



Quaestiones Informaticae

Vol. 2 No. 3

September, 1983

Quaestiones Informaticae

An official publication of the Computer Society of South Africa and
of the South African Institute of Computer Scientists

'n Amptelike tydskrif van die Rekenaarvereniging van Suid-Afrika en
van die Suid-Afrikaanse Instituut van Rekenaarwetenskaplikes

Editor: Prof G. Wiechers

Department of Computer Science and Information Systems
University of South Africa
P.O. Box 392, Pretoria 0001

Editorial Advisory Board

PROFESSOR D. W. BARRON

Department of Mathematics
The University
Southampton SO9 5NH
England

PROFESSOR K. GREGGOR

Computer Centre
University of Port Elizabeth
Port Elizabeth 6001
South Africa

PROFESSOR K. MACGREGOR

Department of Computer Science
University of Cape Town
Private Bag
Rondebosch 7700
South Africa

PROFESSOR M. H. WILLIAMS

Department of Computer Science
Herriot-Watt University
Edinburgh
Scotland

MR P. P. ROETS

NRIMS
CSIR
P.O. Box 395
Pretoria 0001
South Africa

PROFESSOR S. H. VON SOLMS

Department of Computer Science
Rand Afrikaans University
Auckland Park
Johannesburg 2001
South Africa

DR H. MESSERSCHMIDT

IBM South Africa
P.O. Box 1419
Johannesburg 2000

MR P. C. PIROW

Graduate School of Business
Administration
University of the Witwatersrand
P.O. Box 31170
Braamfontein 2017
South Africa

Subscriptions

Annual subscriptions are as follows:

	SA	US	UK
Individuals	R6	\$7	£3,0
Institutions	R12	\$14	£6,0

Circulation manager

Mr E Anderssen
Department of Computer Science
Rand Afrikaanse Universiteit
P O Box 524
Johannesburg 2000
Tel.: (011) 726-5000

Quaestiones Informaticae is prepared for publication by Thomson Publications South Africa (Pty) Ltd for the Computer Society of South Africa and the South African Institute of Computer Scientists.

A Disk Space Management System

A.J. Cuthbertson

University of the Witwatersrand Computer Centre

I.J. van Niekerk

University of the Witwatersrand Computer Centre

T. Turton

IBM South Africa

Abstract

This paper describes the research that was undertaken in designing a disk space management system. The system has strong similarities to the management of real storage in a virtual storage environment. These similarities were exploited in the design and evaluation of algorithms for implementation of the system. Research was based on modelling the system with historical data during the design phase. Research was conducted in three phases. Firstly, in determining actual historical disk space requirements quantitatively. Secondly, in modelling the proposed system with historical data to verify an observation that most users use the computer intermittently. Thirdly, in modelling the proposed system with historical data to evaluate alternative disk management policies. The system was implemented and actual performance monitored against research results to validate conclusions.

INTRODUCTION

The IBM VM/SP operating system was installed at the Computer Centre of the University of the Witwatersrand in late 1981 as the main vehicle for educational and research computing. The Centre provides a computing service to the campus and some outside users. It supports batch and online processing. All staff members and students may potentially make use of the Centre's facilities. At present, some 3 000 users have been registered to do so.

A characteristic of VM/SP is that there is a fixed minimum amount of online disk space that can be allocated to an individual user. With the standard system each user has a 'minidisk', which is a subdivision of a real disk, that is assigned in increments of 450k bytes. Providing such space to all users was regarded as uneconomical as most are under-graduate students and would require less than the minimum of 450k bytes of permanent storage.

A further consideration was that while a user is logged onto the system, there will be a requirement for an amount of workspace on disk to cater for compiler work areas, program listings, and similar purposes.

Initially, a disk management system was set up where groups of users shared a single large permanent minidisk. A group would typically consist of all the users in a particular department. In addition, when logging on, a user was allocated 450k bytes of disk for temporary work space. This was automatically reclaimed when the session was completed and it was the users responsibility to transfer any files to be retained to the permanent disk before logging off. This system did not work successfully as it impacted the functional capabilities of the operating system. Owing to this an alternative approach was sought.

