07S-SIW-076

Distributed Simulation Run Automation Capability for the Navy P_{RA} Testbed

Sarah Trbovich
Richard Reading
William Photinos III
VisiTech, Ltd.
99 Canal Center Plaza, Suite 420
Alexandria, VA 22314
703-535-6640
trbovich@visitech.com, reading@visitech.com, photinos@visitech.com

Shala Malone
PEO Integrated Warfare Systems 7D
1333 Issac Hull Avenue SW
Washington Navy Yard, DC 20376
202-781-2133
shala.malone@navy.mil

Keywords:
Navy Probability of Raid Annihilation (P_{RA}) Testbed, Run Automation, OneButtonStart, Operational Test & Evaluation (OT&E), Ship Self Defense (SSD)

ABSTRACT: *The Navy Probability of Raid Annihilation (P_{RA}) Testbed implements HLA federated simulations of ship combat system elements against independent, reactive threat raids in a common environment to formulate an overall combat system assessment. The P_{RA} Testbed is a cornerstone of the U.S. Navy Ship Self Defense Test & Evaluation (T&E) Enterprise. The LPD 17 ship class is the first to implement a P_{RA} Testbed baseline as a formal component of ship class OT&E. Products and lessons from the LPD 17 baseline are being transitioned to multiple ship classes including DDG 1000, LHA 6, LCS, and CVN 21.*

The P_{RA} Testbed LPD 17 baseline is deployed as a secure, geographically distributed, time managed federation with three nodes: the U.S. Naval Research Laboratory (NRL) Washington, DC; Johns Hopkins University (JHU) Applied Physics Laboratory (APL) Laurel, MD; and Naval Air Warfare Center (NAWC) Weapons Division China Lake. Federation execution is controlled centrally from the NRL node.

The P_{RA} Testbed LPD 17 baseline federation is required to conduct nearly 2000 simulation test events during the course of LPD 17 P_{RA} Assessment. Each scenario execution involves computation intensive calculations that progress much slower than real time. The development team is using Run Automation software, dubbed "OneButtonStart" to automate batch runs of the federation from a single control node. OneButtonStart is an important part of the LPD 17 testing program because it dramatically reduces manning requirements and takes advantage of all available computer time to minimize schedule. Yet, there are numerous challenges to implementing automated control of a classified, non-realtime federation that includes both legacy and new software.

In this paper, we describe the Run Automation capability and how it will be used in the P_{RA} Testbed LPD 17 baseline. We also share lessons learned in automating runs for a secure, geographically distributed, time-managed simulation with multiple nodes.

1. Why use a Run Automation Capability?

The Navy Probability of Raid Annihilation (P_{RA}) Testbed implements HLA federated simulations of ship combat system elements against independent, reactive anti-ship cruise missile threat raids in a common environment to formulate an overall combat system assessment. The U.S. Navy Ship Self Defense Test & Evaluation (T&E) Enterprise has codified the P_{RA} Testbed as a formal component of ship class Operational Test & Evaluation (OT&E). Distributed simulation is being used to satisfy OT&E requirements that are impractical or unsafe to

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **2007**	2. REPORT TYPE	3. DATES COVERED **00-00-2007 to 00-00-2007**

4. TITLE AND SUBTITLE **Distributed Simulation Run Automation Capability for the Navy Pra Testbed**	5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **PEO Integrated Warfare Systems,1333 Isaac Hull Avenue SW,Washington Navy Yard,DC,20376**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**		**6**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

address with live T&E. The P_{RA} Testbed has been instantiated in an HLA federation architecture which is:

- Time managed;
- Reliant on computation intensive physics-based models;
- Geographically distributed;
- Security classified at a minimum Secret level.

Accreditation for OT&E is a tall order for any simulation in a resource constrained environment. However, there are also practical concerns with achieving a successful simulation-based OT&E event with the P_{RA} Testbed. Due to the complexity of the models, the results for each test scenario require hours to compute. However, to assess system-of-systems performance, a span of thousands of scenarios is required. To meet the OT&E schedule, P_{RA} Testbed execution depends on maximizing use of CPU time 24 hours a day. Human intervention to initialize and configure the Testbed on a per-run basis is impractical; hence, the need for a run automation capability.

1.1 Federation Requirements

The Navy P_{RA} Testbed is a distributed, classified, non-realtime federation that includes both legacy and new software. The run automation capability responds to simulation requirements driven by an amalgam of technical, cost, and schedule factors.

