Employing Service Orientation to Enable Training as a Service in the U.S. Army

Jeremy T. Lanman, Ph.D.
Project Manager Training Devices
United States Army PEO STRI
Orlando, FL, USA
Jeremy.Lanman@us.army.mil

Panagiotis K. Linos, Ph.D.
Dept. of Computer Science & Software Engineering
Butler University
Indianapolis, IN, USA
Linos@butler.edu

***Abstract*—Currently many training systems acquired, fielded and sustained by the U.S. Army are unable to seamlessly comply with a continuously evolving and often complex computational environment. The current state of such training systems must advance to a Training as a Service (TaaS) future state in order to adapt to a volatile defense budget, conform to policy updates, and enhance the training capabilities afforded to the Warfighter. TaaS will transform current Army training applications into distributed web-based services, allowing them to be accessible across any location via thin client workstations and wireless mobile devices. The motivation behind this migration is coming from the Common Operating Environment (COE) Architecture Guidance published by the U.S. Army Chief Information Officer and the Assistant Secretary of the Army for Acquisition, Logistics, and Technology. In order to achieve the COE objectives, the Army recently launched a pilot study on the Common Training Instrumentation Architecture (CTIA). CTIA is the foundation architecture of the Army's Live Training Transformation (LT2) Product Line that provides software infrastructure and services to live training product applications.**

This paper describes the migration of CTIA to TaaS using a specific set of modern computing technologies that will enable rapid delivery of training capabilities across servers, mobile devices, and heterogeneous platforms. Specifically service-oriented architecture (SOA) and cloud computing are considered which can satisfy the requirements of the TaaS and COE, and meet user demands for an enhanced training experience. Furthermore, this paper discusses the approach taken to elicit the future needs of the Army's live training community, and how cloud computing and SOA are leveraged to meet required capabilities. Lastly, this paper discusses some unique considerations on SOA-related security issues.

Keywords-service oriented architecture, modeling and simulation, cloud computing, mobile computing, distributed computing, training, web services

I. INTRODUCTION

The Department of Defense (DoD) is constantly striving to improve the training of soldiers while reducing related costs. More specifically, three major areas of improvement have been identified within the military simulation and training domains. First, these systems often lack the ability to interoperate with one another unless extensive measures are taken to natively interface them. Second, when users require on-demand capability, software applications, and upgrades, they must wait for fielding support and personnel to provide installation on each client. Third, massive volumes of data are being stored and processed by a variety of unmanaged clients and servers requiring excessive physical space.

The U.S. Army in particular, is considering two strategies to address the above issues and recommend improvements. These strategies include service-oriented architecture (SOA) and cloud computing. SOA migration will enable total system interoperability, resulting in composable, reusable, and loosely coupled services. Cloud computing will allow services, components, software applications, software updates and upgrades to be readily available where consumers can access them as needed.

Currently, the Common Training Instrumentation Architecture (CTIA) is one of the three architectures defined by the U.S. Army's Live Training Transformation (LT2) product line. It is used by LT2 products to define interoperability standards among live training applications to support force-on-force and force-on-target training. Using an introspective approach, honest dialog and user feedback, it was determined that CTIA must evolve to address technology obsolescence and meet the growing needs of the live training community. Therefore, in order for the architecture to meet those needs, a Service Oriented Architecture (SOA) approach was identified as the preferred strategy. The CTIA team conducted a series of workshops and utilized SOA training and Human Centered Design (HCD) techniques in order to identify and prioritize the strategic business goals and objectives for the LT2 product line.

Moreover, the CTIA team selected and prioritized service oriented design principles, which are being applied to the architecture in order to achieve those goals. These efforts resulted into a roadmap and high level design for the SOA migration and evolution of CTIA to Training as a Service (TaaS)--The term TaaS is used by the U.S. Army internally and it refers to an "on-demand training environment" delivery model in which training software and its associated data are hosted centrally (typically in the cloud) and are accessed by users using a thin client, normally using a web browser over the Internet.

This paper entails five sections which discuss the following: 1) a high level description of the COE 2) a migration road map to a future SOA-based state 3) a description of the conceptual view of the TaaS architecture 4) a sequence of transition architectures (with a more in-depth discussion of the first transition) that will lead to the desired SOA state 5) a summary of the overall

progress accomplished on this project so far as well as some future directions. Finally, we provide a glossary entailing all related terminology used in this paper as a quick reference guide (see Appendix A).

