DYNAMIC TERRAIN IN A DISTRIBUTED SIMULATION ENVIRONMENT WITH LOW COST PC

Rita Simons
Simulation, Training and Instrumentation Command (STRICOM)
Orlando, Florida

Jesse Liu
AcuSoft
Orlando, Florida

Graham Upton and Tim Woodard
Diamond Visionics
Vestal, New York

ABSTRACT

As technology utilized in simulation has grown, so have the requirements for a realistic solution to the dynamic terrain problem in the synthetic environment. In order to support the DoD Simulation Based Acquisition (SBA) initiative, the need for a high fidelity Synthetic Natural Environment simulation is fundamental and critical. Specifically, a realistic dynamic terrain solution is required by the Advanced Concepts and Requirements (ACR) community, and maneuver forces using simulation to support their collective training objectives.

Research has previously been conducted in the area of dynamic terrain implementation, and the dynamic environment. Dynamic terrain is not new to the simulation community, however previous efforts have required high-end computational platforms, were unable to perform in real-time, and were often low fidelity in appearance. With the fast paced improvements in the performance of Personal Computers (PCs) and image generators, the realism that is required for a dynamic terrain implementation is now achievable on a PC. The US Army STRICOM sponsored a Phase I Small Business Innovative Research (SBIR) topic addressing these requirements, which has progressed to a Phase II effort. In the Phase I effort, Diamond Visionics Company (DVC) and AcuSoft teamed to provide a PC based technology demonstration of dynamic terrain incorporating simple soil dynamics. Phase II objectives include the development of a platform independent software solution that has an open architecture and application program interfaces, providing the fundamental functionality required by digital synthetic environments to implement dynamic terrain in a DIS/HLA network. The developed solution will use SEDRIS (Synthetic Environment Data Representation and Interchange Specification) as the underlining data standard.

This paper will address the use of dynamic terrain in a Distributed Simulation Environment utilizing low cost PC platforms. It will examine the challenges of implementing dynamic Synthetic Natural Environment in a distributed simulation environment, specific issues related to DIS networking, and the challenges and advantages associated with HLA migration. It will also address interoperability with simulations and systems that encompass a wide range of fidelity, resolutions and application domains.

Biographical Sketch:

Rita Simons is currently a Visual Systems Engineer at STRICOM. She is the COR on the Low Cost PC Based Real-Time Dynamic Terrain Phase II SBIR. She earned a Bachelor of Science degree in Electrical Engineering from the University of Central Florida (UCF) in Orlando, Florida and is currently pursuing a Master of Science degree in Simulation, in the Industrial Engineering Program at UCF.

Jesse Liu is the President of Acusoft, Inc., a simulation software company based in Orlando, Florida. Mr. Liu has a Master of Science degree in Electrical Engineering from the University of Florida in Gainesville, Florida. Mr. Liu has 16 years of experience in research and development of simulation and training systems.

Graham Upton is the Director of Engineering at Diamond Visionics Co. in Binghamton, NY. He has 26 years of experience in systems engineering and management for flight and ground vehicle simulators. He holds a degree in Aeronautical Engineering from Southall College of Engineering in London, England. As a design engineer for Link Flight Simulation for over 21 years, he was involved in the flight and ground vehicle products. He has led R & D teams to develop new products in this fast-paced industry, and has managed both Army and Commercial programs. He has written many papers in areas of simulation and training.

Tim Woodard is a software developer at Diamond Visionics specializing in real-time graphics. He has developed software to support high-speed deformation of visual databases on PC-based platforms. This technology has been used in laparoscopic surgery simulation and dynamic terrain simulation. Tim has completed two years of undergraduate study in the Honors Program at Heidelberg College in Tiffin, Ohio studying Computer Science and Mathematics.

