Simulation Dashboard to Manage Multi-Federate Simulation Events

Eberhard K. Kieslich
ARCIC/JAMSD
Radcliff, Kentucky
eberhard.k.kieslich@lmco.com

Roy Zinser
ARCIC/JAMSD
Ft. Eustis, Virginia
roy.f.zinser4.ctr@mail.mil

ABSTRACT

The Battle Lab Collaborative Simulation Environment (BLCSE) is a multi-site, multi-federate, entity-level, real-time simulation environment with the primary mission to enable Army concept development, technology evaluations, and doctrine research. As each lab has its own R&D capability and experimental objectives, simulation baselines, scenarios, and force structures are ever-changing. The physical architecture, simulation protocols, and models are also constantly evolving to accommodate higher entity counts, new combat models, and capabilities. The complexity of the overall system makes it difficult to detect problems as they occur. Protocol issues, configuration errors, and system or network overload conditions can cause cascading effects that are extremely difficult to isolate. Without early identification these conditions can lead to sluggish network performance and even complete simulation breakdowns.

The simulation dashboard is an array of web-based tools that enable real-time management and diagnosis of the simulation network; it helps federation control to maintain a fully operational distributed simulation environment. This combination of GOTS and COTS products ensures network integrity, configuration control, and real-time status of model performance. It shows the exercise director how the scenario is progressing through the use of scoreboards, showing attrition and logistics status among other key statistics. Its logging capability offers an automated restart that allows complete data recovery after catastrophic failures, and facilitates rapid exploration of other exercise branches and excursions.

This paper will describe the dashboard that has revolutionized the management and control of the simulation environment. Rather than spending days debugging and troubleshooting the network during integration or in the middle of an event, this toolset allows us to detect problems as the systems come on line and keep the community informed about the state of the environment every step of the way. The implementation of this capability has greatly reduced setup and integration time and has increased stability and reliability.

ABOUT THE AUTHORS

Eberhard Kieslich is the chief architect for Lockheed Martin Simulation Support Operations. Current work includes BLCSE support, where he performs system integration, testing, capability development, planning, cost and risk analysis. Mr. Kieslich has over 20 years experience in modeling and distributed simulation. He earned a BSEE-equivalent (state-certified Engineer) diploma from Wurzburg Technical College in Germany. He is a certified modeling and simulation professional (CMSP). He received a Lockheed Martin Special Recognition Award in 2006 and the Lockheed Martin Global Excellence Award in 2012.

Roy Zinser served in the U.S. Army for 20 years, initially as an Infantry Officer and then later as a FA57 Simulation Operations Officer. M&S key assignments included postings to the Maneuver Battle Lab at Fort Benning, Georgia and at Space and Missile Defense Command in Colorado Springs, Colorado. Following retirement from the U.S. Army Mr. Zinser is working for Trideum Inc. supporting the TRADOC ARCIC Battle Lab Collaborative Simulation Environment at Ft Eustis, VA.

Simulation Dashboard to Manage Multi-Federate Simulation Events

Eberhard K. Kieslich
ARCIC/JAMSD
Radcliff, Kentucky
eberhard.k.kieslich@lmco.com

Roy Zinser
ARCIC/JAMSD
Fort Eustis, Virginia
roy.f.zinser4.ctr@mail.mil

Introduction

The Battle Lab Collaborative Simulation Environment (BLCSE) is a distributed and collaborative modeling and simulation environment that enables Concept and Capabilities Development across the U.S. Army. It consists of a persistent and secure network enabling collaboration and interoperability across several Army and TRADOC organizations; a federation of constructive and virtual models and simulations with supporting functional interoperability, event management, and data collection and analysis tools; an accessible repository that provides certified scenarios, data, standards, and procedures; and video teleconferencing, white board capability, and voice over internet protocol communications. The BLCSE currently connects over 20 sites via the Defense Research and Engineering Network (DREN). The BLCSE federation connects via the procedures of the High Level Architecture (HLA) standard Institute for Electrical and Electronic Engineers (IEEE)-1516; and Enumerations and protocol data units under procedures of the Distributed Interactive Simulation (DIS) standard IEEE-1278.