II. Common Operating Environment

The Army Enterprise Network (AEN), illustrated in Figure 1, is comprised of four networks: the Global Defense Network, the At Home or Temporary Duty Station Network, the At Post/Camp/Station Network and the Deployed Tactical Network. The AEN enables unified land operations through all phases of deployment. In order to implement the AEN, the Common Operating Environment (COE) was established and published by the U.S. Army's Chief Information Officer [Cloud Computing Strategy, CIO, DoD, Technical Report, July 2012]. It entails an approved set of computing technologies and standards that enable secure and interoperable applications to be developed and deployed rapidly across five defined computing environments including 1) Enterprise Server, 2) Tactical Server and Client, 3) Platform (ground and air), 4) Mobile, and 5) Sensors. Each computing environment has a minimum standard configuration that also supports the Army's ability to produce and deploy high-quality applications quickly while reducing the complexities of configuration, supporting and training associated with the computing environment.

The current Army approach to information technology implementation and management is cumbersome and inadequate to keep up with the pace of change. The acquisition process focuses on the development and fielding of systems by programs established to deliver capability for a specific combat or business function. Based on functional proponent requirements, program managers individually choose and field hardware platforms and software infrastructures. Meanwhile, to support ongoing conflicts, Army and combatant commanders independently procure commercially available solutions, often installing and customizing them in theater. As a result, deploying and deployed units frequently must plan and execute operations using multiple computer systems with different hardware, operating systems, databases, security configurations and end-user devices. The extraordinary scale and scope of this complex integration raise cost, decrease interoperability, increase network security risk, expand the deployment footprint and add a tremendous burden to managing configurations. Most importantly, the process carries significant operational impacts.

The intent of the COE architecture is to normalize the computing environment and achieve a balance between unconstrained innovation and standardization. In the commercial sector, computing environments have become commodities and applications are developed and delivered on commoditized and inexpensive systems (for example, the Apple iPhone™ and Google Android™ mobile devices). With a COE, the Army can establish a framework similar to industry best practices [COE Architecture Memo, U.S. Army, 2010]. Also all interested communities within a COE will be able to: produce high-quality applications quickly and cheaply; improve security and the defense posture; reduce the complexities of configuration and support; and streamline and facilitate training. It is worth mentioning that this is a wholesale shift from the Army's traditional procurement of systems with dedicated software and hardware. Instead, applications will be designed, developed and deployed on a common computing environment, allowing the end user to download what he/she needs when he/she needs it. The complete COE description is provided in [Common Operating Environment Architecture, U.S. Army CIO, July 2010].

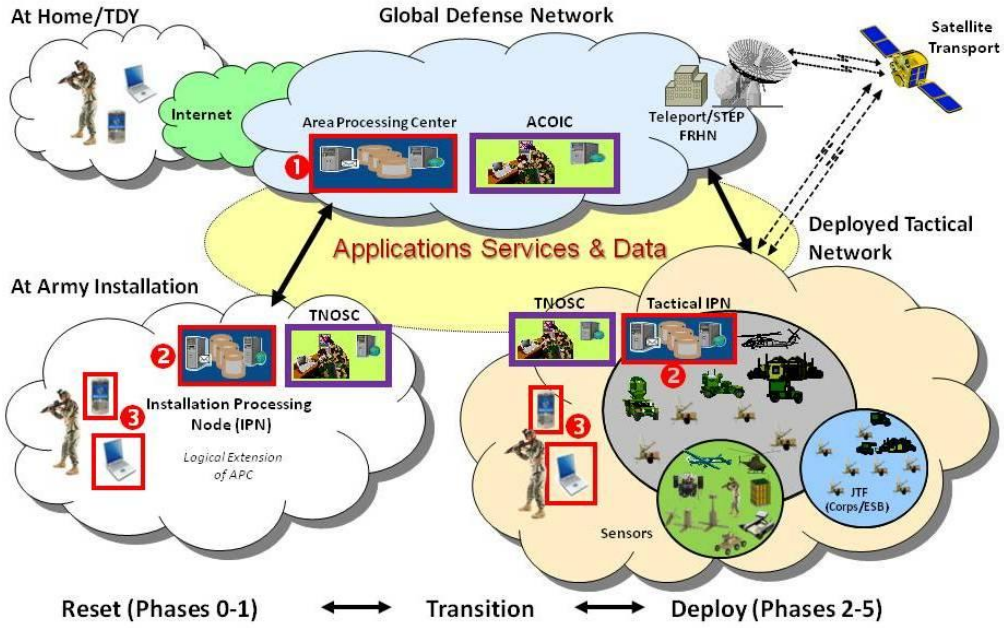

Figure 1. U.S. Army Enterprise Network (AEN)

III. MIGRATION ROADMAP

The SOA-based migration roadmap adopted by the U.S. Army is described in [Lanman, Horvath and Linos, 2011]. This strategy leverages modern cloud computing and virtualization technologies, which ensure effective interoperability among the Live, Virtual and Constructive (LVC) training systems and related applications. In addition, typical cloud engineering principles have been adopted while developing and orchestrating reusable, highly cohesive and loosely coupled software services at various granularity levels (i.e. both coarse and fine grain).

