Using Business Technologies to Cut Simulation Support Costs

Richard Crutchfield, Adam Heeren
The MITRE Corporation
Orlando, Florida
crutchfield@mitre.org, aheeren@mitre.org

Paul Dumanoir
U.S. Army PEO STRI – PdM WTI
Orlando, Florida
paul.h.dumanoir.civ@mail.mil

ABSTRACT

The use of simulation to achieve training and test and evaluation goals requires technical support staff to execute complex processes and complicated, labor-intensive activities. The need for support increases as the Army interoperates live, virtual, and constructive simulations together.

Program Executive Office for Simulation, Training, and Instrumentation (PEO STRI) developed an enterprise architecture, the Live-Synthetic, Training, and Test and Evaluation Enterprise Architecture (LSTTE EA). One of the goals of this enterprise architecture was to allow stakeholders from the training and test & evaluation communities to manage the linkages between the Technical Reference Architecture (TRA), business processes and governance structures and to facilitate discussions within the Modeling and Simulation (M&S) community. Toward that end, PEO STRI developed a Proof-of-Concept (PoC) software implementation of the LSTTE EA technical reference architecture, also known as the LSTTE Infrastructure Architecture (LSTTE IA).

This paper describes our use of Infrastructure as Code (IaC) and business process modeling in the PoC implementation of the TRA, and how it reduces the need for technical support and expertise. We give an overview of the LSTTE EA and the LSTTE IA and a detailed explanation of the framework that includes the IaC and the business process modeling. Using this approach, we can capture processes and automate the execution of tasks today performed by "touch" labor and technical experts. This is not to say we eliminate all human tasks. Certain tasks require direct human decisions or input and are not fully automatable. In our experience, these decisions are generally operational in nature and do not call for great technical expertise in the underlying training simulation systems. The paper presents a comparison of the technical support and expertise needed to conduct training today and how the PoC reduces the need for this technical support and expertise, thereby also reducing cost.

ABOUT THE AUTHORS

Richard Crutchfield is the Army Training and Learning Technology Group Leader at The MITRE Corporation. He has been part of the U.S. DoD M&S community since 1992 and has been at MITRE since 2007, starting out as a Modeling and Simulation Engineer supporting the Joint Land Component Constructive Training Capability (JLCCTC). He has worked as a system engineer, software architect, and software developer on Joint and Army training and analytical simulations systems, including CSSTSS, FSCATT, WARSIM, JLCCTC, and LVC-IA. He holds a B.S. in Mathematics with a concentration in Computer Science.

Adam Heeren works as a developer for The MITRE Corporation's Army Training and Learning Technology Group. He has been a professional software developer for 5 years and has been with MITRE for a little more than a year. His activities center on implementing industry-standard practices and technologies within MITRE efforts. He earned his B.S. in Computer Engineering from the University of Central Florida in 2013.

Paul Dumanoir is the Chief Engineer for the US Army Product Manager for Warrior Training Integration (PdM WTI) under the Project Manager for Integrated Training Environment (PM ITE) at PEO STRI. He currently manages WTI modernization risk reduction projects and the synthetic product line and has 30 years of experience working in DoD simulation and training programs as Product Manager, Project Director, and Systems / Software Engineer. His current interests include component-based product-line engineering, enterprise architectures, and system of system integration and interoperability. He earned his B.S. in Electrical Engineering from the University of South Alabama in 1987 and his M.S. in Computer Systems from the University of Central Florida in 1991.

Using Business Technologies to Cut Simulation Support Costs

Richard Crutchfield, Adam Heeren
The MITRE Corporation
Orlando, Florida
crutchfield@mitre.org, **aheeren@mitre.org**

Paul Dumanoir
U.S. Army PEO STRI – PdM WTI
Orlando, Florida
paul.h.dumanoir.civ@mail.mil

INTRODUCTION

Per (Moore, 2016), a contributor to Smarter with Gartner, "automation improves accountability, efficiency and predictability while reducing cost, variability and risk." By the same token, the lack of automation of information technology (IT) processes supporting training, test and evaluation, and other activities drives up cost, variability, and risk. If the government is to fully benefit from the use of modeling and simulation (M&S), the M&S support activities require systematic automation.

The Program Manager for Integrated Training Environment (PM ITE) developed the Live Synthetic Training, Test and Evaluation Enterprise Architecture (LSTTE-EA) (Johns Hopkins Applied Physics Laboratory, 2015), with goals that included:

- Reduce operational complexity
- Reduce operations and sustainment costs
- Increase agility
- Reduce development costs
- Increase consistency of outputs and outcomes across [Army] training and test & evaluation
- Streamline IA [information assurance] re-certification.

