SIMNET and Beyond:
A History of the Development of Distributed Simulation

Duncan C. Miller, Sc.D.
dmiller@springrock.net

Why Is SIMNET Important?

SIMNET stands for SIMulator NETworking. Initiated in 1983, it was the first "shared virtual reality" distributed simulation system, which continues to have significant influences. It was sponsored by DARPA, the Defense Advanced Research Projects Agency, the Department of Defense's principal high-risk, high-payoff research and development organization, established in 1958. In 1991, a study of various DARPA initiatives by the Potomac Institute for Policy Studies listed SIMNET as one of six programs that have had the most profound effects on the DoD. To put this in perspective, the other five were the ARPANET (the predecessor of the Internet); the individual computer workstation; phased array radar; the stealth technology used to make aircraft such as the F-117 fighter and the B-2 bomber "invisible" on radar; and ATACMS, the low cost, long-range tactical artillery rocket system used successfully in Desert Storm to destroy enemy surface-to-air missile sites and other targets (Potomac Institute for Policy Studies, 1991). That's quite a distinguished list in which to be included!

About The Author

Duncan C. Miller received B.S., M.S., Mech.E. and Sc.D. degrees from MIT, all in mechanical engineering. He was employed at Bolt, Beranek, and Newman, Inc. (BBN) from 1963-1993 and again from 2001-2008. From 1983–1993, he managed BBN's SIMNET group, which developed the SIMNET protocols and software, including vehicle simulation, network interface, graphics interface, communications simulation, and data analysis components.

From 1993–2000, he led the Distributed Systems Group at MIT Lincoln Laboratory, where he facilitated technical cooperation across government programs and industry participants in the development of Distributed Interactive Simulation (DIS), as well as the implementation of the High-Level Architecture (HLA) Run-Time Infrastructure (RTI) software that supported DARPA's Synthetic Theater of War (STOW). In 1997, STOW set the record for the largest real-time distributed simulation yet conducted (50,000 entities on 500 computers at 7 sites) using RTI software developed by his group. In 2001, he returned to BBN as Vice President and Manager of the Mobile Networking Systems Department.

From 1990-1996, he served on the Distributed Interactive Simulation Standards (DIS) Steering Committee, chairing the DIS Technical Committee from 1994-1996 during the development of the DIS Standards (IEEE Standard 1278). From 1994-1995, he was on the Government/FFRDC team that developed the High Level Architecture (HLA) for Modeling and Simulation (IEEE Standard 1516). In 1996-1997, he served on the DIS/SISO Transition Team, which established the Simulation Interoperability Standards Organization (SISO) as a 501(c)(3) non-profit organization. From 1997-2001, he served as Chair of SISO's Board of Directors, Chair of SISO's Conference Committee, and as a member of SISO's Executive Committee. From 2001-2012, he was SISO's Executive Director, after which he retired.

SIMNET-Related Standards

The SIMNET protocols (summarized below) were the foundation for the Distributed Interactive Simulation (DIS) protocols (IEEE Standard 1278-1993, with subsequent updates, extensions, and supporting standards) and were used for the Army's Close Combat Tactical Trainer (CCTT), Aviation Combined Arms Tactical Trainer (AVCATT), and subsequent procurements. DIS, in turn, was a primary source for the High Level Architecture (HLA) (IEEE Standard 1516-2000, again with various updates, extensions, and supporting standards). Subsequently, NATO based its standardization agreement for modeling and simulation (STANAG 4603) on HLA. These standards, all descendants of SIMNET, coalesced under the jurisdiction of the Simulation Interoperability Standards Organization (SISO), founded in 1996.

In 2003, the Institute of Electrical and Electronics Engineers (IEEE) Computer Society Standards Activities Board voted unanimously to grant the SISO Standards Activities Committee (SAC) status as a recognized IEEE Sponsor Committee. SISO now maintains three families of IEEE standards: IEEE 1278 (Distributed Interactive Simulation, or DIS), IEEE 1516 (High Level Architecture for Modeling and Simulation, or HLA), and IEEE 1730 (the Distributed Simulation Engineering and Execution Process). In addition, SISO maintains several standards and reference documents under its own auspices. It also maintains a relationship with the International Organization for Standardization (ISO) International Electrotechnical Commission (IEC) Joint Technical Committee 1, Subcommittee 24 (ISO/IEC JTC1 SC 24), which is responsible for responsible for defining international standards for computer graphics, image processing, and environmental data representation. The SEDRIS standards, which resulted from earlier SIMNET efforts to represent and exchange environmental data among heterogeneous simulation systems, are maintained by SC 24. For detailed information about these various standards, visit the SISO (sisostds.org), IEEE (IEEE.org), SEDRIS (sedris.org), and ISO (iso.org) websites.

Overview of this Paper

Previous papers by I/ITSEC Fellows (particularly Thorpe, 2010; Gorman, 2011; Shiflett, 2013) have described the development of distributed simulation from a historic and programmatic perspective and another SIMNET veteran has done so from a more philosophical perspective (Ceranowicz, 2014). While incorporating many of their insights, I will not repeat them here. Instead, I will focus broadly on the SIMNET software developers' and engineers' perspective – key design decisions and milestones, how the architecture and software were developed and tested, and how the team tried to cope with what seemed to be an exponentially growing list of user demands for both additional features and more rigorous validation.

Most people who saw SIMNET at various stages of its evolution inevitably focused on the most visible elements of the system: the crew compartments, displays, and controls of the various simulators, and the out-the-window views of the terrain, other combat vehicles, and dynamic effects. While these components were obviously central, there were many other elements of SIMNET that were essential to organizing, initializing, managing, observing, recording, analyzing, and replaying exercises, as well as providing the Local Area Network (LAN) and Long-Haul Network (LHN) communications that allowed simulations to include (ultimately) thousands of entities at multiple sites in a coherent, real-time event.

A brief listing of these components and their principal functions is presented in the sections below. For those who are interested, each component is described in detail in technical reports listed in the bibliography at the end of this paper.

There are several parallels between BBN's role in SIMNET and its earlier role in developing the architecture and software of the ARPANET, which became the foundation for the Internet. Although I was not personally involved in that program, many of my colleagues were, and the ARPANET open architecture philosophy was deeply imbedded in the BBN corporate culture and in many ways guided how we approached the development of SIMNET.

While this paper is presented from the perspective of hundreds of developers, engineers, and field support staff who took SIMNET from a crude initial concept demonstration to the foundation of the current international standards for distributed simulation, it includes very few technical details. Instead it focuses on the key ideas behind SIMNET and the people and events that brought them to fruition. Space limitations preclude mentioning more than a handful of the many individuals who participated in this development. However, a few SIMNET alumni have set up a website, simnet-history.org, on which we plan to remedy this situation by listing as many SIMNET participants as possible, along with their primary contributions, by the time this paper is published.

Key SIMNET Concepts

The core concept of SIMNET was the networking of multiple simulators, with each simulator providing its own controls, displays, and computational resources. No central control system scheduled events or resolved interactions among the simulation nodes. Instead, each node was autonomous, maintaining authoritative status for one simulated entity (e.g., a tank, helicopter, or missile system) and transmitting messages about the state and actions of its simulated entity to other nodes on a peer-to-peer basis. Each node was also responsible for receiving, interpreting, and responding to messages regarding events that might affect its own entity (e.g., a projectile impact, an exploding mine, a collision, etc.) and for reporting any resulting changes in its entity's state (e.g., damaged, destroyed, or unaffected.) As a result of this concept, the simulation network was inherently scalable, with each simulation node adding the necessary computational resources to support its own operations.

Another crucial design concept was that all information exchanged over the simulation network was, by definition, "ground truth." A simulator that fired a weapon was uniquely responsible for determining what object (if any) was hit, using both deterministic and probabilistic calculations. A simulator that was notified that it had been struck by a particular type of weapon, and where it was hit, was similarly responsible for determining what damage (if any) it suffered, using probability tables, as appropriate.

The "autonomous node" and "ground truth" design principles ensured that the simulation network would scale linearly with respect to computational resources required. However, as more nodes were added to the network, the number of potential interactions and messages communicated among the nodes grew rapidly, resulting in substantially greater network traffic. To limit overall network traffic, two more principles were employed. First, a simulation node would send updates only when its state changed. For example, a stationary entity would not send state updates (except for minimum "heart-beat" updates that assured other nodes that network communications had not been lost). Second, an entity moving in a straight line at a steady speed would send state updates only when it changed its speed or direction.

The latter design decision had several significant implications. It meant that each simulation node needed to maintain at least a simple, first-order model of every other entity that was within potential interaction range, and it would extrapolate the state of each such entity until it received new state information from it. This algorithm was called "Remote Vehicle Approximation" (RVA). In briefings on the SIMNET architecture, I often referred to this algorithm as "dead reckoning," comparing it to the ancient technique used by ships to estimate their position until they had an opportunity to obtain a more precise navigational fix. Each node had to maintain a complete RVA table for all entities with which it might interact.

A further implication was that each node had to maintain an RVA of the state of its own vehicle(s), so that it could determine when its "ground truth" state diverged by more than some agreed-upon threshold from the extrapolations that other nodes were using in their RVA tables. Whenever this occurred, the node was responsible for transmitting a new state update message, and all receiving nodes then corrected and updated their RVA tables. These updates included all externally "visible" information, including not only position, velocity, and altitude, but also the orientation of a turret, gun tube, or any other component that could move independently of the vehicle's main body. The updates also included such visible effects as dust clouds, smoke columns, muzzle flashes, etc., as well as any thermal or electromagnetic emissions that might be detectable by another entity equipped with the appropriate capabilities.

In describing these design principles and interactions in numerous briefings and presentations, I used the viewgraphs shown in Figures 1-3 to trace out an illustrative sequence of events. Figure 1 shows the main software modules and interfaces within an individual vehicle simulator. In this figure, the RVA table is shown as the "other-vehicle state table."

Figures 2 and 3 show the sequence of events that occurs when Vehicle A (on the left) fires a shell at Vehicle B (on the right). (The specific types of messages transmitted on the Local Area Network will be described in a later section.)

