Proceedings of the 1990 Winter Simulation Conference
Osman Balci, Randall P. Sadowski, Richard E. Nance (eds.)

SIMULATION OF THE LAN BEHAVIOR IN THE DISTRIBUTED HP-UX ENVIRONMENT

Joseph J. Devai

Manufacturing Test Division
Hewlett-Packard Company
Loveland, Colorado 80537

Thomas C. Kerrigan
Michael K. Molloy

Workstation Systems Division
Hewlett-Packard Company
Fort Collins, Colorado 80525

ABSTRACT

The capability of the local area networks to handle the large amount of data transfer among distributed workstations is frequently questioned. The LAN study reported in this paper investigated both the Ethernet LAN and the Distributed HP-UX data communication protocol implemented on it. A "triad" approach of experimentation, analytical and simulation modeling was followed. The models were vigorously validated against each other and the measurements.

The performance study of the DUX/LAN system concentrated on investigating the causes of the message transit time and its variability. According to this study, the message communication time is dominated by transmission delays and not LAN contention and the LAN performs well in the 30-50% channel utilization range.

1. INTRODUCTION

A significant segment of the workstation marketplace is engaged in team computing, i.e., the sharing of computer resources and data among members of a team working on a common project. In such a distributed environment, major resources, such as disks and high performance CPUs, are usually assigned to specialized workstations functioning as servers. Knowledge workers using client workstations have access to the servers via a local area network (LAN). Given the large amount of data transfer, a widespread concern is that the LAN becomes the bottleneck in distributed systems.

Current technologies to model multiuser, time sharing environments can not be easily extended to model distributed systems connected by LANs. For example, as a service center in a distributed system, the LAN exhibits a combination of the First-Come-First-Served (FCFS) and Processor Sharing queueing disciplines which does not fit within the assumptions of product-form Markovian queueing solutions. Furthermore, workstation customers tend to use large interactive software packages, like CAD/CAM or CASE tools. For them, the most important performance metric is user response time and consistency of the response time.

To understand possible system bottlenecks and lay the foundation for developing capacity planning tools, a small team in the Workstation Systems Division of Hewlett-Packard (HP) followed a "triad" approach of using controlled laboratory experimentation, simulation, and analytical modeling to study performance.

Experimentation generally provides the most realistic information regarding performance issues, although it is an expensive technique and is often impractical. Since it only measures "what is" and not "what can be," its ability to perform capacity planning functions is limited.

Analytical models are quick, easy to use, and answer capacity planning questions readily. However, most analytical models require simplifying assumptions about the system dynamics which may limit the type and accuracy of performance metrics that can be predicted.

Through logical recreation of a system's operation, simulation may provide more accurate predictions than an analytical model. Both the cost and execution time of a simulation model usually lie between that of the analytical model and an experiment.

The new contribution of the work reported here consists of

- modeling the LAN and the data communications protocol simultaneously
- modeling different request and response message size distributions
- predicting both message and frame transit times
- predicting mean and variance of message transit times.

This paper concentrates on the simulation leg of the triad and on those results that could not have been predicted without simulation.

2. THE DUX/LAN SYSTEM

HP's distributed Unix operating system (DUX) contains a specialized, request-reply protocol which resides entirely within the kernel. At boot time each diskless client discovers the Ethernet address of its server, and the disk server accepts that client. These addresses are the only ones the stations use. All communication using the DUX protocol is hence between a particular client and its server. With few exceptions, such as **I_am_alive** messages, the structure of all communication follows the same paradigm. A station sends a request message to another station, which in turn responds with a reply. The requesting station then acknowledges the receipt of the response with an acknowledgment message. Eight Kbyte file system buffers and 4 Kbyte memory pages constitute typical DUX messages.

DUX relies exclusively on the error detection mechanism of the LAN. If a receiving station loses a message or receives a message with errors, the DUX software retransmits the entire message after a timeout period.

The Ethernet/IEEE 802.3 LAN is an integrated hardware and software system which can connect up to 1,020 workstations within a limited geographic area. A single channel, network interface hardware, and the channel access protocol comprise its major components. Data is transmitted in packages called frames of 72-1526 bytes at a transmission rate of 10 megabits per second.

The detailed description of the Ethernet/IEEE 802.3 LAN and the Carrier Sense, Multiple Access with Collision Detection (CSMA/CD) channel access protocol [IEEE 1985] is beyond the scope of this paper.

To prepare a message for transmission on the LAN, the DUX protocol layer decomposes messages longer than 1454 bytes into several data blocks. It precedes every data block by the IEEE 802.3 and IEEE 802.2 headers (32 bytes, including the preamble) and the DUX protocol header (24 bytes). In addition, if the data block is the first block of the message, the protocol layer appends the DUX message header (12 bytes) after the DUX protocol header and before the data. Concatenation of data with these headers and the CRC constitutes a frame. All frames, with the possible exception of the last one, have a length of 1526 bytes, the maximum allowed on the LAN by the Ethernet protocol.

