digital

Software
Services

AMAR-20
REFERENCE MANUAL
VERSION 4.1

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by Digital or its affiliated companies.

The following are trademarks of Digital Equipment Corporation.

digital

DEC
DECUS
DECnet
DECsystem-10
DECSYSTEM-20
RSX
VAX
VMS
UNIBUS
RSTS
PDP
AMAR
AMAR-10
AMAR-20

AMAR™, AMAR-10™, and AMAR-20™ are new trademarks of the Digital Equipment Corporation.

AMAR-20 REFERENCE MANUAL

RELEASE 4.1

TABLE OF CONTENTS

CHAPTER 2 WORKLOAD AMAR

blank page

PREFACE

The AMAR (Automatic Measurement, Analysis, and Reporting) software monitor looks at computer system performance and resource use on a continuous basis and maintains an historical database. It provides periodic reports which are useful for problem detection and analysis, load balancing, and capacity planning.

Currently there are two parts to the AMAR software monitor - System AMAR and Workload AMAR. System AMAR monitors the activity of the computer as a whole and the activity of individual devices such as tape drives and disk packs. Workload AMAR (also called the Workload System) monitors the corresponding activity of individual jobs.

The AMAR-20 Reference Manual provides an overview of how AMAR works on the DECsystem-20. It describes both System AMAR and Workload AMAR. It provides an explanation of sample reports, instructions for obtaining special reports, and a description of normal processing procedures.

A companion volume, AMAR-20 Error Messages, documents all error messages produced by AMAR programs, their likely cause, program action, and recommended user action.

blank page

INTRODUCTION

The AMAR (Automatic Measurement, Analysis, and Reporting) software monitor monitors computer performance and utilization on a continuous basis. It maintains a history of the computer's activity in a set of databases and provides periodic reports on the status of the system.

AMAR is intended to be used to detect bottlenecks and trends in computer usage. It provides the opportunity for you to avoid poor performance by directing efforts and resources to the appropriate problem area. It reduces the lead time required to diagnose problems and provides data which can be used for forecasting and other planning purposes.

AMAR consists of two parts. The first part, which is referred to as System AMAR, collects data continuously on a set of system performance and utilization variables. A database exists which maintains several summary levels of historical data. In addition, there is a menu of reports which can be generated on a daily, weekly, or monthly basis. Reports fall into five categories as follows:

1. System Utilization Summary Reports: Reports characterizing a day's, week's or month's activity. These reports are primarily used for performance and utilization problem analysis and tracking. They provide a graph of CPU utilization and overhead; a summary of problem periods and resources; and a summary of system availability.

2. 'Typical Day' Reports: Reports on system utilization and problem identification for the "average" workday and "average" weekend day of a week or month. These reports aid load balancing by highlighting the typical busy periods.

3. Trend Analysis Reports: Reports which indicate both short term and long term trends in computer utilization. They also project periods when resource consumption may become critical. These reports aid forecasting and capacity planning. They are available both weekly and monthly.

4. Disk Reports: Reports which summarize utilization of the disk subsystem. All disk related information, such as mount time, time in use, average data transfer rate, etc., is presented in a single place.

5. Tape Reports: Reports which summarize utilization of the tape subsystem. All tape related information, such as assigned time, time in use, average data transfer rate, etc., is presented in a single place.

The second part of AMAR is referred to as Workload AMAR. Job specific data, such as job number, user name, program name, working set size, CPU utilization, page fault activity, and so

forth, is collected at 5 minute intervals. It is then summarized and, depending on summary level, may be reported in intervals of 5 minutes to a day, week, or month.

The data, also retained in an historical database, is used to locate and solve problems associated with excessive utilization. This data is also expected to provide information for use in a number of areas among which are - forecasting; justifying hardware acquisitions; and providing workload descriptions to be used in benchmarking hardware and software in order to determine the optimal configurations for use by data centers.

Workload characterization reports are generated by a program which allows user definition of report contents. You may specify items around which the report is to be summarized (for example, program name, user name, account name) and sort order (for example, largest users of CPU first).

Reports for both parts of AMAR are designed to be as self-explanatory as is technically feasible. Most reports can be obtained automatically. Moderately flexible data inquiry and report generating capabilities are also provided to address special needs.

This manual describes how to use AMAR on the DECsystem-20. It is divided into two main chapters plus several appendices. Chapter 1 describes how System AMAR works, it's standard reports, instructions for obtaining special reports, and normal processing procedures. Chapter 2 provides comparable information on Workload AMAR. The appendices contain detailed information on several topics presented in Chapters 1 and 2. They are intended primarily for reference.

GLOSSARY

CLASS WIDTH - The length of the intervals into which the frequency distribution for a variable is divided and into which the individual sample values are grouped. For example, % IDLE TIME (IDLE) has a class width default of 5 which means individual sample values are grouped into the following intervals: 0%; 0.01-5.00%; 5.01-10.00%; 10.01-15%; etc. A separate class is always set up for 0 values regardless of the class width specified.

DATA COLLECTION - The process of sampling and recording on disk the selected System AMAR items and workload information.

DATA INPUT - The process of incorporating the raw records from the monitored system into the database.

GRANULARITY - Refers to various summary levels of data; lowest level, or finest granularity, is a raw record, highest level, or coarsest granularity, is a summary record for the longest rolled-up period, which is a usually a fiscal month.

HIGH THRESHOLD - The limit above which an item value is considered to be unacceptable or beyond the range of expected or customary values.

ITEM - A specific variable, which can be measured, related to the utilization or performance of a system. For example, idle time, overhead, swapping rate, etc. are items.

KEY ITEM - Any item appearing in a System AMAR report subsection which has been labeled as "key" in the RFD file.

LOW THRESHOLD - The limit below which an item value is considered to be unacceptable or lower than the range of expected or customary values.

METERED - Refers to the type of measurement of an item in which the exact change, since the last measurement, is known or can be computed. The measurement is accurate and is unaffected by the length of the sample interval or number of samples taken.

MONITORED SYSTEM - A computer system on which an AMAR data collection program collects raw performance data.

RAW DATA - Refers generally to the values stored in the raw records, whether in core or on the disk. Raw data may also refer to the finest level of granularity of data on the system which is being monitored.

RESOURCE - An item, regardless of whether or not it represents a physical "resource".

ROLLUP - The AMAR process of consolidating detail data into the

appropriate summary level according to the fiscal calendar.

SAMPLED - Refers generally to the measurement of an item (either metered or snapped).

SAMPLE GROUP INTERVAL - The frequency at which the data collection program writes raw records onto the disk.

SAMPLING INTERVAL - Frequency at which the data collection program takes samples or measures raw data.

SNAPPED - Refers to the type of measurement of an item in which the value of the item is a 'snapshot' of conditions existing at that moment in time and does not reflect other values for that item which may have occurred since the last sample. The accuracy of this form of measurement is dependent on the number of samples taken during a given period of time.

UNDEFINED STATE - If the system crashes or the input or rollup programs fail while the System AMAR database is being updated, errors could be introduced into the database records. Further use of the database may result in erroneous information being obtained. The jobstream will first attempt to recover by using the backup copy on disk. If this fails, restore the database from a tape backup copy and resume processing.

WORKLOAD - The collection of user programs running on the system which is being monitored.

