I/O Performance Modelling & Evaluation - giving your company the competitive edge

Mr. Mobin Waja (Edgars Stores Ltd)
Edgardale
P.O. Box 100
Crownmines, 2025
South Africa
(011) 495-7617

Mr. Raoul Blignaut (IBM South Africa)
70 Rivonia Road
Sandton, 2146
South Africa
(011) 320-8128

Abstract : I/O Subsystem performance is one of the key components that impact the business operations of today's organisations. The objective of this paper is to provide a detailed performance evaluation of various storage architectures and strategies that can be used to improve performance. The evaluation is based on modelled peak on-line database workload which provided initial response time estimates. The paper then examines the actual performance improvements that were achieved by moving the workload to devices that support the following architectures:

1. **a traditional single stage cache,**
2. **a managed single stage cache,**
3. **a two stage cache,**
4. **and a three stage cache architecture.**

The effects of multi-stage cache are investigated specifically in relation to cache hit ratios and the performance implications of a second level segmented cache. Cache management techniques are reviewed in relation to the performance improvement that was achieved. Finally the paper reviews the positioning and management of various performance, availability, enhanced functions and reliability options based on the business requirements of the organisation.

Keywords: I/O Subsystem, performance, business requirements, storage architectures, cache management.

Computer Science Category:

Note:

The authors wish to note that this paper does not attempt to describe technical details of any specific vendor's equipment or the principals of IBM MVS I/0 systems. Additional technical and costing input was taken into account as part of the evaluation, which is not considered part of this analysis. It should also be noted that although every effort has been made to maintain consistency across the workload comparisons there are a large number of variables that will effect the overall performance of the database and I/O subsystems. Thus Edgars specifically state that the results do not represent any formal type of performance guarantee.

1. Introduction

Edgars is a large clothing retail organisation with its head office and computer centre in Johannesburg South Africa. The organisation's primary business is the retail of upmarket clothing and cosmetics. A significant proportion of the business is derived from the use of the stores own charge card. The Credit Checking application for this charge card is critical to the operation of Edgars while other large database systems such as the General and Merchandising Systems form some of the core business systems. High transaction rates would often result in the transaction times increasing, ultimately resulting in longer delays at the point-of-sale within Edgars and Sales House (a division of Edgars) stores, especially over the December period. Analysis of the systems revealed that the major component of service time had been caused by the DASD I/O subsystem.

A project was initiated to provide a comprehensive and objective evaluation of various strategies that are currently available for improving the performance of the I/O subsystem. After several working sessions with Edgars, analysing the current systems, it was decided to conduct a detailed study to determine the benefits of implementing one or several of the available options. Multiple options were modelled and cost performance comparisons made to determine the which strategies would be tested using Edgar's live production workloads. Although this paper focuses on the effect of performance on the business operation, aspects such as integrity, data availability and reliability were included in the overall evaluation.

2. Current storage issues

One of the key issues facing Edgars was to substantially improve the overall performance of the production database systems in the most cost effective manner. Edgars had, subsequent to this study, gone through a number of exercises to improve the performance without additional hardware expenditure. Effective exploitation of the storage hierarchy is key in providing the highest returns for your business, and in addition, it is critical to manage the storage hierarchy in-line with the business requirements. Edgars stores currently exploit DFSMS, DFHSM and DFDSS to manage archive, backup and performance of non-database systems. It was in the area of I/O subsystem performance, especially the database systems, where it was proving difficult to gain substantial improvements without additional hardware expenditure. This prompted Edgars to investigate various hardware options available to the company in providing performance improvements.

3. The Base System

A stratified sample was drawn not to represent the general population (all I/O's and response times) but on obtaining what Edgars considered its peak on-line day. These periods were determined to occur between 10:00 am to 12:00 noon from Monday to Friday. A systematic sampling technique was then used to select the members on an hourly basis from this stratified sample. The base system was considered to be the sample taken for the week 18/09/94 to 23/09/94. From this point on, changes to the database systems that were to be used in the comparisons were kept to a minimum. Because it is impossible to ensure exact week on week workloads, we decided to consider a longer sample period as discussed above. This had the effect of smoothing most large variations which are characteristic of an on-line environment. The workload behind a single Logical Control Unit (LCU 19) was selected because it was the primary cause of the I/O subsystem bottleneck. This workload was used as the base input for all the models and then was subsequently moved behind the selected vendor's equipment to monitor and record the actual performance. The levels of software used in the evaluation were, MVS 4.2, DFP 3.3 and ADABAS 5.2.6. The following section gives some background to the database, applications, database performance. A more detailed analysis of the control unit workload and performance concludes this section.

3.1. The Database and Applications

Each of the Edgars database system is an ADABAS Inverted List system and the associated applications have been programmed in NATURAL. Each ADABAS database requires a minimum of three data sets during operation namely the Associator, the Data Storage and Work Data Sets. ADABAS controls and maintains data in physical blocks within Data Storage. The physical size of these blocks are determined by the physical storage device characteristics on which they reside. Each physical block is automatically assigned a Relative ADABAS Block Number (RABN).

3.1.1. Data Storage (DATA)

The data storage files are MVS Physical Sequential (PS) data sets that are allocated with a 10796 KB blocksize. Due to the random nature of the on-line days queries and updates, and the size of the data set, it typically has a poor Locality of Reference (LOR). This can be seen, for example, with data set DBAS.D042.DATA that typically has a total (read and write) hit ratio of between 38% and 45%, on the base system with 64 MB read and 16 MB write cache. The data storage files are typically large multi-volume data sets.

3.1.2 Associator Data Sets (ASSO)

All ADABAS control information about the database and the application and system files that it maintains is stored in the Associator. The three main areas in the ASSO are the General Control Block, Utility Communications Block and the Data Storage Space Table. The ASSO files provide the index (commonly called the Inverted List) and address converter into the DATA through the use of a RABN (Relative ADABAS Block Number). These files are also MVS PS files with their own internal structure that provides the association through the address converter needed to locate a record in Data Storage. The ASSO data sets typically have a very poor Locality of Reference with hit ratios varying between 25% and 36% for DBAS.DB042.ASSO. These files are also very large multi-volume data sets with a Physical Sequential (PS) format and a 4136 KB blocksize.