During the first year of operation of the VM/SP system the Computer Centre staff investigated the disk space usage patterns and requirements of its users. It was observed that users tend to make intermittent use of the computer. There would be bursts of activity lasting days or weeks followed by weeks or months of inactivity.

Based on these findings, a proposal for disk space management was made. This was that users should be allocated a minidisk by the computer, only on demand. It would be large enough to contain both temporary and permanent files, so no distinction need be drawn between them. The minidisk would remain allocated to a user only as long as it was used. If the

user became inactive for a period of time, the computer system would automatically archive all those disk files to magnetic tape and 'reclaim' the disk space. Reclaimed mini-disks would be returned to a pool, to be allocated to the next user requiring space. A catalog of files archived to tape would be maintained for each user, and made available by the computer on request. The user would then be able to migrate selected files back from tape to a newly allocated online minidisk.

RESEARCH AND DESIGN CRITERIA

User Space Requirements

The first phase of research was to examine the total amount of space used by each user under the original shared mini-disk system. Each departmental minidisk was examined and the space occupied by individual users was determined. The figures were used to calculate a distribution of users against amount of disk space used in 100k byte increments. The results of this analysis are indicated Figure 1.

Over 90 percent of the registered users were utilising less than 200k bytes of permanent disk space. This would mean that in the proposed system a minimum sized minidisk would satisfy almost all the users' space requirements.

Proposed Objectives for Disk Management

The concept of reclaiming inactive user's minidisk and making it available to a more active user has certain similarities to the principles of virtual storage management. Reclaiming the disk of the user who had been inactive the longest, would constitute a Least Recently Used or LRU reclaim policy, which Belady has shown to be a good strategy 1. On these grounds the following objectives were established.

- To minimise the number of minidisks in the pool.
- To provide an acceptable time that a user could be absent without the minidisk being reclaimed. That is, to achieve an acceptable 'Retention Period'.
- To minimise inconvenience to the user.
- To avoid a reduction in the functional capabilities of the standard VM/SP operating system.

