SANDIA REPORT

SAND2005-6642
Unlimited Release
Printed October 2005

Integrating Software Architectures for Distributed Simulations and Simulation Analysis Communities

John M. Linebarger, Daniel Fellig, Patrick D. Moore, Mike Goldsby,
Marilyn F. Hawley, and Timothy J. Sa

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,
a Lockheed Martin Company, for the United States Department of Energy's
National Nuclear Security Administration under Contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd.
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online order: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND 2005-6642
Unlimited Release
Printed October 2005

Integrating Software Architectures for Distributed Simulations and Simulation Analysis Communities

John M. Linebarger and Daniel Fellig
Distributed Computing Department

Patrick D. Moore
National Systems Modeling and Simulation Department

Mike Goldsby
System Studies Department

Marilyn F. Hawley and Timothy J. Sa
Systems Research Department

Sandia National Laboratories
P.O. Box 5800
Albuquerque, New Mexico 87185-5800
P.O. Box 969
Livermore, California 94551-0969

Abstract

The one-year Software Architecture LDRD (#79819) was a cross-site effort between Sandia California and Sandia New Mexico. The purpose of this research was to further develop and demonstrate integrating software architecture frameworks for distributed simulation and distributed collaboration in the homeland security domain. The integrated frameworks were initially developed through the Weapons of Mass Destruction Decision Analysis Center (WMD-DAC), sited at SNL/CA, and the National Infrastructure Simulation & Analysis Center (NISAC), sited at SNL/NM. The primary deliverable was a demonstration of both a federation of distributed simulations and a federation of distributed collaborative simulation analysis communities in the context of the same integrated scenario, which was the release of smallpox in San Diego, California. To our knowledge this was the first time such a combination of federations under a single scenario has ever been demonstrated. A secondary deliverable was the creation of the standalone GroupMeld™ collaboration client, which uses the GroupMeld™ synchronous collaboration framework. In addition, a small pilot experiment that used both integrating frameworks allowed a greater range of crisis management options to be performed and evaluated than would have been possible without the use of the frameworks.

Contributors

Michael M. Johnson, 8114 (Project Manager)
Steven D. Kleban, 6226 (Principal Investigator)

Heidi R. Ammerlahn, 8962
Donna D. Djordjevich, 8114
Mike E. Goldsby, 8114
Marilyn F. Hawley, 8112
Howard H. Hirano, 8152
Timothy J. Sa, 8114
Ann S. Yoshimura, 8112

Mark S. Bastian, 6633
Benjamin M. Currier, 6221
Mark A. Ehlen, 6221
Daniel Fellig, 6224
John M. Linebarger, 6224
Patrick C. Moore, 6226
Andrew J. Scholand, 6221

Table of Contents

This page intentionally left blank

1. Introduction

This report documents the results of a one-year Software Architecture LDRD, which was a cooperative effort between Org. 6220 in Sandia New Mexico and Org. 8110 in Sandia California. The purpose of the LDRD was to create and demonstrate integrating software architectures for distributed simulations and simulation analysis communities. A unifying scenario (a smallpox release in the San Diego area) was used as the demonstration vehicle.

The primary research contribution of the LDRD was the creation of two federations linked together by the unifying scenario—a federation of distributed simulations, and a federation of distributed simulation analysis communities. The federation of two distributed simulations was created to generate and respond to the scenario. The first simulation was BioDAC, a crisis management simulation for biological weapons of mass destruction, which ran in Sandia California. The second was N-ABLE™ (more precisely, a process emulating N-ABLE™), an agent-based microeconomic simulation, which ran in Sandia New Mexico. The IDSim distributed simulation framework was used to create a unified federation of these two simulations. The two simulation analysis communities were those associated with the BioDAC and N-ABLE™ simulations, respectively. The GroupMeld™ synchronous collaboration framework was used to create a collaborative federation of these two simulation analysis communities. To our knowledge, this is the first time that these two kinds of federations linked together in a common scenario have ever been demonstrated. These two federations are schematically depicted in Figure 1.

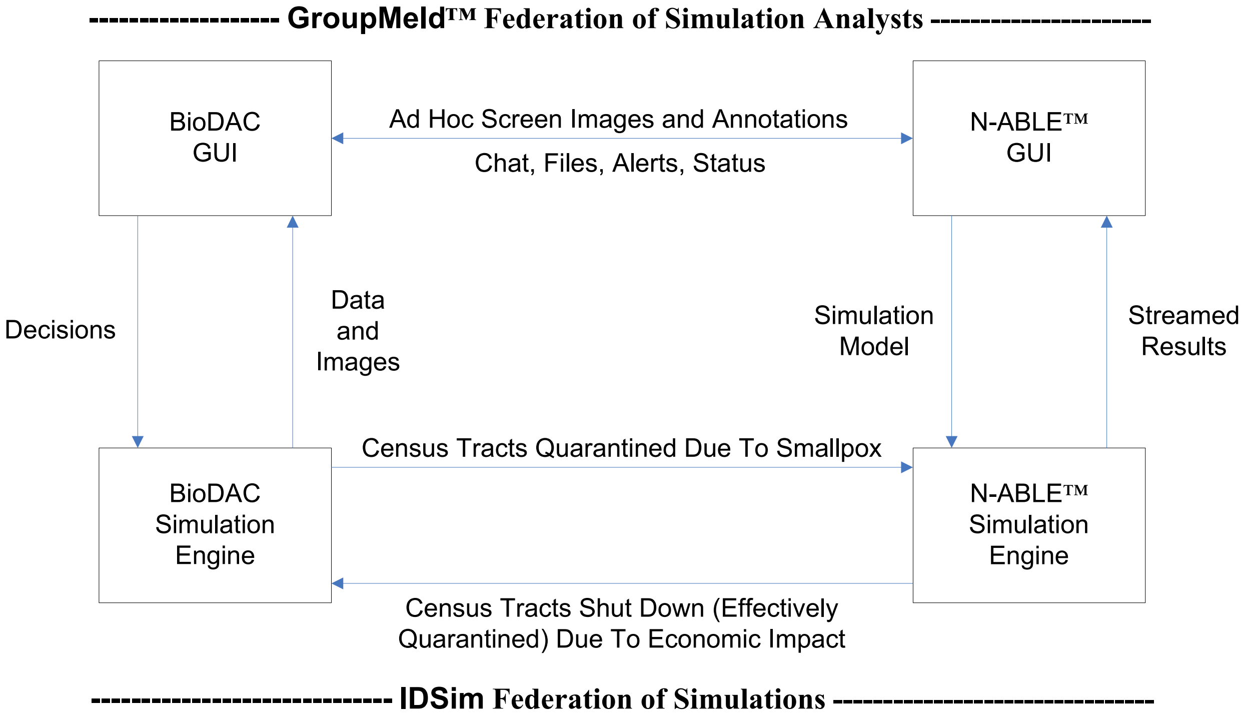


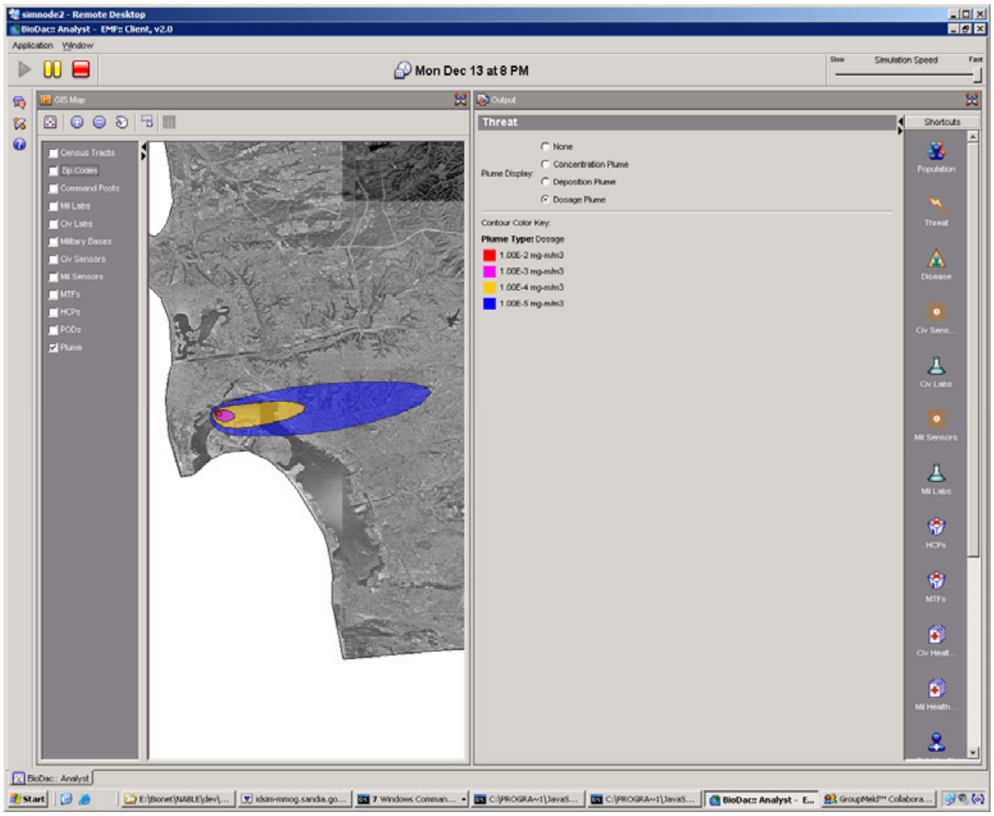
Figure 1. Two Federations under one unifying scenario

Another significant deliverable of the LDRD was the creation of a standalone GroupMeld™ client application. This client application uses the GroupMeld™ collaboration framework to provide a rich set of collaboration services, and is available on demand from an internal Sandia Web site (https://cip-qual.sandia.gov/GroupMeld/GroupMeld.jnlp).

