MODELING AND SIMULATING AN ARMY INFORMATION SUPPORT STRUCTURE

Major Bernard J. Jansen
D/Electrical Eng & Computer Science
United States Military Academy
West Point, NY 10996
jansen@ exmail.usma.edu

Lt.Col. John A. Hamilton, Jr.
Ada Joint Program Office
Defense Information Systems Agency
Arlington, VA 22204-2199
dj7560@eecs1.eecs.usma.edu

Dr. Udo W. Pooch, P.E.
D/ of Computer Science
Texas A&M University
College Station, TX 77843
pooch@cs.tamu.edu

KEYWORDS

Network modeling, network simulation, systems integration, Network Manager's Workbench TCL/TK

ABSTRACT

The Workbench is a component of a larger objective to build a Network Manager's Toolkit. The larger objective is to develop an integrated set of tools that will support eight network performance areas. The Workbench is the component that integrates this set of tools by providing the means for communication with each other. This paper presents a brief overview and background of the Network Manager's Toolkit and the specific design goals of the Workbench component.

INTRODUCTION

The United States Army faces the prospect of maintaining both tactical and base operations (BASOPS) information systems. Current operation plans envision that deploying units will take their tactical information systems with them. However, Army posts, camps and stations will continue to operate and will continue to need information systems to operate the infrastructure. More importantly, the needs of the BASOPS information infrastructure will change as Regular Army units deploy and Reserve and National Guard forces are activated and assigned to their mobilization stations.

How can we predict the BASOPS information systems requirements of an Army post whose wartime requirements will change as frequently as the tenant units? This is clearly a prime domain for simulation study. To conduct these studies, the proper tools are needed.

Information systems modeling and simulation tools are becoming more popular and more affordable (Hamilton et al. 95). However, most of these tools are still specialized, standalone tools. An inexpensive and integrated suite of modeling and simulation tools is required to make BASOPS information system modeling and simulation operationally practical. The Network Manager's Workbench is a network application program interface for integrated network performance management. Additionally, it is available at no cost.

This paper presents a brief overview of the Network Manager's Toolkit and the specific military applications of the Workbench component. The paper also outlines the various languages and programming packages used and Workbench's design algorithm. The paper concludes with some general comments about the project and areas for future improvements and potential for Army-wide deployment.

The proliferation of information networks in the military concurrent with a sustained reduction in military resources has created the need to efficiently manage network resources. Army information systems are often expected to enhance operational efficiency and thereby generate cost savings. Unfortunately, these efficiencies and savings are often not realized due to inefficient implementation and operation of the system.

For those charged with operating military computer networks, simulation is often of academic but not of operational value. To efficiently operate an information network, the network manager must collect information regarding the operation of the network and organize it in a management information database. The network manager uses this information to manage performance, faults, accounting, configuration and security (Caruso 90; Stevenson 96).

Using this information for simulation purposes is problematic. (Terplan 92) observes network simulation models are very sensitive to:

- Workload descriptions
- Network configuration

Initially, the development of this project was entirely in the UNIX environment. The designers did not consider other environments until early 1995. Since then, ports of the Netman tools to MS/DOS and MS/Windows have begun. At present, implementations of the Netman tools are on Sun SPARC stations (SunOS and Solaris), DECstation 5000 series (Ultrix), Alpha workstations (OSF/1), as well as Silicon Graphics IRIS (IRIX) systems. There are plans for ports to other workstations including IBM RS/6000, HP7000 series, and PC-based UNIX systems such as Linux.

Current efforts have concentrated on monitoring Ethernet networks; however, all the tools are able to monitor and manage other LAN technologies. The monitoring tools use the IP protocol. However, the design takes into account other widely used protocols, particularly SNA, Novell Netware and DECnet.

Listed below are some of the current projects being carried out within the Netman Project:

1. Etherman: Ethernet Traffic
2. Interman: IP Connectivity
3. Packetman: Packet Analysis
4. Geotraceman: Geographical Traceroute
5. Loadman: Network Load Monitor
6. Analyzer: LAN Segmentation

Netscope

Netscope (Nafis 94) is an extensible and flexible piece of code that can graphically display a network topology. The following are the capabilities of the Netscope program:

1. Network Components Display: The program can graphically represent discovered network components. The components are routers, hosts and networks. The program specifies routers as connecting components, showing links to all the networks that they connect. The program displays networks connected to a router as clouds.
2. Network Traversal: Netscope has an interactive display. When clicked upon, a cloud icon will show the network pointed to by the cloud. A user can traverse the whole network by stepping through cloud icons one-of-one or jumping directly to a new network.
3. Get network type: Netscope can obtain the network type (Ethernet, Token Ring, FDDI, etc.) by issuing SNMP queries to the devices on that particular network.
4. Display different topologies: The program can display different kinds of networks. The program displays an Ethernet as a straight line with terminators. It displays a token ring as a circle with devices connected all around and an FDDI ring as a double ring. Netscope displays a Star as a concentrator in the center with connections all around and a Token Bus as a straight line with a grounded terminal on one side. The program displays an unknown net type as a straight line with no end connectors.

