Advanced Tools and Techniques for Gateway Performance Testing

Michael J. O'Connor,
Kurt Lessmann, Jim Chase
Trideum Corporation

Huntsville, AL
moconnor@trideum.com,
klessmann@trideum.com,
jchase@trideum.com

Robert Lutz, David Drake

The Johns Hopkins University
Applied Physics Laboratory
Laurel, MD
Robert.Lutz@jhuapl.edu,
David.Drake@jhuapl.edu

Dannie Cutts

AEgis Technologies

Huntsville, AL
Dcutts@aegistg.com

ABSTRACT

Modern Live, Virtual, and Constructive (LVC) simulation environments are highly complex systems. The integration of numerous heterogeneous simulation components and supporting utilities (e.g., viewers, loggers) into a coherent, logically unified, and internally consistent test or training environment is extremely challenging. Additional complexities may also include the need to reconcile differences in the way individual LVC components exchange data at runtime and the need to adjudicate across dissimilar simulation services when multiple simulation architectures are employed in the same LVC environment.

Gateways are intelligent translators that are widely used in the simulation community to translate among the different simulation protocols and data formats that may be present within a given LVC environment, enabling operation across dissimilar architectures. Although gateways are commonplace in LVC events and have a history of effectively accomplishing their stated purpose, there are also a number of well-documented gateway issues that increase both cost and schedule risk for LVC applications and can also adversely affect technical quality. The LVC Architecture Roadmap Implementation (LVCAR-I) is addressing these challenges via a set of new products that allow LVC developers to make better, more informed choices on the gateway that best aligns with their application requirements while also streamlining the process of defining all necessary gateway translations and configuring the gateway for runtime operation.

This paper focuses on the need for gateway performance testing. A Gateway Performance Benchmarks (GPB) Specification was developed to define formal measures for gateway performance along with explicit use cases in which the benchmarks could be applied. The next phase focused on the development of supporting test methodologies, a gateway performance test harness design, and an initial instantiation of the test harness design. These products collectively define an integrated mechanism for measuring gateway performance that allows for direct comparisons of performance characteristics across multiple gateway products.

ABOUT THE AUTHORS

Michael J. O'Connor is a Senior Principal Engineer at Trideum Corporation. Mr. O'Connor has more than 20 years experience in Modeling and Simulation (M&S). He has been a key participant in the development of distributed modeling and simulation standards, including IEEE 1278 and IEEE 1516. He has held many positions in the community, including Chairman of the SISO Standards Activities Committee and Vice-Chairman of the SISO Executive Committee. Mr. O'Connor previously led the development of ITT's Chemical, Biological, Radiological, and Nuclear Simulation Suite, which supports Distributed Interactive Simulation (DIS), High Level Architecture (HLA), and Test and Training Enabling Architecture (TENA). He holds a bachelor's degree in Computer Engineering from Auburn University, and a master of science in Computer Science from the University of Alabama in Huntsville.

Robert Lutz is a principal staff scientist at the Johns Hopkins University Applied Physics Laboratory (JHU/APL) in Laurel, Maryland. He has over 30 years of experience in the design, implementation, and evaluation of M&S

systems for military customers. Currently, he is leading the LVCAR-I "Systems Engineering (SE) Process" and "Common Gateways and Bridges" tasks, and serves as the U.S. Navy's Broad Area Maritime Surveillance (BAMS) Program M&S lead in the airspace integration area. He also leads several M&S standards initiatives within the Simulation Interoperability Standards Organization (SISO), including the Object Model Template (OMT) component to the Institute of Electrical and Electronics Engineers (IEEE) 1516 High Level Architecture (HLA) standard, and the IEEE 1730 Distributed Simulation Engineering and Execution Process (DSEEP) standard. He also serves as a regular guest lecturer in The Johns Hopkins University (JHU) Whiting School of Engineering.

Kurt Lessmann is a Vice President at Trideum Corporation, specializing in providing solutions and services in the area of test and evaluation, M&S, systems analysis, and information technology. Mr. Lessmann has over 15 years experience in supporting LVC integration activities for large-scale Joint Distributed Test activities. Mr. Lessmann also supports the test community by providing Object Model design, software and range system design and integration, and application of M&S techniques. He currently supports the TENA project and the Joint Mission Environment Test Capability (JMETC) Program Office as a senior Event Lead. Mr. Lessmann holds a bachelor's degree in Aerospace Engineering from Auburn University.

Jim Chase is a Project Engineer at Trideum Corporation, with over 18 years experience as an engineer, analyst and generalist specializing in areas of project management, requirements development and management, documentation development, quality management and process improvement, all applied to the task of developing strategies, solutions and processes to solve complex problems. Formerly an embedded software engineer, Mr. Chase has provided concept development and design services in support of software and web-application development, test planning, test integration, test execution and process improvement in both industry and Government ventures, most recently in support of U.S. Army Test & Evaluation Command (ATEC) and System of Systems Integration Directorate (SoSI) initiatives. Mr. Chase holds a Bachelor's of Electrical Engineering degree from Auburn University.