In the pages that follow, the two simulation codes (BioDAC and N-ABLE™) will be briefly described; the integrating architectures (IDSim and GroupMeld™) will be detailed; technical considerations will be discussed; performance measurements for each architecture will be tabulated; the results of a small pilot experiment using the unified scenario will be presented; and future work will be outlined.

2. BioDAC

BioDAC (an abbreviation for the Weapons of Mass Destruction Decision Analysis Center [WMD-DAC] Biological Defense Application) is a component of the Weapons of Mass Destruction Decision Analysis Center (WMD-DAC) suite of simulation components. BioDAC is used to simulate the release of biological agents and evaluate the efficacy of a rich set of response strategies. Three primary roles exist in a BioDAC simulation—Public Health Official (PHO), Navy Official (NO), and Analyst. Figure 2 below portrays the plume associated with the release of a biological warfare agent into the atmosphere.



Plume Display:
○ None
○ Concentration Plume
○ Deposition Plume
● Dosage Plume

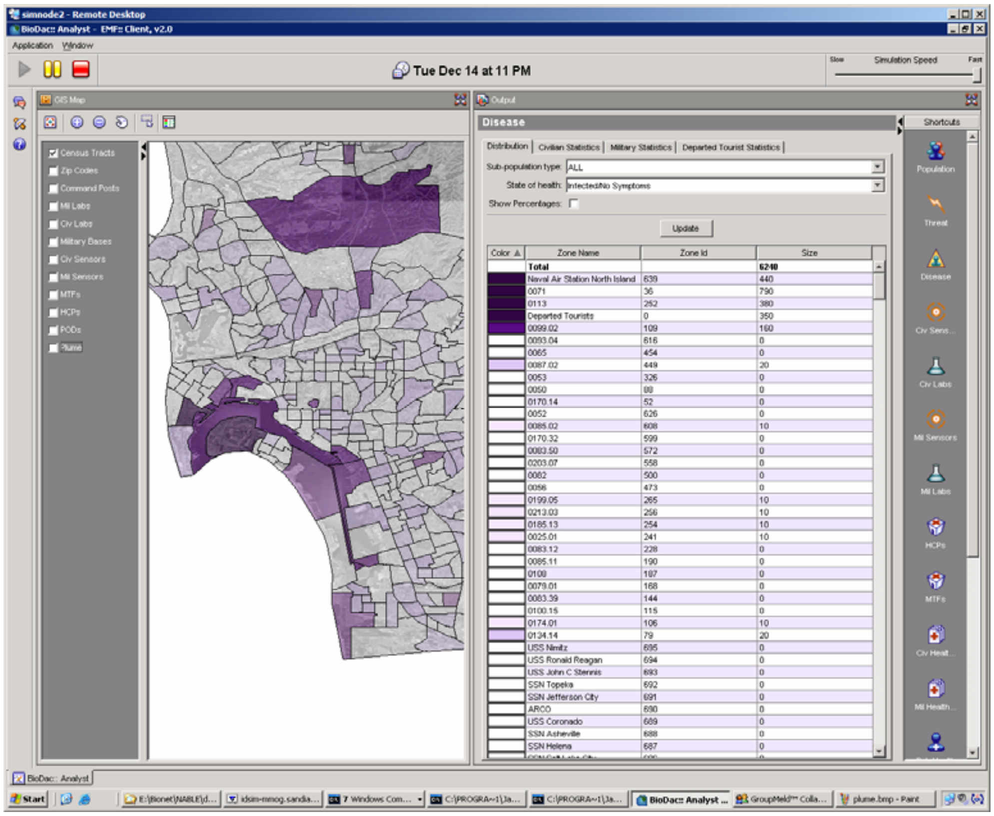
Contour Color Key:

Plume Type: Dosage

🟥 1.00E-2 mg-m/m3
🟪 1.00E-3 mg-m/m3
🟨 1.00E-4 mg-m/m3
🟦 1.00E-5 mg-m/m3

Figure 2. BioDAC plume display

The graphical user interface of a BioDAC screen is dynamically created depending on the role of the user in the simulation. In general, the simulation itself runs on a separate machine than the one used to display the graphical user interface (GUI). The simulation consists of multiple components, such as the warfare agent dispersion model (provided by the Defense Threat Reduction Agency Hazard Prediction and Assessment Capability [DTRA HPAC]), the population movement model, the disease model, and the hospital model. Figure 3 below shows the distribution of the disease based on the infection of a population from the plume dispersion in Figure 2.



Sub-population type: ALL

State of health: Infected/No Symptoms

Show Percentages: ☐

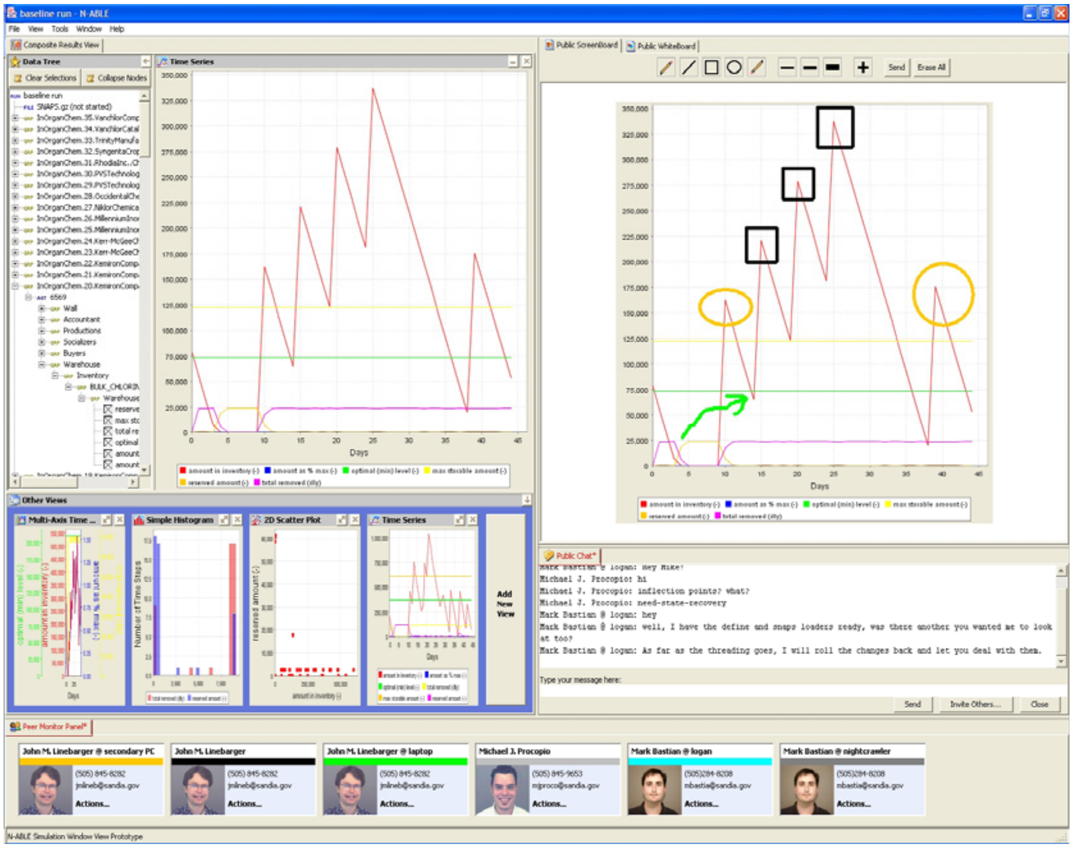
Color	Zone Name	Zone Id	Size
	Total		**6240**
■	Naval Air Station North Island	639	440
■	0071	36	790
■	0113	252	380
■	Departed Tourists	0	350
■	0099.02	109	160
□	0093.04	616	0
□	0065	454	0
▨	0087.02	449	20
□	0053	326	0
□	0050	88	0
□	0170.14	52	0
□	0052	626	0
▨	0085.02	608	10
□	0170.32	599	0
□	0083.50	572	0
□	0203.07	558	0
□	0082	500	0
□	0056	473	0
▨	0199.05	265	10
▨	0213.03	256	10
▨	0185.13	254	10
▨	0025.01	241	10
□	0083.12	228	0
□	0085.11	190	0
□	0108	187	0
□	0079.01	168	0
□	0083.39	144	0
□	0100.15	115	0
▨	0174.01	106	10
▨	0134.14	79	20
□	USS Nimitz	695	0
□	USS Ronald Reagan	694	0
□	USS John C Stennis	693	0
□	SSN Topeka	692	0
□	SSN Jefferson City	691	0
□	ARCO	690	0
□	USS Coronado	689	0
□	SSN Asheville	688	0
□	SSN Helena	687	0

