IBM SE TECHNICAL INFORMATION EXCHANGE (TIE) PAPER

"Survey of System Performance Levels and Resident Option List Needs in OS User Installations

Doug Glading, Birmingham FSC, c. 1971

Abstract:

This paper summarises various figures from up to 20 performance studies during 1969-1971 using the AMAP tool.

Most of the studies were of different live customer installations covering models 40, 50 and 65, almost all running under MFT.

The system performance levels are presented to show how a sample of typical users are actually utilizing the CPU and disks. The difference between actual use and system capability should be noted. It emphasises the potential, and the need, to educate and to increase customer satisfaction with IBM.

OS System Dataset referencing accounts for much of the disk accessing. Actual figures live situations are again given to illustrate this and to indicate the need good Resident Option lists.

Summaries of the loading of Access Method modules, transient SVC call-lib members show that the supplied 'standard lists' are in many circumstances far from ideal. The figures will help the reader to select good basic lists for systems without unlimited core.

Introduction

The intention of this paper is to pass on some of. the experience gained in .various customer performance studies. During the course of these studies many similarities have been noticed across the different installations analysed. This paper should give practical help, and food for thought, to SE's dealing with the multitude of medium sized OS MFT installations.

There is often a large gap between the potential performance of a customer system and the actual performance being achieved. The figures in this report illustrate that this is a very common situation even in quite mature installations. Hopefully the practical points and discussions in this paper reduce the gaps that may exist in many other installations.

The author has drawn on some 20 of his performance studies With the AMAP tool. The studies have covered about 13 different customer systems plus a few personal investigation runs of different subjects. These runs have been under OS Releases 16 to 19 on Models 40 and 50 (with 1 65), all except one run being under MFT. Six of the more recent customer studies are analysed in more detail in the various Sections of this paper. Obviously all the installations remain anonymous, but they are from various industries from insurance to process, although no university or purely scientific-type job shop has been studied.

Since AMAP was the tool used, the figures result from runs, lasting up to an hour, of "Representative Jobstreams". As is well known there is usually no such thing as a truly representative jobstream. However even if individual studies are open to some questioning on this point it is sensible to argue that over all the studies together some reasonably typical work was run. In fact nearly all the studies probably ran the AMAP-traces work noticeably more efficiently (e.g. less blow-ups', idle partitions, mounting delays, operation decisions etc) than day-to-day work.

Thus actual performance levels are very likely to be lower than those present here. Therefore there is even more scope for performance studies to ensure that Proposed systems do do the proposed workload when installed, and that installed customer satisfaction with IBM is high.

Survey of System Performance Levels and Resident Option List Needs in OS User Installations

CONTENTS

TOTAL DISK I/O RATES

STUDY	SYSTEM	No.2314	No.2311	I/O's Per Sec	PERCENT TO INDIVIDUAL DRIVE(s)
A	256K 40	8		21	53% AND 37%
P	256K 40		5	20	59%
G	256K 40	8		21	42%
L	256K 40	8		17	41%, 22%
D	256K 50	8		21	
R	256K 50		5	17	36½%
N1	512K 50	8		24	
N2	512K 50 HASP	8		10	
B1	512K 50	16		21½	53½% AND 30½%*
T	512K 50	8		21	75% TO TWO
X	512K 50	6 FROM 2 BANKS	3	35**	43½% AND 23½%*
B2	512K 65 + LCS	13		33½	26% AND 22%

NOTES

* NOTE: Despite mutliple channels these high usage drives were on the same channel.

** Inflated by the Linkub small ~~period~~ record problem (UK SETH 046). Corrected Figure would be 31½.

i.e. 'DC' in fixes

OS DATASET USE

FIGURES ARE

A. PERCENT OF ALL DISK I/O's

B. AVERAGE I/O TIME EXCP TO POST, IN MILLISECS PER RECORD (WITH FCI FETCH ACCESSES TO LINKLIB ARE ABOUT HALF THE NUMBER OF RECORDS)

MOST STUDIES HAD NO RESIDENT OPTIONS.