The BLCSE is used primarily by the U.S. Army Capability Integration Center (ARCIC) and the U.S Army Training and Doctrine Command (TRADOC) Capability Development and Integration Centers/Battle Labs/Experiment Analysis Elements to support distributed experiments, wargames, and exercises. The TRADOC Battle Labs integrate Doctrine, Organization Training, Materiel, Leadership and Education, Development, Personnel and Facilities and conduct concept and capability development support to specific warfighting functions.

Distributed simulation events take place several times a year. Each year similar issues hinder integration testing. Much time is spent resolving common connectivity issues and simulation configuration problems brought about by infrastructure upgrades, changes of the software baseline, more stringent IA requirements, or addition of new federates to name a few.

To the user or test personnel many of these issues exhibit near identical symptoms. Some of the typical complaints during the first two weeks are: Entities generated at a remote site do not show up at all other locations, while the same remote site sees all entities from other sites, including its own. Shots do not have consistent effects. A new federate joins the Run-Time Infrastructure (RTI), but does not appear on OneSAF, or federates drop in and out. During the FY12 event a never before seen network overload condition occurred, putting the entire event at risk. In distributed environments these overload conditions quickly propagate to every link and can be difficult to isolate.

Layered Architecture

Physical Network

The network infrastructure consists of over 20 independently administered Local Area Networks (LANs) at the Battle Lab sites, the Wide Area Network (WAN), and the Network Operations Support Center (NOSC) located and managed at Ft. Gordon, GA. BLCSE features a classical Hub and Spoke topology as shown in figure 1 below. The links between the Battle Labs and Ft. Gordon are GRE tunnels with a maximum throughput of 40 Mb/s.

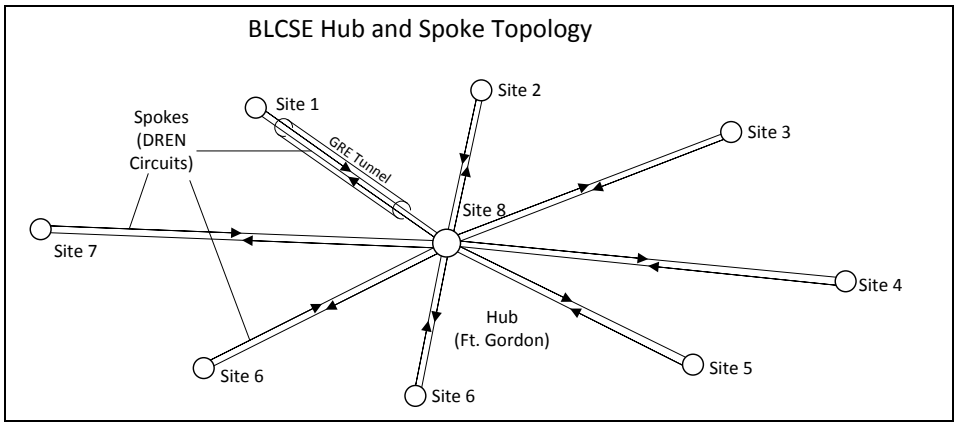

Figure 1. BLCSE Hub and Spoke Topology

On the BLCSE WAN, Open Shortest Path First (OSPF) is used as a routing protocol. All routers and Layer 3 switches on the network are configured for OSPF. When managing an OSPF network, one of the most important things to monitor is changes to the OSPF neighbor states - monitoring how each router within the OSPF network sees other routers within the network. (Keith T. Hutton, Mark D. Schofield, 2008) Some of these state changes are a normal part of the operation of OSPF, but many times a state change is indicative of a problem on the network. There are several ways to monitor these state changes, but typically with either Simple Network Management Protocol (SNMP) based polling and/or SNMP traps.

Physical Network Problems

Driven by Battle Lab site expansion, upgrades or re-structure, changes in the physical LAN topology are very common from one to the next BLCSE event. Physical layout changes affect the logical topology of the network. Perhaps new subnets are established, or additional VLANs have to be routed, etc. If the site had to go through a re-accreditation of its IT system, likely additional modifications to the physical and logical network topologies were directed. Modifications of this type may only have minor impact on a site internally, but as it connects to the BLCSE-WAN far-reaching connectivity problems may result.

