Leveraging Service-Oriented Architectures (SOA) within Live Training: An Assessment

Michelle García Gómez
PEO STRI
Orlando, Florida
Michelle.K.Garcia1@us.army.mil

Thomas Kehr
PEO STRI
Orlando, Florida
Thomas.Kehr@us.army.mil

ABSTRACT

Defense budgets are shrinking and life-cycle savings approaches are more important now than ever. The Live Training Domain encounters numerous challenges in maintaining and upgrading ranges and training equipment to align with technology advances. Today, Live Training Ranges (e.g., instrumented and live fire ranges) consist of a number of stovepipe architecture systems and components, which restricts their reusability by other ranges and diminishes the list of vendors from which compatible replacements and upgrades could be attained. Incompatibility between disparate systems forces removal and replacement of functioning components or the addition of adapters, which can lead to significant increases in cost and development time. Making modifications to components, whether by adding, removing, or upgrading, usually results in range downtime, which reduces valuable training opportunities for the Soldier.

This paper will assess the values, risks, and methods for adopting a modular Service-Oriented Architecture (SOA) framework into an existing Product Line or Family of Training Systems, and will highlight several current implementations of SOA and their effects within the Live Training community. The interface standards defined and established at the onset of a SOA allow for interoperability between various distinct services. The inherent interoperability resulting from a well-structured SOA allows for reusability of its services across other programs as well as backwards and future compatibility of vendor-neutral components. Overall, the implementation of a platform and language independent SOA promotes reusability as well as agile and cost-effective system development. The features inherent in a SOA make it ideal for use in the Live Training domain where technologies are constantly emerging and evolving. A well-structured SOA would be congruent with and extend the existing architecture and product line investments within the training framework.

ABOUT THE AUTHORS

Michelle García Gómez is a systems engineer supporting the Product Manager for Ground Combat Tactical Trainers at the U.S. Army Program Executive Office for Simulation, Training, and Instrumentation (PEO STRI). She previously supported the Project Manager for Training Devices (PM TRADE) in the development of service-oriented architectures (SOAs) within the Live Training Transformation Family of Training Systems (LT2-FTS). Ms. García Gómez holds a B.S. in Mechanical Engineering from the University of Central Florida and is an Acquisition Academy (A2) graduate.

Thomas Kehr is a systems engineering intern supporting the Project Manager for Training Devices (PM TRADE) within the U.S. Army Program Executive Office for Simulation, Training, and Instrumentation (PEO STRI). Mr. Kehr is currently on a rotational assignment at the Simulation, Training, and Technology Center under the Army's Research and Development Command (RDECOM) where he is working as a Liaison Officer between PM TRADE and RDECOM to develop technology roadmaps and transition strategies between the two organizations. Mr. Kehr holds a B.S. in Electrical Engineering from the University of Central Florida and is currently pursuing a M.S. in Electrical Engineering from the same institution. Mr. Kehr's main area of study is in the field of robotics and is actively working to apply this technology to systems within Live Training.

Leveraging Service-Oriented Architectures (SOA) within Live Training: An Assessment

Michelle García Gómez
PEO STRI
Orlando, Florida
Michelle.K.Garcia1@us.army.mil

Thomas Kehr
PEO STRI
Orlando, Florida
Thomas.Kehr@us.army.mil

INTRODUCTION

The latest technology-focused process to synergize information and products is known as Service-Oriented Architecture (SOA). SOA, pronounced "so-uh" or sometimes S-O-A, is quickly being adopted by a number of government agencies as well as industry due to the strategic benefits that it provides and its ability to address inefficiencies in Information Technology (IT) and interoperability. SOA is the next emerging pseudo-technology process approach aimed at life-cycle control and management and represents viable opportunities within the Live Training environment.

CURRENT LIVE TRAINING DOMAIN CHALLENGES

One of the greatest problems facing Live Training today is the obsolescence of training technology and infrastructure. As older technology becomes obsolete, the supply of spares for fielded systems decreases. Maintaining these systems becomes increasingly difficult as spares become scarce and compatible substitutes are no longer available. The Department of Defense (DoD) must then appropriate funds to modify and integrate newer technology into the older systems in order to sustain them.

