Geospatial Correlation Testing Framework and Toolset

Freddie Santiago, Marlo Verdesca, Jon Watkins
Dignitas Technologies, LLC
Orlando, Florida
fsanatiago@dignitastech.com,
mverdesca@dignitastech.com, jwatkins@dignitastech.com

Julio de la Cruz
US Army Research Laboratory
Simulation and Training Technology Center
Orlando, Florida
julio.delacruz@us.army.mil

ABSTRACT

The Modeling and Simulation industry has long been plagued by geospatial database representational flaws and miscorrelation used to represent the synthetic natural environment within military training systems. These errors spawn from a wide range of sources, including design decisions, performance simplifications, bad source data, and unrealistic or erroneous database content. These errors are so pervasive across systems that they are often accepted as inevitable despite Soldier training impacts.

This paper discusses the work conducted under a Phase II SBIR. It provides proposed solutions consisting of toolsets that assess and compare geospatial and geometric data between disparate database formats and representations while providing multiple testing mechanisms such as visual inspection, automated testing and interactive testing using reusable software libraries and analysis artifacts. Real world examples of specific database errors on Army simulation programs will illustrate the complexity of tying geometric flaws with training impact.

This paper examines the challenges, planned approaches, and solutions for both detection and evaluation of correlation and representation errors. The work includes implementation of a testing framework and open standards for test tools and test data exchange, as well as instantiation of that framework in the C-nergy toolset. Moreover, technical transfer of this research by leveraging emerging common Army standards, such as SE Core and One Semi Automated Forces (OneSAF), is critical to successful widespread use of correlation testing toolsets.

ABOUT THE AUTHORS

Freddie Santiago is the Technical Lead for the Correlation research project. He has worked development on the Environmental Runtime Component for OneSAF, which includes the development of SE Core's Dynamic Terrain functionality, support for integration of OneSAF into CCTT, and development of database test tools and processes. Mr. Santiago is a Software Engineer at Dignitas Technologies and holds a Master of Science degree in Computer Science from the University of Central Florida.

Marlo Verdesca is Program Manager for the Correlation research project. She has managed and led numerous technical development efforts on complex real time and simulation systems such as DARPA's Deep Green, SE Core, OneSAF, OneSAF Testbed Baseline and its predecessor Modular Semi-Automated Forces (ModSAF). Ms. Verdesca is the Director of Programs at Dignitas Technologies and holds a Bachelor of Science in Management Information Systems at the University of Central Florida.

Jon Watkins is COO of Dignitas Technologies. He has an extensive background in Army modeling and simulation applications, with a particular specialization if Synthetic Natural Environment representations in constructive and virtual simulations, including format specifications and run-time representations used in many major simulations today. Mr. Watkins holds a Bachelor of Science in Computer Engineering from the University of Central Florida.

Mr. Julio de la Cruz is the Science and Technology Manager and Chief Engineer for Synthetic Natural Environment (SNE) and Simulation Technologies. Mr. de la Cruz has over 20 years' experience in government and private sector. As the SNE Chief Engineer for the Science and Technology at the Army Research Laboratory, Human Research and Engineering Directorate, Simulation and Training Technology Center, he oversees the research and development of emerging Database Generation tools and technologies that supports the Command and the Army current and future needs in Modeling and Simulation.

Geospatial Correlation Testing Framework and Toolset

Freddie Santiago, Marlo Verdesca, Jon Watkins
Dignitas Technologies, LLC
Orlando, Florida
fsanatiago@dignitastech.com,
mverdesca@dignitastech.com, jwatkins@dignitastech.com

Julio de la Cruz
Simulation and Training Technology Center
US Army Research Laboratory
Orlando, Florida
julio.delacruz@us.army.mil

INTRODUCTION

Imagine you are the gunner of a state-of-the-art M1 Abrams tank. You have been tracking the movements of an enemy vehicle when, through the radio chatter, you hear your commander give the go ahead to eliminate the threat. You scan your range finder and determine that the target is still within range but only for another few seconds as it is quickly moving away from your position. As you get ready to pull the trigger, hands trembling, sweat running down your face … the enemy vehicle begins to float off the ground… what?! Not again! You immediately realize once again that you are in the middle of a simulation and that the computer generated enemy tank has run into a database error. This type of error is often caused by miscorrelation in terrain elevation between virtual databases. Such cases can be frustrating to soldiers and commanders as it distracts from the training at hand and the focus of the training scenario.

The Modeling and Simulation (M&S) industry has long been plagued by geospatial database representational flaws and miscorrelation in the synthetic natural environment within military training systems. These errors spawn from a wide range of sources, including design decisions, performance simplifications, bad source data, and unrealistic or erroneous database content. These errors are so pervasive across systems that they are often accepted as inevitable despite soldier training impacts. This situation is also complicated by the sparsity of database quality checking and correlation testing tools. In addition, many of the testing tools that do exist are either very difficult to use or require intimate knowledge of the simulation system to be able to interpret the test results.