Simulation Network

BLCSE leverages the MATREX toolset provided by RDECOM for its distributed simulation needs. It is the HLA interoperability standard supporting simulation war-gaming between proponent simulations located at Battle Lab sites. Through HLA, these simulations interact (that is, to communicate data, and to synchronize actions) with peer systems at other labs regardless of the computing platforms. The interaction between simulations is managed by a central resource called Run-Time Infrastructure (RTI). The communications between federates involve objects and interactions, operating in a publish-subscribe model. A federate can register an instance of an object and then change the attributes. Other federates that are subscribed to the object receive attribute value updates. Interactions work in a similar way, except that an interaction is only used once with a specified set of parameters values and then discarded.

As federates successfully join the federation, these objects and interactions are shared on the network by means of TCP (reliable) or UDP (best effort) packet transfers over the LAN and WAN between hosts. TCP is used for certain interactions deemed critical (single events), such as shots and some reports. Entity state updates are sent as UDP, or best effort. Updates occur frequently (hundreds per minute) and the improbable loss of a single message is deemed insignificant. Table 1 below shows some of the common messages used.

Table 1. Common HLA Messages

Message Type	Description	TCP/UDP
AirPlatform	Entity state information for Air Platforms	best effort
CallForFire	Message sent to request fire support	reliable
CMSC2Collaboration	Carries signals and messages between CMS and the field controller (C2) simulator.	reliable
CounterMineAction	Interactions that report countermine actions	reliable
CounterMineControl	Sets the countermine behavior of the designated platform.	reliable
DamageReport	Sent out by anyone who assesses damage on an entity after receiving a MunitionDetonationinteraction.	reliable
FireMission	An interaction intended for sending a fire mission to a platform that is associated with an FDC.	reliable
FWAUAVSweep	Tasks a fixed wing UAV to execute a sweep mission	reliable
GroundPlatform	Entity state information for Ground Platforms	best effort
ICPlatform	Entity state information for IC Platforms	best effort
MoveAlongRoute	Directs a vehicle to move along a route consisting of one or more routepoints	reliable
MunitionDetonation	Communicates information associated with the impact or detonation of a munition	reliable
MunitionPlatform	Entity State information for Munition Platforms.	best effort
SaluteReport	Reports friendly entity information such as kill reports, location reports, task completion reports, weapon reports, encounter reports, current strength, supply status, etc.	best effort
WeaponFire	Communicates information associated with the firing or launch of a munition.	reliable

Besides the RTI, the BLCSE simulation architecture contains a number of other central resources that aid in consistent performance of the simulation. The Damage Effects Server (DES) processes damage for the entire federation, and the Dynamic Terrain Server (DTS) generates building damage, rubble, craters etc. based on indirect fire detonations. Additionally, JMX and BONST are used to manage OneSAF clusters, at critical sites from the Ft. Eustis Control Center. The Joint Virtual Tactical Radio (JVTR) is a real-time simulation of voice radios used throughout BLCSE. JVTR uses the DIS protocol. Data packets containing encoded voice data are sent half-duplex via UDP multicast.

A variety of tactical Mission Command systems are used as situational awareness tools or to emulate Mission Command operations at various echelons. These systems are stimulated by the HLA network via a number of interface systems, which translate HLA entity state information into JVMF position reports for instance. Some federates provide a direct output in tactical message formats. These data streams are typically unidirectional and provide no direct feedback to the simulation. Additionally, several web-based applications are employed such as streaming video, Chat, Adobe Connect, and SharePoint.

Simulation Network Problems

The MATREX Federation Object Model (FOM) utilizes a wide range of ports for its message-sharing architecture. The assumption is that required ports are open and essential protocols are forwarded by all devices, including firewalls. However, in actuality this assumption is deeply flawed. Changes in the physical and logical network topologies since the last BLCSE exercise may have compromised LAN/WAN integrity for simulation use.