Lanman, Horvath and Linos describe a high-level migration roadmap, which entail a series of path-points as shown in Figure 2.

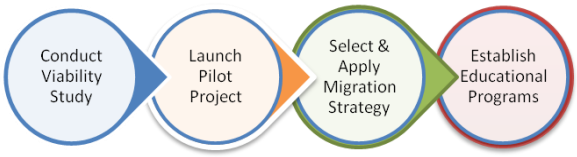

Figure 2. Migration Roadmap

As a first step, the Army intended to assess the overall feasibility of the modernization project for CTIA by considering both technological and non-technological issues. To this end, the first step was to address all non-technological issues by launching a comprehensive analysis. More specifically, this analysis was conducted in order to ensure an alignment of the US Army's business vision with the interests of all stakeholders involved as well as to satisfy all the customer needs. The results and related details of such analysis are described in [Lanman, et al, 2012]. During that effort, all interested stakeholders were identified and evaluated based on the degree of interest and potential impact they had in specifying the future CTIA state. As a result, various categories of stakeholders were considered including: Capability Developers (who define requirements, provide funding and overall direction), Material Developers (people responsible to develop and maintain architectural components), Exercise Executors/Users (interested in system features, performance, reliability and user-friendliness), Integrators (people concerned about integration and testing issues), External Systems (dealing with interoperability issues), Training Units, External Influences (concerned with any changes in external software system dependencies). After that, all stakeholders were classified based on their degree of interest and impact level. For instance, Material Developers exhibited a high interest and a high impact whereas Training Units and External Systems had a low interest and low influence on this project.

During the next step, the business goals were derived using the Goal-Question-Metric (GQM) paradigm [Basili, 2002]. GQM entails a hierarchy of goals defined at the highest level. Each goal is then associated with various questions listed at the next level of the hierarchy. At the third level, various metrics are typically defined, which are associated with each of the questions of the higher level. So, using the GQM methodology, the Army's viability study team created high-level goals and broke them down into related questions about each goal. It then defined related metrics which helped assess how well the goals were met. The main high-level goals identified during this study include: 1) Reduce Operational Costs and Complexity, 2) Align and Support Specific Product Development (with an emphasis on using current live training development efforts to advance the product line), 3) Enable Enhanced Soldier Training Effectiveness, 4) Reduce Development and Sustainment Costs, 5) Increase Technology Agility (with a focus on maximizing the ability of the architecture to incorporate cloud computing and virtualization) and 6) Leverage other existing Army Systems.

These high-level goals were further refined to more specific objectives and then prioritized. Highest priority is given to the introduction of new training capabilities with an emphasis on the effective use of mobile devices. The next priority is given to the enhancement of distributed exercises using centralized data centers and common support personnel. The third priority is to lower overall cost, reduce time and effort related to on-going maintenance and testing across the product line. Other high-priority objectives include the migration of current product development efforts towards a more agile development paradigm that will include reusable services.

In addition, during this study various important constraints were identified. First, it was agreed that some level of backward compatibility must be retained in order to allow the product line to recover a return on investment in the future (about five years). Also this effort must be compliant with the broader business goals and technology objectives provided by the Army's CIO to implement a Common Operating Environment (COE) using cloud engineering and virtualization on the Army's Global Network Enterprise Construct (GNEC). Moreover, it was noted that as the CTIA evolves it must consider the security impact in order to support the information assurance policies, procedures and standards.

IV. TRAINING AS A SERVICE (TAAS)

Training as a Service (TaaS) is the U.S. Army's term that refers to an "on-demand training environment" delivery model in which training software and its associated data are hosted centrally (typically in the cloud) and are accessed by users using a thin client, normally using a web browser over the Internet.

