DEPARTMENT OF THE ARMY
PROGRAM EXECUTIVE OFFICE
SIMULATION, TRAINING AND INSTRUMENTATION
12350 RESEARCH PARKWAY,
ORLANDO, FLORIDA 32826-3276

REPLY TO
ATTENTION
OF

SFAE-STRI-PEO

26 November 2008

MEMORANDUM FOR PEO STRI Project/Program Managers (PMs), Executives, and Principal Assistant Responsible for Contracting (PARC)

Subject: Integration and Interoperability Common Component Certification Statement (I2C3S)

1. As our PEO STRI mission grows, our opportunities for improved business practices and increased value to the Army also increase. In February 2007, I chartered the Integration and Interoperability Advisory Board (I2AB) to provide senior-level, cross-PEO integration of our testing and training activities. The I2 Concept of Operations (CONOPS), approved by the Board of Directors (BoD), identified three focus areas for the I2AB: Program Synchronization, Interoperability and Common Components. As part of the Common Components focus area, this memorandum directs a standard process to ensure we fully consider opportunities offered by the commonality of our varied training and testing products. This implementation of systematic reuse will result in savings for the Army, faster product delivery and enhanced quality.

2. I direct the following:

a. PM/Executive to complete the enclosed certification statement for new contracting efforts, including Engineering Changes, valued at $1M or more. The checklist will help ensure that our common components, including architectures, models and standards, are fully considered as a normal course in the way we operate.

b. The PARC/Acquisition Center to ensure a PM/Executive-signed certification statement is part of the procurement file, before releasing draft Request for Proposals or other procurement solicitation documents.

c. The I2AB to review signed certifications and report metrics regarding certification categories (Common Standards, Products, Architectures and/or Repositories (CSPAR) Compliance, Waiver by Business Case or Expediency of Need Exception) by PM/Executive at the PEO Business Reviews. For the Expediency of Need Exceptions, the reported metric is to include the projected date for the Business Case completion.

3. It is not my intent for the I2AB to be in the approval loop for the certification statements or interject itself as a review authority before the certification statements are approved by the PM/Executive. I fully trust the integrity of the PM/Executives to perform the appropriate reviews, understanding my general guidance to make maximum use of our common components commensurate with good business and management practices.

4. The CSPAR checklist, attached as Enclosure 2, is an evolving document. Changes to the CSPAR checklist do not require this guidance to be reissued. Changes to the CSPAR checklist after a certification has been approved by the PM/Executive will not require the PM/Executive to update the certification to the new checklist, unless there is a contracting change (new award or equipment change proposal) valued at $1M or more.

5. The certification requirement described above will remain in effect unless changed by my written direction.

James T. Blake

JAMES T. BLAKE
Program Executive Officer

2 Encls
1. Certification Statement
2. CSPAR Checklist

CF:
I2AB

Integration and Interoperability (I2) Common Component Certification

Reference: PEO STRI Policy on the Use of Common Standards, Products, Architectures and/or Repositories (CSPAR)

Applicability: This certification is applicable for all new Request for Proposals (RFP) issued by PEO STRI and Engineering Changes, valued at $1M or more for training or testing products.

Under the authority of the I2AB and as approved by the PEO STRI Board of Directors, the following certification is entered "For the Record" on the Request for Proposal or contract modification associated with <Enter Program Name>.

☐ I certify that I have reviewed the CSPAR policy and find that the RFP/contract modification requirements are in compliance with the *applicable* architectures; components; and standards identified in the CSPAR checklist.

☐ I certify that I have reviewed the CSPAR policy and have either waived or allowed deviation from compliance for the subject program RFP/contract modification to applicable architectures; components; and standards as identified in the attached checklist. A business case analysis has been performed to justify each waiver or is supplemented with a migration plan for compliance for approved deviations.

☐ I certify that expediency of need (or other compelling reasons existed) warranted the issuance of an RFP/contract modification without compliance review, but that subsequent analyses will be performed and adjustment made to source selection/contract activities as necessary to achieve compliance or waiver, Not Later Than: _____.

Concurrence/Non-concurrence:

Engineering (Associate Chief Systems Engineer/Chief Engineer): _____

Product Manager/Management Office Director/Assistant Project Manager _____

Approval:

Project Manager/Executive: _____

Signature/Date
Title

Common Architectures/Frameworks, Components, Standards, Interfaces/DIFs, Data Models, and Repositories (As of 16 October 2006)

Architectures/Frameworks	Application	
	Yes	No
Common Training Instrumentation Architecture (CTIA)		
Army Training Information Architecture (ATIA)		
SE Core Virtual Simulation Architecture (VSA)		
Test and Training Enabling Architecture (TENA)		

Components	Application	
	Yes	No
OneSAF Simulation Core Product		
Warfighters' Simulation (WARSIM) Aggregate Models/Behaviors		
WARSIM Intelligence Model (WIM)		
Common Synthetic Natural Environment (SNE) [OneSAF Environment Runtime Component (ERC)]		
Common C4I Adapter		
SE Core Common Moving Models		

Standards	Application	
	Yes	No
Synthetic Environment Data Representation and Interchange Specification (SEDRIS)		
High Level Architecture (HLA)		
Objective Terrain Format (OTF)		
Sharable Content Object Reference Model (SCORM) for Advanced Distributed Learning (ADL)		

Common Architectures/Frameworks, Components, Standards, Interfaces/DIFs, Data Models, and Repositories (As of 16 October 2006)

Common Interfaces/DIFs	Application	
	Yes	No
Military Scenario Definition Language (MSDL)		
MILES Communication Code (MCC)		
Targets Common Digital Architecture		

Data/Object Models	Application	
	Yes	No
OneSAF Objective System (OOS) Environmental Data Model (EDM) (Default EDM)		
FCS Brigade Combat Team Information Model (BCT IM) with Training IPT Logical Data Model (LDM) extensions		
Federation Object Models (FOMs):		
- Joint Live, Virtual, Constructive (JLVC) FOM		
- Joint Land Component Constructive Training Capability (JLCCTC) Entity Resolution Federation (ERF) FOM		
- Joint Land Component Constructive Training Capability (JLCCTC) Multi Resolution Federation (MRF) FOM		
- Real-time Platform Reference (RPR) FOM		

Guidelines (reference only)	Application	
	Yes	No
Live Training Transformation (LT2) Graphical User Interface (GUI) Style Guide	NA	NA
Product Line Architecture Frameworks such as OneSAF Objective System's PLAF	NA	NA

Repositories	Application	
	Yes	No
Army Model Exchange (AME)		

PEO STRI

Common Standards, Products, Architectures and/or Repositories

(CSPAR) Baseline Document

Version 1.0

16 October 2006

U.S. Army Program Executive Office (PEO)

Simulation, Training and Instrumentation (STRI)

12350 Research Parkway

Orlando, FL 32826-3276

Table of Contents

Introduction 1
1.0 Architectures / Frameworks 2
2.0 Common Products / Components 13
3.0 Standards 21
4.0 Common Interfaces/DIFs 19
5.0 Data Models 25
6.0 Processes/Guidelines 26
7.0 Repositories 28
8.0 Near Future Components 28

Introduction

The PEO STRI Common Standards, Products, Architectures, and/or Repositories (CSPAR) baseline document provides overview information associated with items identified in the PEO STRI CSPAR Policy memo dated 16 October 2006.

This document is divided into eight sections, one for each type of artifact and a final section on potential future components or standards. Each artifact section provides details on the identification, description, and current version number/description and a point of contact. The potential future components section is a list of items that are in development that the EASC may incorporated into future versions of the baseline document.

1.0 Architectures / Frameworks

1.1 Architecture/Framework ID: Common Training Instrumentation Architecture (CTIA)

1.1.1 Architecture/Framework Description: CTIA is the software framework by which the PM TRADE Live Training Transformation (LT2) strategy will develop product line components that are re-usable and composed to instantiate multiple Instrumentation Training Systems that shall be deployed to Combat Training Centers, Homestations, and instrumented ranges. The CTIA program provides the framework (protocols, standards, interfaces, etc.) and interfaces with other live, virtual and constructive environments. CTIA is also a Future Combat Systems (FCS) complementary program that is a major contributor to the FCS Training Common Components. CTIA represents PEO STRIs common architecture for the Live Training Domain and its strategy to interoperate with other PEO STRI Virtual and Constructive Domains.

Additional information on CTIA can be found at the following web sites:

LT2 SharePoint: https://ace.peostri.army.mil/sites/pmtrade/LT2/default.aspx
LT2 Portal: https://www.lt2portal.org/

Components are a defined portion of the architecture that can be "plugged in and out." A component has a set of requirements it must meet, a well defined and managed interface specification, and a well-defined capability. Within CTIA these components support requirements variability, reuse, extensibility, adaptability and autonomous development. The CTIA compliant plug and play components may be common to multiple products or unique to a specific product. The plug and play components have been divided into two primary types, instrumentation and processor. These two types of plug and play components are described below.

Figure 1 below provides a layered view of the component based CTIA client-server architecture. The infrastructure components of the architecture include:

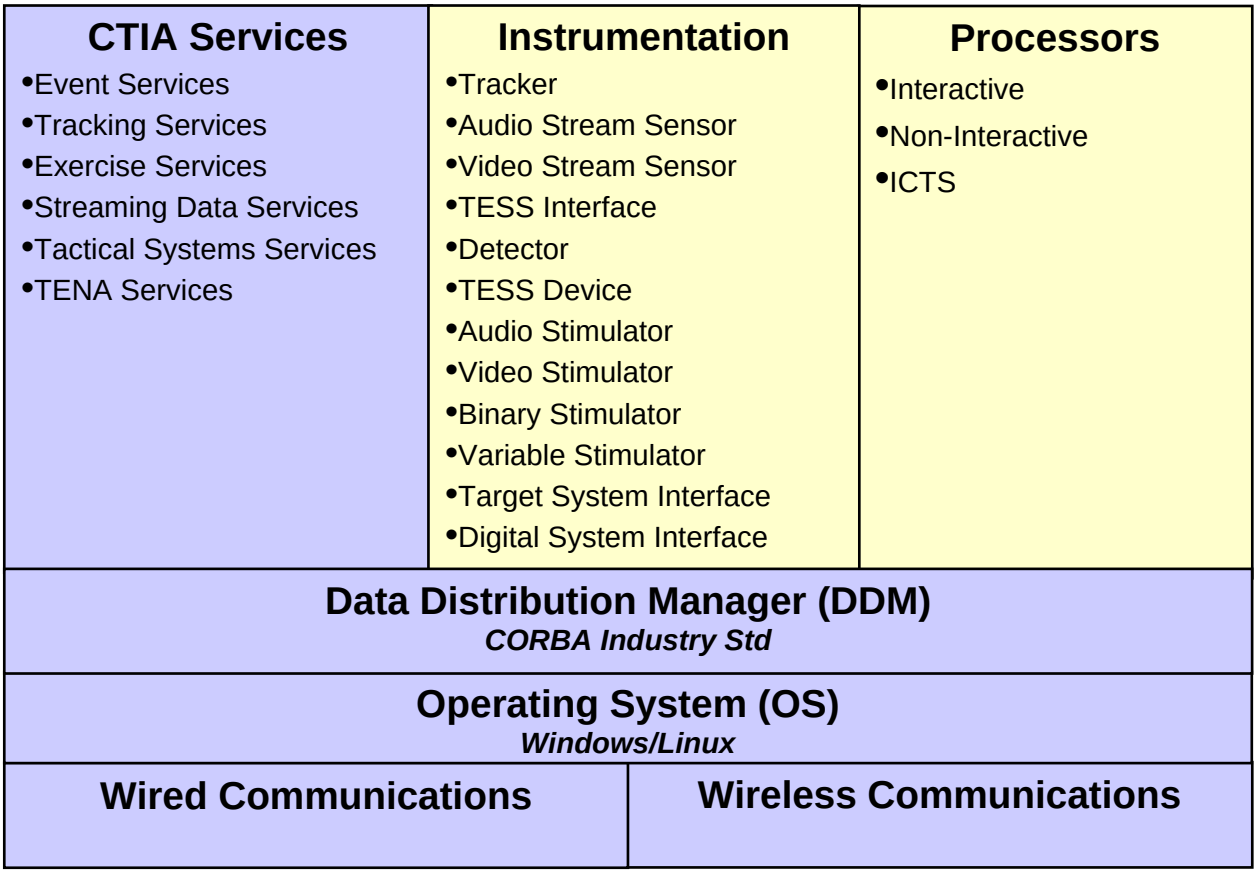

Figure 1 – CTIA Client-Server Architecture

- Processors – This category of components have the capability of producing and consuming all types of CTIA data. This includes tools like After Action Review (AAR) Analysis and Exercise Monitoring as well as Computer Generated Forces (CGF). Processor components can be interactive or non-interactive. Interactive processor components have a user interface and are comprised of the common toolset required across the family of LT2 systems to plan, prepare, execute and evaluate training. Non-interactive processor components include gateways to other simulation or training systems and instrumentation system-based simulations (e.g., Area Weapon Effects). Processors components encapsulate computational functions that have the capability of producing and consuming all types of CTIA data.
- Instrumentation – This category of components encapsulates the hardware and software needed to collect data from and control live entities. Instrumentation is typically associated with live participants but can be used for simulated. Instrumentation components provide the interfaces to other subsystems and systems such as Tactical Engagement Simulation Systems (TESS), target systems, and Command and Control (C2) systems. In addition, they provide encapsulation of instrumentation such as individual TESS devices, trackers, video cameras, Battlefield Effects Simulators, and control devices in a Mobile Operations on Urban Terrain (MOUT) facility.
- CTIA Services – The CTIA Services provide domain specific services to support plug and play component clients. When deployed, these services are tailored to account for

things such as training exercise scale, available infrastructure, and network variability. The service interfaces use a predefined object data model to ensure component interoperability and eliminate "stove pipe" systems. These interfaces are defined using the Common Object Request Broker Architecture (CORBA) interface definition language (IDL), which defines object data structures without methods. The CTIA Object Models provide methods and higher-level abstractions (e.g. proxies for remote objects). The CTIA services maintain objects representing exercises, organizations, and participants. It provides services accessible through the Data Distribution Management (DDM) such as unique ID, entity filtering, and brokering control of instrumentation. It provides access to databases for exercise specific and exercise independent data, and encapsulates the databases.

- Communication – These components provide communications between system elements either through wired or wireless networks.
- DDM and Operating System – These components are necessary to complete the definition of the system. DDM provides the back-bone into which other components plug into.

1.1.2 Architecture/Framework Version: CTIA Spiral 5

1.1.3 POC: Will Samper (407) 384-3626

1.2 Architecture/Framework ID: Army Training Information Architecture (ATIA)

1.2.1. Architecture/Framework Description

The ATIA-Migrated (ATIA-M) architecture is divided into components represented by Automated Information Systems (AIS) that support, by applying a "self-organizing-to-task" operations concept, six ATIA-M User Configurations. The functionality of the architecture resides in nine AISs as software objects, frequently as Enterprise Java Beans (EJBs). These AISs are used as the building blocks for six user configurations. An analogy would be to compare the AISs to collections of Lego™ pieces. Each user configuration is assembled, as needed, from the component functions contained in the AISs. Data are not allocated to AIS. All training data, including reference tables, are stored in the ATIA-M Enterprise Database (EDB). All software objects are part of the common architecture, subscribe to common development standards, and use the EDB as the common storage location. The physical instantiation of ATIA is called the ATIA-M.

Two AISs, Common Core Services (CCS) and Digital Library/Data Repository (DLDR), are common to all user configurations. The former provides much of the infrastructure services, such as login, security, and user interaction, while the latter manages the collection of finished training products in the Reimer Digital Library and the "bench stock" (courseware component parts) for training developers and other required data in the Data Repository. Access to and interoperability with ATIA-M is through CCS. Access to ATIA-M products and data is through DLDR.

1.2.2 ATIA-M Interoperability Domains

ATIA-M currently provides for interoperability in three separate domains. They are as follows:

- Data Extraction
- Data Injection
- Function Point Development

1.2.2.1 Data Extraction Domain

The Data Extraction Domain is the simplest level of ATIA-M interoperability. Extraction of data is accomplished via web services.

Access to ATIA-M objects is accomplished by using both the CCS and DLDR AISs. The CCS will provide authentication and identification and the DLDR will provide access to the ATIA-M data objects repository.

Access to both AISs will be via ATIA-M web services.

1.2.2.2 Data Injection Domain

The Data Injection Domain is the next level of ATIA-M interoperability and represents injecting new or modified data into ATIA-M. ATIA-M uses well defined application programmers interfaces (API) that are used to inject data objects into ATIA-M. It is also possible to inject documents into the DLDR. These documents must be in one of the accepted formats for documents.

The ability to inject data into ATIA-M is accomplished by using the CCS and the DLDR AISs. As with the Data Extraction Domain, the CCS will provide user authentication and identification and the DLDR will provide access to the ATIA-M data objects.

Access to both AISs will be via ATIA-M web services.

1.2.2.3 Function Point Development Domain

The Function Point Development Domain is the most complex level of ATIA-M interoperability. This domain describes how a software development team might develop a function point that would be fully integrated (and thus, interoperable) with ATIA-M. The ATIA-M architecture is such that many development teams may write software that will exist within this domain.

To interoperate with ATIA-M in the Function Point Developer Domain, the developer will have to conform to the ATIA-M development and coding standards. The ATIA-M Function Point software will have to be compiled and packed in the same manner as ATIA-M AIS software.

1.2.2.4 Architecture/Framework Version: ATIA-M Version 5.3 which was deployed to the ATIA-M Production Environment in April of 2006.

1.2.2.5 POC: Harry Sotomayor (407) 384-3608

1.3 Architecture/Framework ID: SE Core Virtual Simulation Architecture (VSA)

1.3.1 Architecture/Framework Description: PM CATT has established the Virtual Simulation Architecture (VSA) as the architecture for the virtual simulation domain. The Synthetic Environment (SE) Core program is developing VSA as a common Product Line Architecture (PLA) supporting the development of new and the evolution of current PEO STRI virtual simulation training systems. The VSA applies the PLA concepts to provide a set of reusable products, components, services interfaces, and standards that allow current and future PEO STRI programs to satisfy their service needs.

The PM CATT end goal is to have a full product line of virtual training simulation systems based on the VSA architecture and products. For current systems, VSA products can satisfy the need for increased functionality, technology refresh, and reduction of operational and maintenance costs. A business case analysis and phased evolution path will determine how rapidly the current systems will adopt the VSA.

The SE Core program is influencing the VSA from a product line perspective by providing a product line tool-box consisting of architectural definitions and rules along with a set of common core product line assets or building blocks.

1.3.1.1 Major Technical Drivers

Four motivations are identified as the main technical drivers for the VSA: interoperability, reuse, current system investment and adaptability and extensibility.

1.3.1.1.1. Interoperability

VSA defines and addresses interoperability between multiple products deployed within a training system and between multiple training systems. In each case, the interoperability capability must address:

A common data model/protocol set to provide a common language (e.g., understand and speak the same language), a common time management to provide time synchronization throughout the simulation, a common sense of place provided via a common correlated synthetic natural environment (SNE) and a commonality of essential model behavior to ensure realistic interactions between simulation entities (fair fight).

1.3.1.1.2. Reuse

To eliminate the duplicate development and sustainment costs of the traditional stovepipe development, major software applications must be available for reuse across the domain. This

requires a common repository for the reusable software such that individual programs can locate, understand, and tailor reuse candidates with the minimum amount of effort. Additionally, the VSA must maximize support for incorporating software from current training systems.

1.3.1.1.3. Current System Investment

The Army has significant human and capital investment in existing training systems. Much of the near term usage of the VSA will be to update and evolve the current PEO STRI virtual training systems. The VSA must provide a cost effective path for these programs to migrate toward the VSA to allow those systems to reap the reuse and interoperability benefits of the VSA. Consideration must also be made to avoid interruption to the ongoing training capabilities of these systems during that migration.

Additionally, there is a great deal of working functionality within the current systems that should be leveraged and made available through the VSA. The VSA will facilitate the minimization of effort associated with reusing that functionality.

1.3.1.1.4. Adaptability and Extensibility

The realities of the contemporary operating environment (COE) (emerging technologies and doctrine that support the Warfighter) are important to Army transformation and result in the need for virtual simulation systems to support new platforms, doctrine, and tactics, techniques and procedures (TTP). These new capabilities are added to current and future virtual simulation systems in the form of new and modified models, visuals, and behaviors. Consider also the diversity of the Army's virtual training domain, which includes individual, crew, collective and combined arms. This, along with the advancements in virtual simulation technology, is broadening the operational capability that can be represented by these virtual training systems.

Virtual simulation systems must be extendable to support these changes. The flexibility to adapt quickly, affordably, and sustainably must be designed into a virtual simulation system. The future training system requirements for these domains can be anticipated to a limited extent. For VSA to be relevant now and in the future, it must be extendable to support the evolving requirements of each of these training domains. Otherwise, the significant development investment in the virtual simulation system is lost because the system becomes irrelevant over time.

1.3.2. VSA Definition

The VSA is specified in the Product Line Architecture Specification (PLAS) document. The PLAS provides SE Core program stakeholders (end users, clients, customer, developers, etc.) with multiple integrated architectural views of the VSA. The primary focus of this document is product line decomposition, architectural boundaries, and overall interoperability interfaces, which are all necessary for proper component development and use. Figure 2 - PLAS Document Breakdown, illustrates the various specifications and architecture views contained in the VSA PLAS.