During the first few weeks of Spirals 1 and 2 the simulation may work only partially, inconsistently or not at all. The impact of partial or marginal LAN/WAN connectivity on the simulation environment differs significantly from case to case, is difficult to pinpoint and causes delays in integration testing. Objects and Interactions may be shared between some federates, but not others for example. In some cases only a subset of the objects or interactions are shared, but not in all. The MATREX toolset does not provide visibility into the lower layers for fault isolation and offers only limited federation management ability, which leads to uncertainty and frustration of the test personnel and delays in testing.

Dashboard

The idea of the simulation dashboard is to automate the majority of steps an engineer routinely performs when troubleshooting a problem. One of the very first steps is typically to ping remote interfaces. Providing the right balance of monitoring is not easy, and the purpose of the dashboard is not to replace the knowledge and experience of the technical staff, but to enable the operations staff to quickly determine who to contact about a possible technical issue.

For the dashboard to effectively provide this capability, a combination of COTS tools and custom applications is employed. The concept of a "Single Pane of Glass" provides a very simple top view to show the operator at a glance that the network is intact, traffic is flowing and load levels are within a "normal" range (see Figure 2). A consistent coloring scheme is used throughout. The underlying software performs a number of checks on a 60s to 300s sample rate, logs data into databases and displays it on the screen(s).

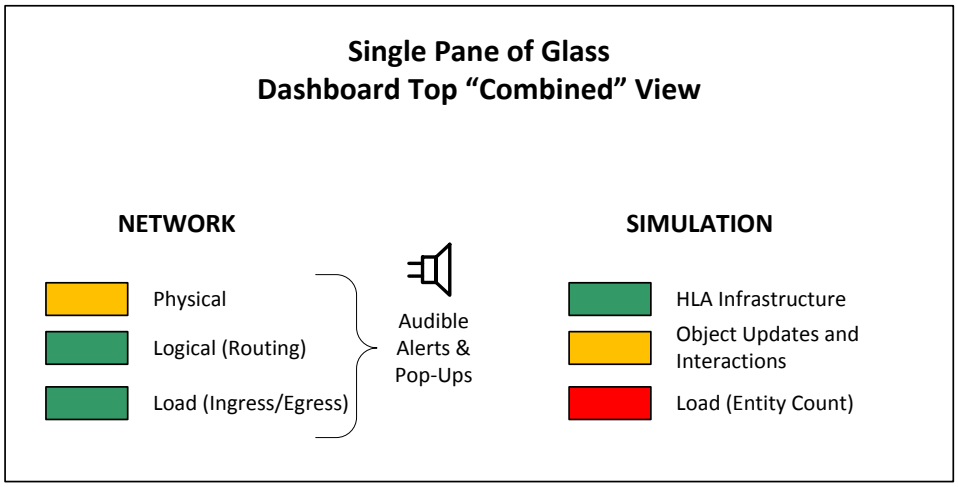

Figure 2. Dashboard Top View

SNMP and Netflow are used for measuring utilization and bandwidth. Netflow is a network protocol developed by Cisco Systems for collecting IP traffic information. The tool of choice for monitoring the physical network is Solarwinds Orion Network Performance Monitor (NPM). It gathers SNMP and Netflow data along with routing protocols and a number of other network protocols, interface and CPU status information from all devices on the network. When trying to understand why a particular icon is not green, a "drill down" to layers below is possible. In case of the example of an amber physical connection, the next layer shows a map view of the sites and their connections to the Ft. Gordon NOSC (see Figure 3).

The same color coding is used to show the state of the links. A green site icon and link mean the connection is stable and the interfaces respond and the network protocols are traversing over the connections. A yellow icon identifies an overheating power supply, a clogged air filter in a critical device or an intermittent connection. Conditions like these often lead to outages later, if not addressed.