This paper will present concepts on testing terrain databases for military training use cases. Also discussed are some reasons why terrain database testing is important to training, the reasons why correlation and quality testing are difficult, and some ways to mitigate those difficulties. Lastly, an extensible framework developed to provide a faster way to test databases and share test results among peers will be presented. Through research and technology some of the difficulties related to the testing of production databases can be diminished. Where applicable real-world examples from major Army M&S programs will be used to provide context.

KEY CONCEPTS

Throughout this paper, the focus will be on examples and solutions for a few key principles, or lessons-learned which we propose to resolve or mitigate.

There have been many attempts in literature, requirements specifications, and development efforts to come up with a rigorous, quantitative definition of correlation [Schiavone, 1994]. Emphasis is on the training effect of correlation in end system use, without attempting to focus on an overarching or all-encompassing definition. A showstopper error in one system or use case may be irrelevant in another. This requires flexible toolsets with highly parametric tests as well as human evaluation of results.

Many toolsets focus largely or exclusively on database content, with a particular emphasis on attribution or geometry [Palmer, 2011]. While this is a critical component, it is most important to focus on the functional result of the data instead. As a simple example, rather than focus on the direct contents of One Semi Automated Forces (OneSAF) Terrain Format (OTF) itself [Stevens, 2001], we consider what the software wrapper around OTF, Environment Run-time Component (ERC), provides. ERC services are directly relevant to simulation systems operating on OTF since they all use ERC, not OTF directly [Dukstein, 2007]. This distinction can be important because elements of geometry are reconstructed at run-time in many cases, including ERC (e.g. in the case of converting a point feature building into a geometry for line of sight intersections).

Test tool developers will often concentrate primarily on automation, wherein tests are run automatically with an attempt to draw key conclusions without human intervention. While this is a laudable goal, human interaction is critical since there are too many variables to the question of what matters in a training system context. As a result, the focus is on a mixture of automation to find possible concerns combined with interactive (human-in-the-loop) testing to help interpret the results. Automation is primarily used to help focus the interactive test efforts.

A critical component for a successful testing capability is software rights and openness. Too many correlation toolsets have been proprietary (and thus not extensible by the community in general), or very highly specialized to a single program or context. Government purpose rights software development and an extensible framework is an important aspect of this project. This will help mitigate another common downfall of database testing toolsets in the past, namely irrelevance as simulation standards and formats shift. Sizable investments in toolsets focused around Synthetic Environment Data Representation and Interchange Specification (SEDRIS) [Mamaghani, 2004] or individual programs (such as highly specialized testers implemented in Ada for the Close Combat Tactical Trainer (CCTT)) have become hard to leverage for today's M&S community in general. In the end, our test utilities focus on specific formats which themselves can become obsolete, but the openness of our implementation will help mitigate this risk by allowing others to reuse the approach in their specialized use cases.

Finally, most database test utilities in active use are very difficult to execute, and the results generated are challenging to interpret or evaluate. While such low-level test utilities with voluminous output have a place, the goal is to simplify execution of tests and evaluation of their results so that a wider range of simulation development personnel can be involved in terrain database test and evaluation. As an example of this concern, during OneSAF integration into the Aviation Combined Arms Tactical Trainer (AVCATT) and CCTT, Synthetic Environment Core (SE Core) developers created an OTF tester called the Automated Test Driver. This application scanned an OTF database looking for the key indicators of problems that had plagued SE Core's effort to convert legacy (CCTT/AVCATT) databases into OTF from various inputs including SEDRIS. The tester was extremely valuable in identifying problems, but only a handful of developers with very deep knowledge of ERC/OTF, and their virtual application could interpret the results.

IMPORTANCE TO TRAINING DOMAINS

From the M&S standpoint, terrain database quality and correlation has a direct effect on the quality of training for Soldiers in many dimensions, and across virtual, constructive, live, and operational use cases.

Across all domains, terrain database errors often result in large sections of databases being declared unusable by training facilitators. Therefore, the development investment in creating large area databases can be rendered useless if database quality issues discourage the use of large swathes of the simulation area.

Virtual Training

One of the goals of virtual simulation is to immerse the trainee in a realistic world environment in order to help achieve a "train as you fight" effect. Any anomalies caused by terrain database problems, such as vehicles appearing to float above the visual ground, take away from this immersion and distract the trainee from the task at hand. These types of issues can also suspend the trainee's sense of urgency or belief in a simulation context. Directly related to this concern is the concept of "fair fight." Fair fight requires that no entity in a simulation, whether human-controlled or computer generated, should have an unfair advantage over others [Marshall, 1996]. The value of training sessions is diminished if the trainee is able to perceive that he can exploit existing database anomalies to his advantage.

An amusing anecdote from the Simulation Network (SIMNET) helps to illustrate why pure geometry tests are insufficient for testing purposes. SIMNET operators had learned very specific coordinates to place manned simulators such that they would "balance" on a power pole. In this case, the geometry and content of the database was valid, but the runtime services for elevation calculations and entity placement would allow a vehicle to be suspended at the top of a power pole. Here, run-time services were more important than pure geometry.