For example, if a target controller is beyond repair and its technology is obsolete, ranges must procure a new target controller. With a different target controller, the ranges will likely also need to procure and integrate new target control software, which is an additional cost burden on the ranges.

Another significant issue facing Live Training deals with system upgrades. Where maintenance alone is not viable, a system may need to be upgraded, which also results in significant expenses.

For instance, if a range needs to utilize new target control software, the range may find it more cost effective to upgrade the entire range to new target controllers and target control software instead of trying to integrate the older target controllers with the new target control software. However, this results in the loss of the still-functioning, older target controllers in addition to the costly upgrade. In other situations, a range may need to develop, test, and install adapters in order to integrate the older target controllers with the new target control software.

In either case, an upgrade could lead to a significant amount of range downtime, which would negatively impact training time and throughput. The financial cost may be manageable, but the cost of training time for the Soldier is irreplaceable.

As DoD doctrines mature and information assurance is enhanced, the older, existing systems and technologies are not capable of supporting the current standards. In addition to those new directives, Live Training must constantly stay abreast with the latest training technologies as they drastically advance. These swift changes often result in drastic overhauls of older, existing systems as well as exorbitant costs.

Adding to the challenges, the DoD is migrating toward a Live-Virtual-Constructive integrated architecture to enhance training opportunities for Soldiers. With this migration, Live Training not only has the challenge of being interoperable with the smaller scale exercises associated with Virtual Training, but also with the larger scale maneuvers found in Constructive Training. This requires a broad range of programs to provide time and funds to support interoperability across the training spectrum, which is an ongoing DoD goal. The implementation of SOAs within Live Training allows each of these challenges to become an opportunity. When the SOA methodology is used to meet these challenges, the SOA can become a mechanism to facilitate a requirement from concept to fulfillment.

BRIEF OVERVIEW OF SOA

Due to the ever-evolving nature of the SOA standard, there is not a distinct and globally-accepted definition. A commonly accepted description of a SOA is a software approach used to define an interface, which allows for interoperability between separate self-contained services. While predominately considered as simply an IT approach, a SOA can also be thought of as a more abstract approach to structurally organizing and composing common efforts. When regarded as such, it becomes easier to envision general tasks as services for potential SOA applications.

The service is the basic building block of a SOA. A service often refers to a collection of related capabilities that are needed to perform a certain task. The services are transparent to the SOA, in that a SOA does not require an understanding of how a particular service accomplishes its task, only that it can accomplish the task. The services and overall SOA can be language independent because the services are transparent. Within a SOA, the service and service consumer could be any imaginable entities such as a person, a system, application, or even another service. Communication between these services and consumers is achieved through a service contract.

In a SOA, a contract refers to an agreement between the consumer and service provider that establishes a common interface standard for the service request and response. The contract includes service expectations such as service availability, reliability, performance indicators, and cost. The contract lasts for the duration of the service transaction, provided that the contract does not change, and could last an indefinite amount of time. This does not mean that the service cannot change; the service can be upgraded and the service provider can be changed without breaking the transaction or changing the contract.

VALUE OF ADOPTING A SOA

If the time is taken to develop a well-structured SOA, its features will allow for a highly distributed development and support system. It is imperative for success that any SOA be properly architected and developed. Design must be open but bounded, and its features must allow for loosely coupled services that are interoperable, reusable, and composable.

Interoperability is the ability to have services communicate with each other without the need for middleware. It results from initially defined interface standards and utilization of open standards. Many of the benefits that a SOA provides stem from this ingrained interoperability between services. This interoperability allows for proactive and responsive service development. In theory, a service could be developed for a projected need in the future or in response to a need in the present. In either case, the service would be interoperable with other services in the SOA. Interoperability also allows for the incremental addition of supplementary capabilities to a service. For example, a training range could upgrade one of five pan-tilt (PT) cameras to a pan-tilt-zoom (PTZ) camera and be able to utilize its added capability of zoom without needing to upgrade to a new camera control service. Not only does interoperability increase manageability, ensure backwards compatibility, and save development time and costs, but it also allows for the reusability and composability of services.

