Service Development and Orchestration for Common Data Mediation, Translation, and After Action Review

Robert Wittman, PhD
MITRE Corporation
McLean, VA
rwittman@mitre.org

Paul Dumanoir
US Army PEO STRI – PdM WTI
Orlando, FL
Paul.h.dumanoir.civ@mail.mil

Sean Barie, Chris Holmes
MITRE Corporation
Orlando, FL
sbarie@mitre.org, caholmes@mitre.org

Kelly Palmer, Gary Seamans, Andrew Younge
MITRE
Tampa, FL, Ft Huachuca AZ, Bloomington, IA
kpowell@mitre.org, seamans@mitre.org, gjyounge@mitre.org

ABSTRACT

This paper presents an innovative modular service-based Enterprise After Action Review (EAAR) architecture. Through prototyping the architecture shows promise in addressing the high maintenance costs, complexity, and functionality gaps associated with legacy AAR systems and simulation to Command and Control (C2) gateways that have proliferated across the training, test, and operational communities. The resulting EAAR prototype supports a broad range of requirements across the test, training, and operational communities to include data collection, presentation, and visual comparison of raw simulation data; simulated data traversing real-world C2 systems in their native formats; and real-world C2 information and data originating within the C2 environment. This paper describes the evolution of the EAAR prototype, reuse of C2 data mediation and translation design patterns, and how the resulting EAAR prototype not only delivers data capture and presentation services, but how low-level EAAR services can also be deployed independently to support data mediation and translation between simulation and C2 systems. The paper will provide the reader with an overview of the functional and non-functional requirements addressed; it will then describe the guiding development principles and how these shaped the composable design that supports both the EAAR and C2 to Simulation interoperability capabilities; and finally how these capabilities can be extended into the future.

ABOUT THE AUTHORS

Robert L. Wittman Jr. currently works for the MITRE Corporation as the Simulation Engineering Department Head. He has been part of the U.S. DoD M&S community since 1990 and with MITRE since 1992 starting out as a graphical user interface programmer and system analyst for the TACWAR aggregate analytical simulation system. He has worked as systems analyst, chief software engineer, and software architect on Joint and Army training and analytical simulations systems including: TACWAR, the Joint Theater Level Simulation System (JTLS), the Joint Simulation System (JSIMS), OneSAF, and Joint Urban Test Capability. He holds a B.S. in Computer Science from Washington State University, a M.S. in Software Engineering from the University of West Florida, and a Ph.D. in Industrial Engineering from the University of Central Florida

Paul Dumanoir is the Chief Engineer for the United States Army Product Manager for Warrior Training Integration (PdM WTI) under the Project Manager for Integrated Training Environment (PM ITE) at PEO STRI. He is currently the PdM WTI Modernization Risk Reduction project manager and has 28+ years of experience working in Army and 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.

Sean Barie is a Senior Software Engineer for The MITRE Corporation in Orlando, FL. He has 10 years of software development experience in Modeling and Simulation working with U.S. Department of Defense (DoD) programs. He is currently the project leader supporting the U.S. Army Program Executive Office for Simulation, Training and Instrumentation (PEO STRI) for the Integrated Training Environment (ITE).

Chris Holmes is a Senior Modeling and Simulation Engineer for the MITRE Corporation in Orlando FL. He holds a B.S. in Electrical Engineering from George Mason University. He is currently the software lead supporting the U.S. Army Program Executive Office for Simulation, Training and Instrumentation (PEO STRI) for the Enterprise After Action Review (EAAR) and the Converged Enterprise Mediation Services (CEMS) efforts.

Service Development and Orchestration for Common Data Mediation, Translation, and After Action Review

Robert Wittman, PhD
MITRE Corporation
McLean, VA
rwittman@mitre.org

Paul Dumanoir
US Army PEO STRI – PdM WTI
Orlando, FL
Paul.h.dumanoir.civ@mail.mil

Sean Barie, Chris Holmes
MITRE Corporation
Orlando, FL
sbarie@mitre.org, caholmes@mitre.org

Kelly Palmer, Gary Seamans, Andrew Younge
MITRE
Tampa, FL, Ft Huachuca AZ, Bloomington, IA
kpowell@mitre.org, seamans@mitre.org, gjyounge@mitre.org

INTRODUCTION

This paper describes a Program Executive Office (PEO) Simulation Training and Instrumentation (STRI), Project Manager Integrated Training Environment (PM ITE) project to develop, evaluate, and offer composable and reusable next generation Enterprise After Action Review (EAAR) and data/message translation capabilities. While the requirements for these capabilities are not new, the service-oriented approach and reuse of open-source enterprise service bus infrastructure is novel and worth further exploration. Particularly as it supports a common approach for integration within Mission Command (MC) and simulation environments. The main objective of this multi-year effort was to create capabilities that are easily extensible and integrable into a number of existing and emerging Programs of Record (PORs) within the PEO STRI portfolio.

