY_TRAP	X
PROXIMITY	X
PRESSURE_PLATE_WEIGHT	X
IS_HIDDEN	X
TYPE	X
AZIMUT	X
POSITION	X
PREVIEW_PIC	X
DISPLAY_NAME	X

Table 4. VBS2 Intel Class

<i>Attribute-List</i>	<i>Intel_Class</i>
POSITION	X
DISPLAY_NAME	X
OVERCAST	X
OVERCAST_WANTED	X
FOG	X
FOG_WANTED	X
RAIN	X
SNOW	X
WIND	X
WIND_DIR	X
SEP	X (x6)
SETTIME	X
YEAR	X
MONTH	X
DAY	X
COORD_TYPE	X
SETORIGIN	X
RESETDECLINATION	X
RESETCOORD	X
EASTING	X
NORTHING	X
ZONE	X
CIVIENEMY	X
FOW_WEST	X
FOW_EAST	X

Table 5. VBS2 Marker Class

<i>Attribute-List</i>	Marker_Class
NAME	X
LASTNAME	X
MARKER_TYPE	X
UNIT	X
SIZE	X
PREVIEW_SIDE	X
PREVIEW_UNIT	X
PREVIEW_SIZE	X
PREVIEW_MODIFIER	X
SIDE	X
POSITION	X
COLOR	X
DISPLAY_NAME	X
PARENT	X
PARENT_UNIT	X

Table 6. VBS2 Object Class

<i>Attribute-List</i>	Object_Class
NAME	X
URN	X
VECTORDIR	X
VECTORUP	X
SIDE	X
TYPE	X
INIT	X
ADVANCED	X
BACKGROUNDS(BG)	X (x2)
HEALTH	X
AMMO	X
FUEL	X
PROBOFPRES	X
AZIMUT	X
POSITION	X
COLOR	X
SUB_TEAM	X
DISPLAY_NAME	X
DISPLAY_NAME_TREE	X

Table 7. VBS2 Trigger Class

<i>Attribute-List</i>	Trigger_Class
NAME	X
TEXT	X
ACTIVATION	X
ON_ACTIVATION	X
ON_DEACTIVATION	X
ADVANCED	X
POSITION	X
EFFRESOURCE	X
EFFOBJECT	X
DISPLAY_NAME	X

Table 8. VBS2 Unit Class

<i>Attribute-List</i>		Unit_Class
NAME		X
URN		X
DESCRIPTION		X
SIDE		X
FILTER_BY_FORCE		X
FILTER		X
TYPE		X
BASE_TYPE		X
EDIT_UNIT		X
NEW_UNIT		X
DELETE_UNIT		X
IS_EDIT		X
INIT		X
COMBATMODE		X
BEHAVIOUR		X
PLAYER		X
ADVANCED		X
BACKGROUNDS(BG)		X (x14)
HEALTH		X
AMMO		X
PROBOFPRES		X
TRAINING		X
EXPERIENCE		X
ENDURANCE		X
LEADERSHIP		X
FLEECHANCE		X
HEIGHT		X
BMI		X
AZIMUT		X
POSITION		X
COLOR		X
SUB_TEAM		X
DISPLAY_NAME		X
DISPLAY_NAME_TREE		X
PARENT_UNIT		X
LOADOUT		X