David Drake is a member of the Senior Technical Staff in the Global Engagement Department at JHU/APL, performing research and development in the area of modeling and simulation supporting the U.S. government and military. Mr. Drake supports the LVCAR-I "Common Gateways and Bridges" task and was the lead on the LVC "Service-Oriented Architecture" task. Mr. Drake has 10 years experience in M&S design, development, and standards and 24 years as a computer security professional in computer security design, implementation and evaluation. Mr. Drake is a member of the SISO Standards Activity Committee (SAC) and is the chair of the SISO Board of Directors (BoD). Mr. Drake received his bachelor's degree in Mathematics from State University of New York (SUNY) at Buffalo. He is published in the areas of simulation, service-oriented architecture, grid computing, and security, and has a patent on the process for enterprise-wide intrusion detection.

Dannie Cutts is a Senior Computer Scientist with AEgis Technologies Group Inc., supporting the Joint Staff J7 Joint and Coalition Warfighting (JCW) organization in Suffolk, Virginia. He has over 20 years experience in M&S for NASA and DoD and has been involved with the HLA since 1995, serving on the Interface Specification and Time Management Working Groups. He has provided HLA Training and Cadre support for the Defense Modeling and Simulation Office (DMSO), and currently serves on the IEEE Drafting Group for the HLA IEEE 1516 standard. Mr. Cutts is a Certified Modeling and Simulation Professional (CMSP). At JCW he is involved in efforts to improve interoperability between LVC assets for Joint Training. Mr. Cutts holds Masters Degrees in Mathematics and Computer Science from the University of Mississippi.

Advanced Tools and Techniques for Gateway Performance Testing

Michael J. O'Connor, Kurt Lessmann, Jim Chase
Trideum Corporation

Huntsville, AL
moconnor@trideum.com, klessmann@trideum.com, jchase@trideum.com

Robert Lutz, David Drake

The Johns Hopkins University Applied Physics Laboratory
Laurel, MD
Robert.Lutz@jhuapl.edu, David.Drake@jhuapl.edu

Dannie Cutts

AEgis Technologies

Huntsville, AL
Dcutts@aegistg.com

INTRODUCTION

The need to measure the performance of distributed simulation gateways was identified as part of the Live, Virtual, Constructive Architecture Roadmap Implementation (LVCAR-I) (JHU, 2010). The LVCAR-I is an effort to implement some of the recommendations in the LVCAR Final Report (IDA, 2008). The LVCAR and LVCAR-I are widely documented and will not be discussed further here. The Bridges and Gateways efforts have also been documented in several technical papers (Lessmann, 2011) (O'Connor, 2011). The LVCAR-I bridges and gateways team has focused on enhancing the user's gateway experience. The enhancements for gateway users are divided into selecting a gateway and using a gateway. One key element of selecting a gateway is to determine if it meets the performance requirements of the event or exercise. There is no current standard to measure the performance of a gateway. This has made it difficult for users to specify their performance needs and for gateway vendors to provide standard data. The methodology presented here addresses these issues.

OVERALL CONCEPT

The purpose of the Gateway Performance Test Methodology and Test Harness is twofold: to allow gateway users to determine if gateways meet their performance requirements and to allow the gateway vendor to advertise the performance of their gateway to prospective customers. The Gateway Performance Test Methodology and Test Harness defines a repeatable process and a structured test environment for producing performance benchmarks (JHU, 2011). To allow the user to select the gateway that is the closest match for their requirements, the performance results for each gateway must be comparable. Therefore the specified test methodology and harness must be sufficiently explicit to allow gateway users or vendors to conduct the tests in a repeatable manner. Gateway vendors may be hesitant to publish the performance data for their gateway unless they believe the other vendors are following the same well-defined method. The process defined here will allow users or vendors to generate performance data for gateways and can further be used to determine which gateway is the best fit for a federation's requirements.

While the concept of measuring the performance of a gateway seems straightforward, there are significant challenges. In the early days of super computers, the key metric was millions of floating point operations per second (MEGAFLOPS). This was very useful as most super computer applications were heavily dependent on floating point operations. Gateways are not so simple. A gateway performs a number of different types of operations including network access, geographic coordinate conversation, database access, and data storing. This does not lend itself to a single metric. The performance of a gateway is also dependent on the types of federations it is connecting. For this discussion a federation is a group of federates (including a group of one) using a common Architecture/Simulation Data Exchange Model (SDEM). A gateway may be able to process updates very quickly for a light traffic load, but not so quickly under a heavy traffic load. To address these issues the Gateway Performance Test Methodology process defines a set of Use Cases and Metrics.

Because a gateway may perform differently depending on the federations connected, a single test will not allow users to evaluate if a gateway meets their requirements. The user needs the test to be run in an environment that is similar to their intended use. Vendors need the ability to demonstrate a particular gateway's performance strength or the ability to support a wide range of operational environments. To support these needs, a set of Use Cases were defined to cover a wide range of likely federations.

A set of metrics was also defined that addresses the concerns of gateway users. The set of metrics covers gateway specific measures. A user or vendor would develop a set of metrics for each Use Case.

It is critical that each test be conducted the same way. A Gateway Test Methodology was developed to ensure that the performance tests can be performed by multiple organizations in a repeatable manner. The methodology is divided into five groups that are each composed of multiple steps. Each step has a set of inputs and outputs and directions on how to perform the step.

