Basic rate ISDN workstation traffic patterns

George Pavlou and Graham Knight present the results of an assessment on how modes of usage currently prevalent on LANs might translate to the ISDN workstation

At present, the PC class of machine equipped with a modem and connected to the public switched telephone network (PSTN) finds many applications; access to electronic mail services, access to office computers by home-workers, occasional access to a mainframe from an isolated office, etc. Two developments are coinciding which will affect this usage; PCs are being upgraded into much more sophisticated workstations, and the PSTN is to be replaced by the integrated services digital network (ISDN), a digital service offering much greater bandwidth. These developments will have a significant effect on the ways in which isolated machines are used. Some results are presented from an assessment made at University College London on the extent to which modes of usage currently prevalent on LANs might translate to the ISDN workstation. Measurements have been made of traffic on a LAN, and extrapolations have been made to gauge the effects of interposing ISDN both on performance and on cost.

Keywords: basic rate ISDN, LAN, workstation, network file system, performance

Integrated services digital network (ISDN) services are gradually being introduced in Europe and North America. Two classes of access to the ISDN are on offer; basic rate ISDN (BRISDN), which offers two 64 kbit/s data channels to ordinary telephone subscribers down the existing twisted pair wiring, and primary rate (PRISDN), which offers 30 64 kbit/s data channels down a 2 Mbit/s trunk[1].

Department of Computer Science, University College London, Gower Street, London WC1E 6BT, UK

PRISDN is aimed mainly at the connection of PABXs to the public telephone network.

By today's standards, the bandwidth offered by these two services is unexceptional. Existing local area networks (LAN) are already in the 10 Mbit/s range, and the newer class of high-speed LAN and metropolitan area network (MAN) are operating in the 100 Mbit/s range. The proposed 'broadband ISDN' (BISDN) is intended to operate in the 100 Mbit/s range or higher, though public BISDN services are still many years off. This disparity in bandwidth has led some people to conclude that BRISDN and PRISDN are technologies that have come too late, and that they will be of little use for modern data applications. This conclusion ignores the one unique feature of ISDN; potentially it is ubiquitous. Every one of the existing 400 M telephone connections in the world is potentially an access point to a reliable 128 kbit/s, low error rate digital service.

It seems likely that users who currently make use of 1200 bit/s connections across the public switched telephone network (PSTN) will welcome the opportunity of a 100-fold increase in bandwidth. Many of these users will have experience of the style of working which is prevalent on today's LANs; shared servers, transparent access to servers, especially file servers, graphical interfaces, etc. They will be keen to see similar facilities available from their home workstations. Ideally, their view of their computing resources from home should be identical to their view from work.

At first glance there seems to be a gross mismatch between LAN bandwidth and the maximum 128 kbit/s available from BRISDN. However, it should be remembered that the 10 Mbit/s on an Ethernet is simplex and shared between many stations, while the 128 kbit/s BRISDN bandwidth is full-duplex and dedicated to a

0140-3664/90/010587-08

single workstation. In our study we have concentrated on the analysis of traffic due to the operation of the Sun Network File System (NFS)[2], which provides transparent access to distributed filestores. NFS traffic constitutes the majority of the traffic generated by the workstation in our environment.

Unfortunately, at the time of writing there was not a proper BRISDN service in the UK, so practical experiments with real users on BRISDN workstations were not possible. The Computer Science Department at University College London (UCL) is keen to conduct such experiments, and has begun planning the design of workstation software and LAN-ISDN gateways in anticipation of a proper service during 1990; we expect international services to follow soon after. To try to assess the likely traffic patterns from a workstation and through a gateway, we have measured traffic from workstations on a LAN — concentrating on remote file access. We have then attempted to assess the performance penalties and costs that would be incurred if such traffic were imposed on the ISDN.

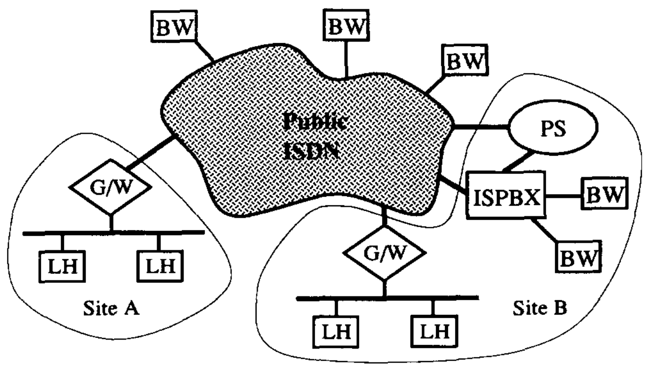


Figure 1. Remote workstation access through ISDN

This configuration with individual workstations and LANs attached to ISDN and able to interwork is shown in Figure 1. The two LAN-based sites A and B can use PRISDN connectivity to link their LANs via gateways. Any of the BRISDN workstations (BW) — possibly used by home-workers — may access the LAN hosts [LH] via the same gateways. The PRISDN server (PS) may be accessed by the BRISDN workstations directly, and by the LAN hosts via a gateway. Site B has an ISPBX conforming to CCITT ISDN standards which provides connectivity between local BRISDN workstations and the server.