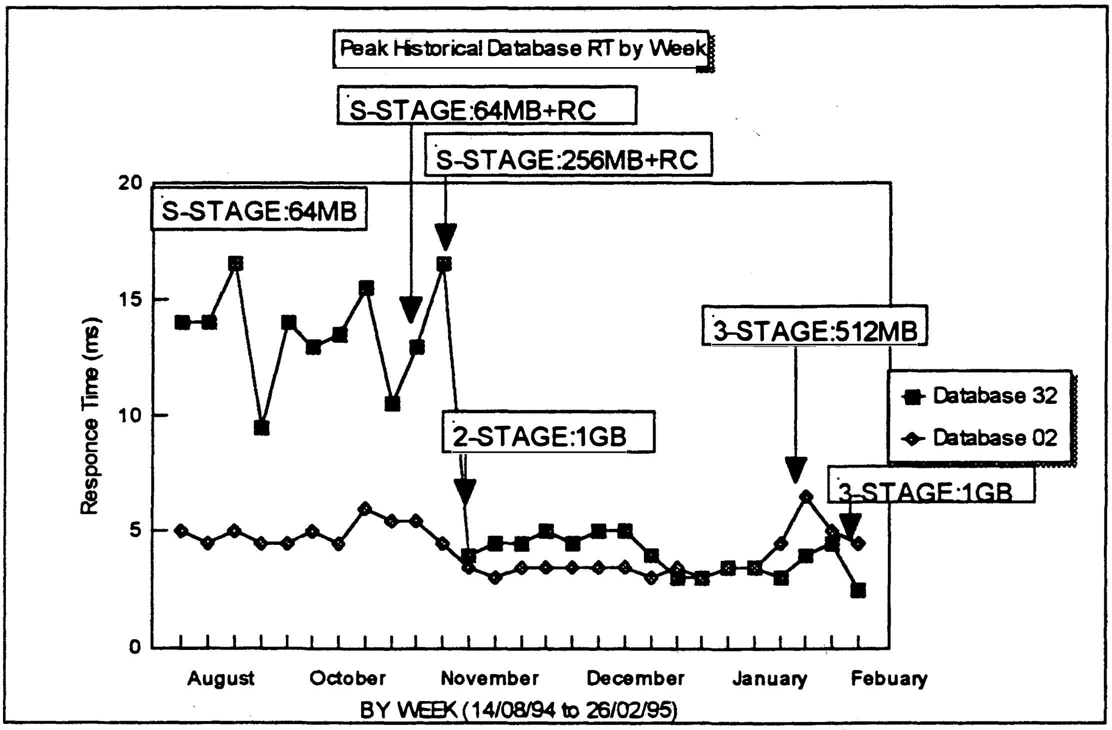
3.1.3. Work Data Sets (WORK)

The work data set is used by the Nucleus (ADABAS) as a temporary storage for processing large lists of records, sorting and error recovery and working area. The work data sets typically have a very good locality of reference with DBAS.DBA042.WORK obtaining between 98% and 100% hit ratios.
Additional data sets include the Protection Log (PLOGs) which contains all before and after images needed for ADABAS's maintenance of integrity and the Command Log that is used to perform audit trails and performance monitoring. The PLOG data sets also have a very poor LOR of between 21% and 27%. Both the work and logs have a larger blocksize of 13682 KB. The Database Administrators (DBAs) at Edgars typically decide on the physical placement of based on the hardware resources available and the possible contention between the different types of data sets. In the case of Edgars there were two main data base systems analysed namely DB02 which is the General and Merchandising Database and DB32 which is the Sales House Credit Database. These two data base systems in addition to three other volumes from, DB22, DB42 and DB62 respectively, were on the existing I/O subsystem.

3.2. Database Performance

The database performance was becoming a major problem with a command rate average for the peak on-line day, 10 am to 12 noon, of between 600 and 700 per second for DB02 with approximately 1600 and 1800 user query entries. Typically, DB32 had a command rate of between 400 and 500 per second with approximately

1100 and 1400 user query entries. Reviewing the ADABAS command response time for DB32 by week shows that it was over 16 ms. A very high variation between the peaks also resulted in erratic end-user response times, see Graph 1: Peak Historical Database R.T. by Week - DB32, August and October 1994.



Graph 1: Historical Response Times - database (batch and on-line)

NOTE: S-STAGE represents the Single Stage, +RC represents Read Record Cache, 2-STAGE the Two Stage and 3-STAGE the Three Stage architectures with the various cache sizes indicated.

The above graph shows the overall ADABAS command response time, 24 hour sample, by week. Analysis determined that the major component of the time was the wait on I/O. Database DB02 showed peak on-line response times of 6 ms. This database was more CPU bound (see Table 1: Database workload by pool). The DASD response time was typically over 30 ms with certain DB32 volumes having response times between 54 ms and 66 ms. See Appendix A: DASD activity report - (25/09/94). **It should be noted that a large number of variables influence the overall (batch and on-line) database response times and this graph is included to give a general indication of the data base performance.**

3.3. Control Unit Performance

The system that all the models were based was a Plug Compatible Manufactures (PCMs) 3990 type control unit with 56.7 Gigabyts of 3390 type DASD (5.25 inch). The controller had 64 MB of cache, 16 MB non-volatile storage (NVS) and was attached to an ES/9000, with four ESCON channels. This system is typical of the previous generation controllers and can be considered a single stage cache architecture (Houtekamer, 1994). Statistics were sampled for this controller and used as the base input to the model.

The base report was daytime DASD activity by logical control unit (LCU) obtained from CMF run on 25/09/94. Statistics were sampled for the week of the 18/09/94 to 24/09/94 for the peak two hours, 10 am to 12 noon, of the on-line day workload for LCU 19. Other samples were also statistically analysed to check any deviation from the base report. The total I/O rate for the system during this period was 793 I/O's per second with an average response time of 20.15 millisecond seconds (ms). The average I/O rate for the controller was 255 I/O's per second with an average response time of 34.56 ms. LCU 19 had twenty 3390 type volumes configured behind it. It is interesting to note is that the standard deviation for this volume response time sample was 16.3 indicating a large variance in performance. This typically would result in erratic database and thus erratic end-user response times. The ADABAS database systems on these volumes were as follows:

DB Pool	Total Volumes	LCU 19 Volumes	% Total CPU	% Total I/O
DB02	10	10	**15.44**	10.19
DB32	8	7	11.29	**19.85**
DB42	3	1	0.22	0.29
DB63	5	1	1.23	2.4
DB22	6	1	3.86	7.37

Table 1: Database Workload by Pool

This table shows that DB02 and DB32 constitute over 26% of the total CPU utilisation and 29% of the total I/O utilisation. DB32 is more I/O intensive while DB02 is more CPU bound. These database systems have additional volumes behind other control units that were not included in the study. It can be seen from this table that a large portion of the total I/O workload (32.1%) was directed to this control unit. The response times on average were over 71% worse than the system average even though LCU 19 had the largest cache memory of 64 MB read and 16 MB write cache. A 16.4 standard deviation was high showing a large variation in volume response times from the mean (see Graph 2).