Table 9. VBS2 Waypoint Class

<i>Attribute-List</i>	
NAME	
URN	
DESCRIPTION	
SIDE	
FILTER_BY_FORCE	
FILTER	
TYPE	
BASE_TYPE	
EDIT_UNIT	
NEW_UNIT	

APPENDIX C. PRELIMINARY XML SCHEMA DESCRIBING A REPRESENTATION OF VBS2 SCENARIO OBJECT CLASSES

The following is an initial XML schema describing a structure and content for representing VBS2 data from a .biedi file. This schema should be considered as a skeletal structure at this time. Each of the VBS2 object classes needs to be fully designed based on current knowledge of the scenario content and on greater study and understanding of VBS2 scenario data.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns="http://www.nps.edu/XML/schemas/scenarios/VBS2"
targetNamespace="http://www.nps.edu/XML/schemas/scenarios/VBS2" elementFormDefault="unqualified">
  <!-- *****ROOT ELEMENT DECLARATION***** -->
  <xs:element name="VBS2Scenario" type="VBS2ScenarioType">
    <xs:annotation>
      <xs:documentation>Root element of the XML representation of VBS2 scenario
data.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:complexType name="VBS2ScenarioType">
    <xs:annotation>
      <xs:documentation>Declaration of the complex data structure of the root element
of the XML representation of VBS2 scenario data.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <!-- TODO: expecting to add other components of VBS2 scenario data into this
element structure -->
      <xs:element name="Classes" type="ClassesType"/>
    </xs:sequence>
  </xs:complexType>
  <!-- *****VBS2 CLASS DATA STRUCTURE DECLARATIONS***** -->
  <xs:complexType name="ClassesType">
    <xs:annotation>
      <xs:documentation>Declaration of the complex data structure for object classes
that can be used in a scenario (e.g., group, unit, vehicle, waypoint) in version 2.0 of
VBS.</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="PrefixClass" type="PrefixClassType"/>
      <xs:choice minOccurs="1" maxOccurs="unbounded">
        <xs:element name="ArtilleryClass" type="ArtilleryClassType"/>
        <xs:element name="ArtilleryStrikeClass"
type="ArtilleryStrikeClassType"/>
        <xs:element name="BoomerangClass" type="BoomerangClassType"/>
        <xs:element name="CameraClass" type="CameraClassType"/>
        <xs:element name="ChemLightClass" type="ChemLightClassType"/>
        <xs:element name="ControlLinkClass" type="ControlLinkClassType"/>
        <xs:element name="CREWLinkClass" type="CREWLinkClassType"/>
        <xs:element name="EmptyVehicleClass"
type="EmptyVehicleClassType"/>
        <xs:element name="FleePointClass" type="FleePointClassType"/>
        <xs:element name="GlintObjectClass" type="GlintObjectClassType"/>
        <xs:element name="GroupClass" type="GroupClassType"/>
      </xs:choice>
    </xs:sequence>
  </xs:complexType>

```

