Proceedings of the 1993 Winter Simulation Conference
G.W. Evans, M. Mollaghasemi, E.C. Russell, W.E. Biles (eds.)

THE TELECOM FRAMEWORK:
A SIMULATION ENVIRONMENT FOR TELECOMMUNICATIONS

Brian W. Unger

Computer Science Department
University of Calgary
2500 University Drive NW.
Calgary, Alberta T2N 1N4
CANADA
email: unger@cpsc.ucalgary.ca

Greg A. Lomow

Software Engineering & Training
Paradigm Shift Inc.
12011 Westwood Hills Dr.
Herndon, Virginia 22071
USA
email: lomow@va.parashift.com

ABSTRACT

The telecommunications industry is undergoing rapid change. Jade Simulations International has developed a set of integrated simulation tools to support the design and analysis of a variety of telecom network planning and operation problems. The simulation environment is designed to aid network planners, operators and marketers meet the challenges associated with rapid change. This paper presents an overview of the Jade environment which supports high fidelity modeling and simulation of public telecommunications networks.

1. INTRODUCTION

The Jade simulation environment, described in detail in [Baezner 93], includes:

1) libraries of high fidelity simulation models of network hardware and software components;
2) an interactive graphical user interface, called the Telecom FrameWork, for defining, managing, and analyzing network simulation experiments using these component models; and
3) high performance simulation executives that make high fidelity network simulations practical.

The model libraries include a Trunk Network Model (TNM), a Signaling System 7 (SS7) model, and models of a network controller. The level of detail supported within these models typically characterizes every signaling packet sent and received within a signaling network, and each connection event associated with call set-up and tear-down within the TNM. The telephone subscriber's characterization includes such information as the time it takes to enter a credit card number for an automated billing call and the re-call probabilities when a destination rings busy.

The TNM routing algorithms are modeled completely representing fixed automatic alternative routing (FAAR) and dynamic automatic alternative routing (DAAR). Central office, inter-exchange carrier and tandem office switch functions are differentiated.

The model libraries use object-oriented techniques and C++ thus allowing new or modified components to be easily added and allowing the model libraries to be used in conjunction with each other. For example, the Trunk Network model can be used to drive the SS7 model.

The models have been used in a variety of areas [Baezner 92] and [Corson 92] including telecom studies:

1) comparing alternative routing algorithms in a network of central offices to validate the claims of a switch manufacturer;
2) studying the survivability of a trunk network under four different failure scenarios;
3) technology assessment involving evaluating alternative scenarios for transitioning a network from switches manufactured by one vendor to those provided by another vendor; and
4) the impact of new service introductions on network congestion and failure.

The Telecom FrameWork provides an integrated environment for using these network component models to build a particular regional telecom network model, and to automate common simulation activities. For example, the graphical interface of the Telecom FrameWork supports data input preparation, setting simulation parameters, executing experiments, and analyzing the results of simulation experiments. Besides the core set of tools, the open architecture of the Telecom FrameWork supports the inclusion of other application-specific tools from third-party vendors.

The major barrier to using high fidelity models to simulate large, complex telecommunication networks has been long execution times and large memory requirements. Jade's *Sim++* object-oriented simulation class library and high performance simulation executives enable very detailed model experiments within reasonable run-times. Either a fast sequential executive or the TimeWarp parallel simulation executive can be used to execute Sim++ simulations, the latter in parallel using networks of Unix workstations.

Existing analytical techniques are not adequate to address the design and analysis problems facing network planners and designers. The emerging network architectures are dynamic, adaptive and consist of heterogeneous hardware and software components. The interaction of usage demands, different component types, new services, congestion and failure cannot be modeled analytically. Simulation models, however, can be made arbitrarily accurate. The downside of highly detailed models are long simulation development times and long execution times. Jade's environment is designed to address both of these disadvantages.

This paper is organized as follows. First, Jade's call-by-call simulation approach is discussed. Second, the model libraries are presented. Next, the Telecom FrameWork that integrates the use of these models is examined. Finally, we describe the underlying object-oriented, parallel simulator that makes high-fidelity, call-by-call simulation possible.