A more tangible example from a virtual program is AVCATT Compact Terrain Database (CTDB) databases. During OneSAF integration [Hughley, 2006], OTF data was compared to CTDB data to ensure correlation. One test result pointed to a massive difference in building heights between the representations, a major concern for an air simulation. However, it turns out that AVCATT's use of CTDB was ignoring the building height attribution entirely, so there was no functional effect from the legacy CTDB's large building height values. Again, this shows that functional and visual effect are critical when trying to understand correlation in a training context.

Our work to date has been primarily focused on virtual simulation, particularly the use case of OneSAF integration into CCTT and AVCATT via the SE Core program. Thus, the formats addressed for proof of principle were visual databases and OTF/ERC, which provides a common terrain representation for OneSAF and CCTT/AVCATT manned simulators.

Constructive Training

Database quality and correlation is often strongly associated with virtual simulations, wherein anomalies can be seen between the visual and supporting geometry or reasoning databases. However, the same phenomenon can be seen in constructive simulation as well [Prager, 2002]. For example, if vehicles on a road march will not cross a bridge as a result of a topology error in the underlying reasoning databases, this can impact the effectiveness of a constructive training simulation.

Live Training and Operational

Both live training and operational uses for geospatial data can require very strong correlation with the real world in order to be fully effective. The concept of geopairing in live simulation is intended to eliminate the "tricks" that can be played with emitters and sensors for hit/miss detection, such as blockage from smoke. Geo-pairing would use a very high resolution geometric representation of the world to determine when hit/miss occurs based upon ray tracing of simulated rounds. Similarly, mobile and embedded apps attempting to provide terrain reasoning [Borkman, 2010] require close correlation to the real world to be effective. In this context, toolsets are needed to compare geometry databases used in applications to high resolution source data representative of the real world.

Interoperability

An additional source of correlation requirements arises from the need for interoperability. As military training has evolved from individual training to full-scale combined arms training it is only natural to require that increasing numbers of systems be interconnected into one simulation, for example, the Live, Virtual, Constructive-Integrating Architecture (LVC-IA) program [Bizub, 2007]. Often the requirement to interoperate faces the challenge of having to handle multiple generations of databases and technology. For example, when the Army introduced the concept of OneSAF to provide greater commonality across simulation systems, there was a need to temporarily interoperate with the legacy databases (such as CTDB) as well as convert the old databases to the new OneSAF OTF format. Both of these cases required that some method of correlation comparison be implemented to verify that all systems were in sync. Unfortunately correlation testing mechanisms have been largely non-existent or highly specialized to a particular context, and programs have had to adopt a "reactive" approach to database problems. Today, database testing often consists of manual, sampled database reviews supplemented by very specialized toolsets in limited cases. This type of testing is even more time consuming and limited when interoperating between multiple complex training systems.

TYPES OF TEST

Two classes of database testing were focused on, which will be referred to as *Quality Testing* and *Correlation Testing*. Each of these types of tests are complicated by the need to assess test parameters and results in a context that targets their effect on simulation and training.

Quality Tests

Quality tests are generally conducted in the context of a single database format and focus on the nature and relationships of data within that database. For example, quality testing can verify that content meets the format spec (are attributes in range?) and that database content will provide the desired effect within the using simulations' functionality context.

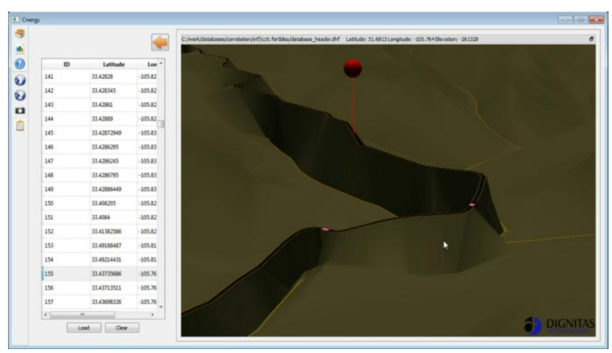

Figure 1: Unrealistic "Wall of China" in the CCTT Fort Bliss Database

A simple example of a quality test is the "Wall of China" effect sometimes seen in virtual programs like CCTT as a result of strict pitch parameters on roads combined with rough terrain. Such features look unrealistic, suspending the trainee's belief in the simulation context (Figure 1).

Correlation Tests

Correlation tests are those which ensure that two or more database formats or instances contain functionally equivalent data. For example, verifying that an important road exists in both databases with the same geometry, or ensuring that line of sight is blocked at the same location in both databases.

In a perfect world, two databases of the same world region could be assessed in a purely geometric manner, but this is only feasible for limited use cases. Different run-time databases use different tradeoffs in storage, representation, and accuracy. This, however, may be fully valid within the system context of the databases. For example, the CCTT radio database uses a coarse grid of elevation samples to support radio degradation, while the electronic map uses regular contours, and the underlying geometry database uses a triangulated irregular network. These three representations have inherent geometric miscorrelation that is fully acceptable within the context of their uses in that simulation.