```

<xs:element name="IEDClass" type="IEDClassType"/>
<xs:element name="IntelClass" type="IntelClassType"/>
<xs:element name="LightBeaconClass"
type="LightBeaconClassType"/>
<xs:element name="LightSourceClass" type="LightSourceClassType"/>
<xs:element name="LookAtClass" type="LookAtClassType"/>
<xs:element name="MarkerClass" type="MarkerClassType"/>
<xs:element name="MeasureDistanceClass"
type="MeasureDistanceClassType"/>
<xs:element name="MineClass" type="MineClassType"/>
<xs:element name="ObjectClass" type="ObjectClassType"/>
<xs:element name="RecordClass" type="RecordClassType"/>
<xs:element name="SandstormClass" type="SandstormClassType"/>
<xs:element name="ScriptClass" type="ScriptClassType"/>
<xs:element name="SoundSourceClass"
type="SoundSourceClassType"/>
<xs:element name="SurrenderClass" type="SurrenderClassType"/>
<xs:element name="TriggerClass" type="TriggerClassType"/>
<xs:element name="UnitClass" type="UnitClassType"/>
<xs:element name="VBS2FiresGunlineClass"
type="VBS2FiresGunlineClassType"/>
<xs:element name="VBS2StrikeAircraftClass"
type="VBS2StrikeAircraftClassType"/>
<xs:element name="VehicleClass" type="VehicleClassType"/>
<xs:element name="VerbalCommandClass"
type="VerbalCommandClassType"/>
<xs:element name="WaypointClass" type="WaypointClassType"/>
</xs:choice>
<xs:element name="PostfixClass" type="PostfixClassType"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ArtilleryClassType"/>
<xs:complexType name="ArtilleryStrikeClassType"/>
<xs:complexType name="BoomerangClassType"/>
<xs:complexType name="CameraClassType"/>
<xs:complexType name="ChemLightClassType"/>
<xs:complexType name="ControlLinkClassType"/>
<xs:complexType name="CREWLinkClassType"/>
<xs:complexType name="EmptyVehicleClassType"/>
<xs:complexType name="FleePointClassType"/>
<xs:complexType name="GlintObjectClassType"/>
<xs:complexType name="GroupClassType"/>
<xs:complexType name="IEDClassType"/>
<xs:complexType name="IntelClassType"/>
<xs:complexType name="LightBeaconClassType"/>
<xs:complexType name="LightSourceClassType"/>
<xs:complexType name="LookAtClassType"/>
<xs:complexType name="MeasureDistanceClassType"/>
<xs:complexType name="MineClassType"/>
<xs:complexType name="MarkerClassType"/>
<xs:complexType name="ObjectClassType"/>
<xs:complexType name="PostfixClassType">
<xs:attributeGroup ref="commonClassAttributes"/>
</xs:complexType>
<xs:complexType name="PrefixClassType">
<xs:sequence>
<xs:element name="Version" type="xs:string"/>
<xs:element name="AddOns" type="AddOnsType"/>
<xs:element name="Layers" type="LayersType"/>
</xs:sequence>
<xs:attributeGroup ref="commonClassAttributes"/>
</xs:complexType>

```

```

<xs:complexType name="RecordClassType"/>
<xs:complexType name="SandstormClassType"/>