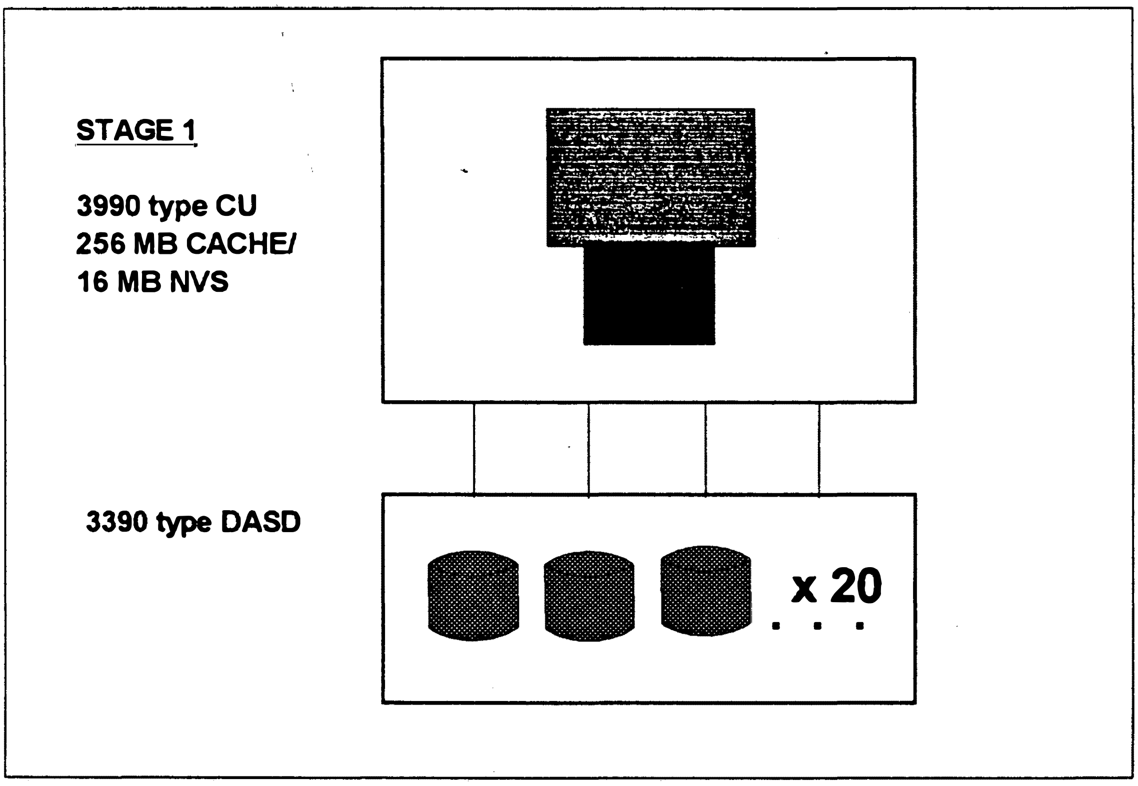


Diagram 1: 3990/3390 Type - traditional single stage cache architecture

Taking additional statistics obtained from Consul/SMS DSCAT Analysis Tool of the database pools showed that the typical cache hit ratios were between 35% and 49% with a read write ratio of 3.8 to 1. These hit ratios are typically considered poor due to the additional staging overhead caused by a read or write miss.
The CMF report, see Appendix A, shows that the Disconnect time was very high (16.4 ms) indicating that the for a large number of I/O's the physical disk had to be accessed resulting in seek, latency and rotational positioning sensing (RPS) delay. The second largest component was IOS Queue time (9.9 ms) which is typically caused by a device busy condition. Device busy is in turn caused by data transfer (Connect time) and staging the track (partial) into cache (Houtekamer, 1993). With the average connect time of 2.6 ms (approximately a 8 KB transfer) the stage time (t stage) would be:

$$t_{stage} = (t_{rotation} - t_{transfer})/ 2$$

With the average rotation been 16 ms, the stage time would be between 6 and 7 ms (or approximately an additional 28 KB transferred). The result is divided by two because an average, half a track, excluding inter-record gaps is staged on a read or write miss condition. This 'phantom' device busy results in considerable additional overhead (device busy) for very little additional benefit when the cache hit ratios are low. Two key considerations were to improve the hit rate and / or to reduce the staging overhead. The poor hit ratio is a direct result of the randomness of the database access and the average holding time in cache. Because the database access was determined by user usage patterns this variable could not be changed. Examining the subsystem shows that at 255 I/O's per second and a hit rate of 40% the average holding time, t hold, would be 13 seconds. Note that this was calculated for read cache only.

$$t_{hold} = ((\text{number cache slots})/2) / (1 - \text{hit ratio} * \text{I/O rate})$$

The number of cache slots for 64 MB cache is 4096 with cache segmented into 16 KB slots and an average of two slots used per I/O (partial track staging). On the basis of statistics from Mc Nutt's CMG 1994 paper, A Survey of MVS Cache Locality by Data Pool, it was determined that the optimal holding time for an on-line database workload would be between 90 and 120 seconds. A number of different strategies were investigated to improve the holding time. These included:

1. **Reducing the I/O rate** - by reducing the I/O rate into cache would improve the overall holding time and reduce the additional staging caused by cache miss operations. This could be done by turning cache off at a volume level but this would also effect the performance of data sets that were benefiting from cache. Advanced caching algorithms, which would dynamically turn cache on or off, (Dynamic Cache Management Enhanced, DCME), based on data set hit ratio through IBM DFSMS software was also evaluated (Houtekamer, 1993). Converting the data to DFSMS was planned for the medium term and thus was discarded for the purpose of this exercise.
2. **Additional cache** - by increasing the cache size the average holding time would increase proportionately. Fixing the holding time at 120 seconds and solving for cache size it was estimated that a cache size of approximately 590 MB would be the most effective. The modelling tool was primarily used to determine the estimated response times based on different cache sizes.
3. **Record level caching** - this option would improve the effectiveness of cache by only caching the block (10 KB for ADABAS Data Storage) and reduce the phantom device busy for data sets that would benefit from this algorithm. At the time of writing, the PCM's 3990 type control unit could implement this function (for reads). Section 5.1 investigates the benefits obtained by dynamically managing the existing cache (64 MB) through dynamic read record level cache on the existing equipment.