Modelling a Least Recently Used Policy

The VM/SP operating system keeps accounting records of user

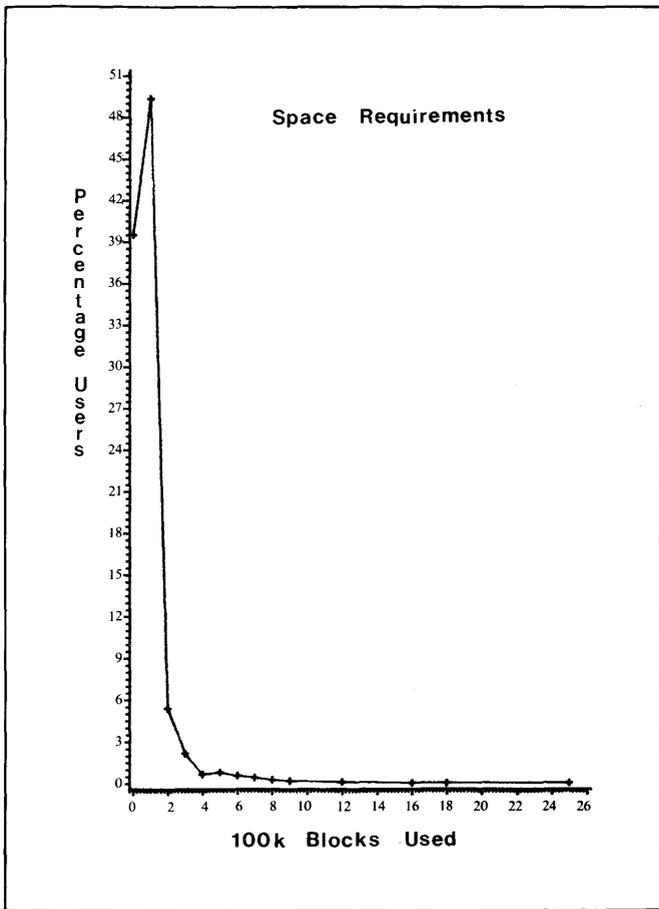


FIGURE 1

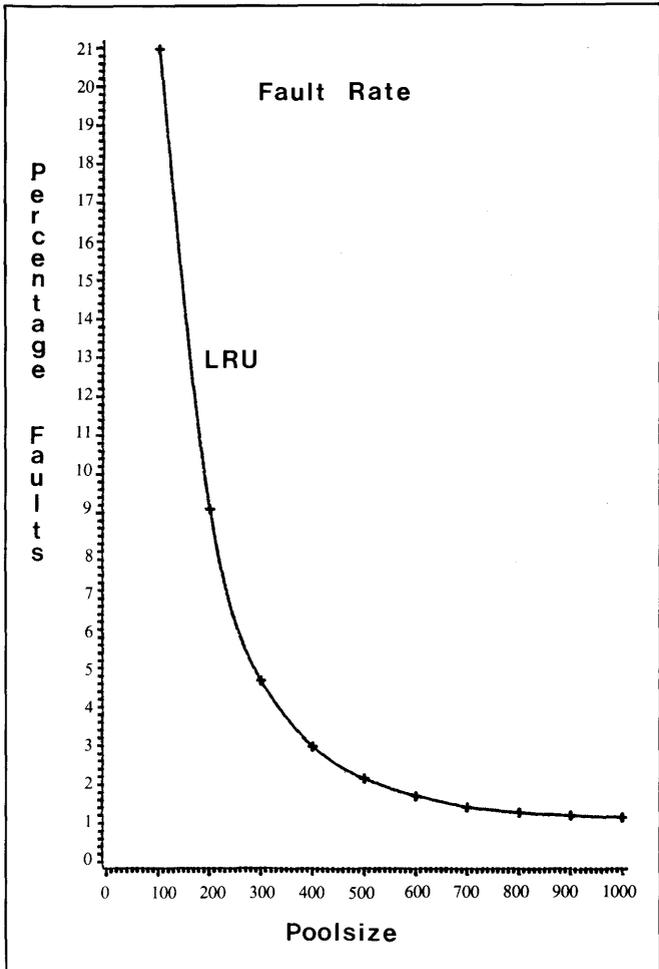


FIGURE 2

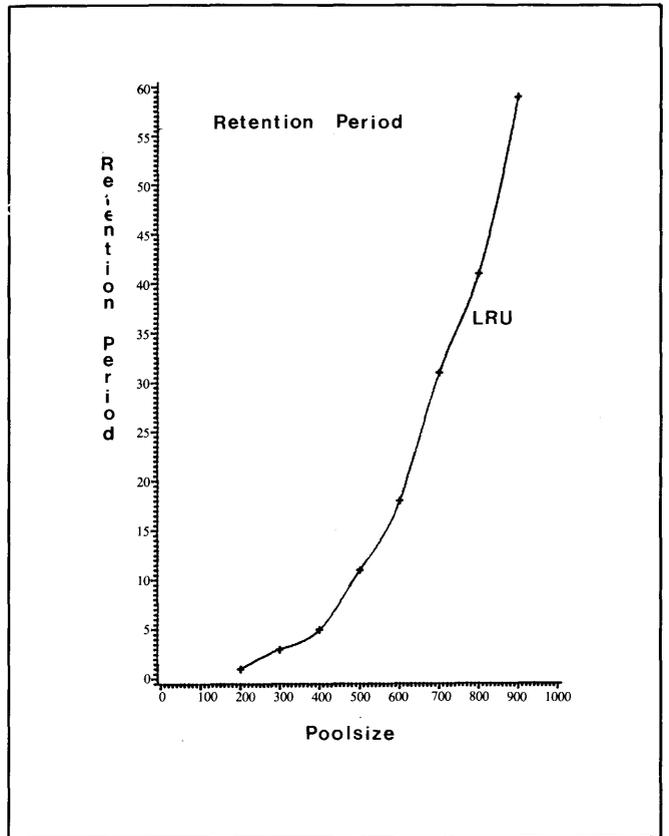


FIGURE 3

activity. A record is generated each time a user logs on to the system recording the time and date. A computer program was written to analyse logon records that had been accumulated over a nine month period covering most of the 1982 academic year.