The TaaS strategy is to build functional components and the supporting intermediate infrastructure according to SOA principles and practices. The TaaS strategy decomposes the system into components and layers. To obtain maximum flexibility and the greatest opportunity for reuse, each component exposes its capability through services available to the end-user and to other applications on the AEN. By designing software around a set of services rather than a set of applications, TaaS aligns with the DoD migration to net-centricity [DoD, 2012] and

architectural patterns emerging in industry [Erl, 2009]. The architecture segregates the software that exposes persistent information (data services) from functional (or business logic) and presentation services, as depicted in figure 3. Both TaaS and the CTIA SOA are built upon layered architecture frameworks. The details of the CTIA SOA layered architectural framework is discussed in the next section and addresses separating the service capabilities represented in an industry standard SOA framework. Furthermore, figure 3 illustrates the mapping between the Regional Training Center concept and individual training range (or facility) as defined in [Lanman, Horvath and Linos, 2012] where common services can be deployed and used. TaaS and CTIA SOA embraces consistent SOA concepts and architectural tenets, but differs in the sense that CTIA SOA is focused on defining architectural patterns that, while consistent with the TaaS objective architecture, focus on the unique issues of the instrumentation training environment rather than the holistic enterprise environment.

In the current state, each installation of CTIA has dedicated infrastructure ranging from server racks full of equipment to installation on a laptop. In the future, the Army wishes to embrace cloud computing through the development of a regionalized and distributed training capability that provides the hardware and software at central locations. This relieves the units being trained from having to operate and maintain their own service infrastructure. While this may not be entirely practical for the large Combat Training Centers (CTC), many of the smaller training ranges, including home station training, could leverage this model. Additionally, there may be a blend of local and central resources where the majority of the infrastructure is hosted at central locations with range assets supplementing it where performance, security, or other restrictions require it.

Going forward CTIA must support the mobile computing world. It must enable trainers to use mobile devices to capture training observations and evidence just like one might use an app to post a picture to a social networking site. Figure 3 illustrates the conceptual view of this capability

While CTIA is moving towards support for Training as a Service (TaaS)—distributed and web-based training—both the DoD and Army are also moving toward this direction by providing new guidance on cloud computing, SOA and mobile strategy development. Moreover, the COE is defining a reference architecture [Hong and Baochun, 2013] for the Army community and a cloud computing environment that would host their services in the AEN.

Since many of the CTIA goals line up well with the COE goals, and the COE is specifically referenced by the CTIA constraint for Army policy compliance, it seems very natural that CTIA would seek to comply as much as possible with the principles found in the COE. Both the COE and the CTIA are built upon layered architecture frameworks. The SOA-based objective CTIA is consistent with the COE software view in that it embraces proper SOA concepts and architectural tenets. The mapping of layers is straight forward and provides a necessary logical connection between the objective CTIA and COE so that the CTIA architectural concept can be described. The objective CTIA concept, to a large extent, leverages the COE. To fully realize the strategic goals of the Army Live Training Community the target architecture is designed around a Regional Training Center (RTC) concept use case [Lanman, Horvath and Linos; 2011]. As seen in Figure 3, the RTC would provide "cloud" access to CTIA infrastructure, services, widgets, and web applications for mobile and web users. There would also be legacy services available to support current LT2 applications and ensure backwards compatibility.

The "Enterprise" in the LT2 context would be comprised of the entire Live Training Community. The RTC would be available to run multiple disparate training exercises from different facilities simultaneously. All of the facilities would be using a common hardware and software infrastructure provided by the RTC's "cloud".

This concept also supports the use case where it would be necessary to provide some local infrastructure for a specific facility. The local infrastructure would simply be a federate of the RTC and would provide all of the same capabilities locally or by way of reaching back to the cloud. Services would be federated and data would be synchronized dynamically with the "cloud".

One of the biggest initial challenges with applying SOA principles to an existing architecture is changing the way one thinks about the problem. CTIA is already composed of segregated services; a natural inclination is to wrap these with web interfaces. To avoid this pitfall and help ensure a clean bottom up approach was taken, the Architecture team applied service-oriented analysis techniques [Erl, 2007] to identify the core functions of the system. To this end, the basic training system business processes were modeled first using simple high-level flow charts. This resulted in a loose set of functional capabilities that must be met. Those capabilities then were assigned to use cases, which in turn were iteratively refined until a set of service candidates were identified around those contexts. This iterative process takes into account applying the SOA design principles, performance, the deployment environment, and other factors to create agnostic services that are reusable and composable.