In addition to a methodology, a well-defined test harness is also required to support performance testing. A design for the overall test harness is shown in Figure 1. The test harness is composed of three components: Traffic Generators, Data Collector, and Data Analyzer. These components are selected and configured to support a "black-box" test environment to ensure consistency in the collection of gateway performance metrics.

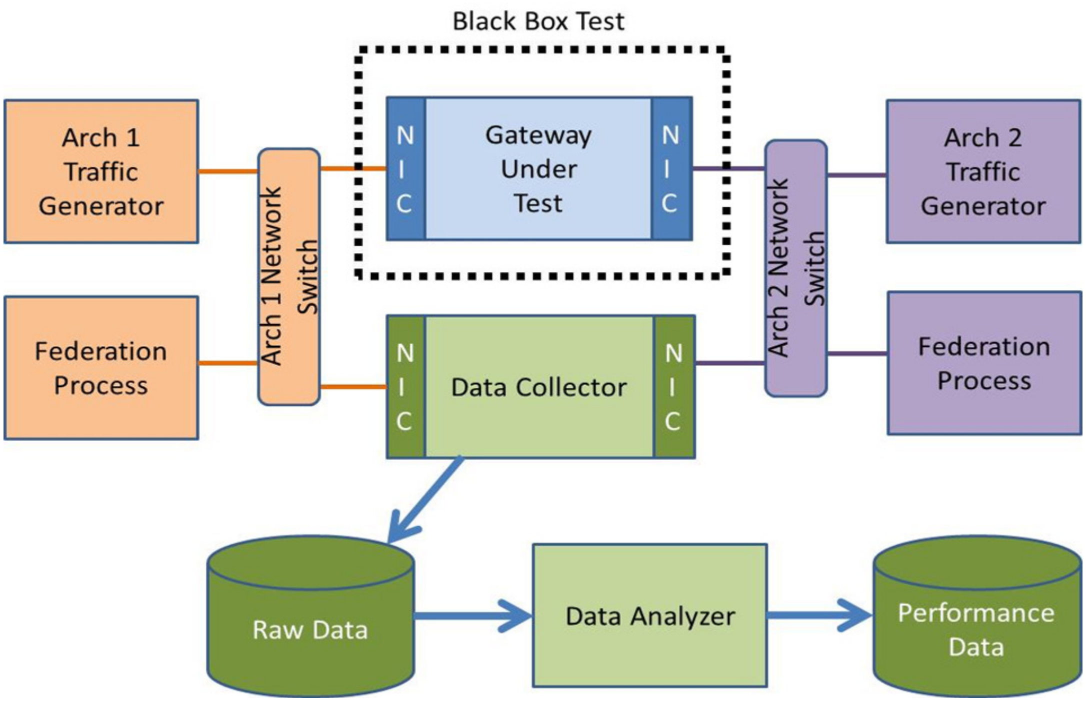

Figure 1. Test Harness Design Concept

USE CASES

A key element of the gateway performance benchmarks is generating performance data based on realistic user environments. Federations vary greatly based on their purpose. To address this, a set of Use Cases covering a wide range of federation types was developed. This allows the user to select the performance numbers that are based on a test that is close to their environment. The full set of Use Cases is defined in "Live-Virtual-Constructive Architecture Roadmap Implementation Common Gateways and Bridges Gateway Performance Benchmarks" (JHU, 2011).

The Use Case parameters define the number of objects in the two federations connected via the gateway. For example, the total number of persistent objects in the Medium Persistent Object Count value is 1,000 with the objects split equally between the federations on each side of the gateway. This applies to the Transient Object Count and Update Rate for Persistent Objects as well. All parameter values can vary from the stated values by up to 10%.

The Persistent Object Creation and Deletion dynamic value will change the number of persistent objects present at one time. For the dynamic value, 50% of the total number of the persistent objects shall be created at the start of the test. The remaining objects shall be created once 25% of the total execution time for the test has been reached. Once the full number of persistent objects has been created, 50% shall be destroyed in the next 25% of the execution time. This process shall be repeated once. Table 1 shows the parameters for each Use Case and their definition.

Table 1. Use Case Parameters

Parameter	Definition
Persistent Object Count	The number of persistent objects the gateway will have to process. Note that this does not refer to the number of persistent objects in the federation, but rather how many will be processed through the gateway.
Transient Object Count	The number of transient objects the gateway will process.
Update Rate for Persistent Objects	The rate at which the attributes of persistent object is updated. The updates must be consistent with the rules of the Architecture/SDEM
Traffic Pattern	Traffic patterns can be generated in either a continuous or burst mode for persistent and transient objects.
Complexity Of Translation	The level of computational difficulty to translate between different simulation data exchange models.
Persistent Object Creation and Deletion	Persistent Object creation and Deletion types are based on the time the objects are created relative to the duration of the simulation exercise.

METRICS

Because of the unique and diverse application of gateways by Federation users, developing a set of performance metrics that are relevant to all users is non-trivial, but is critical to the success of an approach for a consistent and usable Gateway Performance Metrics system. The LVCAR-I Gateways Team originally defined six performance metrics for gateways. These were documented and presented to a group of gateway users and developers in a workshop setting. The list included resource utilization, latency, throughput, scalability, stability, and accuracy. The group attending the workshop discussed the usefulness of each metric and how it would be measured. After much discussion the group determined that latency and throughput were the primary considerations. The group determined that the other metrics could not be measured in a repeatable manner or that they were not as important to users in selecting a gateway.