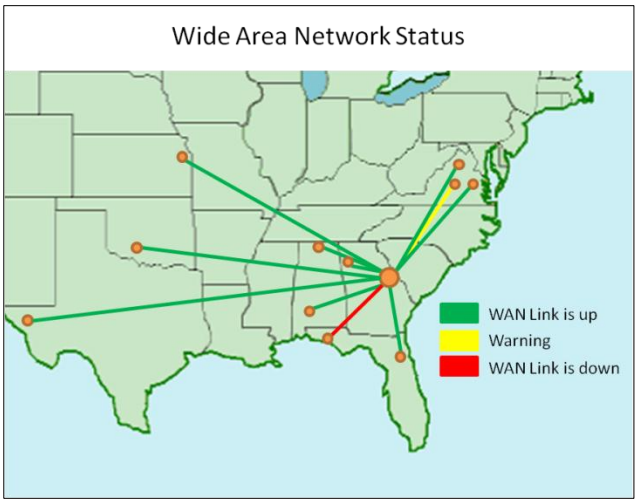

Figure 3. WAN Status Map View

A red line means the link is down. Either the actual network connection is lost or an interface on a switch or router directly connected fails to respond. In any of those cases, the NOSC would be responsible to address these issues. If a line was green, but a particular computer at that site could not be reached, the problem would be on the site's local network. In the case just described, the dashboard helped the Operations Manager readily determine the nature of the problem and then provided a detailed description to the site's network administrator. LAN/WAN connectivity is no doubt one of the most fundamental prerequisites for communication between the HLA federates to take place.

The NPM dashboard features an interactive web-design, allowing quick expansion of views, if the situation calls for monitoring of specific measures during testing. For instance, an OSPF routing issue can be monitored and analyzed on the respective pages. Multicast Routing of the simulation protocol across WAN and LAN firewalls have become an emerging issue during the FY13 initial testing phases. Several sites added internal firewalls and restricted ports and protocols enough for the HLA traffic to get partially blocked. This tool allows monitoring of Multicast routing on the WAN, but at this time offers only limited ability to trace multicast routing issues from source (site LAN) to destination (other subscribed sites) and vice versa.

A host of other requirements for the physical network exist for a stable distributed simulation environment. One of the most critical requirements is that allowable bandwidth not be exceeded at any link at any time or packet loss is imminent. Packet loss degrades the performance of the simulation, may cause federation-wide crashes and affects data collection. (J Scott Haugdahl, 2000) Latency of the traffic is another measure to observe with similar effects, as it is also related to an overload condition. This web-based tool shows gauges and line charts for ingress and egress of bandwidth in real-time for each of the GRE Tunnels, which can be viewed at any browser-equipped host on the network. However, just determining that the total traffic across a particular tunnel is too high is not going to help to identify the application. A variety of administrative applications are also used during war-gaming, such as VTC, VoIP, JVTR, Streaming Video, and a suite of Mission Control technologies (some stimulated by the HLA network). Additionally, there are several web-sites, portals and data sharing on SFTP sites. All of these are managed by different groups and at different locations.

Using the collection of Netflow data in NPM and programming the tool to monitor ports and network protocols allows us to track traffic by application on each WAN connection in a combined, continuous timeline view (see Figure 4). All known traffic categories are displayed in real time as shown below. NPM was programmed to monitor the vast majority of network issues seen on BLCSE to date.