P_{RA} is a platform-level metric that is levied on the complete ship self defense system-of-systems. Consequently, the simulation requirements encompass numerous systems and system interactions, including hardkill, softkill, and ship signature control. Detail must be sufficient to pass muster with the Navy's Operational Test and Evaluation Force (OPTEVFOR) for accredited use in OT&E. Current computing capability will only support much slower-than-real time execution for many of the participating federates in the P_{RA} Testbed. Additional runtime performance constraints are levied by the secure wide area network, those these are far outweighed by the computation time necessary for the complex mathematical algorithms within each federate.

Each federation run provides the results for a single ship defense engagement. Currently, each engagement can take up to two hours to complete. Adding runs increases the overall time to conduct the runs by days and weeks rather than minutes and hours. To complete a P_{RA} assessment over the required span of thousands of runs in a reasonable amount of calendar time, it is necessary to conduct runs 24 hours a day, seven days a week. As such, each participating federate is required to support operation without human intervention and accept commands from a single control node.

Yet, the federation is also required to run at a classified security level and geographically distributed across the United States. Meeting those requirements while also supporting automated 24/7 execution is challenging. It is difficult to run a classified federation; adding the complexity of multiple distributed nodes in two different time zones makes implementing the simulation even harder. The working hours of the facilities must to be correlated during testing to take advantage of all model experts for analysis. When one facility is down, testing ceases until all models can participate. Establishing graceful degradation and recovery processes become more complicated when relying on automated control. Further, identification of key or anomalous events, with additional data collection, becomes more difficult without human observers present. Also, access and control of individual simulation federates through a secure network presents its own challenge.

Lastly, a major hurdle to clear is successfully mixing legacy and new software in the federation. While some functionalities have been around for years, there are those which have not been modeled until now. The earlier models were created before HLA frameworks were standardized for government usage, and if they have been converted to an HLA framework it was for a specific application and they need to be tailored to the Testbed architecture. The Testbed also utilizes some re-hosted tactical software. The tactical software components have particular expectations for operator interaction, e.g., for system configuration such as mode settings; these must be addressed to provide an automated run capability for the P_{RA} Testbed.

2. First Instantiation of the Navy P_{RA} Testbed for OT&E

The LPD 17 amphibious ship class is the first to implement a P_{RA} Testbed baseline as a formal component of ship class OT&E. Products and lessons from the LPD 17 baseline are being transitioned to multiple ship classes including DDG 1000, LHA 6, LCS and CVN 21.

The P_{RA} Testbed LPD 17 baseline is deployed as a secure, geographically distributed, time managed federation with three nodes: the U.S. Naval Research Laboratory (NRL) Washington, DC; Johns Hopkins University (JHU) Applied Physics Laboratory (APL) Laurel, MD; and Naval Air Warfare Center (NAWC) Weapons Division China Lake. Federation execution is controlled centrally from the NRL node.

The P_{RA} Testbed implements a "Virtual Range" capability including a common environment and a defined threat set. The LPD 17 baseline includes 20 different Scenario

Natural Environment (SNE) conditions; the SNE conditions are identified by combinations of two locations (open ocean, littoral), two seasons (Summer, Winter) and five times of day. Weather is limited to a moderate sea state and no precipitation. Five different anti-ship cruise missile types are included in the virtual range for the LPD 17 ship class assessment. A total of 2000 cases are required for LPD 17 assessment.

3. Design of the Run Automation Capability

For the P_{RA} Testbed LPD 17 baseline, the Run Automation capability was defined by the testing constraints on the simulation. There were three key attributes the run automation capability must possess to fully automate the simulation runs at testing:

1. ability to start-up and control the status at each federate's state as it transitioned through the launch, initialization, execution and termination of each run;
2. ability to transition all federates between the termination of the previous run to the launch of a new run with changed variables; and
3. ability to monitor key events and terminate 'bad' runs when the federation fails to function properly.

3.1 State Transition during a Simulation Run

The LPD 17 P_{RA} Testbed consists of three nodes (NRL, APL and China Lake) with the federation management controlled at NRL. The run automation capability will be run with the federation management software at NRL to control status of the APL and China Lake federates. To begin a simulation run, all federates are to be launched without human intervention. After all federates have completed start-up, they are initialized with a dedicated scenario including specific information on the radials relative to the ship, ship signature, threat type and SNE information. The simulation run will commence and continue through the execution state. Once the simulation run has proceeded through all key events the run is terminated and all federates are brought down. Figure 1 depicts the four states of the simulation run requiring transition management of all federates by the run automation capability.

