

Next Generation of Distributed Training utilizing SOA, Cloud Computing, and Virtualization

Jeremy T. Lanman, Steven D. Horvath
U.S. Army PEO STRI
Orlando, FL

jeremy.lanman@us.army.mil, steven.d.horvath@us.army.mil

Panagiotis K. Linos
Butler University
Indianapolis, IN
linos@butler.edu

ABSTRACT

Product-line architectures (PLAs) have received considerable attention within the U.S. Army's Live, Virtual, and Constructive (LVC) training domains. The software PLA framework paradigm has proven successful in minimizing stovepipe system development, and reducing system life-cycle cost and schedule. However, as the Department of Defense (DoD) and Army strive to further reduce system life-cycle costs, and as technology improves, LVC PLAs must evolve to align with future acquisition strategies and technology insertion. Three major areas that require improvement for PLAs have been identified within the LVC environment. First, traditional PLAs lack the ability to interoperate with one another unless extensive measures are taken to natively interface them. Second, when users require on-demand product-line components, software applications, and upgrades, they must wait for fielding support and personnel to provide installation on each computer. Third, massive volumes of data are being stored and processed by large complex database servers requiring excessive physical space.

There are three state-of-the-art strategies that can be implemented to improve PLA for LVC training, and collectively reduce system life-cycle costs. These strategies include service-oriented architecture (SOA), cloud computing, and virtualization. This paper will evaluate the concepts, technologies, challenges, and benefits for adopting these complementary approaches into the LVC training domains. Numerous features and benefits will be identified by the realization of these architectural strategies. SOA migration will enable total system interoperability, resulting in composable, reusable, and loosely coupled services. Cloud computing will allow product-line architectural services, components, software applications, and software updates and upgrades to be readily available in a logically centralized repository where consumers can access them as needed. Virtualization will improve organization between database servers and reduce hardware footprint.

ABOUT THE AUTHORS

Dr. Jeremy T. Lanman is the lead systems architect for the U.S. Army PEO STRI Common Training Instrumentation Architecture (CTIA) and Consolidated Product-line Management (CPM) construct supporting the Army's Live Training Transformation Family of Training Systems. His professional experience includes 10 years of DoD acquisition and systems engineering of military modeling and simulation, and training systems. His research interests include service-oriented architectures, live, virtual, and constructive integration and analysis, and distributed computing. Dr. Lanman holds a PhD in Modeling and Simulation from the University of Central Florida, an MS in Software Engineering from Embry-Riddle Aeronautical University, and a BS in Computer Science from Butler University.

Mr. Steven D. Horvath is an intern in the U.S. Army Program Executive Office for Simulation, Training, and Instrumentation (PEO STRI)'s Student Temporary Employment Program (STEP). Mr. Horvath works for the Project Manager of Training Devices (PM TRADE) in the Live Training Transformation Product Line system engineering initiative. He is currently pursuing a BS in Mechanical Engineering from the University of Central Florida.

Dr. Panagiotis K. Linos is a professor of Computer Science and Software Engineering at Butler University. Before joining Butler, he was the chairperson of the Computer Science department at Tennessee Technological University. Dr. Linos has more than 20 years of academic and consulting experience. His funded on-going research activities focus on developing innovative tools and techniques to facilitate the modernization of legacy software systems. He is a former or current consultant to various organizations including the U.S. Naval Research Laboratory, Kaman Sciences Corporation and Sallie Mae where he has been engaged in projects related to the migration of large legacy systems to SOA. He received a PhD in Computer Science from Wayne State University in 1990.

Next Generation of Distributed Training utilizing SOA, Cloud Computing, & Virtualization

Jeremy T. Lanman, Steven D. Horvath
U.S. Army PEO STRI
Orlando, FL

jeremy.lanman@us.army.mil, steven.d.horvath@us.army.mil

Panagiotis K. Linos
Butler University
Indianapolis, IN
linos@butler.edu

INTRODUCTION

The live, virtual, and constructive (LVC) training systems and information technology (IT) processes that are acquired, fielded, and sustained by the U.S. Army are disorganized and unable to comply with the constantly evolving computational environment. Throughout the 1970's and 1980's, the Army developed their training devices in order to perform a specific set of combat or business functions. As a result, training ranges installed their own hardware and software infrastructures based on customized requirements, which limited interoperability, increased the hardware footprint, and raised life-cycle costs. To overcome these obstacles, the Army adopted the first generation product-line architecture (PLA) among its training programs. This approach allowed training components such as Exercise Control (EXCON), After Action Review (AAR), and data recorders to be realized as common frameworks, architectures, and standards, which enabled processes and software to be reused across the product-line.

The current state of LVC training must evolve to the next generation of distributed training in order to adapt to a shrinking defense budget, conform to policy updates, and enhance the training capabilities afforded to the Warfighter. The next generation of distributed training will transform product-line software 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 the Common Operating Environment (COE) Architecture Guidance published by the U.S. Army Chief Information Officer (CIO)/G-6 and the Assistant Secretary of the Army (Acquisition, Logistics, and Technology) (ASA(ALT)). This document describes and demands a specific set of computing technologies to be implemented by the Army Enterprise Network in order to enable the rapid development of software capabilities across servers, mobile devices, and platforms. Three cutting edge technologies exist that can satisfy the requirements of the COE Architecture and improve the product-line architectures for LVC training. These include service-

oriented architecture (SOA), cloud computing, and virtualization.