CHAPTER 1

SYSTEM AMAR

1.1 MAJOR FEATURES

Features of System AMAR include:

1. Low overhead, continuous data collection.
2. An historical database with:
   a. Data summarized at the hourly, daily, weekly, monthly, and "typical day" levels.
   b. Data values stored in frequency distributions.
   c. Flexible retention periods for data with monthly values typically kept at least a year.
   d. Automatic deletion of old data.
3. A menu of standard reports which:
   a. Track utilization and performance over the period of a day, week, month or year.
   b. Automatically check for and flag problem items and time periods.
   c. Analyze and report on short term (up to 13 weeks) and long term (up to 12 months) trends.
   d. Consolidate all disk and tape information onto separate reports.
4. Special reporting programs which:
   a. Are user runnable at a terminal or via a batch stream.

b. Provide for ad-hoc report generation.

c. Enable you to directly access any piece of information contained in the database.

d. Provide data in human readable form or, alternately, in a form useable by other programs.

5. A single daily batch stream which will:

a. Automatically produce daily, weekly, and monthly reports according to a fiscal calendar.

b. Maintain the System AMAR database.

c. Prevent buildup of data files on disk.

1.2 OVERVIEW OF SYSTEM AMAR

The three functions of System AMAR are data collection, database management, and reporting. These functions are performed by seven programs which are described briefly below. Refer to Figure 1-1 for an overview of program and data flow. The bold portions of Figure 1-1 refer to functions that are normally performed automatically.

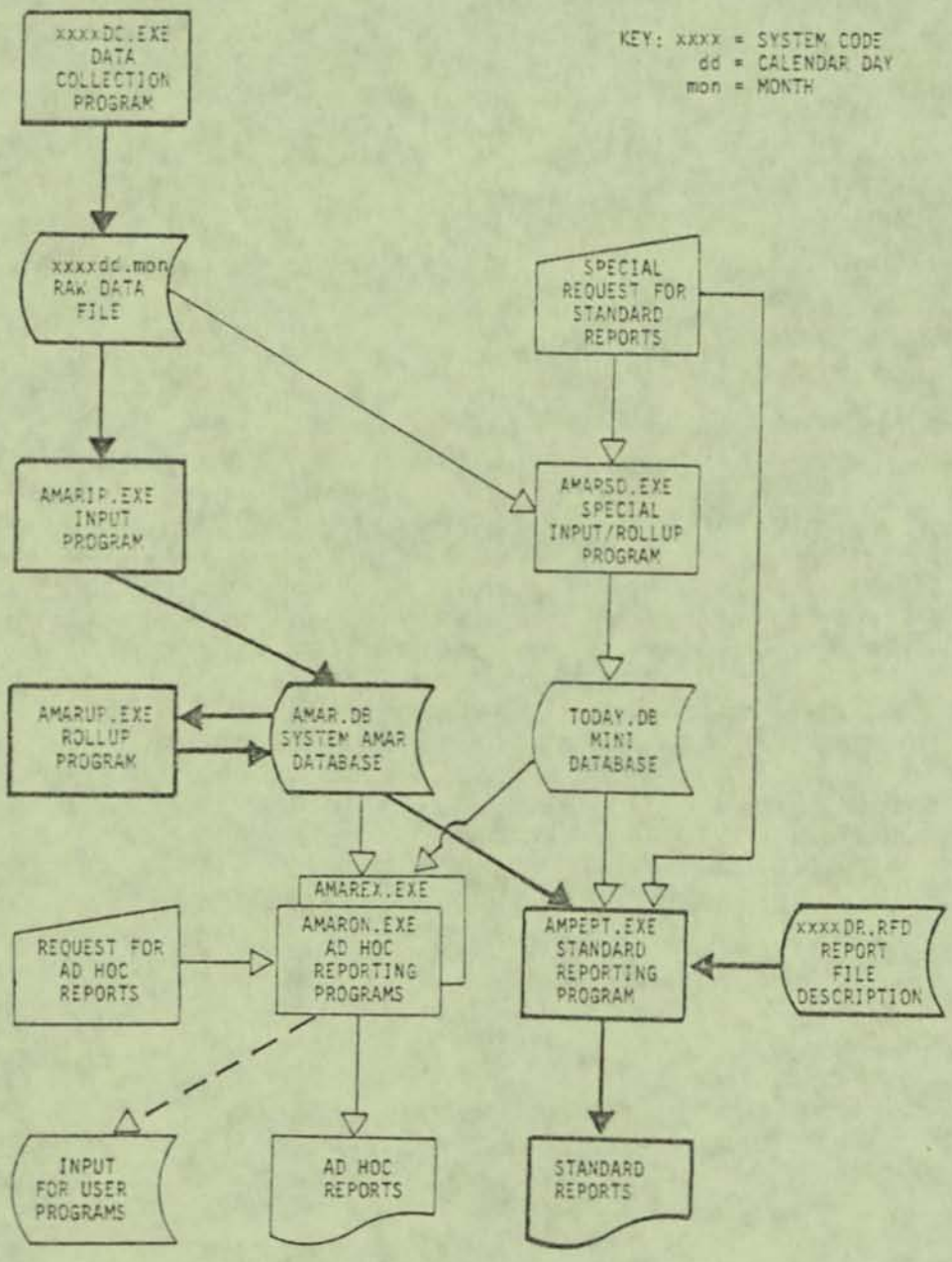
1.2.1 Data Collection

The data collection program, named xxxxDC where xxxx is a 4-character system code, collects data related to the general utilization and performance of a CPU and its associated subsystems - memory, disk and tape.

xxxxDC interfaces with the monitor, and does the actual sampling and measuring of the various System AMAR items which may be specified by the user. The sample data are collected in core. At the end of a sample group interval (1 hour) raw records are created from the sample data and are written to disk. Raw records contain a frequency distribution of the sample data values for a given item. Each record represents a separate predefined range of values, contains the average for the range, and the number of samples whose values fall into that range. xxxxDC runs 24 hours a day, preferably as a SYSJOB subjob.

1.2.2 Data Input

The data input program, AMARIP, performs the first half of the database management function. AMARIP inserts the raw records into the database. It keeps track of when data was last input and uses a grace period to determine how many days worth of data should be input into the database at one time. AMARIP requests a separate raw data file for each day since the last raw file was input up to the current day or last day of the grace period, whichever comes first. The grace period is equal to the number of days that hourly data is kept in the database. It prevents unnecessary processing in the case where the operator may have incorrectly set the machine date too far in advance.



SYSTEM AMAR OVERVIEW
FIGURE 1-1

1.2.3 Rollup

The rollup program, AMARUP, performs the second half of the database management function. It rolls hourly data up into daily, weekly, monthly, and "typical day" records. AMARUP also deletes any data records which have expired their retention periods.

1.2.4 Report Generation

Automatic report generation is performed by AMREPT. AMREPT produces a menu of standard reports, described in the Section called "Annotated Sample Reports". These reports provide information on system utilization and performance, problem resources and periods, system availability, and usage trends. You may elect to generate only a subset of these reports or the entire menu. Reports may be produced on a daily, weekly or monthly basis according to a fiscal calendar. Weekly and monthly reports are triggered by the passing of the end of the fiscal period to be reported upon. Reports are automatically generated only once per fiscal period by AMREPT. AMREPT, however, may be run interactively to recreate reports for past time periods.

In addition to automatic report generation, System AMAR provides you with the capability to generate special reports. There are three programs in this category, AMARSD, AMARON, and AMAREX.

AMARSD is a variation of the input program AMARIP. AMARSD makes a separate mini database from the raw daily file, by-passing the System AMAR database. The mini database is then forwarded to AMREPT for output to a Daily System Utilization Report or Disk or Tape Report. AMARSD enables you to generate a report from any raw file including the one currently being created, thus obtaining up-to-the-hour information on system utilization.

