Early Adoption of Common Operating Environment (COE) Standards and Guidelines

Robert Wittman, PhD	**Amit Kapadia**
MITRE Corporation	**US Army PEO STRI – PdM OneSAF**
McLean, VA	**Orlando, FL**
rwittman@mitre.org	**Amit.m.kapadia.civ@mail.mil**
Burt Grippin, Sean Barie, Chris Holmes	**Paul Dumanoir**
MITRE Corporation	**US Army PEO STRI – PdM WTI**
Orlando, FL	**Orlando, FL**
bgrippin@mitre.org, sbarie@mitre.org,	**Paul.h.dumanoir.civ@mail.mil**
caholmes@mitre.org	

ABSTRACT

This paper presents the driving requirements, results, and user-oriented use cases behind the Enterprise After Action Review (EAAR) prototype. The activity, funded in FY14 by Program Manager Integrated Training Environment (PM ITE), focused on producing a reusable AAR architecture with transferrable technologies and lessons learned for early adoption of the Common Operational Environment (COE) across PM ITE and PEO STRI. At project initiation it was clear the successful execution of a prototype depended on collaboration between the PM ITE team and engineers supporting the Command Post Computing Environment (CP CE) within PM Mission Command (MC). This paper starts by introducing earlier efforts that provided critical insight and software that led to the feasibility of the EAAR prototype investigation using COE constructs. It continues by highlighting common AAR requirements across the training and operational MC community. This sheds light on the potential for AAR system lifecycle cost-savings through shared AAR component development and use across the two communities. The paper then explores the specific COE design constructs and technologies employed as part of the EAAR prototype effort. Finally, the paper concludes with a section on technology transfer progress and a listing of the COE and other standards-based technologies employed and their training or operational MC source.

ABOUT THE AUTHORS

Robert Wittman is the Simulation Engineering Department Head for the MITRE Corporation in McLean, VA. 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.

Amit Kapadia is the Product Manager OneSAF Lead MC Systems Engineer at the U.S. Army PEO STRI in Orlando, FL. He holds B.S. and M.S. degrees in Electrical Engineering from the University of Central Florida.

Burt Grippin is a Lead Modeling and Simulation Engineer for the MITRE Corporation in Orlando, FL. He holds a B.S. in Electrical Engineering Technologies from the State University of New York. He is the Project Lead for the Live Synthetic Training and Test & Evaluation (LS TTE) architecture prototype and OneSAF projects.

Sean Barie is a Senior Software Engineer for the MITRE Corporation in Orlando, FL. He holds a B.S. In Computer Science from Plattsburgh State University of New York. He is the Lead Architect for the LS TTE architecture and Enterprise After Action Review prototypes.

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 Lead for the OneSAF Project.

Paul Dumanoir is the Chief Engineer for the US Army Product Manager for Warrior Training Integration (PdM WTI) under the PM for PM ITE at the Program Executive Office for Simulation, Training, and Instrumentation (PEO STRI). He earned his B.S. in Electrical Engineering from the University of South Alabama and his M.S. in Computer Systems from the University of Central Florida.

Early Adoption of Common Operating Environment (COE) Standards and Guidelines

Robert Wittman, PhD

MITRE Corporation

McLean, VA

rwittman@mitre.org

Amit Kapadia

US Army PEO STRI – PdM OneSAF

Orlando, FL

Amit.m.kapadia.civ@mail.mil

Burt Grippin, Sean Barie, Chris Holmes

MITRE Corporation

Orlando, FL

bgrippin@mitre.org, sbarie@mitre.org, caholmes@mitre.org

Paul Dumanoir

US Army PEO STRI – PdM WTI

Orlando, FL

Paul.h.dumanoir.civ@mail.mil

ENTERPRISE AFTER ACTION REVIEW (EAAR): INTRODUCTION AND BACKGROUND

This paper presents prototyping efforts associated with an Enterprise After Action Review (EAAR) development activity that has engaged the training, Mission Command (MC), and test communities. It specifically highlights efforts surrounding the primary goal of the EAAR effort to provide insight into the early adoption of the Army's Common Operating Environment (COE) to facilitate lower cost, more integrated, and more rapidly deployable simulation-based training and testing tools (Albert & Rosemergy, 2011), all of which are consistent with the guiding principles of the COE (ASA(ALT), 2011). The paper starts with an introduction of the Army COE and then provides background and context from the efforts that provided the foundation for the EAAR. It then moves into the work performed and the resulting capabilities and their importance. The paper concludes with a section on technology transfer progress and a listing of the COE and other standards-based technologies employed and their training or operational MC source.