In May 2011, the Project Manager of Training Devices (PM TRADE) hosted a workshop with industry and academia to discuss SOA, cloud computing, and virtualization concepts and technologies with focus on soliciting best practices, technical expertise, and guidance in developing a roadmap for the Army's Next Generation of Distributed Training environment.

CONCEPT AND TECHNOLOGY BACKGROUND

Prior to the advent of a product-line architecture approach to live training systems, the Army acquired stand-alone systems to address local requirements. These systems, commonly referred to as stove-piped systems, cannot communicate with each other unless extensive measures are taken to link them together one pair at a time. These connections have deteriorated the connectivity and flexibility of the Army's live training environment.

The Army's First Generation Live Training Transformation (LT2) Product-line Architecture

The Army addressed many of these issues in 2001 with the establishment of the Live Training Transformation (LT2) Product-Line. LT2 is an Army strategy that utilizes product-line engineering and management to guide the acquisition, development, and sustainment of integrated and interoperable live training capabilities. The overarching objective of the LT2 Product-Line is to maximize component reuse, reduce fielding time and schedule, and minimize costs of live training systems. The collection of live training systems built from common frameworks, architectures, and standards is identified as the LT2 Family of Training Systems (FTS). The architecture that defines the capabilities offered by the LT2 FTS is the Common Training Instrumentation Architecture (CTIA) (Dumanoir, 2005).

Service-oriented Architecture (SOA)

The term SOA is difficult to define for two reasons. The first reason is that there is no universally accepted definition. The second reason is that an SOA is not a concrete architecture, but rather an approach used to obtain one. A frequently used definition of an SOA that many can agree on is a software paradigm used to manage business capabilities across distributed systems. An SOA consists of the following key technical elements: services, interoperability, and loose coupling.

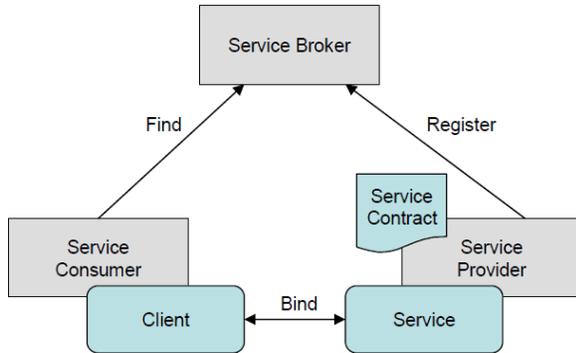


Figure 1: SOA Concept

A service is an IT representation of a self-contained business capability. The capabilities can be as simple as storing data or as complex as a processing a customer's order. Services and the service consumers are related by interfaces and contracts. The interface can return information for multiple messages, which are shared between service providers and consumers. The interface should be well defined and unambiguous. As shown in figure 1 (Erl, 2007), a contract is a complete specification of a service between a provider and consumer that establishes a standard between the service request and response. Services are generally classified according to nonfunctional aspects such as Quality of Service (QoS) and Service-Level Agreement (SLA).

Interoperability includes the capacity to have services communicate with each other throughout a wide distributed network. The infrastructure that allows services to interoperate is the Enterprise Service Bus (ESB). The ESB enables consumers to call services by providing connectivity, data transformation, routing, security, etc. Other responsibilities of the ESB are to ensure rapid and economical service deployment and allow services to be reusable and scalable throughout their lifecycle. There are two types of ESBs including protocol-driven and Application Programming Interface (API)-driven. A protocol-driven ESB establishes a set of rules, roles, and processes that providers and consumers have to match. The ESB and

any corresponding connected systems are decoupled, meaning that they don't share any code. An API-driven ESB permits providers and consumers to implement and/or call services. As a result, protocol details remain transparent.

Loose coupling refers to the concept of minimizing the amount of system dependencies. A loosely coupled system leads to business processes that are immune to the consequences of modifications and temporary failures. One benefit of a loosely coupled system is the ease at which system upgrades can be integrated. In addition, loose coupling allows a system to be hardware, software, and platform agnostic, meaning the system is vendor independent (Josuttis, 2007).

There are many benefits that can be achieved through the adoption of an SOA. These benefits include the following:

- Loosely coupled business services across any system or network
- Enhances reliability and reduces hardware acquisition costs
- Leverages existing technology investments

While the benefits of an SOA seem abundant, several limitations can delay its adoption. These limitations include the following:

- Substantial startup costs due to the renovation of an existing architecture
- Integration and implementation is a slow and complex process
- Standards are immature

Cloud Computing

As with SOA, cloud computing also has many definitions in use worldwide. In essence, cloud computing is an environment that provides users secure access to multiple server-based computational applications via a digital network. As illustrated in figure 2, the consumer gains access to the cloud server through a simple thin client or wireless mobile device, which possesses just enough processing power necessary to connect to the cloud network. Data storage and processing is managed by the centralized cloud server, freeing the user from downloading and installing applications on their own hardware. There are two important aspects to consider regarding cloud computing including its architecture and applications.