STUDY	SVCLIB		JOBQUEUE		LINKLIB		TOTAL OF A's
	A	B	A	B	A	B	
B1	39%	75.6	18%	76.3	10%	52.3	67%
D	29%	44.7	21%	-	6.5%	48.9	56½%
A	17%	71.6	9%	93.6	13%	95.5	39%
P	18%	66.5	10%	-	12%	118.6	40%
R	27%	69.6	12%	62.5	11¼%	69.4	50½%
B2	18%	47.5	9%	33.7	6%	53.1	33%
G	12%	42.5	9%	62.0	16%	40.0	37%
L	21½%	72.6	12.8%	69.7	*20.3%	62.2	54½%
X	15%**	55.6	9%	68.2	*18½%	93.5	42½%

* Figures subject to the small record problem described in UK SETH 046.

** With full resident RAM list.

CPU USAGE

STUDY	SYSTEM	WORK RUN	CPU UTILIZATION WHEN JOBS RUNNING	CPU TIME PER I/O (milliseconds)
A	256K/40	SORT, COBOL E, RPG, ISAM	65%	26½
P	256K/40 (2311)	PL/1, FORTRAN COMPILES SORT, PNS	66%	29½
G	256K/40	SORTS, PL/1 COMPILES, ISAM JOB	75%	30½
L	256K/40	PL/1 TEST AND PRODUCTION	78%	40
D	256K/50	ASS AND COBOL F, TEST WORK	63%	21½
S	256K/50	ISAM INFORMATION RETRIEVAL JOB	35½%	13
R	256K/50	COBOL E TEST, TAPE PRODUCTION JOBS	50%	17½
N1	512K/50	COBOL E TESTS, FORTRAN JOBS	78%	20
N2	512K/50 (HASP)	COBOL E PAYROLLS	61½%	28½
B1	512K/50	PRODUCTION AND TESTS, ISAM	67½%	24
T	512K/50	PL/1 COMPILES, TESTS	53%	17
X	512K/50	TAPE PRODUCTION, COBOL E TESTS	66½%	13½
B2	512K/65 (LCS)	PRODUCTION, FORTRAN TESTS, CPS	65%	15

NOTE - IN MOST CASES THE DAY-TO-DAY CPU UTILIZATION WILL BE NOTICEABLY LESS THAN FOR THE AMAP RUNS.

- READING AND WRITING FUNCTIONS ACCOUNTED FOR A HIGH PROPORTION (UP TO 40%) OF THE CPU TIME USED.

SECTION H LINKLIB and BLDL

H1 The first observation to make is that the IBM supplied BLDL list is completely useless in a normal user environment. Whilst the supplied RAM and RSVC lists are not unreasonable (though by no means perfect) as initial untailored lists, the supplied BLDL list refers only to assorted utility programs instead of to system modules that are used ten or a hundred times more frequently.

H2 It is being suggested to the producers of the relevant SRL that either there should be a warning that the standard BLDL list (in contrast to the RAM and RSVC lists) should not be treated as a usable list, or that there should be a more practical standard BLDL list. In fact two versions would be required, one for MVT and one for MFT, as there are distinct differences in the LINKLIB members accessed.

H3 A BLDL entry eliminates references to the directory of LINKLIB. As the entries are small (40 bytes in MFT) they can be used to pad out * the nucleus (re fixed area of core more precisely) to a 2K boundary. On performance grounds alone a list covering, say, 15 members is usually justifiable.

H4 The greatest difficulty in deciding on a BLDL list is due to many of the key members of LINKLIB having, for some obscure reason, numerous ALIAS names. If an alias name is used to refer to a member then that alias is what should be in the BLDL list. Aliases and proper member names are used in different proportions in different members. Some names are barely used and so would be wasted as entries in the BLDL list.

continued ...

