customer success

National Advanced Driving Simulators & Simulation Center

RTI is integrated into the distributed virtual proving ground

> *"As we evaluated different networking middleware options, including CORBA, DCOM and HLA, it became clear that RTI was not only the easiest to use, but also the most effective at controlling update rates, message ordering, and message latencies."*
>
> Dr. Yiannis Papelis,
> Chief Technical Officer of NADS-SC

Figure 1. Simulator at NADS-SC

The development of heavy equipment ground vehicles such as farm tractors or US Army HMMWVs historically required building one-of-a-kind physical prototypes and subjecting them to lengthy, human assisted testing. These prototypes were expensive and test cycles for them were time consuming. Today, however, with advances in high-performance computing, virtual prototypes in digital form can often replace physical prototypes. Moreover, with the advent of networking technologies, geographically dispersed sites of virtual prototypes can be connected to create a virtual proving ground (VPG). The use of VPGs substantially increases the types and quantity of tests that can be performed while decreasing the overall time needed for testing.

Under a recent National Automotive Center (NAC) Dual Use Application Program (DUAP) contract, a distributed VPG was developed and demonstrated through a cooperative effort between the University of Iowa National Advanced Driving Simulator and Simulation Center (NADS-SC) and the US Army TACOM-TARDEC Ground Vehicle Simulation Laboratory (GVSL). Researchers connected several high-fidelity simulators to demonstrate how US Army vehicles and components can be tested and evaluated using VPG technology. Researchers chose the RTI Data Distribution Service (formerly NDDS) network middleware from Real-Time Innovations (RTI) to tie together the VPG simulator sites. This paper describes why integrating RTI into the distributed VPG was vital to its success.

Distributed VPGs Accelerate Vehicle Development

The VPG developed under the NAC-DUAP contract was built to study the interrelationship between the performance of heavy equipment ground vehicles and their human occupants. By having both the vehicle operator and the design engineers collaborate on different aspects of vehicle performance and comfort, various design trade-offs can more quickly be made. For example, different suspension components might be swapped out and compared against one another to see how different designs affect the ability of the user to perform tasks while riding over diverse terrain. Future versions of such VPGs will enable the end-users of the ground vehicles to become an integral part of the design, development and acquisition process.

The beauty of a distributed VPG is that equipment or subsystem design engineers do not have to be co-located with the simulator research facilities in order to participate in the testing. One study of the NAC-DUAP VPG, for example, was performed with the vehicle simulator and its operator at NADS-SC in Iowa (Figure 1) and a design engineer at TACOM-TARDEC in Michigan. The simulated scenario had the vehicle immobilized in a dry riverbed unable to climb a steep slope. The design engineer used his workstation to communicate directly with the remote simulator and scale various powertrain parameters in real-time until the vehicle was able to climb out of the riverbed.

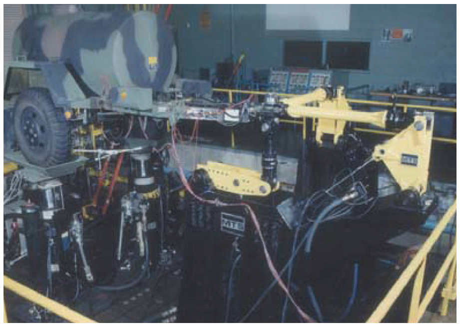

Figure 2. Pintle Motion Based Simulator (PMBS)

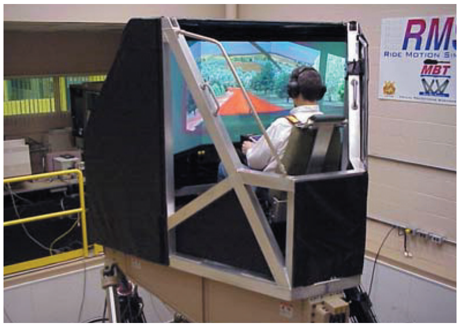

Figure 3. Ride Motion Simulator (RMS)

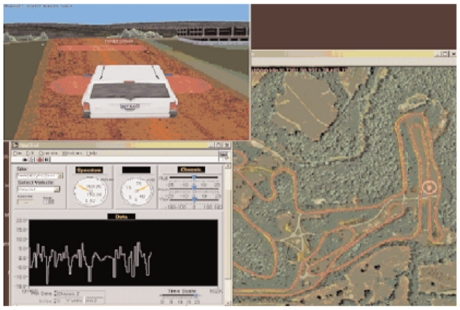

Figure 4. Screen Capture From NAC-DUAP VPG Tools

A distributed VPG also permits separating the functions of a single vehicle system over two or more geographically dispersed simulators. For example, a scenario being discussed would include a HMMWV driver situated in the NADS-SC facility, while a trailer may be mounted on the Pintle Motion Based Simulator (PMBS) pulled by the HMMWV (Figure 2) hundreds of miles away. Both simulators would be representative of a single vehicle system, but the effects of the vehicle design on the driver and on the hardware (either the trailer itself or it's cargo) could both be examined simultaneously in a single experiment.