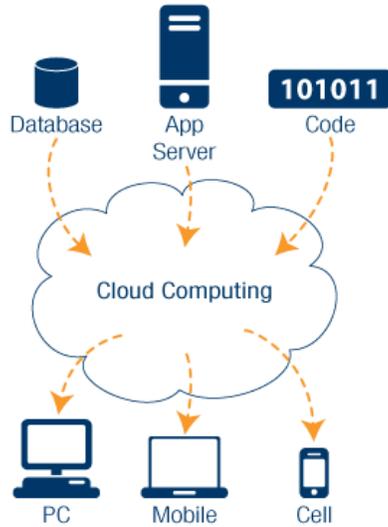


Figure 2: Cloud Concept

As shown in figure 2, a cloud computing architecture consists of two sections: the front end and the back end. The front end is the thin client that comprises a simple interface software, while the back end is the cloud consisting of databases, control nodes, computer networks, and application servers. The two ends interoperate with each other through the internet with a web-based service. A well structured cloud computing environment has the ability to guarantee on-demand service capabilities, ubiquitous network access, elasticity, measured billing, and resource pooling. A cloud network can be broken down into three services including Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) (Strickland, n.d.).

- IaaS is the back bone of cloud computing and the realization of computing as a completely outsourced service. The consumer has the option of utilizing as much infrastructure as necessary, where managed hosting and development environments are offered as services. Dynamic scalability and access to IT assets are the end results of IaaS.
- PaaS allows consumers to write their code, while the provider uploads that code onto the web. It's a remarkable service that provides a development platform to developers. It also provides services to develop, test, deploy, host, and maintain applications in an integrated development environment.
- SaaS operates on the principle that one can rent software from a service provider on a centralized network instead of purchasing it yourself. The software is available on the web, which promotes flexibility, scalability, and on-demand access.

The capabilities inherent in a cloud-centric environment can provide many benefits to the training community. These benefits include the following:

- Reduction of hardware infrastructure on the front end and a reduction in investment costs
- On-demand access to any application at any time from any location, provided the client has network access
- Improved processing power due to the ability to harness all available processors on the back end

Although the benefits of adopting a cloud-centric environment provide an improved IT enterprise, there are some limitations that can delay its adoption. These limitations include the following:

- Elevated security vulnerabilities due to the remote usage of IT resources
- Limited portability between cloud providers due to immature standards
- Reduced operational governance control due to dependencies on off-site cloud-hosted IT resources

Virtualization

The creation of a virtual entity from a physical entity such as a hardware platform, operating system, or server is known as virtualization. Many different types of virtual machines exist in the IT enterprise, yet the most common is server virtualization. The process aims to partition one physical server, called the host, into multiple virtual servers, called guests, each with their own unique operating systems and processing capabilities. There are three types of virtual servers including full virtualization, para-virtualization, and operating system virtualization.

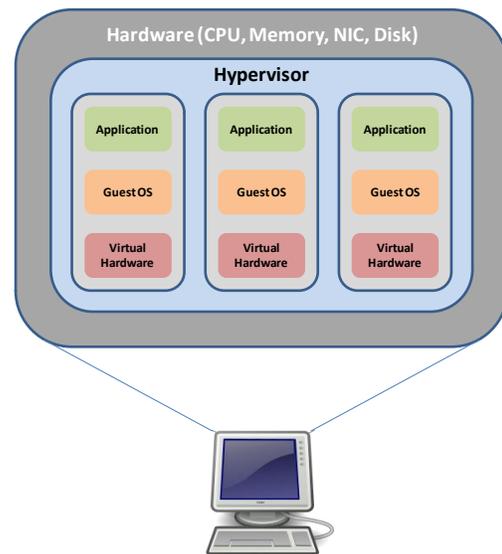


Figure 3: Virtualization Concept

As shown in figure 3, full virtualization uses a hypervisor, which is a special type of software that interfaces with the host servers central processing unit and disk space. In other words, the hypervisor acts like a platform for the virtual server's operating system, while keeping it isolated from the other virtual machines running on the host. The hypervisor manages the physical server's resources by directing resources from the host to the appropriate guest.

In a para-virtualized system, the guest servers work in unison, as opposed to a full virtual system where they are isolated from each other. The para-virtual hypervisor does not require as much processing power to manage the guest operating systems because each operating system is aware of the demands of the other operating systems on the host.

Operating system level virtualization acquires its virtualized capabilities directly from the host operating system, meaning it does not require a hypervisor. Each guest server remains isolated from one another, but each guest must run on the same operating system. This type of homogenous system operates faster and more efficiently than the other types of virtual servers. (Strickland, n.d.)

There are several benefits that can be afforded to the training community through the execution of a virtualized environment. These benefits include the following (IBM, 2008):

- Reduction in hardware footprint due to the consolidation of physical machines into virtual machines
- Increased redundancy without acquiring additional hardware
- Ability for developers to test applications remotely

There are also several limitations that can degrade training assets. These limitations include the following:

- Decreased processing power for servers dedicated to high demand applications
- Inability to transfer virtual servers from one physical machine to another unless they use the same manufacturer's processor
- Increased risk of voiding license agreements that don't support virtual machines

MOTIVATION

This section provides the rationale for implementing SOA, cloud computing, and virtualization within the U.S. Army. These reasons include the recent development of the Common Operating Environment

published by the U.S. Army CIO/G-6 and ASA(ALT), the threat of technology obsolescence, and the potential benefits afforded to the training community.