Software Architectural Pattern Trends

Current trends in software engineering demonstrate the advantages of defining clear programming interfaces to access needed capabilities and separating these interfaces from the functional implementation. The ability to support multiple end-user implementations and multi-business functions with such implementations has promoted this software design to a top level software system architectural concept. When combined with specific software services and governance patterns the style of architecture is known as Service-Oriented Architecture (SOA) (The Open Group, 2013b).

As part of this effort we are exploring the use of SOA architectural and design concept to support our training simulation-specific domain objectives to access the following SOA benefits (The Open Group, 2013b):

- *Reducing Cost:* By providing design pattern concepts to cull redundant application functionality and transition to new more efficient implementations and technologies while leveraging existing investments.
- *Agility:* Ability to structure domain solutions based on a set of domain-oriented services in such a way as to enhance the rapid restructuring and reconfiguration of the processes and solutions that consume them.
- *Increasing Competitive Advantage:* Provide the opportunity to use and configure domain services in new and innovative ways without impacting current implementations.
- *Time-to-Market:* Deliver domain-aligned solutions faster by providing composable easily configured services.
- *Consolidation:* Allows for the easier identification of services and end-use applications across historically silo'ed solutions and organizations.
- *Alignment:* SOA enables organizations to better align Information Technology (IT) to domain specific goals, enabling the organizations to understand and prioritize software-specific services.

BACKGROUND

The motivation for this project was to reduce the quantity of similar AAR applications being developed and maintained for different training and simulation systems within the PM ITE portfolio. A PM ITE risk reduction project, called the EAAR project. was launched to address this objective. Initial project analysis results showed that there was approximately 95% AAR commonality across the following four fielded STRI training systems under the PM ITE portfolio (Knox 2013):

1. The Joint Live Component Commander Training Capability (JLCCTC) – Multi-Resolution Federation-Warfighter's Simulation (WARSIM) (JLCCTC MRF-W);
2. The JLCCTC Entity Resolution Federation (JLCCTC ERF);
3. The Live Virtual Constructive – Integrated Architecture (LVC-IA); and
4. The One Semi-Automated Forces (OneSAF) simulation.

Based on the results of that analysis, PM ITE decided to develop an EAAR prototype in 2014 which addressed three primary objectives:

- First to identify risks, challenges, and mitigation techniques in using open architecture and web-enabling technologies to create a common AAR product line for PM ITE;
- Secondly, to compare, evaluate, and identify shortfalls associated with these technologies to those specified within the Army Common Operating Environment (COE) standards and guidance; and
- To assess the AAR product line capabilities against the Network enabled Mission Command (NeMC) requirements.

During a subsequent phase of this EAAR prototype effort the focus was on adapting and extending earlier work to better align with service-orientation. Service-orientation was seen as a way to design for more software component reuse between AAR and simulation to military message translation capabilities. The remainder of the paper focuses on the driving principles behind this service-oriented approach, the resulting implementation and reusable services, why they matter, and the success and challenges encountered.

In 2015 the underlying EAAR architecture was used as a starting point to continue to validate the extensibility and flexibility of the EAAR architecture. Not only in terms of expanded EAAR functions but also to support translation services as part of the Converged Enterprise Mediation Service (CEMS) project. The CEMS project is another PM ITE risk reduction project whose objective is to converge PEO STRI mission command simulation/stimulation mediation services with PEO C3T mission command (Command Post Computing Environment (CP CE)) mediation services. The overall EAAR architecture, as shown in Figure 1, is defined by the following 3 component groups and their interfaces:

1) User Facing Client Applications & Widgets component group;
2) Infrastructure Services component group (now also provides CEMS support); and
3) Data Providers component group.