The enterprise architecture includes a Technical Reference Architecture (TRA) to institutionalize best practices and standards for improving IT automation for development, operations, and maintenance.

This paper gives a brief overview of LSTTE-EA. It describes the use of simulations to create a synthetic environment that supports training, test and evaluation, and other activities. It focuses on the supporting activities required to deploy and use simulation. More specifically, the paper examines use of Infrastructure As Code (IAC) and business process management (BPM) to reduce the need for technical support and expertise. We give a detailed explanation of the framework that includes these business technologies. This approach transforms tasks now executed manually by "touch" labor and technical experts into automated tasks. This is not to say it eliminates all human interactions. Certain tasks require direct human decisions or input and are not fully automatable. The paper presents a comparison of the technical support and expertise needed to conduct training and how systematic automation reduces the need for technical support and expertise, thereby reducing cost, variability, and risk.

LSTTE ENTERPRISE ARCHITECTURE OVERVIEW

The LSTTE-EA provides an objective framework for the enterprise architecture, the initial governance approach, and the business architecture. As stated in (Johns Hopkins Applied Physics Laboratory, 2015), "enterprise architecture will provide a clear frame of reference that will allow stakeholders . . . to understand the linkages between the technical architecture and the business and strategic objectives, as well as facilitate discussions within and between the [Modeling and Simulation] communities." As shown in Figure 1, the LSTTE-EA consists of several important layers. The top layer articulates the vision of the enterprise. The business architecture layer includes the business models and components needed to meet the needs of the LSTTE-EA. The governance layer contains policy activities. The reference architecture layer defines the technical architectural template for implementing solution architectures.

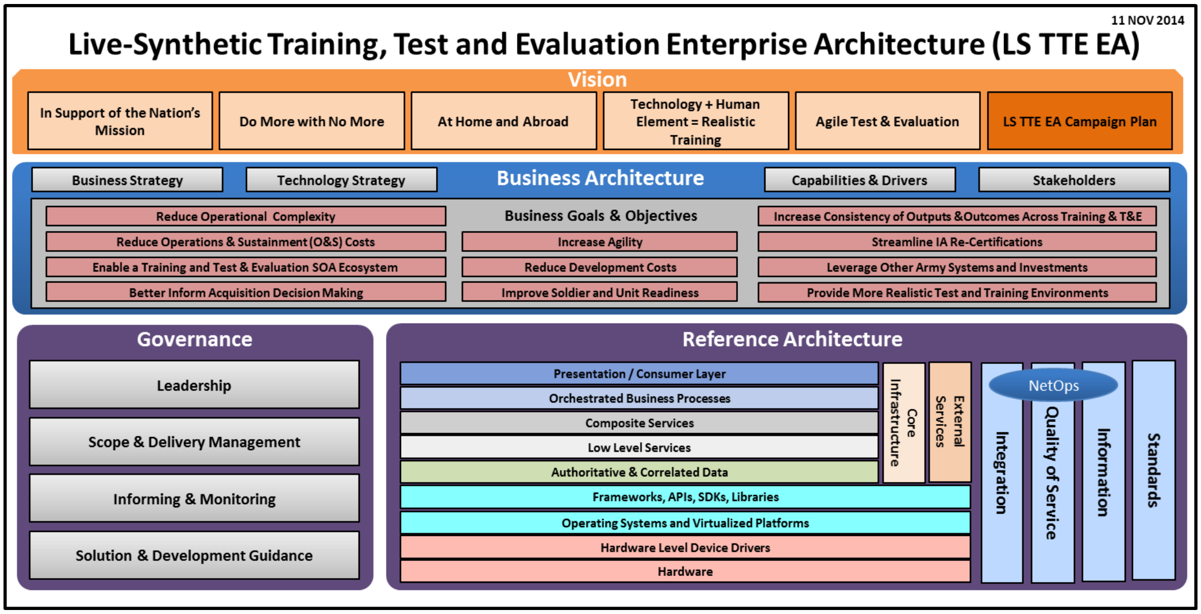

Figure 1. Live Synthetic Training, Test and Evaluation Enterprise Architecture

LSTTE INFRASTRUCTURE ARCHITECTURE OVERVIEW

PM ITE developed a Proof of Concept (PoC) software implementation of the TRA, also known as the LSTTE Infrastructure Architecture (LSTTE IA). A key assumption is that any TRA-compliant instance deploys in a cloud computing environment (CCE). These CCEs comply with the Army Common Operating Environment (COE)[1]. The LSTTE-IA is a layered architecture (see Figure 2): one vertical and six horizontal. Within these layers, the infrastructure components are categorized as either core infrastructure or shared assets components. The core infrastructure components are essential to delivering the LSTTE-IA capability. The shared assets components, while not essential to the delivery of LSTTE-IA, are capabilities used by two or more system compositions. The LSTTE-IA core infrastructure components implements the business technologies (i.e. IaC and BPM). Note that additional capabilities exist with LSTTE-IA but are not covered in this paper. These following subsections provide a brief description of each layer.