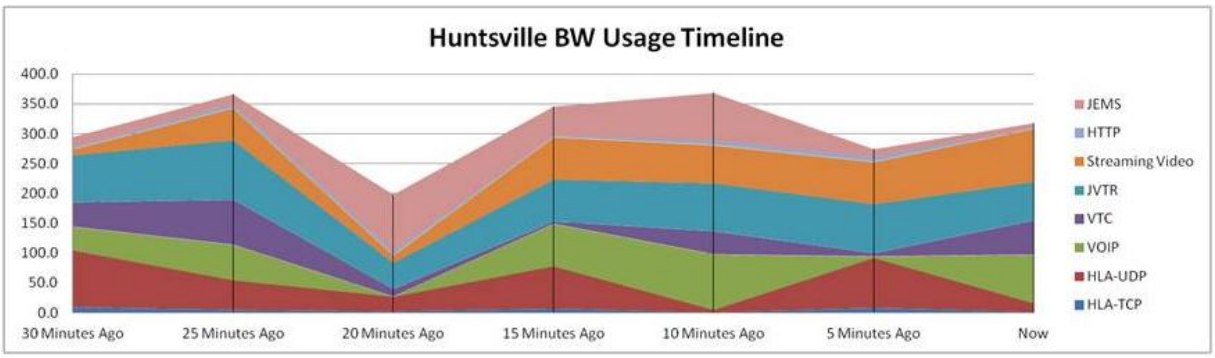

Figure 4. Traffic by Application

During Spiral testing as federates come on line, important benchmarking can take place to determine the limits for each category. Although a relationship between network load and entity count exists, much of it is determined by the scenario and full load at the height of the war-gaming scenario is difficult to predict. Having the Netflow tool and Federation Reporter – Entity Count - pages side by side as shown in Figure 5 enables the test director to evaluate network load and to ensure the scenario progresses as planned by exercising the congestion mitigation plan before an overload condition is present.

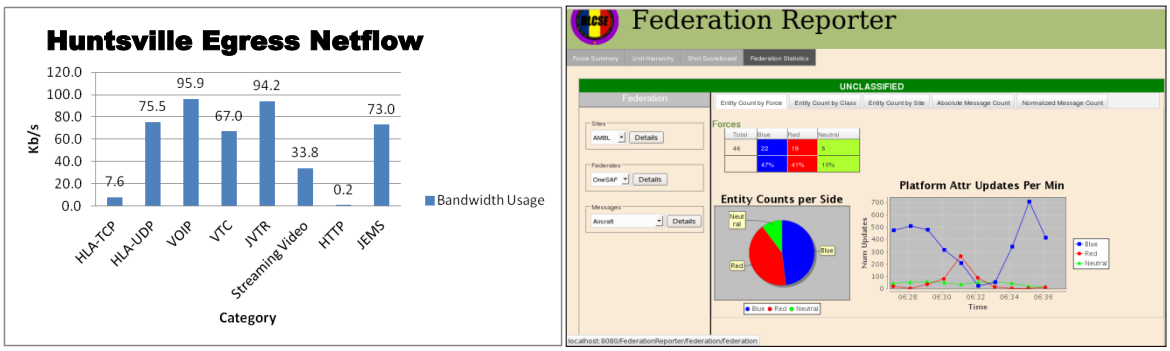

Figure 5. Netflow Traffic Display and Federation Reporter Entity Count

Coincidentally, the use of the Mission Control tools is likewise very scenario-dependent. As integration progresses, Spiral testing is using more and more of the full suite of equipment and applications and the scenario becomes more representative of the actual SIMEX. Running the physical network within these defined load limits, is an indispensable first step to achieving stability during a simulation event. For this purpose the dashboard is shown on a large flat panel at the Ft. Eustis Simulation Control Center. Audible alerts also exist to draw the staff's attention to the screen in case a critical link goes down or a threshold is exceeded.

However, the actual work of the test personnel involves evaluation of new software baselines with new innovative features, explicitly modeled simulated ground, air and human entities maneuvering on a just-compiled terrain database. The Mission Command architecture is steadily evolving as well. There are many new variables. Load is no longer just network load, but load generating realistic behaviors and performance of the entities and units in the simulated environment. Some computations can be very CPU-intensive, and may cause a slow-down of the scenario similar to a congested network. The terms simulation load and network load describe different underlying causes, not to be confused.

Additional tools associated with the applications (federates) and the distributed architecture are required to instantaneously evaluate problems with the simulation. A tool called the Federation Reporter directly interfaces with the HLA network and provides key details and analytics about the simulation environment by dissecting the HLA

traffic. This capability was developed at ARCIC to support debugging and analysis of HLA during the integration process. The Federation Reporter is likewise web-based and provides the users a variety of statistics in real-time. Examples are:

- Message Count by federate, providing message update rates, attribute updates over time
- Entity count by site, force, class, absolute count, normalized counts
- Graph of update rate for a given federate class (i.e. OneSAF)
- Graph of update rate for a given message type (i. e WeaponFire)
- Health status of force structure (Task Organization)
- Shot scoreboards, showing shots for a given task org in tabular fashion

Figures 6 through 8 show screen shots of some of the pages of the Federation Reporter.

