

LTEC: Enabling Componentized TESS Procurement using a Service Oriented Architecture

Matt Janisz, Phil Sowden
Applied Research Associates, Inc.
Orlando, FL
mjanisz@ara.com,
psowden@ara.com

Kyle Platt, James Grosse
U.S. Army PEO STRI
Orlando, FL
kyle.platt@us.army.mil,
james.grosse@us.army.mil

Gary Hall
General Dynamics C4 Systems
Orlando, FL
gary.hall@gdc4s.com

ABSTRACT

The U.S. Army currently performs live training with many different Multiple Integrated Laser Engagement System (MILES) training devices. The MILES Tactical Engagement Simulation (TES) hardware and software for these systems is vendor proprietary. This software provides the same general capabilities for each TES system including: conformance to the MILES Communication Code (MCC) Standard, battle damage assessment (BDA) feedback and exercise event storage for after action review. These proprietary systems do not allow the Army to interchange components between them or to ensure that the BDA is performed consistently across vendors.

This paper describes a service-oriented approach to solve these problems using the Live Training Engagement Composition (LTEC), an Army-owned architecture framework and set of composable software services. The Army plans to require the use of LTEC in future solicitations. The software is available through the Army's Live Training Transformation Portal. To enable reuse across TESS hardware configurations, LTEC abstracts operating system and communication protocol details from its services and it uses services to integrate with TES system components. The composability of the architecture also allows for easily configuring the capabilities provided. For instance, a low-fidelity BDA service could be swapped out for one that supports medic training or support for energy weapon engagements could be added to LTEC just by adding a service. By providing reusable TES software, LTEC reduces the cost to develop TES systems, ensures fair fight between those systems, and provides a path forward to embed/append common TES onto platforms and in future TES devices.

ABOUT THE AUTHORS

Matt Janisz is a senior software engineer at Applied Research Associates, Inc (ARA). He spent the last 8 years developing live training systems for the U.S. Army. Programs he's worked on include: LTEC, LT TES TCC, RPEL, AWES, and OneTESS. He earned his Master of Science (MS) degree in modeling and simulation and Bachelor of Science (BS) in Computer Science from the University of Central Florida (UCF).

Phillip Sowden is a staff software engineer at ARA. He has 6 years of experience developing live training systems for the U.S. Army. Programs he's worked on include: LTEC, LT TES TCC, RPEL, AWES, and OneTESS. He earned a BS degree in Computer Engineering from UCF.

Kyle Platt is a Project Director for PM Live Training Systems within the U.S. Army's Program Executive Office for Simulation, Training and Instrumentation (PEO STRI). He has worked various Army programs such as OneTESS, A-TESS, T-IS, PM TRADE Strategic Requirement Integration (SRI), AVCATT and has also served as the LT2 Framework Architect. Mr. Platt earned both a BS and MS in Aerospace Engineering from the University of Central Florida (UCF) with a primary focus on power generation and microwave electro-thermodynamics.

James Grosse is the Chief Engineer APM Training Devices within PEO STRI. He has over 25 years experience leading engineers in the development of tactical engagement simulation systems and in managing the development of Live, Virtual, and Constructive simulation systems for the U.S. Military.

Gary Hall, senior lead engineer at General Dynamics C4 Systems, has spent the last 22 years developing live, virtual and constructive training systems for the U.S. Army and Navy. He has been the technical lead in the development of embedded training. He earned an MS degree in Computer Engineering from UCF.

BACKGROUND

In the Live training domain, the Project Manager Training Devices (PM TRADE) has been lifecycle managing Force-on-Force (FoF) Tactical Engagement Simulation Systems (TESS) for over 30 years and has fielded over 267,000 systems worldwide. The various TESS systems acquired range from instrumented and non-instrumented dismount kits through highly complex, digitally integrated instrumented tank kits. The Army is now mandating an embedded training capability on its newest digital weapons' platforms. From an interoperability perspective, the only thing that has remained relatively constant is the Multiple Integrated Laser Engagement System (MILES) Communication Code (MCC) Standard. The MCC Standard defines a laser communications protocol to send engagement data between shooter and target, enabling FoF Brigade and below Live collective training at Homestation and Combat Training Centers (CTC).