Figure 1 also highlights COE alignment and standards based interfaces developed as part of the effort.

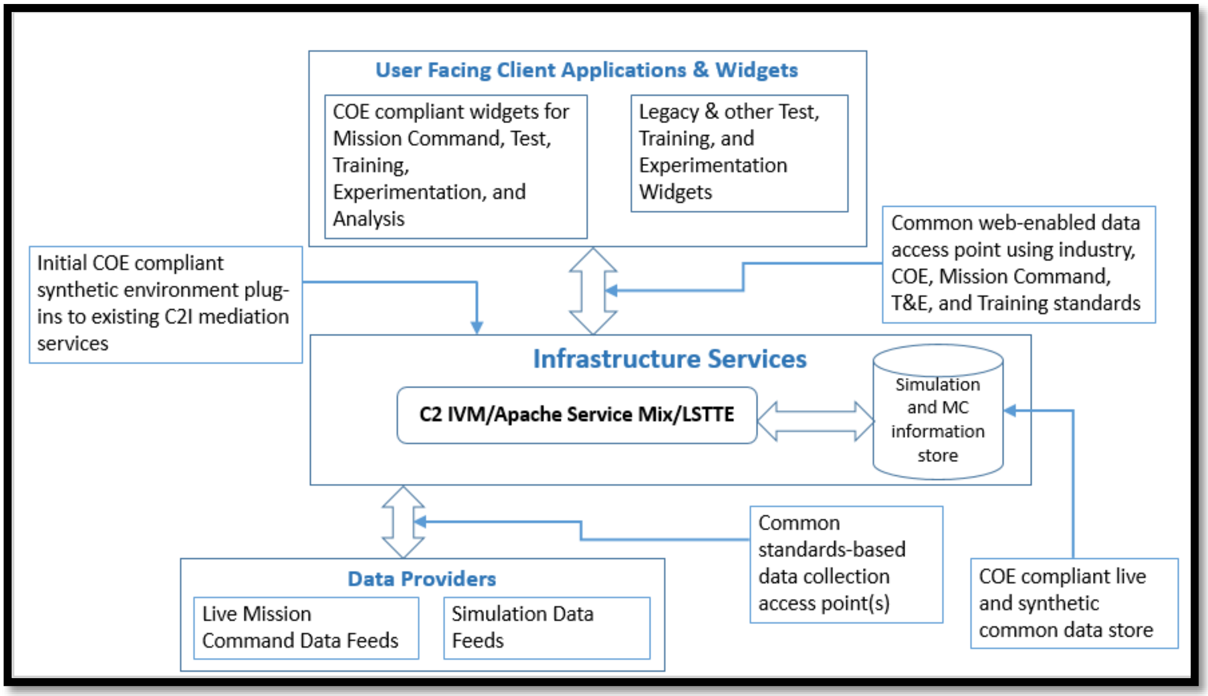
User Facing Client Applications & Widgets
COE compliant widgets for Mission Command, Test, Training, Experimentation, and Analysis
Legacy & other Test, Training, and Experimentation Widgets
Common web-enabled data access point using industry, COE, Mission Command, T&E, and Training standards
Initial COE compliant synthetic environment plug-ins to existing C2I mediation services
Infrastructure Services
C2 IVM/Apache Service Mix/LSTTE
Simulation and MC information store
Common standards-based data collection access point(s)
COE compliant live and synthetic common data store
Data Providers
Live Mission Command Data Feeds
Simulation Data Feeds

Figure 1. EAAR Overall Design

We will use this overall architecture to step into finer grained service component definitions and show where individual services used within the EAAR capability are also used in support of the CEMS project. Additionally, we will highlight how each of these end capabilities: EAAR and CEMS can be employed in standalone, combined, or integrated within a predefined architectural framework such as the Live Synthetic Training Test and Evaluation Infrastructure Architecture (LS TTE IA) (Dumanoir et al., 2015).

The following sub-sections provide basic service-orientation, mediation and translation capabilities, and AAR definitions for future reference.

Services and Orchestration

Service-orientation as defined by the Open Group at https://www.opengroup.org/soa/source-book/soa/soa.htm:

- "Is a way of thinking of in terms of services and service-based development specifically focused on the outcomes of services. Where a service
  - is a logical representation of a repeatable business activity that has a specified outcome;
  - is self-contained;
  - may be composed of other services;
  - is a "black box" to consumers of the services."