As an introduction, the **Army COE** is defined as an approved set of computing technologies and standards that allow secure interoperable applications to be rapidly developed and deployed across a variety of Computing Environments (CE).(Farah-Stapleton, 2011). The COE is seen to be of critical impact to current and emerging training, operationally focused, and test-focused simulation systems as it provides requirements, standards, and guiding principles for interoperability and software asset reuse across the Cross Cutting Capabilities (Minor, 2015) and the six CEs (ASA(ALT), 2011) that include:
1. The Command Post (CP) CE;
2. The Mobile/Hand-Held CE;
3. The Mounted CE;
4. The Sensor CE;
5. The Data Center/Cloud/Generating Force CE;
6. The Real-Time/Safety-Critical/Embedded CE.

In the past simulation interoperability with many, if not all, MC systems has been challenging due to a lack of a common architecture between simulations and MC systems. This has resulted in hard-to-manage interface complexity, longer development cycles, and elevated cost as simulations struggle to support the rapid change of message formats and technologies within the MC community (Kapadia et al., 2013). This paper discusses the specific COE standards and design patterns within the CP CE and how they are used by the EAAR project to facilitate current and future program adoption to enable lower cost, more integrated, and more rapidly deployable simulation-based training and testing tools.

The CP CE, as part of the COE, intends to simplify the hardware and software infrastructure for Army CPs. A critical step toward the user-facing simplification is the consolidation of several warfighting functions: the fires, logistics, intelligence, airspace management, and maneuver areas into a consistent, accessible workspace (PEO C3T, 2015). The Ozone Widget Framework (OWF) provides a single integrating foundation for the broader software development community to contribute COE/CP CE compliant web-based "apps" for warfighter consideration and use. This along

with the development of open interface specifications, software development tools, and common map-based Application Programmer's Interface (API) provide real opportunities for integration of new and innovative simulation-based MC tools. The resulting core set of simulation tools are envisioned by the authors to be multi-purpose in support of training, mission planning and rehearsal, situational understanding, and other commander decision-making processes in a decentralized collaborative environment (PEO C3T, 2015).

With that brief COE introduction, we can now move to the set of activities that have contributed critical concepts, software infrastructure, and path finding for the development of the EAAR and other MC integrated simulation capabilities. The EAAR prototype was made possible from the software tools and lessons contributed from the efforts below:

- The Army's Office of the Assistant Secretary of the Army for Acquisition Logistics and Technology ASA(ALT) COE effort as described above;
- The Simulation to Mission Command Interoperability Overarching Integrated Product Team (SIMCI OIPT) sponsored Mission Command Adapter Visualization Tool (MCAVT) effort;
- The Program Executive Office and Simulation Training and Instrumentation (PEO STRI) and Simulation Training and Technology Center (STTC) Risk Reduction Test Bed (RRTB) AAR commonality assessment;
- The PEO STRI and STTC RRTB EAAR prototype effort; and
- Initial coordination with the Integrated Live-Virtual-Constructive Test Environment (ILTE) team.

With the exception of the COE each of these efforts will be briefly described in the following paragraphs.

The **Simulation to Mission Command Interoperability Overarching Integrated Product Team (SIMCI OIPT) sponsored Mission Command Adapter Visualization Tool (MCAVT)** was developed between May 2012 and April 2013. This, first of its kind effort, produced a browser-based map-display of message data generated from the OneSAF MC Adapter (MCA). The effort explored a number of web-enabling technologies and services emerging within the MC and simulation communities. These technologies included: a NoSQL Mongo Database for storing the military message data; Apache ServiceMix for its web-enabling infrastructure; OpenStreetMap, World Wind, and Google Earth map services; 2525B graphical icon services; and a host of HTML5 technologies including websockets, javascript, and associated graphics libraries for creating web-enabled user tools.

During, and continuing beyond, the execution of the SIMCI MCAVT project the team work closely with PEO C3T/ PM MC engineers to support potential future integration with MC components. This collaboration evolved into a technology reuse plan as part of a common architecture concept to reuse the components of the MCAVT to meet Network enabled Mission Command (NeMC) AAR and CP CE Version 2.0 "transcribed collaboration" requirements.

The **Risk Reduction Test Bed EAAR Requirements Development effort** was executed under a joint initiative between the U.S. Army Research, Development and Engineering Command (RDECOM) Army Research Laboratory (ARL) Human Research & Engineering Directorate (HRED) Simulation Training & Technology Center (STTC) Advanced Simulation Branch (ASB) and PEO STRI. The joint initiative, called the Risk Reduction Test Bed (RRTB), prioritizes the identification and early application of emerging technologies to STRI Programs of Record (PoR) without impacting their development and fielding timelines. The EAAR Requirements Development effort was executed during March through December 2013 and prioritized assessing requirement and architecture commonality across four fielded STRI training systems to support or refute common AAR capability development. The final report showed significant commonality, approximately 95%, across the AAR requirements of the four PM ITE systems (Knox, 2013):

1. The Joint Live Component Commander Training Capability – Multi-Resolution Federation-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.