H5 Tools such as AMAP give the references to LINKLIB members as references to physical records on LINKLIB. On systems with PCI Fetch (ie Model 50 up) the AMAP LINKLIB summary report can initially appear a little confusing. This is because some of the physical records that make up a member can be brought in by one access. Systems without PCI Fetch give an accurate access count for each physical record. In either case the number of uses of a particular member can be seen, but there is no indication of what actual names were used to cause the member to be loaded into core.

H6 There is at least one measurement tool which will record the name used to request that a member be fetched. This is called CATCH (available to World Trade as AIDS-WT-07R, or -08R for MVT). Unfortunately at the time of writing this is restricted to IBM Internal use only. However some private runs of CATCH have lead to the table 'LINKLIB MEMBERS and ALIASES'. This shows roughly the proportion of references made by each name and alias for the more popular MFT LINKLIB members. The table could certainly be refined if more data was available. There is definitely a need for more information on the frequency of use of aliases.

DG/jac

KLIB MEMBER USAGE

	MEMBER	APPROX BYTES	X	L	B_2	R	B_1	D	Totals	Refs per minute	Refs per min per kilobyte per BLDL entry
			NO OF REFERENCES IN EACH STUDY								
	DIRECTORY		1,137	1,070	746	1,003	1,309	1,684	6,949	21.38	
1	IEESD 563	1,700	167	187	53	1(?)	75	353	836	2.6	65.8
2	IEESD 564	1,500	155	172	52	0(?)	64	336	779	2.4	61.4
3	IEFSD 510	39,000	49	50	47	52	99	85	382	1.2	30.1
4	IEFSD 512	7,000	86	34	52	35	90	68	365	1.1	28.8
5	IEFSD 526	33,500	45	44	35	37	97	74	332	1.0	26.2
6	IEFW21SD	33,000	46	34	35	37	98	75	325	1.0	25.6
7	IEFSD 685	3,500	31	48	44	60	103	32	318	1.0	25.1
8	IEFVHA	33,000	12	33	35	42	91	41	255	0.8	20.1
9	IEFSD 086	3,000	31	28	37	50	59	20	225	0.7	17.7
10	IEFVHG	5,500	9	24	25	28	92	32	210	0.7	16.5
	IEFXVAVR	19,500	17	21	19	33	46	42	178	0.6	14.0
	IEFSD 070	800	16	32	23	30	52	18	173	0.6	13.6
13	IEFSD 087	3,000	16	15	23	30	54	18	156	0.5	12.3
14	IEFSD 080	6,500	17	20	21	21	11	7	97	0.3	7.6
15	IEFSD 078	400	16	18	17	20	10	6	87	0.3	6.9
16	IEESD 562	1,900	17	20	5	3	11	23	79	0.2	6.2
17	IEESD 565	700	12	15	1	1	11	17	57	0.2	4.5
18	IEFSD 094	4,000	16	0	7	0	10	6	39	0.1	3.1
19	DEVMASKT	400+	46	0	33	0	0	0	79	0.2	6.2
	IEFALENT	50	43	0	26	0	0	0	69	0.2	5.9

1. SVCLIB and RESIDENT SVC's

11. This paper discusses machines where there is a core space problem. On these machines one cannot necessarily find space for the supplies list of Resident SVC's (which requires over 28k) even if it does seem worthwhile on performance grounds to have that list resident.

12. However without Resident SVC's activity to SVCLIB is a very important factor. The 'OS Dataset Use' table shows figures of between 12% and 39% of all disk I/O's being to SVCLIB. Often the total of all I/O times to SVCLIB (EXCP to POST) approaches the time traced – ie. systems can be virtually always doing I/O to SVCLIB. So it is important to use the option of making some transient SVC's resident.

13. The supplied list, besides being rather large for many configurations, does not include the two SVC routines that invariably seem to head the popularity pole for transient SVC's. IGC0007B and IGC0107B, the two operator communication modules, are always highly used in non-HASP environments. Making them resident also has the useful result of giving the operators the impression that the machine is going faster.