Security Issues

There is a security issue with the Workbench component as with any TCL/TK application. The impact on military networks will vary depending on the operational environment of each network. The concern centers on the *send* command, which accepts an interpreter, command, and command arguments. This command arranges for the interpreter named by the interpreter variable to execute the commands and arguments. The command returns the result or error from that command execution. The interpreter variable must be the name of an interpreter registered on the display and associated with the application that invoke the command. This interpreter need not be within the same process or application. If no arguments are present, then the command argument entirely contains the command. If one or more arguments are present, they are concatenated to form the command to be executed, just as for any evaluated TCL command.

The *send* command is potentially a serious security loophole. Any application that can connect to the server can send scripts to the applications using the server. These incoming scripts can use TCL/TK to read and write to files and invoke subprocesses under the invoker's name.

Host-based access control heightens the security risks. Host-based access control, such as that provided by xhost, is particularly insecure. It allows anyone with an account on particular hosts to connect to the host server. If disabled, it allows anyone anywhere to connect to the host server.

To provide at least a small amount of security, TK checks the access control being used by the server and rejects incoming *sends* unless:

1. The xhost-style access control is enabled (i.e., only certain hosts can establish connections) and
2. The list of enabled hosts is empty. This means that applications cannot connect to your server unless they use some other form of authorization such as that provided by the xauthority command.

monitoring program, and then enter the necessary node data into the packet monitoring program.

Clearly, this situation is not acceptable for military applications. The transferring of data and separate invocation of programs reduces the productivity of the network manager and increases the opportunity for errors. Furthermore, it makes network simulation impractical. It also requires a high degree of technical knowledge about the current state of the network, which may not be possible in a turbulent personnel situation.

Still, tools developed by the research community offer several important advantages along with the disadvantages:

Advantages	Disadvantages
Cost	Ease of use
Full access to source code	General applicability
Extensibility	Support

Table 1: Pros and Cons of Research Tools.

Openness, the ability to modify the tools as needed is a critical advantage over proprietary systems. This open systems approach is what makes the Network Manager's Toolkit practical. Although far from perfect, the Network Manager's Toolkit also makes an attempt at addressing ease of use by offering a menu driven graphical user interface (GUI).

The Network Manager's Toolkit can provide the basis for an Open Simulation Architecture by providing the systems integration necessary to support network simulation on an operational vice an academic basis. The NMWB is a tool that provides the system integration component of the Network Manager's Toolkit.

AN INTEGRATED TOOL FOR NETWORK MODELING AND SIMULATION

A specific goal of the NMWB is to provide an integrated network modeling and management system that is application neutral. To accomplish this goal, the Workbench uses an object oriented design. Separate objects represent each network problem-solving program in the Workbench. The objects can be any performance management tools available to the network manager. One can also transfer the Workbench concept to network management areas other than performance, such as security, fault analysis, configuration, and accounting management. For the military network administrator, this means that the Workbench is expandable and can accommodate network tools that the administer may currently be using.

The Workbench consists of two components. The first component is a function that accepts and passes information between objects through a standard communication interface protocol. The second component is a GUI that gives the user access to the resulting information.

In simple terms, the Workbench is the communication link and GUI between objects and provides the user a uniform working environment. This reduces cost in terms of training time and improves operational efficiency. Figure 2 illustrates the major components and information flow of the Workbench.

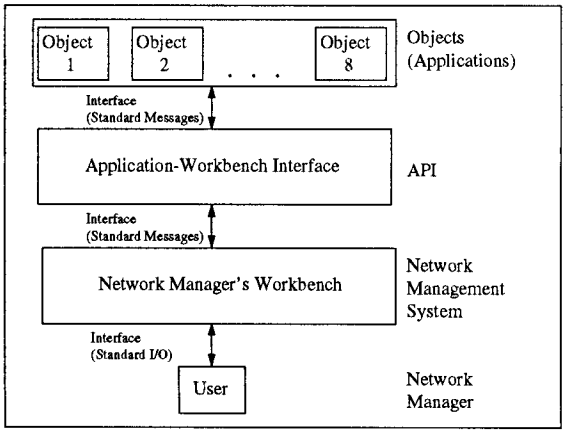

Figure 2: Major Components for the Workbench.

Along with the objects, the Workbench defines an interface structure among performance applications. Although all the specific structures are not yet defined, the matrix in Figure 3 illustrates the overall concept of the Workbench data structures.