Option two (2) was considered the only way to obtain substantial I/O subsystem throughput benefits in the short term. The other strategies would be combined, at a later stage, with the benefits obtained by increasing the cache. Read record level cache on the 3990 type control unit was activated as this was available and considered a viable short term option.

4. Performance Modelling

Key to the success of the project was to obtain accurate modelled statistics that would enable Edgars to select which, of the many options available, hardware equipment to install and evaluate. The main variable against which the performance improvement was measured was the overall cost of the solution. Cost included the overall cost of purchase and maintenance. The value of continuous availability provided by total redundancy and RAID architecture was also taken into account. **This paper does not investigate the cost side of the equation due to the transient nature of this variable.** IBM's objective was to provide consultancy services to assist Edgars in making long term DASD purchasing decisions based on the requirements for performance, data availability and the exploitation of current technology. The exercise was also used to calibrate and test the accuracy of the modelling tools so as to provide accurate input into future storage purchasing decisions.

4.1. The Modelling Tool

The tool, DASD Magic/2 is an I/O subsystem modelling package, that provides performance and cache hit estimates based on manually entered performance data. The purpose of the tool is to quickly provide performance predictions at a control unit level. It is a PC based application running under IBM OS/2 2.1 or higher. DASD Magic/2 was developed and is a product of Consul Risk Management B.V., The Netherlands and is currently licensed to IBM for marketing support purposes. The tool provides a high level (control unit level) analysis of various control unit types, cache size variations, DASD units and workload types. The tool accepts IBM Resource Management Facility (RMF) device type and caching statistics obtained from IBM Cache RMF Reporter (CRR) as input. As CRR data was not available, Consul / SMS DSCAT reports were used to obtain an approximation of the cache hit ratios. The modelling tool can also project increasing workloads on any of the modelled options. The model was used primarily to obtain the estimated response time at 255 I/O's per second (base input) for the various hardware configurations modelled. Selected models were also modelled under increasing workloads (I/O rates) to examine the degradation in performance with increased growth.

4.2. Modelled Results

Numerous models were run of which five were selected by Edgars to test based on a cost performance evaluation. Three major types, single stage, two stage and three stage cache architectures were represented.

1. The traditional **single stage** cache was represented by the installed 3990/3390 type devices with 64 MB cache and 16 MB NVS. There were two variations of the single stage architecture that were evaluated. This includes a managed single stage cache, that was represented by implementing an engineering change on the installed 3990/3390 type devices (cache and NVS sizes kept the same) and increasing the cache size to 256 MB on the single stage architecture. See Diagram 1.
2. The **two stage** cache architecture was represented by an Integrated Cached Disk Array implementation using a large single cache (1024 MB read and write) with Small Computer Synchronous Interface (SCSI) 5.25 inch disks and associated disk buffers (Houtekamer, 1994). See Diagram 2.
3. The **three stage** cache architecture was represented by a 3990-6 control unit with Redundant Array of Independent Disks (or RAID) disk implementation. This uses a control unit cache (1024 MB) and NVS (16 MB), a second level drawer cache (32 MB * 10 drawers) and SCSI 3.5 inch disks and associated disk buffers. There was one variation to this that modelled and tested a 512 MB control unit cache. See Diagram 3.

Table 2 shows results obtained from the modelling process for the five selected options (options 1 and 2 are single stage, 3 is a two stage and 4 and 5 three stage architectures) that were selected for live production workload performance comparisons. It should be noted that Edgars did not base its decision to test these option solely on the results of the modelling process but solicited various vendors for pricing and technical input.

	CU & DASD Type	Cache Size	Modelled RT. *Optimistic*	*Realistic*	*Pessimistic*	Notes
	3990 & 3390-3 type	64 MB	29.4 - 34.6			Actual
1	3990 & 3390-3 type	64 MB	-	-	-	Note 1
2	3990 & 3390-3 type	256 MB	14.8	17.2	18.8	
3	Integrated Cache Disk Array	1024 MB	8	12	15.1	
4	3990-6 & RAID	512 MB	9.2	15.1	20	
5	3990-6 & RAID	1024 MB	6.1	10.9	15.3	

Table 2: Modelled Performance Estimates

Note 1. The model could not separately model PCM's 3990/3390 type adaptive read record cache.

Because any modelling process is based on a number of assumptions, three runs were performed on each option to obtain an optimistic (best case), realistic (considered most likely) and a pessimistic (or worst case). These results are compared to the actual performance obtained later in this paper.

5. Workload Comparison Results

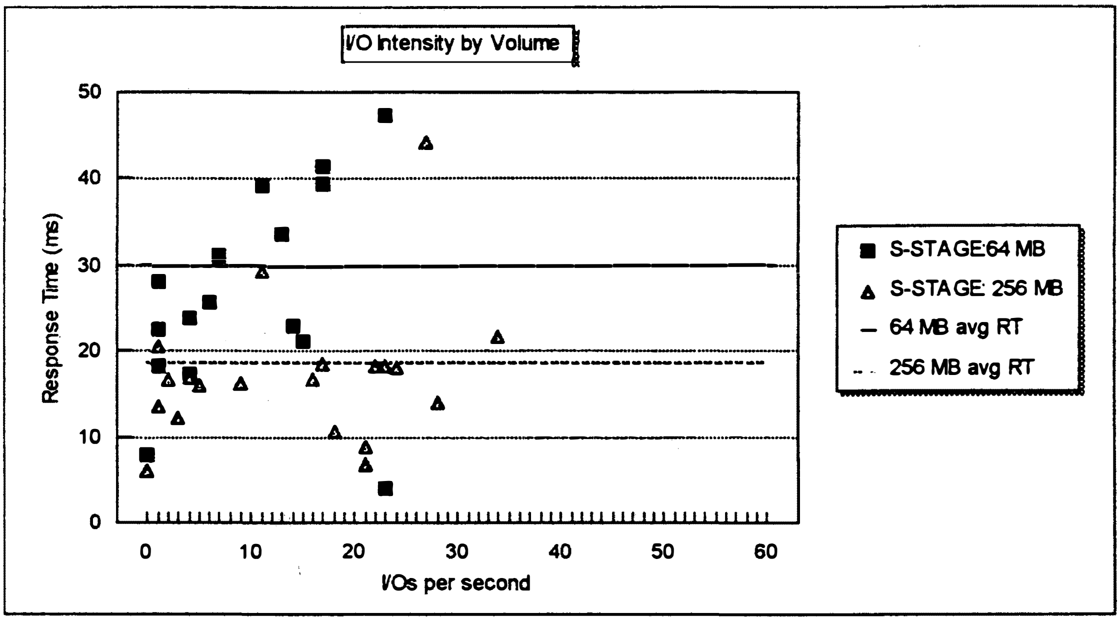
The vendors supplying the different types of architectures were requested to install their equipment over the period from September 1994 to February 1995. The base system, the 3990 type control unit and 3390-3 DASD was previously installed thus required engineering (adaptive record level cache) and upgrade (64 MB to 256 MB cache) changes to the existing control unit. An Integrated Cached Disk Array storage director and disk was installed, with 1024 MB cache in October 1994 to represent the two stage cache architecture. Finally a 3990-6 control unit RAID DASD was installed in January 1995 with 1024 MB cache and an engineering change set it at 512 MB. See Graph 1 for details for which periods the various architectures were installed. It should be noted that after the week ending 19/02/59, the comparisons were considered complete and additional workloads were balanced across all the installed control units.