Figure 3. BioDAC disease distribution display

More precisely, BioDAC has 16 federates that model a number of relevant factors. These include the population, modeled as individuals who can move from one location to another; the geography of the greater San Diego area; the dispersion of an aerosol plume containing the biological agent; the spread of the disease caused by the agent among the population; and the disposition of navy ships (whether they are in harbor or at sea). It also models the action of biological sensors and the laboratories that analyze the samples collected from them; the health surveillance system, which may provide an indication of an epidemic in progress; and the health care system, including how and where patients are treated and the distribution of prophylactic drugs from local stores and the Strategic National Stockpile (SNS).

BioDAC is designed for use in simulator-based exercises involving officials from the various interested military and governmental agencies. There are plans to extend it to make it capable of use as a decision support system during an actual event (for instance, for estimating the outcomes of various different hypothetical scenarios or decision paths.) BioDAC provides views of the simulated events that correspond to the information the PHO and NO would have access to in the course of an actual attack, including sensor inputs, lab test results and various indicators from the health surveillance system (numbers of patients with various symptoms reported by hospitals and emergency rooms, sales of particular types of remedies by pharmacies, etc.) The role of Analyst provides a "God's eye view" that is useful for viewing ground truth.

The decisions that the PHO or the corresponding Navy official can make include the ordering of lab tests, the collection of additional samples for testing, the distribution of prophylaxis and the closure of selected parts of the city by evacuation or sheltering in place. “Evacuate” means “no one can stay,” while “shelter-in-place” means “no one can leave.” The timing of the various actions the officials may take can have a crucial effect on the outcome in terms of the number infected and the number of deaths.



Mark Bastian @ logan: hey Mike!
Michael J. Procopio: hi
Michael J. Procopio: inflection points? what?
Michael J. Procopio: need-state-recovery
Mark Bastian @ logan: hey
Mark Bastian @ logan: well, I have the define and snaps loaders ready, was there another you wanted me to look at too?
Mark Bastian @ logan: As far as the threading goes, I will roll the changes back and let you deal with them.

John M. Linebarger @ secondary PC
(505) 845-8282
jmlineb@sandia.gov
Actions...

John M. Linebarger
(505) 845-8282
jmlineb@sandia.gov
Actions...

John M. Linebarger @ laptop
(505) 845-8282
jmlineb@sandia.gov
Actions...

Michael J. Procopio
(505) 845-9653
mjproco@sandia.gov
Actions...

Mark Bastian @ logan
(505)284-8208
mbastia@sandia.gov
Actions...

Mark Bastian @ nightcrawler
(505)284-8208
mbastia@sandia.gov
Actions...

Figure 4. N-ABLE™ collaboration, with awareness, chat, and screenboard

3. N-ABLE™

The NISAC Agent-Based Laboratory for Economics (N-ABLE™) is an agent-based economic modeling and simulation package. It consists of two components—an agent-based simulation engine that can execute either serially or in parallel, and a rich graphical user interface (GUI) that enables collaborative analysis of the simulation results. The N-ABLE™ GUI was the first software tool to use the GroupMeld™ collaboration framework to provide collaboration services inside of the N-ABLE™ application. Figure 4 above shows a screenshot of N-ABLE™ with several collaboration services visible—group awareness, public chat, and public screenboard with annotation capability.

A canonical simulation analysis cycle using N-ABLE™ iterates through four stages—modeling, simulation, analysis, and software development. First the simulation model is created using XML (eXtensible Markup Language). Next, the model is submitted to the simulation engine. Then the results of the simulation are validated via a sampling procedure—key outputs of various agent types for representative firms at each level of the supply chain are displayed graphically and shared with others using the screenboard collaboration capability. This validation process is called a "deep dive." Finally, if anomalous results are discovered, a review of the simulation software code is performed, which often results in code changes. This triggers a new iteration of the cycle, in which the simulation model is resubmitted to the simulation engine to run against the updated code, and the results revalidated.

Note that because of the impact of Hurricane Katrina, a model of the San Diego economy was not able to be created in time for the final experiment, so a proxy for N-ABLE™ was used instead of the full N-ABLE simulation engine (see section 8 below).

4. IDSim

The Interoperable Distributed Simulation (IDSim) framework provides the means to federate two or more autonomous simulators. The motivation for the development of IDSim was the mandate from the National Infrastructure Simulation and Analysis Center (NISAC) to combine "best of breed" infrastructure models. This translates into the need to integrate disparate simulations, distributed both geographically and organizationally, across a WAN (wide area network). To make this feasible, IDSim was developed with the following design goals in mind:

- Ease of integration
- Interoperability between platforms and languages
- Low usage of client resources
- Secure network communications
- Built from standard open technologies

Presently, IDSim is built on top of the reference implementation of the Open Grid Services Infrastructure (OGSI), Globus 3.2.1. This implementation uses Web Services technologies as a means of interoperating among different software applications running on heterogeneous platforms over a network; most often a WAN (wide area network).

IDSim builds on Globus 3.2.1 through inheritance of core functionality, including its Factory, ServiceGroupRegistration and ServiceGroupEntry services, among others.

These services provide the underlying basis for IDSim's FederateFactory, FederationFactory, Federation, Federate, and FederateEntry services. The factory services create instances of Federate, Federation and FederateEntry services. Each Federate service provides the interface for one federation-participating simulator. The Federation services manage a group of federation-member federate services, supporting the joining and leaving of the federation. The FederateEntry service provides an interface for each Federate service that has joined a federation. It acts as an entry point into the Federation service.

The "events" or messages that are sent between simulators are specified using XML and XML schema. These technologies provide a language-independent means of defining the data structure to be shared within the federation. Once the data structure is defined, it is compiled into any language that Web Services tools support.

Figure 5 shows the different architectural components of IDSim and their relationship to each other. The XML repositories provide the definitions to create the IDSim services and the data structures that are used in interactions among federates. The Grid Web Services Description Language (GWSDL) is compiled to Web Services Description Language (WSDL), and finally compiled into both client and server language stubs. Soft state refers to state that is kept during the federation run, and hard state is kept after the run has finished. The soft state associated with each of the services can be obtained synchronously through queries (pull model) or asynchronously through callbacks (push model). Note that if the federation has components that cross firewalls, only the pull model can be used; callbacks are not allowed through firewalls.