2. CALL-BY-CALL SIMULATION

Jade's environment supports very detailed call-by-call modeling and simulation. Every telephone call, call type, and data transfer is represented. All routing decisions, major network components and relevant software components are characterized.

These call-by-call models can be arbitrarily accurate. For example, each routing decision, each database access, each access control decision and each network management message generated because of a specific telephone call can be simulated. This level of detail enables the analyst to see the load that each call imposes on the trunk and signaling networks and the ways in which they affect one another.

In a call-by-call simulation, network statistics such as call blockage, throughput, utilization, and grade of service are collected at all levels of detail from the system level to the level of an individual message or switch. Furthermore, statistics can be collected on both transient behavior and steady state behavior.

Since each piece of equipment, each database, and each algorithm is modeled explicitly, the user can add, modify, or replace any component in the simulation. This allows alternative network configurations to be easily studied and compared. It enables the characterization of mixed vendor networks. It also means that complex scenarios involving peak loads, sudden changes in loads, and interactions among complex protocols are easy to examine.

This provides a great deal of flexibility in that different components of the network can be modeled at different levels of detail, depending on the focus of the simulation study. For example, detailed models of routing algorithms can be used with simplified, partly analytic, models of error detection and correction algorithms.

This flexibility also makes studying network reliability and failure scenarios possible. All of the following can be represented:

1) the failure of one or more switches;
2) the partial or complete failure of a trunk;
3) the failure of a signaling link;
4) defects within a software module;
5) buffer overflow and congestion;
6) corrupted messages; and
7) new services and the demand for these services.

The duration of failures can be controlled to study how the network responds when failures are corrected and service is restored. Time scales of the network's response can range from fractions of a second to hours.

Accurate, flexible, call-by-call simulation is made possible by two technologies: object-oriented techniques and very efficient execution. Object-oriented techniques make it easier to create, modify and maintain models. Very fast model execution is achieved through carefully designed sequential executives that are tuned for telecom applications and through parallel processing. The latter enables applying scalable computing power to a given simulation problem.

3. THE TRUNK NETWORK MODEL (TNM)

Jade's Trunk Network model (TNM) models circuit-switched, voice traffic on inter-office trunk networks. The model includes objects for simulating central offices, tandem offices, inter-exchange carrier offices, each trunk group (subsequently referred to simply as a trunk), each network processor and a variety of call types.

The TNM is a call-by-call simulation, i.e., each call is characterized. Call models includes setup, the routing decisions made by each office, and call tear down. The TNM includes routing algorithms for fixed, hierarchical, adaptive, and dynamic routing, i.e., FAAR and DAAR. In the case of dynamic routing, the TNM also models the network processors that monitor network traffic and dynamically calculate and update office routing tables.

```
void central_office::body()
  sim_event ev;  /* A Sim++ event */
  int       tg;
  int       action;

  while ( NOT Ended() ) {
    /* Time of next call or next call hangup */
    hold_to = next_hold_to(tg, action);

    /* Advance time */
    Advance_Time(hold_to - sim_clock(),ev);

    if ( ev == SIM_NO_EVENT ) {
      /* The hold was not interrupted */
      if ( action == TERMINATE_CALL )
        terminate_call( tg );
      else
        new_call( tg );
    } else if ( ev.type() == LDR_OVERFLOW ) {
      /* Complete routing call to destination */
    } else if ( ev.type() == LDR_MISS) {
      /* Record call and delete from hangup list */
    } else if (ev.type() == LDR_NP_CALC) {
      /* Calculate usage report and send to NP */
    } else if ( ev.type() == LDR_NEW_TANDEM) {
      /* Record new tandem recommendations */
    }
  } /* End while loop */
```

Figure 1 *Sim++* for a Central Office

Realistic calling patterns are simulated based on either actual data or forecasted data. Full runs of TNM typically simulate millions of calls.

TNM is implemented using *Sim++*, Jade's C++ class library for discrete-event simulation. Using the object-oriented features of C++, each object in the real system (e.g., office or trunk group) is modeled by a separate *Sim++* entity. Figure 1 illustrates *Sim++* source code for the main body of the model of a central office.