Reusability is a direct result of the interoperability ingrained within a SOA. This characteristic allows for previously created services to be reutilized by a different consumer and is the key to the SOA mantra, "Write one time, use many times". Figure 1 illustrates three unique development projects using previously existing services as well as adding their newly created services to the service database to be reused by other projects. Utilizing these previously created services saves development time and costs.

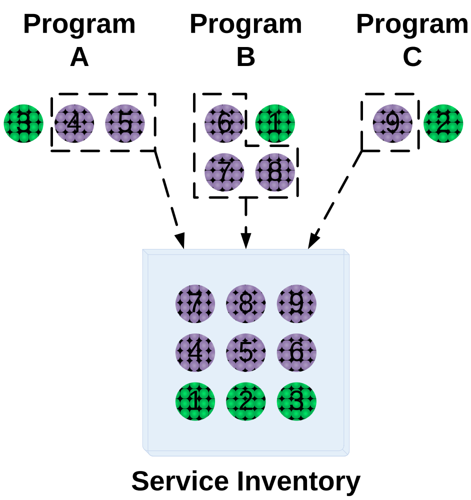

Figure 1. Reusability of Services

Composability is another result of interoperability that allows for a relatively effortless combination of services for applications. It is often the case that services are composed to allow for a compilation of services to create a new and more comprehensive service. Figure 2 illustrates the process of creating a new service from previously created services in the service inventory, which also saves development time and costs.

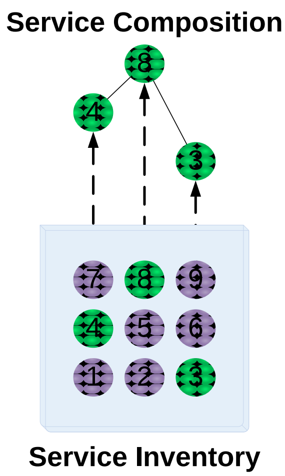

Figure 2. Composability of Services

Loose coupling results from time-invested selections of the services, standards, and frameworks that are desired for a particular SOA. It allows for service providers that make no assumptions of the consumers' purposes. These loosely coupled services can be changed without modifying the consumer as well as upgraded without inhibiting other services. For example, a range could replace a piece of damaged hardware without changing the control system for said hardware or requiring additional range downtime. In a truly loosely coupled SOA, the system makes no prejudices about the hardware and replacements would be a plug-and-play solution.

A comparison of tight and loose coupling is illustrated in Figure 3. If a service becomes obsolete in a tightly coupled architecture, the client must search for, request, and re-establish a new service, which requires additional time and effort. If the Live Training service provider modifies, updates, or replaces a service, the change will be transparent to the client and require no additional effort from the client.

Loose coupling also allows consumers to utilize services not only through the service provider, but also outside of the service provider, thus offering greater versatility within the system.

Each of these characteristics (interoperability, reusability, composability, and loose coupling) yield benefits throughout a system. However, when these features are combined within the construct of a SOA framework, they yield greater returns than the individual parts. In addition to complementing one another, these characteristics allow for an easier implementation of their features and a more comprehensive approach to system development.