This program was designed to model an LRU reclaim policy. It processed each logon record and assigned an imaginary minidisk to a user from a pool of available or 'free' disks. The total number of disks in the pool was termed the 'poolsize'. When a minidisk was assigned to a user, it was deemed to be 'active'.

If there were no free disks to be allocated, a 'fault' was deemed to have occurred and the least recently used active disk was reclaimed from its current user and assigned to the new user. The retention period for each reclaim was checked and the minimum for the run was recorded.

During the nine month period about 1500 different users had used the system. The model was run with pool sizes ranging from 100 to 800. Two graphs were produced from the results. The first indicates the percentage of logons which resulted in a fault occurring. The second indicates the minimum retention periods reported. The graphs are shown in Figures 2 and 3.

These results were more promising than expected. The curves indicated that for 750 disks in the pool, there would have been a minimum retention period of 35 days. Only 1.3 percent of the logons would have been faulted. The intermittent use of the computer meant that if this disk management policy could be implemented, a 50 percent saving in disk space would be achieved.

For practical reasons, the LRU policy could not be implemented as modelled. If a disk had to be reclaimed at logon time, the user would have to wait for all of the previous owner's files to be archived onto magnetic tape. This would cause an unacceptable delay.

To solve this problem, it was proposed that disks be preemptively reclaimed at night so that there would be sufficient free disks to meet the number of faults that might occur on the next day. The policy would be based on reclaiming all disks which had not been used for more than a fixed number of days.

This was termed the 'Fixed Retention Period', or FRP policy.

Modelling a Fixed Retention Period Policy

This algorithm was built onto the modelling program. It was run with various pool sizes and a fixed retention period of 21 days. This was considered to be a reasonable threshold between computing activity and inactivity.

In addition to the number of faults, the program reported on 'Pool Empty' situations, which occurred if the faults exceeded the number of free disks. The results are presented in Figure 4.

The Fixed Retention Period policy does not utilize the disk space as efficiently as the LRU policy. For a pool size of 750 disks the percentage of logons faulted is 1,8 compared to 1,3. A higher fault rate was however, considered to be a smaller user inconvenience factor than having to wait during logon for a disk to be reclaimed.

A Fixed Number Policy

At this point, an alternative pre-emptive policy was considered. This would be to reclaim a fixed number of the least recently used disks nightly, so that there would be enough free disks to service the next days faults. This was termed the 'Fixed Number', or FN policy. This Fixed Number policy was also built into the modelling program and it was run on the same basis as before. Percentage faulted logons are shown for all three policies in Figure 5.

For 750 disks the fault factor is only 0,05 percent worse than for the LRU policy. The characteristics of this policy are similar to the LRU policy in that the retention period varies for each pool size.

Conclusions Drawn from Model

Both the Fixed Retention Period and Fixed Number policies would have the advantage of being pre-emptive. Although the fault rate is higher for the former policy, the retention period is fixed and therefore predictable.

It was decided that the Fixed Retention Period policy would be the most desirable. Being pre-emptive, the logon times would be acceptable and the disadvantage of a higher fault factor would be outweighed by offering a completely predictable service to the user.

The Fixed Retention Period policy has another favourable property in a University environment. During the long vacation at year-end there is little usage of the system. The FRP policy reclaims a large number of disks during this time. This means that the system is well equipped to accommodate the large number of new users who appear at the start of the next academic year.

SYSTEM IMPLEMENTATION

The historical data used for the model reflected computing activity during 1982 which was the first year that the VM/SP operating system was available. During that year, users were migrating from an older computer system and it was expected that there would be a substantial growth in 1983 when all users had migrated. For this reason, a pool size of 1 100 minidisks and a retention period of 21 days were chosen to cater for some 3 000 registered users.