The next issue in measuring performance was how to collect the metrics data. The test harness concept describes how the metrics will be collected. This method, where the data is collected external to the gateway, was selected because it did not impact the performance of the gateway under test and it did not require modification to the gateway. All of the data required for the metrics are captured by the data collector.

When researching data collection techniques, issues were raised on how the metrics could be derived from the collected data. The issues were based on the differing rules of the potential architectures and SDEMs that could be implemented for persistent objects in gateways. Persistent objects are created and exist in the simulation environment for a period of time and then may be destroyed. The attributes of persistent objects may also be updated. Transient objects are published one time and are not updated.

Latency and throughput are relatively easy to derive for transient objects. The transient objects selected for the test will have representations in both selected Architecture/SDEM pairs. Each time a transient object is received by the gateway on one side a corresponding transient object will be generated on the other side. The latency is simply the delta between the time the message is received on the first side and the time the message is published on the other side of the gateway. Throughput is generally measured in the amount of data passed between two points. The transient object throughput rate is the number of transient objects passed through the gateway in a given amount of time (one second).

Measuring throughput and latency with persistent objects is more challenging, as an update from the Architecture/SDEM on one side does not always generate an update on the other side. Also a gateway may need to publish an update not based on the receipt of a persistent object, but based on the rules of the Architecture/SDEM. This is why analysis profiles are critical to the calculation of the metrics. The Data Analyzer has to determine which persistent object updates should generate updates on the other side of the gateway. The received updates from the Traffic Generator that should not generate outgoing updates on the other side of the gateway are not counted as dropped and do not affect the latency numbers. Persistent Object throughput is calculated three ways: 1) persistent object updates that require publishing an update on the other side, 2) persistent object updates that do not require publishing an update on the other side, and 3) persistent object updates that are required but are not the result of a received persistent object (i.e. heart beating required by DIS). These are all measured in object updates per second.

Some distributed simulation architectures do not require the full set of persistent object attributes to be published. DIS requires the full object definition to be published even if only one field has changed. HLA allows for only the changed attribute of a persistent object to be published. Because of this, calculating gateway throughput relative to object size is dependent on the architectures selected. This is the reason object updates per second instead of bytes per second were selected to measure throughput.

Table 2 lists all of the metrics calculated by the Data Analyzer. Each metric is calculated for each side of the gateway. The Data Analyzer will calculate a total of 16 metrics for each test, eight for each side.

Table 2. Gateway Performance Metrics

Metric	Calculation
Transient Object Latency	Average of (Receive Time Side 1 minus Receive Time Side 2 minus Switch Latency). **Note**: the calculation of Switch Latency is discussed in the Methodology
Transient Object Throughput	Total number of Transient objects / total seconds in test
Number of Dropped Transient Objects per second	(Number of transient objects received by Side 1 minus Number of transient objects published on Side 2) / total seconds in test
Persistent Object Latency	Average of (Receive Time Side 1 minus Receive Time Side 2 minus Switch Latency)
Persistent Object Throughput – Published	Total number of Persistent objects / total seconds in test
Persistent Object Throughput – Not Published	Total number of Persistent objects / total seconds in test
Persistent Object Throughput – Required Published	Total number of Persistent objects / total seconds in test
Number of Dropped Persistent Object Updates per second	(Number of persistent objects received by Side 1 minus Number of persistent objects published on Side 2) / total seconds in test

METHODOLOGY

General performance testing and benchmarking processes are designed to provide a common and controlled test environment suitable to test and assess the performance of components or subcomponents within an integrated solution, whether that solution be in a system or software paradigm, to support comparative, competitive and verification analysis for the component under test. While similar, the distinction between performance testing and performance benchmarking is the intended use of the results. Typically, performance testing allows comparison of a product to a set of performance requirements, whereas performance benchmarking assesses the relative performance of a product against that of similar products. Developing a methodology to apply this type of testing to gateways involves a systems engineering process that identifies, defines and groups a series of related activities or steps to enable a structured, repeatable approach that meets the needs and requirements of vendors and users alike. For gateway performance testing and benchmarking, the methodology consists of five primary stages: Gateway User Planning, Gateway Vendor Planning, Test Setup, Test Execution and Data Analysis.

The Gateway Performance Test Methodology offers two points of entry: Gateway User Planning and Gateway Vendor Planning. These two stages are similar in structure, but are tailored to the unique requirements of each distinct community. Vendors will typically engage the methodology with a focus on benchmarking their product against similar products, while users will apply the methodology to assess the performance of specific gateways against predefined Federation requirements necessary to support specific distributed simulation events. Each community, however, shall leverage a set of predefined gateway performance Use Cases, select the Architecture/SDEM pairs for each side of the gateway interface, and select appropriate hardware to support the desired gateway configuration. The planning stage(s) are followed by the typical testing sequence of Test Setup, Test Execution and Data Analysis. Test Setup involves the stand up and configuration of the gateway performance test environment ("test harness") to include selections and connections for switches, traffic generators, the data collector and any supporting Federation processes. Also included in Test Setup is the creation or selection of scenario files and the analysis profile based on Use Case and Federation requirements. These inputs are then loaded into the gateway test environment and activated during the Test Execution stage. At the conclusion of the test, the captured traffic and data are written to logs, which in turn are loaded into a data analyzer (along with the analysis profile) to support post-test Data Analysis. Gateway performance and benchmarking reports are the end products of the Data Analysis stage and of the overall methodology.