Common Operating Environment (COE)

One of the greatest contributors to migrating toward a distributed web-based training environment is the COE (figure 4) for the Army Enterprise Network. The COE was established by the CIO/G-6 in order to initiate the development of 'as is' and 'end state' network architectures to guide evolution of network procurements and enhancements. The Army Enterprise Network encompasses all phases of training and deployment, while enabling full-spectrum operations.



Figure 4: U.S. Army CIO/G6 Common Operating Environment Concept

The COE will ensure seamless interoperability and security of computing capabilities including servers, clients, mobile devices, sensors, and platforms across the Army Enterprise Network. The requirements that must be satisfied to guarantee the success of the COE include the following: hardware/software solutions must remain agnostic, scalable, and Commercial-off-the-shelf (COTS), application must be standards based as stated in accordance with DoD and Army policy. The end-state objectives of the COE are to provide backward- and forward- compatible software capabilities to the Warfighter anywhere on the network, reduce complexity and risk through standardization, and consolidate Command, Control, Communications, and Computers (C4)/Information Technology (IT) resources. (U.S. Army, 2010)

Technology Obsolescence

As the DoD evolves training doctrines, improves information assurance, and advances training technologies, older existing resources become incompatible with current standards. Training resources including target controllers, hardware infrastructure, and software products are experiencing numerous challenges from a technology obsolescence point of view. The LVC training PLAs are suffering from technology obsolescence at an accelerating rate

due to new products superseding older products and/or the reduction in supporting technology used to repair and replace malfunctioning equipment. In addition, the supply of spare parts is decreasing, which increases maintenance and sustainment costs because the DoD must design customized modifications to integrate new technology into older systems. For example, live training ranges are currently facing this issue, especially with legacy target systems including Enhanced Remote Equipment Target Systems (ERETS) for example. When an ERETS fails, the range operator must acquire a different target controller and new target control software. This effort not only increases costs, but also increases range downtime and decreased training productivity. The realization of SOAs, cloud computing, and virtualization technologies and processes within live training would allow a more seamless transition of technology insertion as products become obsolete. (Gomez, 2010)

Potential Benefits Afforded to the Training Community

According to Gartner, SOA will be used in more than 80% of mission-critical operational applications and business processes by the year 2010 (Gartner, 2007). Analysis of the literature indicates that the SOA, cloud computing, and virtualization visions lead to a belief that significant efficiencies and cost savings can be gained. As the U.S. DoD training community moves forward with its vision of highly distributed net-centric capabilities in current and future programs, it will be difficult for the training community to deploy, maintain, and evolve capabilities without the benefit that SOA, cloud computing, and virtualization provides.

SOA, cloud computing, and virtualization offer the promise of cost savings, data sharing, interoperability and increasingly agile operations. By definition, SOA services are to be reusable. In an organization as large as the DoD, the existence of reusable services creates many opportunities to reduce redundancy and increase efficiency. From a mission effectiveness perspective, there are many areas where SOA, cloud computing, and virtualization could add value. SOA promises to increase interoperability within and among the services through discoverable standardized service contracts (Erl, 2007). Through reusable data services, information can be shared across the cloud-centric enterprise increasing dissemination and knowledge transfer. Readiness can be improved through efficiencies gained in information access. Additionally, widespread SOA, cloud-centric, and virtualized environments throughout the DoD training

community will increase organizational ability to deal with rapid change.

PROPOSED USE CASES

Several operational use-cases exist that are beneficial to the realization of SOA, cloud computing, and virtualization within live training. While these concepts and technologies are applicable to the LVC environment, this section will focus on live centric use case examples.

A proposed simple use case for SOA and cloud computing is with the eGreenBook concept used by Combat Trainers or Observer/Controllers (O/Cs) during training exercises. The objective is to store and distribute data from field or main posts to individual Units on-demand. A use case for virtualization is the use of virtual servers at Digital Range Training Systems (DRTS). The objective in this case would be to increase availability, reduce the hardware footprint, and eliminate full up workstations from controller workstations.

Electronic Green Book (eGreenBook) Data Collection Tool

During training exercises, Combat Trainers record observations by hand in a green book (spiral bound notebook) and then must re-type the data at a later time. The current process includes the following manual steps and resulting issues:

- *Step 1:* During training exercises, Combat Trainers takes note of an observation and may sketch a picture.
Issue: Many observations are missed when time does not allow for note taking.
- *Step 2:* After the training exercise, the Combat Trainers elaborate on their observation notes.
Issue: Details and observations are lost if Soldier cannot remember after the training period.
- *Step 3:* Leaders collect paper forms from all the Combat Trainers to analyze the observations.
Issue: Considerable time is spent consolidating notes and standardizing the feedback.
- *Step 4:* Unit leadership collates observations into an After Action Review (AAR).
Issue: Manual analysis and missing information slows the process and manual reports cannot be automatically compared to past events.