Initially, TNM was developed to study the effects of dynamic routing on effective network capacity. In particular, these studies addressed the specific problem of estimating the effects of dynamic routing on the exhaust point as forecasted traffic was increased over a four year period. Subsequently, TNM has been used to study the effectiveness of dynamic routing with regard to network survivability in an NPA that included toll switches and several city networks.

4. THE SIGNALING SYSTEM 7 MODEL

Common channel signaling is at the heart of the new consumer services offered by telephone companies including voice mail, call forwarding, incoming caller identification, and call waiting. Common channel signaling enables many enhancements to 800 services, calling card services, and virtual private network services. Signaling networks based on the Signaling System 7 (SS7) protocol are being deployed across North America and will likely be universal by the mid-1990s. A simple SS7 network is illustrated in Figure 2.

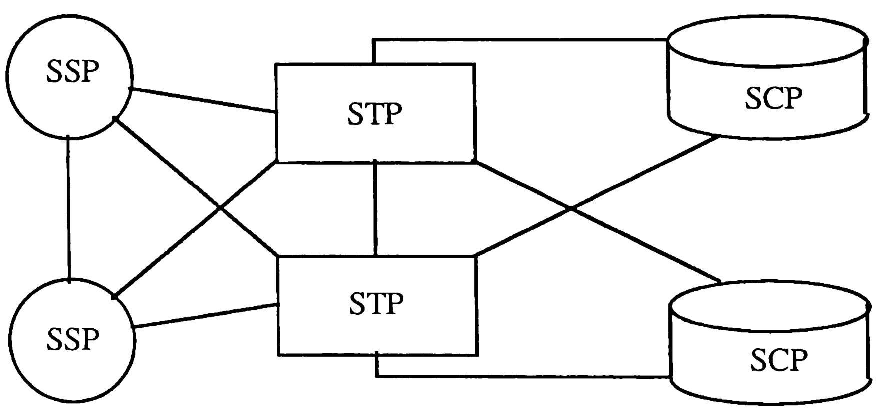

Figure 2. SS7 Network Simulation

As with any new technology, SS7 requires further study and refinement before its full potential is realized. In particular, the impact of new services that place extensive demands on the signaling network is not easy to determine. Jade's SS7 network simulation model [Unger 93 & 92] mimics most of the message handling and network management functions of SS7. An SS7 model offers a testbed for investigating issues such as

1) the introduction of new services;
2) network survivability under various loads;
3) congestion control and network architectures;
4) the network's response to failures; and
5) screening policies at network connection points.

These factors can be studied under normal and overload conditions. Overload conditions are particularly important to understand since failures often occur in just these situations. This has been demonstrated in a number of well publicized network outages.

The components of an SS7 network, including each signaling point (SP) and signaling link (SL), are modeled in sufficient detail to enable the prediction of network behavior under stress. These components include SSPs, STPs, SCPs and SLs.

The message transfer part protocol layers 1-3 are all characterized within the model. The effects of unreliable data transmission and the resulting SP congestion and level 3 notification are characterized. The specific processing of each significant traffic class with the ISDN user part and TCAP are represented.

The SS7 model supports call-by-call simulation in that if used in conjunction with the TNM, each call and the associated signaling messages can be traced through the simulation. The SS7 model library is implemented using *Sim++* and each component of the SS7 network is modeled as a separate *Sim++* entity.

5. A NETWORK CONTROLLER MODEL

Jade's third model library is used for modeling the performance of packet-switched networks carrying control and monitoring messages for a network of intelligent T1 channel banks. These channel banks multiplex voice and data circuits from customer premises onto T1 lines. Voice and data traffic can be intermixed on a single T1 line, and bandwidth may be re-allocated dynamically from a network management workstation.

The re-configuration of T1 links and monitoring of line states is mediated by network management messages carried by the control system. The control system connects the intelligent multiplexers to each other and to the network management workstation. The control network consists of several inter-connected sub-networks, including Ethernet LANs and point-to-point sub-networks made up of T1 channels.