The constantly growing number of TESS devices has led to an exponential growth in sustainment cost as multiple hardware and software baselines must be maintained and supported for 20+ years. This coupled with Under Secretary of Defense for Acquisition, Technology and Logistics (AT&L) Mr. Frank Kendall's recent update to the Better Buying Power initiative to "Emphasize competition strategies and create and maintain competitive environments," has required a better means of designing, developing, integrating, and lifecycle managing TESS.

TESS ACQUISITION PROBLEM

In a traditional TESS acquisition, PM TRADE and its Product Manager for Live Training Systems (PM LTS) issues contracts with an intent to purchase a full TESS kit. This kit fulfills the need of the Instrumented MILES (I-MILES) product lines: Individual Weapon System (IWS), Tactical Vehicle System (TVS), Combat Vehicle TESS (CV-TESS), Universal Controller Device (UCD), Shoulder Launched Munitions (SLM), Aviation TESS, Improvised Explosive Device Effects Simulation (IEDES), One Tactical Engagement Simulation System (OneTESS), and in other circumstances one of the various customer funded TESS devices. PM TRADE has developed a new FoF reference architecture using a modern architectural modeling tool to drive the solutions it is acquiring. Before the development of this architectural model to drive the performance of a system, each system was never afforded the reusability and commonality benefit of the advertised product line. Rather these TESS devices were treated more like disparate systems with two key points of interoperability: The Instrumentation System (IS) radio via the IS-TESS Standard and the MILES protocol via the MCC Standard. Since most of these programs are in the post deployment/sustainment phase of the acquisition cycle, access to paradigm shifting technology insertion essential Research & Development dollars is limited. As a result, these systems continue to be acquired with limited interoperability and commonality; yielding skyrocketing stove-piped sustainment bills.

Since the demise of the Future Combat System (FCS), the Army has revitalized their Live, Virtual and Constructive (LVC) embedded training initiatives (PEO STRI, 2011). To support the embedment of Live Training, key software products, strict architecture and interface management are the key to ensuring resource informed outcomes, emphasizing reusability and reduced support and sustainment costs across the future of embedded and appended TESS. The TESS of the future will need to be componentized so that different compositions can be fielded to support new digital platforms like the Ground Combat Vehicle and their embedded training requirements. These platforms may require software embedded on the vehicle and specific capabilities (e.g., laser detectors or transmitters) appended on the outside of the vehicle. Currently, there is no software product line that manages any TESS software to allow for reuse across multiple platforms or dismount applications.

PROBLEM SIMILARITIES WITH EXCON

The challenges mentioned above for addressing embedded training and the skyrocketing sustainment bills have been seen across other areas within the Live domain.

Over a decade ago the TRADOC Capabilities Manage – Live (TCM-Live) and PEO STRI examined their approach for developing instrumented training capabilities at the CTCs and Homestations specifically focused on the Exercise Control (EXCON) functions. There was a realization that each time the Army replaced or updated any of these systems with new capabilities that it would spend tens of millions of dollars to produce, field and sustain them. This occurred across the Army's Live training ranges and CTCs and was becoming unaffordable and very difficult to configuration manage. The same set of requirements for data collection, after action reviews and exercise

management were being implemented with contractor/site specific software and hardware systems. To address this issue in the Live domain, the Army through PEO STRI developed what has come to be known as the Live Training Transformation (LT2) Common Training Instrumentation Architecture (CTIA). PEO STRI developed a product line approach to this challenge where the Government controlled software development, implementation, and configuration management. CTIA became the first set of capabilities under the LT2 Product Line developed for the Live community where software applications could be reused across multiple systems. In addition a set of governance protocols, interface documents and standards have been developed by the Live community to ensure commonality of components and maximization of reuse (PEO STRI, 2008). This approach has saved the Live community over \$400 Million via cost avoidance over the past 6 years (Dillon, 2012).

This EXCON Product Line approach has been so successful that external organizations have utilized and become part of the process. The US Marine Corps and Air Force have reused many of the software common components to address their Live training requirements. Additionally the Army's Testing community is performing an analysis of the LT2 Product Line to determine how it can be applied for their unique testing requirements.

SOLUTION - COMPONENTIZE THE TESS PROCUREMENT PROCESS

The future of FoF TESS is the Army Tactical Engagement Simulation System (A-TESS). A-TESS will have a foundation in architecturally driven hardware and software products, coupled with LT2 governance, processes and policies, and modernized legacy TESS. A-TESS systems will be driven by strict interface management and commercial and Government standards.