5.1. Managed Single Stage Cache Architecture

This option required an engineering change (microcode update) to allow it to be activated on the 3990 type control unit. This was performed, and CMF reports were run on 23/10/94 and statistics sampled for the week 16/10/94 to 21/10/94 for the peak on-line day. This system had the original installed cache of 64 MB. The sample showed the average I/O rate of 294 I/O's per second with an average response time of 31 ms. Although this workload had on average 39 more I/O's per second it was still considered a representative sample. This fundamentally shows that by managing the cache resource more efficiently has resulted in a 10.4% performance improvement without any additional capital expenditure. The biggest improvement was obtained by a reduction in IOSQ time that is primarily caused by excessive staging (partial track) with low cache hit ratios.

5.2. Managed Single Stage Cache Architecture (larger cache)

This option required an upgrade to the base control unit from 64 MB to 256 MB cache. The sampled statistics were collected for the period 23/10/94 to 27/10/94. For this sample the average I/O rate was 288 per second with an average response time of 18.8 ms. See Graph 1. This represented a 45.6 % improvement in performance over the base sample. This improvement was the combined performance improvement of both cache management and larger cache size. The primary reduction was the IOSQ and Disconnect time resulting from reduced device busy and improved cache hit ratios. Although the average DASD response had improved substantially it was not sufficient to have any marked effect on the database response times. Graph 1 shows large variations in performance on DB32.



Graph 2: Workload Comparison - single stage architecture

The standard deviation of 8.9, although still high, was down from the base of 16.7. This showed that a larger managed cache also reduced the variation in volume response times thus improving the service level to the end-user. There was no modelled data for this option to be compared to, however it provides estimates of what performance benefits advanced cache management techniques deliver.

5.3. Two Stage Cache Architecture

The two stage architecture was represented by an Integrated Cached Disk Array storage director and associated disk system. The two stage cache architecture comprises of control unit cache and SCSI disk level buffers. See Diagram 2. The stratified sample was taken for the period 23/01/95 to 27/01/95. The storage director was configured with 1024 MB cache (non-volatile read and write) with 56.75 GB of SCSI 5.25 inch disk storage.

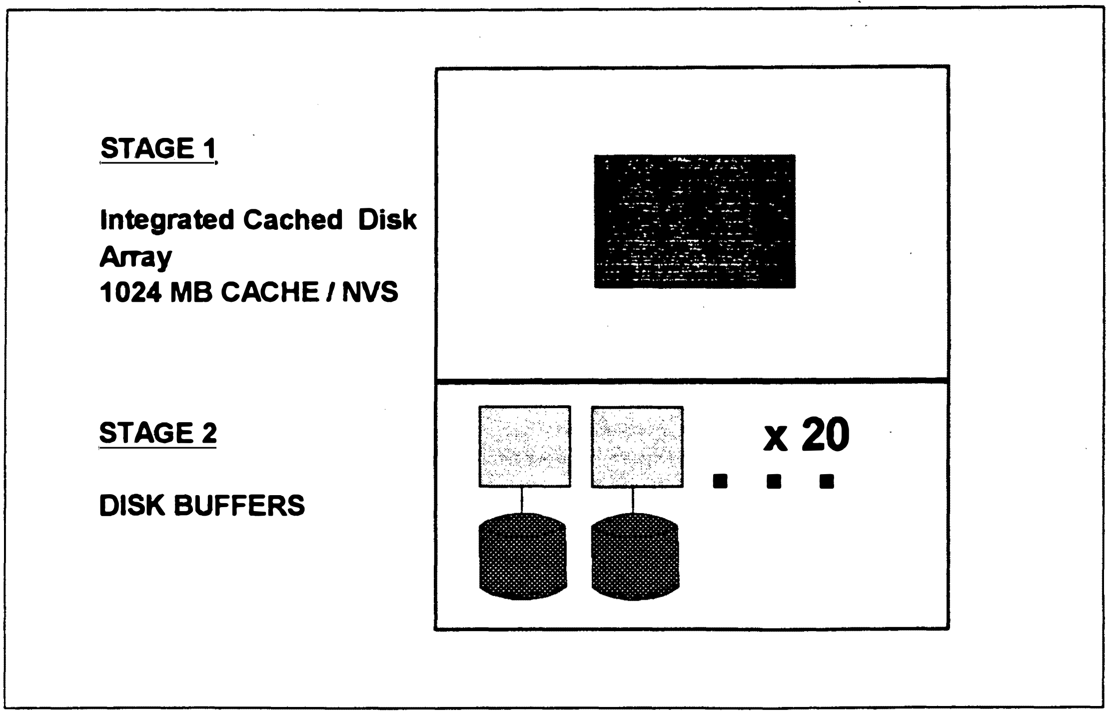
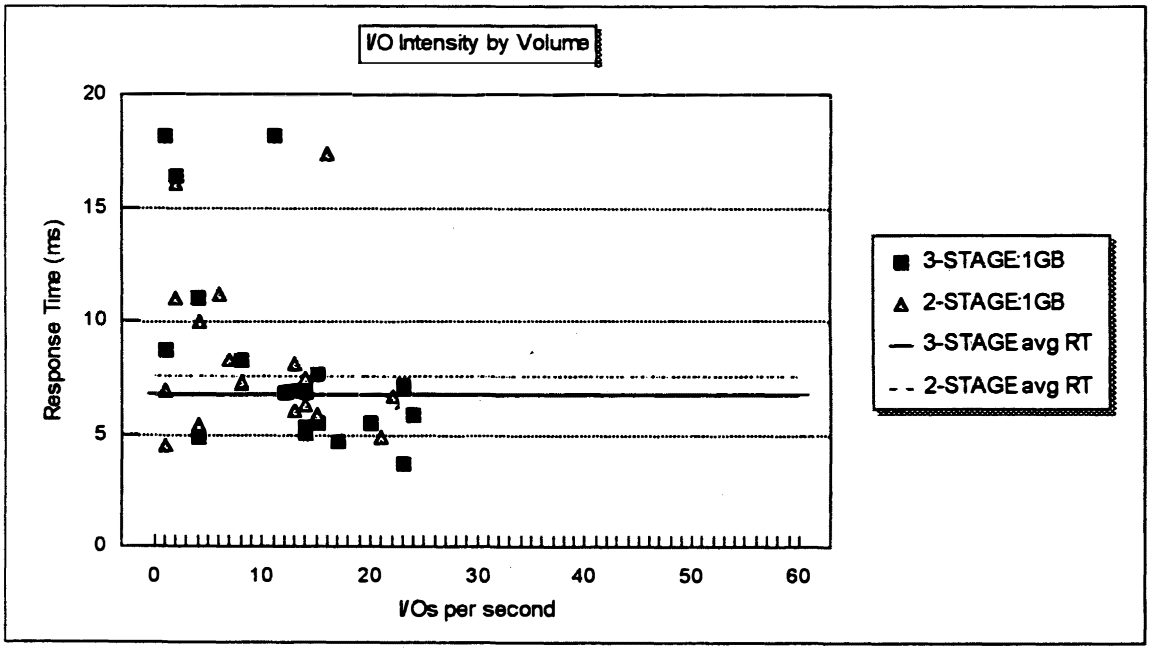