TYPES OF TRAFFIC

Traffic to the PSTN workstation is dominated by remote terminal access. Here the establishment of the PSTN circuit is directly controlled by the user. Traffic consists of typed characters and responses, either local or remote echo may be used. Even at 1200 bit/s the use of screen editors is awkward, and most editing is done on the local PC with the results transmitted to the remote host. Thus, short file transfers are embedded in the terminal session. Lack of bandwidth and the unsophisticated nature of the communications software and PC operating system mean that, almost invariably, no other activity may take place while the file-transfer is in progress. Lack of bandwidth also precludes the use of graphics-based applications. Even non-real-time applications, such as the transmission of multimedia mail, becomes difficult.

On a LAN, however, bandwidth is in relative abundance and is free at the point of use. Workstations have bit-mapped screens and run multi-tasking operating systems. Thus, at a given moment, a user may be transparently accessing a file on a remote server while, at the same time, a daemon process on the local machine may be receiving mail and updating the user's mail box. It is not unusual for half a dozen network activities to be taking place simultaneously on a workstation. The control of communications is taken out of the user's hands and he has the illusion of using a single large system.

The activity which dominates the UCL LANs is transparent file access. We are heavy users of the Sun NFS, and a typical workstation mounts around eight filestores via NFS. The key question we have set out to answer is how well this style of working would operate across the ISDN. The aim would be to allow transparent access to shared files from a set of BRISDN workstations. This would allow several home workers to cooperate on a single task with a shared file-store being maintained centrally. It could also provide access to remote mail boxes.

SUN NETWORK FILE SYSTEM

Sun NFS[2] allows file sharing between Unix systems in an almost completely transparent way. Files stored on remote systems may be accessed by user processes exactly as if they were on the local disc. The NFS software — which resides in the Unix kernel — utilizes a Remote Procedure Call (RPC) mechanism to move data blocks to and from the remote system. Such a system clearly allows the sharing of physical resources such as disc space, it also promotes user mobility (in that the same directory hierarchy can be seen from many workstations), and collaborative work (in that several individuals can access a single set of data).

NFS traffic falls into two broad categories: operations relating to navigation of the directory 'tree' (directory listings, etc.), and transfers of data blocks. NFS operates 'read-ahead' and 'write-behind' strategies to minimize network traffic — even at Ethernet speeds, access via the network is slower than access to a local disc. The effect of these strategies is that, if NFS finds it necessary to move a single byte between machines, it tends to move 8 kbytes, i.e. there is extensive caching.

8 kbyte RPC messages must be fragmented before transmission on the Ethernet: this is done in the network layer. The maximum sized Ethernet frame is approximately 1500 bytes. Thus, a frequent NFS transfer pattern is 8 kbytes fragmented into five 1.5 kbyte and one 0.9 kbyte frames (the surplus is accounted for by packet and frame headers).

EXPERIMENTAL ENVIRONMENT

The target for the experiments was a Sun 3/50 workstation with 150 Mbyte local disc on which were stored the Unix operating system and heavily used programs. The user's home directories were also accessible remotely. Although this configuration would not be a realistic one for an ISDN workstation — the home directory would then be on a local disc — it was a good one from the point of view of studying remote file access. The workstation under test was situated in an individual's office, and was effectively single user.

Measurements were taken on another Sun 3/50 operating in promiscuous mode and making use of the Sun 'NIT' interface. Potentially, this allows the recording of all traffic on an Ethernet. In practice, some frames are missed when the load on the monitoring machine is high. However, all NFS RPCs have a sequence number which is used to match replies to requests. Inspection of these sequence numbers suggests that only a tiny portion of the total number of frames is missed. Figure 2 shows the experimental set-up: the workstation (WS) is being monitored by the monitoring station (MS) while accessing the remote filestore through the fileserver (FS).

Filtering was applied so that only traffic to and from the target workstation was recorded. Further filtering was applied to select NFS traffic which always uses a well-known port. Measurements were generally conducted over a period of approximately one hour. The user was not informed when measurements were being taken. Experiments were conducted over a period of several weeks. A file was recorded for each measurement session, showing for each frame: the time it was recorded, source and destination addresses, port numbers, the length of the frame in bytes and the NFS sequence number. NFS uses the DARPA UDP and IP protocols, and the headers for these protocols were included in the length records.