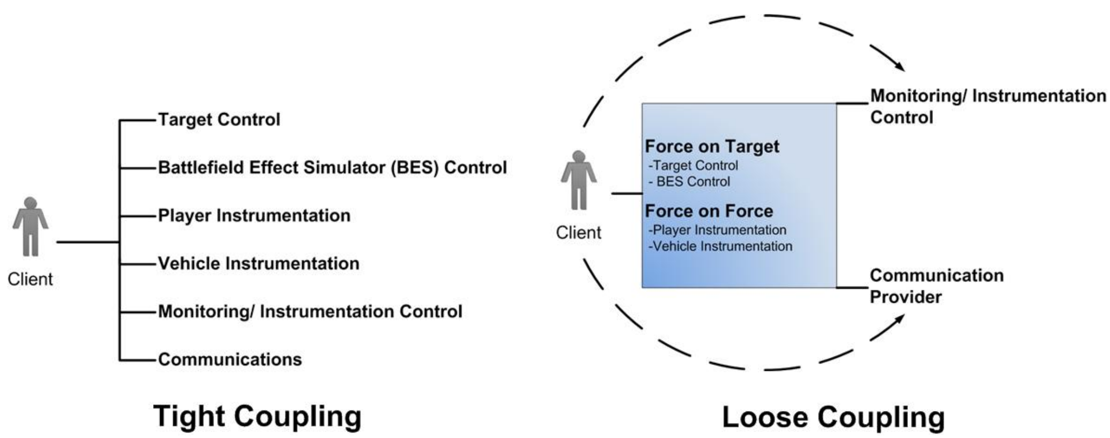

Figure 3. Tight Coupling versus Loose Coupling of Services

RISKS OF ADOPTING A SOA

Adopting a SOA is not an overnight process. Along with the benefits that a SOA can provide, there are also several risks to consider. Careful planning and organization is required to mitigate many of these risks.

In most cases, adopting a SOA will require increased costs during the initial phases of adoption. These costs can be attributed to the training, process, and organizational changes necessary to implement the SOA. Once a sufficient number of services have been added to the service inventory, these costs will start to decrease as a result of reusing and composing services. Due to the evolving nature of SOAs, there are also concerns of long-term costs associated with their evolution. As organizations are working to define a SOA, many fear that their already implemented SOA infrastructure will become obsolete and require costly upgrades or replacements.

A common misconception organizations make when adopting a SOA is that it can be implemented as an “Out-of-the-Box” solution. This is simply not the case with SOAs. Careful planning must be taken to correctly select the services and frameworks necessary to complete your goals. Because of this, a SOA requires a dedicated architect to take ownership of the SOA and invest the time and effort needed to ensure its success. It is the role of this architect to develop a SOA governance plan that defines the goals, strategies, and constraints necessary to mature the SOA, because without one, an organization risks reverting to a traditional distributive type of architecture. As standards associated with SOA are still being defined by standardization organizations, architects must take this into consideration when developing the governance and transition plans.

ADOPTING A SOA

The first and most important step for SOA adoption is to define the goals and strategies needed to accomplish the training mission. The goals and strategies should be aligned with current training requirements and logistics goals in order to provide the greatest value and cost efficient implementation. It is important to remember that there is not a “one size fits all” SOA strategy and that, once your goals are in place, you must define the standards, policies, and procedures that align to the current project or training application. Utilizing this approach, SOA maturity could show trends of a wave-like evolution over time.

Most organizations will begin their SOA implementations utilizing a project-based approach, meaning that all efforts and tools leveraged are within the scope of a specific project. This allows for organizations to see a success with a SOA before they invest the time and effort needed to align their business model to a SOA-based approach. A service inventory will begin to rapidly grow as more projects are implemented and additional tools and infrastructures are added to realize project-based requirements.

As organizations gain a sufficient amount of services within their inventory, the organization can evolve their SOA implementations from a project-based to a service-based approach. In a service-based approach, an organization identifies the most heavily shared services and invests more resources in developing and managing them. Utilizing a service-based approach has been demonstrated in industry to yield a higher return of investment, because it allows organizations to accurately identify the services and processes that have the highest impact and value to them. As reuse of these services increases, governance must also increase, because each step in SOA maturity brings more sophisticated policies and procedures in order to provide an efficient use of services across the training environment.

Without being able to measure the progress of a SOA, it is impossible to know how close the SOA is to achieving the prescribed goals. That is why it is important to also define the metrics for success when adopting SOA architecture. Within the Live Training environment, these metrics should align with current training environments, and success should be reflected in the training quality that is provided to the Soldier. As feedback from users and clients is collected, the metrics of the SOA should be matured along with the SOA itself.