<xs:complexType name="ScriptClassType"/>
<xs:complexType name="SoundSourceClassType"/>
<xs:complexType name="SurrenderClassType"/>
<xs:complexType name="TriggerClassType"/>
<xs:complexType name="UnitClassType">
  <xs:sequence>
    <xs:element name="Name" type="xs:string"/>
    <xs:element name="URN" type="xs:string"/>
    <xs:element name="Description" type="xs:string"/>
    <xs:element name="Side" type="xs:string"/>
    <xs:element name="FilterByForce" type="xs:string"/>
    <xs:element name="Filter" type="xs:string"/>
    <xs:element name="Type" type="xs:string"/>
    <xs:element name="Base Type" type="xs:string"/>
    <xs:element name="EditUnit" type="xs:string"/>
    <xs:element name="NewUnit" type="xs:string"/>
    <xs:element name="DeleteUnit" type="xs:string"/>
    <xs:element name="IsEdit" type="xs:string"/>
    <xs:element name="Init" type="xs:string" minOccurs="0"/>
    <xs:element name="CombatMode" type="xs:string"/>
    <xs:element name="Behaviour" type="xs:string"/>
    <xs:element name="Player" type="xs:boolean"/>
    <xs:element name="Advanced" type="xs:string"/>
    <xs:element name="Backgrounds" type="BackgroundsType" minOccurs="0"/>
    <xs:element name="Health" type="xs:decimal"/>
    <xs:element name="Ammo" type="xs:decimal"/>
    <xs:element name="ProbabilityOfPresence" type="xs:decimal"/>
    <xs:element name="Training" type="xs:decimal"/>
    <xs:element name="Experience" type="xs:decimal"/>
    <xs:element name="Endurance" type="xs:decimal"/>
    <xs:element name="Leadership" type="xs:decimal"/>
    <xs:element name="FleeChance" type="xs:decimal"/>
    <xs:element name="Height" type="xs:decimal"/>
    <xs:element name="BMI" type="xs:decimal"/>
    <xs:element name="Azimuth" type="xs:decimal"/>
    <xs:element name="Position" type="PositionType"/>
    <xs:element name="Color" type="ColorType"/>
    <xs:element name="SubTeam" type="xs:string"/>
    <xs:element name="DisplayName" type="xs:string"/>
    <xs:element name="DisplayNameTree" type="xs:string"/>
    <xs:element name="ParentUnit" type="xs:string"/>
    <xs:element name="LoadOut" type="LoadOutType" minOccurs="0"/>
  </xs:sequence>
  <xs:attributeGroup ref="commonClassAttributes"/>
</xs:complexType>
<xs:complexType name="VBS2FiresGunlineClassType"/>
<xs:complexType name="VBS2StrikeAircraftClassType"/>
<xs:complexType name="VehicleClassType"/>
<xs:complexType name="VerbalCommandClassType"/>
<xs:complexType name="WaypointClassType"/>
<!-- ***** COMPLEX TYPE DECLARATIONS FOR SUPPORTING ELEMENTS ***** -->
<xs:complexType name="AddOnsType"/>
<xs:complexType name="BackgroundsType">
  <xs:sequence>
    <xs:element name="Background" type="BackgroundType"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="BackgroundType">
  <xs:attribute name="id">

```

