

# THE E2C-PART TASK TRAINER IN A REAL TIME DISTRIBUTED SIMULATION ENVIRONMENT: CAN DISTRIBUTED PROTOCOLS WORK FOR ALL?

James Panagos  
Loral Western Development Laboratories  
San Jose, Ca

John Morrison  
MaK Technologies  
Cambridge, Ma

Distributed Interactive Protocols (DIS) promise that any simulator "speaking" the protocol can link up to a network without major modification. This paper will consider this premise in the light of a case study of connecting the E2C Part Task Trainer to Simnet. We postulate that, although some changes do need to connect to a distributed network, they can be categorized and organized to independent software which can interface to the network and minimize disturbance to the simulator. We hope that this study can guide future conversion efforts to distributed simulation.

## About the Authors

John Morrison is the original author of Simnet M1 and SAFOR protocol implementations at Bolt Beranek and Newman. He has been involved in the Simnet and DIS since 1985. Currently, he owns a small company which specializes in Distributed Simulation technology. Mr Morrison holds a B.S. in Computer Science and Electrical Engineering from the Massachusetts Institute of Technology.

James Panagos has been involved in Simnet and Distributed Simulation since 1987 at Bolt Beranek and Newman. He was currently involved in implementing AGPT, a distributed trainer in Ada for the German Leopard2 tank. Mr Panagos holds a B.S. in Computer Science from the State University of New York at Albany and a M.S. in Computer Science and Electrical Engineering from the Massachusetts Institute of Technology.

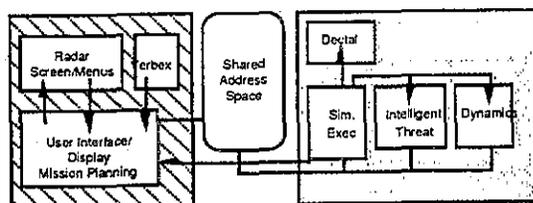
## INTRODUCTION

How difficult is it to connect to a distributed network? This paper attempts an answer to this question by discussing the adaptation of SIMNET protocols to an existing desktop system: the E2C Part-Task Trainer (E2C-PTT). An E2C-PTT capable of distributed communications was never fielded. However, the problems have been studied and a prototype was begun. It is our belief that this study sheds light in the requirements of connecting any system to a DIS-type network. Although we acknowledge the differences between Simnet and DIS, for the purposes of this paper, we have used the terms interchangeably without detracting from the conclusions.

## THE E2C PART TASK TRAINER

The E2C-PTT is a knowledge based desktop system designed to train E2-C Navy aircraft radar operators in conducting and managing fighter intercepts. The task of the radar operator is to identify threats, hand them off to an inventory of fighters, and if necessary, conduct an intercept by relaying specific commands to the fighter pilot.

The E2C-PTT is comprised of the following functional modules:



The User Interface process supports a keyboard, menu and mouse interface. It passes user requests to the Simulation process. The Simulation process itself runs unimpeded by user interrupts as separate processes were spawned by the windowing system of the user interface upon each request. The light weight Symbolics process discipline is crucial in separating the *asynchronous* Frontend process from the

*synchronous* Simulation process but yet allowing the easy exchange of data.

## REQUIREMENTS FOR DIS COMPLIANCE

In connecting the E2-C to the Simnet network, we had to ensure that it met the fundamental DIS requirements in the areas of

- maintaining fidelity
- processing network events
- creating network events.

### Fidelity

**Fidelity of Space**-The difference in terrain representations was a difficult issue. The E2C-PTT uses a gridded Defence Mapping Agency (DMA) representation of the battlefield; Simnet uses a polygonal representation. The lack of correlation between these two could create anomalies in:

- *Intervisibility*. Detection and engagement algorithms, the most significant users of this function, could be gravely skewed
- *Placement of vehicles*. Placement of vehicles could be imprecise if the exact location of the terrain is not known.
- *Terrain interaction*. Incorrect representation of terrain data representation and granularity could create occlusion problems for out of the window graphical representation of the terrain.
- *Terrain gaming area*. The larger gaming area of the E2C-PTT could cause correlation problems with Simnet simulators.

**Fidelity of Time**-The E2C-PTT has an update rate of approximately 3Hz. In interacting with the essentially 15Hz. Simnet network, E2C-PTT aircraft would have to:

- keep up with the rate of incoming events. Appearance packets would have to be processed at about 15Hz.
- keep a steady flow of information to the network. Appearance packets had to be put out for basic 15 Hz observers

Processing Externally Generated  
Inputs

**Internal vs External Events-** Whereas the E2C-PTT originally generated and knew opponent behavior and plans, on Simnet, the state of a externally generated vehicles is unknown; it would have to be inferred from current behavior and trajectory. This is a very difficult task.