Once metrics are defined, it is important to put the necessary governance mechanisms in place. This process represents the physical implementation of the previous steps discussed in this section. It is important to have the majority of one’s governance processes automated in an effort to make the SOA efforts as scalable as possible; however, it is sometimes advantageous to leave some of these processes as manual in an effort to ensure that every department is progressing towards the same goals. If a SOA is left ungoverned, the result will be a disorganized collection of services that make the organization no better than if it had not adopted the SOA in the first place. It is best practice to make education of the SOA governance process a primary focus over the actual enforcement.

As a SOA matures, new governance policies and procedures should be implemented by the organization in order to allow for SOA maturity and goal attainment. As one proceeds forward with a SOA, it may be necessary to refine and re-evaluate one's strategy and objectives in order to stay aligned to current training requirements, organizational goals, and customer feedback.

When a DoD program is adopting a SOA and its governance plan, it is necessary to garner industry-level investment and support; otherwise, it will be hard to deliver the full capabilities of the SOA on a government and industry-wide level. It is also important that the creator of the governance policy refrains from making the governance process too burdensome in order to allow industry's innovation to help guide the SOA towards a common goal.

SOA APPLICATIONS WITHIN LIVE TRAINING

There are many opportunities for well-defined SOAs within Live Training and the Live Training Transformation (LT2) Product Line. The SOAs supplement product lines and other architectures with self-contained, reusable services that allow for rapid customization of products in order to fulfill ever-changing requirements.

One of the areas within Live Training where SOAs offer great potential in reducing development and integration time and costs is video services.

Video SOA

Most Live Training ranges use cameras/video for safety and exercise monitoring as well as capturing video that can later be utilized in After-Action Reviews (AARs). Currently, the cameras, camera controls, and video feed distribution, capture, and database managers vary from one range vendor to another depending on the specific range configuration.

As new camera and video components are added, they may not be compatible with the existing camera control service, thus forcing the range to upgrade to a new camera control service. This forces the other cameras to be upgraded as well, which results in an expensive upgrade, both in material and range time.

The Video SOA was developed to be vendor/device independent with a focus on the core features utilized within Live Training and is defined to fit within the existing LT2 Product Line framework. The Video SOA is a mechanism to organize and utilize a range's distributed video capabilities in a uniform manner to provide compatibility with multiple consumers across a network. It would allow for each of the components in the video system to be replaced or upgraded without affecting the other systems as well as enhance video interoperability between ranges and systems. It would also allow for the incremental addition of capabilities throughout the system's life-cycle.

Figure 4 (on the following page) illustrates the benefit of implementing a Video SOA at an existing training installation. The left side of the image represents an installation before a Video SOA has been implemented. It can be seen that each consumer is directly coupled to a video system without any consideration of interoperability between them. On the right, the Video SOA implementation shows that a single consumer is able to enter into a contract with the service provider, gain access to any number of video systems on the network, and distribute the feeds to any client workstation.

The most heavily leveraged Video SOA compliant system in the Live Training domain is the SIGHT software developed and maintained by Riptide Software in Orlando, Florida. SIGHT has been fielded to a variety of locations to support several major defense programs over the last year. Currently, SIGHT supports video recording and monitoring for the Training Range Automated Control and Recording (TRACR) system, Urban Operations Training System (UOTS), and the Military Operations on Urban Terrain Center of Excellence (MOUT CoE) at Camp Blanding, and it was recently included as the Video System for the newly awarded Digital Range Training System (DRTS) contract.

SIGHT is a video capture solution that leverages the LT2 Video SOA Interface Control Document (ICD) and was designed to meet the evolving requirements of the Live Training community. The software is able to operate in a stand-alone mode or integrate into other LT2 applications (TRACR, DRTS, etc.). SIGHT's ability to readily integrate with these established applications as a result of the Video SOA, is advantageous to range operators looking for a quick and effective way to add video capabilities to their ranges. The Video SOA also allows for scalability that makes SIGHT suitable for both small range exercises and large scale maneuvers.