[1] Per the Army's Chief Information Officer (CIO)/G-6, "The Common Operating Environment (COE) is 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."

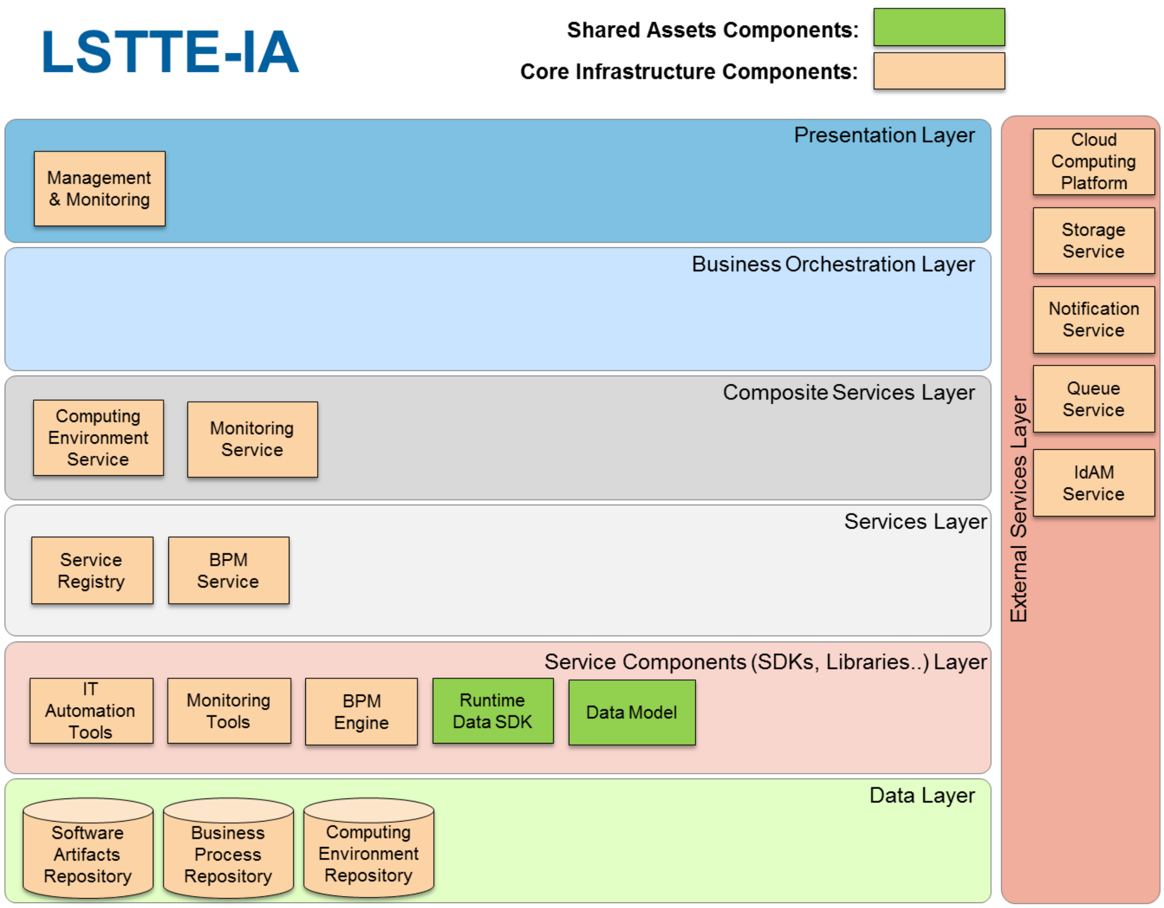

Figure 2. LSTTE-IA Proof of Concept Solution Architecture

External Services Layer

The vertical layer, external services, shows externally provided services necessary to establish an instance of the RA. These services are external capabilities provided via the CCE and other authorized providers. The RA description document (The MITRE Corporation, 2017) assumes services provided by a surrounding CCE are accessed via this layer. Examples include the use of a cloud service provider's Application Programming Interfaces (APIs) (e.g., the Elastic Cloud Computing API of the Amazon Web Service GovCloud) and the VMWare APIs that adhere to appropriate standards (e.g., Cloud Infrastructure Management Interface). Another example is use of the Global Directory Services (a.k.a. DOD PKI DOD411) solution that will be available to use for Identity and Access Management (IdAM).