**Amounts of Events-**The sheer amounts of events coming from the network would overtax the Symbolics. A burst of activity (such as a barrage of indirect fire) could dominate the simulation at critical times. This could detract, for example, from the updating of aircraft acceleration and velocity parameters while in quick steep turning thus sending it to the ground or into a hill.

### Generating External Events

The E2C-PTT would have to adopt new behavior for interacting with DIS. No longer orchestrating movement throughout the simulation world, it would have to check for collision with external vehicles and generate collision packets. Collision processing would have to be integrated and targeting algorithms would have to be adjusted for events previously not considered such as the existence of ground threats.

### HARDWARE PROBLEMS

The E2C-PTT hardware platform, the Symbolics 3600, lacked adequate compute power to handle the network. At little over 1 MIP, the Symbolics on the DIS network would have to run the expert system, handle the User Interface displays and generate packets for its vehicles at 15 Hz.

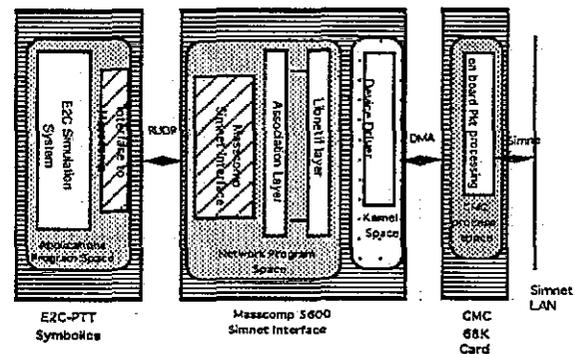
The architecture of the Symbolics compounded this problem. The regular Simnet solution to the speed problem (intelligent network devices, Direct Memory Addressing, etc) were not available to the unique L-Bus architecture of the Symbolics. Furthermore, different byte ordering on the Symbolics required costly corrections. And finally the garbage collection scheme of the Symbolics could at any time take ahold of the processor cycles

thus leaving the simulation dry of needed cycles.

### SOLUTIONS TO THE NETWORK PROBLEMS

To meet the full range of issues caused by DIS, a Unix workstation, the Masscomp 5600, was architected to interface to the network for the Symbolics. The Masscomp would

- update vehicle dynamics
- calculate exogenous events such as damage and collision
- calculate fire and hit possibilities
- process network traffic
- adopt and interface with Simnet terrain databases
- use Simnet intervisibility routines



Architecture of the E2C-PTT on the Simnet Network

This approach solved problems that degraded the fidelity of the network. It

- maintained fidelity of space by using the same databases and data-based tools as did the rest of Simnet
- maintained fidelity of time by tending to the updating of input and output packet queues in a timely fashion.

In addition, separating the E2C-PTT processing from the network shielded the simulation from network "babbling".

Furthermore, the garbage collector had to be turned off to keep it from

interfering with communications to the Masscomp.

tactics and future behavior easily predictable

## CONCLUSIONS

We found the following architectural assumption in Simnet that forced us to tailor our architecture.

- Part Task is Not Enough. We were forced to add functionality to cover point to point communications such as damage and collision packages. Without it, the E2C-PTT would corrupt interaction with its neighbors.
- Distributed but Equal. The response of the new simulation to events on the distributed battlefield must be on a par with all others. Efforts have to be made to fine tune characteristics such as damage, hit statistics, and vehicle performance. When all vehicles do not share similar algorithms such as targeting and damage, engagements and other interactions are skewed.
- Distributed but correlated databases. Having identical database or an closely correlated database is paramount to eliminate visual anomalies.
- Speed. This is key. Without it, DIS traffic can bring a part-task system to its knees.

However, not all problems were solved. The architecture created new problems;

- it caused the duplication of information between the Masscomp interface layer and the Symbolics application layer.
- it added latency to processing data.
- it added extra layers of code between the two systems.
- it forced the simulation to work with uncertainty. No longer were

The Masscomp network interface software was also successfully used as a network front end in SAFOR systems and the VINT2 system. Its reusability and longevity proves that by understanding and addressing the crucial issues of network connectivity, a network interface such as that designed for the E2C can make the connection to DIS as easy and repeatable (via software reuse) as possible.