Diagram 2: Integrated Cached Disk Array - two stage cache architecture

The sampled statistics were collected for the period 23/01/95 to 27/01/95. The average I/O rate for the period was 207 per second with an average response time of 7.6 ms. This 78% improvement reduced the ADABAS Command Response from peaks of 16 ms down to 5 ms for DB32 and eliminated the high variation in response that was previously experienced. See Graph 1.



Graph 3: Workload Comparison - two and three stage cache architectures

The graph shows the I/O rate and associated response time for every volume sampled and thus shows the spatial distribution of the I/O intensity. There was a very high correlation, 0.92, between the two and the three stage architectures with a 86% associated confidence. This indicates that the workload between the two periods was very similar and that similar cache sizes produce comparable results. A 3.4 standard deviation was the lowest obtained for any of the comparisons showing a large reduction in the variation in volume response times from the base. Interesting to note is the large improvement in the database response times on DB32 which is I/O bound, compared to DB02 which is CPU bound. See Graph 1.

5.4. Three Stage Cache Architecture

The final workload included two benchmarks run on different cache sizes on a 3990- 6 control unit with 56.75 GB of RAID DASD attached. The 512 MB cache size was implemented to test the effect of diminishing returns when implementing larger cache sizes (Gray, 1990). The data sampled for 512 MB cache comparison was for the period 29/01/95 to 03/02/95. The average I/O rate was 212 per second with an average response time of 10.4 ms which represents a 69.3% improvement over the base configuration.