Data Layer

The Data layer shows data sources internal to the LSTTE-IA. For the purposes of this document, 'internal' means under the governance of the PM ITE. Data sources outside PM ITE governance (external data sources) are accessed via the External Services layer. The Data layer contains three artifact repositories supporting LSTTE-IA core capabilities:

- Software artifacts
- Business process model artifacts
- Compute environment artifacts.

Service Components Layer

The Service Components layer contains software development kits (SDKs), libraries, and other applications. These components are characterized as having a well-defined API. The components in this layer are generally designed to be directly included at compile time in other software components and/or services. Components in this layer may be stored in repositories in the data layer.

Low-Level Services Layer

The Low-Level Services layer contains service components that actualize atomic services and provides a functional and technical specification of the service. Generally, at this layer users cannot expect choreographing and/or orchestration of services. Services register with the Service-Oriented Architecture (SOA) Registry for both design time and run-time discovery by other services and component services. The service description generally takes the form of an interface definition (e.g. Swagger spec file or Web Application Description Language file).

Composite Services Layer

The Composite Services layer contains services whose implementation calls other services. Generally, services at this layer are structured as a set of services implemented as a choreography or an orchestration of multiple services. Another general characteristic of this layer is short-lived execution (i.e. relative to a business process that may take months to complete). Like the Low-Level Service layer, Composite Services are registered in the SOA Registry for description, discovery, and integration.

Orchestrated Business Processes Layer

The Orchestrated Business Processes layer contains the business process flows, which are described using the BPMN and are decoupled from the underlying services. Services orchestrated by a controller characterize these process flows. They can include external business rules to customize and control the execution of the process flows, which can support execution over a long period. This layer is where the STRA provides a clear separation of concern between the business and the IT domains. Business analysis experts can define the processes using graphical tools with little or no programming expertise in a domain-specific language. Standardization of rule language is immature now; however, several initiatives are ongoing in this area, such as the W3C's [World Wide Web Consortium's] Rule Interchange Format, the Object Management Group's (OMG's) Production Rule Representation, and Haley Systems' proposed rule language standard called RML.

Presentation Layer

Users interact primarily with the Presentation layer when accessing the STRA. This layer is the point of entry for interactive consumers, including humans and other applications and external services. It provides multiple client-independent channels to deliver functionality for consumption and rendering on client platforms and devices, using standard web interfaces (HTML5, JavaScript) or native applications to support end-user applications.

BUSINESS TECHNOLOGIES

The TRA identifies architectural attributes important to achieve the strategic business objectives. The LSTTE-IA implements two important themes, IAC and BPM, that go toward meeting business objectives. The following subsections discuss each theme in further detail.

Infrastructure as Code

Why is IAC important? (Patrizio, 2015) says, "One of the major trends in IT over the past few years has been increased automation and a concurrent decrease in the need for human or manual intervention." IAC[2] is an infrastructure provisioning process that automatically provisions, builds, and manages resources through code. Setting up and configuring a virtual machine with IAC provides a fast and repeatable process for replicating

[2] IaC is also known as programable infrastructure.

machines. This approach lets code be the system documentation. The infrastructure code is placed under configuration management like other software development artifacts.

Several products are available that support IaC. For provisioning, products such as Terraform[3] and Red Hat® CloudForms[4] allow for automatically provisioning and managing resources (i.e., computational, storage, network). Other supporting products such as Ansible, Chef, and Puppet let the environment be automatically built out (e.g. application installation, configuration, and setup), managed, and monitored.

The LSTTE-IA provides a Compute Environment Service (CES) using the IAC paradigm. This service takes a template of all computational resources, applications, and configurations desired for a given environment. The template is then instantiated in a controlled manner.

This capability helps to address several of the business objectives and goals. The operations and sustainment costs are reduced because human "touch labor" and manual intervention (e.g., a human sitting in front of a terminal manually creating virtual machines, installing applications, etc.) are no longer necessary. With IAC, developers do not have to wait for operational staff members to deploy the environments needed to test, integrate, or instantiate infrastructure. The organization has more agility to modify an environment in code and re-deploy. This lets organizations reduce the "shadow IT"[5] that creeps in when developers build their own solutions and introduces risk into an organization. With the infrastructure under configuration management as code, the consistency of outputs and outcomes increases, since it is known exactly what is deployed in each environment – perhaps one of the more important aspects in reducing risk.