Because of the complex interactions between these networks and their components, it is difficult to analytically estimate the throughput and response times for the control network. Consequently, a simulation model was developed to answer questions such as

1) What is the optimum location of the network management station?
2) What is the required bandwidth of the control network links?
3) Which links should carry the network management traffic?
4) What is the performance of various network topologies and client/server configurations for the network management workstations?

The models for simulating the packet-switched control system includes the following features:

1) The mode of operation of both Ethernet and point-to-point networks is simulated in detail;
2) Every network transaction and message is simulated as a separate event;
3) The simulation model is data-driven; the network topology, Ethernet and client/server configurations, and the control traffic loads are all defined in input data files.

6. THE SIMULATION FRAMEWORK

Jade's Simulation FrameWork is an application-independent collection of general tools and interfaces that automate many common simulation activities. These activities include specifying simulation scenarios, specifying experimental parameters, automatically running multiple replications of simulations, and collecting and archiving simulation results. As shown in Figure 3, the Simulation FrameWork supports:

1) specifying input data (scenario definitions and simulation parameters) and component models,
2) running simulations using either the sequential or distributed simulation executive,
3) collecting and storing results in a database, and
4) the analysis of results.

All of these functions are controlled through the graphical user interface labeled the Simulation FrameWork Interface. The Simulation FrameWork is not intended to be used directly. Rather, Jade tailors it to specific applications by "populating" it with application-specific model libraries, additional results analysis tools, and additional graphical user interfaces.

The Simulation FrameWork and all of its associated tools are written in C++. This allows existing components to be reused and easily adapted for new applications, and simplifies adding new components.

applications, and simplifies adding new components.

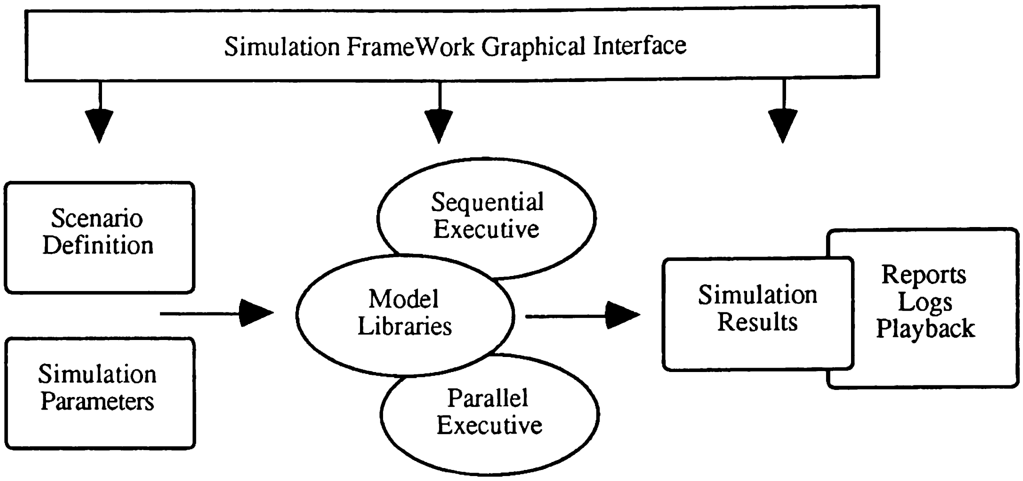

Figure 3. Simulation FrameWork

7. THE TELECOM FRAMEWORK

Jade's Telecom FrameWork is an instance of the Simulation FrameWork that is tailored to supporting the simulation and analysis of telecommunication systems. The Telecom FrameWork differs from the Simulation FrameWork in the following ways.

1) The scenario definition module for the Telecom FrameWork is specialized to allow users to specify network definitions, workloads, and events appropriate for telecommunication systems.
2) The model library module for the Telecom FrameWork includes the Trunk Network Model, SS7 Model, and Packet-Switched Model which replace the generic simulation libraries shown for the Simulation FrameWork.
3) The Telecom FrameWork includes data preparation and data translation utilities for converting actual network data from an operational, on-line database into a form that can be used by the simulation models.
4) The Telecom FrameWork includes additional, application-specific animation and results analysis tools.