As future tactical systems are developed, FoF TESS will move to a hybrid environment where appended and embedded devices will be used on the same platform. Tactical platform and dismounted PM may choose to fully embed a TESS capability, partially embed or even choose to not embed a TESS capability at all; in which case PM TRADE will still need to acquire appended TESS. To be fiscally responsible and support fair-fight between embedded, partially embedded and pure appended future TESS, PM TRADE is attacking the problem from two different directions: top-down with the LT2 TESS Architecture and bottom-up with the Live Training Engagement Composition (LTEC).

The LT2 TESS Architecture is a DoDAF reference architecture, written in the Unified Profile for DoDAF and MoDAF (UPDM) architecture development language. The LT2 TESS Architecture was completed in a Commercial Off The Shelf (COTS) Model Based Systems Engineering (MBSE) environment. This afforded the Government the ability to link PM TRADEs LT2 TESS Architecture to the LT2 Instrumentation System (IS) – the Common Training Instrumentation System Architecture (CTIA), Future Army System of Integrated Targets (FASIT), Live, Virtual and Constructive Integrating Architecture (LVC-IA) and to the Army, Department of Defense and Federal Enterprise Architectures. This integrated architecture affords PM TRADE the ability to not only instill architecture driven interfaces to the RFPs it solicits, but also supports high fidelity, resource informed, impact assessments as changes and capability insertions are made to the PM TRADE live training portfolio. Strict interface and architecture management will allow PM TRADE to componentize its TESS acquisitions, promoting competition and reducing time to market, while also growing an emerging industry base, which is lean and highly productive. Componentization is not only critical to the reduction of support and sustainment cost for the future of appended TESS; it also affords the Government agility with its procurements to support emerging hybrid-embedded live training, where by a tactical platform PM may only be interested in purchasing a Combat Vehicle Kill Indicator (CVKI) because they have embedded everything else. Another example of this approach is where a small arms laser transmitter from one contractor system can be competitively procured and potentially awarded to a different contractor – today PM TRADE is forced to sole source a contract to the Original Equipment Manufacturer (OEM).

Although PM TRADE can use this new architectural modeling tool to determine performance requirements for its componentized TESS devices, a furthering of the LT2 Product Line to include the TESS software was needed. Taking advantage of the cost avoidance and the lessons learned over the past 10 years through the development of the CTIA and the LT2 software product line, PM TRADE has architected a complimentary software product line, LTEC. The vision of LTEC is to reduce development cost, promote technology insertion and lower testing/integration risks, while also providing PM TRADE the ability to realize cost avoidance not afforded by stove-piped FoF TESS acquisitions. LTEC will promote interoperability at the software service level and was designed using a Service Oriented Architecture (SOA) approach. This organically promotes custom composable

applications and discounts un-needed and un-desired bloat. LTEC is at the heart of the Army's approach to embedded Live training and is the soul of the future of live FoF appended training.

ARCHITECTING THE LTEC SOFTWARE SOLUTION

The requirements for the LTEC software architecture that satisfy PM TRADE's TESS procurement problem cover both functional and non-functional (quality attribute) requirements. The functional requirements are problem domain-specific, however the quality attributes may be applicable for procurement problems in other domains if their use cases are similar.

Functional Requirements

At a high-level, the currently implemented and planned functional requirements for the architecture are to provide the MILES shooter and target system functionality provided by a generic MILES (TES) Control Unit or TCU that represents a single entity (vehicle, dismount, or crew-served weapon) in the live training simulation.

The general MILES capabilities provided by the TCU include: receiving of decoded MCC data from a MILES detector device, routine decoding of the MCC data, performing the Lethality Effects Assessment Routine (LEAR), MCC administrative command processing, stimulation of visual and audio cues in response to simulated events, stimulation of a MILES transmitter to simulate weapon fire, and data exchange with the Instrumentation System (IS) radio for command, configuration, and event reporting. Most of the TCU capabilities are defined in the MCC Standard (PEO STRI, 2011). The interface between the TCU and the IS radio is defined in the IS-TESS Standard (PEO STRI, 2012).

Since LTEC must support an assortment of different entities which share a significant overlap of MILES capabilities, there is also a requirement for LTEC to provide common software that is entity type agnostic. Not only does this requirement make sense from a software reuse perspective, but the common software also provides consistent behavior for all MILES systems leveraging LTEC. Thus, common software addresses a major issue identified in the problem description.