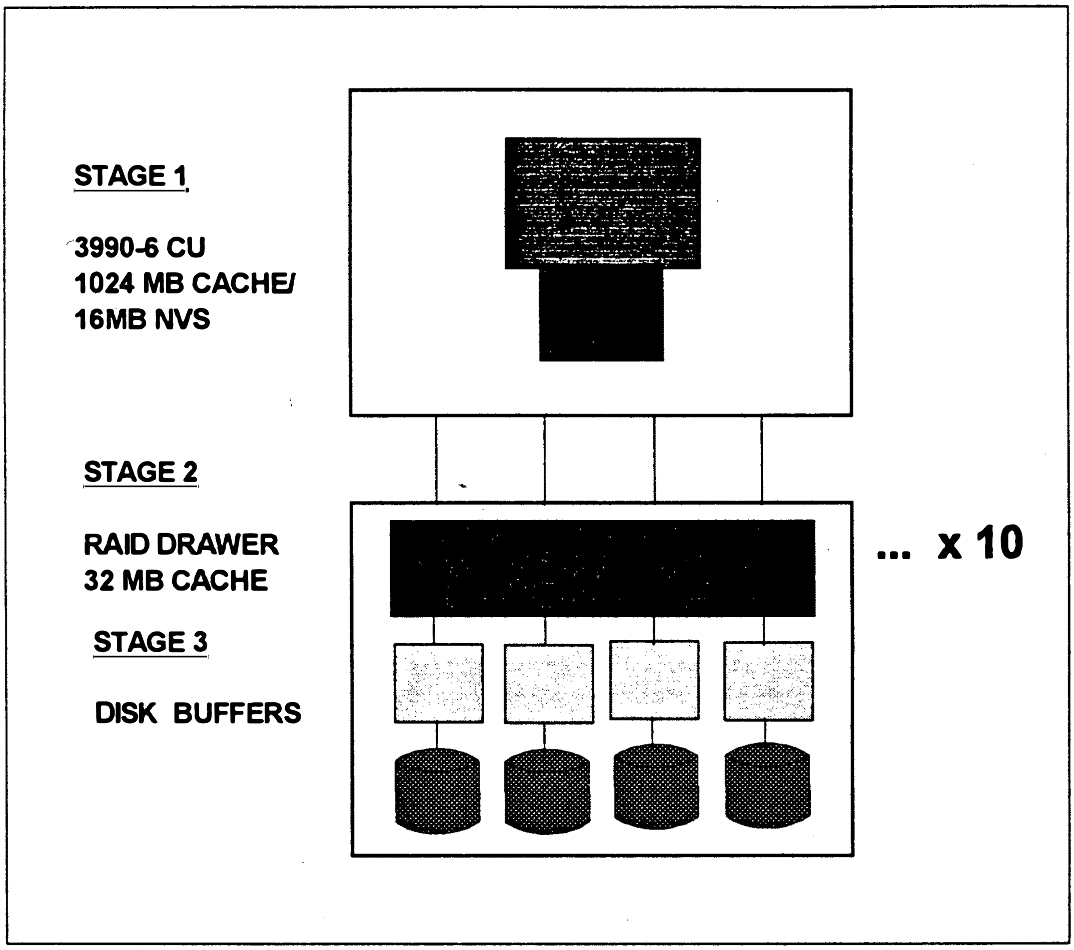
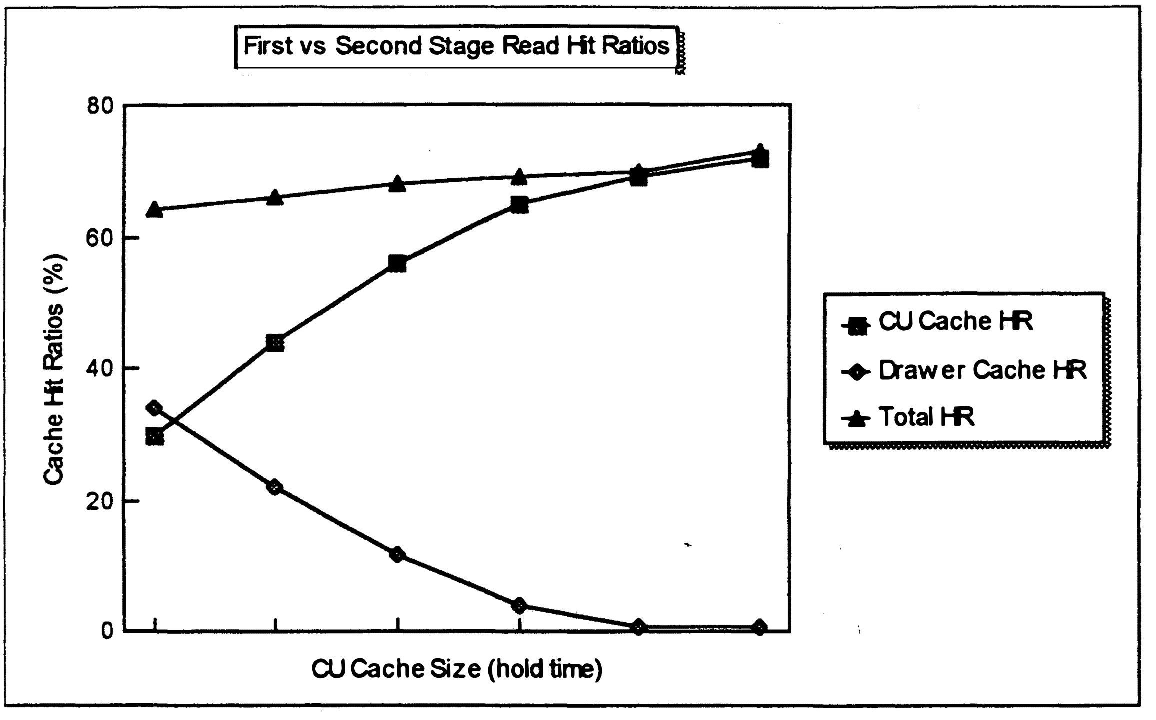


Diagram 3: 3990-6 and RAID DASD - three stage cache architecture

The final sample was taken from 13/02/95 to 17/02/95. The average I/O rate was 239 per second with an average response time of 6.8 ms. This represents an 83% improvement from the base configuration (64 MB cache) to 1024 MB cache. Any additional cache above 1024 MB would achieve very little benefit for this workload, clearly showing the effect of diminishing returns on larger cache sizes . A 4.4 standard deviation indicates a low variation in volume response times.

The second level caches provided additional buffering for reads and writes. DB02 did not seem to be as effective on the three stage as on the two stage cache architecture possibly because the workload was more CPU bound. Analysing control unit and drawer hit ratios, shows that the benefit of the second level drawer cache diminishes as the hit ratios for the control unit (first stage) are increase. Modelled hit ratios show that the drawer cache provides a secondary, segmented buffer to the first stage cache especially when there is contention within the first level of cache. See Graph 4 below.