Figure 4 shows a screen dump from a session using the SS7 Graphical Analysis Tool. It is one of the results analysis tools available through the Telecom FrameWork. The SS7 Graphical Analysis Tool is a graphical data analysis tool for visualizing and analyzing the results produced by an SS7 simulation. A number of views are provided:

1) Schematic View - Allows the user to see the logical connectivity of the nodes and links which comprise the network. Multiple network components can be aggregated and displayed using one icon.
2) Matrix View - Allows the user to see every node and connecting link between every pair of nodes in either the full network or a subset of the network. Each square in the matrix view shows the status of a link. Colors are used to highlight areas of interest and extreme situations.
3) Playback Control Window - Allows the user to control how fast the playback advances, pause it, jump directly to a specific simulation time, or even reverse time to review some interesting interval.
4) Custom Matrix View - Allows the user to filter the data collected by the simulation and customize the presentation of that data.
5) Data Selection View - Allows the user to select which data from the playback file is represented by which display icons.

8. *SIM* ++ SIMULATION ENVIRONMENT

Jade's *Sim++* is an object-oriented, discrete-event simulation library written in C++. *Sim++* is a general-purpose package suitable for any form of discrete-event simulation including telecom, transportation, military, written in *Sim++*. Detailed descriptions of the environment, related patents and technology can be found in [Baezner 93], [Cleary 91], [Jefferson 91 & 85], [Lomow 88], [Unger 90] and [West 88].

Sim++ simulations are defined in terms of entities and events. Entities are independently executing objects with their own address space. Entities communicate and synchronize their actions by scheduling and receiving events. Entities receive events in timestamp order. *Sim++* also provides simulation objects for random number generation, data collection and reporting, linked-list processing, and automatic event tracing.

Sim++ is unique because it is designed for writing simulations that can either run sequentially on a single workstation using a high performance centralized event list or run in parallel on multiple workstations using a special synchronization executive called TimeWarp (see next section). *Sim++* simulations can be moved between these two run-time environments without source code modifications and without recompiling.