A well-defined infrastructure and application composition and configuration reduces the amount of effort involved in IA re-certification. The IAC baseline serves as the infrastructure documentation and the IT automation tools keep the configuration from drifting. IAC thus provides a strong foundation for helping meet several of the LSTTE-EA goals.

Business Process Management

Organizations must efficiently and consistently achieve stakeholders' goals and objectives. BPM helps organizations focus on the big picture by enforcing uniformity in execution of activities, and assists organizations in executing their vision and mission by being relatively easy to create from scratch or modify. BPM helps to transform an organization into a process-centric, customer-focused organization by eliminating functional silos. When an organization has efficient business processes, it reduces its expenses because it wastes fewer resources. Consistent processes allow organizations to achieve more reliable operations and outcomes because the operations are all codified in a standard language.

[3] See https://www.terraform.io/
[4] See https://www.redhat.com/en/technologies/management/cloudforms
[5] We define shadow IT as IT systems and solutions built and used inside organizations without explicit organizational approval.

MTC USE CASE

Background

The U.S. Army says, "The Mission Training Complex (MTC) provides individual operator training on Army Mission Command Systems (ABCS) and support for collective simulation and gaming based training exercises."[6] Interviews with MTC staff show a human-intensive, manual process for planning, preparing, and executing training events. In fact, the standard lead time on training with live, virtual, constructive, and gaming (LVC-G) support is six months. Such training involves multiple organizations (e.g., operational unit, MTC, range control) and roles (e.g., Simulation Operations officer, S3/Operations officer, MTC technical staff) in coordinating and synchronizing a large amount of resources to achieve a military commander's training objectives.

Tools for Automating the Technical Staff Activities

The following sections show where the LSTTE IA PoC realizes the reduction of technical staff effort. We used BPMN to create the diagrams.

It is not our intent to give a complete tutorial on BPMN, but we do mention a few items to assist the reader in comparing the 'As Is' and 'To Be' diagrams shown later. At a high level, BPMN contains a starting point, activity nodes, sequence flow connectors, control logic, and an end point, labeled in Figure 3.

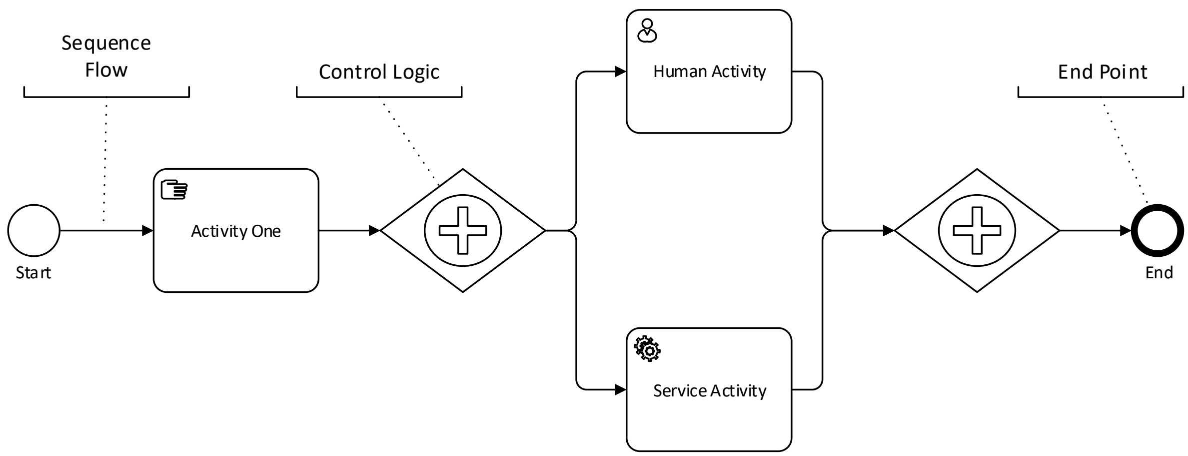

Figure 3. Example BPMN Diagram

Additional points of interest are the icons in the upper left-hand side of an activity node.

- An icon of a human indicates an activity requiring interaction with a human. Such an activity might indicate a human (not necessarily a technical staff member) interacting with the LSTTE IA PoC via a web page.
- A hand icon indicates a manual activity. A manual activity shows the activity that a technical staff member primarily accomplishes.
- An icon in the shape of a gear indicates the activity is an automated service. For example, Terraform and Chef might accomplish the automated service activities.