Through-out this modernization project we followed Erl's approach where all services are logically grouped into three layers [Erl, 2009]. First, the utility layer includes non-business related services that support higher-level services such as logging. The next layer entails all entity services that model the real world business entities and provide the set of operations on those entities. The third layer consists of all task services which perform business processes that can be composed of multiple other services. It is worth mentioning, that a breakthrough for those involved in this process came while defining the entity services. Initially, the Architecture team fell into the trap of just wrapping SOA around the old architecture design [Erl, 2007]. In the legacy architecture, instrumented

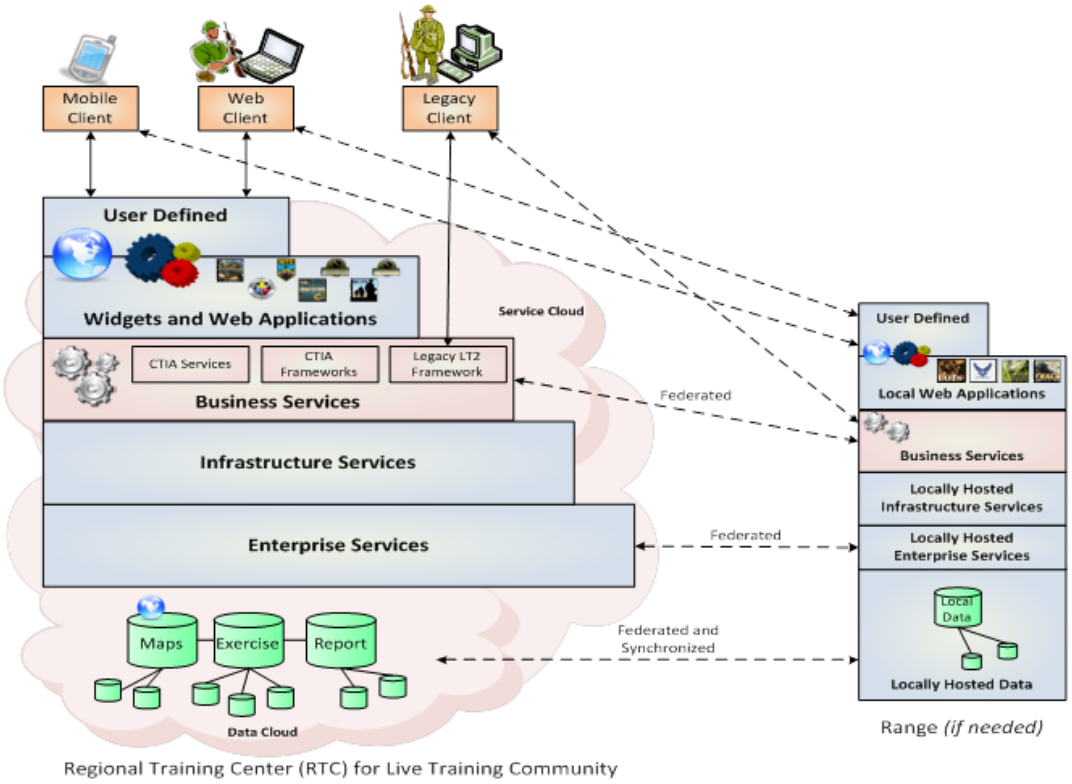

Figure 3. Training as a Service (TaaS) Architecture Conceptual View

devices, targetry systems, and cameras, for example, were modeled by the same service that modeled platforms and people. Additionally, interoperability gateways supported conversion from these devices to CTIA and software components. However, weapons platforms and people are very different from real world devices. Instrumented devices, targets, and cameras are real world devices that are controlled, have communications intricacies, have status, etc. The team naturally started along the same path as the legacy architecture, until service oriented analysis was really taken to heart. Business entities then started falling out as their own services, with operations specific to their context. This provided clean, well defined interfaces to a set of reusable services for not only modeling real world devices but also interfacing with them.

V. Transition Architectures

In order to support project, business, and product line goals, the services and capabilities of the target architecture were allocated into five transition architectures. Each such transition architecture was based on a specific use case for tracking soldiers in individual soldier and small unit training as described in [Lanman, Horvath and Linos, 2011]. Moreover, each subsequent use case increases in scale, capability, and complexity from the one preceding it by introducing services such as 2-dimensional visualization and training exercise control in collective and larger unit training also described in [Lanman, Horvath and Linos, 2011]. Considerations for transition architectures were also constrained by project funding. The objective SOA services and capabilities were then allocated to five transition architectures i.e. TA1 through TA5. Services allocated to each transition architecture instantiation will enable progressive levels of product team adoption. In addition, product teams will be able to orchestrate the architecture services to meet their intended training use case, and develop user level application interfaces. Finally, each such transition architecture will support integration with first generation CTIA to the extent of the services provided.