14. For most users at least the leading 5 members of SVCLIB should be made resident. These 5 (IGC0007B, IGC0107B, IGG0201Z, IGG0190N, IGC0001I) will occupy approximately 4,412 bytes of core-module sizes may vary with Release Level. These 5, on the example figures in the 'SVC Members Use' table, will eliminate some 14½% of the activity to SVCLIB. The standard (Ria) list at 6½ times the core requirement would save only about 2½ times as many accesses. The 21 SVC's of the 'SVC Member Usage' table would save rather more accesses at only three-quarters of the core.

15. To Summarize:

	Entries	Approx core	% of SVCLIB activity eliminated
Standard list	28	28,600	35½%
Top 5	5	4,400	14½%
Top 21	21	21,200	36%

16. This indicates the importance of a good RSVC list. Do not forget that the above figures represent averages over a number of different users. A list properly 'tailored' to a particular user gives even better improvements over the standard list. If you are looking for a basic list to use until a tailored list can be established then the 'SVC Member Usage' table should be a good basis for selecting 5 SVC's or 10 or 15 or however many you wish.

17. The problem with advising customers to tailor their RSVC lists is that they then ask how to find out what SVC's to make resident (and perhaps how many to make resident). AMAP traces only sample a small portion of a customers workload so that one AMAP run could produce an inaccurate picture. SVC use should be monitored over extensive periods to get a true pattern of use, and thus develop an optimum list.

* 18. An ideal tool for studying SVC (and LINKLIB) loading is CATCH. This is available to World Trade as AIDS-WT-07R (08R for MVT) in the SERUM series. There is however the unfortunate difficulty that it is only available for use on IBM Internal machines at the date of writing this paper.

19. A generally available but less tidy approach is to use the CE Service Aid program TFLOW. This is a trace program available in different versions for different OS Releases. Exit routines are fairly easily written for TFLOW. To study SVCLIB accessing an exit routine can be produced that causes the tracing of only the TTR's of I/O's to the SVCLIB disk (or just SVCLIB). These records can then be sorted, totalled and related to member names by a PDS Directory list. A simple PL/I program is advised for the totalling rather than the slow

exit is strongly advised to be the better approach.

110. Note that it is important to record actual SVCLIB member accessing rather than the issuing of a SVC. For example one OPEN can result in the loading of many members.

111. Also note that certain SVCLIB members are very prone to being refreshed. One important sufferer from frequent refreshing is IGG0190N which is one of the top 5 list above. On a busy system with a number of partitions it is therefore even more necessary to have this member resident. Some studies elsewhere, so far unpublished, illustrate that SVC Transient Area Refresh is a definite handicap to large MFT systems. For a 360/65 a ratio of 1.25 accesses to SVCLIB for every transient SVC executed has been found.

112. Monitoring the loading of members implies running for these periods with no SVC's resident. This requires that the operators can remember to IPL this way - if not then change the PARMLIB default RSVC entry to one innocuous member. Luckily running with no RSVC's will be harmless, in contrast to the dangers of running with no Resident Access Methods.

113. Even if you know from AMAP the average I/O time to SVCLIB when running with no Resident SVC's it is difficult to estimate the effect of a particular RSVC list. One can calculate the total I/O time that will be directly eliminated. Bear in mind when doing this calculation that some members require directory references (eg. IGC0107B) so that making them resident saves two accesses per use. This directory is only referred to generally for second and subsequent loads of SVC's, but excluding routines such as OPEN, CLOSE etc. which avoid the directory. This is mentioned in the section on BLDL lists for SVCLIB in the Systems Programmers Guide SRL (R20.1 level).