The protocol layer next passes the frames, one at a time, to the LAN device driver. The driver then copies a frame into a transmission buffer on the LAN card. Upon completion of this transfer, the driver sends a permission to transmit signal to the

LAN card. The frame is transmitted across the LAN according to the CSMA/CD channel access protocol. When the frame is successfully transmitted, the LAN card sends an interrupt to the driver. After processing this interrupt, transport routines send the next frame to the device driver, which in turn transfers it to the LAN card.

The receiving LAN card also sends an interrupt to its DUX transport layer at the end of a successful transmission, and subsequently transfers the frame upward. Once all transmitted frames arrive, the DUX protocol layer of the receiving station reassembles the frames into the original message.

3. THE SIMULATION MODEL

3.1 Model Overview

Simulation is an experimental technique which recreates the relevant processes of a system under investigation in abstract form, through the use of computer programs. Observing and aggregating results of numerous reenactments of the processes generate the output data. This technique often incorporates statistical models and does not attempt to optimize results.

The function of the DUX/LAN model [Devai 1989] is to accurately predict data communications performance for a group of workstations in a distributed workgroup environment. It consists of two independent modules (Figure 1).

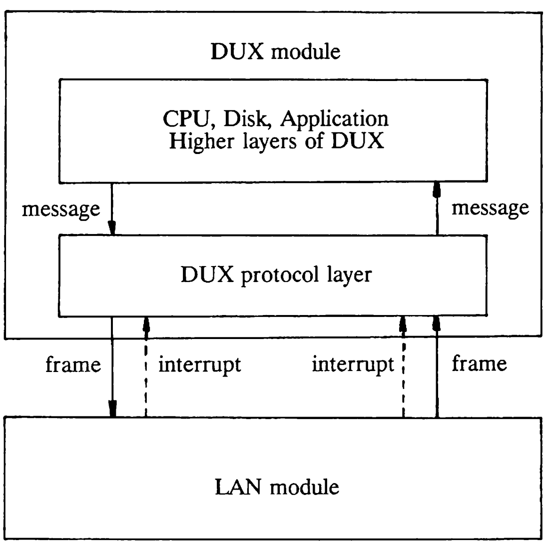

Figure 1. Structure of the Simulation Model

3.2 DUX Module

The DUX module is a rudimentary model which simulates the usage of a CAD software package on the Distributed DUX platform. Part of the module includes models of both the CAD application and the top layers of DUX. This part of the module generates the message arrival pattern characteristic for an interactive application running on DUX. In addition, the module models in detail the DUX protocol layer, which breaks up messages and transfers frames to the LAN, because the protocol layer directly influences message transit times. The DUX module assumes that requests for CPU-processing by the DUX protocol layer have immediate access to the CPU.

The elements of the DUX module are the CLIENTs, SERVERs, TRANSACTIONs, and MESSAGEs.

CLIENTs represent the workstations on the network under investigation. Every CLIENT is permanently assigned to one and only one SERVER. A CLIENT generates request messages that it sends to the SERVER, and it receives subsequent response messages from the SERVER. Upon receiving a response message, a CLIENT returns an acknowledgement message to the server, following a constant time delay. The message processing time between receiving a response message and sending the next request message is a random sample from the negative exponential distribution.

SERVERs represent the file servers of a workgroup. The DUX module models a SERVER as a single service center queueing system with a FCFS queueing discipline. Service times are random and follow the normal distribution.

A single user command initiates TRANSACTIONs, a set of interactions between clients and servers. Think time between TRANSACTIONs is a random time delay which is drawn from the negative exponential distribution. A TRANSACTION spawns MESSAGEs.

MESSAGEs represent the data package that a workstation transmits between a CLIENT and SERVER over the network. Random sampling of the empirical transaction size distribution determines the number of messages in a TRANSACTION. Similarly, a random sampling of the empirical request message length distribution yields the message length. Upon transmission of a client request MESSAGE to the SERVER, a random sampling of the empirical response message length distribution determines the response message length. Experimental results for the application under investigation specify all these empirical distributions.

For each CLIENT, the mean think time, the mean message processing time, and the request message length distribution are necessary inputs. Both the interrupt processing time and the acknowledgement message turnaround time must also be input. However, these are the same for all CLIENTs. The mean and standard deviation of the service time distribution as well as the server interrupt processing time define the workgroup SERVER.

The principal DUX module outputs are the mean and standard deviation of the message transit time distribution. Message transit time is the time interval between the arrival of the first frame of the message at the LAN interface card of the sending station and the reassembly of the message at its destination.

3.3 LAN Module

The LAN module is a general model of multiple Ethernet/IEEE 802.3 LANs connected by bridges. It can include both single cast and broadcast background traffic. The LAN module simulates detailed movement of every frame through the network. This module assumes that every hardware device on the network is always operational and that the channel provides a perfect electrical connection.

The elements of the LAN module are the CHANNELs, REGULAR and PHANTOM STATIONs, BRIDGEs, CONNECTIONs, and FRAMEs.