SOA and cloud computing can be implemented for the distribution and storage of data from observation collection to reporting as depicted in Figure 5.

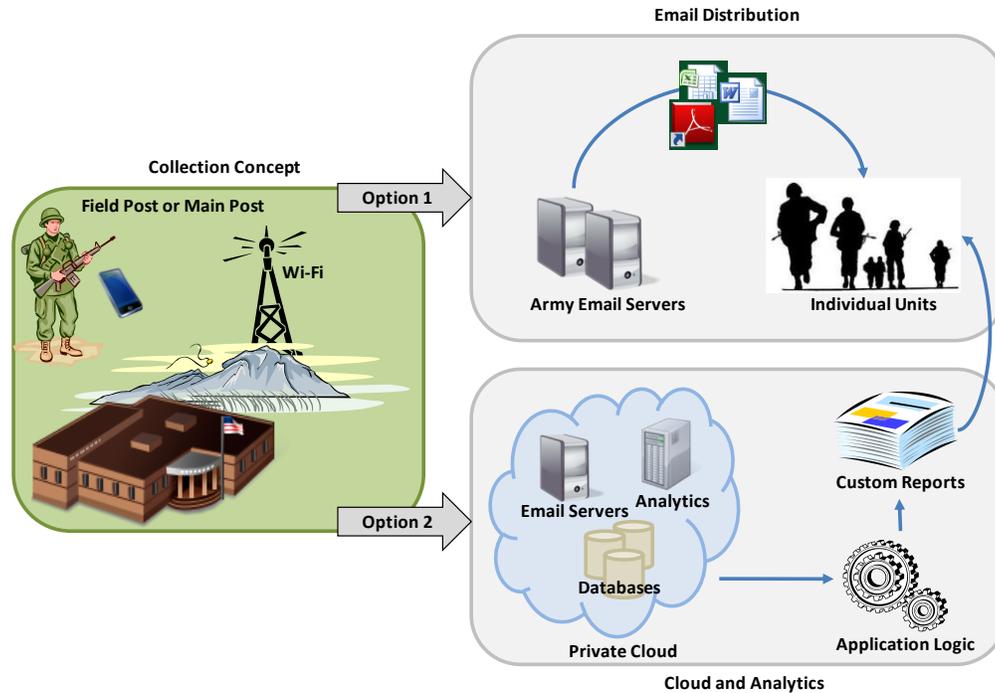


Figure 5: eGreenbook Use Case Concept

Using a hand-held mobile device, Combat Trainers collect observations using the appropriate application (notepad, picture and video features, etc.). Once observations have been collected, the Combat Trainer walks over to a location with Wi-Fi coverage and either sends observations directly to Soldiers via email or the observations are backed up to the “cloud”. Analytics are applied to the observations within the “cloud” to generate specialized reports. Finally, specialized reports (e.g. AARs and Situation Reports) are sent to the deployed Units to improve training exercises.

The potential benefits derived from this use case include:

- Providing combat trainers with a highly automated method for collecting and reporting training observations in real-time.
- Providing leadership with advanced analytics on the observations for trend analysis.
- Enabling mid-event test modifications based on near-real time observation trends.
- Capturing situational data to tailor and improve training exercises.
- Providing situational metadata associated with the observation collected.
- Eliminating current labor intensive process that is paper driven.
- Expanding and improving functionality by incorporating maps, pre-loaded events, scenario information, instant messaging, etc.

The potential return on investment (ROI) includes:

- Army leadership may utilize the analytics capabilities downstream to improve future efforts.
- Automated data collection thus time savings for personnel resources will be realized.
- Frees up personnel from labor intensive processes to focus on other tasks.

Digital Range Training Systems (DRTS)

A DRTS is a type of heavy tactical vehicle trainer designed to provide the infrastructure and instrumentation necessary for Abrams, Bradley, and Stryker live fire gunnery training systems and qualifications. The DRTS supports enhanced training data collection and AAR capabilities by leveraging a modernized suite of digital capabilities.

The DRTS is currently utilizing virtual servers in order to align with technology insertion. There are five disparate servers dedicated to the DRTS including the file system, CTIA service, gateway, database one, and the domain controller. The current configuration virtualizes the first three servers into one physical server, while the remaining two possess their own physical server. The Product Manager of Digitized Training (PM DT) is working to consolidate the five original servers into one physical server, while providing an additional physical server to act as a back-

up. A third server would remain offline to act as a standby server in case of emergencies. Another preliminary effort occurring with the DRTS is the reduction of workstation infrastructure. Traditionally, two computers were supplied to each individual workstation. The future goal is to eliminate full up workstations from these controller workstations in order to acquire virtualized or thin-client workstations.

The potential benefits derived from this use case include:

- Reduced hardware footprint
- Improved fail over capabilities

The potential return on investment includes:

- Reduction in utility expenses and cooling costs due to a decrease in physical machines
- Decreased infrastructure costs due to a reduction in physical workstations

ADOPTION CHALLENGES

The value of the migration and evolution to SOA, cloud computing, and virtualization concepts and technologies for the proposed Army use cases is clear; however, there are certainly technical and organizational challenges to address.