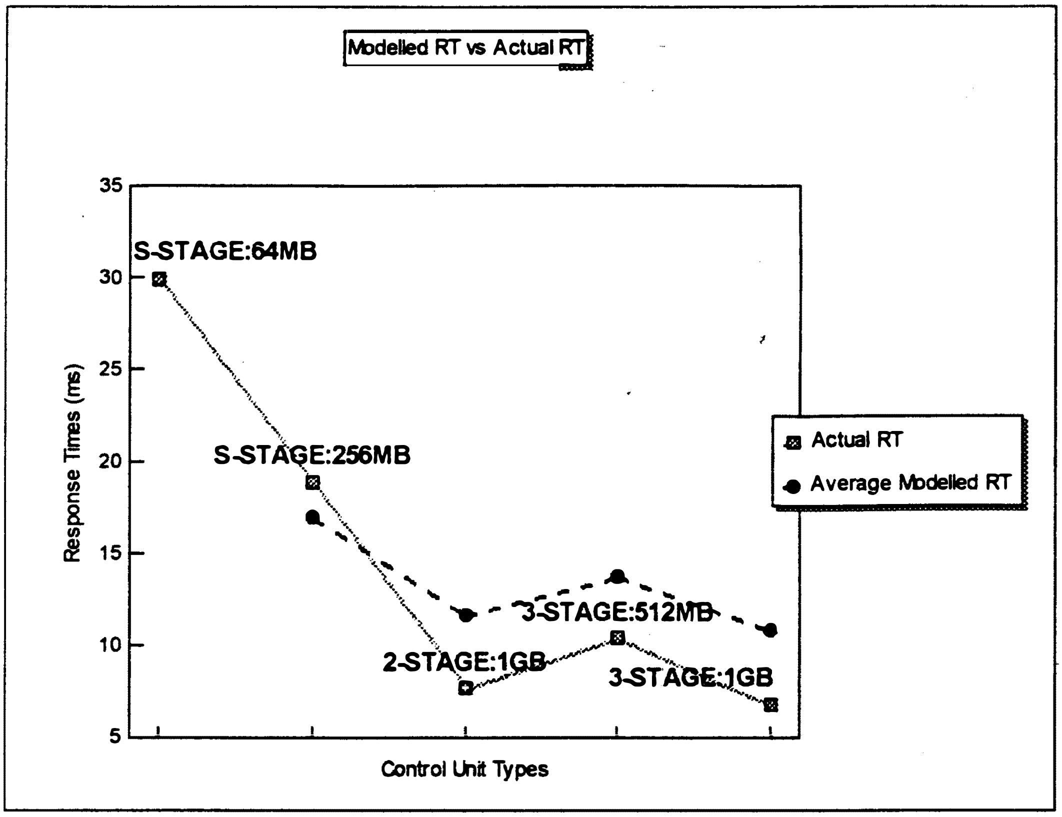


Graph 4: Effects of Second Level Cache

This has the effect of improving the overall hit ratios and thus response times, resulting in a more even performance distribution for varying first stage cache hit rates. The three stage cache architecture would thus be able to sustain higher I/O workloads with less effect on the overall performance.

7. Modelled vs Actual

The following graph, Graph 5, summarises the actual response times with the average of the three (optimistic, realistic and pessimistic) modelled runs. It should be noted that due to the small sample selected from the entire population there is likely to be bias in the sample. To reduce bias because of the small sample the sample standard deviation and variation (n-1) were used. The closest modelled result was accurate within 0.3 ms of the actual workload . With the change in controller type and implementing larger cache sizes, the models tended to err on the conservative with the *optimistic* models been more accurate. With implementing vary large cache sizes the workload tended to become more responsive to cache from what was originally anticipated. See Graph 5 below.



Graph 5 : Summary of Results (actual vs modelled)

There is a very high correlation of 0.98, between the two samples indicating a high level of consistency when between the various control unit modelled. Due to the small sample size available the confidence of this value is only 47%. The graph shows that typically the response times that were achieved were better that the modelled results.

Control Unit	Actual	M-Mean	M- Variance	M- Deviation	M- Range
S-Stage-256MB	**18.8**	16.93	2.7	2	4
2-Stage - 1GB	**7.7**	11.7	8.5	3.6	7.1
3-Stage-512MB	**10.4**	13.68	19.5	5.4	10.8
3-Stage-1GB	**6.8**	10.7	14.1	5.4	9.2

Table 3: Basic statistical comparison (modelled and actual)

As the table shows the most accurate results were obtained by increasing the cache size, one change, on the existing control unit with a variance of 2.7 and standard deviation of 2. The largest variations and deviations were obtained when modelling a number of changes for example, moving the workload from a traditional single stage architecture to a more complex three stage architecture. This showed a variance of 14.1 and a standard deviation of 5.4. When reviewing the modelled results, it shows that the *optimistic* results were closest to the absolute values obtained, compare Table 2 (optimistic) and Table 3 (actual). This seems to indicate that the workload became more responsive to the increases in cache than was originally expected.

6. Additional Considerations

The objective of this paper was to document the performance variations between modelled and measured workload comparisons for various cache architecture implementations. The model proved to be sufficiently accurate in estimating the effects of cache size and cache architecture changes. The modelled results were typically more conservative when adding large cache size increments. The overall evaluation was based on a number of different criteria including performance, availability, enhanced functions and reliability, based on the business requirements of the organisation. Certain applications have different requirements and respond differently to cache size and cache architecture changes (compare DB32 and DB02). Thus it is key to evaluate the options based on your applications and business requirements.

7. Acknowledgements

We would like to acknowledge Mr. Y. Ranchod (Edgars), Mr. D. Kennedy (Edgars), Mr. A. Mason (Edgars) and Mr J. Jagers (IBM S.A.) for their input and assistance.

Appendix A

Following is a table showing the base sample taken from 18/09/94 to 23/09/94, peak on-line day (10 am to 12 noon), by volume.

3990 TYPE (64mb cache/ 16 MB NVS)						
3390 TYPE DASD						
Dev	IO Rate	Resp	Conn	Disc	IOSQ	Pend
A40	23.06	4.1	1.8	1.2	0.4	0.7
A41	14.96	21.3	2.2	14.1	4.1	0.9
A42	0.08	7.9	0.9	5.2	0	1.8
A43	14.27	22.9	2.9	13.5	5.6	0.8
A44	3.86	23.9	3.1	18.3	1.5	0.9
A45	7.27	31.2	3.3	22.1	5	0.9
A46	0.93	22.6	3.4	18	0.2	0.9
A47	16.8	41.4	2	21.7	16.6	1.1
A48	13.21	33.6	2.2	20.1	10.3	1
A49	6.33	25.8	3.4	19.4	2.2	0.9
A4A	1.03	18.2	2.7	15.4	0.2	0.9
A4B	4.39	7.4	2.7	12.8	1.1	0.8
A4C	11.22	39.1	3.6	24.8	9.7	0.9
A4D	21.78	66	3.2	21.1	40.6	1.1
A4E	27.2	54.9	1.8	17.9	34.1	1.1
A4F	22.95	45.1	1.9	18.4	23.6	1.1
A50	24.5	7.2	2	3.4	1.1	0.8
A51	17.7	39.4	2.5	19	16.9	1
A52	23.25	47.3	2.4	18.6	25.2	1.1
A53	0.91	28.1	3.7	23.3	0.4	0.8
TOTAL	255.7	29.37	2.59	16.42	9.94	0.98

References

Artis, A. Pat, **Record Level Caching - Theory and Practice**, CMG Proceedings, Orlando, 1994, late breaking paper.

Consul, **Consul / SMS DSCAT 1.1 Users Reference Manual**, Consul Risk Management B.V., The Netherlands, 1991.

Gray, W.T., **The Caching Process In The Storage Hierarchy**, CMG Proceedings, Orlando, 1990, pp 38-49.

Houtekamer, G. E. & Artis, A. P, **MVS I/O Subsystems - Configuration Management and Performance Analysis**, McGraw-Hill Inc., New York, 1993.

Houtekamer, G.E., **DASD Magic/2 - Help Text**, Consul Risk Management B.V., The Netherlands, 1994.

Mc Nutt, B., **A Survey of MVS Cache Locality by Data Pool: the Multiple workload Approach Revisited**, CMG Proceedings, Orlando, 1994, pp 635-643.

WH&O International, **Inside ADABAS**, Wellesley Massachusetts, USA, 1992.