CHANNEL represents the cable segments, repeaters, point-to-point links, and medium attachment units in a subnetwork. The LAN module models the signal propagation across the CHANNEL by increasing frame transmission time by a constant amount.

REGULAR STATIONs represent the interface card in a workstation or server that belongs to the system being investigated. They enforce the 9.6 microsecond interframe gap and schedule the retransmission of frames after a collision. PHANTOM STATIONs represent the interface card of an imaginary station that transmits a random stream of FRAMEs which simulates the background traffic of other computers.

The model element BRIDGE represents a single purpose hardware device that connects two Ethernet/IEEE 802.3 subnetworks of not necessarily identical transmission rates by selectively transmitting information between them. CONNECTION is a model element without a direct analog in the physical system. It represents the unidirectional connection from one subnetwork to another, including all required processing and conversion. A FRAME is an indivisible data package that is transmitted on the LAN.

Transmission rate and channel propagation delay specify the LAN's essential characteristics. To analyze a complex network the LAN module requires this data input individually for every subnetwork. In addition, the module also needs both the deterministic routing between subnetworks and the bridge delay times. The mean frame interarrival time and the frame length and frame destination distributions must be input for the PHANTOM STATIONs on every subnetwork.

Major output data of the LAN module include the channel utilization, mean frame transit time, client and server collision probabilities, channel collision probabilities, and the distribution of the number of collisions experienced by a frame. The module collects this data individually for every subnetwork.

3.4 Implementation

The simulation program implements both the DUX and LAN modules in the PC version of the GPSS/H simulation language. The program consists of approximately 300 block and 50 control statements, which together with the input data nearly exhaust the 640 Kbyte memory available under MS/DOS. The DUX and LAN modules are independent to the extent permitted by the syntax of the language.

4. THE EXPERIMENTAL METHODOLOGY

The experimental leg of the triad uses technology developed to automate the measurement of real applications on various sized diskless workstation clusters [Jones and Lindheimer 1989]. This technique allows simple empirical performance characterization of specially instrumented applications in the Performance Measurement Center of the Fort Collins Workstation Lab. "Hooks" in the application code demarcate user transaction boundaries and compute transaction response times. These hooks log performance information regarding CPU, disk, and LAN activity to the file server disk. Hooked applications run on each node in the cluster. The server also records performance data, using a different set of tools. The LAN activity is also measured directly by an HP4972 LAN analyzer. This hardware device collects frame size and interarrival time distributions and average collision rates.

Two applications, Printed Circuit Design System (PCDS), an HP software package, and Unigraphics, a mechanical engineering drafting, design, and manufacturing package of the McDonell-Douglas Company were characterized through measuring actual design sessions [Molloy 1989; Morgenstern n.d.].

The experiments automatically replay a captured design session in various cluster configurations. Results of the experiments provide input to both the analytical and simulation models as well as data points for validating the models.

5. THE ANALYTICAL MODEL

The analytical leg of the triad [Kerrigan 1989] focuses on developing a model to estimate the means and variances of the times for request and response messages to traverse the LAN. The resulting model mathematically analyzes a diskless cluster as a network of service nodes using the product-form theory for closed networks of Markovian queues. The original contributions of the model are the use of a system level approach to LAN performance analysis, the ability to account for varied traffic types, and the inclusion of a background traffic class.

The model employs two major approximations. First, it simplifies the queueing network by tagging one of the N clients and reducing the other N-1 clients to a homogeneous complementary class. Mean think times, processing times, and request and response message lengths characterize each client in the complementary class. These means are the averages over all clients in that class. The result is three distinct traffic classes: the tagged client class, the complementary client class, and the background traffic class.

Second, since the LAN service discipline does not strictly conform to the hypotheses required to obtain a product form for the probability distribution of system states, the model uses an adjustment which discounts the message population at the LAN by the response messages queued for LAN service at the diskless server.

6. TRIAD VALIDATION
6.1 Validation Procedure

Validation is the process of corroborating results of experiments and predictions of models for a system under investigation. The objective is to determine if the models' assumptions are appropriate and to demarcate the boundaries of the models' applicability. This section discusses the validation of each leg of the triad against the other.

17 experiments of running PCDS on a single-server diskless cluster with an HP Series 350 server and HP Series 319 clients were used for the validation (Table 1).

Table 1. Validation Experimental Configurations

Number of Clients in the Investigated Clusters			
Think Time in seconds	Background Load Level in % of LAN utilization		
	0%	7%	31%
8	1, 4, 8, 12, 16	1, 8, 16	1, 8, 16
3	1, 8, 16	-	-
0	1, 8, 16	-	-

6.2 Selected Results

Validation of DUX/LAN Model for PROPHET [Devai et al. 1989a] lists and discusses the results of all validation experiments. The graphs and assessments below describe selected results of the eight second mean think time and zero percent background load case.