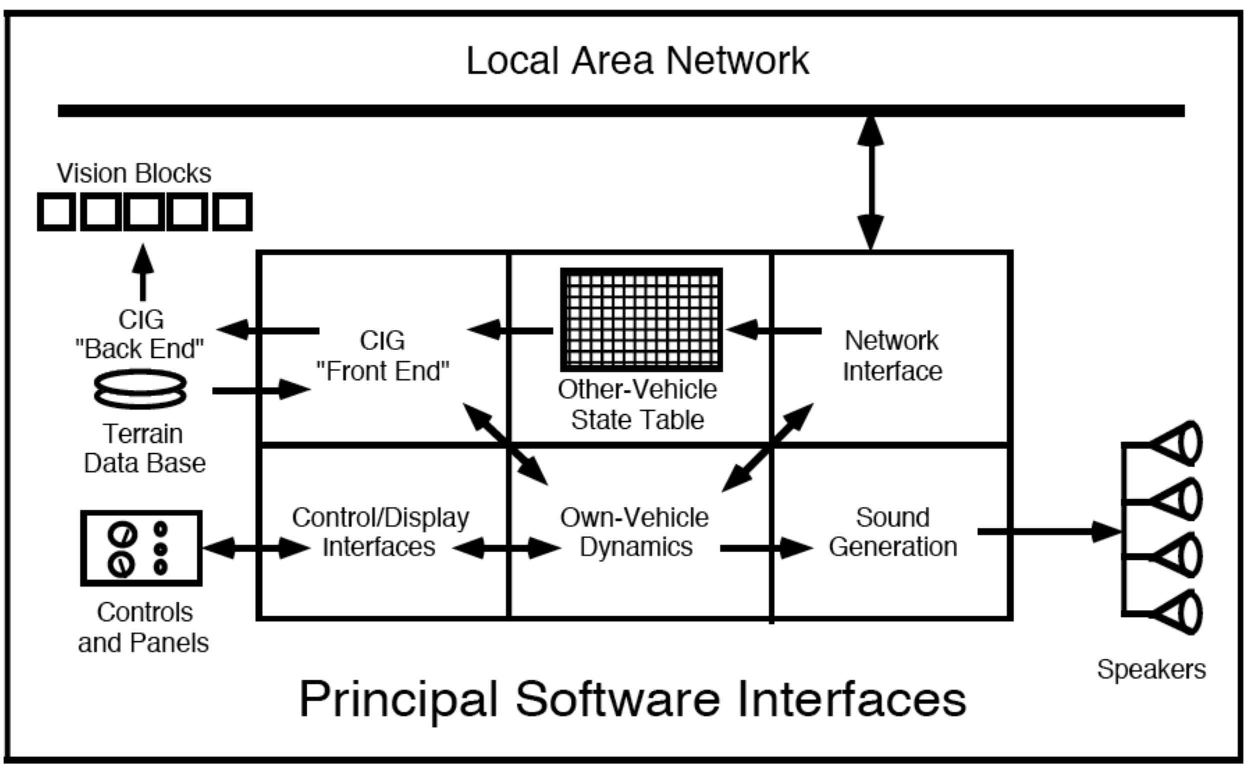

Figure 1: Principal software interfaces for a typical simulator

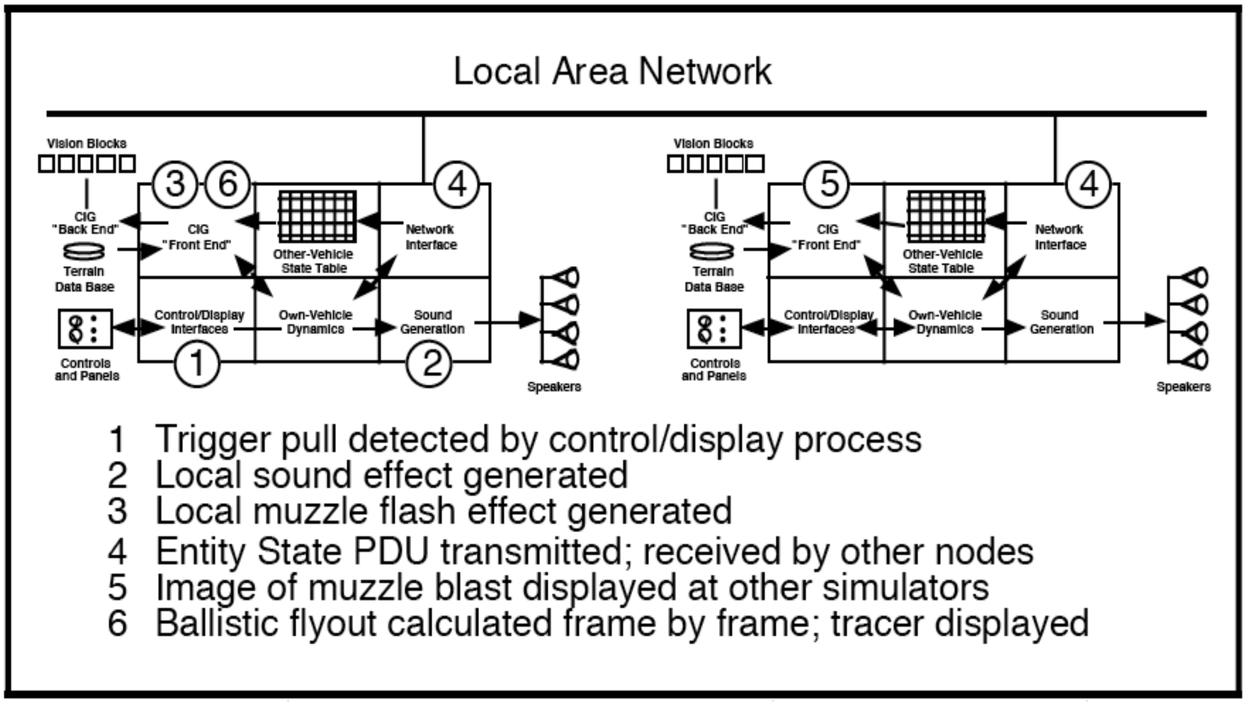

Figure 2: Sequence of events when Simulator A (on left) fires at Simulator B (on right)

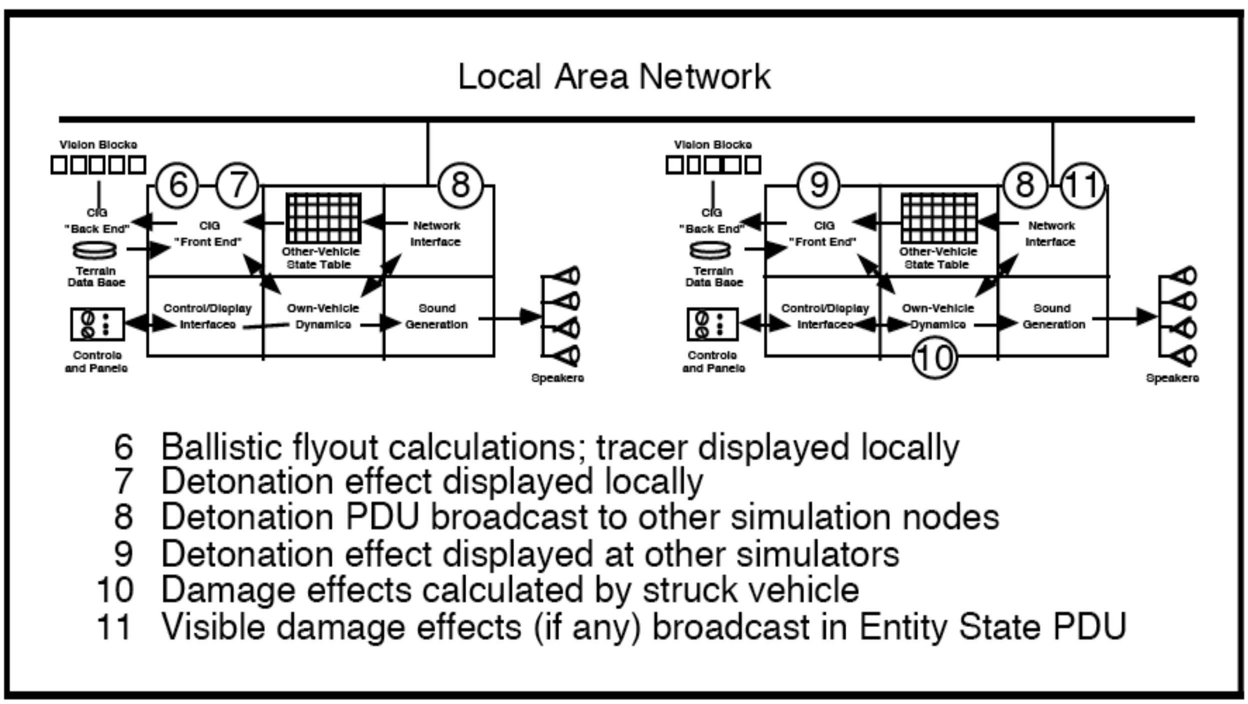

Figure 3: Sequence of events, continued

A final key design principle was that each simulator's controls and displays should include only the most important ones for tactical training. This approach was called "selective fidelity," sometimes loosely termed "the 60% solution." A frequently cited example of this principle: we didn't include a functional bilge-pump switch on a tank simulator, or model its operation, because doing so added no value for platoon- and company-level tactical operations. There was no firm criterion for making these selective fidelity decisions. Indeed, after a new type of simulator was used in serious exercises by real troops, their comments were often used to revise decisions regarding which system controls and dynamics were important to include, and at what level of detail. An example of this was the decision to induce "jiggle" in the vision block views and a "rumble" in the crew's seats to give them clues as to how fast they were moving and what kind of terrain they were traversing.

How SIMNET Began

In 1978, then-Captain Jack A. Thorpe was at the Air Force Office of Scientific Research, where he authored a white paper, "Future Views: Aircrew Training 1980-2000" that proposed a network of simulators for combat mission planning, rehearsal, and execution. Although the ARPANET existed and was expanding rapidly at that point, no one had yet considered including communications among simulators. The Air Force, for reasons that Jack outlines in his paper, wasn't really interested. However, they did assign him to DARPA, where the idea took root. In April 1983, DARPA awarded contracts for the development of "Large Scale Gaming" to Bolt, Beranek, and Newman (BBN) of Cambridge, MA and Perceptronics of Woodland Hills, CA.

Coincidently, COL Gary W. Bloedorn had recently retired as Director of Training Development at the US Army Armor School, Ft. Knox, KY. He, too, had envisioned the possibility of networking tank simulators for the purpose of crew training. There seemed to be no way to achieve this goal. However, MG Frederic J. Brown, the Commanding Officer at Ft. Knox, remembered the concept, and when he later heard Thorpe discussing it at a meeting (another coincidence), he tracked down Bloedorn and got the two of them together. Soon thereafter, Bloedorn became the guru who would teach a bunch of young "computer geek" software engineers enough about heavy armored vehicles that we could begin simulating them. Gary also worked with Perceptronics and a gifted industrial designer, Ulf Helgesson, to figure out how to use the "selective fidelity" principle (a term originated by Bob Jacobs of Perceptronics) to build low-cost crew compartments and controls for the simulators.

In a third key contribution, Mike Cyrus and others at the Boeing Corporation developed innovative algorithms to allow Computer Image Generators (CIGs) to efficiently process large numbers of overlapping, textured polygons to be viewed from arbitrary viewpoints in an environment in which the viewpoint may be moving among and through terrain, fixed objects (buildings, tree lines), moving objects (other vehicles, both friendly and enemy), and dynamic or transitory effects (explosions, dust and smoke clouds). The view presented on each display would have to be computed from scratch for each new frame. Achieving a minimum of 15 frames per second per channel in a cost-effective manner under these conditions demanded multiple innovations and efficiency improvements of at least an order of magnitude over image generators then in use. Boeing management declined to pursue this idea, however, so Mike Cyrus, Jay Beck, and Drew Johnston left Boeing and formed a start-up called Delta Graphics.

BBN and Perceptronics worked out an interface design that cleanly separated the internal crew compartment electronics (developed by Perceptronics) from the vehicle dynamics, weapon system simulation, network interface, and CIG interface software (developed by BBN). Delta Graphics designed and produced the first real-time, multi-channel hybrid z-buffer graphics system. It was initially a subcontractor to Perceptronics, and was later acquired by BBN.

How SIMNET Was Funded

DARPA's charter involves undertaking high-risk, high-payoff programs that, if successful, promise revolutionary results. These are programs that would be very difficult to justify and fund through normal DoD procurement processes, which emphasize clearly defined requirements, standardized funding and contractual mechanisms, and a low tolerance for failure or cost and schedule overruns. Such programs are almost as difficult to justify and fund through the DoD science and technology laboratories, for similar reasons. Yet DARPA competes for annual funding with these same programs and laboratories, so DARPA rarely has the resources to undertake a major program without the collaboration of the Military Services, their laboratories, and Congress, which provides funding to all of these agencies. Thus, cooperation is essential. An existing program may have a problem they are unable to solve, or a DoD Laboratory may have a challenge that they don't have the resources to address. In other words, established interests must have fairly powerful incentives in order to move precious budget resources to underwrite a new DARPA effort.

In the case of SIMNET, I witnessed a key step in its initiation. After Thorpe and I had given several presentations about the potential capabilities of SIMNET, we were invited to the office of the Undersecretary of the Army, James R. Ambrose, on 22 February 1985. His office at the Pentagon was the largest office I had seen at that point in my life. Only the three of us were present. As I presented the briefing, Ambrose, a grandfatherly figure in a rumpled sweater, was sitting at his desk eating his lunch: a peanut butter sandwich and a carton of chocolate milk. At the conclusion of our briefing, Ambrose asked several questions and then said to Thorpe, "I'm going to see that you get the funds to do this. Do you know why?" "Why, sir?" Thorpe responded. "Because if you can do what you're telling me," Ambrose said, "it will change the way the Army manages its weapon systems procurement."

So SIMNET moved forward, through a series of demonstrations, experiments, and tests that persuaded other key Army leaders that SIMNET was indeed capable of doing what we said it could. Most of these activities are highlighted below in the section entitled "SIMNET Tests and Demonstrations." Many of these events included, or resulted in, additional funding, through Military Internal Procurement Requests (MIPRs) and/or direct contracts. One of the most important of these was a live demonstration in the chamber of the Senate Armed Services Committee in May 1992. But first, an overview of the key SIMNET components will help to define the system and to set the stage for the descriptions that follow.

SIMNET Components

As noted in the Overview section, the most obvious components of SIMNET are the simulators themselves, but several other software applications are also required, each involving substantial software development efforts with tens or hundreds of thousands of lines of code. These "support components," all of which are connected via the Local Area Network (LAN), are briefly described below. The present tense is used in these descriptions because all of these components, as well as individual vehicle simulators (many of which have subsequently been updated), are still in daily use at multiple sites.

Management, Command and Control (MCC) system. Loads the selected terrain database (described in the "Terrain Databases" section), and initializes all simulators at appropriate locations on the terrain. Simulates artillery, air support, supply, and repair functions. Can function as a Tactical Operations Center (TOC) or Army Logistics Operations Center (ALOC). Each MCC typically supports one battalion's worth of assets. Typically uses one computer and multiple monitors, organized according the different functions being provided in the exercise. Multiple MCC systems can be used to permit multiple exercises to take place simultaneously on the same network.

Network Operations and Management (NOM) system. Allows simulators to be initiated en masse or individually. Provides an overview of the status of all nodes on the network and all simulators at a site. Alerts operators to network errors and other anomalous events, and provides the ability to send test packets to diagnose and debug problems. Typically uses one computer with one or more monitors.

Data Collection and Analysis System. Provides After-Action Review (AAR) and data analysis functions. Includes the following components:

Plan View Display (PVD). Provides a variable-resolution map display of the battlefield, intervisibility computations (i.e., which entities can see which others), plus detailed information on the status and location of all simulated vehicles. Typically uses one computer with one or more large display screens.

Data Logger (DL). Records all data packet traffic on the network and can play packets back onto the network, so that a simulator, an AAR vehicle, or the PVD, shows exactly what was happening at any point in time and space. Packets can be played back in real time, faster, or slower. The DLs in use circa 1990 could record approximately 900 vehicle-hours on one 500 MB disk. Typically uses one computer with one or more large disk drive, and (optionally) one or more recorders for voice traffic.

Voice Logger (VL). A 16-channel recorder for recording analog voice communications among vehicles on up to 16 radio frequencies. Synchronizes with the Data Logger, so radio communications can be compared with the recorded events. The VLs in use circa 1990 could play back any two selected channels at a time, and could record approximately 8 hours of voice communications on a standard tape cassette. Later, digital voice logging was added.