The requirements covered three main topic areas: 2D/3D data replay and display; dynamic data report generation; and take-home package creation. In total a set of 424 individual requirements were identified across the projects (Knox, 2013).

The **Risk Reduction Test Bed Enterprise After Action Review (EAAR) Prototype** followed the AAR requirements effort and was executed March through December 2014. The effort produced a prototype addressing 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 Common Operating Environment (COE) standards and guidance; and
- To assess the AAR product line capabilities against the NeMC requirements.

The MCAVT was identified as a reuse product for the EAAR prototype for two primary reasons. First, it had demonstrable AAR capabilities; and secondly it was developed using COE/CP CE technologies, architecture constructs, and standards. This allowed the team to make immediate prototype-based progress toward the three objectives identified above. The earlier RRTB effort requirements were used as the primary source to prioritize user interface enhancements, prioritize the infrastructure service development, and to prioritize data collection enhancement for the EAAR product line.

Under PEO STRI's leadership, stakeholders across the training and test community were invited to EAAR reviews for awareness, comment, and feedback. This outreach resulted in direct and supportive participation by the US Army Operational Test Command (OTC) and the Project Manager for Instrumentation, Targets and Threat Simulators (PM ITTS), which led to the identification of common capabilities, technologies, and tools for use across the training and test communities. This coordination and engagement has the potential for substantial savings through early identification and development of common capabilities.

These efforts and their use of the COE/CP CE architecture, standards, and guidelines provides important lessons for near-term integration efforts and future PoRs. The remainder of the paper looks across the EAAR effort to describe progress, identify challenges, lessons learned, and opportunities to improve COE integration across simulation-focused development projects.

IDENTIFYING COMMON AAR REQUIREMENTS

This section summarizes the process of identifying and the resulting collection of driving requirements for AAR capabilities across the training and operational communities using high-level requirements-oriented source documents. The analysis identified general AAR commonality and highlighted the cross domain reuse advantages of requiring and developing a core AAR capability that adheres to COE/CP CE architecture and standards and that is easily customized without dramatic architectural or code changes.

Within the **Training Community**, as discussed above, the RRTB requirements covered three main topic areas: 2D/3D data replay and display; dynamic data report generation; and take-home package creation. In total a set of 424 individual requirements were identified across the projects.

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 in a meaningful way graphical tools are necessary to provide quick and accurate feedback with regard to mission performance. The non-functional or 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.

Within the operational **MC Community** initial CP CE Version 2.0 AAR requirements focus on a "transcribed collaboration" capability where Transcribe Collaboration is defined as, "… function (that), given all information exchanges of the collaboration; including actions, decisions, and products; transcribes collaboration and posts or publishes to a shared space using application support services." (Cronen et al., 2013)   Not only is this seen as a significant review capability in its own right, but is also envisioned as allowing the sharing of common data between currently disparate systems in the CP.   The specific operational MC requirement areas can be categorized along the following two dimensions (Cronen et al., 2013):

- Data Presentation
  - Controls must be provided to authorized user to control the transcribed collaboration capabilities;
  - The capability must alert the users that they are taking part in a transcribed session; and

o The collected data must be "playable" with both visual and audio data on a standard client system
- Data Collection
  o The system will collect and store available visual, voice, and proprietary audio format that is accessible for presentation as an integrated data set; and
  o The system will provide transcribe collaboration sessions within the common workspace.

Requirement Commonality

By categorizing the requirements within "data collection and storage"; "data access"; and "data presentation and visualization" sets across the 2 domains provides an understandable set of common requirements:

Data Collection and Storage:
- The ability to configure, select, manage, and monitor data to be store during an event;
- The ability to store the selected data without impact to the event during the event;
- The ability to time-stamp and link appropriate meta-data to data being stored;
- The ability to store a wide range of data formats including: text, alphanumeric, binary messages, voice/audio, imagery, etc. that is of interest during an event; and
- The ability to store information about live, virtual, and constructive objects, actors, and events that are of interest during the event;

Data Access:
- The ability to access or filtered part of the stored data;
- The ability to deliver the stored data as part or in whole as a take-home package;
- The ability to remotely access the stored data; and
- The ability to access stored data without impact to the storage of new data;

Data presentation and Visualization:
- The ability to link through analysis and visualization the command and staff decisions with resulting outcomes;
- The ability to view and present in graphical, report, or audio modes the data that was collected;
- The ability to filter and query specific subsets of the data;
- The ability to customize and the data presentation for specific user types;
- The ability to limit data access based on role-based authorizations and authentication;
- The ability to use existing and standards-based hardware, software, and graphical presentation techniques applications, and assets to present the data visually and/or in an auditory mode; and
- The ability to present captured data while new data is being captured.