DYNAMIC TERRAIN IN A DISTRIBUTED SIMULATION ENVIRONMENT WITH LOW COST PC

Rita Simons
Simulation, Training and Instrumentation Command (STRICOM)
Orlando, Florida

Jesse Liu
AcuSoft
Orlando, Florida

Graham Upton and Tim Woodard
Diamond Visionics
Vestal, New York

INTRODUCTION

Real-time dynamic terrain is one area that hasn't successfully been incorporated in the synthetic environment. As the use of simulation has become a major contributor to training, realism has become the requirement. For the maneuver force and the combat support elements, a disadvantage has been the absence of dynamic terrain. For the maneuver force training in a collective environment, dynamic terrain provides cues that affect high order commands. Track marks in the soil provide indications of strength and direction of forces that have previously passed over the terrain. Freshly dug up dirt may give an indication of some sort of obstacle being placed, such as a mine. A cleared lane through a minefield indicates area of safe passage. These are all dynamic terrain cues that provide information. In today's synthetic environment, this capability does not exist. Many efforts have researched the area and idea of dynamic terrain, but none have provided a reasonable solution that can be used by the soldiers for collective simulation-based training. With the wide assortment of simulators currently fielded in the various battle labs and simulation training facilities, a platform independent software solution for dynamic terrain is an Army training requirement.

BACKGROUND

There have been numerous research efforts investigating dynamic terrain. Both the Institute of Simulation and Training (IST), and the Synthetic Theater of War (STOW) investigated dynamic terrain and the use of dynamic terrain in a distributed simulation. Results were computationally intensive efforts that inhibited real-time simulation. With the advances in technology, STRICOM sponsored a phase I Small Business Innovative Research (SBIR) contract to study an approach for developing a low cost, Personal Computer (PC)-based dynamic terrain solution. The technical objective was to demonstrate the feasibility of an approach for realistic, real-time and distributed dynamic terrain simulation in a synthetic environment hosted on low cost PCs. One of the results of the phase I contract was a determination that current PC graphics hardware may be more suitable for dynamic terrain than some high-end graphics systems. Rendering can be completed in an "immediate mode", sending a complete update of the scene to the hardware. In one of the phase I efforts, Diamond Visionics (DVC) teamed with Acusoft, Inc., demonstrated dynamic terrain on a section of the Primary 1 Close Combat Tactical Trainer (CCTT) terrain database. Based on this result, a phase II contract was awarded for a prototype dynamic terrain implementation. Objectives of the phase II effort concentrated on analyzing and optimizing the performance required for a dynamic terrain implementation on a PC to provide a real time (30 Hz minimum), realistic simulation. It is a continuation of the phase I efforts. Specifics are to determine the most suitable way to implement the dynamic tessellation of the terrain database, incorporate the University of Iowa's higher fidelity tire soil interaction model, implement dynamic texture to achieve realism, utilize level-of-detail (LOD) management, integrate database paging, and integrate these efforts in a distributed synthetic environment. The phase I effort determined that Modular Semi-Automated Forces (ModSAF) 5.0/OneSAF Test Bed (OTB), Distributed Interactive Simulation (DIS)/High Level Architecture (HLA), and virtual simulators can support real-time dynamic terrain objects simulation.

Requirements of the phase II effort include:

1) PC Based.
2) Use of a network protocol (DIS).
3) Use of Semi-Automated Forces (ModSAF 5.0).
4) Alter the terrain database in real time.
5) SEDRIS.
6) Platform independence.

Two exercises incorporating the above objectives and requirements will be demonstrated at the conclusion of the phase II effort. These exercises consist of a heavy and wheeled scenario generated by the Maneuver Support Battle Lab (MSBL), Ft Leonard Wood, MO. The exercises will show specific instances of dynamic terrain (such as a breach, footprints, explosions, buildup of soil on wheels/tracks, and effects of saturated soil on vehicles). The scenarios will be implemented using both manned simulators, and ModSAF, on a DIS network. In addition, the CCTT P3 (Ft Hood) Synthetic Environment Data Representation and Interchange Specification (SEDRIS) Transmittal Format (STF) will be converted to Openflight and utilized as the terrain database. The architecture utilized for the exercises are outlined in figure 1, with the dynamic terrain implementation identified as Dynamic Terrain (DT) Core.