Semi-Automated Forces (SAF). A system of manned workstations representing the decision-making command post in the SAF organization, which allows a few controllers to project substantial enemy forces onto the battlefield and/or to provide friendly flanking and supporting units. Four experienced controllers can manage a Soviet Motorized Rifle Regiment (typically including 120 T-72 tanks and 330 Armored Personnel Carriers and Infantry Fighting Vehicles). Vehicle movements in subordinate units are controlled by the SAF software. Subordinate units receive orders and semi-autonomously plan their actions and maneuvers, reporting information back to their commanding units as they would in a real operation. Subordinate units behave sufficiently realistically that the SAF commander does not need to constantly micro-manage them to produce credible military operational behaviors. SAF includes one or more computers and one or more color map displays showing the current state of the battlefield as reported by the SAF units. SAF software was extended to support fixed- and rotary-wing aircraft, artillery units, and logistics trains for fuel and ammunition.

Stealth/After-Action Review (AAR) vehicle. Includes a simple set of controls for navigating around the battlefield. Each vehicle includes out-the-window and map displays that allow the operator an “omniscient” perspective, superior to that of any real-world vehicle. There were three versions in use circa 1990. Other variations have been added since then.

The **Stealth-1** included a high-capacity image generator with a single large-screen display, a Data Logger, a Plan View Display, and an analog radio receiver.

The **Stealth-3** contained a standard image generator (which was sometimes borrowed from an existing on-site simulator) and three low-resolution monitors, a Data Logger, a Plan View Display, and an analog radio receiver.

The **Stealth-8** was similar to the Stealth-3, but with 8 low-resolution monitors, which could be arranged in any desired pattern. Other components are the same as the other versions.

These vehicles can be employed in various ways, both during and after an exercise. For example, a Stealth vehicle (which was initially called a "flying carpet") sends no data packets and hence cannot be seen by any other vehicle. It can be flown freely, driven over the ground, or tethered to another vehicle, either occupying the same location or following it at a fixed distance. There is also a "water-skiing" mode that allows the Stealth vehicle to swing around the vehicle to which it is tethered to observe it and its surroundings from different perspectives.

When replaying a pre-recorded exercise, the Stealth vehicle can travel in time as well as space; i.e., it can jump backward or forward to an earlier or later time in the exercise. This is particularly useful in providing after-action reviews to the troops to discuss what occurred at significant points.

Generic Vehicle (GV). By modifying the controls and vehicle dynamics, and by broadcasting appropriate vehicle appearance packets, a Generic Vehicle can be used to represent any existing or hypothetical vehicle. Examples include a First Sergeant's vehicle (e.g., a HMMWV), a logistics vehicle (e.g., a HEMTT, an Air Liaison Officer's or Forward Air Controller's vehicle, a maintenance vehicle (e.g., an M88 recovery vehicle), an M113 (Armored Personnel Carrier) or M577 (Command Post Carrier), or almost anything else within imagination. No doubt additional applications have since been added.

Long-Haul Network (LHN) interface. Aggregates packets from each Local Area Network (LAN) and transmits them to remote sites. Includes a gateway computer with links to wire or fiber-optic cables. Circa 1990, two 56-kbps telephone lines were used for each site being connected. Dial-up lines were sufficient to accommodate 100 simulated vehicles per site.

Long-Haul Voice (LHV) communications. Requires an additional 56-kpbs line. Circa 1990, control signals were sent via the phone lines to indicate when transmissions began and ended, because the analog radio voice activation protocol then in use would not work over analog circuits. Subsequently converted to digital voice communications for better voice fidelity and to interoperate with the Electronic Model of the Battlefield (EMB) component.

Electronic Model of the Battlefield (EMB). Models the interaction of radio transmissions with the terrain and other electronic communications. Voice transmissions are digitized at the transmitter and compressed to reduce network bandwidth requirements. Header information for each transmission includes the transmitter location, frequency, emitted power, and antenna orientation. The EMB computes the received signal strength for each source and receiver based on the path length between them, including diffraction effects across major obstacles such as hills or mountains. A receiver model determines which of the competing signals are captured (FM radios, for example, lock in on the strongest signal received). The "winning" signal is then decoded and mixed with the appropriate level of noise. This signal could be a jamming signal (intentionally or unintentionally). The results of these computations (using the Longley–Rice model) are communicated to the various radio receivers, which then convert digital signals to analog form to be fed into the headsets of the crews. Transmission data may also be used by other simulators for determining source location, either for navigational purposes or for anti-radiation weapons (such as signal-seeking missiles).

SIMNET Combat Vehicle Simulators

All simulation vehicles include a crew compartment with various controls and displays, vehicle dynamics software that models the vehicle's dynamics and kinematics in response to control inputs and the terrain over which it is moving, the state of its hydraulic, electrical and fuel systems, and an image generation system that computes and displays what is visible "out the window" for each crew position. In addition, other software components model the ballistic trajectories of whatever weapons the vehicle fires, as well as its ammunition and fuel supplies, and the damage (if any) caused by a munition striking the vehicle. Damage resulting from such strikes, as well as deterministic and stochastic system failures and the results of any repair attempts are also modeled, using probability tables loaded at the time the simulator is initialized. Although the initial simulators represented ground vehicles, additional simulators were soon added to represent rotary-wing and fixed wing aircraft, and eventually dismounted infantry troops.

The interface to each vehicle's controls and displays occurs via an Interactive Device Controller (IDC) board, which constitutes the basic line of separation between inputs internal to the vehicle (e.g., control movements, switch activations, weapon firings) and those driven by external events. The IDC board also controls the various indicator lights and gauges visible at each crew position. An analog sound system, also local to each vehicle, generates the appropriate sounds of its own operations, its own weapons and those of nearby vehicles, nearby munition detonations, and communications heard over the vehicle's radio channels.

The Computer Image Generator (CIG) interface provides information to the CIG regarding the vehicle's current location and orientation on the terrain database — a minimum of six degrees of freedom, plus any additional data resulting from the rotation of a turret, the orientation and elevation of a gunsight, or any other independently movable vision devices. This information and the information received over the simulation network regarding the position and orientation of other vehicles is sent to the CIG, which, in combination with locally stored terrain information, generates the appropriate image for each viewport. The CIG's locally stored database also includes multiple versions of each type of vehicle being represented in the simulation (e.g., different resolutions at near, medium, and far distances, in operational, damaged, and destroyed conditions) as well as various dynamic effects (e.g., muzzle blasts, missile firings, burning vehicles, shell bursts, dust clouds from moving vehicles, etc.). These, too, are loaded as part of the exercise initialization process.

Terrain database generation is a technical and artistic challenge, in which database designers must abstract the most essential objects and features for a tactical scenario, while keeping in mind the computational limitations of the CIG's processors. In the 1980s, these limitations were quite stringent, especially when one considers that the simulators (and their accompanying dynamic effects) are constantly moving through the environment, so that the designers can never be certain how many objects might have to be processed in a worst-case situation. These constraints are still a consideration in today's database and CIG designs; as computational capabilities have increased, so have the requirements for complexity. A later section of this paper will focus on the processing algorithms, design tools, and "fail soft" techniques that were developed to facilitate the process and maximize the productivity of database generation.

M1 Abrams Main Battle Tank. The M1 has been the Army's primary heavy combat vehicle since entering service in 1980. It carries a crew of four: the driver, who reclines in a very tight space inside the forward hull compartment, plus the tank commander, the loader, and the gunner, who sit inside the turret compartment. The driver's orientation is fixed, but the turret can rotate 360 degrees (quite rapidly), so the turret's orientation can be independent of which direction the tank is moving. The tank commander's role is to control and coordinate the tank's overall operation, as well as to coordinate tactically with other commanders. He tells the driver where to go, designates targets for the gunner, and tells the loader what type of ammunition to pull from the rack behind the blast-proof doors and load into the breech of the 105 mm or 120 mm cannon. The crew communicates with each other over a local intercom, and the commander communicates over a multi-channel radio.

Similar to the real vehicle, the M1 simulator has two separate compartments: one for the driver and one for the other three crew members, each with the most essential controls and displays (chosen in accordance with the "selective fidelity" principles previously noted). The commander sits above the gunner, with the loader located to their left between the ammunition rack and the gun breech. A full description of the simulator crew compartments and controls from the SIMNET era can be found in (Chung, 1988). The driver has three "vision block" displays, the gunner has a stabilized gunsight with high/low magnification, the commander has three vision blocks plus an auxiliary sight that duplicates the gunner's sight, and the loader has a periscope display. Both the commander's and loader's cupolas can rotate to simulate additional vision blocks in a closed-hatch M1.

M2/M3 Bradley Fighting Vehicle. There are several Bradley variants, including the M2 Bradley infantry fighting vehicle and the M3 Bradley cavalry fighting vehicle. The M2 holds a crew of three: a commander, a gunner and a driver, as well as space for six fully equipped soldiers. The M3 mainly conducts scout missions and carries two scouts in addition to the regular crew of three. The M2/M3's armament includes a 25 mm cannon and a coaxial 7.62 mm medium machine gun, as well as twin missile launchers for TOW (Tube-launched, Optically tracked, Wired-guided) antitank missiles.

The M2/M3 simulator has three stations, for the driver, the gunner, and the commander. The driver's controls include a steering yoke, gear selector, accelerator, and brake, along with four "vision block" displays. The gunner's station has a control handle, palm switches and trigger, and an integrated sight unit (ISU) with high/low magnification. The sight unit display includes reticles for the 25-mm cannon and the TOW missile launcher. The commander's station includes controls for rotating the turret and for aiming and firing both the 25-mm cannon and the TOW missile launcher. The turret includes three "vision block" displays and an ISU identical to the gunner's display. (Yoo, 1985), (Bloedorn, 1985)

Rotary-Wing Aircraft (RWA). Generic attack/scout helicopter, approximating the AH-64 or OH-58. Includes pilot and copilot/gunner stations with flight controls and displays. Includes a slewable electrical-optical sensor for use in target acquisition and engagement. It is armed with HELLFIRE laser-guided anti-tank missiles and air-to-air STINGER missiles, in addition to a 30-mm chain gun. The pilot and co-pilot share visual access to eight out-the-window displays, arranged in a "three over five" configuration, plus a single sensor channel, switchable between a thermal image and a daylight TV image. The RWA uses typical helicopter cyclic and collective controls, along with a throttle, for flight maneuvers. (Harrison, 1992)

Fixed-Wing Aircraft (FWA) Close Air Support (CAS). Can be reconfigured to approximate an A-10 Thunderbolt II ("Warthog") or a basic F-16 "Fighting Falcon". The crew compartment accommodates a single pilot. The visual system is similar to that of the RWA, with a "three over five" display arrangement. The primary controls are a side-stick and throttle rather than the cyclic and collective controls used in the RWA. (Harrison, 1992)

The SIMNET Protocol

Information is exchanged across the SIMNET Local Area Network (LAN) and Long-Haul Network (LHN) using a connectionless data transfer service capable of both point-to-point and broadcast delivery. Data transfer occurs via a set of Protocol Data Units (PDUs). The SIMNET protocol does not assume that the network is fully reliable. It is designed to be robust despite occasional transmission errors. Some PDUs require acknowledgement; if none is received, the PDU is retransmitted. Other PDUs represent minor updates of the state of a simulated entity; if one of these PDUs is corrupted or lost, the next update will correct any discrepancy that has occurred.

The primary PDUs are listed below, along with a brief summary of their purpose and what data they contain. It should be noted that the DIS Protocols (IEEE 1278.1), while based on the SIMNET protocols, reorganized and renamed many of the PDUs. The list below uses the original SIMNET terminology. The "SIMNET Components" section above defines the various abbreviations used in these descriptions.

- **Activate Request.** Transmitted by the MCC System to initialize a simulator in an exercise. Superseded by the DIS **Create Entity** PDU. Assigns a unique ID to the simulator and designates the ID of the exercise in which it is participating. This permits multiple simultaneous exercises to take place on the same network.
- **Activate Response.** Transmitted by the designated simulator to confirm receipt of the activate request and to confirm that it is joining the designated exercise. Superseded by the DIS **Acknowledge** PDU, which combines the functions of responses to management directives.
- **Deactivate Request.** Transmitted by the MCC System to remove an entity from the designated exercise. Superseded by the DIS **Remove Entity** PDU.
- **Deactivate Response**. Transmitted by the designated entity to confirm receipt of the deactivate request and to confirm that it is leaving the designated exercise. This function is now incorporated in the DIS **Acknowledge** PDU.
- **Vehicle Appearance.** Transmitted by each simulator to update its state (e.g., its location, appearance, operational status, orientation of all articulated components, etc.) as required by the Dead Reckoning algorithm being used. Superseded by the DIS **Entity State** PDU**.**
- **Radiate**. Transmitted by each simulator whenever it is emitting detectable electromagnetic radiation. Superseded by the DIS **Electromagnetic Emission** PDU.
- **Fire.** Transmitted by each entity to report the firing of projectile, and what type of munition it is. Superseded by the DIS **Fire** PDU.
- **Impact**. Transmitted by each entity to report the impact of projectile it has fired, the type of munition it is, and the target, if known. Superseded by the DIS **Detonation** PDU.
- **Indirect Fire.** Transmitted by the firing entity (e.g., artillery) to report impact of indirect fire and the type of munition. Incorporated into the DIS **Impact** PDU.
- **Collision.** Transmitted by a simulator to report a collision with another simulated entity or with a fixed object in the database. Superseded by the DIS **Collision** PDU.
- **Service Request.** Transmitted by a simulator to request resupply of fuel or munitions. Superseded by the DIS **Service Request** PDU.
- **Resupply Offer.** Transmitted by a logistics vehicle simulator (e.g., a HEMTT) offering to resupply fuel or munitions to the requesting simulator. Superseded by the DIS **Resupply Offer** PDU.
- **Resupply Received.** Transmitted by a simulator to acknowledge receipt of some or all of

offered fuel or munitions. Superseded by the DIS **Resupply Received** PDU.