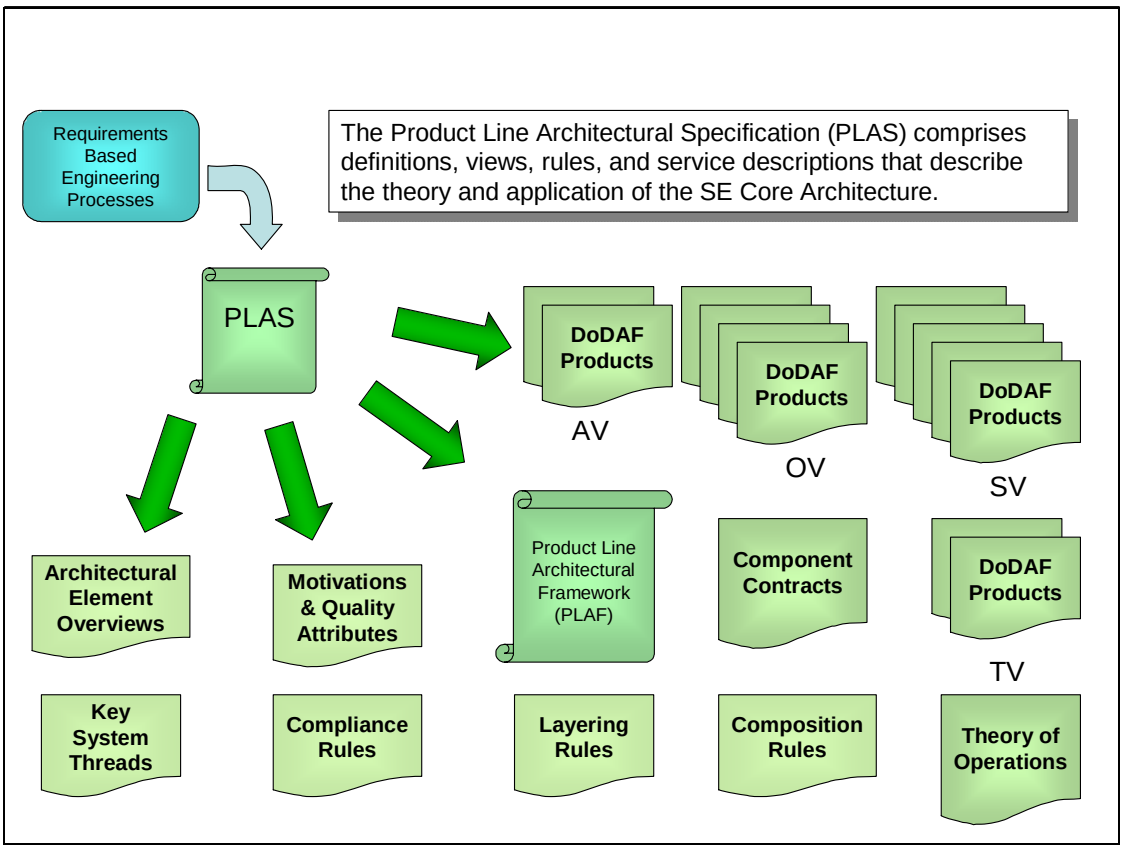

Figure 2 - PLAS Document Breakdown

1.3.2.1 VSA PLAF

One essential view contained within the PLAS is the Product Line Architecture Framework (PLAF), which is a two-dimensional view providing a high-level quick reference for engineering and other technical staff. The boxes contained in the bottom of Figure 3 (labeled "System View" on the right) represent the functional pieces of the VSA PLA. The PLAF view shows the architectural layered organization of the VSA. That same layering is also reflected in much of the VSA PLAS document structure. The PLAF is a tool intended to assist the developers of systems, by helping them identify the architectural components, boundaries, breakdowns, and typical compositional relationships between the layers of the architectural elements.

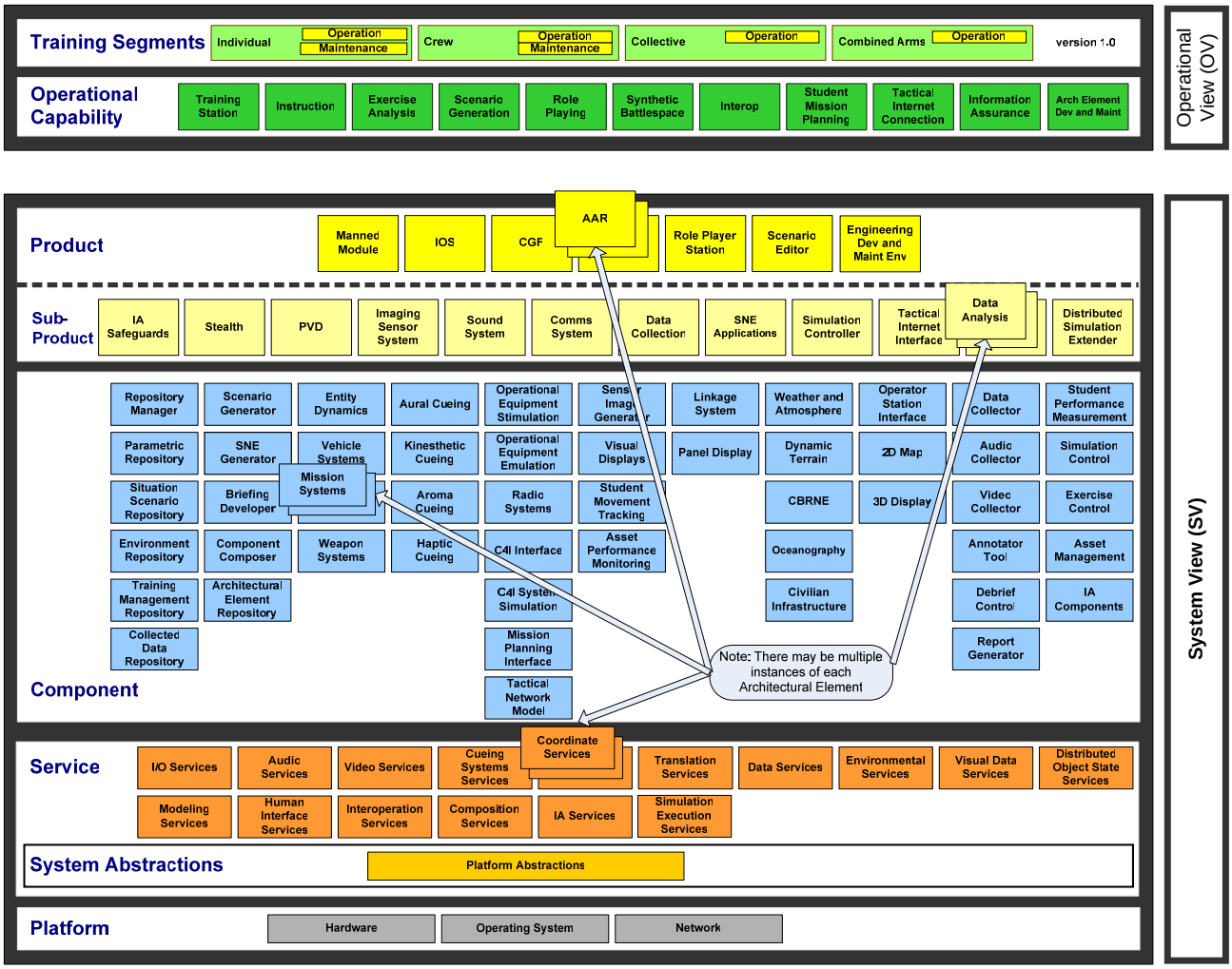

Figure 3 – VSA PLAF

The VSA PLAF system view area is divided into the following layers:

1. Training Segments – Training segments list the major groupings or segments of training systems within the VSA domain.
2. Operational Capability – The operational capability layer shows the high-level training operational activities performed by the domain training systems. The operational activities describe the major tasks/functions that are required for the domain training system sites to accomplish their missions.
3. Product – Products are stand-alone, end-user visible functionality representing the very high level applications or application suites that are typically deployed as a unit. They represent significant architectural pieces of a training system, such as an AAR or instructor operator station (IOS). The VSA defines the specific interface protocols to facilitate the Product level interoperability.
4. Subproduct – Subproducts are just smaller scale products and maintain the same characterizations as a product. The hardware analogy is that of a line replaceable unit (LRU), allowing substantial subsystem level functionality to be swapped out within a

training system. Subproducts will often be deployed into a training system as a collection composing a full Product; however, they may be deployed individually as necessary to meet a specific training system's needs. For example, a simulation controller Subproduct may be deployed at an AAR workstation, allowing an operator to perform exercise control from that physical station area.

5. Component – Components are the systematically reusable building blocks of Products and Subproducts. This is where the majority of software is produced within the VSA framework. Components are built on the VSA services providing further software reuse, portability, and interoperability.
6. Service – The VSA services are a set of common software service interfaces that provide the framework or infrastructure on which VSA common components are built. The common services promote systematic reuse and consistency for component distribution, component and service discovery, data models, data distribution, component communications/messaging, scaling, and portability across the VSA common components.
7. Platform – The platform layer represents the host hardware, operating systems, and network technology supported by the VSA. This is typically commercial off-the-shelf (COTS) or open source and is not being developed by SE Core Architecture and Integration (A&I). The VSA will, however, specify requirements on this layer such as real time execution support

1.3.2.2 Architectural Element Specification

The contents of the Product/Subproduct, component, and service layers are referred to collectively as VSA architectural elements. These architectural elements represent the set of reusable core assets that can be leveraged by the developers of systems. Each architectural element within the VSA is specified by a component contract. A component contract is a uniform collection of information and artifacts providing a description of an architectural element within the VSA and is intended for use by architectural element and training system developers. It provides a description of the architectural element, specifies functional requirements and interfaces, and captures other information such as known limitations, specific to each implementation instance. It supports distributed development and systematic reuse by describing the necessary interactions of the "black box" component with its environment.

1.3.3 VSA Client (Customer) Support

For the VSA and the SE Core A&I program to be successful, the VSA definition, products, and architectural elements must be readily and easily accessible, must be understandable, and must assist users with processes and tools to help stand up or extend a development environment. These needs are fulfilled by the SE Core repository.

The SE Core repository is web-based, containing the VSA definition documentation, architectural elements and their corresponding documentation, composition tools, development processes, and build procedures.

VSA client programs and product developers will access the repository though a web-based portal. Interfaces for developers to locate Products and Components of interest are currently

being developed and include a graphical drill-down capability through an interactive PLAF view and a keyword search capability. Once located, the component contract will point to all information necessary to select, build, interface to, test, and extend the element. System developers then improve and expand the SE Core A&I baseline with feedback of to the VSA, including requests for new or expanded elements and discrepancy reports against existing architectural elements. Evolution and maintenance of the VSA and elements within the repository are managed by an architecture review board. Repository procedures will be developed to ensure existing users of the element are not adversely affected by changes and the modified element meets the specifications in the component contract.

Another client program support feature is the VSA Evolution Plan (VEP). In accordance with the major motivation to preserve investments in current training systems, the VEP describes the philosophy and processes for planning and migrating current systems to the VSA. It will aid in identifying areas of opportunity for evolution including the development of the business case and estimating the engineering effort. Analysis of the candidate program architecture helps identify low hanging fruit where the program can reap improvements with minimal costs. Compliance assessment tools help evaluate the distance of a system from the VSA and the cost of the evolution. As programs move through this evolution process, data will be collected to form a historical base. Access to that history will help to refine the processes and improve the estimation speed and accuracy.

As programs adopt VSA products, they also adopt the VSA framework protocols and rules, making it easier to adopt and interoperate with further Products. The end state of this evolution is a training system that conforms to, and is consistent with, the VSA Products line.

1.3.4 Architecture/Framework Version: SE Core VSA Version 1.0

1.3.5 POC: Brian Kemper (407) 384-3816

1.4 Architecture/Framework ID: Test and Training Enabling Architecture (TENA)

1.4.1 Architecture/Framework Description

TENA is a common architecture and requisite software suite developed by the Office of the Secretary of Defense (OSD) Central Test and Evaluation Investment Program (CTEIP) to integrate testing, training, simulation, and high performance computing technologies distributed across many facilities. The establishment of TENA fosters reuse and interoperability of range assets as well as reducing range development, operation and maintenance costs. TENA is also the Joint National Training Capability (JNTC) architecture for live training and is used primarily as a communication architecture.