Security and Information Assurance

The largest challenge affecting the functionality of SOA, cloud computing, and virtualization is security. Security practices are necessary in order to protect systems from viruses and data leakage; however, security degrades system performance. Therefore, a proper balance must be achieved. The greatest contributors to security risks include boundary layer and trust uncertainty.

In an SOA, cloud computing, and virtual environment, services are constantly being shared across organizational boundaries. Organizations that rely on traditional network based intrusion prevention systems will be unable to locate and mitigate security risks that occur outside the boundary layer. The boundary layer is the interaction between service providers and consumers or the traffic between two internal virtual machines within a physical server. (Zetlin, 2009)

Another contributor to security risks is the uncertainty of trust that occurs when sharing services in an SOA or applications in a cloud environment. The distribution of valuable and sensitive information to an offsite provider reduces the amount of control one has over their infrastructure. Information assurance, which is the

process of managing risks associated with the use, storage, and transmissions of data, is a concern with SOA, cloud computing, and virtualization, especially within the U.S. Army. Unclassified, classified, and secret level information must remain protected and isolated from inappropriate personnel, which is a challenge when information is being processed in an SOA, cloud, or virtual environment.

While these security risks pose many dangers to a highly distributed SOA, cloud computing, and virtualized environment; there are ways to minimize their effects. The first approach is to implement the accreditation process early on in the system's design. Valuable use cases and lessons learned can be leveraged from the SOA and cloud computing efforts of the Army CIO/G-6 and the Defense Information Systems Agency (DISA). Another approach is to simply educate leadership on the risks and benefits that SOA, cloud computing, and virtualization present.

Technical Performance

There are several performance-based issues concerning the adoption of SOA, cloud computing, and virtualization. Performance is an important factor regarding an IT enterprise system because it affects network bandwidth, speed, and robustness. The most notorious performance issues degrading SOA, cloud computing, and virtualization adoption include bottlenecking, latency, and Central Processing Unit (CPU) overhead, respectively.

Bottlenecking can compromise the performance of an SOA by reducing total system scalability. Traditionally, an SOA consists of service components and client applications. The client application can be a web service or another application that relies on the service component to perform its functions. The client application is typically broken down so that the components can be run on multiple servers as separate services. The bottleneck effect occurs when application data is being stored and/or accessed by a single database (Khan, 2010).

Latency, which is the time delay that occurs during a systems executable instruction, is an adoption challenge affecting cloud computing. Latency issues are most predominant with SaaS systems. During an interaction between the cloud provider and user, communication is constantly taking place over the internet. The latency arises due to the constant back-end machine-to-machine conversation that occurs between the SaaS provider and web browser (Linthicum, 2010).

CPU overhead, which is a system operating at its peak load capacity, is an adoption challenge affecting virtualized servers. This is a common issue that can occur after a system is fielded when the network experiences a higher workload demand than originally expected. The end result is increased CPU power usage, which can cause erratic and degraded system performance (Salsburg, 2007).

All of the aforementioned performance issues will need to be addressed in SOA, cloud computing, and virtualization solutions for the LVC training architecture environment.

Acquisition Culture

A computing environment is not service-oriented just because capabilities are delivered using shareable services. A computing environment is not truly service-oriented unless it takes advantage of existing services where available and develops needed services taking into account the bigger picture of uses beyond the current need. DoD contracts focus on the particular capability being contracted for and make no provisions for delivering beyond that. Contractors are paid for the capability they deliver, making it desirable to maximize capability developed for a specific contract. This is not to suggest that the contractors for particular projects should be responsible for the creation and maintenance of an SOA, cloud, or virtualization framework suitable to meet DoD requirements. Contractors working on specific projects should intend to take advantage of existing DoD SOA, cloud, and virtualization frameworks. Contractors however, should be encouraged to embrace SOA, cloud computing, and virtualization for their projects by leveraging the use of services existing within those frameworks and considering the greater good when developing new services to be made available through those frameworks.

Neither the sponsor nor the contractor is rewarded or incentivized to provide a service-based, cloud-centric, or virtualized solution, which meets a greater good and provides additional enterprise benefit for the whole of the DoD. There are limited explicit incentives to take advantage of existing services when possible that meet program needs. The DoD has unwittingly tied the hands of these very talented professionals by not providing a mechanism to encourage a specific focus on enterprise benefit.

Acquisition cultural and organizational changes are necessary if the DoD is going to be successful with full-scale SOA, cloud computing, and virtualization solutions. Contractors and project sponsors should be

encouraged through policy changes and funding incentives to think beyond the current problem. Both the contractor and customer sponsor need to be incentivized to develop services that will solve problems the DoD might not yet realize that they have or issues that might not be relevant to the contracting agency but that could have significant impact on another agency.

ARCHITECTURE MIGRATION AND EVOLUTION STRATEGY

This section describes a high-level roadmap towards establishing an architecture migration and evolution strategy for LVC domains. The ultimate goal of the *strategy* is to craft an SOA and cloud computing framework that enables full interoperability among LVC training systems by developing and orchestrating reusable, highly cohesive and loosely coupled software services at various granularity levels (e.g. fine and coarse grain).