- **Resupply Cancel.** Transmitted by a simulator to abort transfer of fuel or munitions. Superseded by the DIS **Resupply Cancel** PDU.
- **Repair Request.** Transmitted by a simulator to request a repair. Incorporated into the DIS **Service Request** PDU.
- **Repair Response.** Transmitted by a maintenance vehicle (e.g., an M88 recovery vehicle) to report the completion of a repair. Superseded by the DIS **Repair Complete** PDU.
- **Minefield.** Transmitted by the MCC System to describe the parameters of an emplaced minefield. Incorporated into the DIS (1998) family of **Minefield** PDUs.
- **Breached Lane.** Transmitted by the MCC System to describe a path cleared through a minefield. Incorporated into the DIS (1998) family of **Minefield** PDUs.
- **Marker.** Transmitted by the MCC System to announce the marking of minefield with flags. Incorporated into the DIS (1998) family of **Minefield** PDUs.

Ten years after the SIMNET contract began, a broad consortium of simulation industry experts codified and reorganized the original SIMNET protocols in the IEEE 1278.1-1993 standard. This standard was revised and extended again in 1995, 1998, and most recently in 2012. The brief summary above, which does not attempt to address the details of these revisions, should make it clear that the international standards in use today are based firmly on the foundation laid by the SIMNET program.

SIMNET Image Generators

The SIMNET Computer Image Generator (CIG) design involved an exchange of updates between the CIG and the simulation host computer. Under this protocol, the host computer provides information about the current position and orientation of the simulated entity and its current motion. The CIG provides the host computer information regarding the characteristics of the local terrain and collisions with fixed or moving objects. The CIG then begins computing the out-the-window and/or sensor images for each of the vehicle's display screens. This involves retrieving local terrain database information as well as information about nearby fixed and moving objects, and processing these for display via the graphics pipeline. This pipeline includes:

- Field of view processing: elimination of objects that are not within the rectangular boundary being displayed representing the "pyramid of vision" for a particular display.
- Viewpoint transformation: computation of the relative angles of the terrain and object polygons to be displayed as seen from the display screen.
- Clipping: elimination of those portions of polygons that fall outside the pyramid of vision and hence do not need to be processed.
- Backface elimination: elimination of polygons that are facing away from the point of view of the display and hence do not need to be processed.
- Face shading: computation of the relative brightness of each visible polygon, depending on the angle and color of the incident light.
- Perspective projection: computation of the perceived shape of each visible polygon as projected in display screen coordinates.
- Polygon tiling: filling the area of each polygon using the appropriate texture map. This includes transforming the texture pattern to match the perspective projection.
- Hidden surface elimination: processing polygons, beginning with those closest to the display screen moving backwards (into the scene) and calculating the relative distance to each polygon on a pixel-by-pixel basis. This procedure uses depth buffering, also called the

Z-buffer, to discard information about any point in a polygon that is obscured by a closer polygon.

Tactically relevant ground-based scenes inherently involve high degrees of complexity, including the terrain itself plus static and dynamic models. These elements had to be modeled as textured polygons, often exceeding 3000 polygons per sq. km., which represented a major challenge in terms the processing power available in the late 1980s. For comparison, aviation simulations, operating at higher altitudes, might include only 500 polygons per sq. km. for the same terrain. In graphics terms, the issue is represented in terms of "depth complexity." A depth complexity of 4 would mean that for each pixel on the display, a ray drawn from the relevant eye point to the horizon would pass through an average of 4 polygons – for example, a nearby tree, then a tank, then a building, then terrain behind the building. Ground-based simulations might require a "depth complexity" of 2 - 8, as opposed to only 1 - 2 for aviation simulations. (Bess, 1992)

Furthermore, the depth complexity can vary substantially from one display to another on the same vehicle if the displays are representing different directions. In a "hard pipelined" CIG, one display may be severely overloaded while another is almost idle. Overload management techniques are essential. The most obvious such technique is to slow the frame rate of the display, but any reduction below a computation rate of approximately 15 frames per second quickly becomes quite noticeable. Moreover, if intense combat activity is occurring on a particular display, this is where the crew's attention is almost certainly focused, and where it is most important to preserve the frame rate.

An alternative is to relax the level of detail being displayed. It may be feasible to fall back to lower-resolution forms for vehicles and fixed objects, but it is critical to maintain the resolution of the terrain itself. This is because going to a lower terrain resolution might cause a vehicle that was previously hidden by the terrain to become visible. Even if a hidden vehicle only briefly "flashes" into view, this can seriously compromise the integrity of the simulation. Another alternative is to relax the processing of texture mapping of polygons. While texture is important to add perceptual realism to a scene, using simpler images during periods of peak loading is preferable to allowing overloads to create anomalies that could create situations similar to using lower terrain resolution.

It is also important to maintain close correlation between the computations used for intervisibility and those used for determining projectile flyout trajectories. A vehicle hidden behind a building or an earthen berm may not be visible to an opponent, but if a projectile fired at the building or berm instead passes through it and strikes the hidden vehicle, another unacceptable anomaly has occurred.

More sophisticated techniques are also possible. For example, using head tracking or eye tracking may permit a simulation to determine where a crew member is focusing his attention, and more processing resources can be diverted to that channel and away from other displays. Obviously, this requires more apparatus and greater expense, which was beyond the scope of what was possible within the constraints of SIMNET.

The bottom line concerning image generators is that there are many tradeoffs involved in meeting specified performance criteria while meeting specified cost targets. There will always be situations that can overload the processing capability of the CIG. Graphics designers must strive to minimize the circumstances under which these situations can occur, and must provide fallback or "fail soft" techniques to minimize the consequences when they do. On the other side of the ledger, users and their procuring agencies must recognize that avoiding any possibility of an overload can be achieved only at substantial cost and by adding processing capacity that goes unused a large percentage of the time. Jim Shiflett commissioned a videotape to explain these tradeoffs briefly but clearly.[1]

SIMNET Databases

Tools and infrastructure. A key SIMNET principle was the use of pre-distributed terrain data and models so that all simulators included the same features, attributes, and models in the same relative locations. (This principle has been continued in the DIS and HLA standards, discussed below.) An extensive set of graphical modeling tools (collectively called S1000) was developed to facilitate the creation of realistic terrain and vehicle model databases. Databases up to 50 km by 75 km were generated, which represented significant accomplishments at that time. Starting from Defense Mapping Agency (now National Geospatial-Intelligence Agency) data, these tools enabled graphic designers to abstract the most essential elements of the terrain as a textured polygonal surface with key fixed objects (buildings, tree lines, etc.) in their proper locations.

As automated tools were improved, databases could be created with labor requirements of about 1 person-hour per square kilometer. Some databases (such as the National Training Center at Ft. Irwin, CA) required a 20-km radius "curtain" displaying key navigational points such as distinctive mountain peaks that were used by vehicle crews for orientation and navigation. In the early years of SIMNET, the goal was to maintain a scene computation rate of 15 frames per second at an average depth complexity of 3.8. This performance level was adequate most of the time, but in complex exercises such as the SAF POP, FAADS/ADATS, and WAREX 3-90, some of which included hundreds of vehicles, overloads and "flashing" did occur under certain conditions.

By 1989, database development included not only the visual display information used by the manned vehicle simulators, but also detailed terrain information (e.g., roads, rivers, forests, soil types) needed by the Semi-Automated Forces and the Plan View Displays, "bald earth" terrain needed for the Electronic Model of the Battlefield, and terrain features and objects for the Management, Command and Control System.

To share SIMNET databases with other organizations, BBN developed the SIMNET Database Interchange Specification (SDIS) and made the Hunter Liggett database available in this format. (Wever, 1989), (Wever, 1990) Unfortunately, SDIS was perceived as an attempt to compete with the U.S. Air Force Project 2851 (P2851) Generic Transformed Data Base (GTDB) format. Reaction to SDIS led to the development of Project 2851's Standard Simulator Database (SSDB) Interchange Format (SIF), which, similar to GTDB, focused on visual system needs and did not address networked simulation, SAF, or other non-visual needs. The SEDRIS standards and implementations (discussed below under "Subsequent Developments") were the outcome of these experiences and addressed the broader problem of representing and interchanging environmental data for a wide variety of applications. However, the fact that different systems, such as ground- and air-focused simulations, have significantly different needs (as noted in the "Image Generators" section), hints at the complexity of the problems and the associated solutions.

Numerous databases were created and used in SIMNET demonstrations and tests. The primary databases included the following:

---

[1] www.youtube.com/watch?v=i0Hq54gU-g4.

Ft. Knox database. A 50 km by 50 km representation of the main training area at the Ft. Knox Armor School, extended to 75 km by 50 km in 1988. Installed at all SIMNET sites.

Hunter Liggett database. A 50 km by 50 km representation of the Fort Hunter Liggett developmental test and exercise area near Monterey, CA. Used for the FAADS/ADATS tests.

Fulda database. A 50 km by 50 km representation of the Fulda Gap region of Germany, considered to be a likely area of attack by Soviet forces during the Cold War era. Many wargames and planning exercises were centered on this area.

Bergen-Hohne database. A 25 km by 25 km database used primarily as a practice range for the 1989 Canadian Army Trophy Competition.

National Training Center (NTC) database. A 50 km by 50 km database representing the primary exercise area at Ft. Irwin, CA. Also included the panoramic "curtain" described at the beginning of this section. Starting in 1980, the Ft. Irwin NTC was home to a special unit, which served as the Opposing Force (OPFOR). OPFOR troops used tanks modified to look like Soviet T-72 tanks and BMP-1 fighting vehicles, employed Soviet tactical doctrine, and wore Soviet-style uniforms. The OPFOR conducted 15 operations per year against armored brigades from various US units. Many of these brigades were able to practice maneuvers on the NTC database before heading for Ft. Irwin, which significantly reduced the OPFOR's "home field advantage."

SIMNET Sites

SIMNET simulators, networks, components, and software were installed at seven principal sites that were initially funded via DARPA contracts and subsequently transferred to the US Army and supported by other contracts. Several other smaller, temporary, and demonstration sites were also established for a variety of reasons but, except for specific events described below, will not be further discussed here.

Principal SIMNET installations included the following. The items listed for each site are described in the "SIMNET Components" and "SIMNET Combat Vehicle Simulator" sections above.

Ft. Knox, KY. The US Army Armor School. As of mid-1990, the complement of simulators at the Ft. Knox training site (SIMNET-T) included 42 M1s, 16 M2s, 1 battalion-scale SAF system, 2 MCC systems, a NOM, and a Stealth-3. Each MCC system included a SIMNET Control Console, an Admin/Log Console, a Close Air Support Console, a Fire Support Console, a Maintenance Console, an LHN Bridge Console, and a Combat Engineering Workstation.

When the SIMNET contract transitioned from DARPA to the Army in 1990, the Ft. Knox Developmental site (SIMNET-D), which was renamed as the Mounted Warfare TestBed (MWTB), as well as the Ft. Rucker site, were retained under the DARPA contract to continue supporting the Army Research Institute CVCC and the DARPA CTAS experiments. See the "SIMNET Tests and Demonstrations" section for details. The complement of simulators and other components at these sites varied, as experimental simulators were assembled using modified crew compartments and components borrowed from other simulators, as well as simulators borrowed from other sites. As of mid-1990, the complement included 12 M1s, 2 M2/M3s, 1 FIST-V, 4 ADATS, 1 battalion-scale SAF system, 3 MCCs, 1 NOM, 1 Battalion Tactical Operations Center (Bn TOC), 2 DLs, 2 PVDs, 1 Stealth-1, and an LHN Gateway.

Ft. Rucker, AL. The US Army Aviation School for rotary-wing aircraft. This site was also retained under the DARPA contract to support continuations of the AIRNET experiments. The complement of simulators and other components at these sites varied, as experimental simulators were assembled using modified crew compartments and components borrowed from other simulators, as well as simulators borrowed from other sites. As of mid-1990, the complement included 2 M1s, 2 M2s, 8 RWAs, 2 FWAs, 1 battalion-scale SAF system, 2 MCCs, 1 NOM, 1 Bn TOC, 1 DL, 1 PVD, 1 Stealth-1, an LHN Gateway, and a Voice Gateway.

Ft. Benning, GA. The US Army Infantry School. As of mid-1990, the complement of simulators at Ft. Benning included 10 M1s, 13 M2s, 1 SAF system, an MCC, a NOM, and a Stealth-3. The MCC system included a SIMNET Control Console, an Admin/Log Console, a Close Air Support Console, a Fire Support Console, a Maintenance Console, an LHN Bridge Console, and a Combat Engineering Workstation.