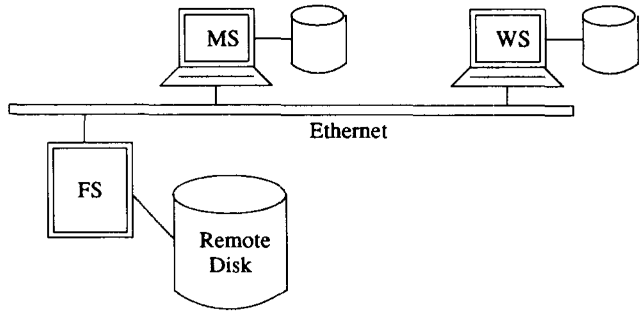


Figure 2. Experimental configuration

The accuracy of the frame timestamps was ±10 milliseconds, as dictated by the Sun 3/50's clock.

PACKET SIZE DISTRIBUTION

Figure 3 shows packet sizes averaged over several measurement sessions. As can be seen, there is considerable asymmetry between traffic to and from the workstation. Also, the distributions are heavily bi-modal. This is exactly what one would expect, given the NFS traffic and fragmentation patterns described above. The peak at around 150 bytes results from file-store navigation activity, and that at 1500 bytes represents full Ethernet frames resulting from fragmented bulk transfers of 8 kbytes. The much smaller peak at 9000 bytes corresponds to the residual data after fragmentation.

THROUGHPUT AND DELAY

A very important determinant of the performance of the ISDN workstation is the throughput it requires from the attached network. If this exceeds the bandwidth available

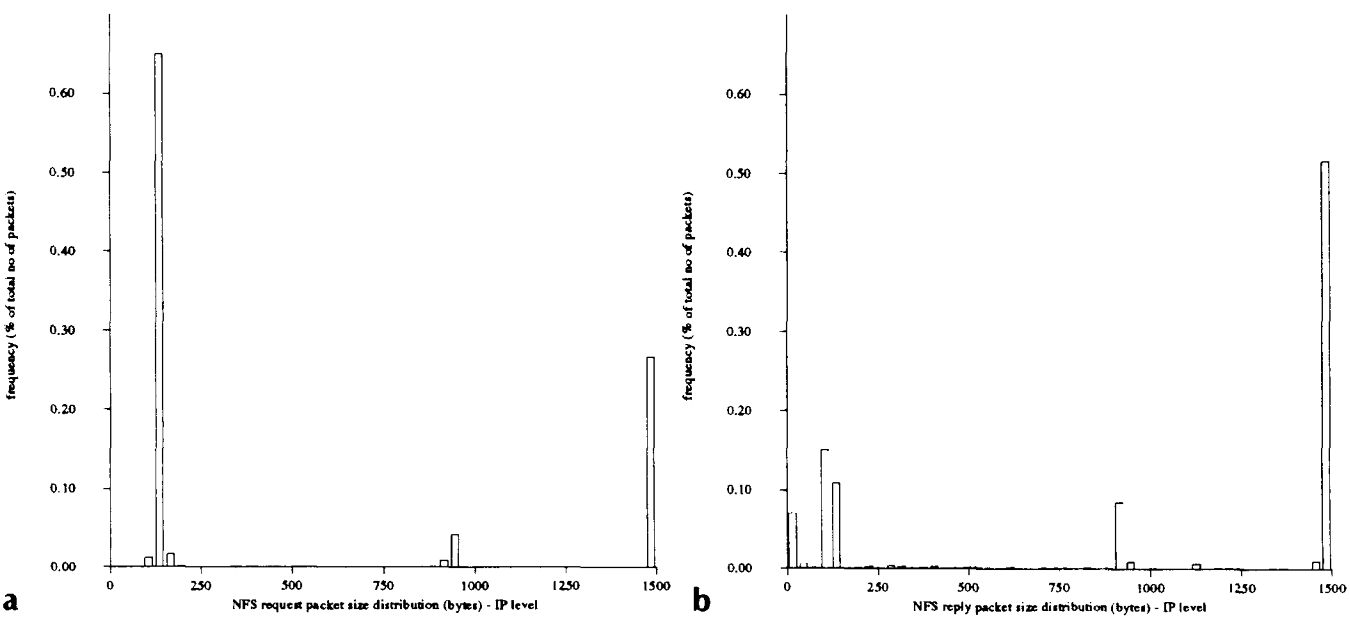


Figure 3. Distribution of NFS packet sizes

Table 1. Average NFS throughput over all recorded sessions

Workstation → Server	5.29 kbit/s	1.19 pkts/s
Server → Workstation	12.72 kbit/s	1.73 pkts/s

from the ISDN, then data will be queued and the user will experience delays. The extent of ISDN-imposed delay is a key factor in determining the acceptability of the complete system from the user's point of view. The mean throughput throughout all our recorded sessions is shown in Table 1.