Previous versions of CTIA were implemented using Common Object Request Broker Architecture (CORBA) Internet Inter-ORB Protocol (IIOP) and Universal Datagram Protocol (UDP) multicast as the primary communications mechanisms between a central set of services (developed by CTIA), and thick client user interfaces and data processing components (developed by the LT2 community). While this implementation loosely followed Service Oriented Architecture (SOA) design principals to limited extents, the target architecture will apply a greater deal of service orientation in order to realize the strategic benefits of a SOA. The driving goals behind the objective architecture are to use the lessons learned from previous versions to make an architecture that decreases sustainment and development costs, while enabling thin (web-based) client applications, mobile applications for wireless devices and supporting the TaaS concept (i.e., implementing cloud computing). Next, we

describe the TA1 in some depth followed by a brief look at the future TA2-TA5 series.

VI. Transition Architecture 1

TA1 provides the Service Oriented Infrastructure (SOI), basic entity creation and tracking for the purpose of exercising the architecture, and the core services that all future transition architectures will build from. This will provide a demonstrable capability for the first year, including a prototype 2-dimensional map visualization application and participant definition tool, orchestrated by task, entity, and utility service compositions. In the following section, we describe the core services considered in TA1 to some extent (due to space limitations).

A. Generic Core Services

The following core services are generic for most SOA implementations and are defined by [Erl, 2007].

B. SOI Management Service

The SOI Management Service provides the ability to configure, monitor, and manage Services within the CTIA Target Architecture. A "Managed Service" is defined as one that registers with the Management Service. Such Services report their status and defined metrics to the Management Service, which can then be viewed by administrators/users. Managed Services can also be configured, started, stopped, paused, etc. through the Management Service. Both "Hosted Services" (those deployed to the SOI) and non-hosted, 3rd party Services can register with the Management Service for monitoring and management.

C. Logging Service

Services will be able to publish logging messages for significant events such as status changes and errors. The Logging Service simply records these messages and provides an interface for retrieving them.

D. Runtime Persistence Service

The Runtime Persistence Service will be a library responsible for providing an abstraction layer around the persistence of runtime data. It will be implemented using the Data Access Object (DAO) design pattern.

E. Service Framework

The Service Framework is a set of support libraries created to make service creation easier and minimize disruption if technologies such as messaging products change. It will provide:

- Coordinate conversion classes (reuse from legacy CTIA)
- Messaging Abstraction interfaces
- Service Discovery interfaces
- A code generation tool for generating classes wrapping the XML data model, database schemas, and any required mapping files.

F. Web User Interface Framework

The Web UI Framework is a collection of utility classes and user interface widgets (e.g. a coordinate widget or a symbol chooser widget) based on the Smart Google Web Toolkit (GWT). It is expected that it will grow over time as needs are discovered.

G. CTIA-specific Core Services

The following core services are specific to the CTIA SOA implementation.

H. Entity Organization Service

The Entity-Organization Service is responsible for creation, modification, deletion, and retrieval of Entity and Organization current state data and their relationships (task organization and spatial relationship). Tracking data is not considered state and is handled separately by the Tracking Service. In addition to simple state modification, this service will provide specific mechanisms to damage and resurrect entities. This is important in the cases where an entity is instrumented and the instrumentation must be contacted to complete the operation.

I. Exercise Service

The Exercise Service is responsible for creation and management of exercises. It does more than simply provide create, retrieve, update and delete (CRUD) operations of exercise objects. It is also responsible for performing the setup, archiving, deletion, and purging of exercises in an automated fashion, eliminating manual steps. This may involve calling other services.

J. DIS Enumeration Service

The Distributed Interactive Simulation (DIS) Enumeration Service simply provides a common place to retrieve and modify the DIS entity type and munitions type hierarchies. This is necessary to support user interfaces for hierarchy navigation and type selection.

K. Tracking Service

The Tracking Service allows clients to query for entities and organizations (and possibly assets in the future) based on their locations. Also, the tracking service performs logic to identify the primary tracker for an entity if it has multiple trackers providing data. Tracking events are then only sent out to the rest of the system if event came from the primary tracker.

L. 2-Dimensional Map Service

The 2-Dimensional Map Service will utilize COTS mapping technologies to show entity and organization situational awareness.

M. Participant Definition Service

The Participant Definition Service is a collection of user interfaces and backend services that allow a user to manage the task organization hierarchy and state of entities and organizations.

N. Product Selection

Table 1 describes each selected product (or technology) and its usage for implementing the generic and CTIA-specific services. The products were selected based on a series of Decision Analysis Reports (DARs) on various COTS and Government-off-the-Shelf (GOTS) products.