Similarly, some buildings are represented in OTF as point features, which results in them being processed as rectangular volumes for performance reasons. However, as a conscious system tradeoff, these same buildings are often represented with peaked roofs in visuals to improve the appearance of the database. Such tradeoffs can result in significant geometric differences based upon conscious system tradeoffs between performance on geometry services and the visual database appearance presented to trainees.

Evaluation of Results

Another problem with quality and correlation testing is that sharing and viewing test results is difficult. It is often necessary to share results across working teams to try and figure out the root of a given issue. This usually consists of passing along test logs which must be parsed by whoever is receiving the results in order to extract relevant data. This is a daunting task since logs can contain many thousands or even millions of errors. This also means that after parsing the logs, someone would likely need to manually input error locations into another application for further analysis. Clearly, text logs are not always the best means to review results. Often a 3D interactive view of location-centric errors will be more effective for rapid evaluation than, say, reviewing geodetic coordinates in text.

Automation versus Manual Evaluation

When it comes to correlation testing there is no single testing mechanism that can provide a thorough understanding of correlation between databases. In order to achieve such results correlation testing requires the use of automated tests, interactive user tests, and visualization. Each of these processes has its advantages and disadvantages. Because of this, one process alone is not able to cover all of the cases necessary to completely understand the correlation state of a set of terrain databases, especially when assessing possible errors in the context of training impact.

Automated tests consist of a collection of tests that can exhaustively or selectively analyze a database based on some user-defined parameters. The problem with these types of tests is that it is difficult to define a set of parameters that are able to identify errors without introducing many false-positive results. Because of this, automated tests usually require that a person sift through the error logs and make a judgment call on whether a given error is truly an error or a false positive. Such a task is not necessarily a difficult one, but rather a time consuming one. Throw in the fact that sifting through these logs often requires that a user sit in front of a computer screen and manually input coordinate after coordinate in order to make a judgment call, and the factors of user-error and fatigue are added to the variables of correlation testing.

Interactive tests give the user the flexibility to test specific locations but it is very time consuming, and arguably impossible, for a user to perform thorough testing with only interactive testing.

Visual inspection, although very beneficial, may not reveal some of the problems that may be embedded deep within the database data structures. For example, it may be very difficult for a user to detect whether a given terrain triangle is traversable, or if it has the correct feature type associated with it.

Only through a combination of these three testing approaches can the most comprehensive test results be achieved. An ideal combination of these testing approaches would be executed by first running a set of automated tests that will output lists of errors and identify areas of interest, then visualize the results to make sure they are not false positives, and then perform interactive tests in those areas for further analysis if necessary.

As an example of why purely automated results may not be effective, the CCTT Korea database had a clear pattern of 960m square sections of terrain that dropped to zero elevation, creating massive and unrealistic looking holes in the visual database. As alarming as these errors appeared to be based upon automated test evaluation, all of the locations happened, by chance, to be deep in mountain ranges with no nearby regions that could realistically be used for CCTT's training use case. In short, this was a seemingly critical error that had no meaningful effect based upon hard-to-measure system criteria of database use and training context. The same error in a virtual air training context would be a major concern, since the anomalies would be highly visible from the air.

CURRENT STATE OF ARMY M&S TESTING

Based upon the authors' real world experience with programs like SE Core, CCTT, AVCATT, Conduct Of Fire Trainer – Situational Awareness (COFT-SA), Warfighter's Simulation (WARSIM), OneSAF, and more, there is very little in the way of automated or assisted testing approaches in practical use. Almost all database evaluation is done via "flyovers" in visual, on a Plan View Display (PVD), or in a system context (driving around). Some low-level test utilities exist, but these are not widely used or output hard-to-read text files. In practice, databases are not thoroughly tested, and such testing is very patchy, with specific areas of strong testing (e.g. OneSAF's standalone test tool), but others with no support, such as visual-to-OTF correlation testing.

There has been a tremendous history of tools that are no longer consistently applied in the authors' experience. As mentioned earlier, the SEDRIS community implemented a wide range of tools, some specific to SEDRIS (rules checker, etc.) and others supporting multiple formats like Side-by-side [CAE, 2010] and SEE-IT. A number of isolated capabilities have been developed, such as Zcap [Sakude, 1998], and Artemis / Venator. Artermis is an example of a toolset that provided high-value feedback to the United Kingdom Combined Arms Tactical Trainer (UK CATT) program [Donovan, 2001] on database quality, but the test processes were highly specialized and often gave "false positives" based upon as-designed differences in database representations. The authors are also familiar with many low-level, specialized toolsets developed for individual programs, such as CCTT's sanity checker, OneSAF's visual test tool, SE Core's validation tester, and more.