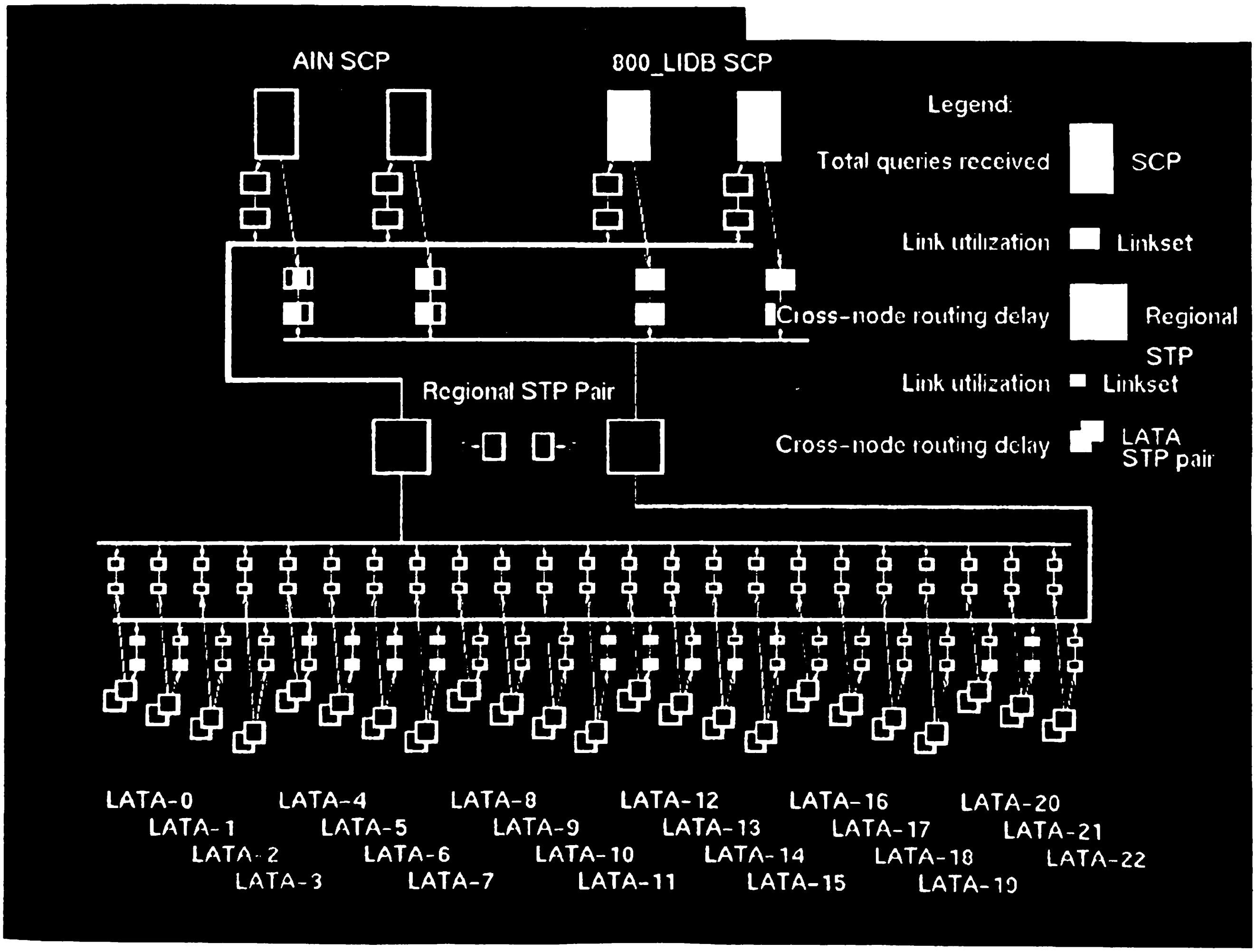

Figure 4. SS7 Graphical Analysis Tool

Figure 5 shows the architecture of the *Sim++* Development Environment. The executive being used does not matter to *Sim++* and is transparent to the user. Both executives support the following modules:

1) **Run Time Configuration** - provides several options for mapping simulations to processors

from completely automatic mapping to user specified mapping.

2) **Performance Analysis Tools** -provides tools for analyzing the execution of simulations so their performance can be tuned and optimized.
3) **Distributed I/O** - provides I/O routines that support distributed and parallel I/O in a manner that is transparent to the user.

Sim++ programs using TimeWarp are transparently scalable and deterministic. Scalability and determinism are both important for call-by-call simulation of telecommunication networks because they allow more processing power to be applied as the size and fidelity of the simulation grows.

Scalability means that the simulation can be executed on varying number of processors. The number of processors is supplied at run time and *Sim*++ maps the simulation objects on to the available processors.

Determinism means that identical runs yield the same results no matter how many processors on which the simulation is executed. Determinism is crucial for verifying the simulation is correct and debugging it.