Whereas "orchestration" is a way to control access to and sequence services to support a larger capability.

Mediation and Translation

In this context we define mediation and translation as follows:

- Mediation is a general term to describe the preparation and handling of data that may include loss-less translation or more intrusive translation where data may be augmented, culled, or merged with other data sets;
- Translation is the process of changing the form of the data from an input form to an output form. The translation may be loss-less or loss-full.

Some of the training-simulation specific uses cases requiring mediation and translation service support are as follows:

1. Translating between two different real-world message formats such as the Joint Variable Message Format (JVMF) and Over the Horizon – GOLD (OTH-GOLD) formats, so the systems that process these message types can share in the information exchange;
2. Translating between simulation specific formats such as the Distributed Interactive Simulation (DIS) protocol and the High Level Application (HLA) protocol so that simulations supporting these standard interoperability protocols can participate in meaningful ways in a federated exercise.
3. Translating between simulation standard interoperability protocols and data models such as the Military Scenario Definition Language (MSDL) and the standards-based real-world Command and Control (C2) message formats and data models so that initialization and simulation runtime data can be exchanged between simulations, simulation federations, and real-world C2 systems.

After Action Review

Within the Training Community, as discussed above, AAR requirements analysis reported three main topic areas: 2D/3D data replay and display; dynamic data report generation; and take-home package creation.

At the core of the training AAR requirements is the ability to connect training audience decisions and actions to specific mission focused outcomes. To do this it is necessary to collect and display both simulation data and information disseminate within the Mission Command network. This should be done with graphical tools that are easy to use and provide quick and accurate feedback with regard to mission performance. The quality related requirements focus on the ability to quickly customize and extend the data collected as well as the data visualization, graphical display, and report generation capabilities across the collected data set.

As mentioned above a critical piece of the AAR system is the ability to collect and provide feedback on information that was created within the simulation and then disseminated and used by Commanders and their staffs using real-world Mission Command systems during a training event. Historically, this has been a difficult and expensive proposition due to the lack of a common architecture between simulations and Mission Command systems (Kapadia et al., 2013). In working toward a solution to this problem this project coordinated with engineers within PM Mission Command and adopted architecture design patterns and existing components from the Command Post Computing Environment (CP CE). The CP CE, as part of the larger COE, in addition to providing warfighting functional applications also provides common and simplifying infrastructure components. One of which is C2 Infrastructure Virtual Machine (C2 IVM). This is a centralized data bus and mediation services used to provide message based translations and bridging between military messaging versions and types (Wittman et al., 2015).

PRINCIPLES GUIDING DEVELOPMENT

To maintain a focus on developing composable and reusable services our entire concept, design, and development philosophy hinged on a set of guiding principles. These principles align nicely with the targeted SOA benefits described in the introduction.

The following is a listing of the guiding principles identified by the project team and managed by the Chief Architect throughout the development process:

1) **Exploit Existing Open Infrastructure Service Components**: To the maximum extent use open and available service supporting infrastructure software to accelerate development and increase functionality. For this effort the Apache ServiceMix provides a core set of infrastructure functions to include: queuing, messaging, etc.
2) **Standards Usage and Compliance**: To the maximum extent possible identify, use, and adhere to existing military, commercial, and simulation standards for developing interoperable and embedded simulation and Mission Command capabilities. Standards from both the C2 and simulation communities are being used to enable reusable software components.
3) **Pattern Based Design for Reuse and Extension**: To the maximum extent possible providing easily identifiable and documented design patterns are critical to the objective of the effort. As such, extensible frameworks are being used when available rather that home-grown customized code. A number of these patterns are described in the section that follow.

4) **Consolidate Redundant Functionality:** To the maximum extent possible develop service component implementations to be used commonly across the EAAR and CEMS capability sets.

The following sections provide additional information and design concepts that are being developed and refined in direct support of the guiding principles.

Exploit Existing Open Infrastructure Service Components

Apache Service Mix is an open-source integration container that unifies the features and functionality of Apache ActiveMQ, Camel, CXF, and Karaf into an Open Services Gateway initiative (OSGi) compliant runtime platform that can be extended for specialized integration and mediation-focused solutions. The C2 mediation service virtual machine relies on and extends the Apache Service Mix as part of its integration platform.