Despite this history, only a very limited set of tools are being applied in database development for large Army programs the authors are familiar with. Where such tools do exist, they are not widely understood or applied across the industry, and solutions are very localized. As an example, the COFT-SA program had to convert OpenFlight/CTDB from the Advanced Gunnery Training System (AGTS) into OTF for use on new gunnery training applications. This conversion process was very difficult, and was greatly benefited by use of an obscure OneSAF toolset called the Visual Test Tool (VTT), which provided a means to assess OTF content. However, the VTT is difficult to use, highly specialized to OTF, and only used by a limited set of developers.

The remainder of the paper describes our open, reusable, user-friendly, and highly applicable toolset for database testing. While there is broad applicability the primary focus has been to leverage the growing use of OneSAF in virtual and constructive use cases, with a particular emphasis on virtual applications such as AVCATT, CCTT, COFT-SA, Construction Equipment Virtual Trainer (CEVT), and more.

OUR SOLUTION

Correlation Synergy: C-nergy

Based on the lessons learned from previous applications (as described in the Key Concepts section above), as well as experience gained from involvement in many Army M&S programs such as SE Core, the technical team set out to develop a flexible framework that would support multiple database formats as well as automated, interactive, and visual tests. "C-nergy" for "Correlation Synergy" is the newly created framework that allows disparate applications, tests, and database formats to work together and augment each other's testing capabilities. The research and development of C-nergy is being performed under Government Purpose Right Software and is funded through a SBIR by the U.S. Army Research Development and Engineering Command, Simulation and Training Technology Center (RDECOM STTC). Through this research, the focus has not been on implementing a comprehensive application that incorporates most of the major available database formats, or supports all of the available tests for any given format. Rather, the focus is on implementing a proof of concept that could provide a solid foundation for future development. Initially the framework has been implemented with two database formats: OneSAF OTF and Openflight. This gives one database type from the geometry/reasoning and visual realms, respectively, in order to gauge the correlation testing capabilities of C-nergy. These formats also provide coverage of virtual and constructive use cases.

To support additional database formats, C-nergy defines a set of interfaces that each format must meet in order to interact with the framework. Upon integration, a new incoming format would be wrapped in a software layer that would interact between our C-nergy interface and the format software. This is beneficial since it allows the C-nergy framework to remain agnostic about any one specific database format, and rather focus on the fact that it can call on Format X to retrieve data such as the height of terrain at a given coordinate. This strategy keeps C-nergy flexible by maintaining the software within a bubble that is independent of whatever database formats are prevalent at the time and allows C-nergy to evolve alongside the industry.

Automated Tests

The next challenge was to implement selected correlation tests within C-nergy. This challenge was simplified a bit by the idea that C-nergy only interacts with a set of predefined interfaces for any given format. The first correlation test that was developed was the Height Of Terrain (HOT) comparison. This test compares the terrain elevation between two formats at given coordinates. One of the challenges faced was that terrain database formats may be defined with different coordinate systems and projections. Because of this the Geodetic coordinate system was selected as the common coordinate "language". Formats that integrate with C-nergy must convert their internal coordinates to Geodetic coordinates. To do this the Geotrans coordinate conversion library is provided as part of C-nergy. Providing this library as the common coordinate converter also avoids the pitfalls of coordinate precision loss due to differing coordinate converters.

As previously mentioned, the C-nergy framework has integrated two database formats (OTF and Openflight). Where possible it is important to make use of existing runtime Application Programming Interfaces (API) for the database format so that the information that is provided to C-nergy tests and services is as close as possible to the functionality seen by simulation systems using each format. For the OTF format the ERC APIs are wrapped in a software layer that meets C-nergy's interfaces and then height of terrain information is extracted through those interfaces. The Openflight format, however, did not contain run-time services used in the simulations that are the focus of C-nergy, so instead the OpenSceneGraph (OSG) library is used to compute the intersection with the terrain triangles at a given coordinate once the Openflight geometry has been converted to OSG's internal format. This approach could be followed for all other integrated database formats, based upon whether they have a run-time services library available for use in targeted simulation systems.

Visual Inspection

Once a terrain format has been integrated into the C-nergy framework it is able to take advantage of C-nergy's visualization capabilities. Two main visualization modes have been implemented for correlation testing: Overlay and Side-by-Side.

The overlay mode renders two databases, one on top of the other (Figure 2). This gives the user the ability to directly compare the content of the two databases since differences in feature attributes, or offsets in feature placement, would draw the user's attention.

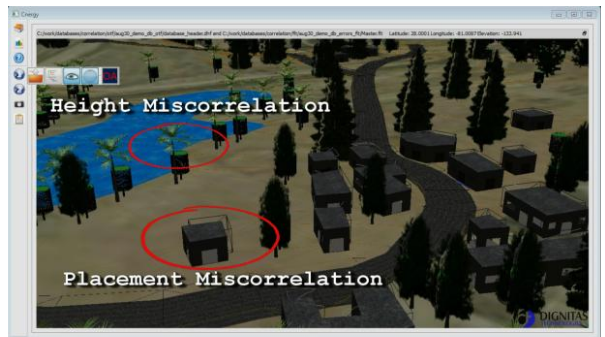

Figure 2: OTF and Openflight Overlay Mode

The Side-by-Side mode renders each database in a separate view window but connects the view cameras so that the user is able to see both databases from the same perspective (Figure 3). In other words, moving the camera in one view window results in an identical reciprocated camera move in the other view window.