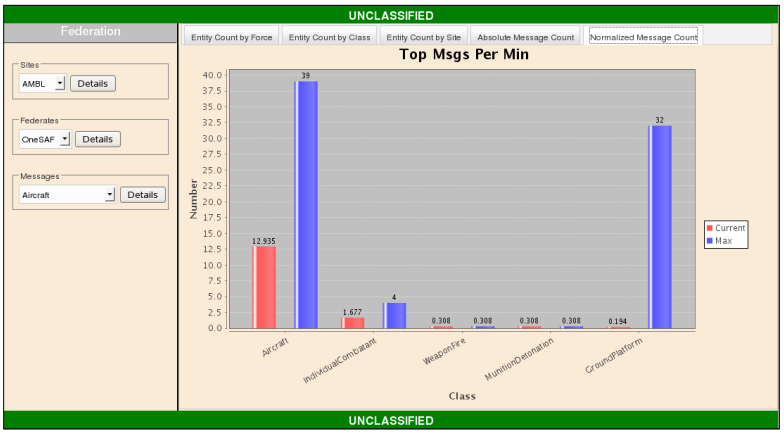

Figure 6. Top HLA Message Counts per Minute

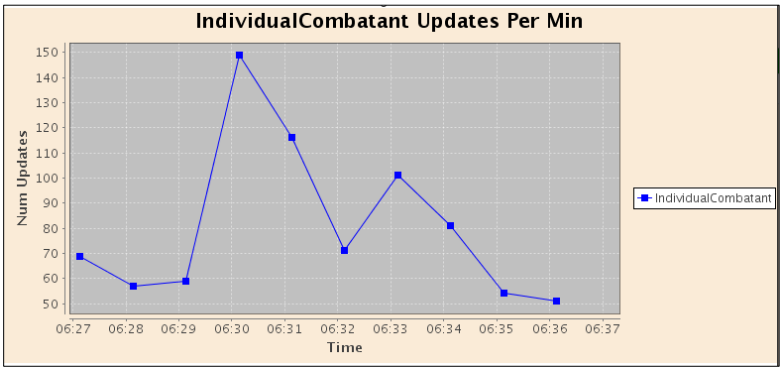

Figure 7. IC Platform Updates per Minute

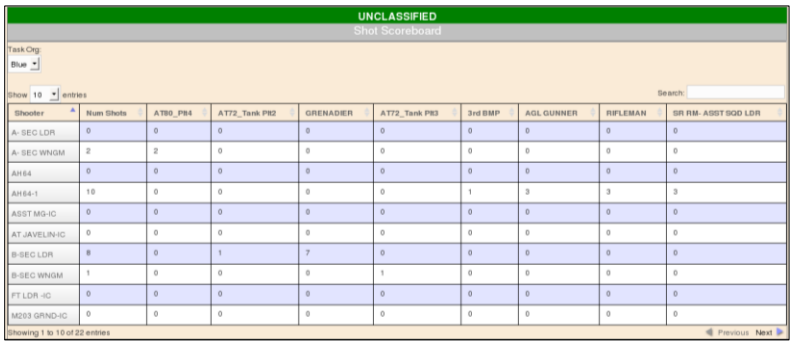

Figure 8. Shot Scoreboard

Message counts, entity counts and update rates are especially meaningful when compared with previous benchmarks from a Spiral test event. The other charts showing health status of force structure scoreboard information expand the usefulness of the dashboard to the experiment director and his staff. As the main simulation driver on BLCSE, OneSAF is equipped with a variety of web-based monitoring tools that provide an additional fault isolation capability. The BLCSE OneSAF Node Status Tool (BONST) provides a customizable web-interface that lets a user monitor particular OneSAF machines across the network. The tool shows memory allocation, CPU load, entity count, configuration information and other relevant information about the health of any particular OneSAF workstation on the network. This tool allows the test personnel to focus on a failure or degradation that is more localized.