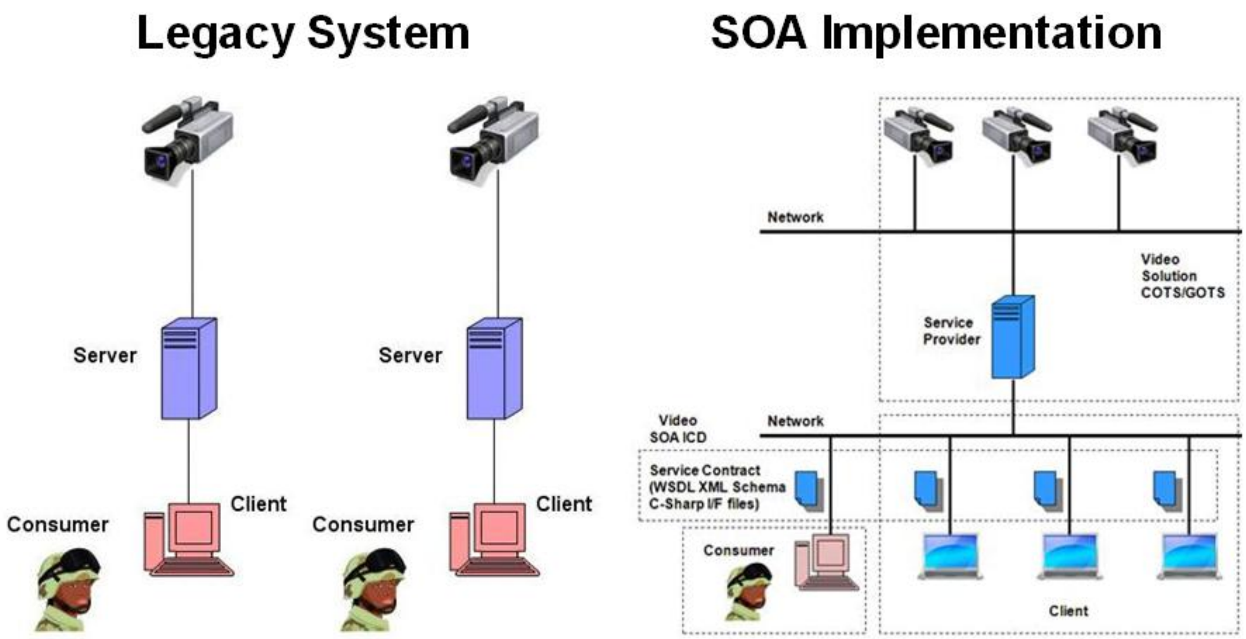

Figure 4. Legacy Video System versus Video SOA Implementation

SOA BENEFITS IN THE LIVE TRAINING DOMAIN

Implementing a well-structured, properly architected SOA can address many of the current Live Training domain challenges.

As programs move forward, they can reutilize services and combine them with other SOA compliant services to fulfill their requirements. This leads to commonality across programs and the advancement of interoperability between them, which results in decreased cost and schedule as well as increased performance.

Cost

Access to an inventory of SOA services, allows programs to search for and analyze previously developed services. A program may find a service that meets some or many of their requirements and can then reutilize that service or, at least, start with a more mature baseline. Instead of fully developing a service from concept to completion, the reutilization of the SOA services decreases the program's developmental costs.

Once developed, the service will likely have to be integrated with existing systems, which requires less effort between SOA compliant services due to the SOA's ingrained interoperability. This interoperability allows for a reduced need for middleware and modifications within the system and therefore, a lower integration cost.

The Infantry Platoon Battle Course (IPBC) at Fort Pickett, Virginia had already upgraded to TRACR, a Video SOA compatible LT2 application, and through the addition of the Video SOA compliant SIGHT software, they were able to implement, at a low cost, a video system at a training environment that did not previously have it.

After fielding, technology advances will lead to newer versions of a system and backwards compatibility will be expected. This backwards compatibility can be more easily achieved through inherent interoperability in the SOA, thus reducing life-cycle costs.

Schedule

Programs have schedules that they need to meet, and minimizing the amount of time between requirements development and fielding is imperative. The faster that a system can be fielded to the Warfigher, the faster they can start training and the longer they have to train. In addition to decreasing costs, reutilizing services from the service inventory and reducing the number of middleware and modifications necessary also decrease the schedule.