The system was implemented by having a service program maintain a directory of all the disks in the pool. At logon time, a message is automatically sent to this program on behalf of the user, requesting access to a disk. If the user is already reflected in the directory, the associated disk is simply attached and the date of last access is updated. If not, the first free disk in the pool is assigned and the users' identity and the date are recorded in the directory.

At night, a reclaim program scans the directory for all entries that are older than 21 days. The files on the associated disks are copied to magnetic tape and an entry added to an Archive Catalog for each file. The program also archives selected files at the request of the user without reclaiming the disk.

A dearchive program is run at hourly intervals during the day to enable users to selectively retrieve files from the Archive tapes.

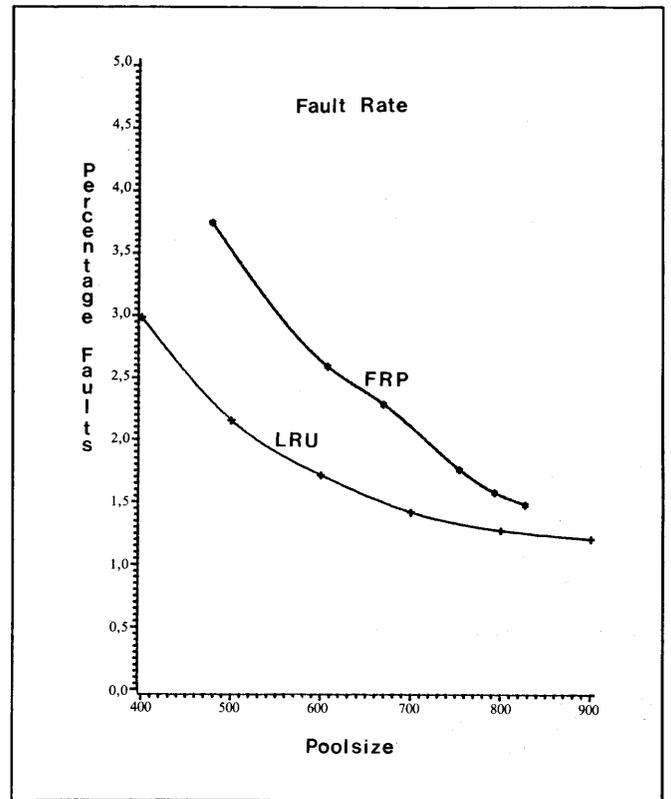


FIGURE 4

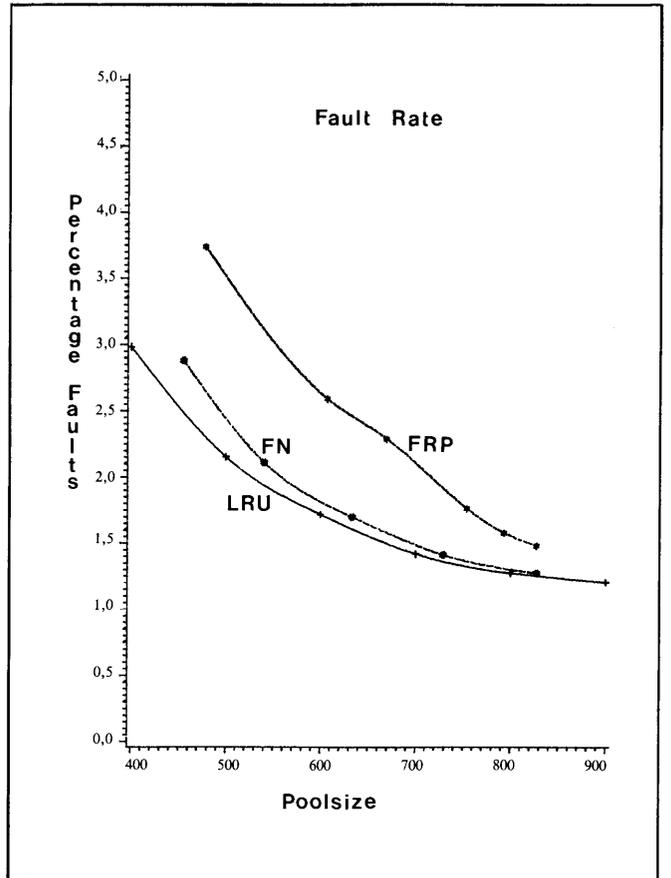


FIGURE 5

The Fixed Retention Period is parameterised in the reclaim program. If the pool of free disks becomes empty, the system operator is requested to run the reclaim program immediately, with a reduced Retention Period. It is hoped that this measure will cater for unexpected fluctuations in user demand on computing.

Based on the analysis of user space requirements, all pool minidisks were set to the minimum size of 450k bytes. If a user required more, a larger disk was added to the pool and marked as 'permanent'. This disk would not be eligible for normal reclaiming. Most of the permanent disks would belong to academic staff and post-graduates and would be managed manually.

Implemented in this way, the disk management system lengthens the logon time slightly, but otherwise has no effect on the functional capabilities of the standard VM/SP operating system.

ACTUAL RESULTS AND FUTURE RESEARCH

The system was implemented in January 1983, and the actual number of faults and active disks were monitored during the first five months. The model has been rerun with the latest historical data to validate the Fixed Retention Period algorithm used.

A modification to the Fixed Retention Period policy has been proposed and the authors believe that this would result in an improved service to the users. The modification would apply the policy as described, but instead of erasing the files when a disk is reclaimed, leave the disk active in the pool, but mark-

ed as free. If that disk then had to be assigned to a different user, the files would be erased at time of assignment. This operation would be almost instantaneous and would not delay the logon time appreciably. If, however, the previous owner of that disk returned before it had been re-assigned, the files would still be online.