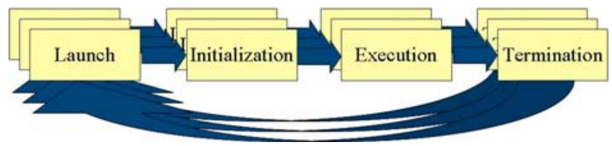

Figure 1. Simulation Run States and Run Automation Process

3.2 Automated Transition from Run-to-Run

Also depicted in Figure 1, is the key attribute of the run automation capability to transition all federates between termination of the previous run to the launch of a new run with changed variables. To perform 2000 simulations runs, the variables must change slightly each run to test each separate case. For the LPD 17 P_{RA} Testbed, the order of simulation runs is listed in Table 1 below. The order will complete all eight radials before changing the ship signature and so on.

VARIABLES	NUMBER
Radials (Relative to Ship)	8
Ship Signature	2
Threat Number	5
Environment Times of Day	5
Environment Seasons	2
Environment Location	2

Table 1. Order of Simulation Runs

3.3 Early Termination of a Run

Time is vital when managing the simulation runs; therefore, from a technical vantage point if the federation is not functioning properly it is important not to waste time finishing the simulation run. The run automation capability is required to monitor the federation and terminate 'bad' runs when the federation fails to function properly. The term 'bad' run has been described as if a federate failed to load properly, a federate drops early from the federation or the correct scenario is not being run by all federates. However, it is important not to terminate 'interesting' runs that contribute to the analysis required to support OT&E requirements. There is a fine line between a 'bad' run and an 'interesting' run, but for the LPD 17 P_{RA} Testbed it has been described as failure to technically function rather than failure in system performance.

3.4 Key Events

Another mechanism used both to boost efficiency and improve results analysis is monitoring of key scenario events. The run automation capability tracks these events to identify opportunities for early run transitions. If a certain scenario event occurs during a run (e.g., the radars cannot detect the threats before a certain range, the threats are defeated by softkill prior to hardkill), then it may not be necessary to complete the rest of that run. Key events are also used to determine when a run should be terminated. (e.g., the last threat is defeated by hardkill, the ship is hit by a threat). A set of these key scenario events was determined federation-wide. The federates can use these "Key Events" to determine the state of the simulation run. All federates can also use these events to assist in collecting detailed data for after action review. The Key Events are used by the run-automation capability to make run transition more efficient. Since there may be no human in the loop during run-automation trials tracking these events will allow the run automation to transition between runs more efficiently allowing for more simulation runs in a given time period.

4. Implementation

For the LPD 17 P_{RA} Testbed, the run automation capability is located in the Scenario Environment Federate (SEF). The SEF consists of three software applications: OneButtonStart, Federation Control and Run Automation. The incremental software development for this capability, shown in Figure 2, was progressed as the separate applications were implemented and tested. Building on the basic Federation Control functionality was the OneButtonStart application which automates the start-up and termination of the federation. Once the OneButtonStart functionality was performing, the Run Automation application could be fully integrated which allows the transition between simulation runs.