Also as noted in the problem description, existing MILES sub-system interfaces are all vendor-unique and are not interchangeable even for systems of the same entity type. Therefore, to facilitate supporting integration with vendor-unique subsystems and to support embedded training, LTEC has a requirement to integrate with specific training or tactical system hardware configurations using adapter software. Ideally, the adapter software uses a standard, such as the Live Player Area Network (PAN) Standard (PEO STRI, 2012) or Vehicular Integration for C4ISR/EW Interoperability (VICTORY) Standards (VSSO, 2012), to integrate with training or tactical system hardware.

Quality Attributes

The quality attributes for the LTEC architecture are driven by three typical business goals: streamline business processes to reduce operating costs, enable easy and flexible integration with various hardware configurations, and adapt quickly to new and changing system requirements. Supporting new requirements was not an original customer requirement, but was made a business goal due to our previous experience on OneTESS (ATSC, 2004) and the knowledge of the customer's long-term goals for A-TESS (ATSC, 2013).

For LTEC, implementing the second goal helps enable achieving the first, so there is significant overlap between the quality attributes that support those goals. The quality attributes supporting those goals include: interoperability, modifiability, adaptability, and performance.

Interoperability supports these goals by allowing the Government to procure TESS as individual components vs systems with well-defined and Government-controlled interfaces and software. Modifiability will allow the TESS vendor base and the PMs opting to embed TESS functions on their platforms the flexibility to tweak the software to meet their specific need or operating environment. Adaptability means that new systems will be able to reuse the LTEC software without having to develop extensive software routines to integrate the applications onto their devices. The performance quality attribute supports the goals by ensuring operation on a variety of platforms ranging from simple processing devices (IWS) all the way through integration on the digital GCV platform.

In addition to those quality attributes, the third goal adds the needs for extensibility and testability. Extensibility supports this goal by allowing the Government to add future TESS capabilities without affecting existing capabilities. Testability supports this goal by reducing the risk involved in updating the capabilities provided by LTEC. Through automated tests that verify both system and service-level requirements, the Government will be sure that adding new capabilities does not break existing capabilities.

Most of the quality attributes are positively impacted by design principles inherent in a SOA approach. Some of these principles that were identified by O'Brien, Bass, and Merson (2005) include: services are reusable, services share a formal contract, services are loosely coupled, services abstract underlying logic, services are composable, services are autonomous, and services are stateless.

Reusable, encapsulated, and autonomous services with formal contracts support interoperability, adaptability, and extensibility by allowing for rapid generation of new service compositions to facilitate requirement changes. Extensibility is also supported via a run-time configured common data model used by all services to communicate. Loose coupling and statelessness of services support modifiability of the architecture.

While SOA is good for some of the quality attributes, it negatively impacts others. These include testability, performance, and adaptability. Testing SOAs can be difficult if the services are distributed across a network. In addition, testing services provided by external organizations that are independently developed can be difficult without source code. Very little support is currently provided for end-to-end testing of distributed SOAs. In addition to negatively impacting testability, the nature of networked communications between services also negatively impacts performance. Performance is also negatively impacted by service look-up (also network related) and adding middleware to translate between services. The SOA issue with adaptability is not inherent, but is a potential issue if well-defined governance processes are not followed in managing service development as noted by O'Brien, Bass, and Merson (2005).

The issues with adaptability are addressed by requiring the governance process laid out by the Consolidated Product-Line Management Program for developing new services. In regard to adaptability, this process includes an upfront review and approval of the proposed service's requirements and design and a verification step after the service is implemented to ensure the original requirements and design were met.

The issues with performance are addressed via a tradeoff with service location flexibility. Given the current LTEC use cases, the architecture does not require services be network-addressable. Instead, services communicate via a publish/subscribe mechanism using application programming interface (API) calls to a service message broker that keeps services loosely coupled. Performance is also improved by requiring all services to use a common data model that all services natively understand.

The API for message communication also facilitates addressing the issues with testability. It supports implementing a service testing framework that can stimulate a single service with inputs and verify the expected service outputs in an automated fashion. Thanks to the composability provided by a SOA, the same framework can also be used to verify compositions of services in the same manner.