```

        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:pattern value="BG\d+"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="file" type="xs:string"/>
</xs:complexType>
<xs:complexType name="ColorType">
    <xs:attribute name="hue" type="xs:float" use="required"/>
    <xs:attribute name="saturation" type="xs:decimal" use="required"/>
    <xs:attribute name="lightness" type="xs:decimal" use="required"/>
</xs:complexType>
<xs:complexType name="LayersType"/>
<xs:complexType name="LoadOutType"/>
<xs:complexType name="PositionType">
    <xs:attribute name="posZ" type="xs:decimal" use="required"/>
    <xs:attribute name="posY" type="xs:decimal" use="required"/>
    <xs:attribute name="posX" type="xs:decimal" use="required"/>
</xs:complexType>
<!-- *****ATTRIBUTE AND ATTRIBUTE GROUP DECLARATIONS***** -->
<xs:attribute name="name" type="xs:string"/>
<xs:attribute name="objectType">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:enumeration value="unit"/>
            <xs:enumeration value="postfix"/>
            <xs:enumeration value="prefix"/>
        </xs:restriction>
    </xs:simpleType>
</xs:attribute>
<xs:attributeGroup name="commonClassAttributes">
    <xs:attribute ref="objectType" use="required"/>
    <xs:attribute ref="name" use="required"/>
    <xs:attribute name="height" type="xs:float" use="optional"/>
    <xs:attribute name="layer" type="xs:string" use="optional"/>
</xs:attributeGroup>
<!-- *****SIMPLE TYPE DECLARATIONS***** -->
<xs:simpleType name="UnitRangeType">
    <xs:restriction base="xs:decimal">
        <xs:minInclusive value="0.0"/>
        <xs:maxInclusive value="1.0"/>
    </xs:restriction>
</xs:simpleType>
</xs:schema>

```

APPENDIX D. XSLT FOR TRANSFORMING AN XML REPRESENTATION OF VBS2 UNIT OBJECT CLASS TO VBS2 SCENARIO INPUT DATA FORMAT

The following XSLT translates an XML representation of the VBS2 unit object class into the VBS2 input data format, as discussed in Chapter V. This is a partial implementation at this time, pending greater familiarity with variations in VBS2 data structures and completion of translations for other object classes.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="2.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:fo="http://www.w3.org/1999/XSL/Format" xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:fn="http://www.w3.org/2005/xpath-functions"
xmlns:vbs="http://www.nps.edu/XML/schemas/scenarios/VBS2">
  <xsl:output method="text"/>
  <xsl:template match="/">
    <xsl:apply-templates select="vbs:VBS2Scenario/Classes/UnitClass"/>
  </xsl:template>
  <xsl:template match="UnitClass">
    <xsl:text>class </xsl:text>
    <xsl:value-of select="@vbs:name"/>
    {
      objectType="<xsl:value-of select="@vbs:objectType"/>";
      layer="<xsl:value-of select="@layer"/>";
      height="<xsl:value-of select="@height"/>";
      class Arguments {
        NAME="<xsl:value-of select="Name"/>";
        URN="<xsl:value-of select="URN"/>";
        DESCRIPTION="<xsl:value-of select="Description"/>";
        SIDE="<xsl:value-of select="Side"/>";
        FILTER_BY_FORCE="<xsl:value-of select="FilterByForce"/>";
        FILTER="<xsl:value-of select="Filter"/>";
        TYPE="<xsl:value-of select="Type"/>";
        BASE_TYPE="<xsl:value-of select="BaseType"/>";
        EDIT_UNIT="<xsl:value-of select="EditUnit"/>";
        NEW_UNIT="<xsl:value-of select="NewUnit"/>";
        DELETE_UNIT="<xsl:value-of select="DeleteUnit"/>";
        IS_EDIT="<xsl:value-of select="IsEdit"/>";
        <xsl:if test="Init!=">
          INIT="<xsl:value-of select="Init"/>";
        </xsl:if>
        RANK="<xsl:value-of select="Rank"/>";
        COMBATMODE="<xsl:value-of select="CombatMode"/>";
        BEHAVIOUR="<xsl:value-of select="Behaviour"/>";
        PLAYER="<xsl:value-of select="Player"/>";
        ADVANCED="<xsl:value-of select="Advanced"/>";
        <xsl:apply-templates select="Backgrounds"/>
        HEALTH="<xsl:value-of select="Health"/>";
        AMMO="<xsl:value-of select="Ammo"/>";
        PROBOFPRES="<xsl:value-of select="ProbabilityOfPresence"/>";
        TRAINING="<xsl:value-of select="Training"/>";
        EXPERIENCE="<xsl:value-of select="Experience"/>";
        ENDURANCE="<xsl:value-of select="Endurance"/>";
        LEADERSHIP="<xsl:value-of select="Leadership"/>";
        FLEECHANCE="<xsl:value-of select="FleeChance"/>";
        HEIGHT="<xsl:value-of select="Height"/>";
      }
    }
  }