The proposed roadmap entails a series of path-points, which must be visited to reach a desirable future state. Some of the path points of the proposed roadmap are briefly described below.

Path Point 1: Viability assessment

The goal here is to assess the overall SOA migration feasibility by considering both technological and non-technological issues and constraints (e.g. economical, business, etc.) More specifically, we will first gather common goals from all stakeholders involved. Secondly, we will perform a health assessment of all current LVC domains and look at the overall technological landscape. This assessment will include an in-depth analysis of all active LVC software applications involved. Based on this assessment we will define the future SOA migration state and specify all necessary requirements to get to that state. Finally, we will create and include a project management plan of the overall migration effort. This viability study will entail a cost/benefit analysis in order to determine all necessary resources required to complete the *strategy* project and its long term ROI.

Path Point 2: Pilot study

Before all necessary resources are committed, we intend to launch a pilot study that aims at the migration of a selected subset of representative LVC legacy systems to an SOA platform. The selection of the pilot study is of key importance since we hope to demonstrate some evidence of early success and also

learn how to apply our approach to larger scale LVC domains. During the pilot study we will explore existing SOA, cloud computing, and virtualization best practices, related enabling technologies, methodologies and tools such as SOA backplane infrastructure and standards for Web services (e.g. ESB, WSDL, SOAP, XML, etc.). In addition, we will leverage available service conception and migration methodologies such as SMART from SEI and others. At a finer grain granularity, we will also explore and utilize best-practices, cutting-edge technologies and mature industrial tools used during re-engineering and modernization of existing legacy software systems in similar domains. In particular, we will use tools such as parsers, complexity analyzers, simulation rules miners, reverse-engineering and design recovery tools in order to detect and extract fine-grain services from our LVC software legacy applications. Our experiences from the pilot study will help us understand how to conduct a large-scale SOA migration strategy across all LVC domains as well as to make a strong business case of a viable long term ROI to all stakeholders involved (Lewis, 2008).

Path Point 3: Select and execute a migration methodology

The lessons learned from the pilot study will be helpful in identifying an appropriate SOA migration methodology for the LVC legacy software applications. So, we will consider and select an appropriate state-of-the-practice migration methodology that will be included in our *strategy*.

The selected methodology will consist of typical activities for detecting and extracting effectively potential services at various granularity levels (e.g. coarse and fine grain). We will start by finding reusable candidate services from our LVC software legacy applications. After that, we will determine the granularity level of each extracted candidate service. Next, we will specify the functionality and interface for each selected service. In addition, for each candidate service we will design test cases and execute unit-test runs to ensure correct functionality and reduction of defects. Moreover, we will establish an orchestration strategy for services. This strategy will entail protocols for the publication of services into a catalog/registry, as well as for the monitoring of the interoperation and use of services. Next, we will deploy and test the integration of services to ensure effective and efficient interoperability. Finally, we will provide support for managing services as well as establish a quality assurance and measurement framework for ensuring the overall quality of our SOA platform.

Path Point 4: Craft and institutionalize an on-going educational program

In order to ensure continuous success of the *strategy* enterprise, we will craft and launch a customized training program for all people involved. Various key subject matter experts (SMEs) will be selected and invited to help us establish a comprehensive cutting-edge educational program across the LVC unit. With this on-going program we will provide an opportunity for SOA, cloud computing, and virtualization training to our developers, testers and managers of services as well as to all stakeholders involved in the *strategy* effort. This educational endeavor will entail training activities before, during and after the execution of the SOA migration effort.

THE NEXT GENERATION OF DISTRIBUTED TRAINING ARCHITECTURE ROADMAP

The current LVC training architecture environment falls short of the stated goal of the Secretary of the Army and Chief of Staff, Army to *“build an Army that is a versatile mix of tailorable and networked organizations...”*

The drivers for this are far and wide reaching, stemming from the need to:

- Deliver better training capabilities to the Operating Force and the Generating Force.
- Expedite the availability and integration of new commercial technology and associated services.
- Institute an enterprise management perspective in a cost-effective manner.
- Provide the right information and tools to enable informed decision making on critical issues.
- Preserve materiel and human resources by optimizing business operations and consolidating and/or eliminating duplicative and non-essential applications.

The imperatives to transform are not just recognized by senior Army leadership but by Congress as well. Language in the Fiscal Year 2009 National Defense Authorization Act (NDAA) not only mandates transformation of business operations, but also prescribes the architecture roadmap as an integral component of transformation (U.S. Army, 2010):

“[develop] a well-defined enterprise-wide business systems architecture and transition plan encompassing end-to-end business processes and capable of providing accurately and timely information in support of business decisions of the military department.”

The use of architecture roadmaps as a means of transformation is not new a new concept in the Army. Many architecture roadmaps representing various business segments (e.g., finance, logistics, acquisition, personnel, training, and health) exist which are being maintained and evolved by the respective functional organization. By focusing on the functional organization rather than the enterprise, these architectures have not been fully applied to maximize their transformational benefit.