The Mongo Database was selected as it was a pre-existing component within the Mission Command Adapter Visualization Tool (MCAVT) capability and it provided an opportunity to assess the advantages and potential disadvantages of NoSQL technologies (MongoDB 2015). During development a number of NoSQL advantages and challenges as compared to traditional SQL databases were discovered and are listed below:

1) The team, experienced SQL developers, experienced only a short learning curve through the install, initial schema definition, and storage and retrieval Application Programmer's Interface (API) use of the NoSQL MongoDB;
2) MongoDB allows for quick document-based dynamic schema definitions (optimal for rapid prototyping and evolving data capture requirements) (MongoDB 2015);
3) MongoDB provides a convenient structured text-based programming model for storage and retrieval using Javascript Object Notation (JSON) and Binary Javascript Object Notation (BSON) (MongoDB, 2015);
4) MongoDB is deployed with convenient Apache ServiceMix libraries for plugging into Open Services Gateway initiative (OSGi) related data and other routing integration platform services (MongoDB, 2015).

It should be noted that the team was able to perform some limited performance benchmarking as provided in Appendix A, but future work needs to be done to identify the performance and scalability boundaries of the Mongo and other NoSQL technology.

Standards Usage and Compliance

In addition to using Army and Military message based standards such as the Variable Message Format (VMF) and the United States Message Transmittal Format (USMTF) this effort is also leveraging software development and interface standards as described in the following document set:

- Hyper-Text Markup Language (HTML) 5.0 Technologies;
- Open Services Gateway Initiative (OSGI);
- eXtensible Markup Language (XML);
- The Distributed Interoperability Standard (DIS) family of standards (Institute of Electrical and Electronics Engineers (IEEE) 1278); and
- The High Level Architecture (HLA);

Pattern Based Design for Reuse and Extension

It is expected that both the EAAR and CEMS capabilities will be expanded in the future. The focus on provided pattern based designs is to allow other 3rd party developers to extend the services without negatively impacting maintainability and existing functionality. Tables 1-4 below highlight the patterns provided in each service area for these extensions.

Consolidate Redundant Functionality

An important part of the service-orientation is to identify and remove redundant services and provide an ability to reuse a single necessary service across multiple end capabilities. The helps in a number of ways. It reduces the overall amount of software to be maintained; and it provides the ability to enhance multiple end capabilities with a single enhanced service or a new service that can be deployed without any modification to existing services. Tables 1-4 below provide a description of services that are used across both EAAR and CEMS and in prior implementations would have been implemented multiple times.

CLIENT SIDE APPLICATIONS, WIDGETS, AND BACK OFFICE SERVICES

The EAAR and Translation capabilities and their supporting services are intended to be used in a variety of settings to include standalone, as a combined capability set, or integrated within a larger fully defined architecture such as the LS TTE IA.

Table 1. Client Side and Application and Widget Functionality

Function Name	Description	Extension Pattern	Capability Use
2D Map Based Replay	Provides a map-based 2D replay capability of real-world and simulation data captured during event.	Reusing standard mission command map based library via provided API	EAAR
3D Map Based Replay	Provides a map-based 3D replay capability of real-world and simulation data captured during event.	Reusing standard mission command map based library via provided API	EAAR
Table Production	Provides a user customizable tabular output of message based simulation and real-world C2 data captured during event.	JSON object to table creation framework	EAAR
Report Production	At the end of 2016 a user customizable report output of message based simulation and real-world C2 data captured during event.	Java report generation frameworks to be investigated in FY16.	EAAR
Miscellaneous GUI capabilities such as data filtering, info hover, etc.	Provides an assortment of user-based GUI convenience features.	Java utility frameworks for providing an assortment of common GUI features.	EAAR

Table 2. Back Office Recording Services

Logical Service Name	Description	Extension Pattern	Capability Use
Data Capture	Provides a common plug-in framework to allow web-based data capture to both simulation and real-world military data streams.	OSGI Plug-in Framework	EAAR, CEMS
Data Tagging	Provides a common mechanism to tag the data source prior to harmonizing data within a common data model allows for filter data analysis and data segregation.	OSGI routing service pattern	EAAR, CEMS
Data Harmonization	Provides a mechanism to reduce n*n translation to 2n -1 translations	OSGI routing service pattern and Common Data Model (CDM) pattern	EAAR, CEMS