These values are well within the bandwidth offered by a single ISDN channel, but this is not in itself much comfort as the traffic is very 'bursty', and it has peaks well above 64 kbit/s. These peaks will be ironed out by buffering in the server, the LAN gateway to the ISDN (if any) and the host. The effect will be to increase the network imposed delay as perceived by the user. We have adopted two approaches in attempting to assess the seriousness of this problem:

1 We have calculated the 'instantaneous' bit-rate used across the Ethernet to assess how frequently this exceeds BRISDN rates.
2 We have calculated the additional delay that would be imposed by the ISDN on our sample data.

Bit rates

To estimate the throughput we have used the moving average method on the sample data. In Figure 4 we show the variation of the bit rate throughout two representative sessions. The period over which we average the bit throughput is one second in each case. This value was chosen with user perceived delays in mind. If the offered bit rate exceeds that available for longer than one second, then these delays are likely to be very noticeable. The figures show traffic in one direction only. For some activities there is heavy asymmetry between traffic volumes in the two directions. In these cases we have concentrated on the heavier volume.

The very large fluctuations in throughput are clearly shown as bursts of activity following idle periods. In the first session (see Figure 4a) it can be seen that the throughput is sometimes very high, at times exceeding 1000 kbit/s. The second session (see Figure 4b) shows much more manageable peaks. The difference between these two sessions is in the user's activity.

In the first session, the user was developing a large software system, cycling between editing, compiling/linking and testing. The large data flows which occur seem to be associated with the linking process. In this, a very large library, stored on the remote machine, was searched and modules were retrieved. The search algorithm makes use of an index which results in random accesses to the library file. In the second session the user was processing electronic mail with the mailbox being on the remote machine. The electronic mail interface in use begins by splitting a file of incoming messages into a set of files, one file per message. This is a local operation on the remote system, and generates no NFS traffic. From then on, mail processing consists of retrieving and displaying complete messages, i.e. complete files.

We mentioned that NFS uses extensive caching by reading blocks of 8 kbytes of data, at least one at a time, as it anticipates further sequential access. This optimization scheme works well in the case of sequential file access, but it has the effect of increasing network traffic in the case of random access. We performed the following two experiments to verify this: in the first we copied the large library mentioned above from the file server to the local