With the wide variety of IG technology currently being utilized in Army simulation, the most important objective of the phase II effort is platform independence. Utilizing DT Core, any Image Generator (IG) will be able to implement dynamic terrain, with minimum changes required for integration.

THE PC COMPUTATIONAL AND GRAPHICAL CAPABILITY

In military simulation, there is an ever-increasing demand to support more complexity in the visualization of synthetic environments. Providing higher fidelity not only requires faster graphics hardware with new features, but also efficient scene management tools that can take advantage of these advancements. The ability of PCs has increased amazingly in recent years. Tasks that not long ago

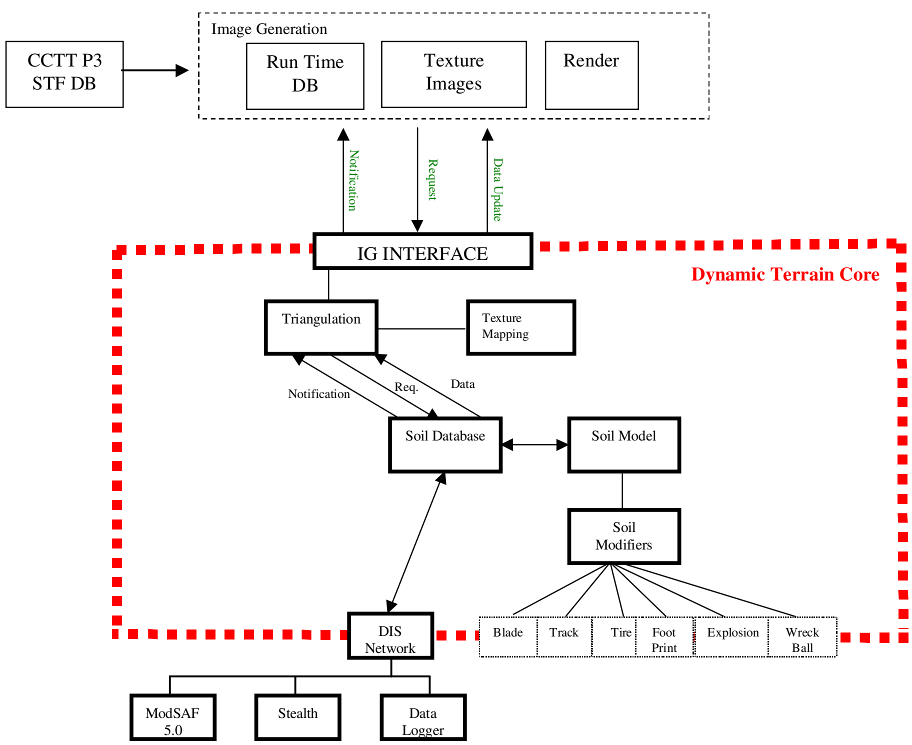

Figure 1. Dynamic Terrain Exercise Architecture

required high-end proprietary workstations can now be performed on low-cost platforms. A primary reason for the major advancements in PC technology can be attributed to the gaming and entertainment community. Because of the popularity of 3D games, graphics board vendors have been adding more and more features in a fierce competition where the fastest frame rate and best image quality wins. These new features include single-pass multi-texturing (which allows for realistic surface detail and lighting), full-screen sub pixel antialiasing to eliminate "jaggies", and hardware supported transformation and lighting ("T&L") to reduce the burden on the PC's CPU, giving it more time to perform other tasks such as artificial intelligence. The result of this competition is that fast high-quality graphics can now be generated using low-cost hardware and standard APIs (Application Programmer's Interfaces).