This modification would guarantee the fixed retention period, but the files may very well remain online for much longer. Further research is being undertaken to evaluate this proposal.

Other research would relate to varying the frequency of disk reclaim operations, but current efforts have been confined to the case of reclaims on a once per day basis.

CONCLUSIONS

By adapting the concepts of virtual storage management to the management of disk space, it is possible to economise on the amount of online disk space required under conditions of intermittent computer access in a multi-user environment.

ACKNOWLEDGEMENTS

The authors would like to thank Mr L Clarke for producing the graphs presented in this paper.

References

- [1] Belady, L.A. A study of Replacement Algorithms for a Virtual Storage Computer. In: IBM Systems Journal Vol. 5 No. 2 (1966).

Notes for Contributors

The purpose of this Journal will be to publish original papers in any field of computing. Papers submitted may be research articles, review articles, exploratory articles of general interest to readers of the Journal. The preferred languages of the Journal will be the congress languages of IFIP although papers in other languages will not be precluded.

Manuscripts should be submitted in triplicate to: Prof. G. Wiechers at:

Department of Computer Science
University of South Africa
P.O. Box 392
Pretoria 0001
South Africa

Form of manuscript

Manuscripts should be in double-space typing on one side only of sheets of A4 size with wide margins. The original ribbon copy of the typed manuscript should be submitted. Authors should write concisely.

The first page should include the article title (which should be brief), the author's name, and the affiliation and address. Each paper must be accompanied by a summary of less than 200 words which will be printed immediately below the title at the beginning of the paper, together with an appropriate key word list and a list of relevant Computing Review categories.

Tables and figures

Illustrations and tables should not be included in the text, although the author should indicate the desired location of each in the printed text. Tables should be typed on separate sheets and should be numbered consecutively and titled.

Illustrations should also be supplied on separate sheets, and each should be clearly identified on the back in pencil with the Author's name and figure number. Original line drawings (not photoprints) should be submitted and should include all relevant details. Drawings, etc., should be submitted and should include all relevant details. Drawings, etc., should be about twice the final size required and lettering must be clear and "open" and sufficiently large to permit the necessary reduction of size in block-making.