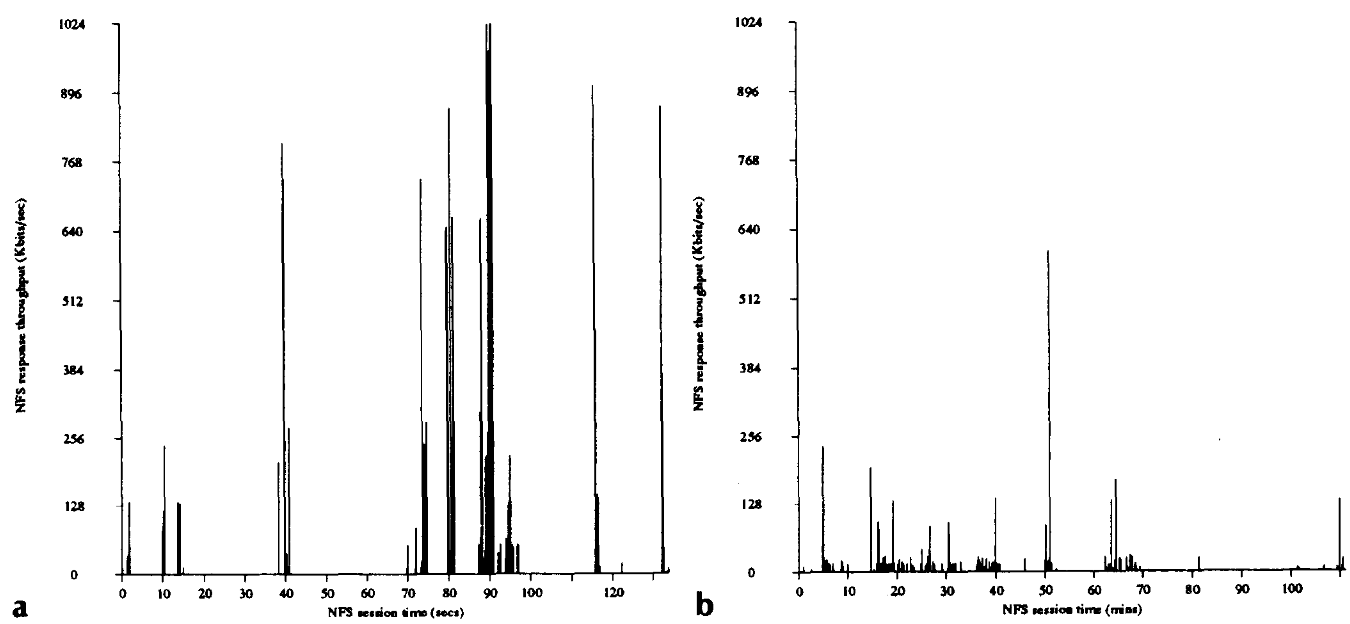


Figure 4. NFS throughput variation for two example sessions

machine. In this case NFS operation was almost optimal, the amount of traffic corresponding roughly to the size of the transferred file. In the second experiment, we performed a linking operation with the remote library. In this case the performance was poor as the search that retrieved 16 kbytes of useful data generated 48 kbytes of traffic.

This explains the difference in throughput observed in the two presented sessions, as in the first the random access defeats the NFS caching scheme increasing the already large amounts of traffic, while in the second, the sequential file access is exactly the kind of operation for which NFS is optimized.

ISDN-imposed delay

The periods in the sessions above during which the bandwidth required exceeds that offered by the ISDN will clearly be subject to some degradation of service in comparison with the Ethernet case. To assess this degradation from the user's point of view, we have tried to calculate the additional delay that would be imposed by the ISDN. For this purpose, we have assumed that just one ISDN B-channel is in use, i.e. that we have 64 kbit/s available. Similar calculations could of course be performed for the case when both the ISDN basic rate B-channels were available.

The ISDN-imposed delay comprises the transmission delay at 64 kbit/s plus the queuing delay (Ethernet transmission delay at 10 Mbit/s is considered negligible). Queueing delay has been calculated by shifting the start times of transmissions to allow for the late termination of the preceding ones. The limitation of this method, though, is that it does not take into account the fact that, in most cases, a transmission may not begin before a related reply is received. At the network level in which we are operating it is impossible to tell the precise dependencies between requests and replies, so we have looked at one way delay only. This means that the *absolute* values of the delays we have found should be treated merely as indicators to the general trend in the delays.

Figure 5 shows the calculated distribution of the delays for the two sessions described above. It can be seen from Figure 5a that compilation and linking using a remote library is not really feasible across the ISDN, as a significant number of frames delayed by more than 10 seconds. In fact, there are frames that experience delays up to three minutes, while only 54% of frames experience delays less than one second. In practice, these long delays would exceed NFS timeout values causing the whole linking process to abort. However, Figure 5b shows that, for mail reading, delays are quite reasonable, with 91% of frames being delayed by no more than one second and the maximum delay to be less than nine seconds.

The conclusion is that operation of transparent file access across a single B-channel is feasible, provided that care is taken in the way in which data is distributed between the local and remote machines. However, types of activity associated with repetitive random access over large files such as linking with remote, big libraries may cause unacceptable delays. In these cases, it might be necessary to maintain shadow copies of data on the local machine to reduce traffic. We expect to be able to tune the performance of NFS by varying the cache size and the timeout periods before cached data are considered stale, as NFS checks periodically to see if the file it is reading was modified. It would also be worth considering alternative caching schemes geared to operation over low bandwidth channels, as the existing scheme assumes the abundance of a LAN bandwidth.