When fielding, the integration with other systems is simplified due to the intrinsic interoperability of the SOA. There are also instances in which a specific location requires systems being fielded to be tailored in order to integrate with its systems, and the SOA's interoperability helps preclude that necessity and reduce the deployment time.

To better meet schedule and decrease development costs through schedule reductions, the Video SOA ICD provides use cases for the function and service calls within the SOA. These use cases ensure that all the users of the Video SOA are aligned when implementing the functions and messaging formats within the standard. Feedback has shown that providing these use cases along with the ICD has increased understanding of the Video SOA messaging structure, thus reducing initial development time and lowering development costs.

Performance

The ingrained characteristics of SOA can also work to increase the performance of a system. A system can be made more reliable by ensuring communication between services and reutilizing proven and mature services. Proper communication between services is made possible through the interoperability that is defined at the onset of the SOA. Through this interoperability, a common messaging architecture is used to ensure services are able to properly translate received messages and avoid communication conflicts.

In addition to improving communication, stability can also be improved by implementing proven and mature services into a systems' design, which is possible due to a SOA service's reusability. By utilizing established services from a service inventory, one can be more confident that the resulting system will perform with minimal incident.

The ingrained SOA characteristics become apparent when talking about the Video SOA's performance as exemplified within SIGHT. Most notably, the loose-coupling between services found in the Video SOA allows SIGHT to be scalable or configurable based upon a program's requirements. SIGHT can be tailored for different programs and locations by enabling and disabling individual services and features. This allows SIGHT to be maintained at a single baseline as opposed to multiple program-specific versions. An example can be seen through the deltas between the TRACR and DRTS implementation of SIGHT. Within DRTS, there is a requirement to restrict video sources based on the scenario step and only certain sources are available for viewing. In TRACR, there are often less video sources, and the operator has the ability to change sources and their views at his/her discretion throughout the exercise. Both programs will use the same SIGHT software install, and yet be configured differently to meet their respective requirements.

Life-Cycle

The lifespan of systems can be extended by allowing them to be more easily maintained, upgraded, and grown through the use of a SOA.

As components deteriorate and become obsolete, replacements and spares are necessary to continue the life of the system. The interoperability between SOA services and SOA compliant components provides a wider range of components that are compatible with the system, and thus it is easier to locate suitable part replacements even as the system ages.

The replacement components could also have a greater functionality than the original to allow for increased capabilities that incrementally upgrade the system. The interoperability with the SOA would ensure the advanced components are backwards compatible, and the loose coupling between the services would prevent degradation of the rest of the system. This upgradability and growth allows for older systems to remain relevant and useful as new training requirements are developed.

The efficiency of Video SOA legacy integration was recently seen with the video system change on the DRTS program. The existing DRTS video system utilized an older baseline of the Video SOA ICD (v 1.1). It took less than half an hour to switch to the SIGHT video system, which was compliant with the latest release of the Video SOA ICD (v 3.0). Had the existing DRTS video solution been a non-Video SOA compliant solution, the change would have meant a possible change in hardware, range infrastructure, and more extraneous software integration.

CONCLUSION

Recently, the United States Army has begun to develop several SOA solutions to be implemented in various aspects of the Live Training domain. The most mature of these is the Video SOA. This SOA implementation is being developed by obtaining input from leaders in the training and video system industry to achieve interoperability across the domain. Success of this implementation would pave the way for future SOA executions within Live Training.

The Video SOA, as leveraged by SIGHT, has been in the field now for almost a year and a half. During this last year, the Government and SIGHT developers have received mostly positive feedback about the system. Both military and range personnel prefer SIGHT to legacy systems because of its ability to integrate with other systems. Recently, both TRACR and SIGHT were fielded at Brookhaven Range at Fort Hood, Texas. The legacy video system required users to record video to a VHS tape if they wished to view their AAR. Through Video SOA messaging, the TRACR AAR software can import the videos, which are recorded automatically by SIGHT from the multiple video sources, into an AAR presentation; the presentation can then be transferred to the AAR building as long as there is a connection between the AAR building and the tower. This feature is popular with range operators, because it allows them to focus on running the training event and spend less time setting up video capture and recording.