LTEC ARCHITECTURE AND SERVICES

Evolution of the LTEC Architecture

LTEC began as a common BDA calculator for an embedded MILES TESS. That is, LTEC would be a government owned, common BDA library that embedded TESS developers could reuse. The idea was to provide BDA commonality among training systems, enable fair fight, and reduce procurement cost for new training systems. In addition to providing common BDA, the original LTEC vision also included communication with sensor and stimulator systems. For example, LTEC would receive laser detection data from a MILES laser detector and command a vehicle's kill indicator to flash upon assessing a kill.

The preliminary design of LTEC consisted of a core, which contained the BDA routines and all business logic, and device adapters. The device adapter was a runtime-loaded library (plug-ins) that handled the low-level details for communicating with a particular device. A device adapter integrated a training device with LTEC core.

Upon further analysis, it was identified that interfacing with a training device involves three distinct parts: the messaging protocol, the transport mechanism (transport layer and below in the OSI Model), and the operating system on which LTEC is running. Additionally, the conclusion was reached that none of those three parts are necessarily unique to a device, since more than one device adapter may need to use any of them. To facilitate reuse and reduce device adapter development time, it was decided that LTEC core should provide support for operating system agnostic communication using a particular transport mechanism and a messaging protocol defined in an LT2 Standard, like the LT2 Live PAN Standard. This resulted in the Operating System Abstraction Layer (OSAL) and Communication Abstraction Layer (CAL) packages in the LTEC Framework.

In the preliminary architecture, LTEC core consisted of common BDA routines and message routing between device adapters and BDA routines. While analyzing the architecture, the realization occurred that the BDA routines were standalone software components that could be interfaced with through the same messaging mechanism used between LTEC core and the device adapters. To provide modularity and composability it was decided that the BDA routines should be implemented as plug-ins. This allows for updates to existing BDA routines or replacement with new BDA routines without affecting the rest of the system.

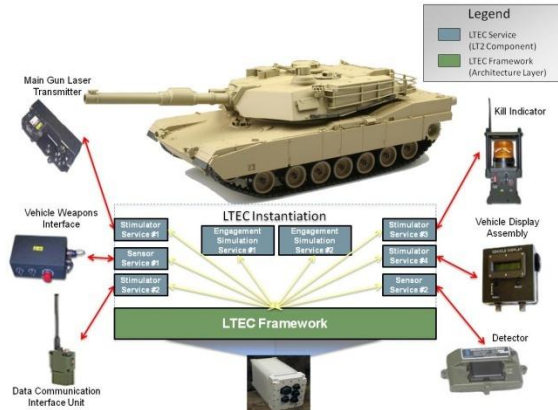


Figure 1. Notional Instantiation of LTEC Architecture

the LTEC Framework for service infrastructure and abstraction from the operating system and from communication protocols. As such, the LTEC Framework consists of three subcomponents: Service Infrastructure, OSAL, and CAL.

Service Infrastructure

Service Infrastructure is responsible for managing services. Service libraries that are in the designated service library folder will be loaded, initialized, and run.

Additionally, Service Infrastructure routes messages between services. These messages are defined in the LTEC data model. The data model is defined in an XML file. LTEC messages may provide the information to calculate engagement results, display effects, or modify behavior. The communication follows a local publish and subscribe messaging pattern via API calls. A service subscribes to specific message types when it is loaded. Any service may publish any message type. By using this type of messaging pattern, services can publish and subscribe to any of the messages defined in the LTEC data model. This gives system developers flexibility in how they configure services to provide functionality. It also reduces the complexity to integrate with new devices by providing a reusable messaging interface.

As a result of these architecture changes, LTEC became a SOA that consists of a collection of services that either perform training simulation or communicate with devices, and these services are supported by a framework that facilitates communication between services.

High Level Architecture

Figure 1 shows a notional concept diagram for LTEC running on a vehicle and integrated with typical live training devices. As shown in Figure 1, an LTEC instantiation consists of a set of services. Each service provides an engagement simulation capability, integration with a sensor device, or integration with a stimulator device that is required for live training. These services depend on

OSAL

OSAL abstracts operating system APIs from the other LTEC software units. This layer allows LTEC Core and the services to be ported to new operating systems by replacing only the implementation of the operating system abstraction layer. OSAL includes support for multithreading, file input/ output operations, and communications.

CAL

CAL abstracts the communication protocols of a device from the service that interfaces with the device. The communication protocols implemented by CAL encompass all of the layers of the OSI Model except for the application layer. This includes both the message protocol and the transport mechanism. CAL uses the OSAL package to make OS-specific API calls for communicating through a particular transport mechanism. Ideally, the communication protocols are defined in a standard, such as the LT2 Live PAN Standard.