Table 3. Back Office Playback Services

Logical Service Name	**Description**	**Extension Pattern**	**Capability Use**
Data access	Provides a common plug-in framework to allow web-based data requests for client side requests.	OSGI Plug-in Framework	EAAR, CEMS
Data streaming	Provides a common stream-based data access controls for client side requests.	OSGI routing service pattern and data streaming interface	EAAR, CEMS
Data reporting	Provides a common report and table-based data access controls for client side requests.	OSGI routing service pattern and data streaming interface	EAAR, CEMS

Table 4: Back Office Message Translation Services

Logical Service Name	**Description**	**Extension Pattern**	**Capability Use**
Message Publication	Provides a common plug-in framework to allow access to standard message-based dissemination protocols.	OSGI Plug-in Framework	CEMS
Format Conversion	Provides output translation support from CDM to C2 Message formats.	OSGI routing service pattern and data harmonization patter, CDM pattern	CEMS

LESSONS LEARNED AND TECHNOLOGY TRANSFER

A number of significant lessons have promulgated from the EAAR and CEMS efforts. These lessons can be categorized along the following dimensions:

- Standards-based infrastructure services reuse enables accelerated development.
- A service-oriented approach provides an excellent basis to help focus and identify common services that can be used across multiple capabilities;
- By using a service oriented-approach with well-defined interfaces and "black-box" services it is straight forward to run EAAR and CEMS as standalone applications or to integrate within other well-defined architecture patterns. Figure 2 shows the EAAR and Mission Command Services integrated with the LS TTE IA.