AMARON is an online inquiry program which can be used to report on any individual item contained within the System AMAR database or the mini-database output by AMARSD. AMARON may output the data as either tables or histograms. Reports may be sent to the terminal or stored on disk for later printing or further program use. Headings may also be stripped from the reports to enable easy access by statistical packages or by your own program.

AMAREX is an online inquiry program which can be used to extract records from the database. The records can be displayed at the terminal or sent to a file. AMAREX records are primarily intended for further processing by user programs.

1.2.5 Operation

The data collection program (xxxxDC) should be run continuously. This program should be set up as a SYSJOB subjob startable by the SYSJOB.RUN file. Except for data collection, System AMAR operation is accomplished through the use of a single job stream, called AMAR.CTL, which will usually be run after midnight. The stream will perform housekeeping on old files, run the input, rollup, and automatic report program, and produce a tape backup of the database if desired. A sample jobstream is provided in the Appendix called "System AMAR Batch Stream - AMAR.CTL". This stream is usually modified slightly to conform to a site's individual requirements.

1.3 ANNOTATED SAMPLE REPORTS

System AMAR is currently designed to generate the following five standard types of reports:

1. System Utilization Reports - Can be generated daily, weekly, or monthly. They provide a picture of the past day, week, or month's activity, primarily to aid in problem identification and solution.

2. Trend Analysis Reports - Can be generated weekly or monthly. They highlight both short and long term trends in utilization and performance to aid in forecasting and capacity analysis.

3. 'Typical Day' Reports - Can be generated weekly or monthly. They aid the load balancing process by providing an indication of "typical" busy/non-busy periods.

4. Disk Reports - Can be generated daily, weekly, or monthly. They combine all disk related information, such as pack utilization, I/O rate, and free space, in one place.

5. Tape Reports - Can be generated daily, weekly, or monthly. They combine all tape related information, such as drive utilization and I/O rate, in one place.

All of the above reports can be produced automatically by the AMREPT program which is run as part of the nightly batch stream, AMAR.CTL. Alternately, the same reports may be produced on demand by running AMREPT at the terminal. Two programs, AMARON and AMAREX are also provided which enable you to generate special reports. AMARON allows you to obtain averages or histogram data for any item contained in the AMAR database. Results can be printed at the terminal or written to a file for later processing by your own program or by statistical packages. AMAREX allows

you to extract any record from the database. The records are written to a sequential file for later processing by your own program.

The following sections describe the System AMAR reports and show annotated examples of each.

1.3.1 System Utilization Reports

Utilization Summary Reports can be generated for a day, week, or month. They are designed to be used together. The report formats are similar. Problems showing up on a monthly report can be easily traced back to specific hours within a day. Utilization Summary Reports show activity on the system as a whole. They should be used in conjunction with those Workload AMAR reports which show corresponding activity by user programs. See the Workload AMAR chapter in this manual. Report similarities and differences are described in the following subsections.

1.3.1.1 Daily System Utilization Report -

The Daily System Utilization Report provides the following features:

A graph of CPU utilization and overhead by hour which denotes busy periods. Overhead is defined under Summary Report on the next page.

A prime/non-prime time summary of key utilization items such as idle time, used time, background time, swapping rates, disk I/O rates, and so forth, for quick scanning by data center management.

Disk free space by pack comparing yesterday with today and giving an indication of trend.

A summary of problem periods.

A summary of problem resources (also called "items").

A summary of system availability for prime/non-prime time.

A list of periods of downtime.

Hour-by-hour averages for key items as well as problem items. This feature allows most problem periods and problem resources to be pinpointed and analyzed without the need to generate any other special reports. In addition, it is

designed to allow easy comparison with the workload reports which produce hour-by-hour summaries of user programs.

The Daily System Utilization Report is separated into three sections - a Summary Report, a Problem Report, and a Detail Report. Refer to Figure 1-2. The format of the report header is as follows:

Lines 1 and 2 - Specify the date for which the report is generated, the name of the report, and the page number. The report period begins at approximately 00:01 AM and ends at approximately 23:59 PM.

Line 3 - Specifies the Data Center Name.

Line 4 - Specifies the 4 character system code and the period of time during the day which is considered prime time.

Summary Report (Pages 1-2) -The purpose of this report section is to give management a quick overview of system utilization and peak processing periods during the previous day.

The graph at the top of page 1 (A) shows CPU utilization (*) and overhead (#) for each hour of the day. The difference between 100% and the CPU utilization (B) line represents idle time plus file wait time. See % IDLE TIME and % IDLE IO TIME in the Appendix called "System AMAR Item Definitions." The difference between the CPU utilization (B) and overhead (C) lines represents CPU time consumed by user processes plus time spent in paging and swapping on behalf of a user (see % USED TIME, % PAGING TIME, % MGMT MEM TIME, and % IDLE SWP TIME.) The overhead (C) line represents scheduler and background time (see % SCHED TIME and % BACKGND TIME.)

Below the graph are prime and non-prime averages for twenty "key" items (D). These averages give a quick picture of the usage during the day. You have the option of selecting any number of key items from the entire list of items that AMAR collects. A default set of key items is supplied with AMAR. You can change this set by editing the xxxxDR.RFD File. See the Section called "How to Tailor the Report Contents". The items and their definitions are listed in the Appendix called "System AMAR Item Definitions". The last column in the row of key items gives the total number of hours (E) each key item exceeded or equaled the thresholds. Hours where the averages exceeded the thresholds are denoted by asterisks (*) on pages 4-6 of the sample report.

The section on page 2 contains disk free space by pack. PERCENT FREE SPACE LEFT ON DISK PACKS is an average of samples taken throughout the day rather than just the reading at the end of the day. The current day's average (F), the previous day's average (G) and the difference (H) between the two are given. If the pack has filled up significantly since yesterday (large negative difference), some housekeeping may be in order.

Problem Report (Page 3) - The purpose of this report section is to give management an overview of potential problem periods and problem items. Problem periods (I) are indicated graphically in the lefthand section of this report. For each hour, the number of items whose averages exceeded the watchdog limits is indicated by a row of asterisks (for key items) and plus signs (for other than key items) (J). Periods of the day which are followed by several asterisks or plus signs should be investigated more thoroughly - generally by first looking at the Detail Report (page 4).

Problem resources (K) are listed in the righthand section of the report. The number of hours the resource values exceeded watchdog limits is indicated graphically by a row of P's (for prime-time hours) and N's (for non-prime-time hours) (L). A brief comment suggests a possible cause or an approach to solving the problem (M). The comments are intended only as a pointer to a possible problem area or to some further course of action. There may be more than one possible cause for a given problem (for example, such as too much background time) and there is often more than one solution which can be applied. The alternatives must always be evaluated carefully. The comments, in and of themselves, are not intended to recommend a solution to a given problem. Comments are defined in the xxxxDR.RFD file. You may modify or delete them.

The SYSTEM AVAILABILITY SUMMARY (N) at the bottom of the page expresses system uptime (P) for prime and non-prime time as a percentage of wall clock time. AMAR measures uptime from time of system reload to within one minute before a system crash.

The percent of time measured by AMAR (Q) may be somewhat less than system uptime, because AMAR measurements (other than uptime) taken in the partial hour before a system crash are not recorded on disk.

A reload (R) will be counted if the system stays up long enough for AMAR to write its reload record.

Periods of downtime are listed (S). The beginning of a period of downtime should be accurate to within a minute. The end of the period of downtime coincides with a system reload.