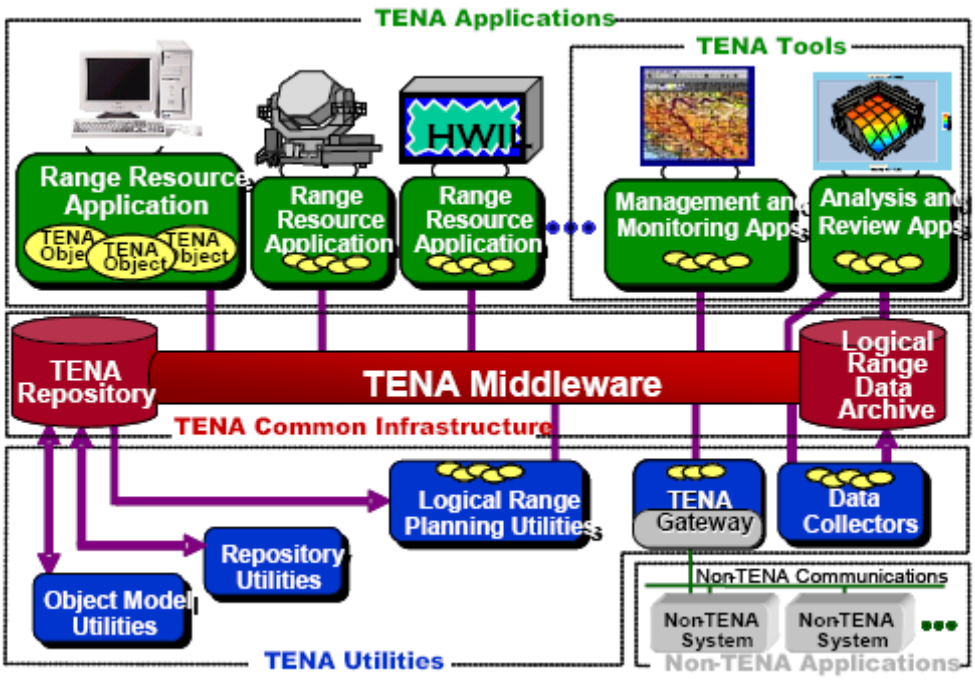

Figure 4 – TENA Architecture

The core of TENA is the TENA Common Infrastructure, including the TENA Middleware, the TENA Repository, and the TENA Logical Range Data Archive. There is also the TENA Object Model, which defines the common data and interfaces shared by all range applications. There are also a number of tools, utilities, and gateways to enable many range resources located at geographically dispersed ranges to be integrated together.

TENA Middleware: A high performance, real-time, low-latency communication infrastructure used by range resource applications and tools during the execution of a range event. The TENA Middleware facilitates all data exchange and control commands between range systems. More importantly, the TENA Middleware provides range system developers with a unified API to support the real-time exchange of software objects, messages and data streams.

TENA Repository: Contains all the information relevant to TENA that is not specific to a given test or training event. The TENA Repository is web-enabled and functions, in essence, as a large database of databases, allowing event planners to browse and select capabilities that can be easily configured and used to support an event.

TENA Logical Range Data Archive: Stores and allows retrieval of all the persistent information associated with a test or training event.

TENA Website: www.tena-sda.org

TENA Sharepoint: https://pmitts.peostri.army.mil/default.aspx
Go to 'Shared Documents'
Go to 'TENA Information'

1.4.2 Architecture/Framework Version: The current release version of the TENA Middleware is Release v5.1.1

1.4.3 POC: Carleton Stargel (407) 384-3878

2.0 Common Products / Components

2.1. Component ID: One Semi-Automated Forces (OneSAF) Simulation Core Product and supporting components

2.1.1. Component Description: OneSAF is a composable CGF that represents a full range of operations, systems, and control processes from the individual combatant and platform level to fully automated BLUFOR battalion level and fully automated OPFOR brigade level. Unit behaviors will be modeled to the BLUFOR battalion and OPFOR brigade level for selected units, and command entities will be modeled to the BLUFOR brigade and OPFOR division level. OneSAF will have a variable level of fidelity that supports modeling and simulation (M&S) domains. It will accurately and effectively represent specific activities of command, control, communications, computers, intelligence, surveillance and reconnaissance (C4ISR), combat support (CS), and combat service support (CSS). It will also employ appropriate representations used to meet the current force (including the Stryker Brigade Combat Team) requirements and facilitate meeting future M&S efforts like Future Combat System (FCS), Modeling Architecture for Research, Technology and Experimentation (MATRIX), or Battle Lab Collaborative Simulation Environment (BLCSE).

2.1.2 Component Version Description (e.g., VDD). OneSAF version 1.0

2.1.3 POC: Tamara Griffith (407) 384-3636

2.2 Component ID: Warfighters' Simulation (WARSIM) Aggregate Models/Behaviors

2.2.1 Component Description: WARSIM is a next-generation, large-scale constructive wargaming system, developed for U.S. Army command and control training. It is being developed to replace the current legacy simulation systems, e.g., Corps Battle Simulation (CBS) and Tactical Simulation (TACSIM). WARSIM is a significant advance in modeling and simulation technology deploying a wide range of resolution, fidelity and abstraction, depending on its specific use. WARSIM is a distributed, constructive wargaming simulation, designed to create a single, seamlessly integrated synthetic battlespace, including a common environmental and operational picture. Interfacing with C4I functions and equipment in the field to provide the interface between the synthetic battlespace and the training audience, WARSIM creates a training environment intended to be indistinguishable from the real-world by the training audience.

WARSIM is a constructive simulation system used to train commanders and staffs at brigade, division, corps and echelons above corps. When conducting an exercise, it can be viewed as

three layers. At the top is the training audience. The training audience consists of the commanders and staffs of the units to be trained, organized and equipped as they would be in an operational setting. Their command posts may be field locations or they may be at a training center, but they are equipped with the tactical C4I devices that would be used to conduct actual operations.

The second layer is a set of "role players." These are people who perform the roles of the subordinate commanders and staffs of the training audience. They interact with the training audience via tactical communications and C4I tactical messages to provide the stimuli that allow a training exercise to proceed. The role players also control the third layer of WARSIM, which is the computer simulation of the battlespace. The role players provide the military skills to direct the simulated units and to represent the persons with which the training audience expects to interact. In particular, the role players provide the person-to-person voice interactions that characterize Army command and control even in this digital era. At this point, there is some ability to exchange message traffic between the simulated units and the training audience without role player intervention, but this accounts for only a small part of the interaction. The three-layer structure is shown in Figure 5**.**

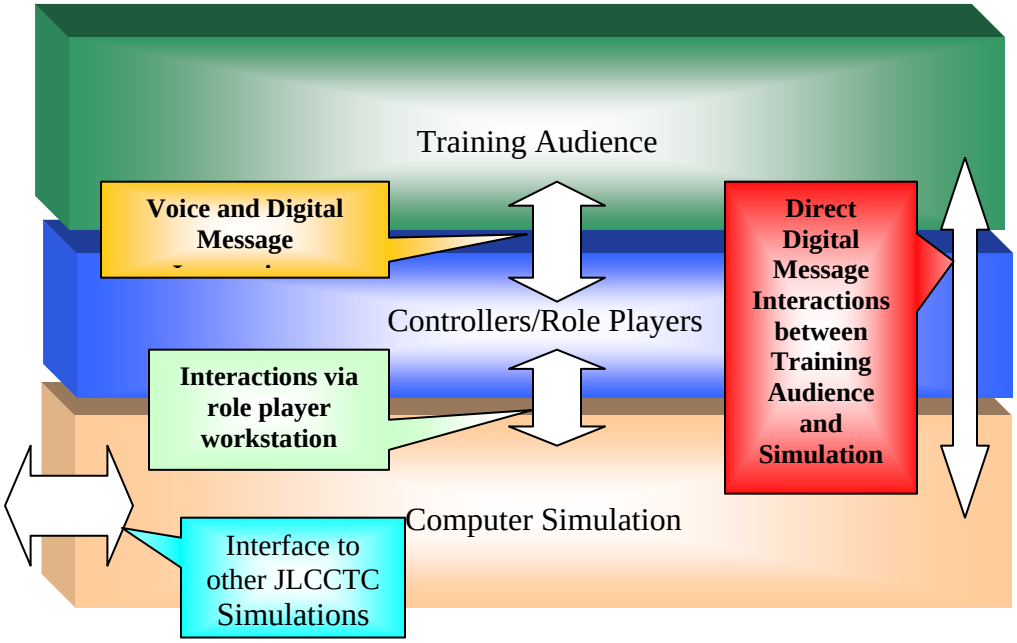

Figure 5 - WARSIM 3-Layer Architecture

Since the training audience operates with its own equipment during an exercise, the boundary of WARSIM consists of the lower two layers and the interfaces to the training audience.

The simulation component of WARSIM is a real-time model of military forces on a highly detailed representation of the terrain. It provides automated units at company level that are capable of accepting orders from role players, planning the execution of those orders and controlling the actions of subordinates (e.g., platoons). The simulation provides a level of resolution such that positions of individual vehicles can be determined. Resolution of combat

engagements occurs via simulation of the weapons effects as affected by both the terrain and the ability and condition of the simulated units. This level of detail allows the simulation to provide detailed output to role players and to the training audience such as would be provided by the Warsim Intelligence Module (WIM) sensors, i.e. Unmanned Aerial Vehicles (UAVs) and by the Moving Target Indicator (MTI) Radar of a Joint Surveillance Targeting and Attack Radar System (J-STARS) aircraft.

WARSIM is a multi-sided game. Opposing forces are controlled by role players trained in OPFOR tactics and operations. Civilians, military and paramilitary forces can be assigned varying allegiances to provide the complex political and military environments necessary to train commanders to deal with today's world.

Because the performance parameters associated with many present day weapons and sensors are classified, WARSIM must be able to handle and control classified information up to Top Secret Special Compartmentalized Information (TS-SCI).

The System Architecture is a composition of the WARSIM hardware and software along with COTS and Government-Off-The-Shelf (GOTS) software products. The design is based on the Defense Modeling and Simulation Office (DMSO) developed High Level Architecture (HLA). Communication between elements of the system is accomplished by use of the WARSIM Federation Object Model (FOM) and the HLA Run Time Infrastructure (RTI). Figure 6 illustrates the abstract relationship between the major components. The Computer Simulation piece can be viewed as four separate partitions:

- Interface to the Training Audience
- Simulation
- Controller Interface
- Infrastructure

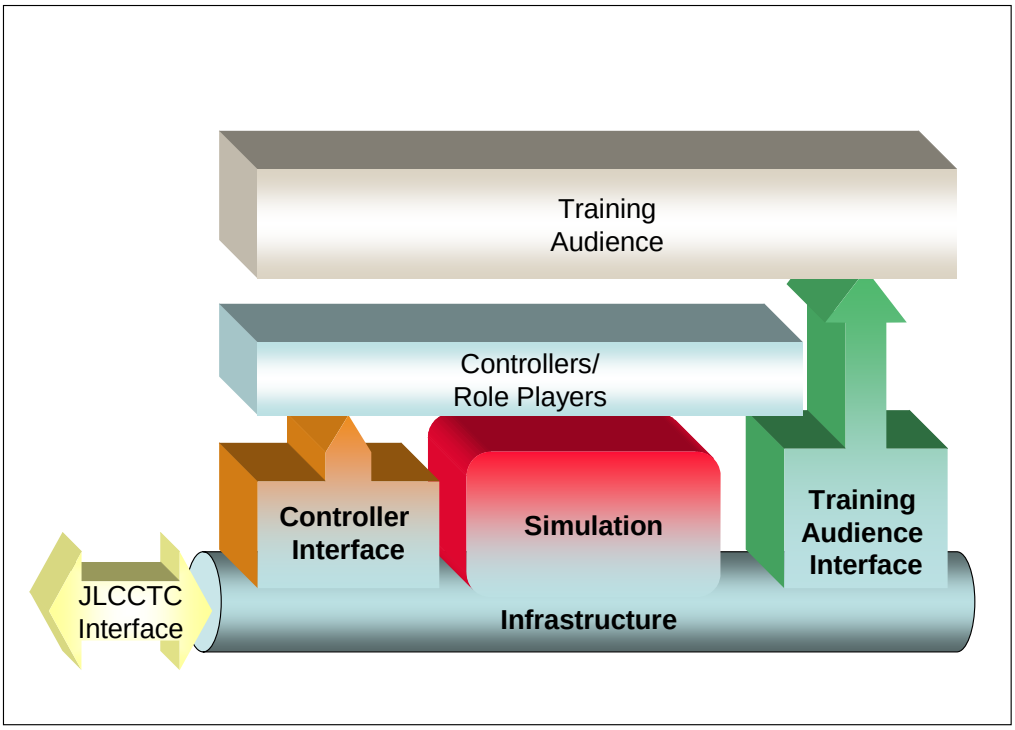

Figure 6 - WARSIM Abstract System Architecture

The three layer structure discussed earlier and shown in Figure can be seen in Figure . The lowest layer represents the hardware and software that is installed at a training center. It is divided into four partitions.