'As Is' Technical Staff Activities

We provide two diagrams for the reader to compare. Looking back to the MTC use case described above, Figure 4 shows the major activities that technical staff now conduct during the pre-execution phase of a training event. The diagram indicates that all the activities are manually executed (and that they happen more than once). Today, primarily technical staff execute the manual activities (see Figure 3). Of course, we recognize that sometimes non-technical staff can participate in activities such as physically moving hardware resources and similar tasks.

[6] See http://www.knox.army.mil/garrison/dptms/trainingdiv/mtc/Default.aspx

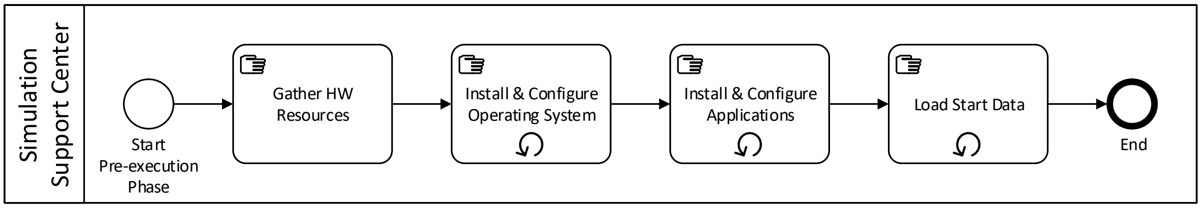

Figure 4. 'As Is' Simulation Center Support Activities

'To Be' Technical Staff Activities

For the same MTC example, the 'To Be' diagram (Figure 5) does not contain any manual processes. The initial step can be conducted by an operational staff member, such as the Battalion S3 staff. The S3 initiates the Request Training Event Support request. This request is made through a 'Turbo Tax'-like wizard that walks users through using operational terms (e.g., support for walk phase of Movement to Contact mission in a training plan), letting the S3 staff tailor it for their mission needs. We see that the request is passing messages to the PoC's Coordinate Training activity. The table icon in this activity indicates that the diagram is using business rules to accomplish its task. These rules would help guide the S3 through how to select the proper training support. The business rules codify the knowledge a technical staff member acts upon to aid the S3 in determining the needs associated with creating the training environment. After the coordination of the request is complete the S3 simply waits for notification from the PoC that all resources are available and ready for the identified training. As shown in the LSTTE IA PoC lane, after coordinating training the system automatically accomplishes all the previously manual steps to get ready for training. The LSTTE IA PoC then sends the S3 a notification that the pre-execution phase is complete.

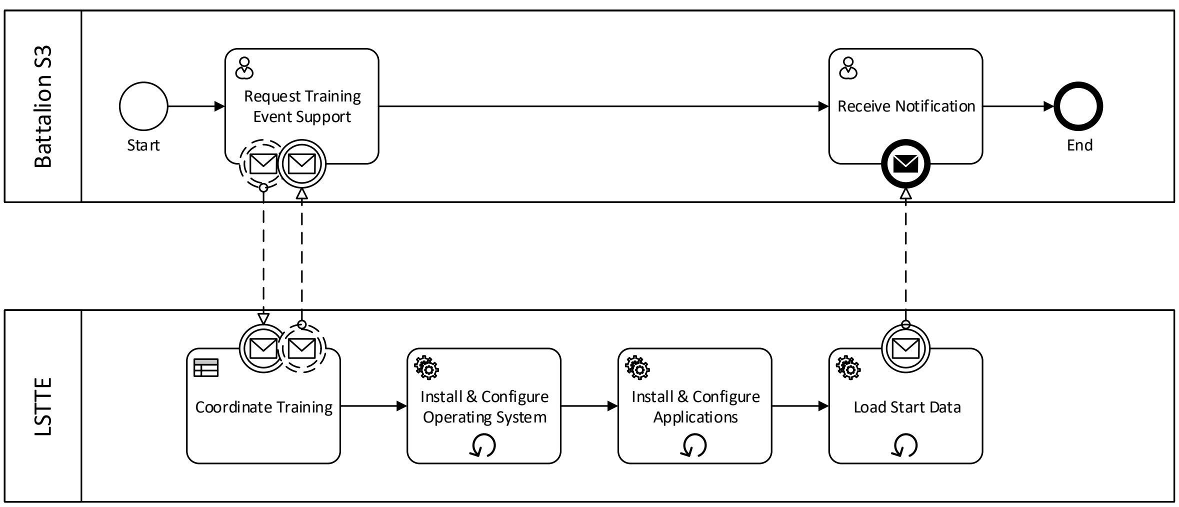

Figure 5. 'To Be' Simulation Center Support Activities

Of course, this is a simplified rendering of the activities needed to accomplish the MTC use case example using the LSTTE IA PoC. We omit many additional details, such as exception handling and interactions with other LSTTE IA PoC-provided services, to ease the comparison between the 'As Is' and the 'To Be' processes. The main purpose was to illustrate that the 'As Is' is a highly manual process whereas the 'To Be' requires no technical staff assistance. This leads to a reduction in both technical staff and time necessary for the pre-execution phase by several orders of magnitude: based on our preliminary findings, we expect the time required to be cut from tens of days to tens of minutes.