Detail Report (Pages 4-6) - The main purpose of this report section is to give the user investigating a problem period or problem item an hour by hour report of item utilization. Problems are flagged with asterisks to make them stand out. This report also provides summaries of item use for prime and non-prime time, with indications of how often the problem items exceed watchdog limits. The detail report consists of a table (T) with a row for each hour of the day and a column for each of ten key items. Under each item name is listed the average value for each hour (U). To see corresponding usage by individual programs, refer to the Workload AMAR "Hourly Report by Program and User". Note that usually averages and percentages are

expressed as whole numbers - except in the case of items whose values are usually less than 1 such as for BAL SET ADJS/SEC.

Any element of this table which exceeds the watchdog limits is flagged with an asterisk (V). A row with several asterisks is likely to represent a problem period. A column with several asterisks usually indicates an overused item. If less than 45% (27 minutes) of the hour was measured, the data values are shown as "---", usually indicating system downtime (or, in the case of disk packs, that the pack was not mounted).

After the row-per-hour table, there are summaries for prime and non-prime time which give average value (W), watchdog limit (X), percent of time over limit (Y), and number of hours (Z) when the average was over or equal to the watchdog limit.

A maximum of ten key items are contained on the first page of this report. Key items are meant to be always printed. Items other than key items may also be always printed or be printed only if at least one hourly average exceeds or equals the threshold that has been defined for the item or the threshold is exceeded or equaled 10% of the time. If items other than key items are over limits, or more than ten key items are selected for daily reporting, these items will be included in additional pages of detail report.

For purposes of the Daily Report, watchdog limits may be set differently for each item, including individual disk packs. For example, this will allow different treatment for a pack which is 95% full, but stable and a pack which should average 30% free space to accommodate peak usage.

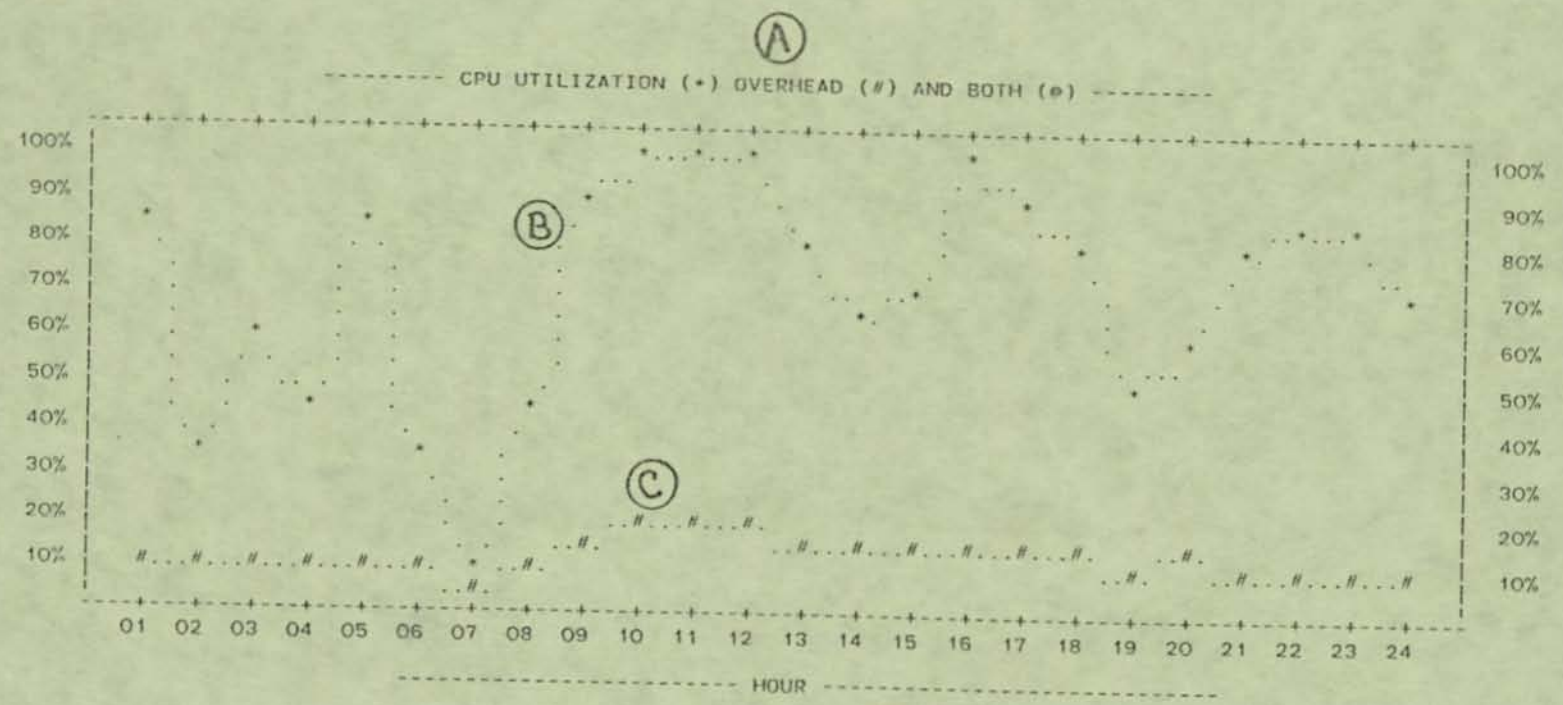
To indicate when it may be useful to check the other pages of the detail report, the column labeled ALL PAGES (AA) indicates the total number of problem items during the period -- if it is different from the next to last column labeled THIS PAGE (BB), the total difference is the number of items on other pages which were over limits during the period.

DATE: 26-OCT-82 (TUESDAY)

- AMAR -
DAILY SYSTEM UTILIZATION SUMMARY REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

Figure 1-2



D

SUMMARY OF KEY UTILIZATION ITEMS

-------AVERAGE-----	# ACTIVE FORKS	# MEMORY WRK SETS	# FORKS BS WAIT	BAL SET ADJS/SEC	# LINES IN USE	# FORK WAKS/SEC	% IDLE TIME	% USED TIME	% BACK GND TIME	% SCHED TIME	NO. OF HOURS KEY ITEMS OVER LIMITS
----PRIME TIME-----	6	159	1	.9	55	10	8	73	5	10	30
---NON-PRIME TIME--	1	40	0	.7	6	3	30	50	4	4	7

E

-------AVERAGE-----	% PAGING TIME	PG FAULT TRPS/SEC	CONTEXT SWTS/SEC	SWAP RDS PGS/SEC	SWAP WRS PGS/SEC	DSK RDS PGS/SEC	DSK WRS PGS/SEC	# MTAS ASSIGNED	# FREE MEM PGS	PGS USER MEMORY	NO. OF HOURS KEY ITEMS OVER LIMITS
----PRIME TIME-----	7	58	41	3	2	22	13	1	596	4542	13
---NON-PRIME TIME--	5	44	19	0	1	20	11	1	3275	4542	7

+ = MORE THAN YESTERDAY - = LESS THAN YESTERDAY

---------- CONTINUED NEXT PAGE ---------

DATE: 26-OCT-82 (TUESDAY)

- AMAR -
DAILY SYSTEM UTILIZATION SUMMARY REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

PERCENT FREE SPACE LEFT ON DISK PACKS

-----PACK NAME-----	ARD	ARDBAK	DSKR	DSKT	DSKW	IRA	MAP	PS	TEST3	USRT
---% FREE TODAY----	17	28	55	21	18	29	44	31	38	26
-% FREE YESTERDAY--	24	12	48	21	19	30	44	23	39	29
----DIFFERENCE-----	- 7	+16	+ 7	+ 0	- 1	- 1	+ 0	+ 8	- 1	- 3

(F) (G) (H)

Figure 1-2 (continued)

+ = MORE THAN YESTERDAY - = LESS THAN YESTERDAY PACK DISMOUNTED ( ) UNMEASURED (__) WRONG DAY (?) BAD FILE (!)