Figure 2 illustrates LAN utilization as a function of the number of clients in the diskless cluster. The measured utilization is higher than the utilization predicted by both the analytical and simulation models. This is due to the absence of some LAN activity in the workload as modeled, such as **I_am_alive** packet transmissions and **path_look_up** requests.

The graph of server collision probabilities versus number of clients (Figure 3) shows an almost perfect agreement between the analytic and simulation models and measurement. Reflecting the ignored LAN activity, as discussed above, the analytical and simulation models predict a slightly lower server collision probability.

The channel collision probability versus number of clients curves (Figure 4) show radical differences between the measured and predicted values. Collision probabilities as measured by the LAN analyzer are significantly higher than the values the models predict. These probabilities are also higher than those that collective measurements by the client and server LAN cards indicate. The cause of this discrepancy is most likely the inconsistent collision counts reported by both the HP and DEC MAU's which was discovered during the validation process.

Only simulation and analytical model data are available for comparing server message transit times. Figure 5 charts the mean, Figure 6, the standard deviation of the server message transit time as a function of the number of clients. The predictions of the analytical and simulation model agree very well for both measures.

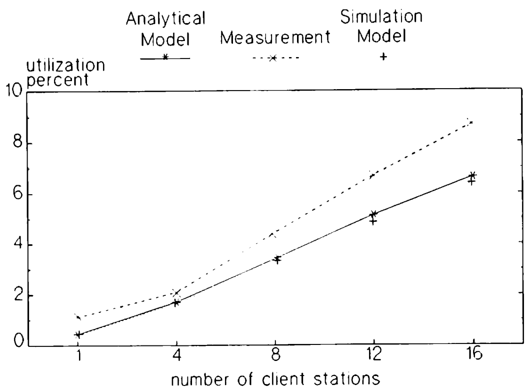

Figure 2. LAN Utilization at 8 Second Think Time, No Background Traffic

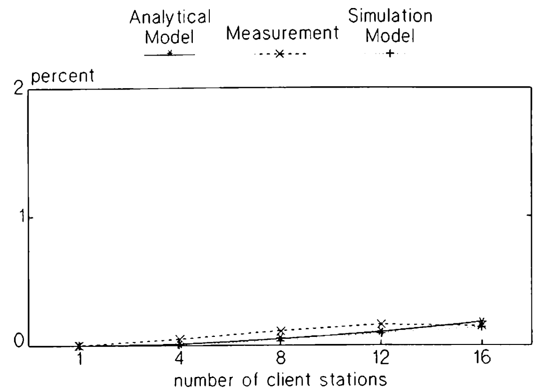

Figure 3. Server Collision Probability at 8 Second Think Time, No Background Traffic

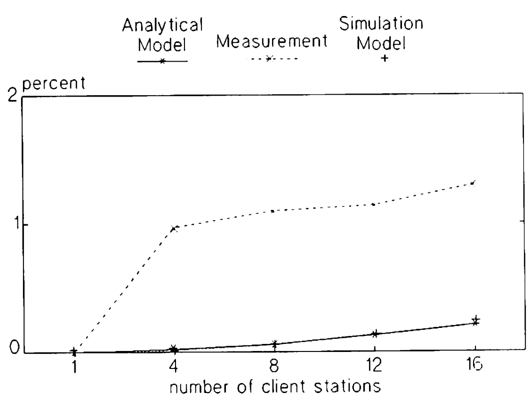

Figure 4. Channel Collision Probability at 8 Second Think Time, No Background Traffic

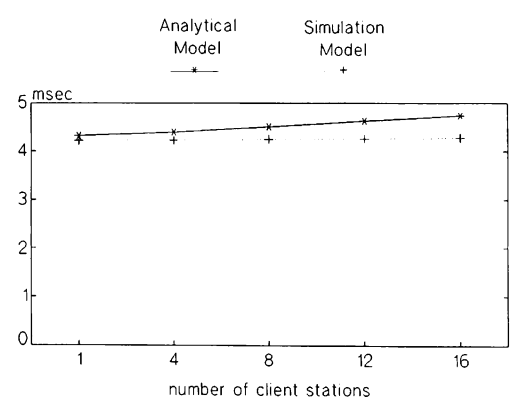

Figure 5. Mean of Server Message Transit Times at 8 Second Think Time, No Background Traffic

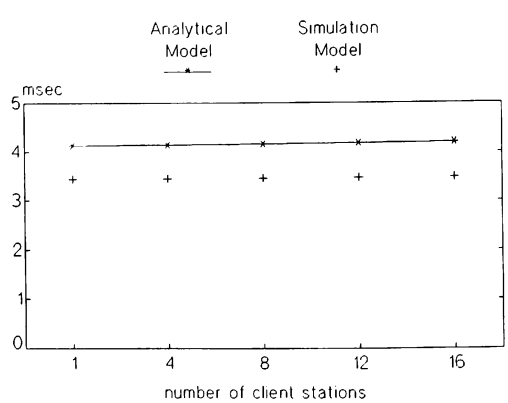

Figure 6. Standard Deviation of Server Message Transit Times at 8 Second Think Time, No Background Traffic

6.3 Validation Conclusion

Based on the validation that is summarized above, the analytical and simulation models are valid models of the DUX/LAN performance within the following limits:

The accuracy of the analytical model predictions declines significantly when the background traffic exceeds 30-35%.