For the other use cases that are only fielded once the benefits are not as clear. The LSTTE IA PoC would use the Chef application to automate the installation and configuration of the initial build of the gaming systems and to maintain them in configurations. The LSTTE IA would also use Chef to install patches (e.g., security updates). Other functions might include pushing out a desired game scenario for training to reduce the need for technical staff and expertise. However, we propose (The MITRE Corporation, 2016) a Virtual Desktop Infrastructure (VDI) with Graphics Processing Unit (GPU) support hosted at the Installation Processing Node (IPN). The LSTTE IA can leverage this IPN infrastructure to more fully automate the classroom environment. Thin clients connected to the VDI solution via the installation's Local Area Network (LAN) would reside in the classrooms. In this scenario, we treat the classroom in the same way as the MTC example. With this deployment model, the classroom environment can benefit from rapid reconfiguration to support training more efficiently.

We do realize that many other deployment models exist and we do not intend to address them all. One Program of Record (POR) deploys full virtual machine images (VMIs) to a VMWare-based hardware stack. In this case, although some of the drivers of cost seen in the MTC use case are substantially cut, the model is not as scalable, flexible, or efficient as our proposed approach. The POR's approach requires a VMI for each machine; additional machine configurations demand additional VMIs. Each of these VMIs can exceed several hundred gigabytes in size, making them impractical to transport over the network. We believe that most PORs developing LVC-G enablers will benefit from an LSTTE IA deployment model that makes use of current business IT automation technologies.

Enhancing Technical Operations

We worked on the systems engineering team that developed the system of systems supporting the MTC use case. This effort included designing, implementing, and integrating the capability. Thus we have first-hand experience of how difficult it is to operate and maintain the system. We have also served as subject matter experts (SMEs) on numerous actual training events to assist the local supporting technical staff in accomplishing their tasks.
Here we present one example, focused on the complexity of isolating and correcting system anomalies. If we detected an anomaly, we worked together with the local technical control staff to isolate the issue and recommend corrective courses of action. In one case of a common issue often caused by degraded network conditions, the systems engineering team developed a flow chart documenting the steps involved in resolving the issue. The lead technical staff member taped this flow chart to the desk at his station, yet even with the flow chart available, the technical support staff did not always accurately follow the steps, and the system did not recover from an anomaly as efficiently as it could have.

The following two subsections illustrate a simplified version of the "flow chart" example. They show how the use of executable business process models and business rules can significantly enhance technical operations and further reduce costs.

'As Is' Technical Operations

In the current MTC training event use case, technical staff members must monitor several sources to ensure the training system is operating within acceptable parameters (see Figure 7). They continuously monitor a display to either 1) receive notification of an issue or 2) review data to deduce that a system issue exists. They also receive notification from other technical staff members experiencing issues (e.g., a network technician indicates high latency on circuits). After many days of 14-hour shifts human fatigue and routine understandably can diminish the efficiency of this human centric process.

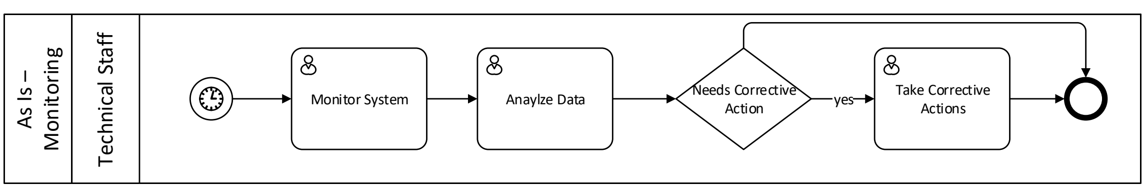

Figure 6. Notional 'As Is' Monitoring Activities

'To Be' Technical Operations

Figure 8 shows a fully automated version of system monitoring, analysis, and corrective actions (if needed). The LSTTE IA PoC implements this version. It automatically gathers metrics from the system, analyzes the metrics using a set of business rules, and processes the results of the analysis to determine if further action is needed. The process automatically takes no action, stops services, and/or flags services running for potential human interactions.