In addition to providing visualization mechanisms the visualizer is also the venue through which the user can perform interactive tests like individual height of terrain or line of sight queries.

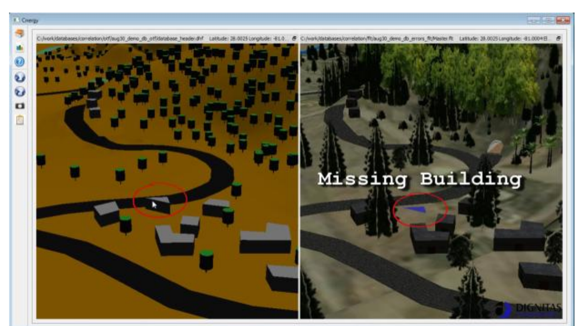

Figure 3: OTF and Openflight Side by Side Mode Extensions to C-nergy

The C-nergy framework simplifies adding new tests, which has been leveraged in response to several program use cases during our development. The recent Joint Improvised Device Defeat Organization (JIEDDO) effort to improve visuals for the Reconfigurable Vehicle Simulator (RVS) resulted in more dense building placements than had previously been used. This, in turn, resulted in obstacle avoidance boundaries for the buildings overlapping each other across urban roadways. The functional end result of this was that OneSAF entities would perceive the roadway as blocked even though the road looked clear in the visuals. The OneSAF Obstacle Avoidance (OA) database was easily and rapidly added to our support formats such that overlaps could be readily seen by database developers (Figure 4).

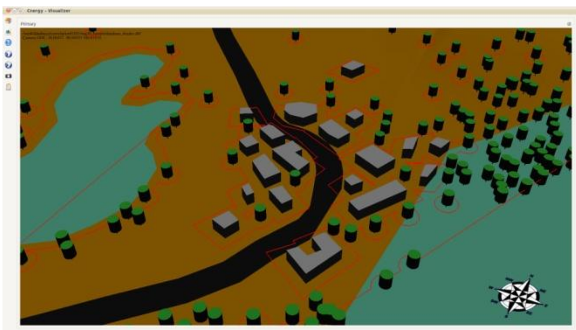

Figure 4: OneSAF OA outlines (red), buildings (gray), and roads (black)

Test Result Exchange

In addition to implementing a framework for Correlation testing, a goal was set to solve the problem of sharing and viewing test results across applications. If testers are able to easily share test results and view them in multiple application contexts, it would increase the speed and accuracy of testing processes. Moreover, testers would not have to manually sort through logs or develop one-off applications to parse log files. Rather, testers would simply import the results file and iterate through the errors giving them the ability to import a series of test results into various applications which will:

- Virtually eliminate the chances of user errors introduced by incorrectly typed locations
- Increase the speed at which testers can examine test results
- Allow applications to recall the parameters of a specific test
- Allow testers to exploit the strengths of disconnected applications by using one application to generate a set of results, then using the error locations to perform more thorough tests in a separate application and within a specific system context
- Allow testers to view results in different applications based upon their focus and need

These ideas led us to the development of the Format Independent Test Results (FITR) library. FITR is being developed as an external (i.e. third party) library and provides a mechanism for importing and exporting collections of tests, test parameters, and test results to C-nergy and other applications. As a proof of concept of the FITR library functionality, FITR was integrated into OneSAF which allowed the C-nergy test result to be imported, then quickly analyzed by teleporting to those locations on the OneSAF PVD.

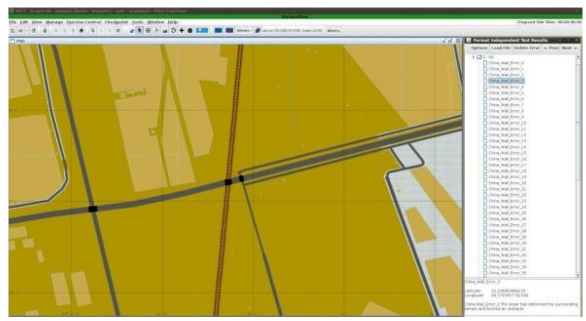

Figure 5: OneSAF with FITR integration

Bringing the pieces together

In essence, the goal of the C-nergy framework has been to provide testers with a "one stop shop" for quality and correlation testing capabilities. Through C-nergy the objective is to minimize (but not eliminate) the need to use multiple applications in order to perform tests on terrain databases. The goal is to simplify database testing by trying to connect the dots between automated, interactive, and visual tests so that the user can quickly "context switch" from one mode of testing to the next without the need to visit multiple applications until a highly application-specific view is needed. Through the use of FITR, communication of test results across toolsets, users, programs, and development/test teams becomes easier. FITR can be used to record errors in a variety of apps, including allowing a tester using the OneSAF Graphical User Interface to capture an error location for later evaluation in a 3D tool such as C-nergy or Ares. The FITR format makes collaboration and database problem solving easier since users would be able to iterate through database error information significantly faster than having to sort through output logs, manually type coordinates into an Image Generator command line or a PVD offset window, or convert between different coordinate systems to view possible errors.