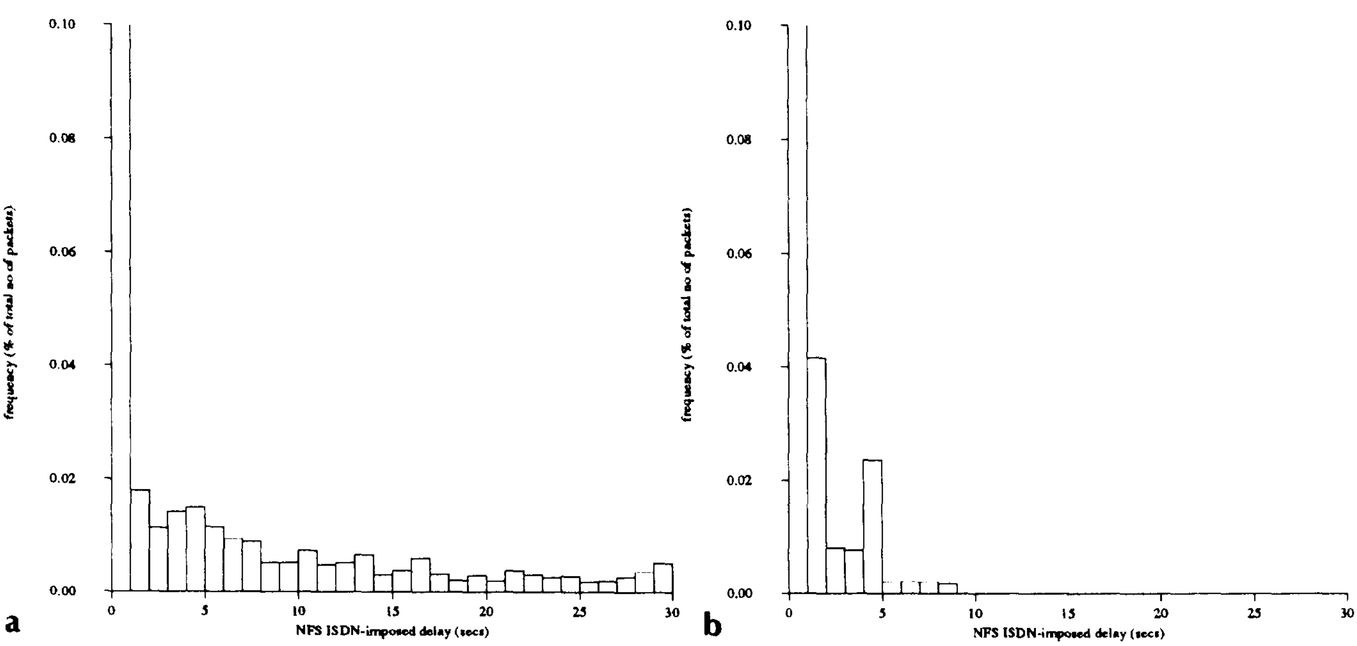


Figure 5. NFS ISDN-imposed delay

CIRCUIT MANAGEMENT

To estimate the real cost of using the ISDN workstation for file access we need to be able to estimate the proportion of the total time for which an ISDN circuit needs to be kept open. In practice, some circuit management strategy would be employed to disconnect an idle circuit.

An optimal strategy for circuit management will be dependent on the traffic characteristics and the tariff scheme. It is not quite clear if the tariffs will penalize circuit set-up, but it seems possible that there might be a circuit set-up charge. This will result in a cost penalty associated with a scheme which disconnects too enthusiastically. Irrespective of the tariff, there is always a penalty for closing down the circuit too often, that is the additional delay incurred while re-opening it. Obviously, a strategy which is too slow to close idle circuits will result in increased cost and possibly wasted bandwidth.

The strategy we have examined is the obvious one of closing the call after a quiet period of (say) q seconds. We have tried to determine a sensible value for q for our experimental sessions in the context of the ISDN tariff to be expected from British Telecom. We have assumed a stepped tariff such as that currently employed for long-distance calls across the PSTN. At peak rate, this charges 4.4 pence for each whole or part period of 18 seconds, while no penalty is imposed for circuit set-up. The stepped tariff complicates matters slightly, since once the call has entered a new 18 second period there is no point closing it until that period has almost expired.

We have tried the above strategy against the data from several sessions modified for the additional queueing delay as described above. In Figure 6 we show the frequency of disconnects per hour and the connection cost per hour as functions of q. The circuit set-up cost was not taken into account as we did not know how much this would be in relation to the stepped tariff assumed. The cost graph shows the ratio of connect time *versus* total session time, knowing that a full hour session costs (3600/18*4.4 = 880 pence. As an example of using the graphs, we can see that if we close calls after idle periods of 5 seconds, the connect time is reduced to approximately 24% of the total session time while disconnects will occur once every 3.75 minutes.

The optimal value for q should balance the user's sensitivity to circuit set-up delays and the cost to minimize these. However, a sensible choice of q can only be made in the light of knowledge of the traffic patterns incurred. Traffic patterns vary throughout the period of a session as the user's activity changes. An adaptive strategy should be sought to vary the value of q accordingly. has led to a study of the distribution of packet interarrival times to pursue an analytical way of devising the optimal value for q. There is much work still to be done in this area: our results so far are presented below.

Packet inter-arrival times

Intuitively, interarrival intervals are of three types:

1 Those resulting from the user thinking or resting. These are of the order of seconds or minutes.
2 Intervals between RPC requests resulting from processing delays. These will range from hundredths of a second to a few seconds.
3 Intervals between frames resulting from fragmentation at the network level. Such frames are normally sent 'back-to-back' and the intervals between them are of the order of milliseconds or less.