By identifying and highlighting this commonality there is an increased possibility of reducing
AAR software lifecycle costs via developing shared core software assets and/or interfaces that can be reused across the domains.

COE ALIGNED EAAR DEVELOPMENT

As introduced in the earlier sections the EAAR development is using standards, guidelines, best practices, and actual software assets from a number of communities. The reuse-focused software engineering approach relies heavily on open collaboration across domain members. This collaboration is critical as each of these communities: training; testing; and operational MC each has their own system level architectures, as well as their own product development, test, and delivery timelines. By emphasizing collaboration, awareness, and strategic alignment with the COE/CP CE the EAAR team has developed an initial prototype that provides a technical foundation and associated insights for an AAR toolset with a potential to support and share software across multiple domains. The following sections describe the technical foundation for the AAR and the relationship to the COE.

Technology Selection and Prototyping

The EAAR prototype provides an example of web-enabled data collection, data processing, data server, and client-side presentation capabilities supporting the training, test, and MC user communities. The development of this prototype provides new technical insights into the early adoption and use of the Army's COE/CP CE Architecture.

The section provides a description of the overall prototype; highlights the critical technologies and alignment with the COE; identifies early and on-going transition and integration of EAAR developed capabilities with the PEO STRI PoR portfolio; and identifies next steps to reduce technology risk to continue and accelerate EAAR with PoR integrations, not only at PEO STRI but also within PEO C3T.

The prototype EAAR system is separated into the 3 components as shown in Figure 1: the user interface (User Facing Client Apps & Widgets), the Infrastructure Services, and the simulation (PEO STRI Simulation Product Line) and MC (Live MC Data Feeds) data providers. Along with these 3 main components Figure 1 highlights COE interface alignment, reuse, and web-enabled standards-based interfaces developed as part of the effort. The following 2 subsections describe the user interface and infrastructure components and how they leverage existing COE standards to access data from the MC and simulation data providers.
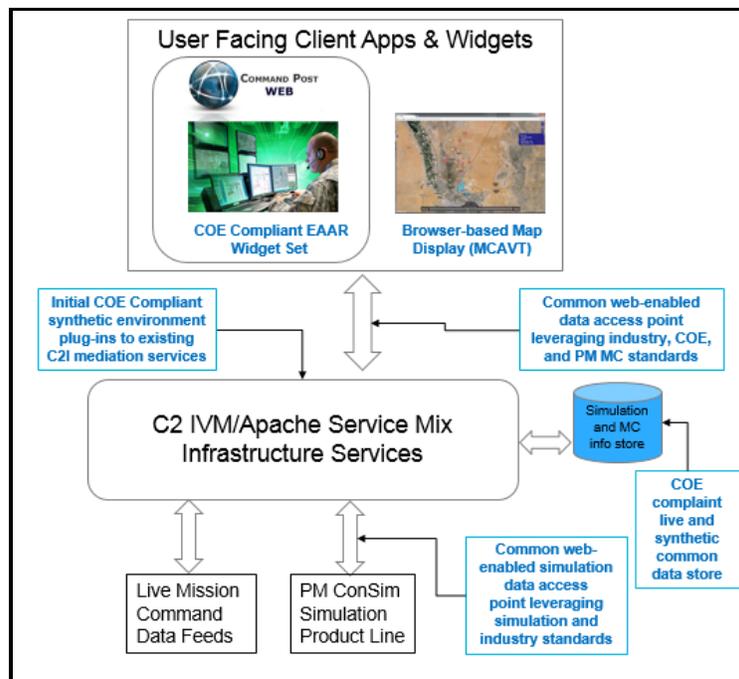

Figure 1 EAAR Overall Design

EAAR Prototype User Interface

By employing a service-based software design, consistent with the COE principles and technical reference architecture, the EAAR supports easier evolution and/or replacement of existing components than if new components or 3rd party technologies were introduced into a tightly coupled database/presentation AAR systems. The EAAR does this by separating and maintaining independence and a well-defined interface between the user facing application/widget set and the underlying database and data model. Additionally, a well-defined Service Oriented Architecture (SOA) and industry-based HTML5 data exchange protocol provide the communications mechanism for client-side data requests and appropriate data payloads in response. For this prototype effort simple 2D/3D display, playback, and report protocols and associated interfaces were defined. This interface contract allows for easy extension and allows both the user interface and database technologies to evolve independently. Thus allowing for the easy introduction of new client-side 3rd party user interfaces or database technologies so long as the interface contract is respected.