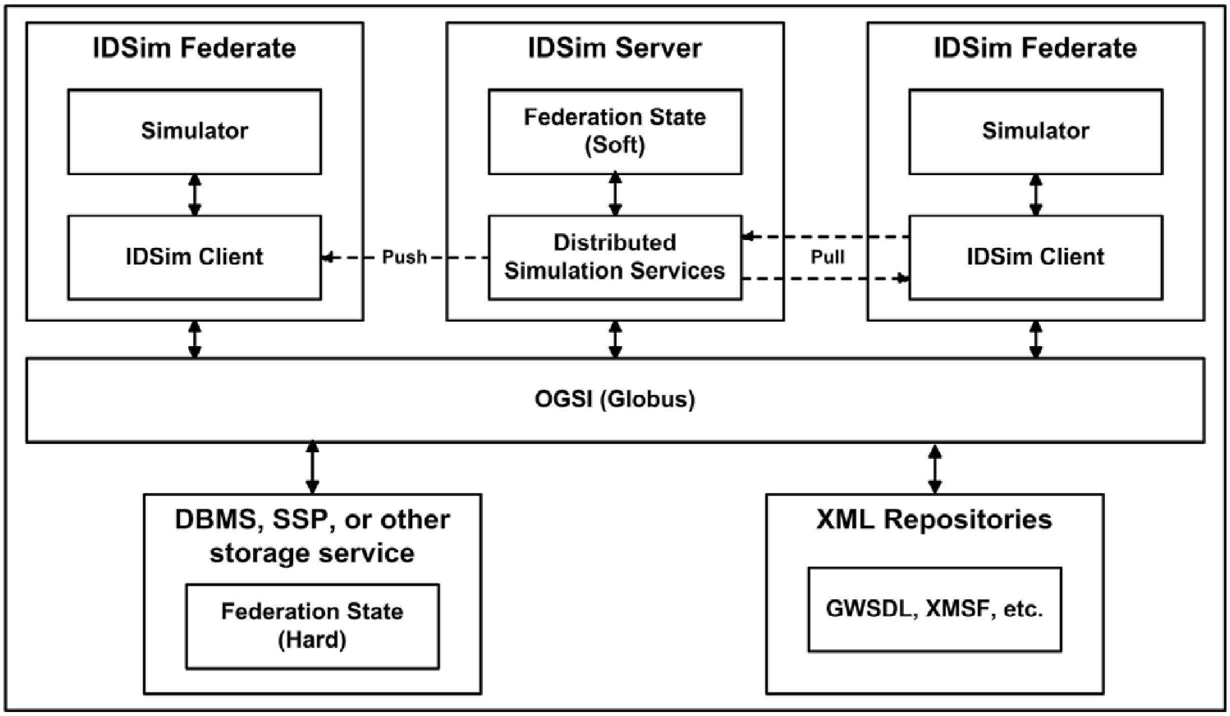


Figure 5. IDSim framework architecture

IDSim was used as the communication and time management component to send and receive time-stamped messages bi-directionally between BioDac and N-ABLE™. BioDac is implemented as an HLA (High Level Architecture)-based federation, with its own time management component. In this integration, it has two key components which provide the required time and data translations from representations within N-ABLE™, to representations within BioDac. The first such component is a specialized federate of the BioDac HLA federation, and the second is an IDSim-based federate, defined as the BioDac proxy. These two components communicate via sockets. The BioDac and the N-ABLE™ proxies communicate and are synchronized by the IDSim framework. Figure 6 below portrays the resulting hardware and software architecture of the LDRD demonstration.

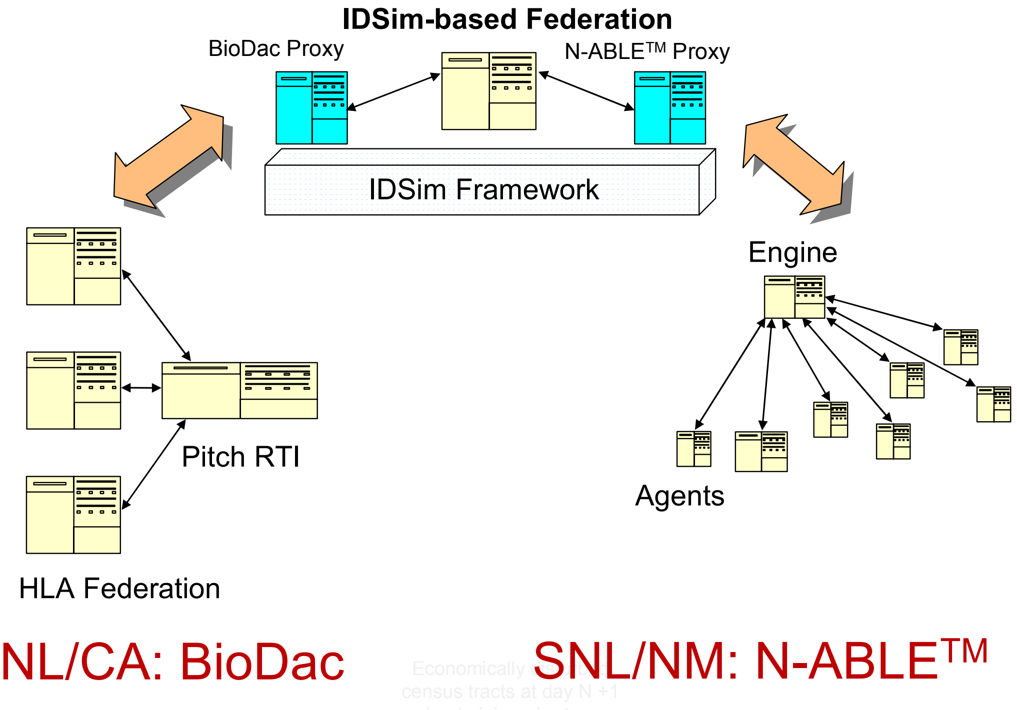


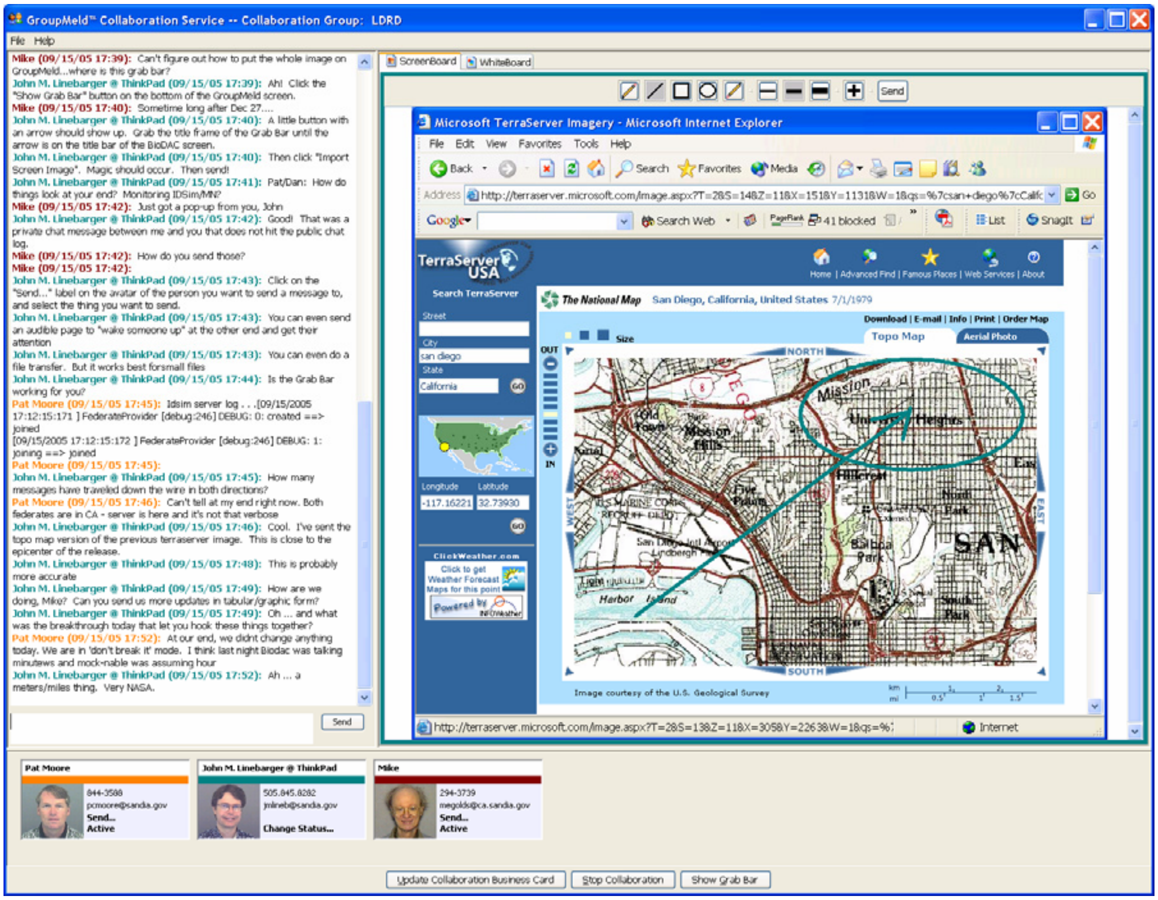
Figure 6. Hardware/Software Architecture for the LDRD demonstration

5. GroupMeld™

To set the stage for the discussion of the GroupMeld™ collaboration framework, some background is in order. The National Infrastructure Simulation and Analysis Center (NISAC), a program under the United States Department of Homeland Security's Information Analysis and Infrastructure Protection (IAIP) directorate, provides advanced modeling and simulation capabilities for the analysis of critical infrastructures, their interdependencies, vulnerabilities, and complexities. These capabilities help improve the robustness of critical infrastructures of the United States by aiding decision makers in the areas of policy analysis, investment and mitigation planning, education and training, and near real-time assistance to crisis response mobilizations.