Ft. Stewart, GA. Home of the 24th Infantry Division. As of mid-1990, the complement of simulators included 14 M1s, 9 M2s, 1 Company-level SAF system, an MCC, a NOM, and a Stealth-3. The MCC system included a SIMNET Control Console, an Admin/Log Console, a Close Air Support Console, a Fire Support Console, a Maintenance Console, an LHN Bridge Console, and a Combat Engineering Workstation.

PM TRADE. The Program Manager, Training Devices supported a small installation at the University of Central Florida Institute for Simulation and Training (UCF/IST). As of mid-1990, the complement of simulators included 2 M1s, 1 Company-level SAF system, an MCC, and 2 Stealth-3 systems. The MCC system included a SIMNET Control Console, a Close Air Support Console, a Fire Support Console, an LHN Bridge Console, and a Combat Engineering Workstation, and an LHN Gateway.

Camp McCain, MS. Home training facility of the Mississippi National Guard, covering 13,000 acres. Training at the facility includes tank maneuvers, artillery training and general training for National Guard troops. As of mid-1990, the complement of simulators included 4 M1s, 1 Company-level SAF system, an MCC system, and a Stealth-3. The MCC system included a SIMNET Control Console, an Admin/Log Console, a Close Air Support Console, a Fire Support Console, a Maintenance Console, an LHN Bridge Console, and a Combat Engineering Workstation.

Grafenwöhr, Germany. A major US Army training facility in Bavaria, Germany and the largest NATO training area in Europe. Now serves as the command and control headquarters of the US Army Europe (USAREUR), formerly the 7th Army. As of mid-1990, the complement of simulators included 32 M1s, 17 M2s, 1 Battalion-level SAF system, 2 MCC systems, a NOM, and a Stealth-3. Each MCC system included a SIMNET Control Console, an Admin/Log Console, a Close Air Support Console, a Fire Support Console, a Maintenance Console, an LHN Bridge Console, and a Combat Engineering Workstation.

Friedberg, Germany. Another USAREUR facility, consolidated with the Grafenwöhr facility and closed in 2007. (Trivia point: Elvis Presley was stationed here in the late 1950s.) As of mid-1990, the complement of simulators included 15 M1s, 19 M2s, 1 Company-level SAF system, an MCC, a NOM, and a Stealth-3. The MCC system included a SIMNET Control Console, an Admin/Log Console, a Close Air Support Console, a Fire Support Console, a Maintenance Console, an LHN Bridge Console, and a Combat Engineering Workstation.

Schweinfurt, Germany. Another Army garrison and training facility, closed in 2014. As of mid-1990, the complement of simulators included 14 M1s, 9 M2s, 1 Company-level SAF system, an MCC, a NOM, and a Stealth-3. The MCC system included a SIMNET Control Console, an Admin/Log Console, a Close Air Support Console, a Fire Support Console, a Maintenance Console, an LHN Bridge Console, and a Combat Engineering Workstation.

Fulda, Germany. This facility had the distinction of being the closest US installation to a major Soviet base, just across the East/West German border. Many wargames involved a scenario in which Soviet forces invaded West Germany through the "Fulda Gap." As of mid-1990, the complement of simulators at Fulda included 10 M1s, 12 M2s, 1 Company-level SAF system, an MCC, a NOM, and a Stealth-3. The MCC system included a SIMNET Control Console, an Admin/Log Console, a Close Air Support Console, a Fire Support Console, a Maintenance Console, an LHN Bridge Console, and a Combat Engineering Workstation

SIMNET Tests and Demonstrations

Initial concept demonstration. One of the most significant demonstrations of the potential of SIMNET took place before there was any operational vehicle simulator. In January of 1984, the SIMNET team set up a pre-recorded demonstration in unused office space in Rosslyn, VA, and invited senior Army and other DoD officials to observe it. This demonstration included full-scale plywood and fiberglass mockups of the MI tank driver and turret compartments, with video screens at each of the viewports. Prototype controls were in place, but they didn't control anything. Instead, those attending the demonstration could see and hear a repeated scenario of a tank platoon engaging a Soviet T-72 tank. They heard the radio communications as well as the internal intercom communications. They saw the T-72 move from concealment and open fire on the US tank. They heard the commander's orders to the driver to move into position, as well as the orders to the loader and gunner to return fire. They saw the T-72 destroyed. At the end of this demonstration, the SIMNET staff involved had heard the radio communications so many times that we were able to gather as a group and recite the entire script in unison.

This demonstration was preceded by a SIMNET briefing (along the lines of the "Key SIMNET Concepts" section above) and an explanation of the concept of a "shared virtual reality" in which dozens or hundreds of simulated vehicles could move and interact freely. The phrase "shared virtual reality" was not used, however, because that term did not yet exist. The demonstration was a real "eye-opener" to those who saw it. Some had a vague concept of what they would be seeing, but it was clear that many did not. The team was peppered with questions. In what turned out to be a powerful strategy, we had the young software developers on hand to respond to these questions. One three-star general said to me, "Usually at events like this, we get some 50-year-old ex-military guy explaining things to us, but it's obvious he has no real understanding of the technology. You've stood up the kids who are actually writing the code, and they can answer every question that comes up. That's brilliant!" I wish I could claim credit for foreseeing the effect this would have, but in reality, this was simply the corporate culture of BBN in action.

The demonstration became a "must see," especially after the Vice Chief of Staff of the Army, GEN Max Thurman, put out the word that all available senior Army staff should sign up if they were available. The demo, initially planned for two weeks, was extended to four. By the time we shut down, at least 25 flag officers had been briefed, along with key members of their staffs, as well as the Secretary and Under Secretary of the Army, four Assistant Secretaries, an Assistant Secretary of Defense and two Deputy Secretaries of Defense. Were our young briefers intimidated? Not a bit.

Two of them came up to me after a particularly full day of briefings to ask, “By the way, what do all those stars and eagles on their uniforms mean?”

Canadian Army Trophy Competition (CATC). A gunnery competition among NATO armored forces, this competition began in 1963 and took place at a range in Germany. The competition is no longer being conducted. It was extremely demanding, involving a platoon of four tanks firing on the move as targets popped up unexpectedly. The crews had to coordinate closely, pick their targets rapidly, and fire with great accuracy. Until 1987, no US team had ever won. The rules prohibited the competitors from practicing on the actual range, though they could practice elsewhere using other ranges, if available. Four of the first M1 simulators being delivered to Ft. Knox were instead sent to the USAEUR facility at Grafenwöhr, Germany, along with a terrain database representing the competition range. The US teams began using the simulators immediately, completing a run about every 15 minutes, with detailed after-action reviews of their scores and refinements of their engagement tactics before the next run. On the last run of the final day of the actual competition, 1st platoon, D Company, 4-8 Cavalry achieved the highest score and won the competition. They credited their practice sessions in the networked simulators as a key contribution to their victory (Kramer, 1987).

Forward Area Air Defense (FAADS). The first use of SIMNET for Developmental Tests occurred in March and April 1988 at Ft. Knox and Ft. Rucker. These tests were feasibility studies to determine whether SIMNET could be used for Force Development Test and Evaluation (FDT&E) and Initial Operational Test and Evaluation (IOT&E). A total of 164 soldiers and pilots from Ft. Bliss, Ft. Knox, Ft. Rucker, and the Army and Air National Guard participated. The initial tests included one week of train-up, one week of exercises, and one week of analysis. Data was collected both manually and using the SIMNET Data Collection and Analysis System. The users were satisfied that SIMNET could be used effectively to address combat development, tactical development, and training.

Subsequent FDT&E tests occurred in September 1989. Martin Marietta, the contractor for the Forward Area Air Defense Line-of-Sight-Heavy (FAADS LOS-F-H) contract used the SIMNET facilities at Ft. Knox for initial tests of the system they were developing. Armored units and Fixed-Wing Aircraft (FWA) at Ft. Knox engaged rotary-wing aircraft simulators at Ft. Rucker. Prototype FAADS air defense vehicles defended the armored forces against air attacks.

These FAADS simulators were assembled in the Ft. Knox SIMNET-D facility using modified components from M1 and M2 simulators. Five FAADS simulators were built, the fifth being the platoon commander’s vehicle. Technical descriptions of the vehicles and weapon systems were provided by the Martin Marietta Electronics and Missiles Group. (Lanpher, 1989)

In one of the first tests, soldiers from the Air Defense Artillery School at Ft. Sill were diverted en route to a field test at Ft. Hunter Liggett, CA, to practice their tactical concept of operations (CONOPS) in the simulators. The soldiers immediately complained that the controls and displays of the simulators did not correspond to those they had originally trained with. A call to the Martin Marietta engineers revealed that several changes had been made in the vehicles they would use in the field tests. The fact that this was discovered and the soldiers were able to learn the new system probably saved several wasted (and very expensive) days on the test range.

As the exercises proceeded, another major problem was discovered. The FAADS commander had intended to take advantage of the superior speed and maneuverability of the M1 tanks and Bradley

Fighting Vehicles to attack his targets, a ground force of T-72 Soviet tanks and Armored Personnel Carriers. The concept involved moving his platoon forward in pairs (a leap-frogging technique called "bounding overwatch," with one pair in fixed position while the second pair moved forward to their next position). It quickly became apparent that this tactic would significantly compromise the maneuver speed of the armored task force, by as much as a factor of two. Intense discussions resulted in a modified maneuver plan. The initial engagement by RED fixed- and rotary-wing aircraft resulted in a successful BLUE defense and the loss of several RED aircraft.

That evening, however, the RED force pilots met separately and discussed tactical modifications, developing innovative ways to defeat the FAADS. On the next day, these tactics prevailed, and several FAADS vehicles were destroyed along with BLUE task force tanks and fighting vehicles. The next night, the BLUE force crews modified their tactics and on the following day regained the upper hand. These developments continued throughout the week until a state of equilibrium was reached. The FAADS crews departed Ft. Knox for Ft. Hunter Liggett with a new portfolio of tactics and a new CONOPS. This process would almost certainly have required at least two multi-day sessions on the test range, so again, great cost savings resulted. (Thorpe, 2010)

Counter Target Acquisition System (CTAS). The FAADS system, now renamed the Air Defense Anti-Tank System (ADATS), faced another challenge. It required a certain minimum range to target and prosecute airborne threats. If a clever aircrew could get inside this range, they could attack with immunity. To counter this problem, DARPA (under program manager Dr. L. N. Durvasula) proposed that one of its battlefield laser projects could be used to augment ADATS to nullify the attacking aircraft's optical systems. This project was called the Counter Target Acquisition System (CTAS). Several alternative systems had been proposed, each requiring differing laser energy output and with significant limitations in the environments where they could be employed (e.g., dusty or wet conditions). To narrow down the set of candidate systems, DARPA, in cooperation with the US and UK Armies, conducted a series of tests at the SIMNET-D facility at Ft. Rucker in October and November 1990.

As in the FAADS tests, substantial evolution occurred in both offensive and defensive tactics with each alternative system. In the end, what had begun with five candidate laser systems was winnowed down to one, with another close behind. The Army was now able to begin a weapon system acquisition decision based on tactically realistic results – experience, not just equations and viewgraphs. (Thorpe, 2010)

Army Research Institute Combat Vehicle Command and Control (CVCC) Studies. From 1989 through 1991, U.S. Army Research Institute for the Behavioral and Social Sciences (ARI) field unit at Ft. Knox used the SIMNET-D facilities to test concepts for a set of Command and Control systems that were successfully integrated into the M1A2 upgrade to the M1 Abrams Main Battle Tank. For these tests, a company-level OPFOR was used, with the SAF vehicles programmed to either return fire or not return fire, as required by various tests. In addition, six SAF vehicles were programmed to fill out two BLUE force platoons, with three vehicles programmed to follow each of two manned M1 simulators controlled by the platoon leaders. (Lickteig, 1989)

Two versions of the CVCC were compared against the standard M1 configuration. The first, the Intra-Vehicular Command and Control (IVCC) system, included a Command and Control Display (CCD) for displaying the vehicle's own location and transmitting reports, plus a Commander's Independent Thermal Viewer (CITV) that allowed the tank commander to designate targets for the gunner. The second configuration, referred to as "full CVCC," included all capabilities of the IVCC condition as well as a mutual position navigation (POSNAV) capability displaying the location of all friendly vehicles and a radio interface unit that allowed reports to be sent and received digitally. CITV enhancements included a target stacking capability and an independent laser locator.

The CCD enabled the tank commander to create and modify routes for navigation and to send route information to his driver. In addition, the CVCC configuration permitted any tank commander to transmit a route electronically to other vehicles in his unit. The CITV sight allowed the commander to independently scan for targets in all weather conditions and through battlefield obscurants. Identified targets could be queued and sent automatically to the gunner without requiring multiple verbal interactions.