Table 1. Product Selection and Usage

Product	Usage
Red Hat Enterprise Linux	Operating system for the SOA infrastructure
Windows 7	Client operating system
JBoss ESB	Integration platform for services
JBoss Application Server	Web server for web applications
Juudi	Runtime discovery
To Be Determined	Inter-service messaging
Smart GWT	Web toolkit for developing the user interface
Ozone	"Webtop." It provides the platform that widgets run in, similar to a desktop interface
Java	Primary programming language for services
To Be Determined	Runtime data persistence for services
Sedris	Coordinate conversion library

VII. Future Transition Architectures

After the initial release of TA1, the CTIA SOA will evolve over the next five years into a series of future transition architectures. More specifically, TA2 will expand on TA1 to provide basic unit instrumentation and tracking. This will provide enough capability for product teams to adopt the architecture and compose the provided services to implement systems for land navigation with database persistence. TA3 will add services to support force-on-target engagements. Services will be available for product teams to implement instrumented ranges with fixed targets and support mobile devices. TA4 will provide services to support basic force-on-force instrumentation for brigade level home station training with constructive data feeds and battle damage assessment. Services will include asset tracking and exercise replay. Finally, TA5 will be the last instantiation of the objective architecture. The final solution architecture will be cloud-based with a deployable SOI supporting the full live training domain. Training will include up to battalion level force-on-force exercises integrating with mission command systems and entail full wrap-around live, virtual, and constructive interoperability capability.

VIII. Security

Security has been identified as one of the greatest challenges for migrating to a SOA and realizing the TaaS concept [Kanneganti and Chodavarapu, 2007]. The functions which impact the major security architecture decisions for CTIA and the LT2 product line are described in some detail below:

Authentication is concerned with validating the authenticity of the request or identity of the user making the request. It is essentially a logon capability that accepts user credentials; which could represent a user, message, or API request; and validates those credentials against a known list. It does not imply or grant any particular access or authorization to perform functions. For LT2 products, authentication is limited to control access to the LT2 system and does not play a role within the LT2 system. Authentication is implemented using COTS in the form of operating system (OS) logins for local users or requests and using encryption certificates for remote users or requests.

Moreover, authorization is concerned with permitting a user or requests to perform a specific function. In general, authorization is dependent on the proper authentication of a user or request, as different users or requests may have different permissions to perform specific functions. LT2 components may provide permissions-type capabilities, but those are not meant to address any security related authentication needs.

Information Assurance essentially exists as a set of processes and means that increase the confidence in the protection of the data and system. A key component of this is data encryption [Fleener and Maxon, 2012]. For LT2 products, this is implemented using encryption when transmitted data across untrusted networks. For security purposes, LT2 product deployments fall into the categories of Unclassified – Trusted Network, Unclassified – Untrusted Network, Classified – Trusted Network, and Classified – Untrusted Network

A trusted network is a wired network that is under the physical and logical control of the US Army and on which classified data is allowed to be transmitted without encryption. Cross-domain guards must be used when data is being transmitted between classified and unclassified trusted networks. An untrusted network is any that does not meet the qualification as a trusted network. Any wireless communications are considered to be untrusted by definition. Transmission of data across untrusted networks must be encrypted. If the data is unclassified, the encryption must meet at least the Federal Information Processing Standard (FIPS) 140-2. If the data is classified, then National Security Agency (NSA) Type 1 encryption devices must be used. The control and maintenance of the security certificates necessary to provide the authentication and encryption are not part of the CTIA architecture. For LT2 products, the instrumentation communications is not considered part of the security umbrella until it gets to an instrumentation or low-side gateway.

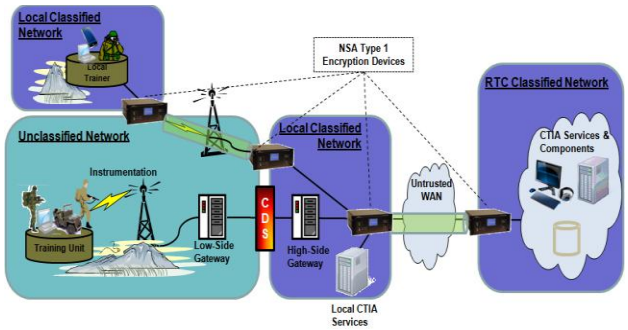

Figure 4. Classified RTC Capability Over Untrusted Networks

Figure 4 illustrates classified RTC capability over an untrusted network and depicts the most complicated deployment scenario from a security stand-point. It shows remote, wireless LT2 devices or components being used by Combat Trainers, some CTIA services on a classified network that is local to the training unit, and remote CTIA services being used as a RTC capability over an untrusted WAN.