CURRENT AND FUTURE WORK

There is a lengthy list of planned improvements the Team is researching and planning to implement in the near future. These ideas are derived from our own extensive experience with a variety of Army training systems, feedback from various programs with immediate needs, and projected requirements. The following subsections describe a few examples of upcoming work.

Profile-based HOT Comparison

The idea of profile-based terrain elevation correlation testing evolved from the need to provide very detailed HOT analysis while also attempting to keep test execution times relatively low. The most common method to testing HOT is a sample-based approach where the user defines a sampling interval and an elevation delta tolerance. The sample-based approach begins by creating a step grid over the database and samples the terrain elevation at the grid locations. The problem with this approach is the higher the resolution of the sampling rate the longer the test takes to run. Any given spacing can still “jump over” a critical but localized error. There are many variables that affect the speed of the test, such as available memory and speed of the system, but in the end the number of samples is directly tied to the area of the database and the inverse of the sample interval.

The profile-based approach does not directly correspond to the number of samples to the overall size of the database. Rather, our approach exploits the fact that if a line is drawn across a triangle with the end points of the line existing at the edges of a triangle, then elevations sampled off of the triangle along the line are guaranteed to be within the range of the elevation at the end points of the line. In other words, no sampled elevation will be higher or lower than the line endpoints. This means that every point along the line can be ignored, except for the endpoints. This also minimizes the risk of a particular sample point “jumping” an error.

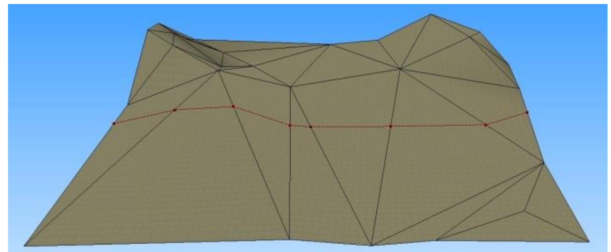

Figure 6: Profile-Based HOT Sampling

Figure 6 shows an example where a terrain was sampled eight times across a section using our profile-based strategy. In Figure 7 the same terrain is shown, but sampled at 10 meter intervals. The interval-based sampling resulted in 176 sample points. This means that, for this specific terrain example, the profile-based approach had to execute only about 5%, or (8/176), of the samples needed by the interval-based approach. Increment the sampling rate to 1 meter and now the profile-based approach only executes 0.5%, or 8/(176*10), of the samples necessary by the interval-based approach.

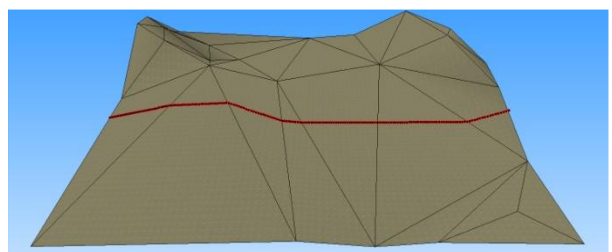

Figure 7: Interval-Based HOT Sampling

This concept could be applied to the HOT test by creating lines parallel to the latitude lines across a database. The distance between these lines would be the sample interval that is defined by the user. Once the lines are collected, a terrain profile would be generated across the line by collecting the elevation of all of the intersections between the line and the terrain triangles. This would result in a cross section of the terrain along the profile line. Profiles would be collected along the same latitude lines for each of the databases being tested. Once the profiles are collected, the peaks and valleys would be compared to determine the elevation delta between the two. After the profile comparison is finished, the points at which the profiles differ by a magnitude greater than the tolerance could easily be derived.

This elevation correlation approach achieves two things:

1. It captures all of elevation information along the test profile and essentially achieves an infinite sample interval across the terrain profile. A typical sampled-interval HOT test would miss any information between two sample points.
2. It is able to skip many of the points that fall within a given triangle since the concern is only with the elevation at the edges of the triangle.

This approach will significantly speed up the execution of geometry comparison terrain tests since many sample points would be skipped entirely. This algorithm can leverage terrain service layers that provide a profile as a service (as ERC does) to provide a functional test where such services exist. It also does not wholly replace the interval-sampling technique, which has the value of directly testing elevation queries so commonly used in simulation applications. It does, however, provide a very fast way to identify problem areas that would need to be analyzed in further detail.

Feature Comparison

Feature comparison testing presents several challenges since database formats have many differing ways of representing the same feature type. For example consider comparing a “tree” feature between OTF and Openflight databases. OTF represents trees with two cylinders, one for the trunk and one for the foliage. Openflight databases, on the other hand, can represent trees as complex models (where each branch of the tree is being represented in detail), a single polygon billboard, or a collection of crossed billboards. Given these variations, how do you compare the two accurately? To complicate the problem further, billboards could contain large areas of alpha (i.e. transparent areas), which make the actual dimension of the represented tree feature much smaller than the billboard size. Therefore using the extents of the billboard is not necessarily accurate.