NISAC and related programs are frequently called upon for Fast Analysis and Simulation Team (FAST) exercises to assess the impact of a potential event on critical infrastructures. The primary metrics for this high-pressure, time-constrained collaboration (which can be characterized as "collaboration in a crisis") are time to solution and quality of solution. A primary time consumer is the information exchange required to establish a common mental model (also called a "common analysis picture") of the problem(s) and solution(s) between all participants in the exercise

To support such FAST exercises, the GroupMeld™ software framework for synchronous collaboration has been developed. The goal of this framework is to facilitate real-time collaborative interaction, in order to allow geographically-distributed analysis teams to integrate multiple perspectives and quickly converge on a shared view of the problem(s) and potential solution(s).



Mike (09/15/05 17:39): Can't figure out how to put the whole image on GroupMeld...where is this grab bar?
John M. Linebarger @ ThinkPad (09/15/05 17:39): Ah! Click the "Show Grab Bar" button on the bottom of the GroupMeld screen.
Mike (09/15/05 17:40): Sometime long after Dec 27....
John M. Linebarger @ ThinkPad (09/15/05 17:40): A little button with an arrow should show up. Grab the title frame of the Grab Bar until the arrow is on the title bar of the BioDAC screen.
John M. Linebarger @ ThinkPad (09/15/05 17:40): Then click "Import Screen Image". Magic should occur. Then send!
John M. Linebarger @ ThinkPad (09/15/05 17:41): Pat/Dan: How do things look at your end? Monitoring IDSim/MN?
Mike (09/15/05 17:42): Just got a pop-up from you, John
John M. Linebarger @ ThinkPad (09/15/05 17:42): Good! That was a private chat message between me and you that does not hit the public chat log.
Mike (09/15/05 17:42): How do you send those?
Mike (09/15/05 17:42):
John M. Linebarger @ ThinkPad (09/15/05 17:43): Click on the "Send..." label on the avatar of the person you want to send a message to, and select the thing you want to send.
John M. Linebarger @ ThinkPad (09/15/05 17:43): You can even send an audible page to "wake someone up" at the other end and get their attention
John M. Linebarger @ ThinkPad (09/15/05 17:43): You can even do a file transfer. But it works best forsmall files
John M. Linebarger @ ThinkPad (09/15/05 17:44): Is the Grab Bar working for you?
Pat Moore (09/15/05 17:45): Idsim server log . . .[09/15/2005 17:12:15:171] FederateProvider [debug:246] DEBUG: 0: created ==> joined
[09/15/2005 17:12:15:172] FederateProvider [debug:246] DEBUG: 1: joining ==> joined
Pat Moore (09/15/05 17:45):
John M. Linebarger @ ThinkPad (09/15/05 17:45): How many messages have traveled down the wire in both directions?
Pat Moore (09/15/05 17:46): Can't tell at my end right now. Both federates are in CA - server is here and it's not that verbose
John M. Linebarger @ ThinkPad (09/15/05 17:46): Cool. I've sent the topo map version of the previous terraserver image. This is close to the epicenter of the release.
John M. Linebarger @ ThinkPad (09/15/05 17:48): This is probably more accurate
John M. Linebarger @ ThinkPad (09/15/05 17:49): How are we doing, Mike? Can you send us more updates in tabular/graphic form?
John M. Linebarger @ ThinkPad (09/15/05 17:49): Oh ... and what was the breakthrough today that let you hook these things together?
Pat Moore (09/15/05 17:52): At our end, we didnt change anything today. We are in 'don't break it' mode. I think last night Biodac was talking minutews and mock-nable was assuming hour
John M. Linebarger @ ThinkPad (09/15/05 17:52): Ah ... a meters/miles thing. Very NASA.

Pat Moore
844-3588
pcmoore@sandia.gov
Send...
Active

John M. Linebarger @ ThinkPad
505.845.8282
jmlineb@sandia.gov
Change Status...

Mike
294-3739
megolds@ca.sandia.gov
Send...
Active

Figure 7. Standalone GroupMeld™ collaboration client

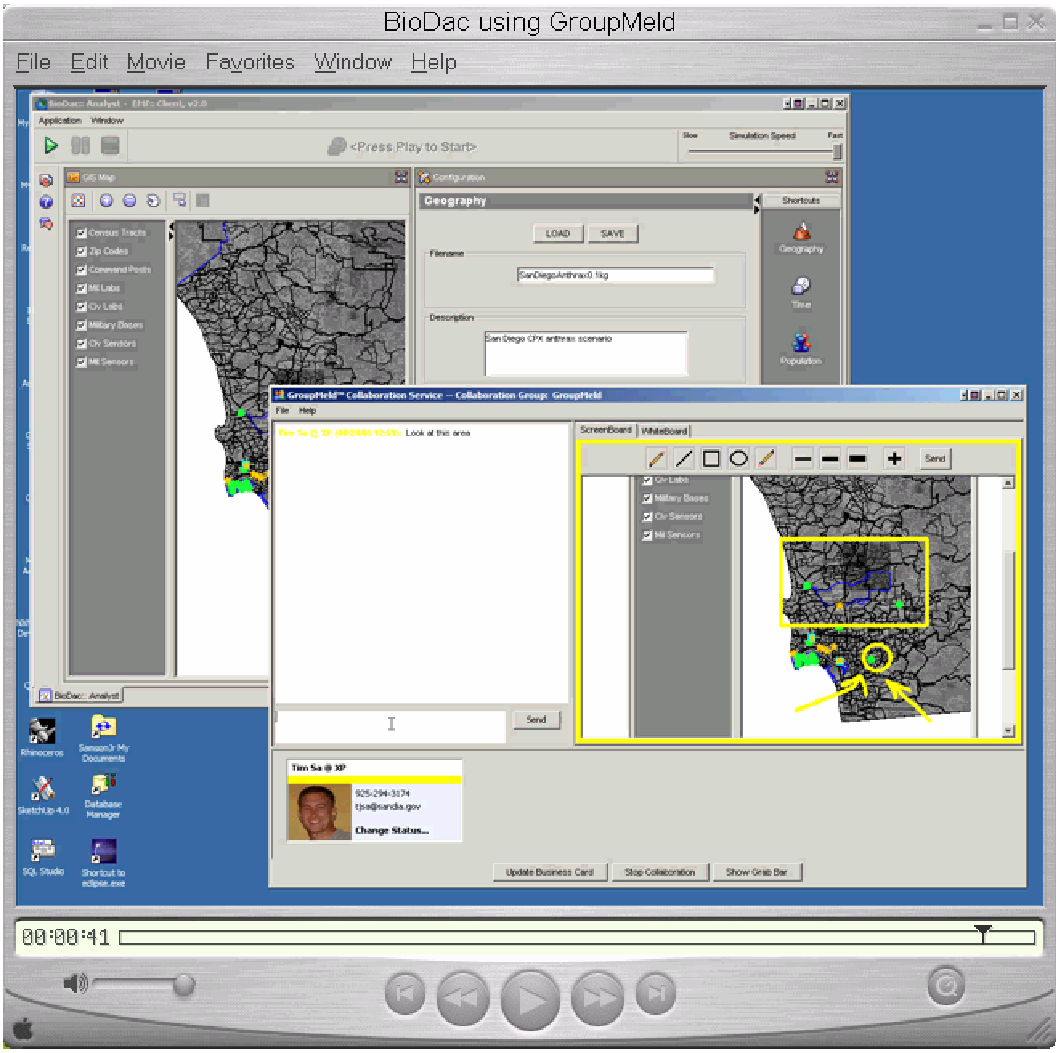
The collaboration capabilities provided by GroupMeld™ include:

- Pictorial awareness of other members of the virtual team, with visual status change indicators
- Real-time chat

- Shared screen images with collaborative annotation capability (a.k.a. “screenboard”)
- Shared whiteboard
- File transfer
- Audible paging capability (to get someone’s attention in case they are working on something else).

The collaboration scope of each capability is chosen from three levels, which can co-exist simultaneously:

- Full group (“public” collaboration)
- Subgroup (“restricted” collaboration)
- Person-to-person (“private” collaboration).



Look at this area

San Diego CPX anthrax scenario

925-294-3174
tjsa@sandia.gov

Figure 8. Invocation of GroupMeld™ from inside of the BioDAC simulation

Three usage models for GroupMeld™ have been observed so far.