Care should be exercised in analyzing the results of the simulation model when the utilization of the server or servers exceeds about 70%.

6.4 Comparison with Other Models

There have been many papers published in the literature about LAN behavior as far back as 1976. Most of the papers describe analytical or simulation models of the Ethernet local area network without considering the data communications protocol implemented on it. Only few papers describe experimental results. Since they all investigate the transmission of independent frames on the LAN, their applicability to the client-server environment is limited.

Unlike the Prophet LAN models, previously published models make the assumption that there is an infinite number of stations on the LAN. The most important implication of this assumption is that no station will ever have a queue of frames because the arrival rate at any individual station is infinitesimally small. As a matter of fact, stations of the LAN are not modeled explicitly. In models such as Lam's [1980], the arrivals are considered to be a merged Poisson process. The models account for the propagation delay indirectly.

To evaluate the impact of the different assumptions in our transaction based request-reply models versus the simpler infinite population models, the mean delay experienced by individual frames at the Ethernet link level predicted by three different models are presented in Figure 7.

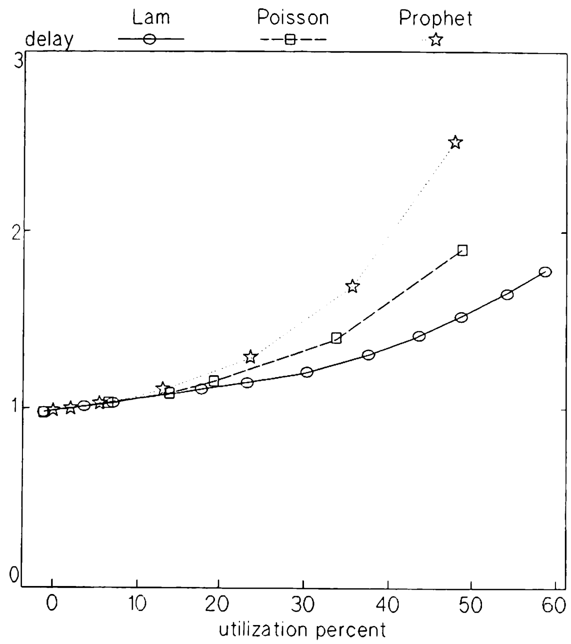

Figure 7. Mean Frame Delay vs. Channel Utilization of Various Models

The first model is the LAN simulation model of Prophet, described in the previous sections.

The second model is a simulation model where the frame arrivals are modeled by an infinite population Poisson process. Arrivals are assumed to occur with a uniform distribution over the channel so the effects of propagation delay are explicitly modeled. The delay shown in Figure 7 for this model does not include any queuing at the station because of the infinite population assumption.

The third model is Lam's analytical model for CSMA/CD networks. The model is relatively simple and easy to use, uses an infinite population model, and assumes that the packet lengths are exponentially distributed.

All three models predict exponentially rising mean frame delay for increasing utilization. Up to about 15% LAN utilization, all models predict approximately the same values. At utilizations above 15%, the Prophet model predicts the largest mean delay, followed by the other simulation model. Lam's analytical model predicts the lowest mean frame delay.

In summary, the frame delay for infinite population models is significantly less than the actual delay seen by a limited number of workstations on the network. The original contribution of both the analytical and simulation Prophet models of Ethernet LANs lies in their capability to predict the frame delays correctly for a group of workstations.

7. BEHAVIOR OF MESSAGE TRANSIT TIMES UNDER DISTRIBUTED HP-UX

Using the validated models several experiments were conducted to investigate the behavior of the DUX/LAN system using the PCDS and Unigraphics workloads (Table 2).

Table 2. Comparison of PCDS and Unigraphics Workloads

Parameter	PCDS	Unigraphics
Mean Think Time	8 seconds	2 sec
Server Message Proc. Time	30 msec	28 msec
Client Message Proc. Time	160 msec	120 msec
Mean Request Message Length	2259 bytes	2120 bytes
Mean Resp. Message Length	2547 bytes	2788 bytes
Messages per Transaction	11	111

To compensate for the differences in the mean think times and expected number of messages per transaction, all results, except utilizations, were normalized to channel utilization [Devai et al. 1989b]. Following is a discussion of the mean and variability of the message transit times and their components.

7.1 Mean Message Transit Time vs. Utilization

Figure 8 shows mean server message transit times plotted against channel utilization. The small differences in message sizes between the two workloads account for the small differences in mean transit time at low channel utilization. As utilization increases, the contributions of initial CPU queueing and LAN contention increase. Mean transit times for the PCDS workload increase faster than for the Unigraphics workload because there are more PCDS clients for a given channel utilization. The same is true for request and acknowledgment messages.