DATE: 26-OCT-82 (TUESDAY)

- AMAR -
DAILY SYSTEM UTILIZATION PROBLEM REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

(I)

PROBLEM PERIODS

EACH * = 1 KEY ITEM OVER WATCHDOG LIMIT
EACH + = 1 OTHER ITEM OVER LIMIT

(SEE THE FIRST PAGE OF DETAIL REPORT)
(FOR HOURLY AVERAGES OF KEY ITEMS)

(K)

PROBLEM RESOURCES

EACH P = 1 PRIME HOUR WHEN THE ITEM WAS OVER THE WATCHDOG LIMIT
EACH N = 1 NON-PRIME HOUR WHEN THE ITEM WAS OVER LIMIT

-----HOUR----	------NO. OF ITEMS------ (J)
00:00 - 01:00	*
01:00 - 02:00	
02:00 - 03:00	
03:00 - 04:00	
04:00 - 05:00	***
05:00 - 06:00	
06:00 - 07:00	
07:00 - 08:00	+
08:00 - 09:00	*****
09:00 - 10:00	**********
10:00 - 11:00	**********
11:00 - 12:00	**********
12:00 - 13:00	****
13:00 - 14:00	***
14:00 - 15:00	****
15:00 - 16:00	*******
16:00 - 17:00	****
17:00 - 18:00	******
18:00 - 19:00	+
19:00 - 20:00	++
20:00 - 21:00	*
21:00 - 22:00	**
22:00 - 23:00	*
23:00 - 24:00	

--------ITEM--------	----NO. OF HOURS-------- (L)
# FORKS BS WAIT	PPP
BAL SET ADJS/SEC	PPPPN
# LINES IN USE	PPPP
% IDLE TIME	PPPPPPNNNNN
% BACK GND TIME	PPPPPPPP
% SCHED TIME	PPPPPN
% PAGING TIME	PP
PG FAULT TRPS/SEC	PPPPPPNNNNN
CONTEXT SWTS/SEC	PPPPPN
DSK RDS PGS/SEC	N
TTY OUT CHRS/SEC	PPPPPP
# WORK SET PGS	PPPPPPPPPNNNN
PS#O RDS/SEC	P
PS#O WRS/SEC	P

---------- COMMENTS ---------- (M)	
SERIOUS	CPU BOTTLENECK OR SCHEDULER SLOW
SERIOUS	SCHEDULER THRASHING: FIND REASON
SERIOUS	TTY LINE USE HIGHER THAN PLANNED
CRITICAL	CPU PRESSED: CHK WORKLD DATA FIRST
CRITICAL	OVERHEAD TOO HIGH: INVESTIGATE
CRITICAL	SCHEDULER PRESSED: FIND CAUSE
SERIOUS	PAGING TOO OFTEN: CHK WORKLD DATA
CRITICAL	PAGING TOO OFTEN: CHK WORKLD DATA
CRITICAL	SCHEDULER TOO FAST: CHK BLOCKING
WARNING	DISK READ RATE HIGH: CHK PACK I/O
CRITICAL	TTY CHAR RATE TOO HIGH: FIND CAUSE
CRITICAL	PGM SIZES LARGER THAN PLANNED
WARNING	I/O RATE HIGH: CHK FOR CONTENTION
WARNING	I/O RATE HIGH: CHK FOR CONTENTION

(N)

SYSTEM AVAILABILITY SUMMARY

	SYSTEM UPTIME (P)	% TIME MEASURED (Q)	RELOADS (R)	PERIODS OF DOWNTIME (S)
PRIME TIME :	100.0%	100.0%		
NON-PRIME TIME:	100.0%	100.0%		

Figure 1-2 (continued)

DATE: 26-OCT-82 (TUESDAY)

- AMAR -
DAILY SYSTEM UTILIZATION DETAIL REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

(T) PERIOD	KEY UTILIZATION ITEMS: # ACTIVE FORKS	# MEMORY WRK SETS	# FORKS BS WAIT	BAL SET ADJS/SEC	# LINES IN USE	# FORK WAKS/SEC	% IDLE TIME	% USED TIME	% BACK GND TIME	% SCHED TIME (BB)	# ITEMS OVER LIMITS: THIS PAGE	ALL PAGES (AA)
00:00 - 01:00	1	34	0	.9	2	1	6 *	76	4	4	1	1
01:00 - 02:00	0	34	0	.3	2	2	64	26	4	2		
02:00 - 03:00	1	34	0	.7	2	1	27	51	4	4		
03:00 - 04:00	1	33	0	.5	2	1	43	38	4	3		
04:00 - 05:00	1	33	0	1.0 *	2	2	0 *	74	4	4	2	3
05:00 - 06:00	1	32	0	.4	2	2	55	26	4	3		
06:00 - 07:00	0	33	0	.0	3	2	92	3	4	0		
07:00 - 08:00	1	66	0	.4	14	5	46	35	4	3		1
08:00 - 09:00	8	157	1	.9	47	11	5 *	76	4	10 *	2	5
09:00 - 10:00	14 (U)	175	2 *	1.0 *	68 *	13	0 * (V)	83	5 *	13 *	6	10
10:00 - 11:00	8	176	2 *	1.0 *	63 *	10	0 *	82	5 *	12 *	6	11
11:00 - 12:00	11	172	2 *	1.0 *	68 *	9	0 *	83	5 *	12 *	6	12
12:00 - 13:00	3	130	1	.8	45	5	12	68	5 *	7	1	4
13:00 - 14:00	2	168	1	.6	50	11	26	48	5 *	8	1	3
14:00 - 15:00	2	172	1	.6	63 *	11	21	54	5 *	8	2	4
15:00 - 16:00	3	158	1	1.0 *	52	9	0 *	85	5 *	10 *	4	7
16:00 - 17:00	2	127	1	.9	35	7	8 *	79	5 *	7	2	4
17:00 - 18:00	2	60	1	.9	13	5	9 *	63	4	11 *	2	6
18:00 - 19:00	1	55	0	.6	12	6	35	36	4	5		1
19:00 - 20:00	1	45	0	.6	10	6	29	48	4	6		2
20:00 - 21:00	1	36	0	.8	6	3	15	71	4	4		1
21:00 - 22:00	1	35	0	.9	5	3	4 *	75	4	5	1	2
22:00 - 23:00	1	37	0	.9	5	3	6 *	75	4	5	1	1
23:00 - 24:00	1	33	0	.8	4	2	20	58	4	5		
----PRIME TIME-----												
AVERAGE VALUE:	6	159	1	.9	55	10	8 (W)	73	5	10		
SHORT TERM LIMIT:	>30	NONE	>2	>1.0	>60	>20	<10%	>90% (X)	>5%	>10%		
% TIME OVER LIMIT:			40.1%	75.2%	50.2%		78.5% (Y)		96.6%	48.9%		
# HOURS OVER LIMIT:			3	4	4		6 (Z)		8	5	30	60
--NON-PRIME TIME---												
AVERAGE VALUE:	1	40	0	.7	6	3	30	50	4	4		
SHORT TERM LIMIT:	>30	NONE	>2	>1.0	>60	>20	<10%	>90%	>5%	>10%		
% TIME OVER LIMIT:				61.4%			61.9%			5.3%		
# HOURS OVER LIMIT:				1			5			1	7	18