- Programmatically embedded inside of a simulation application (*e.g.*, the way it is used by N-ABLE™ in Figure 4 above)
- Standalone GroupMeld™ collaboration client (see Figure 7 above)
- Hybrid (invoked inside the Java Virtual Machine of another simulation application, like BioDAC does in Figure 8 above). This approach allows panels from the simulation application to be dragged and dropped directly onto the GroupMeld™ screenboard.

GroupMeld™ was developed using the Java programming language, and has been deployed as a programmable collaboration library with an application programming interface (API). The library enables collaboration *through* a particular software application, thus forming an application-centered collaboration community. GroupMeld™ was developed in the Java programming language, and uses RMI over IIOP (Remote Method Invocation over the Internet Inter-ORB Protocol) as the distributed communication mechanism. The use of Java provides cross-platform portability—GroupMeld™ currently runs on Windows, Macintosh, and Linux computers. The framework is deployed as a set of Java packages in a single JAR (Java ARchive) file. The Java drag-and-drop API is used to drag a simulation graph or OpenGL (Open Graphics Language) image onto the screenboard panel. Each collaborator is both a client of and a server to all the other collaborators in the session, so the network topology is truly peer-to-peer. The communication functions are multithreaded, so reader-writer locks are used to protect shared data structures. An instance of the CORBA (Common Object Request Broker Architecture) Naming Service is used to keep track of all the participants in the collaborative session as well as their current subgroup structure. Subgroups can be nested to an arbitrary depth. A limitation of this architecture is that all computers must be on the same network or security domain; collaboration transactions cannot currently traverse a firewall.

Two optimizations are performed to reduce network traffic. Shared screen images are compressed in JPEG (Joint Photographic Experts Group) format prior to transmission and uncompressed at the receiving end. And annotations are handled by collecting the coordinates of all mouse-button-down events in a serializable Java object which is sent as soon as the mouse button is released; the annotations are then redrawn from the coordinates by the receiver.

6. Technical Design Considerations

A discussion of technical design issues follows, with respect to the integrating scenario and the HLA-IDSim gateways.

6.1 Scenario

A bio-attack has significant economic consequences as well as medical ones. In order to include such effects in the simulation, groups in 8100 and 6200 undertook a project to connect the BioDAC and N-ABLE™ simulation systems. IDSim, a distributed interoperable simulation framework, serves as an intermediary between the two. IDSim is a distributed operating system that provides a subset of HLA-based functionality. It is based on Grid technology, making it a reasonable locus for solutions to the issues that arise when combining systems that span different organizations and security domains.

Economic consequences naturally follow when people are removed from the workforce through disease. For this prototype effort, however, we decided to invoke the economic model only when the PHO or NO closes down parts of the city by ordering that the people in designated census tracts evacuate or shelter in place. Such actions affect relatively large numbers of people and can be expected to produce a noticeable economic effect.

Based on that decision, we could have chosen to couple BioDAC and N-ABLE™ loosely or tightly. In a loose coupling, BioDAC would send a request that N-ABLE™ start a run based on the affected census tracts and send the output back to BioDAC when it was finished. In a tight coupling, BioDAC and N-ABLE™ would run concurrently, synchronizing their simulation times with each other so that together they made up a composite discrete event simulation. We decided to couple them tightly, because that would give us a more general capability. It might not prove crucial in the present exercise, but it would allow us to model finer-grained features and would be a more flexible basis for any future work.

6.2 HLA-IDSim Gateways

BioDAC is written in Java and runs under the Pitch RTI (Run-Time Infrastructure), a commercial product offered by Pitch AB. We do not have access to the source code of the RTI, so whatever technique we chose to use to meld the two simulator systems (Pitch and IDSim) together would have to be done at the federate level. The design chosen was to add an extra federate to BioDAC which would serve as a "gateway" to the IDSim federation. The IDSim federation would consist of just two federates, one serving as a gateway to the BioDAC federation and one representing the N-ABLE™ simulator. A schematic representation is shown in Figure 9 below. As far as IDSim is concerned, it is just running two standard federates that communicate and synchronize with one another. IDSim itself is unaware that each of the federates communicates with another system. Likewise, the Pitch RTI just runs what it considers 17 standard federates (the original 16 plus the gateway) and is unaware that one of them has something going on the side. The "back door" communication between the BioDAC gateway federate and its IDSim twin goes through a TCP link. Including the BioDAC gateway federate (the *primary* gateway) in BioDAC's configuration file causes BioDAC to create the gateway when it starts up, and the primary gateway creates the gateway federate that runs in the IDSim federation (the *secondary* gateway).

We wanted to constrain the simulation time advancement in the Pitch and IDSim federations so that the requirements for conservative distributed discrete event simulation were maintained but did not want to force them to operate in lock-step, which would surely be very inefficient. Both of the gateway federates are time regulating, each in its respective federation. We achieved overall synchronization by letting each gateway inhibit the other from advancing "too far" ahead of it in synchronization time.

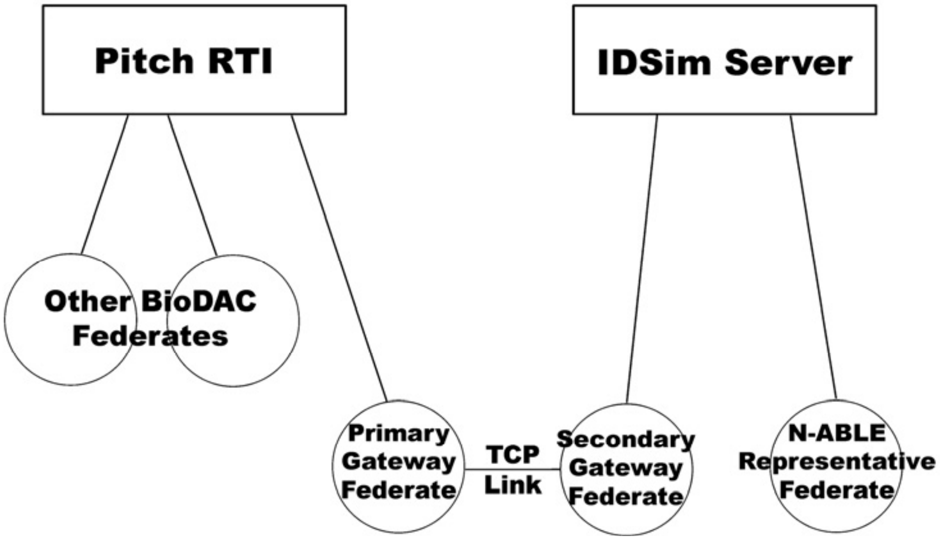


Figure 9. Schematic representation of HLA-IDSim gateways

Each federate has a lookahead value associated with it in its federation. Since messages must flow back and forth between BioDAC and N-ABLE™, it was necessary to consider what the overall lookahead of an event originating in one federation and delivered in the other might be. A federate wishing to send a message to a federate in the other federation would send the message (an *interaction* in HLA terms) into its own federation, and that federation's gateway federate would receive it. The gateway would then forward the message through the TCP link to the other gateway, which would send the message into the destination federation, where it would arrive at the destination federate. There must be a delay (in simulation time) between the sending of the message by the source federate and its receipt by the source federation gateway (at least) equal to the lookahead of the source federate in its federation. Similarly, there must be a delay between the sending of the message by the destination federation's gateway federate and its delivery at the destination federate (at least) equal to the lookahead of the destination federation's gateway federate in its federation. In addition, there could be some "slippage" in simulation time going through the TCP link between the gateways, since the two gateways do not operate in lock-step with one another. Thus the effective overall lookahead from the source federate to the destination federate is equal to the lookahead of the source federate (in its federation) plus the lookahead of the destination federate (in its federation) plus the greatest amount by which the destination gateway federate's simulation time when it receives the message through the TCP link can possibly be ahead of the source gateway federate's simulation time when it sends the message through the link.

For simplicity, assume that the lookahead of the source federate and the lookahead of the destination gateway federate both have the same value, **L**. In order to enable the two gateway federates to synchronize with one another, it is necessary for them to pass a notification through the TCP link to the other gateway every time they receive a time advance grant. Thus each gateway has an approximate idea (a lower bound) of the simulation time of the other gateway. Assume also that each gateway federate requests an advance in simulation time only when it believes itself to be strictly behind its twin in simulation time. Then if each request is granted in full, the gateways will progress in a "two-legged" fashion, first one moving ahead and then the other. (Of course, a request may not be granted in full because of the delivery of a message from another federate.) Further assume that the amount of each advance request is the federate's current simulation time plus **L**. (Each of the gateways has no internal events of its own, so there is no reason to request less than **L**.) Then the amount by which one gateway can be ahead of the other is at most **2L**. According to the reasoning of the preceding paragraph, as long as the application does not require that an inter-federation message arrive sooner than the simulation time of the originating federate plus **L + 2L + L = 4L**, the system can ensure on-time delivery of the message without violating either federation’s synchronization constraints. Note that this approach will work to couple any two simulators that have the concepts of lookahead, time advance requests and time advance grants.