Gateway User Planning

Gateway users may leverage the performance test methodology to meet two primary objectives: 1) to review existing benchmarks and determine which gateways best meet requirements; and 2) to verify the performance of the selected gateway within an operational environment that closely matches a specific distributed simulation environment and scenario. Before entering the process, the user typically pre-selects a gateway. The Gateway User Planning stage involves four steps: determination of Federation requirements, review and selection of Gateway Performance Use Case(s), definition of required Architecture/SDEM pairs, and the hardware selection for the Gateway-Under-Test (Figure 2).

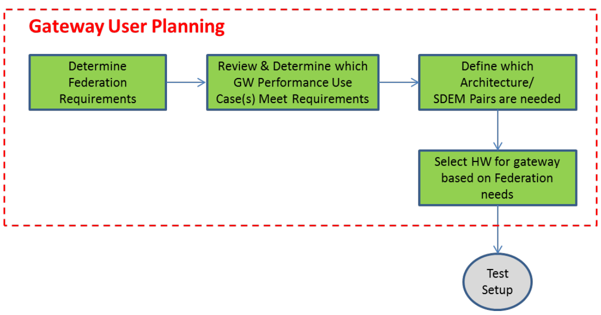

Figure 2. Steps for Gateway User Planning Stage

The first step is to determine Federation requirements for each side of the gateway, as derived from LVC Event requirements, selected Federation Agreements and any applicable simulation specification documentation. From these resources, specific scenario parameters (persistent and transient objects, nature of traffic, etc.) and operational parameters (number of architectures, simulated components, network configuration, etc.) are derived as a prerequisite for the second step, which is to review and select the Gateway Performance Use Case(s) most suitable for representing the anticipated architecture and performance requirements of the intended Federations, relative to the scale of the test event.

With the known Federation requirements and selected Gateway Performance Use Cases, the gateway user then selects a minimum of two architecture and SDEM pairings, one for each interface (side) of the Gateway-Under-Test. Architecture/SDEM pairs should be selected based on the Federation requirements for the distributed simulation event, and compliant with the selected Use Case(s). Finally, based on the selected Use Case(s) and the specifications for the selected Architecture/SDEM pairs, a hardware configuration similar to what is required or available for the distributed simulation event is selected to support the "Gateway-Under-Test" performance test. With the Federation and Architecture/SDEM requirements defined, Use Cases selected, and a gateway hardware configuration determined, the planning stage is concluded and the gateway user proceeds to the Test Setup stage of the methodology.

Gateway Vendor Planning

Gateway vendors have a vested interest in producing gateway applications that meet the performance requirements of Federation users in support of distributed simulation. Gateway vendors may leverage the performance test methodology to assess the performance of their gateway product within an operational environment that closely matches the common distributed simulation environments in use within the distributed simulation test community. Through benchmarking, vendors can exercise selected gateway applications against one or more pre-defined Use Cases for the purpose of establishing a set of performance metrics and/or thresholds that in turn benefits the gateway user community. The Gateway Vendor Planning stage for gateway performance benchmarking involves three steps: the review and selection of Gateway Performance Use Case(s), selection of required Architecture/SDEM pairs, and the hardware selection for the Gateway-Under-Test (Figure 3).

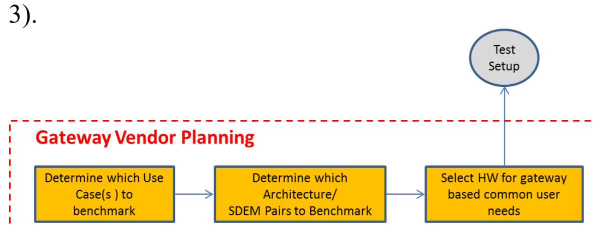

Figure 3. Steps for Gateway Vendor Planning Stage

Using defined scenario and operational parameters for the gateway to be tested, the first step is to review and select the Gateway Performance Use Case(s) most suitable for representing the anticipated architecture and performance requirements of the Federations the gateway is designed to support, relative to the scale of the test event. Use Case selection should be based on the need to ensure that benchmark performance testing meets the anticipated needs of the gateway user community, in terms of architecture and performance requirements.

With the anticipated Federation requirements (scenario and operational parameters) and selected Gateway Performance Use Cases, the gateway vendor then selects a minimum of two architecture and Simulation Data Exchange Models (SDEM) pairings, one for each interface (side) of the Gateway-Under-Test. Architecture/SDEM pairs should be selected based on applicable and anticipated Federation requirements against which the gateway is to be benchmarked, and compliant with the selected Use Case(s). Finally, based on the selected use case(s) and the specifications for the selected Architecture/SDEM pairs, a hardware configuration similar to what is anticipated for a distributed simulation event is selected to support the "Gateway-Under-Test" performance benchmark test. Vendors should further document the configuration parameters to be used for the gateway benchmark test and include them in the benchmark results package at the conclusion of testing. With the Architecture/SDEM requirements defined, Use Cases selected, and a gateway hardware configuration determined, the planning stage is concluded and the gateway vendor proceeds to the Test Setup stage of the methodology.