* = OVER LIMITS > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE --------

Figure 1-2 (continued)

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

PERIOD	% PAGING TIME	PG FAULT TRPS/SEC	CONTEXT SWTS/SEC	SWAP RDS PGS/SEC	SWAP WRS PGS/SEC	DSK RDS PGS/SEC	DSK WRS PGS/SEC	# MTAS ASSIGNED	# FREE MEM PGS	PGS USER MEMORY	# ITEMS OVER LIMITS THIS PAGE	# ITEMS OVER LIMITS ALL PAGES
	KEY UTILIZATION ITEMS											
00:00 - 01:00	5	32	17	0	1	18	14	0	3467	4542		1
01:00 - 02:00	2	16	8	0	0	8	4	0	3478	4542		
02:00 - 03:00	5	37	17	0	0	23	14	1	3471	4542		
03:00 - 04:00	8	46	12	0	0	24	23	0	3507	4542		
04:00 - 05:00	9	68 *	23	0	0	23	15	0	3499	4542	1	3
05:00 - 06:00	3	30	16	0	0	24	6	1	3532	4542		
06:00 - 07:00	0	1	2	0	0	0	0	0	3486	4542		
07:00 - 08:00	2	28	19	0	1	8	1	0	2535	4542		1
08:00 - 09:00	4	34	43 *	2	2	17	4	0	703	4542	1	5
09:00 - 10:00	7	51 *	54 *	5	4	28	14	1	171	4542	2	10
10:00 - 11:00	11 *	70 *	51 *	4	3	39	25	1	191	4542	3	11
11:00 - 12:00	9	60 *	51 *	4	3	35	19	2	180	4542	2	12
12:00 - 13:00	12 *	69 *	32	2	1	37	29	1	1099	4542	2	4
13:00 - 14:00	2	22	32	1	2	7	8	1	640	4542		3
14:00 - 15:00	4	41	38	3	2	16	7	1	199	4542		4
15:00 - 16:00	7	87 *	40 *	2	2	10	7	2	666	4542	2	7
16:00 - 17:00	7	87 *	30	1	1	12	8	2	1517	4542	1	4
17:00 - 18:00	4	59 *	47 *	1	1	49 *	5	2	2831	4542	3	6
18:00 - 19:00	2	29	25	0	1	19	5	1	2922	4542		1
19:00 - 20:00	9	56 *	27	0	1	33	20	1	3068	4542	1	2
20:00 - 21:00	7	104 *	18	0	1	10	6	1	3273	4542	1	1
21:00 - 22:00	9	76 *	19	0	1	19	22	0	3313	4542	1	2
22:00 - 23:00	5	35	20	0	1	18	15	1	3286	4542		1
23:00 - 24:00	5	37	19	0	0	23	12	1	3455	4542		
----PRIME TIME-----												
AVERAGE VALUE:	7	58	41	3	2	22	13	1	596	4542		
SHORT TERM LIMIT:	>10%	>50	>40	>40	>40	>40	>40	NONE	<25	<768		
% TIME OVER LIMIT:	21.6%	54.9%	58.4%			16.6%						
# HOURS OVER LIMIT:	2	6	5								13	60
--NON-PRIME TIME---												
AVERAGE VALUE:	5	44	19	0	1	20	11	1	3275	4542		
SHORT TERM LIMIT:	>10%	>50	>40	>40	>40	>40	>40	NONE	<25	<768		
% TIME OVER LIMIT:		36.7%	9.6%			18.3%						
# HOURS OVER LIMIT:		5	1			1					7	18

* = OVER LIMITS > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

CONTINUED NEXT PAGE

Figure 1-2 (continued)

DATE: 26-OCT-82 (TUESDAY)

- AMAR -
DAILY SYSTEM UTILIZATION DETAIL REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS   PRIME TIME: 0800 - 1700

------------------------------OTHER UTILIZATION ITEMS --

PERIOD	# WORK SET PGS	TTY OUT CHRS/SEC	PS#0 RDS/SEC	PS#0 WRS/SEC	# ITEMS OVER LIMITS THIS PAGE	# ITEMS OVER LIMITS ALL PAGES
00:00 - 01:00	1286	45	1	0		
01:00 - 02:00	1260	50	0	0		1
02:00 - 03:00	1274	45	0	0		
03:00 - 04:00	1222	46	0	0		
04:00 - 05:00	1231	48	0	0		
05:00 - 06:00	1210	52	0	0		3
06:00 - 07:00	1280	134	0	0		
07:00 - 08:00	2646 *	338	0	0		
08:00 - 09:00	5289 *	977 *	2	1	1	1
09:00 - 10:00	5769 *	931 *	7	16	2	5
10:00 - 11:00	5720 *	726 *	3	2	2	10
11:00 - 12:00	5589 *	633 *	45 *	55 *	2	11
12:00 - 13:00	4545 *	392	1	1	4	12
13:00 - 14:00	5254 *	539 *	1	1	1	4
14:00 - 15:00	5680 *	595 *	2	2	2	3
15:00 - 16:00	5155 *	426	1	2	2	4
16:00 - 17:00	3994 *	297	1	1	1	7
17:00 - 18:00	2136 *	246	2	2	1	4
18:00 - 19:00	1966 *	275	0	1	1	6
19:00 - 20:00	1812 *	147	0	2	1	1
20:00 - 21:00	1476	93	0	1	1	2
21:00 - 22:00	1387	68	0	0		1
22:00 - 23:00	1410	75	0	1		2
23:00 - 24:00	1224	58	0	0		1

---PRIME TIME----

AVERAGE VALUE:	5222	613	7	9		
SHORT TERM LIMIT:	>1500	>500	>40	>40		
% TIME OVER LIMIT:	100.0%	61.2%	.6%	.6%		
# HOURS OVER LIMIT:	9	6	1	1	17	60

--NON-PRIME TIME---

AVERAGE VALUE:	1521	115	0	1		
SHORT TERM LIMIT:	>1500	>500	>40	>40		
% TIME OVER LIMIT:	30.5%					
# HOURS OVER LIMIT:	4				4	18

* = OVER LIMITS   > = GREATER THAN OR EQUAL TO   < = LESS THAN OR EQUAL TO

Figure 1-2 (continued)

1.3.1.2 Weekly Utilization Report -

The Weekly Utilization Report provides the following features:

A graph which shows the average CPU utilization and overhead for each 2 hour period for each day in the week. Overhead is defined under Summary Report below.

A prime/non-prime time summary for the week and each day in the week of utilization and performance items such as used time, background time, swapping rate, disk I/O rate, system uptime, etc.

A prime/non-prime time summary of key items and problem items during the week.

A summary of problem days in the week.

The Weekly Utilization Report is separated into three sections - a Summary Report, a Detail Report, and a Problem Report. Refer to Figure 1-3. The format of the report header is the same as for the Daily System Utilization Report.

Summary Report (Pages 1) - This report section gives management a quick overview of system utilization and peak processing periods during the previous week. It is often used in conjunction with the Workload AMAR "Weekly Report by Program" or the "Weekly Report by User."

The graph at the top of page 1 (A) is a bar chart which shows CPU utilization (*) and overhead (#) for each 2 hour period in the day starting at midnight. The difference between 100% and the CPU utilization (B) line represents idle time plus file wait time. See % IDLE TIME and % IDLE IO TIME in the Appendix called "System AMAR Item Definitions." The overhead (C) line represents scheduler and background time (see % SCHED TIME and % BACKGND TIME.)