IX. Conclusion and Future Directions

In this paper, we discussed some of the U.S. Army's successful efforts towards the modernization of its simulation and training legacy software. More specifically, we described a roadmap and a migration strategy to reach a future state of the Army's enhanced training systems. To this end, an effective feasibility study was launched followed by a pilot study on the CTIA within the COE. All related data gathered by such studies including requirements from all interested stakeholders, the user needs and related constraints, commanded the Army to leverage SOA and cloud computing as the enabling technologies to support TaaS capabilities across many heterogeneous platforms. In addition, in this paper we mentioned some related considerations regarding performance, security and governance (i.e. configuration management process). Finally, some of the Army's future efforts include deploying services as mobile applications in a cloud-based network and enabling continuous on-demand training in a distributed, web-based environment.

References

[1] Basili, V.R., et al. (2002). "The Goal Question Metric Paradigm". In: Marchiniak J. J (ed.): Encyclopedia of Software Engineering. New York, pp. 578-583.

[2] Bieberstein et al. (2007). "Executing SOA: A Practical Guide for the Service-oriented Architect". IBM Press books, 978-0132353758.

[3] DoD, (2012), "Cloud Computing Strategy", Chief Information Office, Department of Defense, Technical Report, July 2012

[4] Erl, T., et al. (2009). "SOA Design Patterns". Prentice Hall. ISBN 0-13-613516-1.

[5] Erl, T. (2007). "SOA Principles and Service Design", Upper Saddle River: Prentice Hall PTR. ISBN 0-13-234482-3.

[6] Erl, T. (n.d.). "Fundamental Cloud Computing". Vancouver, Canada: CloudSchool.com

[7] Fleener, G., Maxon, A. (2012). "Information Assurance Impacts of Mobile Architecture in a Training System". 2012 Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), Orlando, FL.

[8] Hong, X, Baochun, L. (2013) "Anchor: A Versatile and Efficient Framework for Resource Management in the Cloud," IEEE Transactions on Parallel and Distributed Systems, Special Issue on Cloud Computing.

[9] Kanneganti, R.; Chodavarapu, P.A. (2007). "SOA Security". Manning Publications. ISBN 1-932394-68-0

[10] Lanman, J.T., Clarke, S., Hillis, S., Darbin, R., Frank, D. (2012). "Applying Service Orientation to the U.S. Army's Common Training Instrumentation Architecture". 2012 Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), Orlando, FL.

[11] Lanman, J.T., Horvath, S.D., and Linos, P.K. (2011). "Next Generation of Distributed Training utilizing SOA, Cloud Computing, and Virtualization". 2011 Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), Orlando, FL.

[12] Lanman, J.T., Proctor, M.D. (2009). "Governance of Data Initialization for Service Oriented Architecture-based Military Simulation and Command & Control Federations". Journal of Defense Modeling and Simulation: Application, Methodology, Technology, Vol. 6, No. 1, 5-16.

[13] U.S. Army. (2010). "Common Operating Environment Architecture". U.S. Army, CIO, Technical Report, DoD.

Appendix A

Table 2 entails a glossary with related terminology used extensively by the U.S. Army and DoD as well as found in this paper. Our intention is to provide a quick reference guide and assist the reader become familiar with such terminology.

Table 2. Glossary

Acronym	Definition
AEN	Army Enterprise Network
CDS	Cross Domain Solution (Multiple levels of security tool)
CE	Computing Environment
CIO	Chief Information Officer
COE	Common Operating Environment
CORBA	Common Object Request Broker Architecture
COTS	Commercial off the Shelf
CT	Combat Trainer (Army training instructor)
CTC	Combat Training Center (Army training facility)
CTIA	Common Training Instrumentation Architecture
DIS	Distributed Interactive Simulation (communication protocol for simulators)
DBMS	Database Management System
ESB	Enterprise Service Bus
IaaS	Infrastructure as a Service
IIOP	Internet Inter-ORB Protocol
IS	Instrumentation System
LT2	Live Training Transformation (Army training product line)
LVC	Live-Virtual-Constructive (modeling and simulation domains)
MIP	Managed Interface Provider
NIST	National Institute of Standards and Technology
PaaS	Platform as a Service
RTC	Regional Training Center (Army consolidated data center concept)
SaaS	Software as a Service
SOA	Service Oriented Architecture
SOI	Service Oriented Infrastructure
TaaS	Training as a Service (Army SOA and cloud-based concept)
UDP	Universal Datagram Protocol
UI	User Interface
VCSA	Vice Chief of Staff of the Army
WAN	Wide Area Network
XML	Extended Markup Language