An important feature of the approach to easily integrate different user interfaces into the architecture was demonstrated as the effort progressed. Two separate experimental AAR user interfaces were integrated within the COE/CP CE architecture and delivered as part of this prototype. The first provided a "first of its kind" browser-based AAR user interface accessible within the CP CE OWF. The second provided the first user interface integrated within the Common Map Widget using the Common Map API and integrated with the COE/CP CE architecture. Both user interfaces re-used significant software assets from commercial and other DoD projects. The browser-based tools now supports the following EAAR capabilities:

- A map display showing military-unit and individual platform locations using the COE 2525B symbols;
- Tabular report generation of military-unit and individual platform data;
- Filter sets to filter and display on a variety of COE compliant message type and data sources:
    - message type: VMF, USMTF, OBSREP, POSREP, etc.;
    - simulation or real-world data sources;

- Time based replay and control via slider-bar;
- Mouse hover-focused data display and drill-down;
- Integrated and easily identifiable real-world versus simulation data;
- Friendly and threat identification notations;
- Newly defined web-based client/server data replay request and report protocol;
- COE complaint and standards-based web map client display; and
- Client integration with the Command Web and COE/CP CE compliant OWF.

As noted above the enhancement to the original MCAVT consisted of performing an initial integration of the tool into the Command Web OWF. The intent and result was to provide insight into the ease with which external browser-based tools could be accessed within the OWF. This was accomplished with minimal effort, essentially just providing a file based URL to the web application so that it could be accessed and started within the OWF.

The second new user interface, shown in Figure 2, was also developed as part the prototype. Like the MCAVT interface, the second interface also integrates within the COE/CP CE OWF and uses the same database access protocols to get to the stored data. Additionally it also reuses the COE Common Map Widget and the Common Map API for the map-based display. Figure 2 shows the "zoomed out" AAR data displayed on the COE Common Map Widget map-background within the OWF.
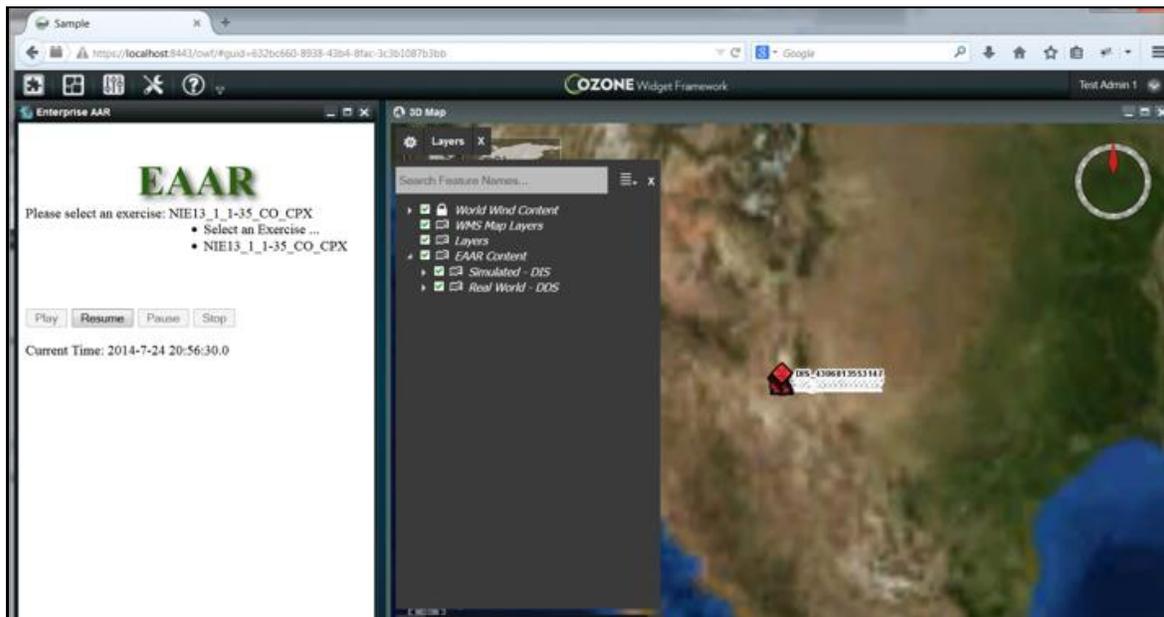

Figure 2: EAAR COE Compliant GUI

EAAR Prototype Infrastructure

Along with the user interface component the infrastructure component prioritized COE and industry standard software reuse. By reusing existing MC, simulation, and open-source components for real-world data access, simulation data access, and database and web-enabling technologies the EAAR product promotes lower cost, more rapid capability development and deployment in support of PEO STRI and PEO C3T requirements. The following reuse components provide the extensible base of the EAAR infrastructure design and development effort:

Command and Control Infrastructure Virtual Machine (C2 IVM)/Apache ServiceMix: The C2 IVM was selected as a reuse access point component for collecting MC AAR related situational, planning, and execution-focused data. It was selected for a number of compelling reasons:

- First, as part of the CP CE it provides existing software connections to real-world data sources (Data Dissemination Services (DDS), Variable Message Format messages (VMF), United States Message Text Format (USMTF), etc.);
- Secondly, its provides data translation and mediation services between these message types and has been extended to support simulation standard formats such as the Military Scenario Definition Language (MSDL) (SISO), 2008) and the Coalition-Battle Management Language (C-BML) (Kapadia et al., 2013) ; and
- Finally, because of its inclusion within the COE V2 application set.

Another important aspect of the C2 IVM is that is implemented using the Apache ServiceMix platform. This provides an open architecture and industry standards-based infrastructure that can be easily extended to support AAR infrastructure capabilities. This effort has demonstrated within the prototype that EAAR capabilities can be executed within the MC C2 IVM or within an off-the-shelf Apache ServiceMix implementation. This allows for EAAR use even when then MC capabilities are not available.

Mongo NoSQL Database: The MongoDB Database was selected, as a reuse component, as it was a pre-existing component within the MCAVT. It was advantageous, as it also provided an opportunity to assess the advantages and potential disadvantages of NoSQL technologies with low-entry cost. During development a number of NoSQL advantages and challenges, as compared to traditional SQL databases, were discovered and are listed below:

1) The team, seasoned SQL developers, experienced only a short learning curve through the install, initial schema definition, and storage and retrieval 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);
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);
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.

It should be noted that while the MongoDB is not part of the existing COE environment, the ServiceMix environment provided convenient mechanisms to interface to MongoDB such that it could be included as an external component with minimal impact to the existing COE components. Figure 3 shows the initial service mix design pattern that was used as guidance for the integration of the C2 IVM and MongoDB database (Future Skies Inc., 2011). The SMART Node (left) in the original graphic is replaced with C2 IVM as shown on the right and "Database" (left) is replaced with the specific MongoDB database (right).
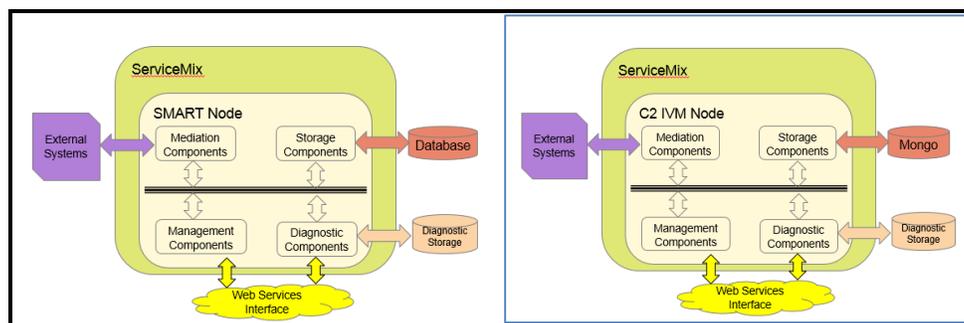

Figure 3: C2 Infrastructure

It is important to note that to support a wide range of AAR use cases the C2 IVM and the Mongo DB combination can be run in a number of configurations to include:
1. Integrated into the same Virtual Machine (VM);
2. Within separate VMs on the same computing environment;
3. Within separate VMs on separate computing environments,
4. Or "old school" on separate computing environments on their native Operating Systems.

The following sections provide a detailed look at the services based upon this high level design construct.

From the basic infrastructure architecture, shown in Figure 3, the EAAR effort modularized and extended earlier SIMCI efforts that had developed simulation control and initialization services within the C2 IVM. Figure 4 shows these services and the interface mechanisms employed to access their capabilities. A description of the various components within the diagram are provided in the paragraphs below.
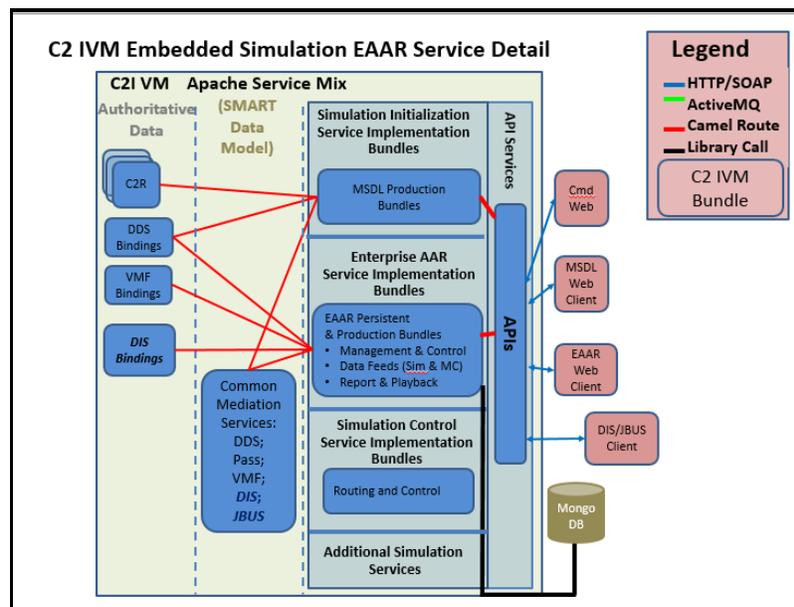