114. Some detailed before and after studies are needed so that the absolute effect of making a particular set of SVC's resident can be measured. Few, if any, such studies are documented, at least not to the authors knowledge. With this lack of factual evidence, the author, probably foolheardedly, & suggests that in a fairly

waiting for I/O to SVCLIB (some proportion of which will be unoverlapped) there are various second order effects. Having a certain set of SVC's resident will not only speed up use of those SVC's by jobs in the machine and by the system, but I will also, give considerable improvements in such things as:

Average I/O time to the rest of SVCLIB
(both the record reading component and the preparatory queuing for channel, device etc.)
Reduced interference with and from other datasets.
Reduced channel road (by perhaps 1 or 2%).
Reduced CPU Utilization (lots of I/O and I/O set-up saved).
Distinct alleviation of that under-appreciated problem of SVC Transient Area Refresh.

115. To complicate matters it is not just a question of choosing which SVC's to put in your list, but also of how to allocate core amongst the various Resident Options (RSVC, RAM, BLDL) and in competition with other options such as PCI Fetch, SMF, etc. Even if one could accurately quantify the benefits associated with each variable it would require an LP program run to find an optimum balance.

However the usual complete guesswork can be improved upon by at least calculating the accesses saved per kilobyte of core used for RSVC, RAM and BLDL. If average I/O times per access are established for the condition of running with reasonable lists then the total I/O time saved per kilobyte can be worked out. This is useful in comparing the worth of one extra resident SVC versus several extra BLDL entries. Unfortunately it can only partially substitute for what the user really needs to know i.e. the performance trade off, on his system, from a change in resident options.

SVC MEMBER USAGE

	MEMBER	CORE REQUIRED	X	L	B2	R	B1	D	TOTAL	REFS PER MINUTE	REFS PER MINUTE PER KILOBYTE	TOTALS FOUND IN SETR-011 STUDIES
	DIRECTORY	-	672 (7.0%)	2110 (15.7%)	1229 (8.1%)	2053 (10.8%)	2393 (7.0%)	1993 (9.9%)	10455 (9.4%)	32.2	-	10,819
1	IGC0007B	304	E=330	912	E=700	721	716	758	E=4137 (3.7%)	12.7	41.9	4,395
2	IGG0201Z	1064	192	309	313	463	740	644	2661 (2.4%)	8.2	7.7	1,601
3	IGG0190N	1064	223	222	408	398	810	416	2477 (2.2%)	7.6	7.2	1,634
4	IGC0107B	1064	E=140	614	E=400	414	441	458	E=2467 (2.2%)	7.6	7.1	2,826
5	IGC0001F	916*	182	155	259	313	387	640	1936	6.0	6.5	2,035
6	IGG0200F	1064	121	229	294	259	705	275	1883	5.8	5.5	1,417
7	IGC0002?	1064	142	209	266	295	713	221	1846	5.7	5.3	1,359
8	IGG0200Z	1064	150	215	302	330	543	236	1776	5.5	5.1	1,703
9	IGG0001I	1064	143	128	E=300	328	570	275	E=1744	5.4	5.0	1,590
11	IGG0200Y	1064	95	180	224	233	627	296	1660	5.1	4.8	1,526
11	IGC0002E	880 *	217	16	321	438	6	385	1383	4.3	4.8	-
13	IGG0190L	1064	136	179	309	297	474	242	1637	5.0	4.7	1,445
14	IGG0190S	1064	146	177	257	295	473	257	1605	4.9	4.6	1,421
15	IGG0191A	1064	137	173	227	286	461	254	1538	4.7	4.5	1,424
15	IGG0190M	1064	136	173	259	297	443	257	1565	4.8	4.5	1,414
15	IGG0199M	1064	130	180	250	286	454	247	1547	4.8	4.5	1,297
18	IGG0200G	1064	125	180	227	289	452	209	1482	4.6	4.3	1,421
19	IGG0190Z	1064	95	165	222	232	460	232	1406	4.3	4.1	1,335
20	IGG0191N	1064	103	162	212	222	443	234	1381	4.3	4.0	1,332
21	IGC0003E	976*	115	237	E=196	195	218	190	E=1145	3.5	3.6	1,295
	TOTAL RECORDS FROM SVCLIB	-	9675	13414	15110	19020	34200	20160	111,579	343.3 (5.7 per second)		