The event loops of the two gateways are very similar, even though they operate in different federations and different simulation systems. The only events are time advance grants, notifications of time advances granted to the other gateway, and the arrival of messages tunneling one way or the other.

As part of the inter-gateway synchronization, the simulation times must be converted into the proper units for the particular system. BioDAC uses minutes as the unit of simulation time and N-ABLE™ uses days. In addition, there are data transformations to be considered. BioDAC sends its interactions with a single parameter which is a serialized Java object, typically an XML string in serialized form. The current version of IDSim has only one interaction type, and its data is a Java array of strings. The primary gateway federate handles all transformations of time and data.

We should expect a certain amount of overhead to be introduced by our technique of coupling the two simulators. For one thing, the frequency of synchronization is controlled by the lookahead values of the federates, as is the degree of concurrency in a multi-processor system. If the application-allowable lookahead between the source and destination federations for inter-federation messages is **K**, the scheme presented above forces us to use **K/4** as the lookahead in the Pitch and IDSim federations. In principle, this lowering of the lookahead value could lessen performance. In this particular case, however, the **K/4** value was still at least as great as the greatest lookahead of any BioDAC federate, so there were no performance consequences.

Another source of overhead comes from the fact that the primary gateway federate is held back from requesting a time advance grant until it has received notification from the secondary gateway that the secondary's simulation time is ahead of the primary's. The IDSim federation operates according to a Web services model in which a new HTTP connection must be made for every transaction (such as a time advance request and its subsequent grant) between a federate and the IDSim server, so we might reasonably expect the overhead from this source to be noticeable. To get an idea of the magnitude of the overhead, we made six runs, three with BioDAC alone and three with BioDAC and N-ABLE. Each stopped at the same simulated end time and ran to completion with no human interaction between the start and end of simulation. The results are presented in Table 1. The relative cost of this overhead depends on the amount of computation the application does between synchronizations. As Table 1 shows, the cost is quite low for this application.

Avg run time without synch. (sec)	**Run time std dev without synch.**	**Avg run time with synch. (sec.)**	**Run time std dev with synch.**	**Number of synchron-izations**	**Number of synchron-izations std dev**	**Avg overhead per synch. (sec.)**
1501.1	2.735	1535.3	3.117	1804	0	0.019

Table 1. BioDAC/N-ABLE™ synchronization overhead

If we had been able to get inside the Pitch synchronization protocol and the IDSim synchronization protocol (which we were prevented from doing by lack of access to the Pitch source code), we could have turned them into a single larger synchronization protocol including the federates of both federations, which might have been more efficient. We could then have enforced a federate's lookahead value inclusively over all federates. Instead of gateway federates, we might have used library code that would handle the necessary services. However, this would have required more design and programming time.

7. Performance Measurements

Performance measurements were captured for both components of the integrating architecture—IDSim and GroupMeld™—in order to describe the performance profile of the use of these architectures for various types of transactions.

7.1 IDSim

IDSim architecture performance measurements were obtained from 10 runs of processes emulating the BioDAC and N-ABLE™ simulators. These processes were federated through the IDSim framework. Two machines were used for this purpose: One machine was an i686, running Linux 2.4.20-19.8, and the other machine was an i686, running Microsoft Windows Server 2003. The IDSim server and the N-ABLE™ emulator ran on the Linux machine, and the BioDAC emulator ran on the Windows Server machine. Both machines are within the Sandia Restricted Network. Note that as far as the performance measurements are concerned, there is no difference between running the actual N-ABLE™ and BioDAC simulations, and their emulator counterparts.

These measurements were obtained by timing the roundtrip of each of the messages sent between the emulators, and using half of this value as the one-way latency. This made possible the usage of one system clock per each measurement, avoiding the issue of clock synchronization between machines. The system clocks of both machines were used, corresponding to the time-managed messages sent by each emulator. The return message to calculate the roundtrip value was sent in receive-order mode, so as to not change the functional characteristics of the federation. The messages of all the runs are aggregated to derive the statistical values displayed below in Table 2. Note that the measurements obtained include the marshalling and unmarshalling stages that the messages undergo as they are sent and received between emulators.

Number of Msgs Sent	**Avg Msg Size (bytes)**	**Msg Size Std Dev (bytes)**	**Min Msg Size (bytes)**	**Max Msg Size (bytes)**	**Avg Δt (ms)**	**Δt Std Dev (ms)**	**Min Δt (ms)**	**Max Δt (ms)**
117	2143	1540	0	5952	154.27	239.99	66.5	1648.5

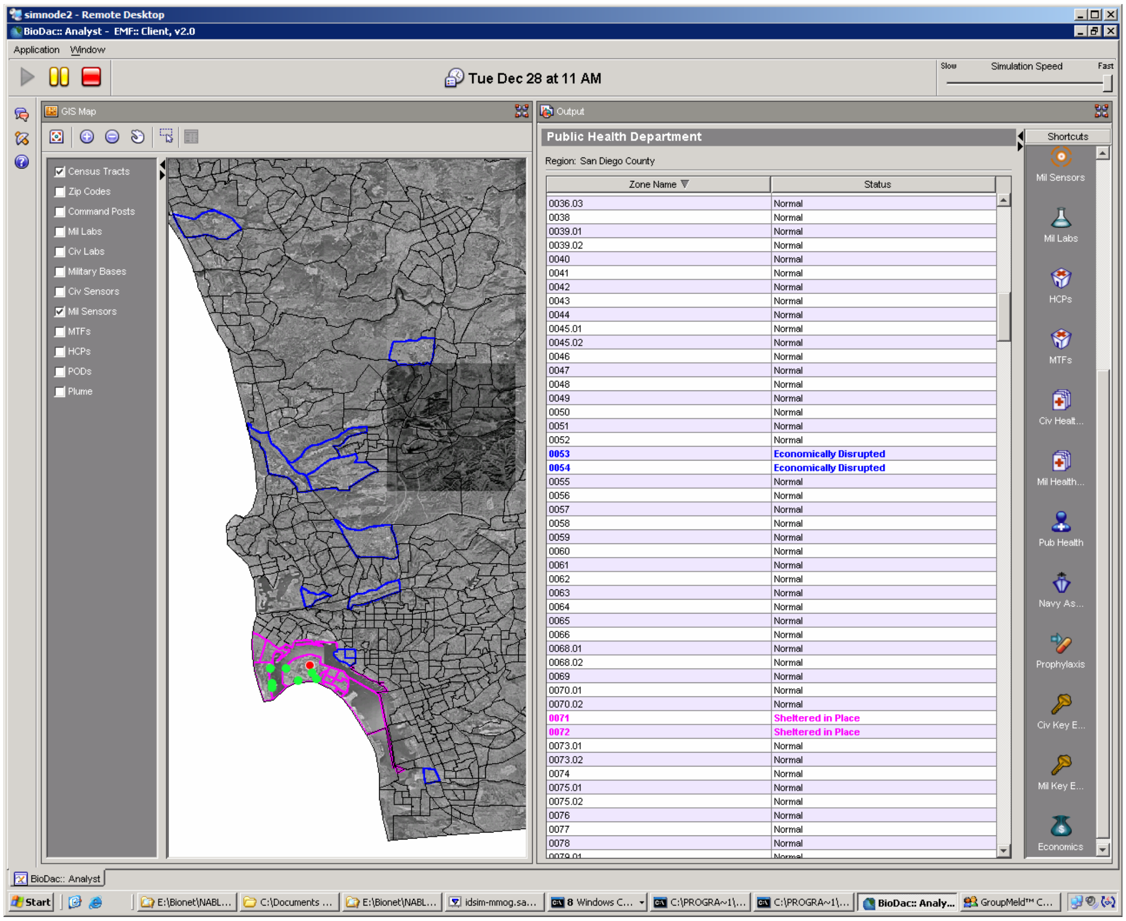
Table 2. IDSim message latency measurements

7.2 GroupMeld™

GroupMeld™ collaboration transaction performance was measured during a 90-minute trial run of the pilot experiments between BioDAC and N-ABLE™ (see section 8 below). Four members were present in the GroupMeld collaboration session, one located in Sandia California, and three located in Sandia New Mexico. All machines were on the SRN (Sandia Restricted Network). Table 3 below summarizes the performance in milliseconds of various kinds of collaboration transactions, where a transaction is defined as the transmission of a particular kind of multimedia collaboration artifact to all *other* members of the collaboration session.