Figure 4: C2 IVM and Embedded Simulation Services

EAAR Prototype External Clients: To the right of the C2 IVM stands the web clients that access the infrastructure services through their defined standardized web services interfaces. The API Services component within the infrastructure provides a convenient location for the storage of these interfaces. The external clients depicted are those that were developed as part of the EAAR activity or as part of the earlier SIMCI efforts and access the available simulation services within the infrastructure. These clients include the 2D/3D map applications within the Command Web and OWF; the MSDL simulation initialization tools; EAAR tools; and simulation systems generating Distributed Interactive Simulation (DIS) protocol activity. The final external component is the MongoDB database that provides the repository for data collected from both real-world and simulated data feeds as discussed earlier.

EAAR Prototype Application Programmer Interface/Web Service Interface Services: Moving into the infrastructure from the right to the left, the first component set is the API Service. The API Server layer holds the interface definitions so that they can be conveniently accessed by external clients. In addition, it provides a central location for the service interface definitions so that automated management mechanism can be employed for registration and discovery, service startup orchestration, service signature configuration management, and other service status monitoring and management capabilities. Although these types of management functions are not being developed by the EAAR activity, the Live Synthetic Training and Test & Evaluation (LS TTE) RRTB activity is actively designing and developing these capabilities based on the overall architecture construct depicted here. The LS TTE IA is a cloud-enabled SOA prototype with an expressed goal of supporting both the Training and the Test & Evaluation communities. LS TTE IA designers are investigating, via active prototyping and coordination with the

EAAR team, the feasibility of COE compliance and suitable for hosting within the COE Data Center/Cloud Computing Environment. (Dumanoir et al., 2015).

Simulation Services: EAAR: Residing below the simulation initialization services are the EAAR services. This set includes data capture, data storage, and client-focused data retrieval services across the simulation and real-world data communities. As part of this prototype effort simulation sources include DIS IEEE standard Version 1278.1 compliant simulations that can produce entity-state Protocol Data Units (PDU). Although the entity-state PDUs are the only ones supporting by the data capture routine at this point, the DIS entity-state PDU design pattern is easily extensible to other PDUs (fire, detonation, emission, etc.) as well as to other simulation interoperability standards including the High Level Architecture (HLA) and C-BML. The real-world data sources currently include a DDS feed that is easily extensible to VMF, Chat and other emerging C2 IVM supported data feeds. A capability multiplier within EAAR service set is the availability and use of the C2 IVM Common Data Model (CDM) allowing the EAAR services to route data via the common data model to support other external clients that may have a use for the common data in their own specialized format.

CDM: The common data model provides the core of C2 IVM data translation and mediation services. It allows data within specialized message format to be mapped once to a common data model and then served out to interested clients. This all happens via common mechanism rather than customized point-to-point interfaces. The CDM and associated mediation services allow the efficient translation between different messages types (DDS, USMTF, VMF, JVMF, etc.) as well as between different versions to support evolving and legacy client MC applications. Both the SIMCI and EAAR efforts capitalize on this CP CE provided capability for its cost reduction, standards-based approach, existing and legacy application support, and sound engineering principles. For the EAAR effort the simulation data feed, specifically the DIS entity-state PDUs, have been mapped to the CDM where upwards of 80% of the PDU data found a consistent data field for its mapping. This means no additional work, other than mapping the data into the common data field to store the data, is necessary to make the data available for interested client access in a variety of formats. This not only provides a useful client-side data access and review capability but it also provides evidence that the C2 IVM can be used as a single data translation and interoperability services for simulation to MC message and data interactions.