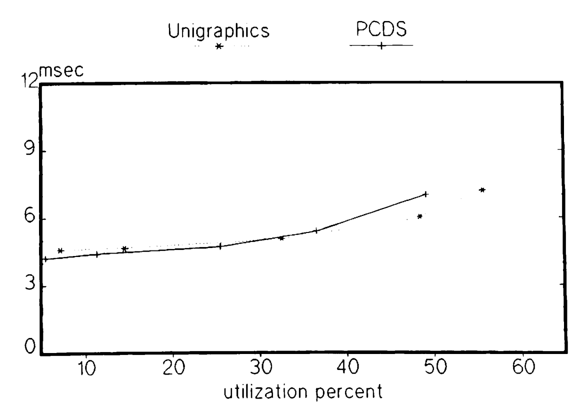

Figure 8. Mean of Server Message Transit Times

Figures 9 and 10 show components of mean transit time for Unigraphics response messages for two values of utilization. Initial CPU queueing delay and LAN contention make significant contributions only at channel utilizations above the moderately high level.

The conclusion is that normalized to channel utilization, significantly different workloads give essentially the same mean transit times. Moreover, mean transit times increase only about 70% at moderately high channel utilizations. Such an increase does not translate into a significant increase in user response time.

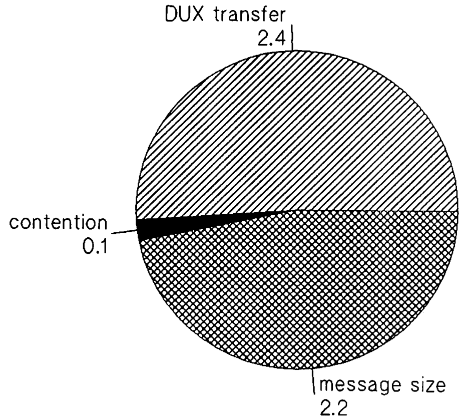

Figure 9. Contributions to Mean Message Transit Times at 16% Utilization (in msec)

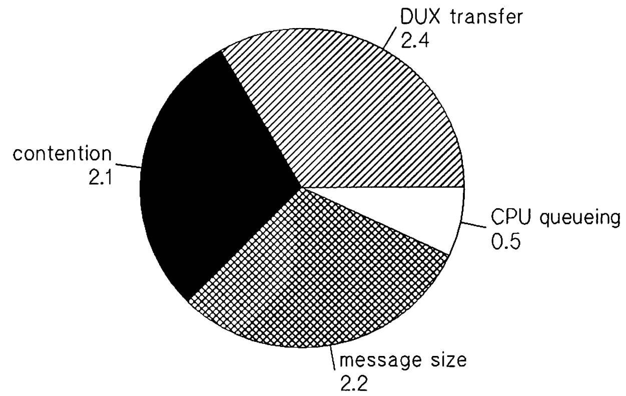

Figure 10. Contributions to Mean Message Transit Times at 57% Utilization (in msec)

7.2 Variability of Message Transit Times

During transmission, messages compete for various resources, such as the CPU and the LAN channel. Consequently, the message transit time will vary for every message. This variability of message transit times is important for two reasons. First, it can influence transaction response times. User productivity declines when response time exhibits erratic behavior. Second, high variability is the first sign of unstable behavior of the DUX/LAN system. If utilization is pushed higher, this unstable behavior will also result in rapidly increasing mean transit times.

Figure 11 shows the standard deviation of the server response message transit times as a function of the channel utilization of the LAN.

The standard deviation has a slight linear increase up to about 30-35% channel utilization and an exponential rise thereafter. The values are very close to the mean response message transit time up to 30-35% LAN utilization, but become greater at higher channel utilization. Both the PCDS and Unigraphics workloads have essentially the same behavior. The standard deviation rises somewhat faster for PCDS.

The standard deviation of the client request transit time vs. channel utilization curves, Figure 12, exhibit the same qualitative behavior as the response message curves. The values of the standard deviation are 10-20% higher than for the response messages.