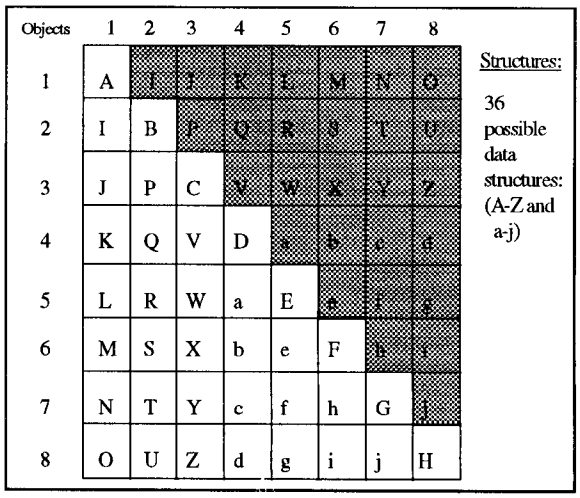

Figure 3: Data Structures and Object Relationship.

There is one data structure for the Workbench, eight application data structures, and twenty-eight intersection data structures. The Workbench structure contains all information in all eight object structures. The eight application structures (A through H) contain all information that the object accepts and returns to the Workbench. The twenty-eight intersection structures (I through Z; a through j) contain the common information between pairs of object structures.

The Workbench allows the network manager to move quickly and seamlessly from one object to another object in the Workbench. Rather than use a MIB, the Workbench establishes a common interface between objects within the Workbench to accomplish this data movement

IMPLEMENTING THE WORKBENCH

The Workbench provides a uniform working environment that increases a network manager's efficiency and productivity when using a variety of network problem-solving programs. The Workbench creates a uniform working environment by establishing a standard interface between these stand-alone, problem-solving programs. In other words, the Workbench allows independent applications to communicate with one another by breaking out of the paradigm that restricts each separate application.

The Workbench currently accepts five objects but is extendible. These five objects are: a network topology package, an Ethernet monitoring package, an Internet Protocol (IP) monitoring package, a network packet tracing package, and a launcher platform for the previous three packages. All are freely available to the military community. The total array of available tools is dependent on the specific packages invoked.

For GUI, the programming code is in TCL/TK and C (Schonwalder and Langendofer 96). For the communication function, the programming code is in TCL/TK, and C, along with the TCL/TK extension XtSend (Newmarch 96). The next two paragraphs briefly discuss the extensions to the TCL/TK programming language.

TCL (Newmarch 96, Outerhout 94) is a type-free interpreted language that is both well suited for GUI development and easy to use. It is designed to be embedded in applications that require a "command language." TK is a set of widgets built above TCL that provide a quick and easy route into X-Window programming. As such, the user community has developed several TCL/TK extensions that perform specific functions.

One of the features (Newmarch 96) of TK is the *send* command. This is a communications protocol based on X-Window properties that allows one TK application to send TCL commands to another TK application. The second TK application accepts the commands and executes them. For example, this communication feature allows:

1. A debugger to send commands to an editor to display the region being debugged.
2. A supervisor to send shutdown messages to all TK applications to gracefully terminate.
3. A file manager to send a selected file to a program that knows how to handle it.

MODELING AND SIMULATION WITH NMWB: INTEGRATING THE TOOLKIT APPLICATIONS

The NMWB supports network modeling and simulation through its integrated applications. These applications provide the analyst with the modeling information needed to apply simulation as an operational tool. Netman and Netscope are two tools which provide can provide the traffic and topology data necessary to conduct a successful network simulation

Netman

The Netman project (Netman 96) is an ongoing research project in the School of Computing at Curtin University of Technology - Perth, Western Australia. The Department of Computer Science at Texas A&M also has an ongoing project to enhance the applications within the Netman project (Nafis 94). The primary objective of this project is to provide tools and techniques for effective network management within small-to-medium sized organizations. The initial motivation for this project was the belief that management of small-to-medium sized networks need be inexpensive and a relatively simple undertaking.

A number of tools have already emerged as a result of this research. Each has a different objective and graphical representation. These collectively form the Netman suite of tools, three of which are Etherman, Interman, and Packetman. The Netman tools were first released in April 1993. The programming language is X-Windows with Athena widgets.

CONCLUSIONS

The NMWB serves as an important systems integration tool that enables military network managers to use efficient network simulation and management at a reasonable cost. It allows the network manager to seamlessly move from one performance application to another. The Workbench provides strong support in the areas of user interface design, error checking, robustness, hypertext and net linked help system, and use of tested, freely available language packages.