```

```

        BMI="<xsl:value-of select="BMI"/>";
        AZIMUT="<xsl:value-of select="Azimuth"/>";
        POSITION="[<xsl:value-of select="Position/@posX"/>, <xsl:value-of
select="Position/@posY"/>, <xsl:value-of select="Position/@posZ"/>]";
        COLOR="[<xsl:value-of select="Color/@hue"/>, <xsl:value-of
select="Color/@saturation"/>, <xsl:value-of select="Color/@lightness"/>]";
        SUB_TEAM="<xsl:value-of select="SubTeam"/>";
        DISPLAY_NAME="<xsl:value-of select="DisplayName"/>";
        DISPLAY_NAME_TREE="<xsl:value-of select="DisplayNameTree"/>";
        PARENT_UNIT="<xsl:apply-templates select="ParentUnit"/>";
        LOADOUT="<xsl:value-of select="LoadOut"/>";
    };
</xsl:template>
<xsl:template match="Backgrounds">
    <xsl:for-each select="Background">
        <xsl:value-of select="@id"/>
        <xsl:text>=</xsl:text>
        <xsl:value-of select="@file"/>
        <xsl:text>; </xsl:text>
    </xsl:for-each>
</xsl:template>
</xsl:stylesheet>

```


LIST OF REFERENCES

- Alt, J., Lucas, T., Sanchez, S., Sanchez, P., Blais, C., & Pearman, J. (2007). *Chemical, Biological, Radiological, and Nuclear Tactical Situation Awareness. TRAC-M-TR-07-016*. Monterey, CA.
- Beris, J. V, & Whittington, E. S. (2008). *Defining Stability, Security, Transition, and Reconstruction (SSTR) Operations Requirements for Future Department of The Navy Training and Analytical Models and Simulations*. Naval Postgraduate School, Monterey, CA. Retrieved from <http://hdl.handle.net/10945/3880>
- Blais, C., Dodds, R., Pearman, J., & Baez, F. (2009). Rapid Scenario Generation for Multiple Simulations: An Application of the Military Scenario Definition Language (MSDL). In *Proceedings of the Spring Simulation Interoperability Workshop* (Vol. San Diego,). Simulation Interoperability Standards Organization. Retrieved from <http://hdl.handle.net/10945/31206>
- Bohemia Interactive. (2012). VBS2 2.0 User Manual. Version 2.12.08. Retrieved from <http://manuals.bisimulations.com/vbs2/2-00/manuals/>
- Chambers, S. (2012). *Joint and Coalition Warfighting (JCW) / Joint Training Data Services (JTDS)*.
- Department of Defense. (2004a). Data Sharing in a Net-Centric Department of Defense. DoD Directive 8320.02.
- Department of Defense. (2004b). Interoperability and Supportability of Information Technology (IT) and National Security Systems (NSS). DoD Directive 4630.05. Retrieved from <http://www.dtic.mil/whs/directives/corres/pdf/463005p.pdf>
- Department of Defense. (2006). Department of Defense Acquisition Modeling and Simulation Master Plan. Retrieved from www.acq.osd.mil/se/docs/AMSMP_041706_FINAL2.pdf
- Department of Defense. (2007). Modeling and Simulation (M&S) Management. DoD Directive 5000.59. Retrieved from www.dtic.mil/whs/directives/corres/pdf/500059p.pdf
- McGroder, D. P., & Lashlee, J. D. (2011). An Emerging DoD M&S Enterprise Data Strategy. In *Proceedings of the Spring Simulation Interoperability Workshop*. Boston, MA: Simulation Interoperability Standards Organization.
- Military Scenario Definition Language Product Development Group. SISO-STD-007-2008. (2008). Standard For: Military Scenario Definition Language (MSDL).

Orlando, FL: Simulation Interoperability Standards Organization. Retrieved from <http://www.sisostds.org/ProductsPublications/Standards/SISOStandards.aspx>

Myers, S. (2012). Live Virtual Constructive Integrating Architecture (LVC-IA).

Scudder, R., Willoughby, M., Carlton, B., Lashlee, J. D., Rose, S., & O'Connell, R. (2011). Rapid Data Generation for Modeling and Simulation. In *Proceedings of the Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC)*. Orlando, FL.

SISO Base Object Model Product Development Group. (2006). Base Object Model (BOM) Template Specification. SISO-STD-003-2006. Orlando, FL: Simulation Interoperability Standards Organization. Retrieved from <http://www.sisostds.org/ProductsPublications/Standards/SISOStandards.aspx>

TCM Gaming. (2010). U.S. Army MILGAMING. Retrieved September 22, 2013, from <https://milgaming.army.mil/>

The Under Secretary of Defense (Acquisitions, Technology, and Logistics). (2013). Implementation Directive for Better Buying Power 2.0 - Achieving Greater Efficiency and Productivity in Defense Spending. Washington, DC: Office of the Under Secretary of Defense (AT&L). Retrieved from [http://www.acq.osd.mil/docs/USD\(AT&L\)BBP2.0ImplementationDirective\(24April2013\).pdf](http://www.acq.osd.mil/docs/USD(AT&L)BBP2.0ImplementationDirective(24April2013).pdf)

World Wide Web Consortium (W3C). (1999). XSL Transformations (XSLT) Version 1.0. (J. Clark, Ed.). Retrieved September 20, 2013, from <http://www.w3.org/TR/xslt>

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California
3. Research Sponsored Programs Office, Code 41
Naval Postgraduate School
Monterey, CA 93943
4. Michael O'Neal
Marine Corps Systems Command (DC SIAT)
Quantico, VA