Where photographs are submitted, glossy bromide prints are required. If words or numbers are to appear on a photograph, two prints should be sent, the lettering being clearly indicated on one print only. Computer programs or output should be given on clear original printouts and preferably not on lined paper so that they can be reproduced photographically.

Figure legends should be typed on a separate sheet and placed at the end of the manuscript.

Symbols

Mathematical and other symbols may be either handwritten or typewritten. Greek letters and unusual symbols should be identified in the margin. Distinction should be made between capital and lower case letters between the letter O and zero; between the letter I, the number one and prime; between K and kappa.

References

References should be listed at the end of the manuscript in alphabetical order of author's name, and cited in the text by number in square brackets. Journal references should be arranged thus:

1. ASHCROFT, E. and MANNA, Z. (1972). The Translation of 'GOTO' Programs to 'WHILE' Programs, in *Proceedings of IFIP Congress 71*, North-Holland, Amsterdam, 250-255.
2. BÖHM, C. and JACOPINI, G. (1966). Flow Diagrams, Turing Machines and Languages with only Two Formation Rules, *Comm. ACM*, 9, 366-371.
3. GINSBURG, S. (1966). *Mathematical Theory of context-free Languages*, McGraw Hill, New York.

Proofs and reprints

Galley proofs will be sent to the author to ensure that the papers have been correctly set up in type and not for the addition of new material or amendment of texts. Excessive alterations may have to be disallowed or the cost charged against the author. Corrected galley proofs, together with the original typescript, must be returned to the editor within three days to minimize the risk of the author's contribution having to be held over to a later issue.

Fifty reprints of each article will be supplied free of charge. Additional copies may be purchased on a reprint order form which will accompany the proofs.

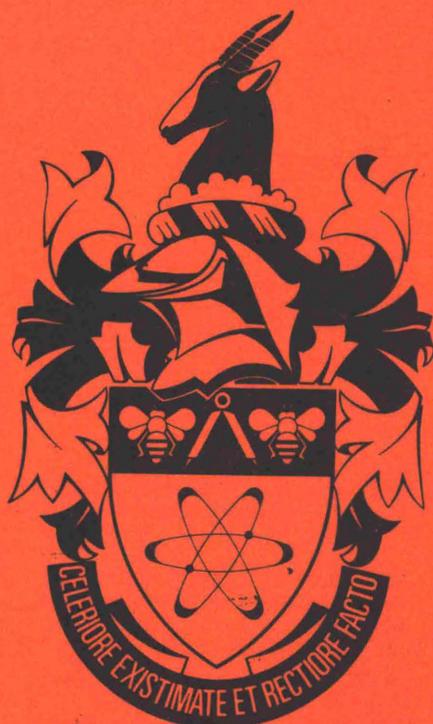
Only original papers will be accepted, and copyright in published papers will be vested in the publisher.

Letters

A section of "Letters to the Editor" (each limited to about 500 words) will provide a forum for discussion of recent problems.

Hierdie notas is ook in Afrikaans verkrygbaar.

Quaestiones Informaticae



Contents/Inhoud

Evaluating the Performance of Computer-Based Information Systems using a Restricted Linear Regression Model.....	1
P J S Bruwer	
An improved Implementation of Grimbleby's Algorithm*.....	7
S R Schach	
Nebulas as Structural Data Models*.....	11
H O van Rooyen and D J Weermans	
Data Structures for Generalized Network Algorithms*.....	15
D C Currin	
Modelling Blocking on Admission of Tasks to Computer Systems*.....	19
S Wulf	
Analytical Model of a Mixed-Workload MVS Computer System*.....	21
R J Mann	
Add: The Automated Database Design Tool*.....	25
S Berman	
A Disk Space Management System*.....	29
A J Cuthbertson, I J van Nierkerk, T Turton	

*Presented at the second South African Computer Symposium held on 28th and 29th October, 1981.