The two main measures of a graphics accelerator's performance are polygon count and pixel fill rate. Polygon count is the number of triangles that can be sent to and rendered by the card in one second. Pixel fill is a measure of the number of pixels that can be rendered in one second. When trying to achieve high-speed, high-quality graphics, both measures are important. High polygon count allows for more complex models to be rendered, where pixel fill allows those polygons to be textured, lit, shaded, and antialiased. For realistic visual simulation, a graphics system must be capable of both high polygon count and pixel fill rate. This allows for an environment that appears real enough so that a user can be immersed as though involved in a real-life scenario. In a dynamic terrain simulation, effects such as dust and visible differences in soil type can be used to increase the level of realism. Effects such as these that were difficult to produce even on high-end systems not long ago are now possible with PC technology. Figure 2 illustrates the type of realism that can be achieved.

One advantage of the shift from high-end workstations to PCs for graphics is that industry standards have emerged for the APIs used in graphics software. The two major APIs used today are OpenGL™ and DirectX®. OpenGL has its roots in high-end graphics and is popular because of its platform independence. DirectX is Microsoft's® standard and is widely used for 3D games that run on Windows®. Both standards provide many advanced features and are constantly improving. Many graphics boards sold today have full support for both of these APIs. If an application is developed using one of these APIs, then it will generally work with any card that supports that API. This not only makes it easier to develop graphics software (because of the large industry support for each of these standards), but it also means that applications written today can take advantage of tomorrow's hardware advancements without the need to change them.

Advancements in consumer-level graphics hardware have also brought about a more subtle advantage: real-time immediate-mode rendering is now feasible. In the past, most high-end graphic systems were based on retained-mode rendering. Retained-mode rendering is when objects are sent to the graphics system once at initialization and are stored (retained) there. Then when that object needs to be rendered, it doesn't have to be re-sent to the graphics system. This makes most graphic operations more efficient but has the disadvantage of making objects un-modifiable. This was not a problem in the past, however, because there was never a requirement for a dynamic database. Dynamic terrain by its definition requires the ability to modify the visual database. With the increase in ability of PC-level graphic boards, immediate-mode rendering can now be used while still maintaining interactive frame rates. With immediate-mode rendering, objects are not stored in the graphics system, but are re-sent each time they are rendered. While this does make some operations less efficient, it allows for objects to be modified in real-time from frame to frame. This is what makes dynamic terrain effects possible on a PC. Today, most graphics cards support both retained and immediate mode rendering simultaneously, so trade-offs can be made to achieve the desired result.

Figure 2.

According to Moore's law, CPU complexity and performance doubles every 18 months to two years. This means that the potential upgrade of hardware could have a 2-fold performance increase on an existing application with little or no changes in software. With this tremendous advance in chip development, the ability to perform real-time, PC based dynamic terrain has now become a reality. With the heavy competition in the gaming community, there is no end in sight to the growing list of features and performance available in PC graphics cards. For the simulation community, it means that by taking advantage of low-cost PC technology, features and capabilities previously limited will add to the value and realism of training and simulation systems.

DYNAMIC TERRAIN AND ADVANCED DISTRIBUTED SIMULATION INTEROPERABILITY

The graphics representation of dynamic terrain effects is one of the most important aspects of dynamic terrain simulation. When in a standalone mode, dynamic terrain effects are easily visible to the simulation that created the effect. However, in an advanced distributed simulation, dynamic terrain effects are much more complicated. In a distributed simulation, all simulation assets now reside on a network, and must be able to communicate any state changes to the other assets. When operating in a distributed simulation, several issues must be addressed. Networking issues to be addressed include:

1) Interoperability of dynamic terrain behavior in a heterogeneous simulation system that consists of simulations and/or simulators of various fidelity levels.
2) The content of information that needs to be communicated.
3) The protocol of the communication.