DESCRIPTION OF SVCLIB MEMBERS

1.	IGC0007B	OP COMMUNICS - Router (SVC72)	N
2.	IGG0201Z	CLOSE - SAM/PAM for DA Devices	Yes
3.	IGG0190N	OPEN - Final Module	Yes
4.	IGC0107B	OP COMMUNICS - 1052 I/O	N
5.	IGC0001F	PURGE (SVC 16)	Yes
6.	IGG0200F	CLOSE - Direct Access Processing	Yes
7.	IGC0002?	CLOSE - Initial load	Yes
8.	IGG0200Z	CLOSE - Where to Go	Yes
9.	IGG0190I	OPEN - DA DSCB Processing	N
9.	IGC0001I	OPEN - Initial Load	Yes
11.	IGC0002E	DA Track Balance (SVC 25)	N
11.	IGG0200Y	CLOSE - Direct Access Processing	Yes
13.	IGG0190L	OPEN - Merge and AM Determination	Yes
14.	IGG0190S	OPEN - Final Load, Rewrite JFCB	Yes
15.	IGG0199M	OPEN - Merge	Yes
15.	IGG0190M	OPEN - DCB Exit	Yes
15.	IGG0191A	OPEN - SAM DEB Construction	Yes
18.	IGG0200G	CLOSE - Delete Subrtn. and Restore DCB	Yes
19.	IGG0190Z	OPEN - DA Mount Verification	N
20.	IGG0191N	OPEN - SAM DEB Construct for DA devices	N
21.	IGC0003E	WTO/WTOR (SVC 35)	N

Notes for SVC Member Usage Chart

E=nnn	indicates that this figure is an estimate as the member was core resistant for this study
Percentages	in parenthesis are percentages of total records from SVCLIB.
Asterisk	indicates size from the PDS directory entry disagrees with storage Estimates SRL. Size can also vary with release.

ACCESS METHOD MODULE USAGE

	MEMBER	CORE REQUIRED	NO OF REFERENCES IN EACH STUDY					TOTAL	REFS PER MINUTE	REFS PER MIN PER KILOBYTE
			L	B2	R	B1	D			
1	CC	216	67	62	98	149	78	454	1.5	7.1
2	CH	136	68	62	96	144	79	449	1.5	11.2
3	CD	552	49	48	66	164	83	410	1.4	2.5
4	AR	168	43	Res	48	154	58	say 350	1.2	7.1
5	AQ	144	58	36	60	122	46	322	1.1	7.6
6	CI	256	58	46	33	118	42	297	1.0	3.9
7	AK	456	30	32	34	134	44	274	0.9	2.0
8	AA	128	46	34	47	85	29	241	0.8	6.4
9	BA	408	23	Res	47	57	50	say 210	0.7	1.7
10	BB	112	27	Res	47	57	50	say 210	0.7	6.3
11	BC	256	19	23	37	35	45	159	0.5	2.1

'Res' indicates the module was core resident for this study

Core Required is PDS Directory size plus 24 bytes. The size can vary with OS Release.