Test Setup

With the requirements from the planning stage determined, and a hardware configuration selected, the gateway to be performance tested or benchmarked can be implemented and integrated into the test environment. The purpose of the Test Setup stage is to complete the selection of components and configure the test environment accordingly. Test Setup includes the determination of switch latency, the selection of traffic generators, the selection or creation of scenario files, the selection of a data collector and the selection or creation of the analysis profile (Figure 4).

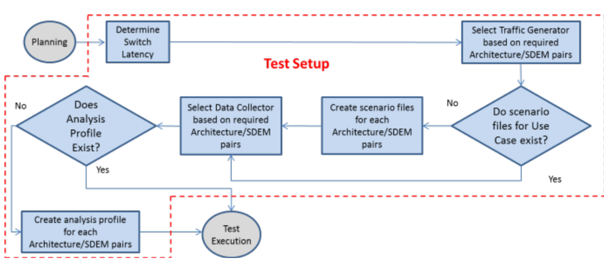

Figure 4. Steps for Test Setup Stage

As with any networked test environment, the characterization of the components and nodes through which test data passes is essential for the proper measurement of performance. On either side of the Gateway-Under-Test, switches are used to link the Federations to the gateway and the data collector. The latency of the selected switch can vary greatly depending on the make and model. These differences could impact performance measurements, and therefore the latency must be measured prior to the test. The one-way packet time through the switch is later used by the Data Analyzer in its performance calculations.

The next two steps involve the selection of traffic generators and the selection or creation of scenario files to be loaded into the selected traffic generators. Traffic generators are used to emulate simulations that conform to the specifications for a particular Architecture/SDEM pair. In addition, the selected traffic generators must support defined scenario requirements relative to data format, construct and semantics in order to exercise the gateway. These scenario files, loaded into the generators, produce the necessary traffic pattern(s) required for each Gateway Performance Use Case. If the scenario files for the selected generators do not exist, they will have to be created and then verified by loading them into the generator and ensuring they produce a traffic pattern compliant with the Use Case.

The fourth step is the selection of the data collector that shall be used to subscribe to and capture data from the test harness infrastructure. The selected data collector must support the Architecture/SDEM pairs used on each interface of the Gateway-Under-Test. The selected data collector should be similar to, or match, the standard data collection instrumentation as may be used during a distributed simulation to monitor gateway traffic.

The final step in the Test Setup stage is the selection or creation of an analysis profile for each Architecture/SDEM pair. The analysis profile defines the publishing rules for Architecture/SDEM pairs, rules by which collected test data is interpreted. This profile is later loaded into a data analyzer during the post-test Data Analysis stage, along with the collected test data. If the analysis profile for the Architecture/SDEM pairs does not exist, it will have to be created. The Test Setup stage is concluded when the full hardware configuration for the gateway performance test harness has been established, and when scenario files and an analysis profile have been selected or created.

Test Execution

The Test Execution stage begins with the final pre-test configuration of the gateway performance and benchmarking test harness (Figure 5). Scenario files are loaded into the traffic generators in accordance with the procedures defined for the selected traffic generator. The next step is to start any processes required by the Federation to support the full implementation of the distributed simulation architecture, where applicable, in accordance with any specific Federation architecture requirements outlined in respective agreements, and/or the specification details of the selected Architecture/SDEM pairs.

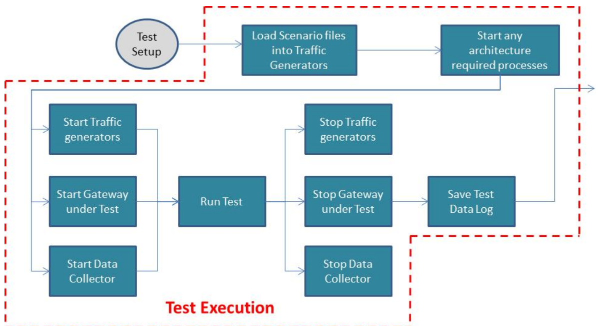

Figure 5. Steps for Test Execution Stage

With final pre-test configuration complete, execution of the gateway performance and benchmark test can begin. Test personnel start all elements in the following order: supporting Federation processes, data collector, gateway, and then traffic generator. Test personnel then verify that each fully joins the Federations to which they are attached. With the gateway performance test environment now operable in accordance with the selected Gateway Performance Use Case, the test run begins with the command to initiate publishing of data via the traffic generators. The test is run until completion of predetermined duration.

At the conclusion of the test run, personnel shall stop the traffic generators and verify that each generator ceases publishing and disengages from their respective Federations. Similarly, additional post-test shutdown procedures commence for the Gateway-Under-Test and the data collector. Test data collected by the data collector is saved to a designated repository in the form of data logs. With these steps, the Test Execution stage concludes and the focus shifts to Data Analysis.

Data Analysis

The Data Analysis stage of the gateway performance and benchmarking methodology takes as input the analysis profile produced or acquired during Test Setup and the saved test data logs from the Test Execution stage (Figure 6). The Data Analyzer is used to generate the necessary performance and benchmarking reports. Following procedures prescribed by the vendor and/or defined in the specification documentation for the Data Analyzer, analysts shall load the analysis profile and the test data logs into the analyzer and run the analysis.