Simulation Services

Simulation services provide all business logic and BDA routines for an LTEC instantiation. These services are reusable across different LTEC compositions for different TESS use cases. For instance, the same BDA service would be reused across all LTEC compositions supporting the vehicle TESS use case whether it is an appended, embedded, or a hybrid hardware solution.

Sensor/Stimulator Services

Sensor/Stimulator services are services that communicate with external devices. These external devices can be categorized into devices from which data is obtained (sensors) or devices that receive commands (stimulators). Sensor/stimulator services reuse LTEC Framework libraries, OSAL and CAL, to communicate with devices. Every sensor/stimulator service converts between LTEC Framework messages and external communication protocol messages. For example, a detector service receives detection information from the device, converts it to an LTEC message and publishes it to the LTEC Framework. Additionally, a sensor/stimulator service implements the application layer of a communication protocol.

A sensor/stimulator service integrates with the supported device in accordance with (IAW) the device's ICD as shown by the red arrows in Figure 1. The service also integrates with the engagement simulation services IAW the interface and messages defined in the LTEC data model. By implementing both interfaces, the service integrates LTEC with the supported device.

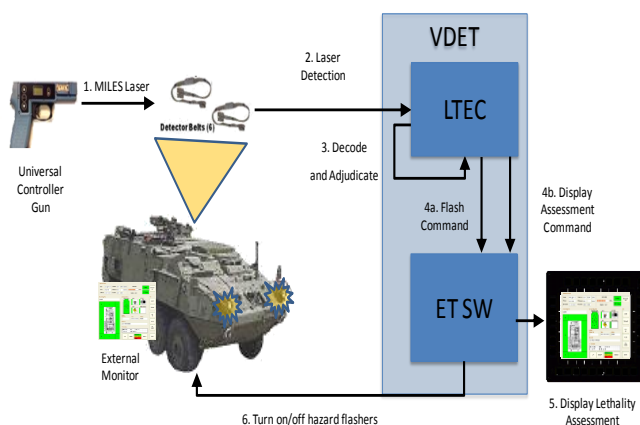


Figure 2. Stryker Appended Vehicle Use Case

detector belts on a Stryker vehicle and executing LTEC software components on the vehicle's embedded computer hardware to perform the adjudication and provide user feedback of MILES laser engagements as shown in Figure 2.

LTEC USE CASES

Several compositions of the LTEC software have been or are currently being implemented and demonstrated. Each implementation has its own composition of LTEC services and provides verification of the LTEC Framework and corresponding services.

Appended Stryker Vehicle Use Case

The first LTEC composition and use case culminated in an appended vehicle (Stryker) training demonstration to several key vehicle PMs and TRADOC Capability Managers (TCMs). The use case involved appending MILES XXI laser

During the demonstration, the Stryker vehicle was engaged by an enemy combatant role player who fired a MILES Shoulder Launched Munitions (SLM) laser at the Stryker vehicle. The LTEC software running inside the Stryker constantly checked the status of the laser detectors to determine if a hit had occurred. Upon receiving the hit detection, the LTEC software ran the laser information through the common BDA algorithms to determine if the vehicle was hit and what type of damage was received. The LTEC software then communicated the results of the BDA to the Stryker Embedded Training software running on the Stryker's Vehicle Data Electronic Terminal (VDET). The LTEC software then updated the health status on the VDET, played an audio cue related to the damage received, and flashed the hazard lights to indicate the appropriate BDA. By flashing the hazard lights, this demonstrated the concept of dual use hardware for both tactical combat operations as well as use in embedded training. Although the dual use of vehicle hazard lights might not be the best example, it demonstrated how utilizing native hardware would save money both in terms of actual procurement of appended training hardware as well as the manpower costs associated with setting up and maintaining the appended devices on the vehicle while it is being used in training.