- The simulation partition models the battlespace and battlespace elements that model the combat activity used to stimulate the training audience.

- The training audience interface partition connects the training audience C4I equipment or surrogates with the simulation and with the controller stations.

- The controller interface partition allows the simulation controllers and analysts to interact with the training audience, control simulated units, and monitor the simulation system.

The infrastructure partition provides common services required by all components of the simulation system.

2.2.2 Component Version: WARSIM Version 3.0

2.2.3 POC: Dan Griffin (407) 384-3984

2.3 Component ID: WARSIM Intelligence Module (WIM)

2.3.1 Component Description: WIM is being developed to replace the current legacy simulation system, TACSIM. WIM reuses applicable technology and functionality from prototypes in intelligence simulations, such as the Federation of Intelligence, Reconnaissance, Surveillance and Targeting Operations and Research Models (FIRESTORM) and Multiple Unmanned Aerial Vehicle (UAV) Simulation Environment (MUSE). WIM provides a comprehensive and accredited simulation to meet Army training support requirements, spanning tactical up to strategic-national levels. WIM provides an environment that ensures training with scenarios that stress commanders and their staffs. WIM will evolve fully to support professional military and senior officer education, mission planning, mission rehearsal, and doctrine development. WIM provides the intelligence capabilities to the WARSIM and the Joint Land Component Constructive Training Capability (JLCCTC), which can be incorporated into various compositions to support different types of training.

WIM simulates the phases of the intelligence cycle for purposes of training multi-service commanders and their staffs. WIM is implemented as a collection of WARSIM and JLCCTC components to simulate tactical intelligence assets and behaviors, interface to C4I systems, support intelligence role-players through user interfaces, and support all phases of training from scenario generation through AAR.

Intelligence products from WIM sensors will exist at varying levels of detail (raw data, initial interpretation, correlated, and fused) depending on the training objectives and the level of the training audience. In current versions of WIM, correlation and fusion is performed by Intelligence Role Players or the training audience if part of the training objectives. WIM models will support training exercises where the training audience is at differing echelons, i.e. brigade, division, corps and echelons above corps.

WIM explicitly models tactical Signals Intelligence (SIGINT), Imagery Intelligence (IMINT), Measurement and Signature Intelligence (MASINT), and Human Intelligence (HUMINT) collection systems, the processes required for their tasking and the communications links and units required for realistically producing and distributing their associated products. These sensor models link to WARSIM platforms and equipment placing intelligence in the combat game-space. Most systems are comprised of interconnected Operations, Sensor, Processor, and Exploitation Models.

Signals Intelligence (SIGINT) Capabilities:
Communications Intelligence (COMINT) Sensor Model:

- BLUFOR, OPFOR, Generic
- Tasked from upper enclave (TS/SCI) Role Player Workstation
- Hosted on WARSIM platform performing “provide air surveillance support”, WARSIM ground platform for COMINT Ground and/or similar behavior
- Hosted on WARSIM platform providing on/off station and location/movement at high altitude and additional air-breathing platforms
- Version 4, to be fielded in 2007, will produce TS/SCI level USSID TACREP and KL messages through Guard Interface
- Produces USMTF 2000 TACREP messages

    - Consumed by upper enclave (TS/SCI) training audience through Guard Interface
    - Automatically downgraded for consumption by lower enclave (Secret) Training Audience through Common C4I Adapter
- Collects against almost all “event.message” interactions sent via radio
- Does not collect against Role Player orders
- Includes “Gist” in upper enclave reports
- Only type of Intel model running in upper enclave

Electronic Intelligence (ELINT) Sensor Model:

- BLUFOR, OPFOR, Generic
- Tasked from Role Player Workstation
- Hosted on WARSIM platform performing “provide air surveillance support” or similar behavior
- Hosted on WARSIM platform providing on/off station and location/movement at high altitude
- Produces Secret level USMTF 2000/2004 TACELINT messages through Common C4I Adapter
- Produces TS/SCI level USSID TACELINT messages through Guard Interface
- Collects against enhanced emitter public objects triggered by “event.physical_illumination” interactions generated by the WARSIM combat models

Imagery Intelligence (IMINT) Capabilities:
IMINT Sensor Model:

- BLUFOR, OPFOR, Generic
- Tasked from Role Player Workstation
- Tasking by Basic Encyclopedia (BE) number
- Re-tasking support for Time Sensitive Target (TST) operations
- Hosted on WARSIM platform performing “provide air surveillance support” or similar behavior
- Hosted on WARSIM platform providing on/off station and location/movement at high altitude and additional air-breathing platforms
- Produces Secret level USMTF 2000/2004 Reconnaissance Exploitation Report messages through Common C4I Adapter
- Produces TS/SCI level USMTF 2000 Reconnaissance Exploitation Report (RECCEXREP) and Imagery Interpretation Report (IIR) messages through Guard Interface
- Collects at the platform level based on de-aggregated equipment groups
- Includes deterministic NIIRS calculation that impacts quality of report
- IMINT Modes: EO / IR / SAR / MTI / FTI
- Utilizes the MUSE application for battlespace Visualization:
    - EO / IR / SAR / MTI Modes
    - Displays smoking dead/damaged platforms in support of Battle Damage Assessment (BDA) activities
    - Support to multiple Visualization Work Stations

Measurement and Signature Intelligence (MASINT) capabilities:
MASINT Sensor Model:

- BLUFOR, OPFOR, Generic
- Models airborne as well as ground based MASINT collection systems
- Tasked from Role Player Workstation.
- Produces secret level USMTF 2000/2004 SALUTE messages through Common C4I Adapter Interface

Human Intelligence (HUMINT) Capabilities:
HUMINT Sensor Model:

- BLUFOR, OPFOR, Generic
- Tasked from Role Player Workstation.
- Hosted on WARSIM platform performing Intel related behavior
- All WARSIM models have an optional text field or "golden nugget" of information to be collected through their interaction with HUMINT models
- HUMINT Models:
  - Long Range Surveillance (LRS)
    - Produces USMTF 2000/2004 SALUTE or EOBSREP messages through Common C4I Adapter
  - Non-Combatant Screening (NCS)
    - Produces USMTF 2004 Counter-Intelligence Information Report (CIIR) messages through Common C4I Adapter
    - Produces USMTF 2000/2004 Salute messages through Common C4I Adapter
  - Counterintelligence Force protection Source Operations (CFSO)
    - Produces USMTF 2004 Counter-Intelligence HUMINT Source Report (CIHSR) messages through Common C4I Adapter
    - Produces USMTF 2000/2004 INTREP and EOBSREP or SALUTE messages through Common C4I Adapter
  - Interrogation of Prisoners of WAR (IPW)
    - Produces USMTF 2000/2004 SALUET and INTREP messages through Common C4I Adapter
    - Produces USMTF 2004 Counter-Intelligence Information Report (CIIR) messages through the Common C4I Adapter
    - Requires "Cage" criteria to include units capable of guarding Enemy Prisoners of War (EPW)
  - Document Exploitation (Doc-EX)
    - Produces USMTF 2000/2004 INTREP messages through Common C4I Adapter

2.3.2 Component Version: Version 3.0

2.3.3 POC: Dan Griffin (407) 384-3984

2.4 Component ID: Environment Runtime Component (ERC)

2.4.1. Component Description: The Environment Runtime Component provides the services to query and modify environmental features and attributes including terrain, ambient weather, and ocean characteristics. These services include representation of the terrain skin as well as the representation of Atmosphere, Ocean, and Space (AOS). These services also include the capability to reason about and plan within the environment, using environmental, physical and cognitive model information such as used for route planning, obstacle avoidance, collision detection, vehicle placement and cover and concealment.

2.4.2 Component Version Description (e.g., VDD). ERC version 1.0

2.4.3 POC: Tamara Griffith (407) 384-3636

2.5. Component ID: Common C4I Adapter

2.5.1 Component Description: The Common C4I Adapter is a tool originally developed by OneSAF and WARSIM to facilitate two way interactions between simulation and C4I. Its core function is to translate tactical messages and provide an interface to the Army Battle Command System (ABCS). The Common Adapter uses DII COE products internally to parse and create tactical messages that are supported by the real-world systems and to provide the network interface and encrypted handshaking used to authenticate with ABCS. A graphical tool is provided to created "mappers," which are what tell the Adapter how to perform the translation from the simulation specific data to the desired tactical format. The Adapter currently supports two-way messaging with ABCS 4.x, 6.x systems, and includes limited support for ABCS 6.4. Over time, additional capabilities will be added to satisfy the requirements of multiple programs, such as C3 Driver, CTIA/LT2, AVCATT, etc. as well as to fully support objective battle command systems. This is a product line development tool with the expectation that the entire command can benefit by reducing development in each program for the same capability. Presently, as battle command systems are updated and changed each program must align their interfaces to those changes, or lose their interoperability capability. By using the Common C4I Adapter those changes will only need to be made at a single interface, with every program benefiting.

2.5.2 Component Version Description (e.g., VDD). C4I version tracks with OOS Version 1.0

2.5.3 POC: Tamara Griffith (407) 384-3636

2.6 Component ID: SE Core Common Moving Models (CM2)

2.6.1 Component Description

SE Core's CM2 effort focuses on the virtual domain. Historically, similar virtual training devices, with common training needs, have different representations of visual moving models. This is a result of stove piped development solutions for individual systems. These disparate efforts have led to duplication, fair fight issues and negative training. Additionally, moving models' documentation; development; and verification, validation and accreditation are time consuming processes. A typical simulation model can take hundreds of hours, from

development to Verification, Validation and Accreditation (VV&A). The objective of this effort is to document the requirements specification for developing CM2s that are system independent and can be exported across multiple simulations with little cost for each system, saving hundreds to thousands of engineering hours under the concept of "build once, use many times." SE Core has the challenge of building CM2s that meet the needs of multiple customers. Once accredited, they will be placed in the SE Core repository for reuse by any program. The CM2 will utilize OpenFlight format.

2.6.2 Component Version Description (e.g., VDD). N/A

2.6.3 POC: Mike Kochmann (407) 384-5491

3.0 Standards

3.1 Component ID: Synthetic Environment Data Representation and Interchange Specification (SEDRIS)

3.1.1. Component Description: SEDRIS technologies provide the means to represent environmental data (terrain, ocean, air and space), and promote the unambiguous, loss-less and non-proprietary interchange of environmental data.