The NMWB is focused on garrison operations because the component tools are designed for BASOPS and not tactical systems. It is perfectly reasonable to expect that the NMWB could integrate the Tactical Internet Modeling System (TIMS) currently under development. The integration of the completed TIMS would provide a gateway to model the interactions of tactical and BASOPS information systems. Further, the NMWB could serve as an integrating platform for tactical network management, modeling and simulation tools.

Too often in today's Army, organizations are given missions without resources. One result of constrained resources is that it will be a long time before information systems on Army posts, camps, and stations are standardized. For the foreseeable future, network architectures throughout the Army will be heterogeneous in nature. The absence of a uniform systems architecture means that there are no "one size fits all" solutions. Rather each Army post will have varying tool requirements to enable them to model and manage their information infrastructure.

To bring the power of simulation to bear, it is critical that network operators are provided the right tools to make network simulation a realistic option. The NMWB provides a foundation for using simulation as an operational means of managing diverse BASOPS networks and information systems.

REFERENCES

Caruso, Richard E., *Network Management: A Tutorial Overview*, IEEE Communications Magazine, Vol. 28 March 1990, pp. 20-25.

Hamilton, J.A., Jr.; Nash, D.A., Pooch, U.W. 1997. *Distributed Simulation*, CRC Press, Boca Raton, Fla..

Hamilton, Maj. Drew, Gary Ratterree, CAPT. Paul Brutch, Anish Karmarkar, Shridhar Muppidi, Chris Cunningham, Eric Fisch, and Dr. Udo Pooch. *Getting Inside Your Local Area Network: Tools, Techniques, and Resources*. Crosstalk. (January 1995).

Nafis, Ahmad, Netscope: A Network Topology Viewing Tool. Unpublished paper, Fall 1994. Computer Science Department, Texas A&M University.

Netman information obtained from site: http:// Newmarch, Jan. (jan@pandonia.canberra.edu.au). The primary site for this is ft.canberra.edu.au. It may also be placed on ftp.x.org and harbor.ecn.purdue.edu.

Ratterree, Gary, Udo Pooch, and Willis Marti. Etherman Version 1.2: A Network Manager's Workbench Application. Technical Report 95-028. Unpublished report. Computer Science Department, Texas A&M University.

Schonwalder, J. and H. Langendofer. TCL Extensions for Network Management Applications. From: http://www.cs.tu.de/ibr/projects/

Stevenson, Douglas W. Network Management: What it is and what it isn't. From Dstevenson@tribune.com.

Terplan, K., *Communication Networks Management, 2e*, Prentice-Hall, Inc., Englewood Cliffs, 1992.

Wood, David C. M., Sean D. Coleman and Michael F. Schwartz, document at site http://library.usask.ca/hytelnet/net/net019.html

Wood, David C. M., Sean D. Coleman and Michael F. Schwartz, *Fremont: A system for discovering Network Characteristics and Problems*, 1993 Winter USENIX, San Diego, CA, pp. 335-348.
www.cs.curtin.edu.au:80/~netman/etherman.html

Major Jim Jansen is currently assigned to the Department of Electrical Engineering and Computer Science at the United States Military Academy. He is also a Ph.D. Candidate at Texas A&M University. Major Jansen has a B.S. in Computer Science from the United States Military Academy. Additionally, he has a Master of Computer Science from Texas A&M University and a M.S. in International Relations from Troy State University. He has served in numerous military communication assignments in the US and Europe. His research interests and expertise include networks, information retrieval, software agents, and computer-human interaction. He is currently conducting research in the combined use of software

agents and information search engines. Email: jansen@exmail.usma.edu.
Web Site: http://www.eecs.usma.edu/usma/academic/eecs/instruct/jansen/

Lieutenant Colonel John A. (Drew) Hamilton, Jr., US Army, completed his Ph.D. in Computer Science at Texas A&M University and is currently assigned to the Department of Electrical Engineering and Computer Science at the US Military Academy. He was previously Chief, Officer Training Division at the Computer Science School, Fort Gordon. Lt.Col. Hamilton has a B.A. in Journalism from Texas Tech University, where he was commissioned in Field Artillery; an M.S. in Systems Management from the University of Southern California and an M.S. in Computer Science from Vanderbilt University. Lt.Col. Hamilton is a graduate of the Naval War College with distinction. Email: dj7560@eecs1.eecs.usma.edu

Dr. Udo W. Pooch, PE, received his Ph.D. in Theoretical Physics from the University of Notre Dame (1969) and is the E-Systems Professor of Computer Science at Texas A&M University. Dr. Pooch is the author of numerous articles and books, and most recently authored Distributed Simulation with Lt.Col. J.A. Hamilton, Jr. and Major D.A. Nash, published by CRC Press. Dr. Pooch is a very active researcher supervising projects in network simulation, network security and fault-tolerant distributed environments. Email: pooch@cs.tamu.edu