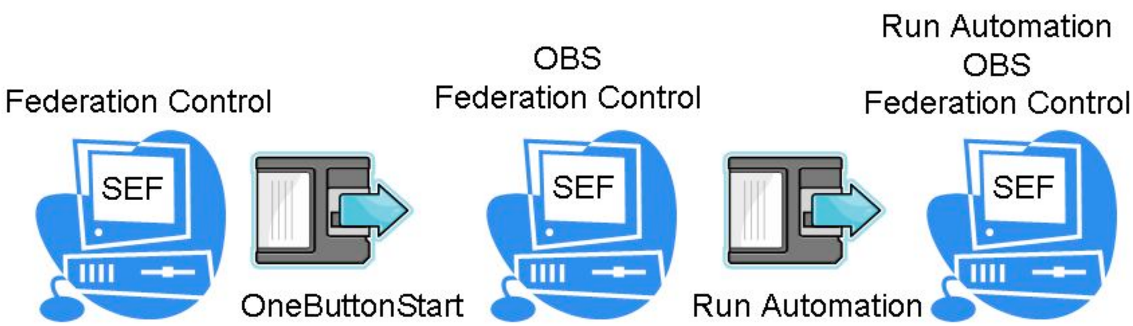

Figure 2. Run Automation Capability Software Buildup

4.1 OneButtonStart and Federation Control

Basic Federation Control functionality includes initialization, identification, synchronization, monitoring and termination of the federation. To implement a run automation capability, these Federation Control functions required cohesive management with the automated federation start-up and termination software, OneButtonStart. Both of these functionalities are located in the SEF federate; however, to participate in the automated management each federate must hold an application of the OBS software locally. To activate the OBS software, the federation manager at the control node opens the OBS application which in turn connects to all machines with an OBS application loaded. In this instance, the OBS application is launching one simulation run for the federation.

Located below in Figure 3 is the process followed by SEF to manage the OBS and Federation Control functions. Note that SEF is listening and sending specific objects and interactions to start up, manage, and terminate the federation. Note in Figure 4, the SEF Graphic User Interface (GUI), the Federation Status Table which monitors current status of all federates at all times during a simulation run.