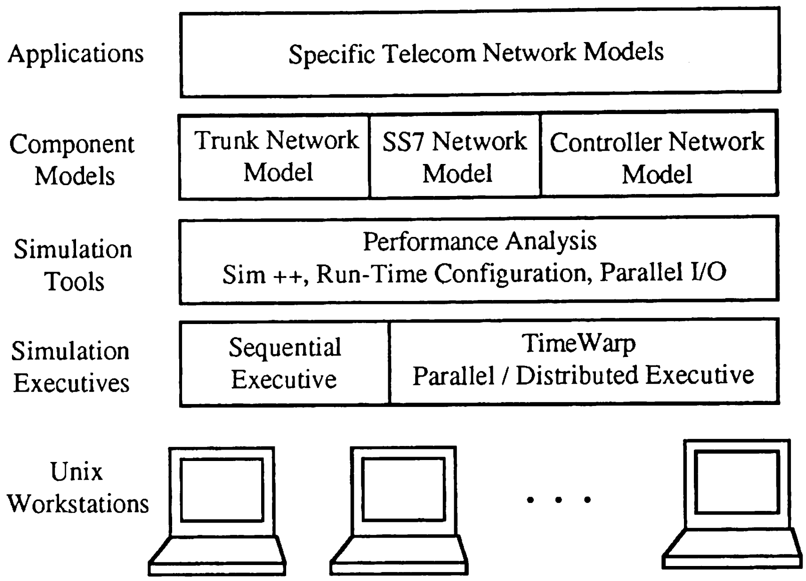

Figure 5. ***Sim*++ Development Environment**

9. JADE'S TIMEWARP

Jade's TimeWarp is a synchronization executive that coordinates a *Sim*++ simulation running on multiple processors. TimeWarp is responsible for delivering events to entities in timestamped order and regulating the execution of the entities running on different processors so that their execution is consistent with an equivalent sequential simulation. At the same time, TimeWarp executes entities running on different processors in parallel when their actions do not affect each other.

The advantages of parallel execution on multiple processors using TimeWarp include:

1) running simulations faster;
2) running larger simulations that have more objects and require more memory;
3) running higher fidelity simulations that have more detailed component characterizations.

Figure 6 shows a speedup curve for a TNM simulation written in *Sim*++ and executed using TimeWarp. The simulation is modeling a network of 40 central offices in a circuit-switched voice network. It is simulating the busiest hour of the busiest day and processes over 2 million phone calls. The graph shows that the execution time is reduced from almost 3 hours when the simulation is run sequentially to about 14 minutes when running on 36 processors. This is a 13.8 times speedup. Presently, *Sim*++ and TimeWarp run on Sun Sparc workstations, IBM RS/6000 workstations, and HP Series 700 workstations. Further details regarding TimeWarp and developing Time Warp applications can be found in [Baezner 03], [Jefferson 85], [Lomow 93] and [Unger 90].

10. CONCLUSIONS

High fidelity, call-by-call simulation is needed to support the design and analysis of complex, dynamic, and heterogeneous telecommunication networks.

The Telecom FrameWork provides an integrated set of tools for defining, managing, and experimenting with models of a telecom network and for analyzing the results of those experiments.

Object-oriented simulation using *Sim++* provides the foundation for building and linking models of the separate subsystems that comprise telecommunication networks. Distributed and parallel simulation using *Sim++* and TimeWarp makes high fidelity, call-by-call simulation practical by providing the needed computational power.