The LTEC software composition, in addition to the LTEC Framework (including OS abstraction layer), to support this demonstration included the following key services:

- *MILES Effects Calculator Service* - receives a MILES routine decoding outcome (McCHit or McCNearMiss), determines the effects from the event, and sends out a MILES assessment indicating the effects to other LTEC Services.
- *MILES Routine Decoder Service* - contains routine decoding functionality as defined in the MCC Standard.
- *MILES XXI Detector Service* - processes MILES XXI System Bus messages from the MILES XXI Vehicle Detection System (VDS) to receive MILES XXI formatted laser detection data and provide out LTEC formatted MILES laser detection data.
- *PAN Display Service* - receives a MILES assessment and creates a display message based on the assessment. The message is put into a PAN Remote Display Message and sent over a TCP socket via the Communication Abstraction Layer (CAL) to a PAN client user interface device.
- *Visual Cue Service* - receives MILES assessment and generates the appropriate visual indicator command IAW the Audio/Visual (A/V) Cue Standard. The visual indicator command, a.k.a. FlashCommand, is sent out to other LTEC Services.

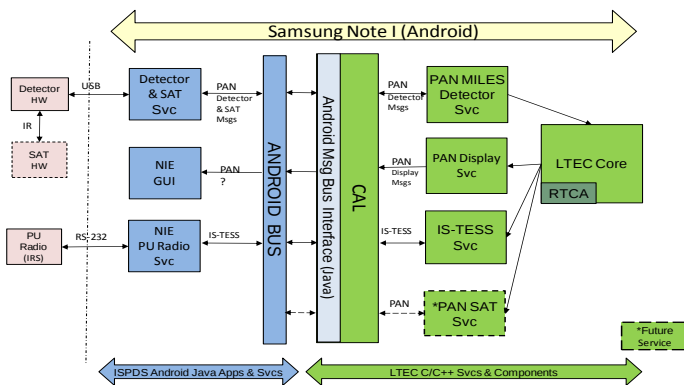


Figure 3. LTEC Embedded Dismount Services and Interfaces

and appended. For the embedded case, the soldier is equipped with a tactical harness which provides an embedded power and data hub with embedded processing capability such as a handheld smart device. This harness allows the connection of future dual use embedded sensors that would provide tactical (Combat ID) and live training (MILES) capabilities. In this use case, LTEC components would run along with tactical applications on the smart device running the Android O/S for example. This capability has been demonstrated on a normal smart device (Samsung Note I) and interfaces with a prototype embedded MILES laser detector system in preparation for a future Network Integrated Evaluation (NIE) event. The LTEC software services and interfaces composed in this configuration are shown in Figure 3.

The second dismount use case is the appended use case. In this configuration, the Dismount Soldier is equipped with a MILES training harness with runs internal LTEC software components. The services composed in the

out to other LTEC Services.

- *PAN Visual Cue Service* - receives an Audio/Visual (A/V) Cue Standard visual indicator command and reformats it into a PAN Strobe Flash Message. The message is then sent over a TCP socket via the Communication Abstraction Layer (CAL) to a PAN client strobe device.

Dismount Embedded/Appended Use Case

For the Dismount Soldier use case, two implementations have been defined, embedded

appended configuration are the same as the embedded use case. In addition to the services described in the appended vehicle use case, the dismount implementation includes a PAN MILES Detector Service, a PAN Display Service, an Instrumentation (Radio) Interface Service, and a PAN Small Arms Transmitter (SAT) laser transmitter service.

Vehicle Embedded Use Case

For the embedded vehicle use case such as what is targeted for the future GCV a different composition of services is required. LTEC SW would be running on a platform's existing computing resources, communicate over its tactical vehicle bus interfaces and leverage dual use vehicle sensors such as laser detectors, GPS receivers, and laser ranging transmitters to support training exercises without appending specific training devices. New services are necessary to tie into future vehicle bus standards such as the VICTORY data bus that will be part of GCV program. These services are currently planned to be implemented to support future programs such as GCV. The new key LTEC services that will be required for a complete embedded training capability are:

- *Detect Weapon Fire Service*- detects that a TES-equipped weapon system has been fired via the VICTORY data bus.
- *Simulate Weapon Fire Service*- receives weapon trigger pull and weapon information and performs ability to fire functions such as remaining rounds, breach and safety settings and misfire/jammed simulations.
- *Transmit Simulated Weapon Fire Service* - provides capabilities to broker engagement parameters and build engagement parameter messages.
- *Fire Stimulator Service* - provides the capability to send stimulator events to appended or organic weapon systems, depending on the vehicle configuration
- *Time-Space-Position Information (TSPI) Device Service*- interfaces with the physical device(s) that provide TSPI data (e.g. a Global Positioning System (GPS) chip or a vehicle bus) and will make that data available to the other LTEC services
- *TSPI Provider & Reporting Services*- manage the TSPI capabilities for all TSPI Device Services associated with a participant and, based on the configured update rate, passes that data to any consumer that needs the data pushed at a set interval such as reporting position for after action review (AAR)
- *VICTORY Audio Cue Service* - provides the capabilities to send audio cues to vehicle intercommunications devices via the VICTORY data bus.
- *Training Performance Data Service*- provides capabilities to build After Action Review (AAR) data based on engagement parameters (e.g. TSPI, Battle Damage Assessment (BDA) outcome, etc).
- *Data Persistence Service*- provides capabilities to persist data such as AAR events, engagement events, and configuration information for later retrieval.
- *User Manager Interface Service*- provides capabilities to display and retrieve information from user interface devices
- *Mounted Soldiers in MILES Effects Calculator Service*- determines the effects from the engagement event, and sends out a MILES assessment to support adjudicating damage to mounted dismount soldiers.

CONCLUSION

The traditional procurement strategy for the Army's FoF TESS fleet, consisting of multiple kits from multiple vendors, has led to an exponential growth in sustainment costs for the U.S. Army. These kits have issues with interoperability and do not support component interchange due to a lack of well-defined interface standards. In addition, the procurement strategy forces PM TRADE to pay multiple times for the development of the software for the TESS, when there is significant overlap in the capabilities between the TESS kits. This lack of common software also allows room for fair-fight issues due to behavioral differences in the vendor-owned TESS software.

PM TRADE successfully tackled a similar problem with their EXCON software. They developed the LT2 CTIA product line approach that allowed them to reuse common software across multiple instrumentation systems. The

product line approach has realized the Live Training community over \$400 Million in cost avoidance, improved the maturity of the product, and reduced the time to develop and deploy new capabilities.

This same approach is driving the development of the LTEC software. The LTEC software was architected to provide the existing TESS capabilities for all kits, while meeting PM TRADE's business goals. The architecture has been proven out through demonstrations of dismount and vehicle use case implementations. Further, the architecture readily supports adding future TESS capabilities as PM TRADE moves toward the development of A-TESS. But, most importantly, LTEC enables the componentizing of the TESS procurement process by leveraging open, Government-owned, standard interfaces in LTEC and using a well-defined software development governance process.

REFERENCES

- Program Executive Office for Simulation, Training, and Instrumentation (PEO STRI). (2011). *Vision for Embedded Training*. Retrieved May 18, 2013, from Live Training Transformation (LT2) Portal: <http://www.lt2portal.org>
- Rivera, J., Samper, W., & Clinger, B. (2008). Live Training Transformation (LT2) product line applied standards for reusable integrated and interoperable solutions. In *Military Communications Conference, 2008. MILCOM 2008. IEEE* (pp. 1-8). IEEE.
- Dillon, M., Rivera, J., Darbin, R., & Clinger, B. (2012). Maximizing US Army Return on Investment Utilizing Software Product-Line Approach. In *The Interservice/Industry Training, Simulation & Education Conference (I/ITSEC)* (Vol. 2012, No. 1). National Training Systems Association.
- Program Executive Office for Simulation, Training, and Instrumentation (PEO STRI). (2011). Multiple Integrated Laser Engagement System (MILES) Communication Code (MCC) Standard. PMT 90-S002 Rev M.
- Program Executive Office for Simulation, Training, and Instrumentation (PEO STRI). (2012). Instrumentation System (IS) Tactical Engagement Simulation Systems (TESS) Interface Standard. PRF-PT-00552 Rev C.
- Program Executive Office for Simulation, Training, and Instrumentation (PEO STRI). (2012). Live Player Area Network (PAN) Standard. PRF-PT-00549 Rev C.
- VICTORY Standards Support Office (VSSO). (2012). Vehicular Integration for C4ISR/EW Interoperability (VICTORY) Standard Specification. Version 1.4.
- U.S. Army Training Support Center (ATSC). (2004). One Tactical Engagement Simulation System Operational Requirements Document.
- U.S. Army Training Support Center (ATSC). (2013). Army Tactical Engagement Simulation System Capabilities Description Document.
- O'Brien, L., Bass, L., & Merson, P. (2005). *Quality Attributes for Service-Oriented Architectures* (CMU/SEI-2005-TN-014). Retrieved May 11 2013, from the Software Engineering Institute, Carnegie Mellon University website: <http://repository.cmu.edu/sei/449>