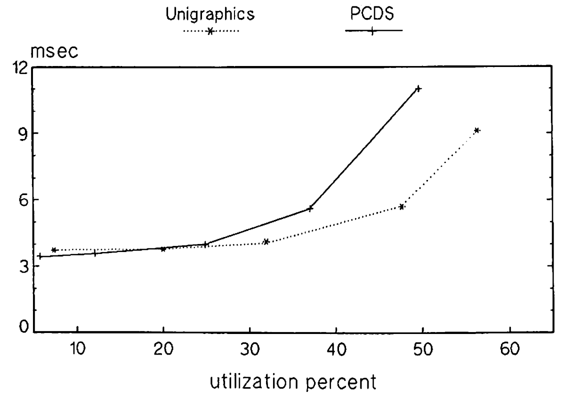

Figure 11. Standard Deviation of Server Message Transit Times

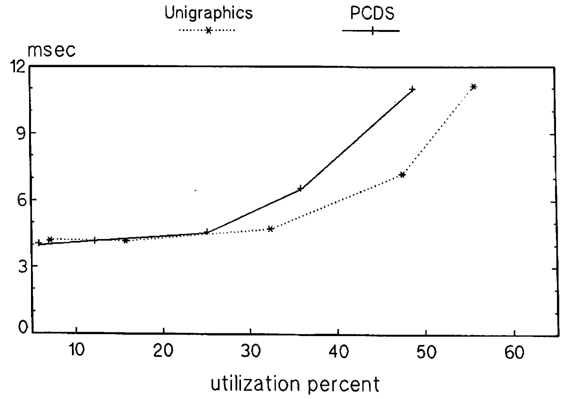

Figure 12. Standard Deviation of Client Message Transit Times

The standard deviation of the client acknowledgement message transit times, not depicted here, increases linearly up to about 30% channel utilization and exponentially thereafter. The values are 10-20% higher than the mean acknowledgement message transit time. Both PCDS and Unigraphics workloads exhibit similar behavior. The standard deviation rises somewhat steeper for PCDS.

The conclusion from these graphs is that the standard deviation of the message transit times is high in comparison to the mean and is rapidly rising once the channel utilization exceeds 30%.

To investigate the causes of variability, the message transit time was decomposed into six mutually exclusive components:

t1 Initial CPU queueing – Delay experienced by the first frame of a message as it waits for service at the DUX level of the sending workstation.

t2 Transfer to the LAN card – Sum of downloading time to the LAN card for the first frame of a message and of the DUX interrupt processing times and downloading times for all subsequent frames at the sending workstation.

t3 Transfer from the LAN card – Sum of DUX interrupt processing times and frame uploading times from the LAN card at the receiving workstation.

t4 Initial deferral on the LAN – Time for which the frames defer to another transmission on the LAN before their first transmission attempt.

t5 Collision resolution – Sum of jam times, backoff times, and deferral times experienced by the frames of a message due to collisions.

t6 Net transmission on LAN – LAN transmission times of the frames of a message.

Using a specially instrumented version of the validated simulation model, the duration of each component of the transit time was measured for every message. At the end of the simulation experiment, the mean, variance, and all covariances were calculated for every component. The analysis below discusses the results of the Unigraphics experiments with 8 and 28 clients (Table 3).

Table 3. Sources of Variability of the Message Transit Times, Unigraphics Workload (All elements in msec*msec.)

Message Type	Request		Response		Acknowledgement	
No. of Clients	8	28	8	28	8	28
LAN utilization in %	16.04	56.95	16.04	56.95	16.04	56.95
Initial CPU queueing [V(*t1*)]	0.00	11.76	0.00	24.66	0.36	1.43
DUX data transfer [V(*t2*)+V(*t3*)+2cov(*t2*,*t3*)]	3.86	3.86	3.08	3.08	0.04	0.04
LAN contention [V(*t4*)+V(*t5*)+2cov(*t4*,*t5*)]	0.24	84.62	0.12	37.82	0.04	27.98
LAN transmission [V(*t6*)]	4.40	4.40	3.89	3.90	0.0	0.0
Message size [2cov(*t2*,*t6*)+2cov(*t3*,*t6*)+ 2cov(*t4*,*t6*)+2cov(*t5*,*t6*)]	8.58	14.04	7.02	10.47	0.0	0.0
Other	0.42	5.32	0.14	3.20	0.10	0.16

Due to the simplified CPU model, the predicted contribution of the initial CPU queueing to the variance of message transit times is estimated less accurately than the other contributions.

Figures 13-14 illustrate the contribution of the various components to the variance of reply message transit times at 16% and 57% channel utilization.

At 16% LAN utilization, the message size has the largest contribution to the variance of the message transit times. The other significant contributions result from the components DUX transfer and LAN transmission, both of which are a direct function of the message length. Contention for the LAN contributes less than 2% to the variance.

At 52% LAN utilization, the contention for the LAN contributes the most to the variance of the message transit times followed by initial CPU queueing. The value of the variance of the DUX transfer and LAN transfer components remained practically the same as at 16% utilization. However, their contribution to the total variance is diminished by the large contribution of the other two elements.

The above analysis indicates that at low channel utilizations the high variability of the message transit times is almost exclusively caused by the variability of message sizes, i.e., by the workload. At high utilizations, this variability is increased by the variability added by the contention for the LAN and to a lesser degree by the contention for CPU.

8. MAJOR CONCLUSIONS

The Prophet team developed and validated analytical and simulation models of DUX/LAN which correctly predict performance measures relevant for the Team Computing environment. We discovered that interactive design engineering application programs that are different from the application's point of view exhibit practically the same behavior from the LAN's perspective. In the process of measuring the lab cluster and working with the models, we reached the following conclusions about LAN behavior:

1. The number of collisions is almost constant at channel utilizations below 8%, increases linearly for utilizations between 8% and 15%, and grows nonlinearly afterwards. Most colliding frames will collide only once or twice before successful transmission.
2. The LAN performs well at 33% utilization, despite claims to the contrary in the literature. Mean message transit times show slight nonlinear increase with rising channel utilization. The transit time increase is about 70% at 50% channel utilization, but that is still small in comparison with the message processing times.