The Data Analyzer shall calculate performance metrics for each side of the Gateway-Under-Test, in accordance with the selected Use Case. For gateway performance testing and benchmarking, the analyzer shall produce both a performance report and a Use Case verification report. The gateway performance report contains the results against defined metrics and may also contain information describing the scenario parameters and operational parameters that serve to characterize the Use Case and place performance results into context. The gateway Use Case verification report is used to verify the proper publication of the defined Use Case traffic pattern for the test and to support additional detailed analysis of the Gateway-Under-Test in the context of the selected performance Use Case. These reports are the end products of the Data Analysis stage, and of the overall methodology.

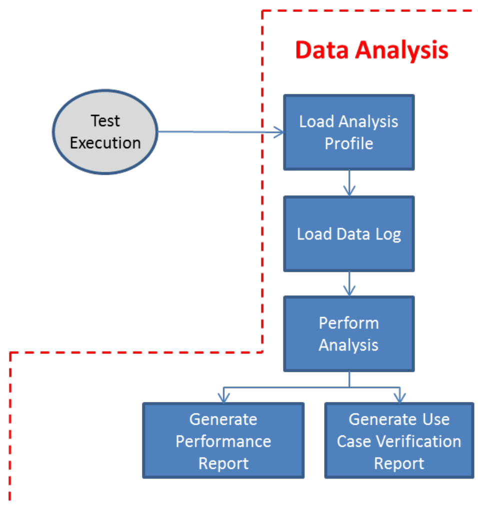

Figure 6. Steps for Data Analysis Stage

TEST HARNESS

In addition to a well-defined test methodology, a defined test harness is required to ensure consistency in performance metrics. A number of different approaches for the test harness were considered, including instrumenting the gateway. This would have required access to the internals of the gateway, which would eliminate the possibility of user-conducted gateway testing. This significant limitation forced a rejection of this approach and a “black-box” test approach was adopted. A “black box” test approach configures the test item as a standalone asset which requires all data injection and collection occur external to the test asset. For the purposes of this document, the test asset is a gateway. The benefits of this approach include running the gateways in a typical configuration, not adding additional load to the gateway and not requiring access to gateway internals. The test harness as defined here does not add any process load to the computer running the gateway. The only requirement for the Gateway-Under-Test is that it supports using separate network interfaces for each side of the gateway. This is a common feature and is found in most gateways.

The Test Harness has three components: Traffic Generators, Data Collector, and Data Analyzer. The configuration is shown in Figure 1. In addition to the Test Harness components, the test includes the Gateway-Under-Test, switches, and any required Federation processes. The switches are standard networking equipment. The two federations shall each use a separate switch. No other computers shall be connected to the switches for the test. Some distributed simulation architectures require standalone processes. If these are required, they shall be run on separate computers so they do not impact the processing of the other test harness components.

Traffic Generators

The Traffic Generators are used to simulate a federation. The traffic profiles are defined by the Use Case selected for the test. The Traffic Generators shall be able to support all distributed simulation architectures. This may be done with one Traffic Generator that supports all architectures, or with traffic generators specific to an architecture. Existing semi-automated forces (SAF) or other appropriate simulations may be used as Traffic Generators. The Traffic Generators must generate the traffic patterns specified in the Use Cases.

Data Collector

The Data Collector is responsible for recording the data that will be used to create the performance metrics for the gateway. Several design options were considered, including having two data collectors running on separate computers joined to the separate federations. To support tagging the object updates with accurate time stamps, this would have required connecting the separated data collectors to an external time source like that provided by the Inter-Range Instrumentation Group (IRIG) time codes. The external time source connections ensure that time is represented accurately and identically in each connected computer, thus producing synchronized time stamps for each data collector. This approach was rejected because of the cost of the IRIG time server. This would have added a significant cost to the test harness and made it more difficult for users and vendors to conduct the tests.

Instead, it was decided to host the Data Collector on a single computer that uses two Network Interface Cards (NIC). This allows the data collector to join both federations on separate networks. The Data Collector must be able to join both federations and subscribe to the required objects. It is critical that the Data Collector not drop any object updates as this will invalidate the performance metrics calculated for the gateway. Considerations as to how the recorded data will be stored are critical to the performance of the Data Collector. The Data Collector shall record all of the data in objects along with the time of arrival. This is required so that the Data Analyzer can determine if updates on the other side of the gateway were required.

Data Analyzer

The Data Analyzer is the final element in the Test Harness and is the most complex. The Data Analyzer has to calculate all of the required metrics based on the file produced by the Data Collector. The Data Analyzer also has to determine if the measured traffic flow met the requirements of the Use Case. Because different architectures have different rules for updates, the Data Analyzer must understand these rules to determine if an update should occur. This information on the architecture's publication rules is stored in profiles for each Architecture/SDEM pair.