Simulations of these key systems were extensively tested in the Mounted Warfare TestBed before any actual systems were developed and installed in real vehicles. These tests identified and eliminated system deficiencies that would have been very expensive to correct once they had been implemented and deployed. (Atwood, 1991)

SAF POP. The first major "proof of principle" demonstrating that credible exercises could be run with a majority of SAF entities involved. This exercise took place from 18 - 24 March 1989. It involved soldiers from the 1st Infantry Division, Ft. Knox, troops from Ft. Riley, KS, Army aviators from Ft. Rucker, and Tactical Air Command pilots from the US Air Force, Myrtle Beach AFB and the Indiana Air Guard. Long-haul links between Ft. Knox, Ft. Rucker, Ft. Leavenworth, the BBN facilities in Cambridge, MA, and the SIMNET Washington office in Rosslyn, VA allowed both manned simulators and SAF to participate in a joint exercise. Staff from the Ft. Leavenworth Combined Arms Center and the Ft. Sill Artillery School were also on-site in Cambridge. Many additional observers were present, as well; in fact, the sheer number of people "observing" became a problem, interfering with the operational staff's ability to organize and control events and the technical staff's ability to identify and resolve problems that were discovered. Among the many lessons learned from this exercise (in addition to technical issues) were the need for better scheduling and orientation of the troops and much stricter control of visitors, observers, and VIPs during a complex exercise. Despite these shortcomings, the participants came away impressed with the capabilities and potential of SIMNET to provide realistic interactions among combined arms forces that rarely had had opportunities to interact in any other way. (Cushman, 1989)

WAREX 3-90. The first major exercise involving multiple facilities connected via the Long-Haul Network. About 850 simulated vehicles participated in the WAREX 3-90 exercise in March 1990, including both manned simulators and Semi-Automated Forces. This exercise included US Army and Air Force aviators, National Guardsmen from the states of Minnesota and Washington, experts from the U. S. Army's Intelligence School, Ft. Huachuca, AZ, as well as opposing forces from the Army National Simulation Center at Ft. Leavenworth, KS and major elements from the First Infantry Division, Ft. Riley, KS. As a result of WAREX 3-90, the Army proceeded to draft concept plans for Advanced Battle Simulation (ABS). (Brooks, 1990), (Cosby, 1990)

Non-Line-of-Sight Anti-Tank (NLOS). The Non-Line-Of-Sight Fiber-Optic Guided Missile (FOG-M) system was developed to counter the increasing threat posed by precision guided weapons delivered by rotary-wing aircraft masked behind terrain features. NLOS FOG-M used fiber-optic cable guided missiles to engage stationary and moving rotary-wing and ground targets masked by terrain or hidden from direct line-of-sight. The NLOS system concept included a launcher with six missiles, gunner's station, and land navigation system mounted on the high mobility multipurpose wheeled vehicle (HMMWV). NLOS was operated by a two-man crew. One crewmember acted as driver and performed radio communications tasks. The second crewmember served as gunner. The NLOS gunner used the gunner's station controls and displays located in the HMMWV crew cab to enter a missile flight path into the system computer prior to missile launch. The missile seeker was programmed to systematically search the target area until the gunner selected a target for engagement. After launch, the missile seeker provided a look-down view of the battlefield on the gunner's console display via the fiber-optic cable. Rotary-wing and ground targets were successfully engaged with missiles launched from the prototype systems in live-fire demonstrations, and the prototypes participated in numerous technical, operational, and simulation system tests between 1988 and 1991.

The NLOS/SIMNET evaluation, using manned NLOS simulators, was conducted in April 1991 at the Ft. Rucker, AL, SIMNET site by the Army Operational Test and Evaluation Command (OPTEC). (Sanders, 1994), (Schwartz, 1996)

Battle Force In-Port Trainer (BFIT). The first exercises incorporating a Naval simulation took place in December 1989 and April 1990. Long-haul links were established between the Fleet Combat Training Center Atlantic (FCTCLANT), Dam Neck, VA, Ft. Knox, and Ft. Rucker. Army and Marine Corps crews operated the tank simulators and helicopter and Fixed-Wing Aircraft simulators were flown by Marine Corps and Army pilots at Ft. Rucker. The AEGIS Training Center and the USS Wasp were also connected to the network. Over-the-Horizon Targeting (OTH-T) information was displayed on board the Wasp and on the FCTCLANT Tactical Flag Command Center's (TFCC) Flag Data Display System. That meant that a helicopter simulator at Ft. Rucker, flying over the simulated terrain at Ft. Hunter Liggett, could be viewed on the radar screens of the Wasp from its simulated position 50 miles off the California shore, while it was physically at the pier in Norfolk, VA. Voice communications circuits were provided for all participants.

The Navy's BFIT System had been developed in 1985 to allow a Battle Group to conduct simulated exercises aboard their own ships using the same battle stations that operators and decision makers would use during actual combat. BFIT exercises initially focused primarily on Anti-Air Warfare, with little involvement of shore-based forces. The BFIT/SIMNET Proof of Principle exercises demonstrated the feasibility of integrating these two major simulation systems. For the first time, armored units came under naval gunfire, and Rotary-Wing Aircraft took off from the deck of an aircraft carrier and flew inland to support armored units. (Tiernan, 1990)

Line-of-Sight Anti-Tank (LOSAT). LOSAT is a highly mobile, all-weather, day/night, direct-fire anti-armor weapon system capable of defeating threat forces at a range of several kilometers. A sophisticated fire control system achieves single-shot hit-to-kill accuracies with a hypervelocity kinetic-energy missile. The LOSAT system is hosted on the Armored Gun System (AGS) platform developed by United Defense, LP. Predecessor concepts used the Bradley Fighting Vehicle as the host vehicle.

The essential elements of LOSAT's Fire Control System (FCS) are a television sensor, a Forward Looking InfraRed (FLIR) sensor, a laser ranging and uplink device, several computers, and the controls and displays used by the soldier to control the system. The missile includes an on-board guidance processor that controls the flight trajectory to the target. During flight, guidance updates are sent via uplink laser to refresh the missile's knowledge of its position with respect to the target.

LOSAT tests were conducted at the SIMNET-D facility, Ft. Knox, between December 1990 and April 1992. (Davis, 1992), (Brown, 1995)

Odin. DARPA's Project Odin was initiated in 1991 under the leadership of CDR Dennis McBride. The purpose of this program was to provide an adjunct to the US Military battlefield command and control systems, drawing on several elements that had been developed and demonstrated in SIMNET. The idea was to provide commanders with the current threat status, showing estimated enemy locations and strength as entities on the terrain database using the SIMNET simulation. This separated enemy state representations from the data collection sources, which always require a different level of security classification. Using components of the SIMNET AAR system, commanders could "fly" through digital renderings of tactically and strategically important areas of interest to observe the terrain and the disposition of friendly and enemy forces. The "time travel" capabilities of the AAR allowed the commanders to see how the situation developed over time. At least conceptually, it allowed them to see projections of planned or anticipated movements of forces. Initial tests were conducted at Ft. Knox and Ft. Rucker in early 1991. The system was designed to be deployed in ruggedized half-size ISO containers mounted on military trucks, with power supplied by a towed generator. It was designed for use in Desert Shield and Desert Storm, though this deployment never actually occurred. By 1993, systems were being tested at CENTCOM HQ, McDill AFB, FL and at the Army Topographic Engineering Center (TEC), Ft. Belvoir, VA. (Seidel et al, 1993), (Thorpe, 2010)

73 Easting. Immediately after the Gulf War (Operation Desert Shield, Aug 1990 – Feb 1991, Operation Desert Storm, Jan 1991 – Feb 1991), extensive reviews were conducted of the key engagements. The Vice Chief of Staff of the Army, GEN Gordon Sullivan, was familiar with SIMNET from his earlier assignment at Ft. Knox. He suggested reconstructing a battle using SIMNET so that it could be played back and analyzed using its AAR system, studying the engagement from different points and perspectives. With the approval of the Corps Commander in Iraq and the DARPA Director, a team was dispatched to Iraq within days. Led by COL (R) Bloedorn, this team interviewed soldiers who fought the battle of "73 Easting" (which was named for its map coordinates). Forensic evidence was collected, including the TOW missile fiber-optic wires that showed where a missile had been fired and where it had struck. Fortuitously, one soldier had used a personal tape recorder which captured the radio traffic during the engagement. Using this information, PDUs were generated and time-stamped to create a Data Logger record as if the battle had been a SIMNET exercise. After six months of painstaking work, the reconstruction team met with the soldiers who had fought the battle, now back at their home stations. They met at the SIMNET facility in Grafenwöhr and played the reconstructed events. As expected, the soldiers provided dozens of corrections as well as supplemental information not previously captured. There were several debates among the troops regarding the exact sequence of events, but using the SIMNET AAR and the tape recording of the radio transmissions, most of these debates were quickly resolved. Unexpected revelations occurred as soldiers saw firefights occurring in adjacent units, of which they hadn't been aware during the heat of battle. A summary of the reconstruction was included in the Senate Armed Services Committee demonstration in May 1992. The summary was narrated by Captain H. R. McMaster (currently LTG McMaster), who commanded the Eagle Troop of the 2nd Armored Cavalry Regiment during the battle. (Thorpe, 2010)

The Senate Armed Services Committee live demo. In many ways parallel to the initial 1984 conceptual demonstration of shared virtual reality on the battlefield, this demonstration took place on 21 May 1992 in the main hearing room of the Senate Armed Services Committee. This demonstration included live long-haul links from Ft. Knox, KY and Ft. Rucker, AL, as well as fixed-wing and rotary-wing aircraft simulators physically located in the Senate hearing room itself. This demonstration was, in effect, showing the Senators the progress that had been achieved with the funds they had provided. It represented a milestone in confirming their confidence in the major simulation procurements that were then being initiated, particularly the Army's Close Combat Tactical Trainer (CCTT) and Aviation Combat Tactical Trainer (AVCATT) programs. [2]

I/ITSEC demo in San Antonio. At the 14th Interservice/Industry Training Systems and Education Conference in San Antonio, TX on 2-5 November 1992, another significant demonstration took place. This demonstration was sponsored by the Defense Modeling and Simulation Office (DMSO) and the US Army's Simulation, Training and Instrumentation Command (STRICOM) in order to show that SIMNET-originated capabilities could be employed using the DIS protocols and the DoD Project 2851 Standard Simulator database (SSDB) interchange format (SIF) in an organizationally heterogeneous setting. 28 organizations took part in a real-time simulation on the exhibit floor of this major trade show. It was the first time that many of these organizations had attempted to use the DIS Protocols, which were soon to be approved as IEEE Standard 1278-1993. A total of 18 simulators and one live vehicle produced PDUs that were transmitted on a network created for this demo. Another 22 devices in "listen only" mode received PDUs and displayed them in some form. The simulated terrain was a 100 by 100 km. section of Ft. Hunter Liggett, developed using the S1000 tools and converted to SIF. (Farsai, 1992) (Loper, 1993)

Part of the significance of this event was that many of the participants had remained skeptical of, and even resistant to, the SIMNET protocols on which DIS was based. The level of participation, however, showed how much attitudes had changed in the industry as major procurements were now requiring the use of DIS. Another aspect was the realization that ensuring proper interaction of the simulated vehicles with the simulated terrain was much more complex than many of the participants had expected. There were multiple instances of ground vehicles losing contact with the terrain, burrowing into the terrain, or failing to pitch and roll properly in response to changes in the slope of the terrain. Sometimes these effects appeared rather amusing; but for everyone involved, this was an important educational experience. Some of the observed anomalies were the result of incorrect coordinate transformations or lack of familiarity with variations in coordinate reference systems; others were the result of incorrect dead reckoning algorithms. Many involved correlation problems between differently rendered databases, which also related to varying database conversion techniques. A report on the problems and inconsistencies that can arise in converting and transforming databases, and the role of the database interchange format in this process, was produced by SIMNET engineers (Farsai, 1992) and a presentation on the topic was provided at the Fall 1992 DIS Workshop (Mamaghani, 1992). Overall, the event was a significant accomplishment, showing that diverse and heterogeneous simulation systems and computational platforms could be integrated together quickly. However, the event also showed that many additional issues remained to be resolved in the development of genuine interoperability standards. These issues are discussed further in the "Subsequent Developments" section, below.

[2] Excerpts from this demonstration may be seen at the end of an online video posted at https://www.youtube.com/watch?v=IzeiuvnTtwM. This posting also includes excerpts from the Army/Navy/Marine Corps BFIT exercise as well as the reconstruction of the Battle of 73 Easting. In the Senate hearing segment, the Chair is Senator Sam Nunn; other familiar faces include Senators John McCain, John Glenn, and Strom Thurmond. The presenter is retired GEN Paul Gorman. (If the link doesn't work, the video may be found by entering the search terms "73 Easting Senate.")

Predictive Value of Distributed Simulation

As illustrated in several of the examples just provided, one of the most powerful applications of Distributed Simulation (and here I am using the broader term, rather than SIMNET) is to assess the impact of battlefield innovations, including new weapon systems, sensor systems, and tactics. Doing this effectively is fundamentally an iterative process. Any significant innovation introduced into a combat situation changes what all of the participants do, both offensively and defensively. An example from the FAADS tests was discussed earlier, but those tests were not extensive enough to permit observation of more than the very initial adaptations.

Using a more extensive series of tests of a new (or proposed) system, combat and systems developers would need to allow for many more iterations and provide both attackers and defenders repeated opportunities to conduct after-action reviews and to modify their tactics accordingly. Only as this interactive process demonstrates evidence of convergence would the developers

- establish parameters for target detection, identification, hit, damage, and kill;
- document individual and small unit tactics for offense and defense and establish doctrines; and
- develop corresponding SAF behaviors for larger-scale testing.

They would then need to continue to monitor the results of further tests for the emergence of innovative tactical variations that suggest changes in the relative effectiveness of offensive and defensive units. These approaches are already standard in Developmental Testing and Operational Test and Evaluation. The primary advantage provided by the use of distributed simulation is to conduct more iterations, with larger units, at far lower costs and with far more detailed data than on instrumented ranges with prototype equipment.