Below the graph are prime and non-prime time averages for twenty-one key items (D). These averages are intended to give a quick picture of the usage during the week. You have the option of selecting key items from the entire list of items that AMAR collects. The items and their definitions are listed in the Appendix called "System AMAR Item Definitions". In addition to key items, the percent of system uptime (E), the percent of time that AMAR measured the system (F), and the total number of system reloads (G) during the week is presented.

Detail Report (Pages 2-3) - This report section lets you compare the total resource usage for the week with the resource usage of each of the days in the week. This report section, in conjunction with the Problem Report, allows you to track back through the week to determine on which days the most resource consumption and/or problems occurred. You can then refer to the Daily Utilization Summary Report for more detail. Averages for

Key utilization items are presented first, followed by averages for other utilization items. Other utilization items are those whose averages have exceeded the long term limit as listed on the Problem Report and as defined in the xxxxDR.RFD Report File Description. Each average exceeding the limit is flagged by an asterisk (*) (H). The weekly average is given first (I) followed by the daily averages (J). Prime time data for both key and other items is given before non-prime time data.

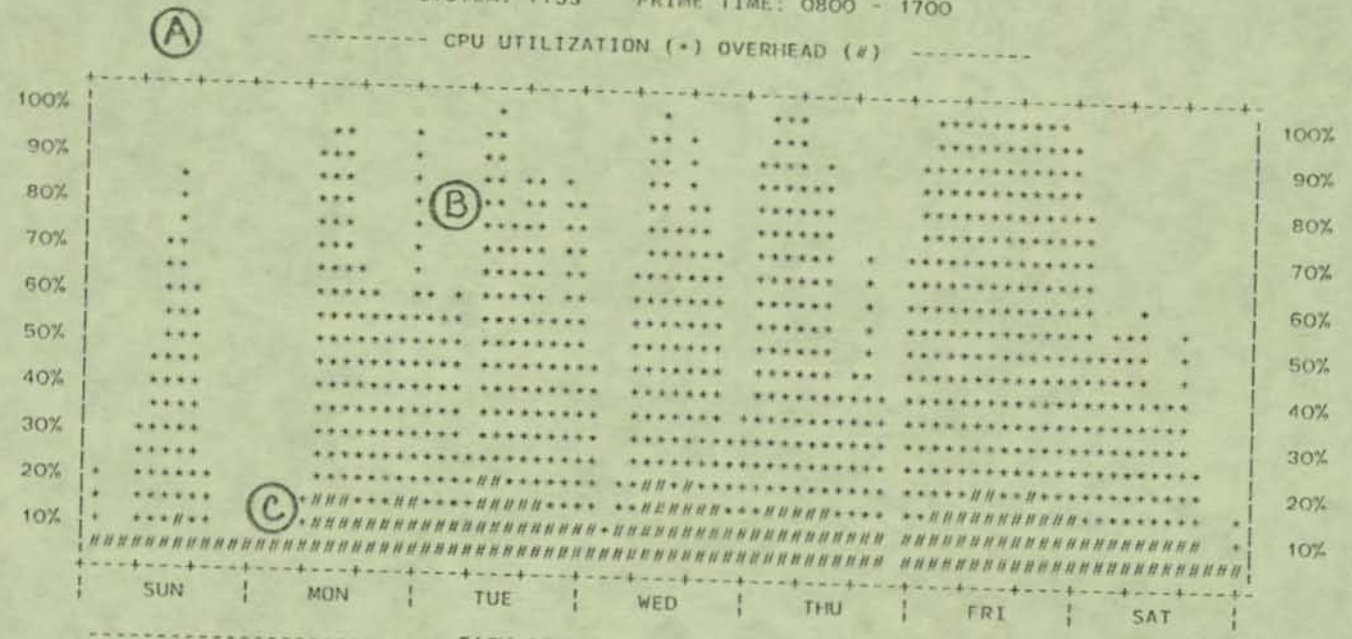
Problem Report (Page 4) - This report section gives you an overview of problem items and problem periods. Only those items whose averages (K) have exceeded the long term limit or whose values have exceeded the long term limit more than 10% of the time (L), are reported upon. The average values for both prime and non-prime time (M), the long term limit (N) and the percent of time over the limit (P) are all given followed by a comment line (Q), intended to point you to an area for further investigation. As in the Daily System Utilization Problem Report, the comments are intended only as a pointer to a possible problem area or to recommend some further course of action. There may be more than one possible cause for a given problem (for example, such as too much background time) and there is often more than one solution which can be applied. The alternatives must always be evaluated carefully. The comments, in and of themselves, are not intended to recommend a solution to a given problem. Comments are defined in the xxxxDR.RFD file. You may modify or delete them.

In addition to reporting which items had problems over the past week, this section also reports which days experienced the most time in a problem state. Again the item is given along with the percent of time that the item's values were over the long term limit during each day (R) in the week. The prime time summary is given before the non-prime time summary.

FROM: 24-OCT-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
WEEKLY UTILIZATION SUMMARY REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700



D — SUMMARY OF KEY UTILIZATION ITEMS

-------AVERAGE-----	% CPU UTIL	# ACTIVE FORKS	# MEMORY WRK SETS	# FORKS BS WAIT	BAL SET ADJS/SEC	# LINES IN USE	# FORK WAKS/SEC	% IDLE TIME	% USED TIME	% BACK GND TIME
----PRIME TIME-----	83%*	5	152	2 *	.8	50	10	11%	70%	3%
---NON-PRIME TIME--	50%	1	40	0	.6	3	2	40%	43%	3%

-------AVERAGE-----	% SCHED TIME	% PAGING TIME	PG FAULT TRPS/SEC	CONTEXT SWTS/SEC	SWAP RDS PGS/SEC	SWAP WRS PGS/SEC	DSK RDS PGS/SEC	DSK WRS PGS/SEC	# MTAS ASSIGNED	# FREE MEM PGS
----PRIME TIME-----	9%	5%	38	41 *	3	2	18	8	1	507
---NON-PRIME TIME--	4%	4%	32	17	0	0	15	7	1	3320

-------AVERAGE-----	PGS USER MEMORY	% SYSTEM UPTIME	% AMAR CLK TIME	# SYSTEM RELOADS
----PRIME TIME-----	4292	99.8%	98.2%	2
---NON-PRIME TIME--	4516	97.0%	96.9%	1

E F G

Figure 1-3

FROM: 24-OCT-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
WEEKLY UTILIZATION DETAIL REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