To successfully operate in a distributed simulation, all simulations and simulators must be interoperable. Interoperability is defined as the ability of a simulation to provide services to, and accept services from, other simulations and to use the services so exchanged to enable them to operate effectively together. Simulations/simulators are interoperable when their performance characteristics support a “fair fight” to the fidelity required for the exercise. “Fair fight” is the ultimate goal of interoperability. Two or more simulations/simulators are considered to be in a “fair fight” when differences in the performance characteristics have significantly less effect on the outcome of the conflict than actions taken by the participants. In a distributed simulation, the warfighters, either warfighter-in-the-loop, or Semi-Automated Forces (SAF) entities must all perceive the same information. This means that any dynamic terrain effects must be relayed to all simulations on the network so that all participants have the same interpretation of the terrain database. Technically, in the context of Synthetic Natural Environment (SNE), this requirement can be characterized as follows:

1) Both the quality and the quantity of the battlefield information delivered to all simulation systems shall be identical.

2) The interpretation of the information shall be identical.

3) The action and reaction based on the information shall be compatible.

Considering the first requirement, in order to provide the same quality of information, one has to adapt the data generated by the dynamic terrain simulation in a commonly agreed-upon fidelity level definition. This is a spatial-domain issue, and a very challenging task. For example, a warfighter-in-the-loop simulator with high performance PC dynamic terrain visual simulation may achieve the terrain resolution of sub-1-meter level, but the SAF entity, such as ModSAF 5.0, can only resolve terrain resolution at a 10-meter level. A multi-resolution reconciliation algorithm has to be established in order to ensure the perceived quality of data is the same. Within the context of the Phase II Dynamic Terrain SBIR project, the research of this topic is deferred for a future phase. However, the need to develop an agreed-upon level definition remains as one of the most important dynamic terrain interoperability issues. The following possible approach is suggested to solve this difficult problem as a direction for future research.

Develop a description language that can describe the “characteristic” of the terrain. For example, the visual system uses textures to present newly developed tank tracks. The visual cue has to be presented as a “message” to the SAF entity as a “TREADMARK-TANK-DEPTH-meter-0.2-NEW”. The interpretation or “perception” of the information can be implemented with various levels of spatial-domain resolution and fidelity levels without distortion. This approach also has the advantage that it can be very data efficient - there would be no need to transfer actual terrain grid or polygonal data across the simulation network. Furthermore, this approach can be implemented without requiring computationally intensive algorithms. This proposed approach is very different from other dynamic terrain research efforts in the past, such as the IST effort, ModSAF (STOW) effort, and CCTT.

On the subject of information quantity, it is mostly a time-domain issue. In a distributed simulation system, all simulators may be operating on their own simulation frequency, or so called tick-time. The target for the PC dynamic terrain visual simulation is 30 Hz. SAF systems run at much lower resolution tick-time, due to the fact that they usually have to handle many entities on a single computer. For example, ModSAF 5.0 uses a soft-15Hz tick rate, with a very wide sliding window of acceptance, i.e. 2Hz. One could relatively easily establish a common sampling frequency, i.e. a "beat" for the dynamic SNE, for the simulation system. The technical challenge is to avoid latencies and un-fairness in the battlefield. For example, a warfighter-in-the-loop simulator may have perceived the removal of an obstacle while the opposing force simulated by a SAF may observe the same phenomenon 100ms later. In a weapon system that has automated 'search-and-destroy" capability, this time-domain quantization error may be totally unacceptable. The communication protocol that is used by the simulation system will have a major impact on this issue. This is addressed further in the sections that discuss the use of network protocols.

Within the context of the Phase II Dynamic Terrain SBIR project, the DIS protocol will be implemented with ModSAF 5.0. ModSAF 5.0 is identified as the preferred SAF implementation due to the level of current user penetration. Various algorithms will be evaluated to minimize the potential time-domain quantization distortion. The objective is to be able to allow the PC dynamic terrain equipped warfighter-in-the-loop simulator to conduct engagements with ModSAF 5.0 entities. Currently the plan is to implement two scenarios - one heavy and one medium - with the PC based dynamic terrain capability, utilizing ModSAF 5.0 simulated entities to participate in breach operations.