Another problem with comparing features is that sometimes it is difficult to detect specific features in some databases. Openflight databases, for example, often contain references to external models for specific features. The problem is that it is not always straightforward to determine what those models are. Are the models buildings, trees, or vehicles? Sometimes, it is possible to infer a model type by looking at the model name. Unfortunately that does not guarantee that the model name gives a clue as to the feature type. Consider the following names for palm tree models:

- Palm_Tree.flt
- PlmTree_large.flt
- oasis_vegetation_105.flt
- p7584.flt

All of these could be valid names for a palm tree models in Openflight. Thus, some database context would be necessary to identify specific features in order to perform feature to feature comparisons.

An idea currently being researched is to identify feature problems by performing a comparison of the bounding volume for the features. The focus would not be concerned with the specific type of feature, but rather the fact that a feature with similar dimensions exists in both databases. Although this test would not do a one–to-one comparison between databases, it would be a good indicator for missing features or features with mismatched dimensions. Testers could then use the error locations to move the C-nergy visualizer to the problem area and quickly determine whether a feature problem exists. Another approach is to fall back to a highly geometry and service-based comparison, such as executing a high volume of line of sight tests in a region. Detecting differences in such geometric or service-based tests will not always indicate what the problem is, but can be a clue for problem areas. One reason we have limited effort in this area is because of other active and successful efforts investigating this area, such as [Palmer, 2011].

Expansion of Formats and Tests

Our work to date has been focused largely on development of C-nergy with a small number of formats and tests to prove the concept. An Alpha release of C-nergy has been provided to a set of near-term transition partners such as the U.S. Army Training and Doctrine Command Intelligence Support Activity (TRISA), Navy Virtual Library (NVL), and Battle Lab Collaborative Simulation Environment (BLCSE) to provide feedback.

Looking forward, several possible terrain formats are being evaluated to add to our supported sets as well as a long list of high value quality and correlation tests beyond those described in detail above. In the coming months, the focus will be on transition of the C-nergy capability to selected Army programs for use in terrain database evaluations.

ACKNOWLEDGEMENTS

The authors would like to thank the U.S. Army Research Laboratory and the Small Business Innovation Research office for the funding and leadership that has allowed us to succeed. We would also like to thank the

programs and organizations that are evaluating our alpha release and providing invaluable feedback to ensure we target the correct test capabilities.

REFERENCES

Bizub, W., & Cutts, D. (2007) Live Virtual Constructive (LVC) Architecture Interoperability Assessment *Proceedings of I/ITSEC 2007*

Borkman, S., Hoffman, M., & Peele, G. (2010) Empowering Our Warfighter: Using iPhones for Situational Awareness, Proceedings of I/ITSEC 2010

CAE, (2010). *Technology Solutions: CAE Side-by-Side.* Retrieved June 11, 2012, from http://www.cae.com/en/military/side.by.side.asp

Donovan, K., Stump, K., & Watkins, J. (2001). Virtual Database Experience with SEDRIS. *Proceedings of the Fall Simulation Interoperability Workshop 2001.*

Dukstein, G., Watkins, J., & Deakins, R. (2007). Integration of OneSAF Environment Runtime Component into Existing Virtual Simulations. *Proceedings of I/ITSEC 2007.*

Hughley, P., Watkins, J., Moerk, J., & Chang, A (2006). Integration of OneSAF Objective System into Existing Virtual Programs. *Proceedings of I/ITSEC 2006.*

Mamaghani, F., Foley, P. (2004). SEDRIS – A Collaborative International Infrastructure Technology. *Proceedings of the Simulation Interoperability Workshop 2004.*

Marshall, H., Chandler, E., McEnany, J., & Thomas, J., (1996). SAF and Manned Simulators Correlation Issues in CCTT. *Proceedings of the 6th Conference on Computer Generated Forces and Behavior Representation 1996.*

Palmer, T., & Boyd, R. (2011). Leveraging Graphics Processor Capability to Uncover Hidden Terrain Database Errors. *Proceedings of I/ITSEC 2011.*

Prager, S.D., Miller, D.D., & Gafford, D. (2002). Terrain Interoperability in Large Federations: Correlation and Consistencey in MC02. *Proceedings of I/ITSEC 2002.*

Sakude, M., Schiavone, G., Morelos-Borja, H., Martin, G., & Cortes, A. (1998). Recent advances on terrain database correlation testing. *Enabling Technology for Simulation Science II, Proceedings Vol. 3369, pages 364-376.*

Schiavone, G., Nelson, R., and Goldiez, B., (1994) Statistical certification of terrain databases. *Proceedings I/ITSEC 1994.*

Stevens, C., et al. (2001). A Common Runtime Format for Virtual and Constructive Simulations. *Proceedings of the Spring Simulation Interoperability Workshop 2001.*