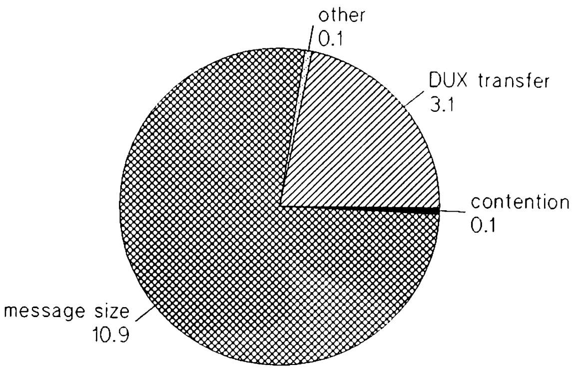

Figure 13. Contributions to Variance of Message Transit Times at 16% Utilization (in msec*msec)

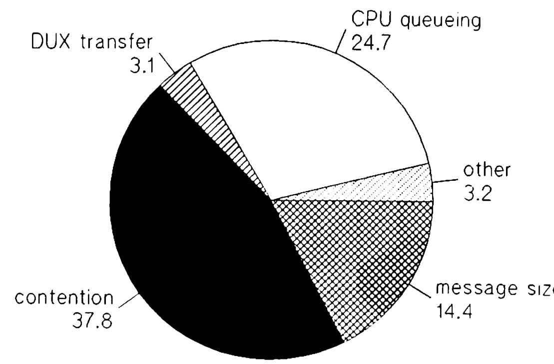

Figure 14. Contributions to Variance of Message Transit Times at 57% Utilization (in msec*msec)

3. Currently, the message communication time across the LAN is dominated by DUX and LAN transmission delay, not by LAN contention.
4. Considerable variability is seen in the message transit time. The standard deviation of the message transit time increases slowly up to a channel utilization of 25% and sharply thereafter. Since this sharp increase occurs at lower channel utilization than the increase in the mean message transit times, an increased standard deviation is an early indicator of performance degradation.

   At low channel utilization, the variability is caused almost entirely by the workload. At about 50% utilization, the variability is dominated by contention for the channel and collision resolution.
5. Broadcast traffic has a direct and significant effect on user transaction response time, but this effect is really due to increasing station CPU utilization.
6. Increasing the channel utilization shows a cascade effect: At utilizations below 10%, the LAN behavior is dominated by the message size distributions. Between 10% and 30% utilization, the collision rates increase but there is little effect on the mean message transit time and its variance. Between 30% and 50%, as the number of collisions increases further, the variance of the message transit time starts to increase rapidly. Above 50% LAN utilization there is a noticeable increase in the mean message transit time and the variance is excessive.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support and help provided by Dr. Carl Morgenstern, project manager, and Mr. Shane Fazzio, member technical staff.

REFERENCES

Devai, J.J. (1989), "Simulation Modeling of Ethernet 2.0/IEEE 802.3 Local Area Networks," Technical Report 40RC-7843-0013, Hewlett-Packard Company, Fort Collins, CO.

Devai, J.J., Fazzio, R.S., Kerrigan, T.C., Molloy, M.K., and Morgenstern, C.F. (1989a), "Validation of DUX/LAN Model for PROPHET," Technical Report 40RC-7843-0011, Hewlett-Packard Company, Fort Collins, CO.

Devai, J.J., Fazzio, R.S., Kerrigan, T.C., Molloy, M.K., and Morgenstern, C. F. (1989b), "LAN Behavior in the Distributed HP-UX Workgroup Environment," Technical Report 40RC-7843-0015, Hewlett-Packard Company, Fort Collins, CO.

Institute of Electrical and Electronic Engineers (1985), *Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specification*, IEEE Standards for Local Area Networks, ANSI/IEEE 802.3, IEEE, New York, NY.

Jones, R.V. and Lindheimer, P.C. (1989), "Performance Analysis of Technical Computer Systems," In *Proceedings of the 1989 International HP-Users Conference*, E. Van Ocken, J. Janssens, and H-J. Zebisch, Eds., Brussels, Belgium, TE/UX/03.

Kerrigan, T.C. (1989), "A Queueing Network Performance Model for CSMA/CD LANs," Technical Report 40RC-7843-0014, Hewlett-Packard Company, Fort Collins, CO.

Lam, S. (1980), "A Carrier Sense Multiple Access Protocol for Local Area Networks," *Computer Networks*, no. 4.

Molloy, M.K. (1989), "Characterization of LAN Traffic for PCDS on HP-UX 6.0," Technical Report 40RC-7843-0012, Hewlett-Packard Company, Fort Collins, CO.

Morgenstern, C.F. n.d., "Performance Study of PCDS on DUX," Technical Report, Hewlett-Packard Company, Fort Collins, CO.