3.1.2 Component Version Description (e.g., VDD). ISO/IEC 18023, 18025, and 18026

3.2 Component ID: Objective Terrain Format (OTF)

3.2.1 Component Description: OTF is the runtime format for the OneSAF Objective System (OOS) terrain database. This format is binary format which contains terrain skin, features, and UHRB models.

3.2.2 Component Version Description (e.g., VDD). OTF version 2.2

3.2.3 POC: Bruce Robbins (407) 384-3866

3.3 Component ID: Sharable Content Object Reference Model (SCORM) Advanced Distributed Learning (ADL)

3.3.1 Component Description: "SCORM" stands for "Sharable Content Object Reference Model." A "reference model" is something that shows what kinds of services will be needed to solve a particular problem, how they can be put together, the relevant standards that apply, and how they might be used. There are three primary criteria for a reference model such as SCORM. First, it must articulate guidelines that can be understood and implemented by developers of learning content. Second, it must be adopted, understood and used by as wide a variety of stakeholders as possible -- especially learning content and tool developers and their customers. Third, it must permit mapping of any stakeholder's specific model for instructional systems design and development into itself. Stakeholders must be able to see how their own model of instructional design is reflected by the reference model they hold in common.

Up-front investment is required to develop and convert learning content for technology-based presentation. These investment costs may be reduced by an estimated 50-80 percent through the use of learning content that is accessible, interoperable, durable and reusable.

Procedures for developing such learning content are state-of-the-art in e-learning, but they must be articulated, accepted and widely used as guidelines by developers and their customers. These goals can be achieved through collaborative development. Collaboration will increase the number, quality and per-unit value of learning content. Such collaboration requires agreement upon a common reference model.

ADL architects recognized early the need for a reference model that would specify learning content and its labeling, storage, and presentation in distributed learning. SCORM provides a coordinating model of standard practices that can be generally accepted and widely implemented throughout the distributed learning community.

SCORM helps define the technical foundations of a Web-based learning environment. It is a model that references and integrates a set of interrelated technical standards, specifications and guidelines designed to meet high-level requirements for learning content and systems. SCORM describes a “Content Aggregation Model” and a “Run-Time Environment ” for instructional objects to support adaptive instruction based on a learner's goals, preferences, prior performance and other factors. SCORM also describes a “Sequencing and Navigation” model for the dynamic presentation of content based on learner needs.

SCORM seeks to knit together the contributions of disparate groups and interests in the distributed learning community. It is intended to coordinate emerging technologies and capabilities with commercial/public implementations.

3.3.2 Component Version Description (e.g., VDD). SCORM 2004, Second Edition, dated 22 July 2004.

3.3.3 POC: Susan Marshall (407) 384-3817

4.0 Common Interfaces/DIFs

4.1. Component ID: Military Scenario Definition Language

4.1.1 Component Description: The Military Scenario Definition Language (MSDL) is an XML based means of specifying the necessary details of a military scenario for consumption by a simulation, planning tool, or any other applicable computerized military application (described in Figure 7). MSDL is an evolving, open specification that is not tied to any one domain or specific function. The MSDL schema is broken down into ten main topics that are used to describe a scenario: options, plan, environment, force sides, units, equipment, installations, overlays, tactical graphics and initial tasking. The developer and major proponent of MSDL is the OneSAF Objective System. OneSAF includes a tool, the Military Scenario Development Environment (MSDE) that provides a user friendly method for end users to create military scenarios. The output of that product is a scenario saved in the MSDL format, which is easily imported into the simulation. The power of an open standard such as MSDL is that any scenario creation tool, planning tool, etc. that can output in MSDL can be used to initialize OneSAF.

Also, any product that wishes to be a consumer of MSDL will be able to utilize the output of those tools (command and control systems, course of action analysis tools, etc.). MSDL has been based upon military standards such as MIL STD 2525B instead of being tied to a specific application. This ensures it is applicable for broad based use within the Army, Joint and even Foreign Military services. The MSDL is currently (June 2005) in the study group process toward becoming a Simulation Interoperability Standards Organization (SISO) standard.

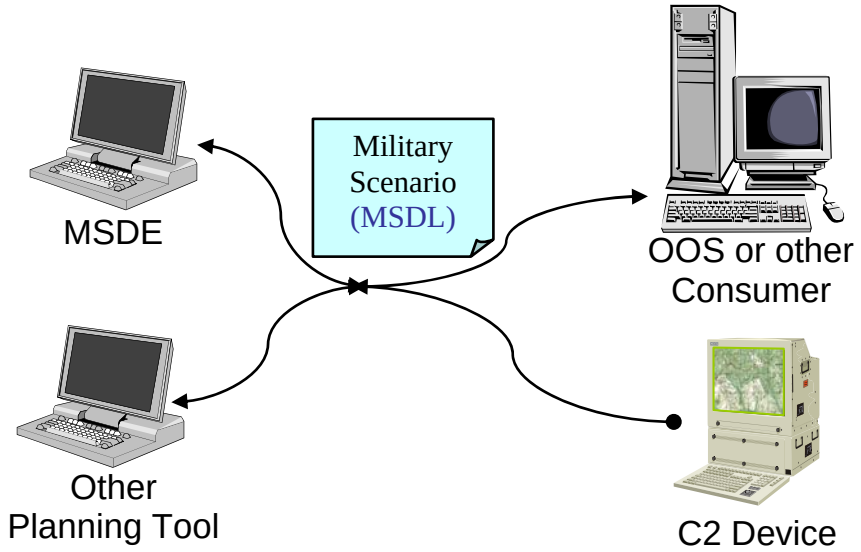

Figure 7 - MSDL

4.1.2 Component Version Description (e.g., VDD). MSDL version 4.2.2

4.1.3 POC: Tamara Griffith (407) 384-3636

4.2 Standard ID: Multiple Integrated Laser Engagement System (MILES) Communication Code (MCC)

4.2.1 Standard Description: This Standard defines the MILES MCC structure for encoding/decoding weapon type, ammunition type, player identification, and weapon/ammunition lethality effects information transported through the MILES intra-system communication channels and interfaces and through interfaces with external systems. This MCC Standard also delineates other Live Training Tactical Engagement Simulation System (LTESS) structures as they apply to the Live Training environment for devices fielded through Project Manager Training Devices (PM TRADE).

LTESS Standard has the information content, format, and functions specified herein.

4.2.2 MCC Encoded Information Content. MCC contains the following encoded information in its structure:

a. Weapon type: For example, 120mm Main Tank Gun, TOW Missile, M16 Rifle, etc.
b. Ammunition type: For example, 120mm Heat Round, TOW II Missile, 50 Cal Round, etc. (Refer to Appendix E)

c. Weapon/Ammunition effects at target as follows:
   1.1 Hit: Heavy weapon class targets with specific degree of lethality effect to be determined by target decoder system
   1.2 Hit: Light weapon class targets with specific degree of the lethality effect to be determined by target decoder system
   1.3 Near Miss for heavy weapon class
   1.4 Near Miss for light weapon class
d. Player Identification (PID)
   1.5 Unique identifier for each designated player: man, vehicle, weapon system, organizational unit etc.
   1.6 Friend or Foe designation
e. Administrative Function Information:
   1.7 Bore Sight Code
   1.8 Reset Command
   1.9 Resurrect Command
   1.10 Time Synchronization
   1.11 Other Functions

4.2.3. MCC Format. MCC format is a digital bit pattern arranged in a clocked time sequence. The time sequenced bit patterns are organized on basic word units that are assembled into successively larger and more information rich structures as follows:

a. The MCC Word, hereafter referred to as Word, is the basic bit pattern unit structure.
b. The Message is a sequence of groups of identical Words. In general, each group will contain an even number of a particular MCC Word. Each group will be separated from the following group by a time delay.
c. The Routine is a sequence of one or more Messages.

Each successive structure complexity level - Word, Message, Routine - adds additional information for transport to a MCC MILES receiver/decoder.

4.2.4 Standard Version: PMT 90-S002J, dated 19 January 2005

4.2.5 POC: Bobbi Parrish (407) 384-3588

4.3. Standard ID: Targets Common Digital Architecture (CDA)

4.3.1 Standard Description: The CDA describes the interfaces between subsystems aboard aerial, ground, and sea targets and is based on the low-cost commercial Controller Area Network Bus (CANBUS) serial digital architecture, which has been adopted by many manufacturers of automobiles, trucks, boats, and ships. CDA is the implementation of a common tri-Service architecture, which was directed for development by the Joint Targets Oversight Council in 1994 and originally funded by OSD. Its primary purpose is to increase interoperability between targets vehicles, subsystems, and payloads. CDA consists of hardware and software standards for information exchange aboard a target vehicle, including status monitoring of subsystems and passage of target subsystem control commands. By agreement among the three Services'

primary development agencies for targets (including PM ITTS), future targets and payloads are expected to incorporate CDA in their specifications, whenever reasonable.

4.3.2 Standard Version: The top-level version has remained at v 1.0 since the original release around 1999. The target CDA includes several subordinate standards for different types of targets and data.

4.3.3 POC: Dennis Brooks (256) 842-0376

5.0 Data Models

5.1 Data Model: OOS Environmental Data Model (EDM)

5.2.1. Data Model Description: The OOS EDM defines all the features and attributes for terrain, atmosphere, ocean, space, Ultra-High Resolution Buildings (UHRB) and Nuclear, Biological, Chemical (NBC) for OOS and WARSIM. The OOS EDM is the baseline EDM for the SE Core program.

5.2.2 Data Model Version: EDM version 1.7

5.2.3 POC: Bruce Robbins (407) 384-3866

5.3 Data Model: FCS Brigade Combat Team Information Model (BCT IM) with Training IPT Logical Data Model (LDM) extensions

5.3.1. Data Model Description: The Training IPT LDM extension uses the OOS EDM as the starting point and further defines those requirements to represent them in a LDM for simulation and training for the FCS BCT IM. The Command and Control Information Exchange Data Model (C2IEDM) is currently the core for the BCT IM. In order to ensure the FCS embedded training KPP is met, the C2IEDM is being extended to capture the full range of FCS operationally relevant features and attributes that the Training IPT must have represented to accomplish its mission. This includes additional requirements in the areas of environmental representation and urban operations.

5.3.2 Data Model Version: BCTIM + TRNG – WS UML Model. Document # 786-0000092250, Version 1.0 dated 22 June 2006.

5.3.3 POC: Thai Nguyen (407) 384-5456

5.4 Data Model: Joint Land Component Constructive Training Capability (JLCCTC) Joint Live, Virtual, Constructive (JLVC) Entity Resolution Federation (ERF) Federation Object Model (FOM) and Multi-Resolution Federation (MRF) FOM. Figures 8-10 provide an overview of the JLCCTC architectures.

5.4.1 Data Model Description: The JLCCTC is a collection of two integrated federations of simulation models and tools that stimulate Battle Command systems to facilitate Command and

Staff training. Simulations and Tools include Aggregate Combat, Intel, Theater Level Operations, High Resolution (Entity Level) Combat, High Resolution Logistics, C4I Stimulation, Non Kinetic Effects and AAR. ACTF currently has two main Federations:

- Entity-Resolution Federation (Entity only): Also know as DBST, similar to JNTC JLVC, future home of OOS. The JLVC FOM that is used in this federation is the federation data model that is considered an evolving quasi-standard for constructive simulation federations as well as the FOM or federation data model of choice when linking to Virtual and/or Live simulations or applications to constructive federations.
- Multi-Resolution Federation (Combination Aggregate/Entity Level): Similar to JNTC JMRM, and future home of WARSIM and OOS.