It should also be noted that implementing the Video SOA is not a solution for every use case. The Video SOA is at an immature life-cycle stage, and there is still much that can be done to improve certain aspects of the standard. This was seen during the testing of a Video SOA system at the Camp Blanding MOUT CoE in Starke, Florida. The MOUT facility had several networked cameras already installed at the location, and the legacy video system was to be removed and replaced with a Video SOA compliant solution. In performing the upgrade, it became apparent that the proprietary communication protocols of the existing cameras would have to be made available in order to integrate with the Video SOA. The manufacturer was unwilling to give up this information at no cost, which forced the contractor to integrate with the cameras through an analog connection. This increased integration costs and delayed the schedule. The Brookhaven Range also discovered that unique encoders for the cameras had to be integrated in order to successfully implement the Video SOA solution. While this is not true for every installation of a Video SOA solution, it generated discussion as to whether all of the SOA benefits can be realized if there seems to still be dependencies on video client implementations. Nevertheless, the Government recommends that careful research of the existing system is done before an integration effort is preformed in order to avoid unexpected system behavior. As the Video SOA is further developed, the Government expects increased vendor support for interfacing with video hardware.

The future of the Video SOA is uncertain, but positive. In order to be successful, the Video SOA will need to be accepted and adopted by industry. For this to occur, the Video SOA standard must become more visible to industry and be able to highlight its associated cost savings. The end users (military and range personnel) of Video SOA compliant systems are satisfied with the benefits that the system is able to provide, and reports are already spreading about the efficiencies of the system. Their overall hope is a reduction in life-cycle management costs and interoperability with other product line solutions. With that, developers of Video SOA solutions forecast an increase in the number of fieldings of these systems to other Army installations and ranges that provide AAR capabilities. The Government has even started to see interest in the Video SOA standard from other military branches.

When working with a SOA, one should always remember the SOA mantra, “Write one time, use many times,” as well as the fact that a SOA requires an investment in order to achieve returns. If implemented properly, the returns will greatly outweigh the initial costs, and the additional SOA implementation costs should not be much greater than traditional system development. One should also keep in mind that a SOA is not an overnight solution, but if the time is devoted to develop a properly structured SOA, its features will allow for a well-distributed development and support system. This long term outlook of a SOA makes it an ideal solution for government and military usage, as they are able to invest the time and effort needed to mature a system over time.

For more information on the Video SOA standard or to obtain the latest release of the ICD visit the Video SOA section of the LT2 Portal at www.LT2Portal.com.

ACKNOWLEDGEMENTS

We would like to thank Mr. James Todd for his direction, guidance, and assistance in writing this paper as well as his valuable insight and recommendations.

We are grateful to Mr. Todd Kohler, Mr. Derrick Sines, and Mr. James Todd for providing details of their Video SOA experience within the Live Training domain.

We would also wish to thank Dr. Carolyn Kinsell for help in focusing the message of our paper by providing effective commentary throughout the draft stages.

Special thanks should also be given to the entire APM TRADE team for their support in all the milestones and events leading up to the submission of this paper.

REFERENCES

Anderson, R. W., & Ciruli, D. (2006) *Scaling SOA with Distributed Computing*. Retrieved from http://www.ddj.com/web-development/193104809

Bosworth, M. (2008) *Service-Oriented Architecture*. Retrieved from http://www.nextgov.com/the_basics/tb_20080222_8587.php

Erl, T. (n.d.) *SOA Principles*. Retrieved from http://www.soaprinciples.com/default.asp

Erl, T. (n.d.) *What is SOA Principles?*. Retrieved from http://www.whatisSOA.com/default.asp

He, H. (2003) *What is Service-Oriented Architecture*. http://www.xml.com/pub/a/ws/2003/09/30/soa.html

McGovern, J., Tyagi, S., Stevens, M., & Mathew, S. (2003) *Java Web Services Architecture*. San Francisco, FL: Morgan Kaufmann Publishers.

U.S. Army. (2009) *Video SOA Interface Control Document and Service Contract*. Orlando, FL: U.S. Army PEO STRI