Intervals in the third category are too small for us to measure with the current techniques and, in any case,

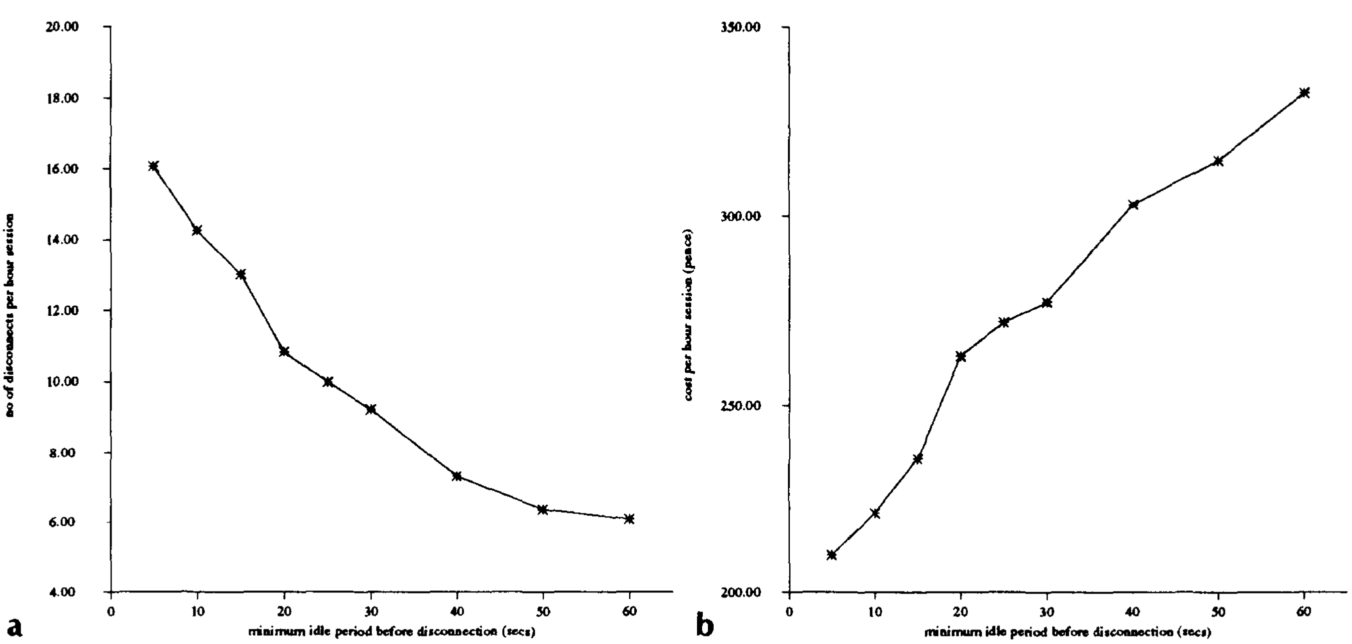


Figure 6. ISDN channel disconnects for NFS traffic

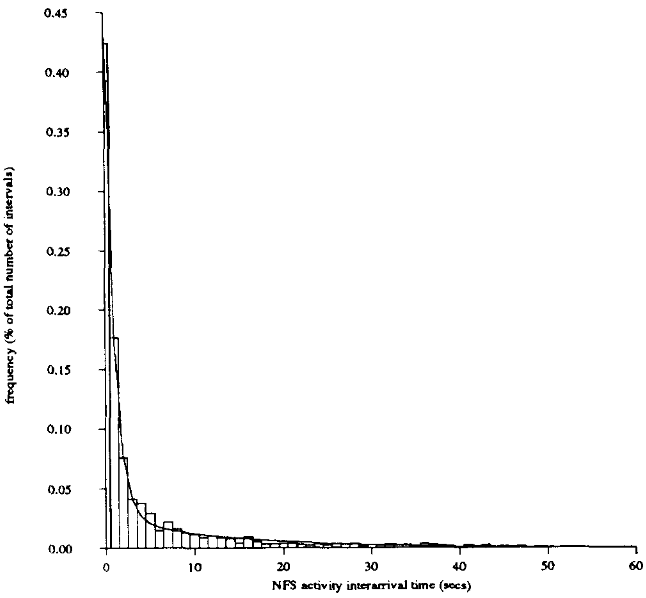


Figure 7. NFS interarrival time distribution

they are too small to have any impact on a circuit management strategy. We have eliminated these small intervals from our statistics by treating a series of frames separated by gaps of less than 0.2 seconds as a single event. We have also disregarded gaps of more than one minute — there are very few of these. With these assumptions, Figure 7 shows a distribution of frame interarrival times. These data were gathered over a one week period from different workstations, and concern the overall traffic between the workstation and the server in both directions. (The distributions in each direction are very similar, as the exclusion of small intervals eliminates the gaps between the back-to-back frames which would otherwise cause asymmetry.)