Semi-Automated Forces

Having mentioned Semi-Automated Forces in multiple contexts, a brief summary of their evolution as a key component of SIMNET is in order.

Initially, when only a handful of manned simulators existed, it was a matter of basic necessity to populate the virtual battlefield with other, simple vehicles – to test the network protocols, to provide targets for weapons systems, and to provide the other complement of vehicles in platoon- and company-level teams in which coordinated maneuvers and tactics had to be executed. Later, as larger and more complex exercises were conducted, many more vehicles were needed to populate flanking and supporting units to provide a realistic context. Even if enough manned simulators had existed to fill these roles, doing so would have required hundreds of troops who would be doing little more than playing the roles of "extras," as in a battle scene in a movie. This was not only cost-prohibitive, but would also have been a colossal waste of manpower.

At the same time, it would have been destructive to the purpose of the exercises for these additional vehicles to just go through simplistic movements that would not have seemed credible to the primary participants. COL (R) Bloedorn, in particular, was adamant that there should be no non-reactive "automated targets," as he had seen in some previous conduct-of-fire trainers. This created a significant challenge for the developers. At a minimum, the unmanned vehicles needed to pass a basic "Turing Test," meaning that it should not be obvious to an observer which simulated vehicles were being controlled by human crews and which by computer algorithms. It was simply beyond the budget and time constraints of the program to even attempt to develop fully autonomous simulators (though significant strides were made over the course of the program, and continue to be made).

During an early demonstration, when COL (R) Bloedorn realized that he was seeing more moving vehicles on the display screens than the number of manned simulators that we had, he objected, "Wait! Those are automated vehicles." Anticipating this, I calmly explained that no, they were only semi-automated vehicles, with each vehicle being monitored and controlled by a human operator at a Plan View Display. So, let it be noted, I am claiming credit for the first use of the now-standard term Semi-Automated Forces! Let it also be noted that these SAF operators were using rather sophisticated controls and displays. While a trained tank crew could climb into a simulator and, with an hour or so of orientation and practice, operate it with considerable proficiency, this is not the case for a SAF operator. They do not need to be computer wizards, but they do need to understand both battlefield operations and learn to use multiple buttons and screen menus that have no direct counterpart in the real world.

Over the course of the next few years, the SAF entities became more and more sophisticated, as more individual and small-unit tactical concepts were built into their underlying algorithms, and especially as we implemented the principle that communications between the SAF and their supervisory controllers should be organized as much as possible to be like the orders (including "fragmentary orders" or FRAGOS) that a unit commander would pass to his subordinates and the reports they would transmit back. This increasing sophistication allowed a trained human controller to supervise greater numbers of vehicles. As a basic example, the controller might exercise detailed control over a platoon leader's vehicle, with the other vehicles in the platoon maintaining appropriate formations with respect to the leader. They would automatically change formations when the terrain demanded it, such as when crossing a bridge or other pinch point. They would also know to seek cover or concealment when they came under fire, without waiting for detailed directions.

SAF vehicles were first used in the CVCC and FAADS tests, though they were still relatively unsophisticated at that point. In the SAF POP and WAREX exercises, they represented the majority of the simulated entities.

Regarding terrain databases, it must be emphasized that SAF software cannot simply use the same databases used for out-the-window visual presentations. Unlike human drivers and commanders, SAF entities cannot "see" the terrain and infer from a listing of encoded polygons the existence of roads, bridges, obstacles, terrain types, etc. This information must be generated separately and stored in a form that the SAF software can process and use in order to exhibit reasonably "intelligent" behavior. As a simple example, the SAF terrain database must store road networks as a connected series of line segments. If there is a gap anywhere in this data, such that two line segments don't quite connect, the SAF software will conclude that no connectivity exists, and will find an alternate route that may appear ridiculous to human observers. Worse, it may conclude that there is no available path to the point it is trying to reach, and report to its controller that it is unable to proceed. Similarly, the databases used by the SAF entities must explicitly indicate impassible terrain, such as deep water or impenetrable forests. For consistency in simulation, the SAF entities must perceive the same information as human participants would under the same circumstances: the visibility of vehicles that may be obstructed by terrain, fixed and dynamic objects, smoke and other obscurants, muzzle flashes and missile trails, detonation of munitions, damaged and destroyed vehicles and similar effects. There must be no simulation anomalies that would compromise the integrity of the exercise. (Bess, 1992)

Development Methodology

In recent years, the concept of Agile Development has become popular. In fact, the principles articulated in the Agile Manifesto (Beck et al, 2001) have been traced back to the 1980s and to various books by James Martin during the 1990s. While making no claim to having developed a formal methodology, the SIMNET software developers applied most of these principles in our rapidly evolving body of software, which had reached a total of 1,000,000 lines of code by 1990. In particular, I would assert that the team utilized nearly all of the following principles (all taken from the 2001 Manifesto) while working with and integrating different perspectives from those inside and outside the SIMNET team principles on a regular basis throughout the 1980s and 90s:

1. Customer satisfaction by rapid delivery of useful software
2. Welcome changing requirements, even late in development
3. Working software is delivered frequently (weeks rather than months)
4. Close, daily cooperation between business people and developers
5. Projects are built around motivated individuals, who should be trusted
6. Face-to-face conversation is the best form of communication (co-location)
7. Working software is the principal measure of progress
8. Sustainable development, able to maintain a constant pace
9. Continuous attention to technical excellence and good design
10. Simplicity—the art of maximizing the amount of work not done—is essential
11. Self-organizing teams
12. Regular adaptation to changing circumstance

Many of these principles were already ingrained in the BBN culture, while others were necessitated by the frequent arrivals of new projects, requests for additions and modifications, lessons learned from frequent tests and demonstrations, and various other demands from both direct customers (those funding the efforts) and indirect customers (those demanding new and modified features as part of tests being conducted).

The "SIMNET Components" section of this paper should make clear that these components are highly interrelated. In most cases, modifying a vehicle simulator necessitated modifications to other components as well. This was particularly true of a new weapon or sensor. Not only did the control and display software of the vehicle to which it was added have to be changed, but also new projectile flyout and hit probability tables had to be generated. In addition, failure probability and damage tables for the new device had to be provided. Sometimes, as in the case of a laser designator, various modes and operating codes had to be taken into account, which could significantly affect the operation, flyout, and hit probabilities of other weapons. Furthermore, the Image Generator databases had to be modified to include and display the weapon signature effects seen by the firing vehicle and by other vehicles, as well as the detonation effects as seen from various distances and in various visual and sensor modes. New Fire and Impact PDU codes had to be added and incorporated into the software installed in all vehicles that would take part in an exercise, whether they were vulnerable to the new weapon system or not. New Activate, Deactivate, and Resupply PDU codes were needed by the MCC, NOM, PVD, DL, and AAR components. In most cases, the software for the SAF units also had to be modified to take into account the new battlefield threat, and behaviors had to be modified, sometimes substantially, to incorporate new tactics. The same changes for PDUs and hit/damage tables would, of course, have to be made for the SAF software.

These factors are mentioned here because the cascading implications of what a customer thought would be "just a minor change" could actually be quite extensive. When conducting final shakedown tests for an experiment involving a new system, I found it useful to keep on hand a PERT chart that demonstrated the chain of interactions a proposed "minor change" would involve. In many cases, it simply wasn't possible to incorporate and test all the necessary software modifications in the time available before the troops arrived and the test was scheduled to begin, and the customers had to understand and accept this situation.

Subsequent Developments

After observing the series of SIMNET tests and demonstrations summarized above, though some results were rather ragged, the Army moved rapidly to transition and incorporate the developments from DARPA's SIMNET R&D program into its formal programs of record, specifically the Combined Arms Tactical Trainer (CATT) Program. The first component to enter the procurement process was the Close Combat Tactical Trainer (CCTT). The program manager, COL James E. Shiflett, realized that the usual practice of letting the winner of the procurement redesign the software and protocols would mean that they would have to re-climb a steep learning curve. To quote from his I/ITSEC Fellow paper (Shiflett, 2013),

> Impressive as they were, the elegant design and integration of the simulators, computers and graphics were not the most important aspects of the SIMNET program. Rather, the glue holding and binding these pieces together, the SIMNET network protocols, eventually demonstrated the greatest and most enduring value. If the benefits of the DARPA effort were going to be successfully transitioned to the Army and Department of Defense (DoD), then all the SIMNET technologies would have to be available to others. SIMNET would have to have an open architecture derived from standards development processes jointly developed and widely accepted. A successful SIMNET program transition of technology to the services would require a clearly defined and documented standard, independent of the SIMNET hardware and software implementation.

Thus he persuaded the Army, in advance of the procurement, to fund an open, industry-based standards development process. This served both to educate the simulation industry, all of whom were intensely interested in the upcoming competition, and to ensure that the winning team would not simply decide to "reinvent the wheel" in terms of the protocols to be employed. The result was the series of DIS Workshops, which were conducted on a semi-annual basis beginning in 1989 and lasting through the CCTT program award in 1992 and beyond. The initial DIS standards were promulgated as IEEE Standard 1278-1993 (subsequently updated three times).

DARPA also continued studies aimed at moving the evolution of the DIS architecture forward to encompass up to 100,000 dynamic entities and to facilitate the further incorporation of live range exercises and large-scale aggregate-level simulations through a program initially called Advanced Distributed Simulation (ADS). (Cheriton et al, 1993) This program was renamed by DARPA as the Synthetic Theater of War (STOW) under the leadership of COL Bob Reddy. In October 1997, a STOW Advanced Concept Technology Demonstration (ACTD), Unified Endeavor 98-1, was conducted, involving the US Atlantic Command (USACOM) and the UK DERA/MOD. This was the largest HLA federation demonstrated to that point, incorporating 500 computers at five sites in the US plus two sites in the UK. (HLA is discussed later in this section.)

As the DIS Workshops continued, the participants realized that the rapid expansion of types of vehicles, weapons systems, munitions, sensor systems, etc. was overwhelming the capability of existing implementations. New listings of entities and effects were being posted by the DIS

“Enumerations” group, eventually on a monthly basis. As explained in the Development Methodology section earlier, each new entry could, and often did, generate a cascading list of changes that would have to be made in other components and other programs. Obviously, these programs were not funded to conduct an open-ended process of updates. Something had to change. The DIS Workshop established a special Protocol Architecture group that focused on ways for a particular program to declare what entities it would support, thereby limiting what updates would need to be incorporated. These efforts were overtaken, however, by the advent of HLA.

At the same time, a group at The MITRE Corporation in Virginia, with support from another DARPA office, was seeking to emulate the success of SIMNET/DIS in the aggregate-level simulation community. Beginning with GRWSIM (Ground Warfare Simulation), AWSIM (Air Warfare Simulation), and CBS (Corps Battle Simulation) in 1991, they and other Federally Funded R&D Centers (FFRDCs) succeeded in exchanging data among these heterogeneous simulations, and soon the Aggregate Level Simulation Protocol (ALSP) was born. The primary differences between DIS and ALSP are that while DIS exchanges data at the individual vehicle level, ALSP exchanges much larger sets of data about much larger units (battalions, brigades, and corps) and, instead of all simulations using continuous “wall-clock” time, the aggregate-level simulations proceed asynchronously. To ensure that no simulation creates changes that are already “in the past” for another simulation, one of two basic approaches is required. The “conservative” approach means that Federation of simulators can’t proceed at a rate faster than its slowest Federate, and the Run-Time Infrastructure (RTI) must enforce this constraint. The “optimistic” approach allows all Federates to proceed as rapidly as possible, but it requires a mechanism for checkpointing their states and “rolling them back” when necessary.

In 1991, the Deputy Secretary of Defense established the Defense Modeling and Simulation Office (DMSO), the predecessor of the current Modeling and Simulation Coordination Office (M&SCO). DMSO served as the executive secretariat for the DoD Executive Council for Models and Simulations (EXCIMS) and was responsible for the development and promulgation of modeling and simulation policy and standards. DMSO was to act to maximize modeling and simulation efficiency and effectiveness across DoD. By direction of the Deputy Secretary of Defense for Research and Engineering (DDR&E), DMSO initiated an ambitious new effort to create a single High-Level Architecture (HLA) that would address the needs of all classes of simulations. The HLA, promulgated as the IEEE Standard 1516 family of standards, includes

- IEEE Standard 1516 Framework and Rules
- IEEE Standard 1516.1 - Federate Interface Specification
- IEEE Standard 1516.2 - Object Model Template (OMT) Specification
- IEEE Standard 1516.4 - Recommended Practice for Verification, Validation, and Accreditation of a Federation—An Overlay to the High Level Architecture Federation Development and Execution Process

These standards were initially promulgated in 2000. As with the DIS Standards, they have been updated several times.

Probably the most controversial aspect of HLA is that each Federation requires a Run-Time Infrastructure (RTI) to handle all communications among the Federates, but RTI developers are free to develop different on-the-wire protocols for doing so. There are several different RTIs that are commercially available. The members of a Federation must agree on which RTI to procure and use. If they also wish to communicate with another Federation that uses a different RTI, they need to procure or develop a Gateway or one or more "Bridge Federates" that can handle both RTI protocols. This is different from the DIS approach, in which the Federates can select subsets or add supersets of PDUs to be exchanged. Both DIS and HLA remain in active use; given the investments in each, this situation will probably continue for years to come.