Currently, there are limitations that prevent users from effectively and efficiently accessing and utilizing training resources consistently within the existing Army training environment. For example, training data access, identity management capability, and collaboration ability changes dramatically when a user moves between Combat Training Centers and Homestations or moves to the Regional Simulation Centers. This reduces the ability to work effectively and efficiently, impacting decision-making ability and responsiveness. The challenges associated with accessing and utilizing enterprise information systems, training assets and applications, and network resources is known as the plug-and-play problem. It can be greatly reduced by a transformed Next Generation of Distributed Training Architecture with unified network access, common identity, collaboration, common enterprise information services, and management tools.

Transforming the Army's LVC training environment from "as-is" to the "end-state" will require significant changes involving many Army organizations and processes. This roadmap must provide guidance to the leaders of these organizations and proponents of these processes to ensure unity of effort. Because transformation calls for fundamental changes in how we work (business models and processes) and alters the products and services (solutions), risk management strategies should also be considered.

CONCLUSION

A well structured SOA, cloud computing, and virtualization roadmap has the ability to offer numerous benefits. An SOA can enable full interoperability among training system through reusable, cohesive, and loosely coupled software services. Cloud computing can provide distributed and stored data on-demand. And virtualization can reduce hardware footprint, while improving database server organization. However, as described earlier, there are challenges that must be addressed with respect to security, information assurance, technical performance, and acquisition processes. But with a fundamental

strategy and architecture migration and evolution roadmap, many challenges can be overcome, and many risks mitigated.

The implementation of SOA, cloud computing, and virtualization concepts and technologies within LVC training is the U.S. Army's strategic plan to migrate toward a distributed web-based training environment. An architecture migration and evolution strategy for developing the Next Generation of Distributed Training roadmap will presumably lead to enhanced training capabilities, improved Warfighter training effectiveness, reduced total system lifecycle costs, and improved technology insertion and obsolescence planning. The next steps toward executing the architecture migration and evolution strategy are to work through the path points described earlier, further the Next Generation of Distributed Training concept, and align with the Army's Common Operating Environment and Army Enterprise Network.

ACKNOWLEDGEMENTS

The authors wish to thank Col. Michael Flanagan, Project Manager Training Devices (PM TRADE), Mr. Rob Parrish, PEO STRI Deputy Director of Engineering, and Mr. Brian Kemper, Chief Engineer, for their leadership, motivation, and expertise in support of the Next Generation of Distributed Training concept. We would also like to thank Mr. Victor Haddock, Mr. Graham Fleener, and Mr. Chris Jackson for their subject matter expertise.

REFERENCES

- Bieberstein et al. (2007). "Executing SOA: A practical Guide for the Service-oriented Architect". (Paperback), IBM Press books, 978-0132353758.
- Dumanoir, P., Rivera, J. (2005). *Live Training Transformation (LT2)-A Strategy for Future Army and Joint Live Training*. Orlando, FL: Interservice/ Industry Training, Simulation, and Education Conference.
- Erl, Thomas. (2007). *SOA Principles and Service Design*, Upper Saddle River: Prentice Hall PTR. ISBN 0-13-234482-3.
- Erl, T. (n.d.). *Fundamental Cloud Computing*. Vancouver, Canada: CloudSchool.com
- Garcia Gomez, M., Kehr, T. (2010). *Leveraging Service Oriented Architectures (SOA) within Live Training: An Assessment*. Orlando, FL: Interservice/Industry Training, Simulation, and Education Conference.

- Gartner, Inc. (2007). *SOA Governance*. Whitepaper. Retrieved from http://www.gartner.com/whitepapers/SOA_Governance.pdf
- IBM. (2008). "Virtualization and SOA Synergies Propel Competitive Advantages", <http://www-03.ibm.com/systems/virtualization/view/101006.html>. Retrieved on 27 January 2008.
- Josuttis, N. (2007). *SOA in Practice: The Art of Distributed System Design*. Sebastopol, CA: O'Reilly.
- Khan, Iqbal. (2010). *Address Scalability Bottlenecks with Distributed Caching*. Retrieved from <http://msdn.microsoft.com/en-us/magazine/ff714590.aspx>
- Lewis et al. (2008). *SMART: Analyzing the Reuse Potential of Legacy Components in an SOA Environment*. Retrieved from <http://www.sei.cmu.edu/library/abstracts/reports/08tn008.cfm>
- Linthicum, David. (2010). *How to Gauge Cloud Computing Performance*. Retrieved from <http://www.infoworld.com/d/cloud-computing/how-gauge-cloud-computing-performance-722>
- Salsburg, Michael. (2007). *Server Virtualization Performance*. Retrieved from <http://www.processor.com/editorial/article.asp?article=articles%2Fp2931%2F31p31%2F31p31.asp>
- Strickland, J. (n.d.). *How Cloud Computing Works*. Retrieved from <http://computer.howstuffworks.com/cloudcomputing.htm>
- Strickland, J. (n.d.). *How Server Virtualization Works*. Retrieved from <http://communication.howstuffworks.com/servervirtualization.htm>
- U.S. Army. (2010). *Common Operating Environment Architecture*. U.S. Army CIO/G-6
- Zetlin, M. (2009). *The Downside to Virtualization: Security Risks*. Retrieved from <http://technology.inc.com/2009/04/01/the-downside-to-virtualization-security-risks/>