It is clear from the shape of the distribution in Figure 7 that it belongs to the exponential family. However, attempts to fit a straight exponential distribution gave poor results when subjected to a chi-square test. The classification of the interarrival gaps into three types as outlined above suggests that a combination of distributions might be more appropriate. This seems to be the case, as much better results were obtained with a hyperexponential distribution; the curve fitted is shown in Figure 7.

A hyperexponential distribution consists of more than one exponential distribution with different means and weightings. The ones we have fitted consists of two exponential distributions. The probability density function (pdf) for a hyperexponential distribution with two terms is:

$$f(t) = \alpha_1 \lambda_1 e^{-\lambda_1 t} + \alpha_2 \lambda_2 e^{-\lambda_2 t} \text{ for } t \geqslant 0$$

with mean:

$$\frac{1}{\mu} = \frac{1}{\mu_1} + \frac{1}{\mu_2} \text{ where } \lambda_1 = \frac{1}{\mu_1} \text{ and } \lambda_1 = \frac{1}{\mu_2}$$

We have developed an algorithm to find the parameters of the hyperexponential distribution that gives the best fit on the sample data. An exponential curve fit is easy enough as the only unknown parameter is the mean, which can be calculated from the histogram. A hyperexponential case is more complicated though: there are three unknown parameters for the two constituting exponential distributions and only two equations, from the mean and variance of the histogram. We used an optimization algorithm to find the values of the parameters that achieve the best fit. We also tried hyperexponential distributions with three terms, but the results for the chi-square test were no better.

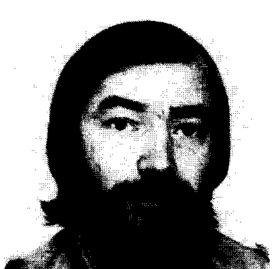
The parameters that gave the best fit were: $\mu_1 = 1$ s with weighting $\alpha_1 = 0.65$ and $\mu_2 = 13.5$ s with weighting $\alpha_2 = 0.35$. The mean interarrival time for all the data is 5.35 s. This fit corresponds to a chi-squared value of 85.1 for 50 degrees of freedom. The hyperexponential assumption is quite consistent across our sample data, allowing successful fittings of distributions with similar parameters on data from a single or multiple sessions.

CONCLUSION

The ISDN may allow new methods of working from the home workstation employing techniques which currently operate across LANs. Principal among these techniques is transparent file access.

An analysis of file access traffic across LANs at University College London has shown that such traffic is extremely bursty, and that the total volume of traffic and the bandwidth requirements are dependent on the distribution of data between client and server, the caching

__George Pavlou__ is a research associate in the Department of Computer Science at University College London (UCL), UK. He obtained his BSc in mechanical engineering from the National Technical University of Athens, Greece, in 1983, and his MSc in computer science from UCL in 1986. Since then he has worked at UCL on various communications-related research topics including distributed systems, network management, performance evaluation and ISDN. He has been involved in several collaborative research projects sponsored by the EEC.



__Graham Knight__ is Senior Lecturer in the Department of Computer Science at University College London, UK. He obtained his BSc in mathematics from the University of Southampton in 1970, and his MSc in computer science from UCL in 1980. Since that time he has worked on research topics involving LANs, network management and ISDN. He became a lecturer in the department at UCL in 1986. He has been involved in several collaborative research projects sponsored by the EEC.

strategy employed, and the nature of the user activity which is generating traffic.

Calculations have been performed to assess the feasibility of operating this sort of traffic across a 64 kbit/s ISDN circuit. The results suggest that such an operation may be feasible provided care is taken in system configuration and choice of activity. There is scope for tuning NFS operation to suit the ISDN case.

A further analysis has concerned the feasibility of closing ISDN circuits during idle periods. There is a trade-off here between cost savings and increased delays seen by the user. However, our figures suggest that significant cost savings are possible.

A study of the interarrival times has found that a hyper-exponential distribution gives a good fit. Further work is needed to determine a circuit management strategy which can adapt to changing patterns in the distribution of the interarrival times.

ACKNOWLEDGEMENTS

This work has been carried out under ESPRIT project 2404 'PROOF' (Primary Rate ISDN OSI Office Facilities). The partners are 3-Net (UK-prime contractor), System Wizards (Italy), Siemens-Nixdorf Computer (Germany) and University College London (UK).

REFERENCES

1 *Recommendation I.412: ISDN User-Network Interfaces, Interface Structures and Access Capabilities* CCITT Blue Book CCITT, Geneva, Switzerland (1988)

2 *Sun's Network File System — Revision A* Sun Microsystems Inc, USA (May 1985)