LINKLIB MEMBERS AND ALIASES

	MEMBER	DESCRIPTION	ALIASES
1.	IEESD563	Master Scheduler – Q Search Setup	
2.	IEESD564	Master Scheduler – Q Search	
3.	IEFSD510 (d)	Initiator – Job Selection	GO(c) IEFSD511 (f) IEFSD516 (e) IEFV4221 (f) IEFW42SD (e)
4.	IEFSD512 (f)	Initiator – Step Initiation	IEFALRET (a)
5.	IEFSD526 (d)	I/O Device Allocation	IEFWC000 (c) IEFWD000 (f) IEFX5000 (f) IEFW41SD (c)
6.	IEFW21SD (d)	Initiator – Initialization	IEFSD556 (c) IEEVMCVL (f) IEFVMI (e) IEFXA (c)
7.	IEFSD085 (f)	Writer – DSB Handler	IEFSD071 (c) IEF085SD (c) IEFSD850 (f)
8.	IEFVHA (b)	Reader – Get Routing	IEFVHAA (f) IEFVHB (f) IEFVHC (f) IEFVHCB (c) IEF536EP (e)
9.	IEFSD086 (c)	Writer – SMB Handler	IEF086SD (d)
10.	IEFVHG	Reader – CPO Routine	
11.	IEFXVAVR (f)	AVR	IEFXV001 (c)
12.	IEFSD070	Writer – Data Set Writer Interface	–
13.	IEFSD087	Writer – Standard writer	–
14.	IEFSD080 (f)	Writer – Initialisation	IEFSD079 (a) IEEVWRT1 (f)
15.	IEFSD078	Writer – Linker Routine	–
16.	IEESD552	Master Scheduler – Syntax Check	–
17.	IEESD565	Master Scheduler – Service Routine	–
18.	IEFSD094	Writer – Job Separator Sheet	–
19.	DEVMASKT	Device Mask Table	–
20.	IEFALENT		–

ALIAS USE: The underlined names above are the names that should be considered for a BLDL list. The letters in brackets alongside each name for aliased members indicate roughly the proportion of references likely by that name. The letters indicate:

- a all, or nearly all
- b over 75%
- c 50 – 75%
- d 25 – 50%
- e below 25%
- f zero, or very few

C. Disk 291

Almost a half (8,061) of the 17,876 accesses to this disk were to LINKLIB. Most of the rest were for SYSIN records. Only 38 accesses were made to the blocked (at 1,040) PROCLIB in this hour or so.

The average access time to LINKLIB was 69.4 milliseconds. Only on 6% of the accesses did the arm seek to LINKLIB. Just over 1,000 of the accesses in this run were to the directory of the library. The BLDL list section of the report discusses the breakdown of references by module of LINKLIB.

Note that several of the most heavily used modules (e.g. the scheduler) are located in the end few cylinders of the 90-cylinder dataset, remote from the directory. 560 seconds of the run involved I/O to LINKLIB.

A 2314-based LINKLIB would involve considerably less seeking as it would require less than 22 cylinders in contrast to 90 cylinders here. LINKLIB records could be generated at 7K maximum record size instead of 3K or so. On 2311 the 19,424 bytes of IEFXVAVR came into core by 34 separate accesses.

Additionally a 360/50 is fast enough to make use of the PCI-Fetch facility of OS. Typical 2314-resident systems bring in about 2 physical records per access averaged over the run - i.e. the number of accesses to LINKLIB (and JOBLIBS) is halved.

The storage requirement for PCI-Fetch is 3,472 bytes. This seems high but should be related to, say, the current BLDL list which costs 2,444 bytes to save for this run 642 accesses. At 69.4 milliseconds per access the I/O time saved would be 45 seconds. If PCI-Fetch saves only 1,000 of the 7,000 non-directory references it is as useful as the current BLDL list.

The effect of PCI-Fetch is difficult to estimate quantitatively. It is probably reasonable to assume a saving of 3,000 separate accesses at approximately 70 millisecs each, i.e. 210 seconds of I/O time over the run. If about half that time is realised as a reduction in running time, PCI-Fetch would be saving 105 seconds in 70 minutes, or $2\frac{1}{2}$%. The accessing of JOBLIBS will also be improved.

PCI- Fetch should be included in the system as it will improve performance, speed job scheduling etc. The improvement should be worthwhile but probably does not justify a special SYSGEN.

D. Disk 292

This disk held SYS1.JOBQUEUE and SYSPRINT datasets. The accesses to JOBQUEUE were 8,358 of the 25,537 total (33%). There was some interference with JOBQUEUE, the arm being elsewhere 21% of the time. The average I/O time to JOBQUEUE was 62.5 milliseconds.

END