Another beneficial tool is Java Media Extension (JMX). Part of the OneSAF distribution, it provides a web interface to inspect and verify property settings and other configurations within the various components of the OneSAF simulation engine for consistency and configuration management purposes. It also offers some controls such as resetting the operator's plan view display (PVD) without the need to restart the application and re-join of the federation. The MATREX RTI provides a table of joined federates. As mentioned before, the RTI monitoring capability is limited to connectivity of its network sockets to federates.

The DES is a central resource, developed at ARCIC, that adjudicates combat damages caused by direct and indirect fire to the entire federation. The tool provides a scrolling output of recent damage assessments whether effects were applied or not. The DES screen updates instantly. These updates are often viewed during debugging of vehicle lethality and vulnerability checks, but can also be useful, if claims are made that a particular munition does not seem to produce effects in the experiment. Future upgrades to DES may include a web-interface to allow real-time view of log entries from anywhere on the network.

Benefits

The hierarchical use of these tools and their dashboard-like outputs, automate the majority of initial steps in the troubleshooting process of the network and the simulation architecture. They also give the operations and test personnel certainty about the stability of the environment. Some of the more advanced pages specifically benefit the experiment director. These pages provide a wide range of status information about the force structure, such as attrition rates by force, shots fired, kills, fuel consumption and distance traveled and many more. In a complete free play environment the scenario may play out differently each time. This capability enables the experiment director to make scenario adjustments early to ensure objectives are met within the time on hand.

In spite of the decisively different mission of each experiment, the approach of using a simulation dashboard in conjunction with the concept of spiral testing enabled ARCIC to take on more experimental objectives without

risking loss of stability due to overload conditions. The tools themselves provide a very minor load (<1%) to the network, but offer tremendous advantages, such as

1) Identify potential problems before they impact the federation
2) Greater stability
3) Improved load management
4) Shortening of integration timeline
5) Quicker fault isolation
6) Prediction Analysis
7) Answers ad hoc requests from director
8) Historical data
9) Information sharing via web interface

Before this innovation, problems that took several days to properly identify are now identified as they happen, or potentially avoided altogether, as the simulation dashboard alerts the staff to issues that can lead to outages and failures. This tool has not only shortened the integration timeline immensely, but has drastically reduced simulation downtime. The experiment director now also has visibility into the progression of the experiment as it unrolls, and can re-direct roleplayers and computer operators to achieve their mission objectives as required.

Summary

The BLCSE is a unique and powerful capability of the U.S. Army and the Department of Defense. It provides a cost-effective way to evaluate Army and Joint war-gaming strategies and collaboration. Previously, technical and administrative challenges were partially addressed by adding VOIP, VTC, and communication portals to the BLCSE arsenal. Adding the Simulation Dashboard's automation and web-centric design enables exercise technical personnel and operations staffs to recognize chokepoints early on, mitigate risk, and focus on the technical aspects of the exercise. The Federation Reporter pages enable the experiment leadership to gain emerging insights, and facilitate early intervention to keep roleplayers focused on the objectives. As all data displayed on the dashboard is also logged into SQL databases, historical data can be analyzed and predictions about load and other measures are possible from one event to the next. With the addition of this toolset, the BLCSE becomes an even stronger collaboration environment. This technology could, at least in concept, have wider applicability for other projects that have complex networking configurations or distributed resources.

ACKNOWLEDGEMENTS

We would like to acknowledge the following individuals for their support. Paul Monday's advice was significant in the overall approach of an automated test tool set that eventually led to this solution. Michael Cobb, Javier Ortizdelgado, Donnie Podhorsky and Brad Richmond fellow developers for helping to define, develop and evaluate the Simulation Dashboard. Thanks to Hans Hess for setting up a myriad of test scenarios in the development lab. And finally, to LTC Kevin Butler, Larry Rieger, Warren Jones, LTC Linda Butler, Donnell Walker, Mark Morrell and the entire BLCSE Community of Practice for their patient provision during testing and implementation of this capability.

REFERENCES

J Scott Haugdahl. (2000) Network Analysis and Troubleshooting, Location: Print on Demand

Keith T. Hutton, Mark D. Schofield & Diane Teare. (2008) Designing Cisco Network Service Architectures (ARCH), Location: Pearson Education