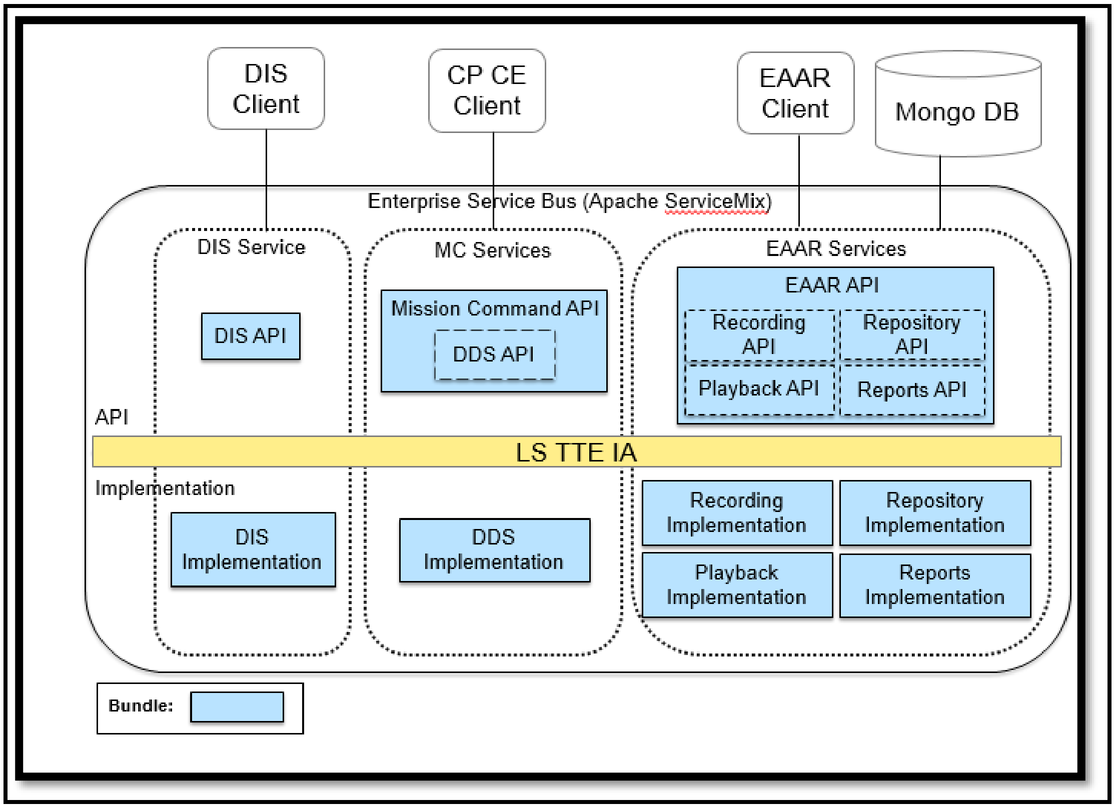

Figure 2. EAAR Services integrated within the LSTTE Architecture

CONCLUSIONS AND PATH AHEAD

The EAAR and CEMS efforts are providing critical technical, developmental, and procedural insights into the advantages of a SOA-based approach. Additionally, and as equally important these efforts are providing early and valuable design feedback in to the evolving LS TTE IA. The team has made significant progress in coordinating and prototyping these tools such that they can be used standalone or integrated as part of LS TTE IA. The EAAR effort accomplished these goals by continuing the use and extension of the:

- PEO STRI interoperability effort design patterns and implementations;
- Simulation (DIS, HLA, MSDL, & the Coalition Battle Management Language (C-BML)), MC (COE, 3D Map API, Ozone Widget Framework), and industry standards (HTML5, Apache Service Mix, etc.); and
- Strategic Mission Command (SMC) provided software (the C2 Infrastructure).

The effort continues to show how the simulation community can leverage industry best practices and open-source software infrastructure to evolve toward more efficient, more cost effective, and more capable training system in support of the warfighter.

Additionally**,** PEO STRI views this effort as an important milestone for continued collaboration with PEO C3T/PM MC. The products and lessons from this effort can now be used in developing future versions of the COE CP CE infrastructure, data and user-facing applications to support common training, testing and operational systems. By necessity COE infrastructure components are planned and funded to provide support for legacy and new MC messages type, fields, and system components. As such PEO STRI can access these components via well defined APIs (Wittman et al, 2015). This relieves PEO STRI programs from having to maintain their own complex and costly translation applications allowing them, instead, to focus on training system development.

Results and prototype products from these EAAR and CEMS efforts are being used as inputs for the establishment of a PM ITE set of core assets. Through the appropriate governance process, these core assets will then be used to provide efficient enterprise level AAR and MC interoperability capabilities across several PEO STRI training and testing PORs.

ACKNOWLEDGEMENTS

The authors greatly appreciate and acknowledge sponsoring support from PEO STRI, the Army Research Laboratory (ARL) Human Research and Engineering Directorate (HRED) Simulation Training and Technology Center (STTC), the Army Modeling and Simulation Office (AMSO), PEO C3T/PM MC, the Army Operational Test Command (OTC), and the Simulation to Mission Command Interoperability (SIMCI) organization. The EAAR effort, at is core, intends to inform future training, operational, and test simulation-based environments in support of the warfighter's training, operational, and test environment and associated tools.

REFERENCES

Dumanoir, P., Barie, S., Crutchfield, R., Grippin, B., Willoughby, M., Wittman, R. (2015). Live Synthetic Training and Test & Evaluation Infrastructure Architecture (LS TTE IA) Prototype. For publication I/ITSEC 2015.

Kapadia, A., Wittman, R., Beebe, W., Holmes, C., Osborne, R., Belfor, N., Webster, S., Berglie, S., Hanley Y., Tran, O., Smith, M., May, C. (2013). Providing OneSAF-based Prototype Simulation Services and Standards into the U.S. Army Command Post Computing Environment, 13F-SIW-010, Fall SIW, Retrieved from www.sisostds.org.

Knox, R. (2013). PM ITE RRTB Final Findings Report for Research Topic 5 – AAR Common Component Research Version 1.2. Unpublished.

MongoDB. (2015). Overview and MongoDB available at https://www.mongodb.org

PEO C3T. (2015). CP CE [Text Description]. Retrieved from http://peoc3t.army.mil/mc/cpce.php

The Open Group Service Oriented Architecture: What is SOA? (2013a) from https://www.opengroup.org/soa/source-book/soa/soa.htm

The Open Group Key Business Benefits of SOA (2013b) from https://www.opengroup.org/soa/source-book/soa/soa.htm

Wittman, R., Kapadia, A., Grippin, B., Barie, S., Holmes, C., Dumanoir, P. (2015). Early Adoption of Common Operating Environment (COE) Standards and Guidelines, Paper 15098 I/ITSEC 2015.