Collaboration Transaction Type	**Count**	**Avg. Millseconds**
Chat message	184	126
Screenboard or whiteboard annotation	45	80
Lock screenboard	12	152
Screenboard image	12	935
Status update	8	49
Clear all whiteboard annotations	2	156
Total	**263**	**154**

Table 3. Sample GroupMeld™ collaboration transaction performance



Public Health Department

Region: San Diego County

Zone Name	Status
0036.03	Normal
0038	Normal
0039.01	Normal
0039.02	Normal
0040	Normal
0041	Normal
0042	Normal
0043	Normal
0044	Normal
0045.01	Normal
0045.02	Normal
0046	Normal
0047	Normal
0048	Normal
0049	Normal
0050	Normal
0051	Normal
0052	Normal
0053	**Economically Disrupted**
0054	**Economically Disrupted**
0055	Normal
0056	Normal
0057	Normal
0058	Normal
0059	Normal
0060	Normal
0061	Normal
0062	Normal
0063	Normal
0064	Normal
0065	Normal
0066	Normal
0068.01	Normal
0068.02	Normal
0069	Normal
0070.01	Normal
0070.02	Normal
0071	**Sheltered in Place**
0072	**Sheltered in Place**
0073.01	Normal
0073.02	Normal
0074	Normal
0075.01	Normal
0075.02	Normal
0076	Normal
0077	Normal
0078	Normal
0079.01	Normal

Figure 10. Census tracts economically disrupted by a quarantine event

8. Results of Small Pilot Experiment

A small pilot experiment was performed in order to investigate the use of these integrating architectures in exploring crisis management strategies during a bioterrorism attack. Three strategies were explored—the use of BioDAC alone; the integration of BioDAC and N-ABLE™ in order to determine which census tracts have been economically disrupted by quarantine actions taken by a PHO, which are subsequently left alone; and the integration of BioDAC and N-ABLE™ in which a PHO immediately shelters in place census tracts that are determined by N-ABLE™ to be economically disrupted. Figure 10 above displays the resulting BioDAC screen when a list of economically disrupted census tracts is received from N-ABLE™. In the figure, census tracts that were sheltered in place by the PHO are outlined in magenta, and census tracts that were determined by N-ABLE™ to be economically disrupted, because they were dependent on labor in the quarantined census tracts, are outlined in blue.

A change to the BioDAC population movement model was necessary in order to treat economically disrupted census tracts differently than census tracts that are evacuated or sheltered in place. The updated population movement model does not allow people to enter an economically disrupted census tract in order to go to work. However, other types of population movement are allowed.

Response Strategy	**Population Category**	**Percent change from running BioDAC alone**
Determine which census tracts are economically disrupted	Avg. Infected/No Symptoms	0
	Max. Infected/No Symptoms	0
	Avg. Mildly Infected	0
	Max. Mildly Infected	0
	Avg. Severely Infected	0
	Max. Severely Infected	0
	Recovered	0
	Dead	0
	Immune	0
Determine which census tracts are economically disrupted and immediately shelter them in place	Avg. Infected/No Symptoms	-3
	Max. Infected/No Symptoms	-4
	Avg. Mildly Infected	0
	Max. Mildly Infected	2
	Avg. Severely Infected	2
	Max. Severely Infected	1
	Recovered	-13
	Dead	-20
	Immune	0

Table 3. Results from small pilot experiment

Because of the impact of hurricanes Katrina and Rita on staff availability, a model of the San Diego economy could not be created in time for the experiment, so the N-ABLE™ simulation proper was not used. Instead, a proxy for N-ABLE™ was developed that took a list of census tracts that were quarantined by a PHO and randomly selected other census tracts that were likely to be economically disrupted by the quarantine due to labor dependencies.

Considerable expertise is required to play the role of a PHO in a BioDAC simulation. As a result, it was challenging to perform simulation runs that were consistent enough to allow valid comparison of the three factor levels of the experiment. Only one full experiment cycle (three runs), by an experienced BioDAC user, met that criteria. The results are displayed in Table 3 above.

It must be stressed that the results are not statistically valid for at least two reasons: because of the extremely small sample size, and because of the use of a proxy for the N-ABLE™ simulation engine instead of the real N-ABLE™ simulation running against a full agent-based model of the San Diego economy. However, the significance of the experiment is that the integrating architectures that were developed and demonstrated as a result of the LDRD allow a far greater range of crisis management options to be explored and evaluated than would be possible without the use of the integrating architectures.

9. Future Work

The most obvious path forward for future work is to involve the real N-ABLE™ simulation engine instead of the N-ABLE™ proxy. Since the operating hypothesis of the experiment scenario was that the economy itself can contribute to the dampening of a pathogen, exercising this path forward is strongly encouraged, since it may lead to significant, publishable, results. Such results would be of interest not only to the crisis management community, but also to the economic simulation and distributed simulation communities. A secondary path forward would be to involve other simulations from other National Laboratories in another integrating scenario. The Interdependent Energy Infrastructure Simulation System (IEISS) from Los Alamos National Laboratory is an excellent candidate.

10. Related Publications

Ammerlahn, H.R., M.E. Goldsby, M.M. Johnson, & D.M. Nicol. (2000). "A Geographically Distributed Enterprise Simulation System." *Future Generation Computer Systems* 17, 135-146.

Eidson, Eric D., & Mark A. Ehlen. (2005). "NISAC Agent-Based Laboratory for Economics (N-ABLE™): Overview of Agent and Simulation Architectures." *Sandia National Laboratories Technical Report SAND2005-0263*, Albuquerque, New Mexico.

Fitzgibbons, J. Brad, Richard M. Fujimoto, Daniel Fellig, Stephen D. Kleban, & Andrew J. Scholand. (2004). "IDSim: An Extensible Framework for Interoperable Distributed Simulation." *Proceedings of the IEEE International Conference on Web Services (ICWS'04)*, June 6-9, San Diego, California, USA, 532-539.

Hawley, Marilyn, Heidi Ammerlahn, Jason Arnold, Pamela Barr, Michael L. Bernard, Midge L. Davis, Donna Djordjevich, Michael Johnson, Timothy Sa, Ricky Tam, William B. Wilcox, & Aliseya Wright. (2004). "Understanding Communication in Counterterrorism Crisis Management." *Sandia National Laboratories Technical Report SAND2004-5390*, Livermore, California, USA.

Linebarger, John M., Andrew J. Scholand, Mark A. Ehlen, and Michael J. Procopio. (2005). "Benefits of Synchronous Collaboration Support for an Application-Centered Analysis Team Working on Complex Problems: A Case Study." Accepted for publication in *Proceedings of the 2005 International ACM SIGGROUP Conference on Supporting Group Work (Group'05)*, November 6-9, Sanibel Island, FL, USA.

Linebarger, John M., Andrew J. Scholand, & Mark A. Ehlen. (2006). "Representations and Metaphors for the Structure of Synchronous Multimedia Collaboration within Task-Oriented, Time-Constrained Distributed Teams." Accepted for publication in *Proceedings of the 39th Hawaii International Conference on System Sciences (HICSS 39)*, January 4-7, Kauai, HI, USA.

Manley, D.K., M.W. Koch, S.A. McKenna, R.L. Bilisoly, M.W. Trahan, A.L. Sobel, D.D. Djordjevich, M.E. Goldsby, E.L. Hoffman, M.M. Johnson, T. Nguyen, T.J. Sa, H.R. Ammerlahn, M.F. Hawley, W.B. Wilcox, & A.S. Yoshimura. (2003). "Enabling Analytical and Modeling Tools for Enhanced Disease Surveillance." *Sandia National Laboratories Tech. Report SAND2003-8154*, Livermore, CA.

Yang, Lynn I., Dawn K. Manley, Susanna P. Gordon, Heidi R. Ammerlahn, Donna D. Djordjevich, Donna M. Edwards, Mike E. Goldsby, Marilyn F. Hawley, Ricky Tam, Rose P. Tsang, & Ann S. Yoshimura. (2004). "Summary of the 2003 San Francisco International Airport Bio-Defense Preparation Exercise." *Sandia National Laboratories Technical Report SAND2004-2225*, Livermore, California.

Distribution

1	MS 9201	Michael M. Johnson, 8114
1	MS 1138	Steven D. Kleban, 6226
1	MS 1138	John M. Linebarger, 6224
2	MS 9018	Central Technical Files, 8945-1
2	MS 0899	Technical Library, 9616
1	MS 9021	Classification Office, 8511 for DOE/OSTI
1	MS 0123	D. Chavez, LDRD Office 1011

Sandia National Laboratories