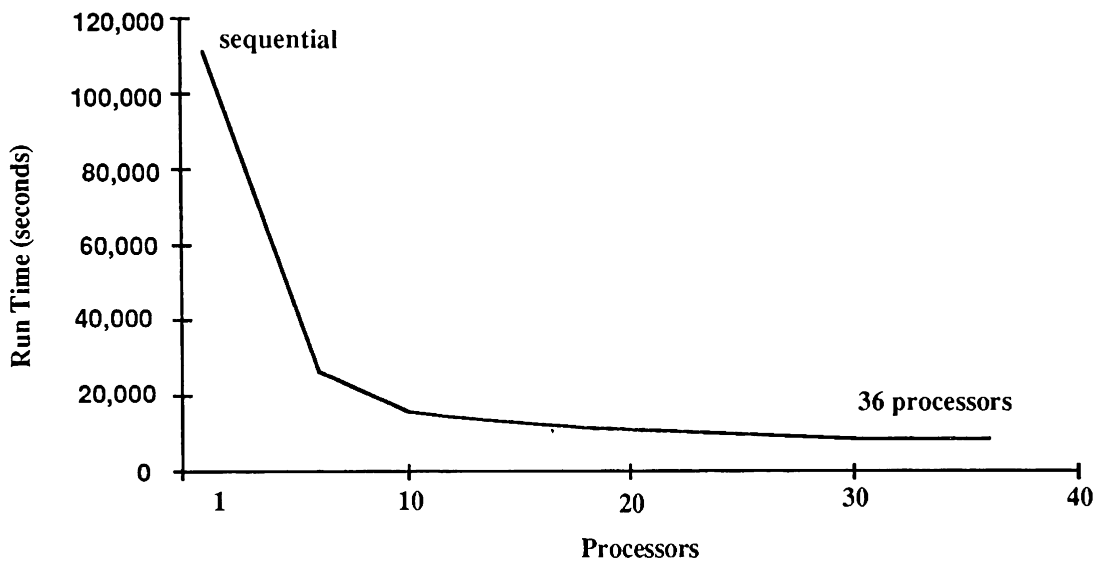

Figure 6. Speedup Graph for Trunk Network Simulation

ACKNOWLEDGMENTS

The material presented in this paper owes its existence to the technical staff at Jade. In particular, Jeff Allan, Dirk Baezner, Oyvind Bjartveit, John Cleary, Alan Covington, Mike Bonham, Steve Maryka, Murray Peterson, Chuck Rohs and Darrin West have all made major contributions to the Jade Simulation environment. All of the errors and omissions are the responsibility of the authors.

REFERENCES

Baezner, D., Lomow, G. and Unger, B. W. (1993) "A Parallel Simulation Environment based on TimeWarp", *International Journal in Computer Simulation*, in press.

Baezner, D., Rohs, C. and Jones, H. (1992) "U.S.Army MODSIM on Jade's TimeWarp", Proc. of the 1992 Winter Simulation Conference, 665-671, December.

Cleary, J., Lomow, G. and West, D. (1991) Patent Application filed by Jade Simulations for Activate, San Francisco, July.

Corson, S. (1992) "A Distributed, Object-Oriented Communication Network Simulation Testbed" Proceedings of the 1992 Winter Simulation Conference, Washington DC., 672-679, December.

Jefferson, D. and West, D. (1991) Patent Application filed by Jade Simulations for Cancelback, San Francisco, January.

Jefferson, D.R. (1985) "Virtual Time", *ACM Transactions on Programming Languages and Systems*, 7(3), 404-425, July.

Lomow, G. and Baezner, D. (1993) "Methods for the Design of Efficient Parallel Simulations", Proceedings of the 1993 Western Simulation Multiconference, 219-223, San Diego, January.

Lomow, G. (1988) "The Process View of Distributed Simulation," PhD Thesis, Computer Science Department, University of Calgary.

Unger, B. W., Goetz, D. and Maryka, S. (1993) "The Simulation of Common Channel Signaling", internal report, University of Calgary.

Unger, B. W. (1992) "The Parallel Simulation of SS7 Common Channel Signaling", The 4th IEEE International Workshop on Computer Aided Modeling, Analysis and Design of Comm. Networks (CAMAD), invited presentation, Montebello, September.

Unger, B.W., Cleary, J., Dewar, A. and Xiao, Z. (1990) "A Multi-Lingual Optimistic Distributed Simulator", *Transactions of the Society for Computer Simulation*, 7 (2) 121-152, June.

West, D. (1988) "Optimizing Time Warp : Lazy Rollback and Lazy Reevaluation" MSc Thesis, Computer Science Department, University of Calgary.

AUTHOR BIOGRAPHIES

Brian W. Unger is a Professor of Computer Science at the University of Calgary and was the founding president of Jade Simulations International Corporation. His research interests include massively parallel discrete event simulation and the parallel simulation of telecommunications networks based on optimistic synchronization mechanisms. Dr. Unger has published over 60 papers, edited four conference proceedings and was the principal investigator of a $1.2 million research project on distributed software development environments. Dr. Unger received his Ph.D. in Computer Science from the University of California at San Diego in 1972.

Greg A. Lomow is a consultant in object-oriented technology with Paradigm Shift Inc. He has eight years of experience in the design and development of simulation applications, simulation packages, distributed systems and object-oriented software design. Dr. Lomow was one of the founders of Jade Simulations where he participated in the development of an object-oriented environment for building distributed systems and parallel simulations. He has co-authored a book on simulation in Ada which has served as the basis for professional development seminars. He is a member of SCS, IEEE and ACM. Dr. Lomow received his Ph.D. in Computer Science from the University of Calgary in 1988.