The LSTTE IA PoC then notifies the technical staff of the results. The underlying business process engine starts another iteration of the process on a periodic basis.

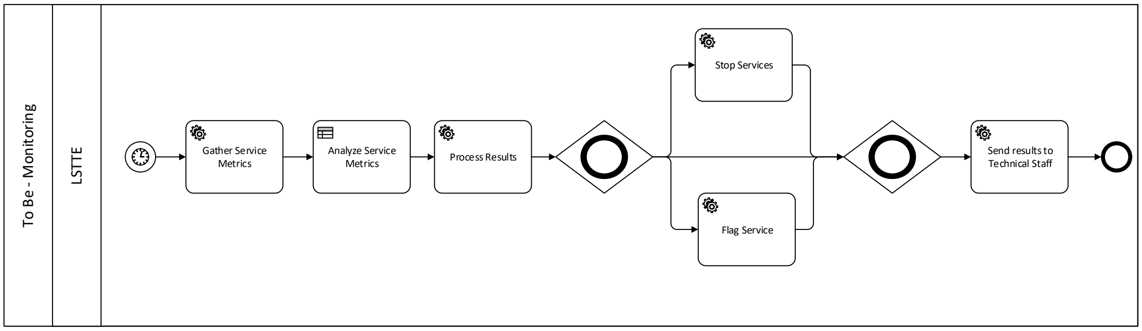

Figure 7. Notional 'To Be' Monitoring Activities

The automatic nature of this process alleviates the need for a human to constantly monitor the system visually for anomalies. It also takes the appropriate corrective actions to bring the system into acceptable performance parameters without introducing human error. This monitoring approach frees the technical staff to focus on other tasks or potentially to monitor multiple training events simultaneously.

CONCLUSION

The PoC Business Orchestration layer implementation within the LSTTE IA demonstrates that in a laboratory environment the use of business technologies can reduce the complexity of operations and the need for technical staff. The IAC provided by the LSTTE IA mitigates the occurrence of human errors in the pre-execution phase and lowers the need for "touch labor" to execute simulation-supported training. The built-in capability to monitor the system and automatically take corrective actions leads to a reduction in the need for technical staff. We recognize that the LSTTE IA POC has not been exercised in an operationally relevant environment and that the overall architecture must mature further. However, we believe that the capabilities demonstrated to date suggest that the Army can realize the LSTTE EA Vision and Business Goals & Objectives.

As part of modernization risk reduction activities, PM ITE plans to inject these business process architectural patterns into PORs beginning in FY17, to help reduce the complexity and cost of existing training systems of systems. The architectural technology injection will occur methodically and systematically to ensure efficient solutions are fielded that meet specific system requirements and Soldier needs. In addition, the T&E M&S community and the operational test community are working closely with PEO STRI to leverage these solutions and apply them to new operational test systems such as the Integrated LVC Test Environment (ILTE). PM ITE also continues to explore and implement innovative business models and governance approaches within the LSTTE EA framework to support existing and new collaborative adopters of the LSTTE IA framework.

REFERENCES

ASA(ALT). (2013). *ASA(ALT) Common Operating Environment DataCenter/Cloud Computing Enviroenement Architecture Compliance v 2.0.*

Johns Hopkins Applied Physics Laboratory. (2015). *The Business Architecture for the Live-Synthetic Training and Test & Evaluation Enterprise Architecture.* Laurel: Johns Hopkins Applied Physics Laboratory.

Moore, S. (2016, May 18). *Automation: The Next Frontire for IT*. Retrieved from Smarter with Gartner: http://www.gartner.com/smarterwithgartner/automation-the-next-frontier-for-it-2/

Patrizio, A. (2015, December 22). *What is 'infrastructure as code' and why should you embrace it?* Retrieved from CIO from IDG: http://www.cio.com/article/3017722/infrastructure/what-is-infrastructure-as-code-and-why-should-you-embrace-it.html

Rosen, R. J. (2015, September 3). *In Praice of Sort-Term Thinking*. Retrieved from The Atlantic: https://www.theatlantic.com/business/archive/2015/09/jobs-automation-technological-unemployment-history/403576/

The MITRE Corporation. (2016). *'Point of Need' Study for PM ITE.* Orlando: The MITRE Corporation.

The MITRE Corporation. (2017). *Synthetic Simulation Transformation (S2T) Synthetic Technical Reference Architecture Description Document.* Orlando.