Commercial Open-Source Apache ServiceMix Integration: Because of the focus on composability the EAAR system can be easily configured to run in situations where the C2 IVM and its CDM, and Authoritative Data Bindings and data are unavailable. These use cases pertain to Coalition exercises, Foreign Military Sales, or when using the capability on non-military provided internet assets. The overall architecture remains the same as shown in Figure 4, but the CDM and Authoritative Data Bindings components are removed leaving the routes with empty endpoints. In our early laboratory experiments this has proven successful and allows a convenient mechanism to test infrastructure capabilities against evolving Apache ServiceMix versions without having to wait for the official C2 IVM to step up to these newer versions.

LESSONS LEARNED & TECHONOLOGY TRANSFER

The **training community**, specifically PEO STRI, is actively using designs, existing software, and concepts from the EAAR software assets. As an example the LS TTE IA prototype, as mentioned above, is leveraging the design and software contributed by the EAAR, as well as using the EAAR as a service provider. The **MC community,** through coordination with EAAR prototype team, views the EAAR tools as a useable COE tool and a test-bed for future exploration in meeting NeMC AAR requirements. This cooperative development is an important step toward greater design commonality across the C3T and STRI portfolios.

By participating within the RRTB EAAR prototype effort the **test community** has gained insight on the direction, technology, and integration of the EAAR prototype across the training and operational communities. Likewise the EAAR team has been made aware of test communities' next generation AAR requirements. At this time it appears that the evolving core EAAR prototype capabilities can be applied across all three communities (training, operational, and test) and that specific EAAR extensions can be developed in concert with the other EAAR capabilities.

CONCLUSIONS AND PATH AHEAD

The EAAR effort is providing critical technical, developmental, and procedural insights into early adoption, compliance, and use of the Army's COE/CP CE Architecture. The team has made significant progress in coordinating and prototyping a common and extensible AAR-based infrastructure with members of the MC engineering team within PEO C3T and other PEO STRI funded RRTB efforts. The prototype also provides useable and exemplar COE/CP CE compliant capabilities that can be used by existing PoRs to rapidly and effectively migrate toward COE compliance. The EAAR effort accomplished these goals by using and extending existing components to include: PEO C3T and PEO STRI interoperability effort design patterns and implementations; Simulation (DIS, HLA, MSDL, & C-BML), MC (COE, 3D Map API, OWF) (Minor, 2015), and industry standards (HTML5, ASM, etc.) (Kapadia et al., 2013); and Strategic MC provided software (the C2 Infrastructure).

The effort has shown the simulation community can extend the PEO C3T provided common Map widget as this project has embedded a browser map-based GUI within the Command Web framework and has also used the COE 3D map widget and API compliant GUI implementation. This initial prototype demonstrates COE infrastructure reusability, extensibility, and maintainability supporting the objective of delivering more effective capabilities to the warfighter.

Additionally this effort has provided evidence that the evolving core EAAR prototype capabilities can be applied across the training, operational, and test communities and that specific EAAR extensions can be developed in concert with the other EAAR capabilities. If governance constructs are identified to collectively develop and manage common capabilities across the 3 communities there is a real potential to drastically reduce AAR software lifecycle costs. While the EAAR team made significant progress in FY14 showing feasibility of a COE-based EAAR capability, more work is necessary to demonstrate overall EAAR scalability and operational performance.

ACKNOWLEDGEMENTS

REFERENCES

ASA(ALT). (2011). Common Operating Environment Implementation Plan Core v.3.0 Draft, Retrieved from http://usarmy.vo.llnwd.net/e2/c/downloads/232001.pdf

Albert, C., Rosemergy, S. (2011). A Framework for Evaluating Common Operating Environments: Piloting, Lessons Learned, and Opportunities. CMU/SE-2010-SR-025. Retrieved from http://www.sei.cmu.edu.

Cronen, C., Easterling, S., Feraci, L., Graff, L., Greenburg, Gottovi, D., Jain, N., Livingston, S., Mackert, D., McNeilly-Anta, B., Silberman, A., Soeder, B., Tardy, P. (2013). Technical Guidance For Common Operating Environment Command Post Computing Environment Version 2.0 Version 1.2. Unpublished.

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.

Farah-Stapleton, M. (2011). ASA(ALT) COE Implementation [PowerPoint Slides]. Retrieved from http://www.acq.osd.mil/se/webinars/2011-07-12-SoSECIE-Army-COE-Farah-Stapleton-brief.pdf.

Future Skies, Inc. (2011). COBRA SMART Software Design Description – v1.1. Contract# W15P7T-09-C-S328.

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

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.

Minor, P. (2015). Army COE Update [PowerPoint Slides]. Retrieved from http://www.afcea-aberdeen.org/files/presentations/AFCEA_Aberdeen_COE%20Update_11March2015.pdf

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

SISO. (2014). Standard for: Coalition Battle Management Language (C-BML).

SISO. (2015). Standard for: Military Scenario Definition Language (MSDL).