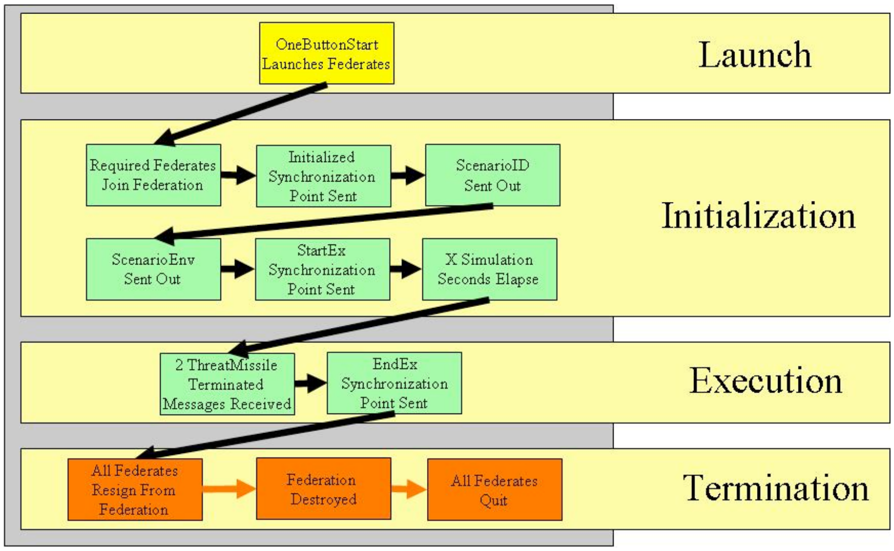

Figure 3. Detailed OneButtonStart/Federation Control States

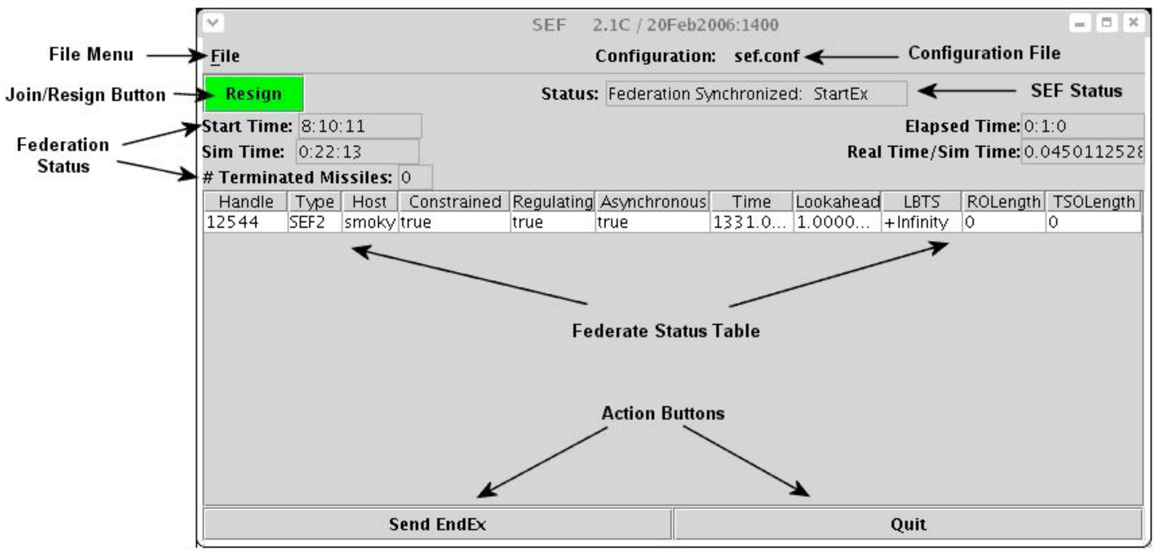

Figure 4. SEF GUI

4.2 Run Automation

While OBS and the Federation Control can automate the start-up, initialization, execution and termination of the federation, the Run Automation application allows the federation to transition between simulation runs. Instead of loading the OBS application with one simulation run, a batch file of simulation run parameters is loaded. The Run Automation application then monitors the OBS as it transitioned to the next simulation run.

The Run Automation application can be made as intelligent as the federation requires. In the LPD 17 P_{RA} Testbed, the Run Automation application is being used to maximize the CPU usage for all 24 hours in a day without the need for a human in the loop. Therefore, the run automation application monitors the Federation Control application and terminates 'bad' runs when the federation fails to function properly. The Run Automation application will reload the same simulation run to test if the error occurs again, but will then move on to the next simulation run. After each simulation run, the configuration file and simulation run results are stored in separate files.

5. Way Forward

With the P_{RA} Testbed LPD 17 baseline runs-for-score imminent in the second half of 2007, there are many lessons to be learned in the near future. In the time leading up to the Final Build of the LPD 17 federation, the Run Automation capability will continue to mature in its ability to intelligently manage the simulation runs.

This capability has because an integral part of the federation management as it dramatically reduces manning requirements and takes advantage of all available computer resources.

Though computing power continues to increase, so does the complexity of the ship self defense system-of-systems. Computation intensity required to meet OPTEVFOR requirements for OT&E accreditation will not wane. As such, runtime efficiency of the P_{RA} Testbed will remain a challenge and the Run Automation capability will remain a key part of meeting the challenge.

6. References

[1] "Test and Evaluation Master Plan No. 1714, Capstone Enterprise Air Warfare Ship Self Defense", Program Executive Office Integrated Warfare Systems, draft, 25 May 2006.

Author Biographies

SARAH TRBOVICH is an Engineer with VisiTech, Ltd. She is currently providing technical leadership and support to HLA federation developments for U.S. Navy ship combat system testing, as well as other multi-national efforts applied to systems acquisition. She is the Secretary of the NATO Sub-Group 61 on Virtual Ships. She is a lead author of the NATO Virtual Ships simulation standard. Ms. Trbovich has also provided systems engineering expertise to the U.S. Navy, NAVSEA 06, in support of Single Integrated Air Picture implementation.

SHALA MALONE is the Combat Systems Performance Manager for the U.S. Navy's Program Executive Office, Integrated Warfare Systems. She is responsible for all T&E Enterprise P_{RA} Testbed baselines. She manages the P_{RA} Testbed architecture and development of the P_{RA} Testbed virtual range components. Ms. Malone chairs the P_{RA} Testbed Configuration and Threat Modeling and Simulation working groups within the Ship Self Defense T&E Enterprise. Ms. Malone is also the M&S Director for PEO IWS.

WILLIAM PHOTINOS III is a Senior Software Engineer/Analyst with VisiTech, Ltd. He is one of the lead engineers working on Synthetic Natural Environment, Federation Control (SEF), and Run-Automation work for the LPD 17 P_{RA} effort. He has been a software/systems engineer for numerous simulation efforts including Navy Fleet Battle Experiments, Foxtrot, Golf, and Hotel. He functioned as a Systems Engineer for USJFCOM sponsored distributed simulation events, J9901 and UV01 and as a Simulation Architect/Engineer for MC02 events. Mr. Photinos was a lead software engineer engaged in the modification of JSAF for a prototype Marine Corps training system.

RICHARD READING is the Systems Engineering Division Lead with VisiTech, Ltd. He is providing technical leadership and support to HLA federation developments for U.S. Navy ship combat system testing, as well as other multi-national efforts applied to systems acquisition. He is the chief scientist leading the simulation development portion of the LPD 17 amphibious ship class combat system operational testing. He is the U.S. industry representative to the NATO Sub-Group 61 on Virtual Ships. He has also served as the International Project Team Leader for the NATO Simulation Interoperability and Re-Use Study (NIREUS).