Perhaps the most complicated demonstration done to date with the NAC-DUAP VPG consisted of a joint operation between two remote vehicle simulators. This demonstration consisted of a simulated HMMWV operated by an occupant of the GVSL Ride Motion Simulator (RMS Figure 3) and a Jeep Cherokee vehicle model operated by a driver in a full motion simulator at the NADS-SC facility. Both simulators operated over identical digital representations of the Churchville B test course modeled from actual terrain at the Aberdeen Proving Grounds in Maryland. Both occupants were able to view the other vehicle just as if the vehicles were driving on the same test course simultaneously. Simulation monitoring tools (Figure 4) were used to capture engineering data and display it in various forms to the design engineers. For instance, an engineer might request to capture the data necessary to create a real-time stress contour plot of a selected vehicle component. The simulation monitoring tools were available at either site and engineers could view data at either site.

Technical Challenges of Designing a Distributed VPG

Creating a distributed VPG requires more than periodically transferring a few data items to other participating sites. Rather, very complex, physics-based simulations must interact in a time-critical manner—literally sharing a complete virtual environment with perhaps thousands of data points. Moreover, the data distribution must be done in real-time with a low-latency transfer protocol that ensures that each participant has an identical view of the simulated world.

Engineers from NADS-SC derived the following requirements for a communications engine that would facilitate data sharing between VPG sites:

- Low-latency, direct transfer of data between sites
- Support for reliable transfer of data
- An easy-to-use API
- User-transparent data conversion between different operating systems and processors
- Communications without explicit point-to-point addressing

For NADS-SC engineers the first choice for creating a data distribution engine was to develop it themselves using either TCP/IP or UDP/IP. However, researchers concluded that TCP generally performed poorly when used for time critical applications, a conclusion that does not come as a surprise, due to the focus of the protocol on reliability rather than timeliness. While they saw how they could easily write some device-specific code that communicates key variables between two or more machines, they realized that this approach was not extensible in functionality or network size. Additionally, TCP methods, while providing reliable communications, are notoriously non-deterministic due to design features such as its rigid retry algorithm and time-outs with exponential back off.

In contrast, UDP methods have less overhead than TCP methods but they provide neither reliable delivery of data nor guaranteed message ordering. While much of the data shared in a distributed VPG is sent in a streaming fashion (i.e., repetitively at periodic intervals) other data items are transferred only once and must be sent reliably. Therefore, if UDP was to be used, a reliable protocol would need to be built from scratch—a task that the NADS-SC engineers believed could potentially take many months of effort.

Network Middleware Options for a Distributed VPG

Realizing that writing communications middleware had no direct benefit to the VPG project, researchers decided to evaluate various off-the-shelf middleware options for the real-time distribution of data.

Several networking middleware solutions were considered—including distributed object technologies (CORBA and DCOM), and software designed specifically for distributed military simulations—the IEEE's High Level Architecture (HLA).

While all of these middleware technologies implicitly or explicitly addressed data interchange across distributed systems as a primary capability, they also had some key disadvantages. First, all are rather complex frameworks that require extensive training before they could be used effectively. Second, they generally utilize a connection-based communication method or a client server protocol that tended to add latency and indeterminism to data transmissions. Finally, they focus on software re-use and language independence first, with performance only as a secondary goal. As previously mentioned, real-time performance is a key requirement for distributed VPG applications.

NADS-SC researchers found what they were looking for with RTI networking middleware. RTI Data Distribution Service is a publish-subscribe middleware designed specifically for use in real-time environments. With a minimal coding effort, researchers were able to use RTI to enable applications to collaborate at different sites via the Internet using inexpensive commodity network interfaces. One important and unique aspect of using RTI is that VPG applications can collaborate in real-time even though they had not been designed with an explicit awareness of each other.

Using the RTI publish-subscribe model, applications use named topics rather than network addresses to distribute data. Publishers simply create a publication and give it a topic name. Then, to send an issue (data) the application just calls a single RTI function. Subscribers simply create a subscription for a topic name and tell RTI what to do when a new issue arrives. Every time the publication has a new issue RTI handles the network I/O, transparently sending each issue from the publisher to all subscribers with a declared interest in that topic.

Conclusion

By using RTI as the communications middleware on the NAC-DUAP VPG, researchers from NADS-SC were able to concentrate on developing the VPG capabilities rather than on developing infrastructure or on learning a complex framework. RTI provides a simple, yet powerful publish-subscribe middleware that meets the demanding real-time requirements of applications such as the distributed VPG. Dr. Papelis sums up his thoughts on RTI, "We will definitely use RTI on future projects, it is high-quality middleware that simply makes our job a lot easier."

About RTI

RTI supplies middleware and distributed data management solutions for real-time systems. With innovative technology and deep expertise in distributed applications, RTI provides an unequaled competitive advantage to customers developing systems that benefit from high-performance access to time-critical data. RTI solutions have been deployed in a broad range of applications including command and control, intelligence, surveillance, data fusion, simulation, industrial control, air traffic control, railway management, roadway traffic monitoring and multimedia communications. Founded in 1991, RTI is privately held and headquartered in Sunnyvale, California.

US HEADQUARTERS
Real-Time Innovations, Inc.
385 Moffett Park Drive
Sunnyvale, CA 94089
Tel: (408) 990-7400
Fax: (408) 990-7402
info@rti.com

www.rti.com