BENEFITS

The fundamental user need that has driven the LVCAR gateway testing activities is the ability to directly compare the performance of different gateways. That is, as the developers of LVC training or test environments identify their functional requirements for supporting gateways and map those requirements against the capabilities that existing gateway products can provide, there also needs to be a way to map performance requirements to the performance achievable by competing gateway products. This latter mapping requires a common set of metrics and well-defined use cases for comparing performance, which are currently provided by the LVCAR Gateway Performance Benchmarks (JHU, 2011). However, side-by-side comparisons of relative gateway performance across different vendors depend on the consistency of the test environments used to produce the benchmark data. The common testing methodology and test harness design discussed in this paper are designed to provide that consistency.

The benefits of this collective set of gateway performance testing products are substantial. To the gateway developer, these supporting test products eliminate the need to identify the appropriate set of gateway performance measures themselves or to develop test methodologies to produce them. It also provides a viable design for a test environment that can “host” the common testing methodology used to produce the benchmarks. The use of these supporting test products thus provides a viable baseline from which an internal test capability can be established (particularly for vendors that have not performed such testing in the past). As testing is performed, the performance benchmark data is included in the LVCAR Gateway Description Language (GDL) and made available to the LVC community though the Enterprise Metacard Builder Resource (EMBR) Portal (JHU, 2010). As users employ the GDL Repository feature of the EMBR Portal to discover gateways that meet their requirements, vendors that produce gateways with superior performance characteristics will have competitive advantage over vendors of lesser products, and can potentially increase market share.

The benefits of these supporting test products for gateway users are even more significant. While all users generally want high-performance gateway products, there are some users that can tolerate only very small amounts of latency in their LVC environments, in which case performance is an especially critical factor during gateway selection. Currently, the availability of gateway performance data is very limited, and when vendors do provide this type of data, inconsistent performance metrics are common. This has resulted in some uninformed (and unfortunate) gateway selections in the past, many requiring expensive corrective actions. The availability of common performance benchmarks in the GDL descriptions of individual gateways allows for side-by-side comparisons of gateways based on performance for the first time. However, if vendors were given the freedom to produce the benchmarks for their gateways without any special constraints, differences in perceived performance could be as easily driven by the unique characteristics of the test environment as by the performance of the gateway itself. The testing methodology and test harness design described in this paper ensures that gateway testing is performed on a level playing field across different vendors, thus allowing users to apply the benchmarks uniformly and consistently across competing gateway products. The effect of this will be better, more informed gateway selections early in LVC developments, reducing the rework and costs associated with less informed decisions.

Users will accrue some additional indirect benefits as well. With gateway performance data more readily available (via the EMBR Portal), vendor competition is likely to generally improve performance throughout the gateway marketplace. Also, the performance benchmarks provide gateway users a consistent vocabulary that enables them to discuss performance with gateway developers and clearly communicate requirements for improvement.

NEXT STEPS

The gateway testing products described in this paper represent a significant step toward achieving LVCAR-I gateway selection and employment objectives. While these products were produced in direct support of user requirements expressed at LVCAR-I workshops and other events, it is recognized that the content and scope of these products will be refined over time based on feedback from “early adopter” users. To facilitate obtaining this feedback, an effort is underway at the Joint and Coalition Warfighting (JCW) facility in Suffolk, VA to produce a prototype implementation of the gateway test harness design and to exercise the test methodology on one or more gateways currently in wide use across the LVC community. This initial implementation will provide the proof of concept necessary to engender early user confidence in the gateway test products.

Longer term, to achieve the degree of market penetration needed to fully realize LVCAR-I cost containment and technical/schedule risk mitigation goals for distributed simulation environments, mechanisms to achieve “buy-in” from government and commercial gateway developers and to facilitate user involvement in the maturation of gateway test products are critical. In CY13, a series of structured workshops are planned to expedite outreach to these communities and directly involve users and developers in the evolution of the gateway test products. This input will ensure that these products remain supportive of community needs in both current and future LVC events.

SUMMARY

This paper has described an effort under the OSD-sponsored LVCAR-I project to produce supporting products for gateway performance benchmark testing. Through a common set of performance benchmarks, a common set of use cases, a common gateway test methodology, and a common design for gateway test harnesses, developers can now generate performance data that is consistent and comparable across different gateway implementations. This consistency allows gateway users to make better, more informed choices for supporting gateways, thus reducing the cost, schedule, and technical risks associated with the application of gateways in LVC environments.

Access to the test products described in this paper will be provided in the future via the Modeling and Simulation Coordination Office website (www.msco.mil). Near-term requests for these products should be directed to the authors of this paper.

REFERENCES

Live-Virtual-Constructive Architecture Roadmap Implementation, Common Gateways and Bridges Execution Plan, JHU/APL NSAD-R-2010-049, June 2010.

Live-Virtual-Constructive Architecture Roadmap Implementation Common Gateways and Bridges, Gateway Performance Benchmarks, JHU/APL NSAD-R-11-072, November 2011.

Live, Virtual, Constructive Architecture Roadmap (LVCAR) Final Report, Institute for Defense Analyses (IDA), September 2008.

Lessmann K., Cutts D., O'Connor M., "LVCAR Enhancements for Selecting Gateways", 2011 Spring Simulation Interoperability Workshop (SIW), 11S-SIW-054, April 2011.

O'Connor M., Cutts D., Lessmann K., "LVCAR Enhancements for Using Gateways", 2011 Spring Simulation Interoperability Workshop (SIW), 11S-SIW-025, April 2011.