DIS ENVIRONMENT CHALLENGES

Simulation has progressed from standalone simulators to distributed simulation. The Army has invested heavily in the use of distributed simulation, using both Distributed Interactive Simulation (DIS) and High Level Architecture (HLA) as the simulation backbone. The decision was made to utilize ModSAF 5.0 as the SAF system. Since ModSAF 5.0 supports the DIS protocol, the focus for this effort has been on the use of DIS as the network protocol.

DIS is a combination of synchronous and asynchronous protocols. It supports the communication of entity states, such as the Entity State (ES) Protocol Data Unit (PDU), and simulation events, such as Fire PDU and Detonation PDU. There are two fundamental approaches to implement the dynamic terrain using the DIS protocol:

1) Transmit terrain changes as a type of state information. This approach is used typically in a "Terrain Server" type of implementation.

2) Transmit terrain modification events. All the simulators and simulations will perform dynamic terrain calculations based on their own implementation.

The obvious advantage to approach (1) is that there is no ambiguity as to what the modified terrain is at any given time. However, the disadvantage could be that it might require too much data to be transferred in a busy battlefield where a lot of terrain modification is occurring. Approach (2) potentially eliminates the need to transfer large amounts of data; however, it has the potential to suffer from interoperability problems due to implementation variations as has been previously discussed.

For the Phase II Dynamic Terrain SBIR project, a combination of (1) and (2) will be implemented. Two categories of information will be established: geometry objects and feature objects. For example, a tread mark is a feature object, but a new tank bunker is a geometry object. The interpretation of the data will be implemented with ModSAF 5.0 using a multi-layer approach. The first layer only accumulates and maintains a list of feature geometry modifiers in a spatial-index database structure similar to existing ModSAF micro-terrain implementation, while the second layer will actually perform the terrain geometry calculation should the modification become relevant to any simulated entity. The deferred interpretation approach is preferred based on estimated computation requirements. This multi-layer interpretation approach also allows a "Terrain Server" implementation on the simulation network if it helps to alleviate the computational constraint on the ModSAF computer.

HLA BENEFIT AND MIGRATION PLAN

As mentioned earlier, we only plan to implement ModSAF 5.0 with DIS protocols for the Phase II Dynamic Terrain SBIR effort. The potential of HLA implementation is provided only as a conceptual model.

HLA has the inherited advantage of being more flexible than DIS. This flexibility combined with the RTI time management capability allows the implementation of dynamic terrain to be much easier than using DIS. The most difficult task in implementing the proposed terrain description language approach is to manage the timing and data distribution. With HLA, simulation federates have greater capabilities to select relevant data without using data filtering mechanics. More importantly, the time-domain, the data quantity issue, and characteristics of the simulation system will be consistent as a result of using HLA time management features. A sophisticated dynamic terrain management federate that maintains the dynamic terrain events vs. an entity fidelity profiles matrix can be implemented. Only relevant data will be delivered on an as needed basis. Combined with the data efficient terrain description language, the HLA migration is anticipated to be very beneficial. As ModSAF 5.1 supports the HLA Agile FOM, the migration to HLA is anticipated to be relatively easy, based on our ModSAF 5.0 DIS prototype.

CONCLUSION

Currently, the SNE lacks a real-time dynamic terrain capability. As the Army migrates to a medium force composition, dynamic terrain capability is instrumental to assist the combat support elements and the maneuver forces. With the advancing technologies of PC chipsets, a real-time implementation of dynamic terrain is now viable. With these advances, real-time, PC-based dynamic terrain interoperability is now possible in a distributed simulation environment. With a platform independent software solution, dynamic terrain can be implemented in the many simulations/simulators being utilized in the Army with minor changes to the software required for integration. As the role of SAF/CGF becomes more important in the advanced distributed simulation system in the future; further research will be required for a complete dynamic terrain implementation in SAF/CGF systems. The proposed terrain description language approach is believed to have great potential to implement SAF/CGF system dynamic terrain behaviors.