In his I/ITSEC Fellow's paper (Shiflett, 2013), Jim Shiflett discusses the critical importance of appropriate standards to the development and implementation of distributed simulation. He points to the power of such standards and relevant supporting policies in facilitating the transfer of innovative new technologies to major procurements for "programs of record." These standards incorporate the hard-won experience of the pioneering developers, which should not be ignored. He also observes that at even a modest level of personnel continuity from the initial developers to the system implementers can yield significant benefits in terms of "corporate memory," capturing why certain approaches were followed, while others that initially seemed promising but later turned out not to be feasible or to have major shortcomings were dropped.

He also notes that, on the other hand, an ill-advised mandate can significantly impede the technology transfer process. He notes cases where substantial expenditures of time and effort had to be abandoned, at considerable expense in terms of both funds and time schedule, because the mandated approach turned out not to be appropriate. I concur with his main points. I would also note that these recommendations are coming from those who led what has been called one of DARPA's most influential programs (see the opening section of this paper) and those who led the transition of the technology developed under this program to the implementation of one of the Army's most important and influential fielded systems. He and I both hope that the readers of our respective papers will take our advice seriously.

Shiflett also discusses the development of SEDRIS, which is designed to facilitate environmental representation and interchange, independent of any specific application. SEDRIS resulted in the adoption of eight standards under the auspices of ISO/IEC, and provides three distinct, but related, technologies:

- a Data Representation Model (DRM), for modeling, expressing, and representing a broad array of environmental data
- a Spatial Reference Model (SRM), for providing a unified approach to specifying coordinate reference frames and inter-converting coordinate values, and
- an Environmental Data Coding Specification (EDCS), for providing dictionaries that allow the assignment of meaning to objects and their characteristics.

The SEDRIS standards, associated implementations, and tools are used in various US DoD (mostly Army) programs, as well as being promulgated as NATO STANAGs. The complexities associated with the representation, exchange, and reuse of environmental data are beyond the scope of this paper. However, the standards and more in-depth understanding of these issues can be found through the resources, papers, tutorials, tools, and sample data at the SEDRIS website.[3]

[3] Refer to the ISO, IEC, and SEDRIS.org websites for a (far) deeper discussion of the issues involved.

Principal Achievements of SIMNET

SIMNET included important innovations in multiple areas. Which of these might be judged most significant depends on which aspects are being considered (architectural, algorithmic, or managerial, for example). Regardless of how these dimensions might be weighed, certain innovations stand out clearly:

The fundamental architecture. While elements of the overall design summarized in the "Key SIMNET Concepts" section had been used previously in the design of distributed systems (the ARPANET certainly being a prime example) they had never before been combined to create real-time distributed simulations that scaled approximately linearly with the number of entities being represented. The "autonomous node," "ground truth," and "remote vehicle approximation" concepts described earlier in conjunction with the overall "selective fidelity" approach were particularly crucial as central concepts of the SIMNET architecture.

Time Travel. Capturing time-stamped PDUs flowing across the network was an inherent capability provided by the fundamental architecture of SIMNET. Nevertheless, being able to replay an exercise as seen from any arbitrary vantage point, while skipping forward and backward in time, was a capability that proved so powerful for instruction and training, After-Action Review (AAR), data analysis, and conceptual visualization that it deserves mention as a key innovation itself.

Efficient terrain database imagery processing. The "hybrid Z-buffer" techniques developed by Delta Graphics (later BBN Graphics Technology) were a significant advance in the technology used to process constantly changing visual representations from viewpoints that could move unconstrained through a complex environment. These techniques included caching and swapping in new terrain "load modules" as a simulator traversed the database, as well as algorithms for representing complex objects as polygons overlaid with texture patterns that made them appear to be more geometrically complex than they actually were. A closely related development was that of efficient tools for use by database designers in abstracting the most important features of real-world representations into machine-independent representations for real-time image generation, production of soft and hard-copy maps, and support of navigation by autonomous and semi-automated systems.

Semi-Automated Forces. The development of software for representing the behavior of individual vehicles and small units on the battlefield represented another significant step forward in the state of the art. The driving concept was that each vehicle and unit had to exhibit credible behavior under a very broad set of circumstances, while requiring minimum intervention by human supervisors. The challenge of achieving this level of credibility, while avoiding cases of obviously absurd behavior, is much more difficult than even experienced software developers imagined. The fact that the SAF entities were able to pass the "Turing test" fairly consistently is a testimony to the leadership and members of the SAF development team.

Long-Haul Network implementation. The software developed for packet compression and routing also involved sophisticated techniques for determining which information from one LAN should be propagated to the others and at what level of detail. As with other aspects of SIMNET, these determinations were not fixed, but were constantly changing, and the routing software had to respond accordingly. This was particularly important in the simulation of radio transmission, which involved tracking which channels were being used by various units, unconstrained by geographical proximity in the simulated world.

Support for open industry standards. An important precedent was the development of the ARPANET, in which BBN's earlier work in interface message processing and network-layer protocols were central. However, the degree to which BBN participated in explaining the techniques embodied in its software to both government and industry organizations, and then actively participated in open standards committees that debated and refined these techniques, is unparalleled in my experience.

Conclusion

In closing, I want to express my appreciation to all of those who contributed to SIMNET and the subsequent developments that flowed from it. It was my privilege to work with some of the brightest people I've known over a career spanning some 50 years, 30 of which were devoted specifically to the development of distributed simulation. These people included not only BBN developers and operations staff, but also our subcontractors and co-contractors, DARPA program managers and consultants, laboratory research staff from the Army and other services, and military officers and enlisted personnel who served as the initial users of a multitude of software components, providing the team with invaluable feedback and frank opinions. In delving through my voluminous files while writing this paper, I found that I could recall the names, faces, and contributions of nearly every one of these people, without whom SIMNET could not have produced the results that it has. It's been, as they say, quite a trip.

References and Bibliography

Alluisi, Earl A., "The Development of Technology for Collective Training: SIMNET, A Case History," Human Factors, Vol 33, No. 3, pp. 343-361, 1991

Atwood, Nancy K., Kathleen A Quinkert, Mary R. Campbell, Karen F. Lameier, Bruce C. Leibrecht, and William J. Doherty, "Combat Vehicle Command and Control Systems: Training Implications Based on Company-Level Simulations," Technical Report 943 AD-A246 460, US Army Research Institute and BDM International, Inc., December 1991

Beck, Kent, et al. "Principles behind the Agile Manifesto". Agile Alliance, 2001

Bess, R., "Tradeoffs in the Configuration of Computer Image Generations Systems," Proceedings of the 11th Interservice/Industry Training Systems Conference, 1989

Bess, Rick D., "Image Generation Implications for Networked Tactical Training Systems," IMAGE VI Conference, Scottsdale, AZ, 1992

Bloedorn, Gary, "SIMNET M2/M3 Controls and Displays," Perceptronics, August 1985

Brooks, Peter S., and Dennis DeRiggi, "The Integration of SIMNET With a Theater-Level Combat Model," IDA Paper P-2491, Institute for Defense Analyses, Sept 1990

Brown, Phillip J., and J. K. Lavender, "Virtual Prototyping: Results Illustrate Utility In Developing Weapon System Requirements, Loral Vought Systems, 1995

Callero, Monti, C. T. Veit, E. C. Gritton, and R. Steeb, "Enhancing Weapon System Analysis: Issues and Procedures for Integrating a Research and Development Simulator with a Distributed Simulation Network," Rand Corporation, 1994

Ceranowicz, Andrew Z., "ModSAF Capabilities," Proceedings of the Fourth Conference of Computer Generated Forces and Behavioral Representation, Orlando FL, 1994

Ceranowicz, Andy, “Metasimulation,” Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), Paper 1F1401, 2014

Cheriton, David R., V. R. Lesser, D. C. Miller, D. L. Mills, and J. D. Fletcher, “Advanced Technology Demonstration One (ATD-1) Architecture Panel Review, IDA Document D-1392, July 1993

Chung, James W., “SIMNET M1 Abrams Main Battle Tank Simulation, BBN Report 6323 (August 1988)

Cosby, Neale, “SIMNET — An Insider’s Perspective,” Institute for Defense Analyses Paper D-1661, Alexandria, VA, 1990

Cosby, Neale, “WAREX 3-90 (A Virtual Warfighting Exercise in Advanced Battle Simulation),” Institute for Defense Analyses, Alexandria, VA, 1990

Cushman, J. H, B. L. Harrison, A. J. Junot, J. Kirk, G. W. Bloedorn, and W. H. Crooks, “ Advanced Battle Simulation: Semi-Automated Forces Proof-of-Principle Demonstration After-Action Report, Perceptronics, 1989

Davis, Gardner, "LOSAT SIMNET Simulation Concept Evaluation Program and System Design Phase Summary," LOSAT-SDP/92DIR-0276, June 1992

IEEE Standard for Information Technology – Protocols for Distributed Simulation Applications: Entity Information and Interaction, IEEE Standard 1278-1993, IEEE, New York, 1993

Farsai, Steve, D. Gafford, R. Gaines, and F. Mamaghani, “Technical Report on the Conversion of S1000 Database to and from Project 2815 SIF Format for Use in the I/ITSEC ’92 Interoperability Demonstration,” 1992

Gorman, Paul F, “Learning to Learn: Reminiscences and Anticipation,” Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), Paper 1F1101, 2011

Harrison, Ben L., “Army Aviation Simulation Survey,” IDA Paper P-2709, Annex C, Appendix 1, Institute for Defense Analyses, Alexandria, VA, July 1992

Johnston, Richard S., "The SIMNET Visual System", 9th I/ITSC, 1987, pp. 264-273.

Kramer, Ronald E., and D. W. Bessemer, “U.S. Tank Platoon Training for the 1987 Canadian Army Trophy (CAT) Competition Using a Simulation Networking (SIMNET System,” US. Army Research Institute Field Unit, Fort Knox, 1987

Lanpher, M., “SIMNET Final Report,” Martin-Marietta Electronics and Missiles Group Report OR-19,776, Orlando, FL (1989)

Lickteig, Carl, and C. K. Heiden, “The use of Semi-Automated Forces in CVCC Company Level Experiments,” August 1989

Loper, Margaret, B. Goldiez, and S. Smith, “The 1992 I/ITSEC Distributed Interactive Simulation Interoperability Demonstration,” Institute for Simulation and Training, University of Central Florida, Orlando, FL, 1993

Mamaghani, Farid, S. Farsai, D. Gafford, and R. Gaines, “Interchange of Databases: Validation and Conversion of SIF/HDI Hunter-Liggett Database for I/ITSC ’92,” BBN Systems and Technologies, presented at the DIS 7th Workshop, September, 1992

Miller, Duncan C., A. R. Pope, and R. M. Waters, "Long-Haul Networking of Simulators" Proceedings: Tenth Interservice/Industry Training Systems Conference, Orlando, FL, December 1989

Miller, D. C. and J. A. Thorpe, "SIMNET: the Advent of Simulator Networking," Proceedings of the IEEE, Vol. 83, No. 8, pp. 1114-1123, 1995

Pope, Arthur R, T Langevin, L. Lovero, and A. R. Tosswill, "The SIMNET Management, Command and Control System," BBN Report No. 6473 (Revised), 1988

Pope, Arthur R. and R. L. Schaffer, "The SIMNET Network and Protocols" BBN Report No. 7627, Cambridge, MA (Revised Version), June 1991

The RAND Corporation, "Feasibility of Applying SIMNET Technology to the Weapon Systems Development and Acquisition Process," RAND National Security Research Division, November 1989

Richardson, J. J., "Transitioning DARPA Technology," Potomac Institute of Policy Studies, Arlington, VA, July 2001

Sanders, William R., "MANPRINT Support of the Non-Line-of-Sight Fiber-Optic Guided Missile System," AD-A278 153, US Army Research Institute for the Behavioral and Social Sciences Research Report 1660 (January 1994)

Schwartz, R. E. and M. M Stahl, "Innovative Analytic Techniques for Distributed Interactive Simulation (DIS)," Institute for Defense Analyses Document D-1902, Sept 1996

Seidel, R. J. and P.R. Chatelier, "Advanced Technologies Applied to Training Design," Plenum Press, NY, 1993

Shiflett, James E., "Observations on the Development and Implementation of Distributed Simulation," Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), Paper 1F1301, 2013

Thorpe, Jack, "Future Views: Aircrew Training 1980 – 2000," unpublished white paper, Life Sciences Directorate, Air Force Office of Scientific Research, 15 Sept 1978

Thorpe, Jack A., "Trends in Modeling, Simulation & Gaming: Personal Observations About the Last Thirty Years and Speculation About the Next Ten," Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), Paper 1F1001, 2010

Tiernan, Tom, K. Boner, and D. Hardy, "Battle Force Inport Training / Simulator Networking: Proof of Principle," Naval Ocean Systems Center, San Diego, 1990

Wever, Peter, E. Lang, and S. Smyth, "SIMNET Database Interchange Specification," BBN Report No. 7108, July 1989

Wever, Pete, and E. Lang, "SDIS Version 3.0 User's Guide: Interchange Specification, Class Definitions, Application Programmer's Interface," August 1990

Yoo Phillip, "SIMNET M2/M3 Control/Display Recommendations," BBN, June 1985