5.4.2 POC: Mike Wright (407) 384-3873

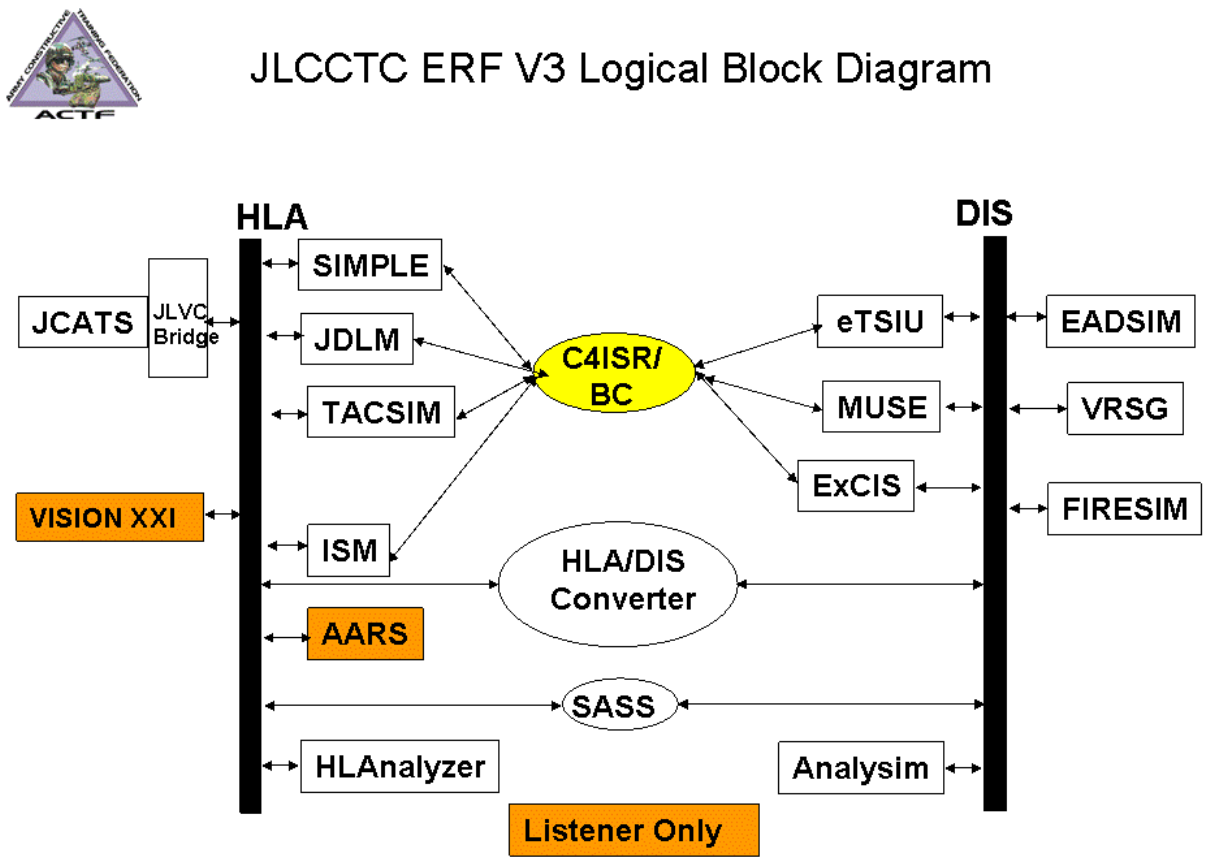

Figure 8 – JLCCTC ERF V3 Logical Block Diagram

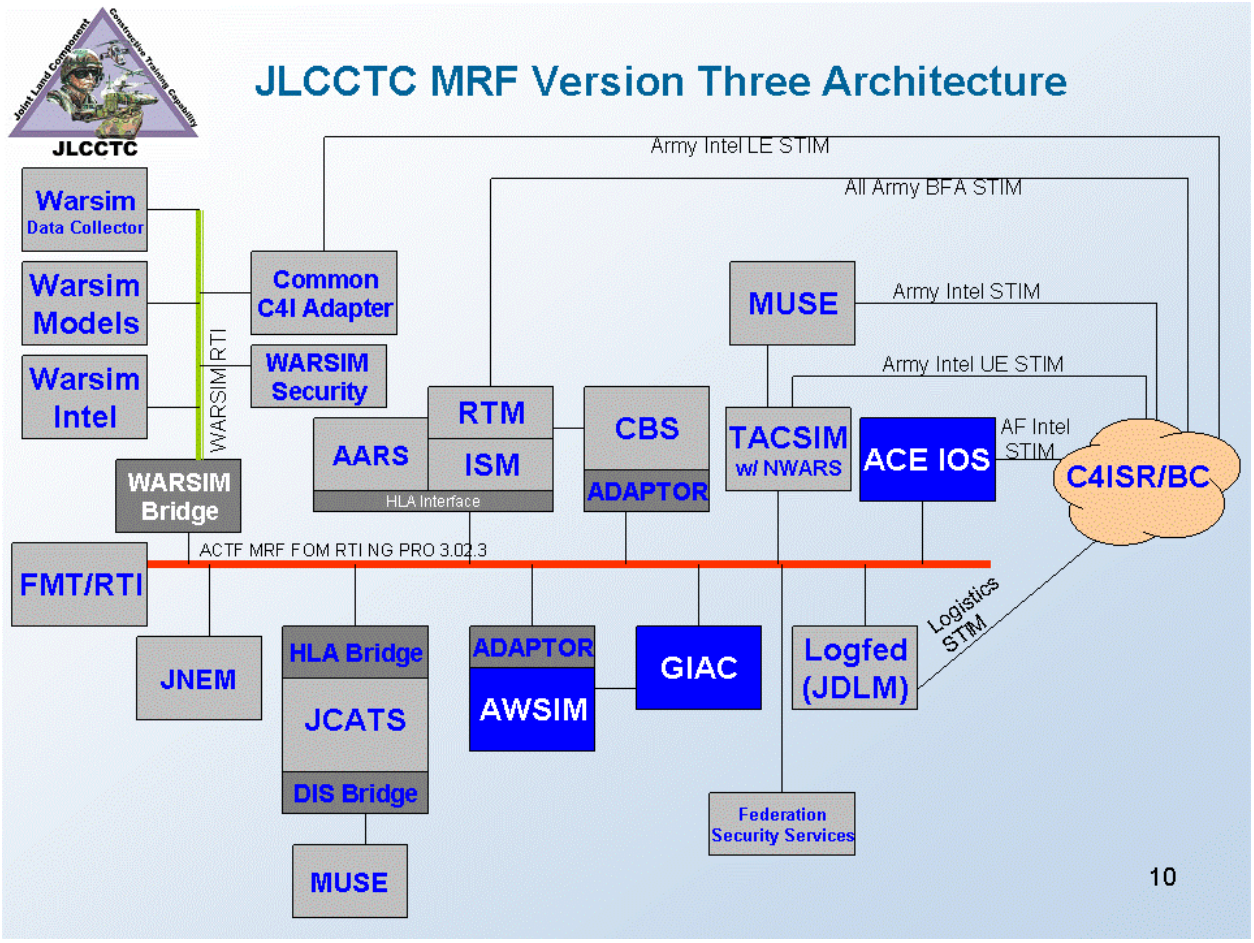

Figure 9 – JLCCTC MRF V3 Architecture

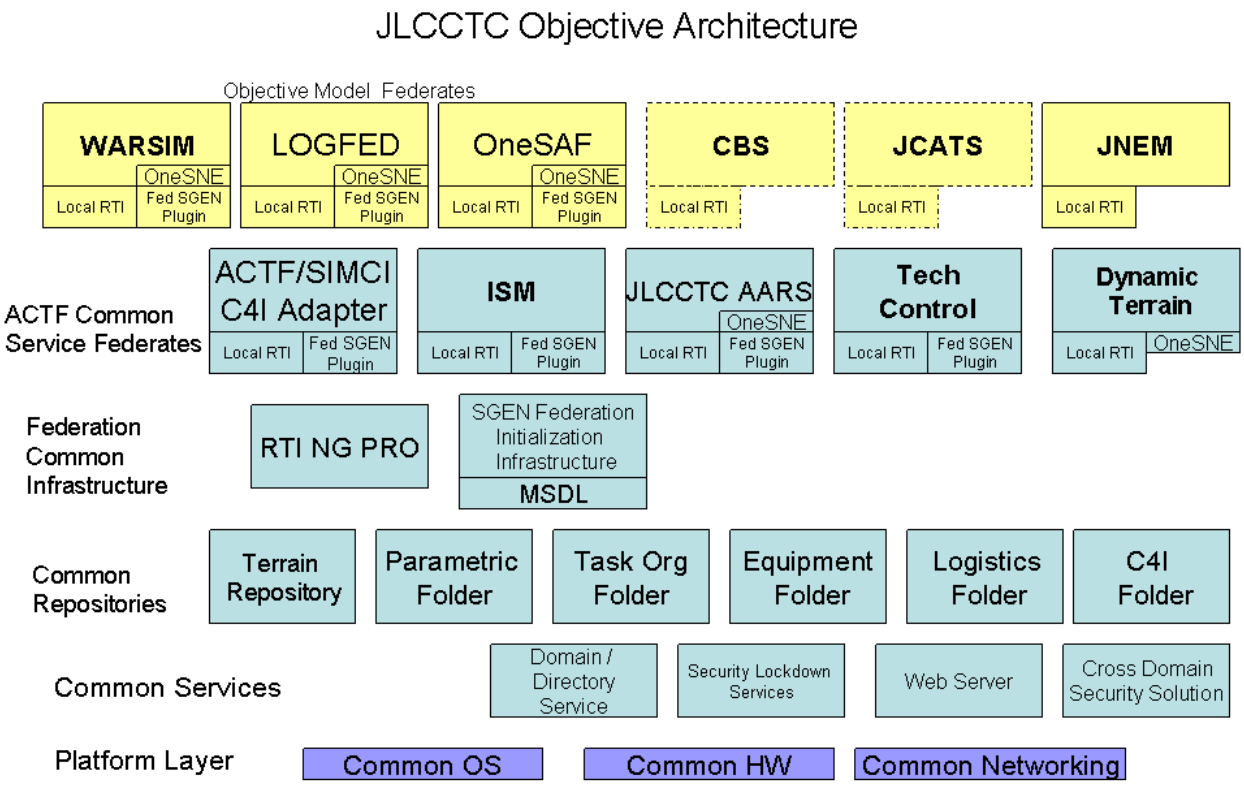

Figure 10 – JLCCTC Objective Architecture

5.4.3. Data Model Version:

5.4.3.1 ERF FOM: The JLCCTC ERF FOM is maintained and Configuration Managed for PEO STRI by the MITRE Corporation. The most current version of the ERF FOM as of 18 August 2006 is JLVC FOM 052506 and is available upon request in html or .omt format to Government agencies and qualified contractors. JLCCTC Systems Engineering POC is Craig Janisz, 407-384-5261 or DSN (970), craig.janisz@us.army.mil.

5.4.3.2 MRF FOM: The JLCCTC MRF FOM is maintained and Configuration Managed for PEO STRI by the MITRE Corporation. The most current version of the MRF FOM as of 18 August 2006 is ACTF_v3_FOM_020617 and is available upon request in html or .omt format to JLCCTC Systems Engineering. POC is Deanna Franceschini, 407-384-5394 (DSN 970), Deanna.Franceschini@us.army.mil

5.4.3.3 POC: Mike Wright (407) 384-3873

6.0 Processes / Guidelines

6.1. Process/Guideline ID: LT2 Style Guide

6.1.1. Process/Guideline Description: This Style Guide leads application developers in the design and implementation of Live Training Transformation (LT2) compliant applications, displays, controls and visual components. A common framework for Human Control Interface (HCI) design is provided, enabling all LT2 applications to appear and operate in a reasonably consistent manner. This style guide has also been coordinated with the OOS development team.

HCI design guidelines in the form of a style guide provide three major benefits:

- Higher productivity
- Reduced development time
- Reduced training time

6.1.2. Audience. The target audience for this style guide includes Application designers and User Interface System designers. Also included are program managers, system managers, other software developers, and those individuals who determine system requirements.

A secondary audience includes users and software maintainers who are interested in the general design of the interface, who wishes to provide feedback concerning modifications and improvements to the LT2 HCI Style Guide, or who wishes to assess the usability of applications in terms of their compliance with the style guide's content.

6.1.3. Design Goals. This Style Guide establishes criteria for the following design goals:

Increase a user's productivity by providing rapid access to all of an application's functions.

- Reduce a user's error rate by requiring explicit action to perform any act that could result in irreversible negative consequences

- Reduce education time required to master the application by helping the user transfer knowledge from existing product experience through the use of industry standard user metaphors

6.1.4. Process/Guideline Version: V2.1, dated 22 October 2004

6.1.5 POC: Will Samper (407) 384-3626

6.2 Architecture/Framework ID: OneSAF Product Line Architecture Framework (PLAF)

6.2.1 Architecture/Framework Description: The PLAF is a mechanism to organize, categorize, and define the layered software structure to incrementally meet the OneSAF requirements.

The PLAF identifies functionally relevant software components that can be used as building blocks for higher level functionality.

Within the Product Line Architecture Specification (PLAS), the PLAF provides a static view of the System Compositions, Products, and Components that comprise the OneSAF Architecture. See Figure 11.

The OneSAF Architectural approach facilitates meeting both current and future undefined requirements.

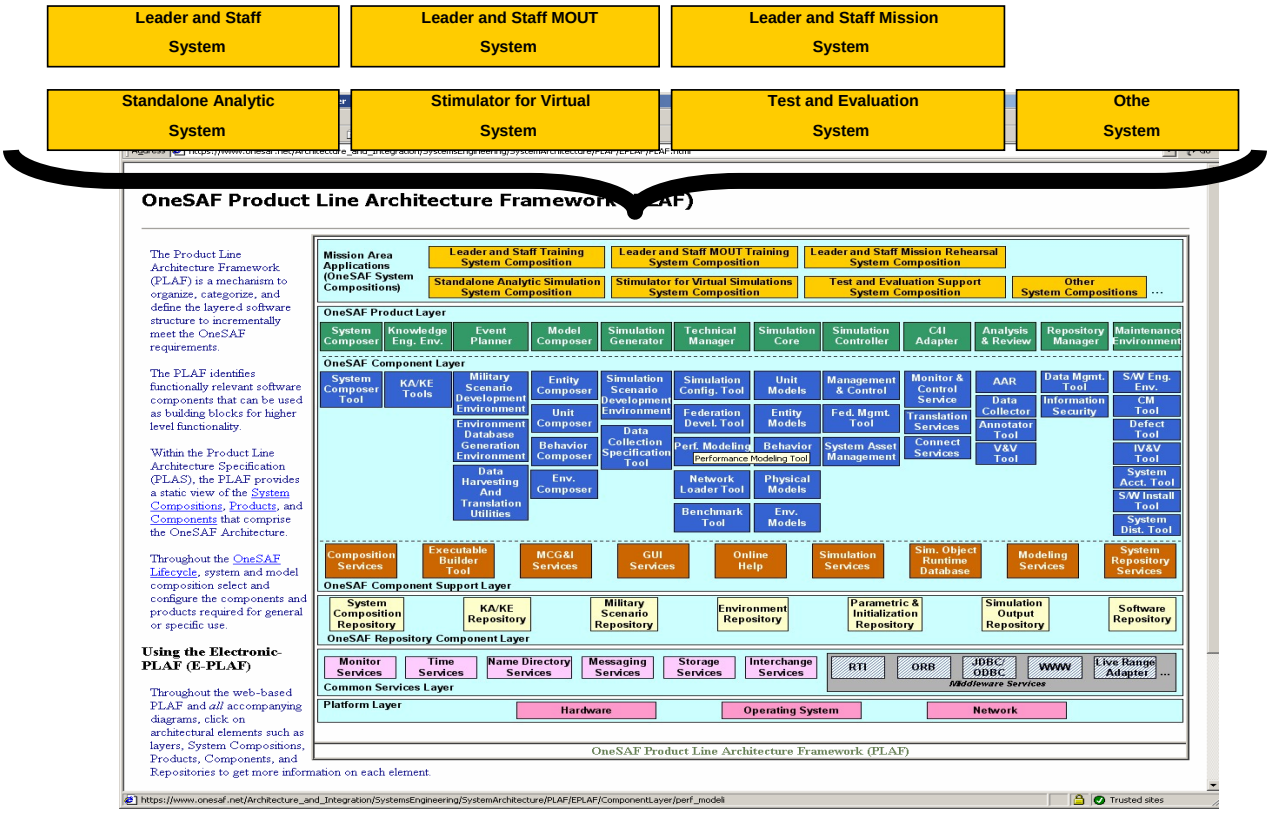

Figure 11 – OneSAF Architecture

6.2.2 Architecture/Framework Version: OOS BLK C UAB

6.2.3 POC: Doug Parsons (407) 384-3821

7.0 Repositories

7.1. Repository ID: Army Model Exchange (AME)

7.1.1. Repository Description: The AME is one component of the Virtual Targets Center, a cooperative effort between PEO STRI and the Aviation and Missile Research, Development, and Engineering Center (AMRDEC). The AME serves as a central repository for multi-resolution 3-D target geometry models. Models are available for immediate download in several industry standard formats from the protected website at https://modelexchange.army.mil.

7.1.2 Repository Version: https://modelexchange.army.mil

7.1.3 POC: Robbin Finley (256) 842-6459

8.0 Near Future Components

8.1 SE Core Standard/Rapid Terrain Generation Capability (STDGC)

Overview

The STDGC is intended to create a single unified process that supports the generation of all of the virtual and constructive databases required by confederate simulation systems. The STDGC will have two major functionality pieces; the first is the generation of a single unified master database (MDB) that is built at the highest level of data resolution possible from available government and commercial sources. This master database will constantly be updated as new data sources are acquired and as the geo-political climate changes. The second functionality piece is that of a database tailoring and formatting tool that tailors the MDB to the training objectives, systems capabilities, and run-times formats required by the confederate training systems.

Requirements

The STDGC has the requirement to generate databases that are 180 km x 180 km in size with a data resolution equivalent to NGA DTED level 3 (terrain surface resolution) and an urban inset within that database that is 2.5 km x 2.5 km with an equivalent resolution of NGA DTED level 5 (terrain surface resolution) to support MOUT/Urban operations. The MDB must be produced within 96 hours using COTS tools, open formats, and automated processes.

Implementation Concept

The overall implementation concept is shown in Figure 12. For the first part of the implementation the initial concept is to use COTS to generate the MDB. Conceptually, the MDB will consist of multiple open formats that facilitate a layered approach to the accessing and storage of the MDB. The MDB will be designed to accommodate data for the entire world but realistically it will only contain data for those parts of the world deemed important (e.g., home stations, training areas, areas of current and potential future military operations, other areas of interest). The MDB will also be maintained at the highest data resolution available from government and commercial sources. The MDB must also support current environmental data models (e.g., the OOS EDM).

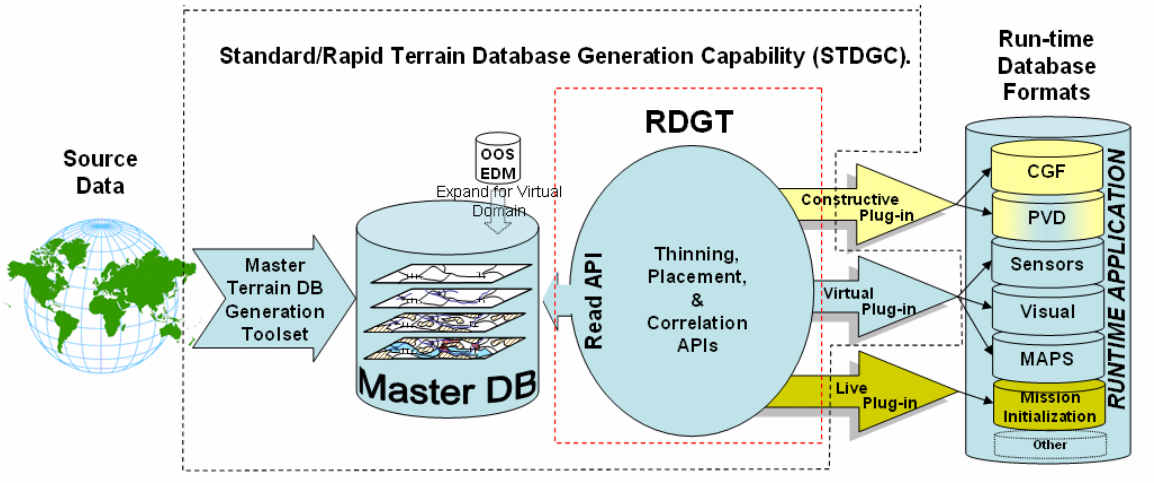

Figure 12 - STDGC Process Concept

The second part of the implementation involves the generation of the individual databases required for the confederate systems. For example, for the AVCATT system this would involve the generation of the visual and sensor databases in the L3 format, the OOS Semi-Automated Forces (SAF) databases and maps (electronic and paper). To achieve this, the conceptual implementation of the Real-Time Database Generation Toolkit (RDGT) would be to run off-line to create static databases in each of the required formats. The RDGT will have three major tasks: extraction of the data required for the training mission from the MDB, thinning, integration, and manipulation of the data to the training and system requirements, and finally formatting the data to the required format for the respective software application. The first task, extraction of the data from the MDB, will be through a government-owned API to facilitate the reuse and interchangeability of the data thinning, integration, and manipulation subroutines within the RDGT. The second task of thinning, integration, and manipulation will be controlled by a scripted process that resolves capability differences between differing simulation systems and provides correlated data to each simulation system in the confederation. The third task is the formatting of the correlated data to the individual simulation systems. To this end the government will develop and maintain an API for writing data to simulation systems. Individual system vendors will be responsible for developing software plug-ins that conform to this API and that will write the data into their individual database formats. These plug-ins will ensure the preservation of the data correlation and accuracy requirements and that the data is formatted and structured to work with their individual systems.

Other aspects of the STDGC concept include the automatic testing of the integrity of the MDB, distributed production facilities that provide local interaction with area commands in the generation of areas of the world, and alignment of the STDGC with other data initiatives within the military (ex. RD3, J-GES, PDI). Also, STDGC will support the generation of databases for FCS.

8.2 Common Information Assurance Products

8.3 Common After Action Review (AAR) and Data Collection

8.4 Instructor/Operator GUI

8.5 Scenario Generation/Development

8.6 Live Training Transformation Components

8.7 IDE Collaborative Environment