--- PRIME TIME ---

KEY UTILIZATION ITEMS		AVERAGE OF CURRENT -WEEK-	-SUN-	-MON-	-TUE-	-WED-	-THU-	-FRI-	-SAT-
		(I)				DAILY AVERAGE			(J)
% CPU	UTIL	83%*		79%*	87%*	88%*	59%	99%*	
# ACTIVE	FORKS	5		5	6	6	2	4	
# MEMORY	WRK SETS	152		172	159	164	101	159	
# FORKS	BS WAIT	2 *		2 *	1	3 *	1	1	
BAL SET	ADJS/SEC	.8		.8	.9	.9	.5	1.0 *	
# LINES	IN USE	50		55	55	53	41	47	
# FORK	WAKS/SEC	10		12	10	10	9	10	
% IDLE	TIME	11%		14%	8%*	5%*	32%	0%*	
% USED	TIME	70%		68%	73%	74%	49%	85%	
% BACK	GND TIME	3%		2%	5%*	5%*	2%	4%	
% SCHED	TIME	9%		9%	10%*	10%*	6%	11%*	
% PAGING	TIME	5%		6%	7%	5%	3%	4%	
PG FAULT	TRPS/SEC	38		42	58 *	36	23	30	
CONTEXT	SWTS/SEC	41 *		44 *	41 *	44 *	31	41 *	
SWAP RDS	PGS/SEC	3		3	3	3	3	2	
SWAP WRS	PGS/SEC	2		2	2	2	2	2	
DSK RDS	PGS/SEC	18		24	22	19	10	13	
DSK WRS	PGS/SEC	8		10	13	8	4	5	
# MTAS	ASSIGNED	1		0	1	1	0	0	
# FREE	MEM PGS	507		499	596	472	354	600	
PGS USER	MEMORY	4292		4542	4542	4542	3195	4542	
% SYSTEM	UPTIME	99.8%		100.0%	100.0%	100.0%	99.2%	100.0%	
% AMAR	CLK TIME	98.2%		100.0%	100.0%	100.0%	90.9%	100.0%	
# SYSTEM	RELOADS	2		0	0	0	2	0	

--- PRIME TIME ---

OTHER UTILIZATION ITEMS		AVERAGE OF CURRENT -WEEK-	-SUN-	-MON-	-TUE-	-WED-	-THU-	-FRI-	-SAT-
						DAILY AVERAGE			
# WORK	SET PGS	4922 *		5190 *	5222 *	5260 *	3555 *	5254 *	
TTY OUT	CHRS/SEC	663 *		725 *	613 *	608 *	574 *	788 *	

* = OVER LONG TERM LIMITS > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO ---------- CONTINUED NEXT PAGE --------

Figure 1-3 (continued)

FROM: 24-OCT-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
WEEKLY UTILIZATION DETAIL REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

--- NON-PRIME TIME ---

KEY UTILIZATION ITEMS		AVERAGE OF CURRENT -WEEK-	DAILY AVERAGE -SUN-	-MON-	-TUE-	-WED-	-THU-	-FRI-	-SAT-
% CPU	UTIL	50%	27%	34%	59%	39%	87%*	97%*	38%
# ACTIVE	FORKS	1	0	1	1	1	2	2	1
# MEMORY	WRK SETS	40	31	36	40	40	46	58	38
# FORKS	BS WAIT	0	0	0	0	0	1	1	0
BAL SET	ADJS/SEC	.6	.4	.4	.6	.4	.9	.9	.5
# LINES	IN USE	3	1	3	6	6	4	3	2
# FORK	WAKS/SEC	2	2	2	3	3	3	2	1
% IDLE	TIME	40%	60%	57%	30%	52%	8%*	1%*	48%
% USED	TIME	43%	25%	28%	50%	31%	77%	85%	32%
% BACK	GND TIME	3%	1%	2%	4%	4%	3%	4%	4%
% SCHED	TIME	4%	2%	4%	4%	4%	7%	8%	2%
% PAGING	TIME	4%	2%	3%	5%	3%	5%	5%	3%
PG FAULT	TRPS/SEC	32	23	30	44	31	43	49	22
CONTEXT	SWTS/SEC	17	11	15	19	17	24	26	12
SWAP RDS	PGS/SEC	0	0	0	0	0	0	0	0
SWAP WRS	PGS/SEC	0	0	0	1	0	1	1	0
DSK RDS	PGS/SEC	15	10	14	20	16	18	20	11
DSK WRS	PGS/SEC	7	4	6	11	7	12	10	5
# MTAS	ASSIGNED	1	1	1	1	1	1	1	1
# FREE	MEM PGS	3320	3566	3440	3275	3289	2902	3129	3358
PGS USER	MEMORY	4516	4542	4542	4542	4542	4268	4542	4542
% SYSTEM	UPTIME	97.0%	100.0%	100.0%	100.0%	100.0%	75.7%	100.0%	100.0%
% AMAR	CLK TIME	96.9%	100.0%	100.0%	100.0%	100.0%	74.5%	100.0%	100.0%
# SYSTEM	RELOADS	1	0	0	0	0	1	0	0

--- NON-PRIME TIME ---

OTHER UTILIZATION ITEMS		AVERAGE OF CURRENT -WEEK-	DAILY AVERAGE -SUN-	-MON-	-TUE-	-WED-	-THU-	-FRI-	-SAT-
# WORK	SET PGS	1415	1118	1322	1521 *	1492	1658 *	1693 *	1370
%ARDBAKO	FREE SPC	18%	2%*	11%	23%	25%	25%	25%	25%
%DSKW#O	FREE SPC	20%	12%	20%	17%	17%	16%	28%	27%

* = OVER LONG TERM LIMITS > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

Figure 1-3 (continued)

FROM: 24-OCT-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
WEEKLY UTILIZATION PROBLEM REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

SUMMARY OF PROBLEM RESOURCES

ITEM		--- PRIME TIME --- (M) AVG. VALUE	(N) LONG TERM LIMIT	(P) % TIME OVER LIMIT	--- NON-PRIME TIME --- (M) AVG. VALUE	(N) LONG TERM LIMIT	(P) % TIME OVER LIMIT	(Q) COMMENTS
BAL SET	ADJS/SEC	.8	>1.0	73.7%	.6	>1.0	54.7%	SCHEDULER THRASHING: FIND REASON
% BACK	GND TIME	3%	>5%	58.6%	3%	>5%	1.5%	OVERHEAD TOO HIGH: INVESTIGATE
# FORKS	BS WAIT	2	>2	34.5%	0	>2	1.3%	CPU BOTTLENECK OR SCHEDULER SLOW
CONTEXT	SWTS/SEC	41	>40	56.5%	17	>40	7.3%	SCHEDULER TOO FAST: CHK BLOCKING
DSK RDS	PGS/SEC	18	>40	6.9%	15	>40	11.3%	DISK READ RATE HIGH: CHK PACK I/O
% IDLE	TIME	11%	<10%	75.4% (L)	40%	<10%	55.0%	CPU PRESSED: CHK WORKLD DATA FIRST
%ARDBAKO	FREE SPC	25%	<10%	8.6%	18%	<10%	26.8%	DELETE UNNECESSARY FILES
%DSKW#O	FREE SPC	21%	<10%	.0%	20%	<10%	12.0%	DELETE UNNECESSARY FILES
PG FAULT	TRPS/SEC	38	>50	22.7%	32	>50	23.5%	PAGING TOO OFTEN: CHK WORKLD DATA
% SCHED	TIME	9%	>10%	49.8%	4%	>10%	7.0%	SCHEDULER PRESSED: FIND CAUSE
TTY OUT	CHRS/SEC	663	>500	68.6%	133	>500	7.1%	TTY CHAR RATE TOO HIGH: FIND CAUSE
# LINES	IN USE	50	>60	31.3%	3	>60	.0%	TTY LINE USE HIGHER THAN PLANNED
# WORK	SET PGS	4922	>1500	100.0%	1415	>1500	29.9%	PGM SIZES LARGER THAN PLANNED
% CPU	UTIL	83% (K)	>70%	74.7%	50%	>70%	41.8%	CPU PRESSED: CHK WORKLD DATA FIRST

SUMMARY OF PROBLEM PERIODS

--- PRIME TIME ---

ITEM		LONG TERM LIMIT	% TIME OVER LIMIT SUN	MON	TUE	WED	THU	FRI	SAT
BAL SET	ADJS/SEC	>1.0						100.0%	
% BACK	GND TIME	>5%			96.6%	92.5%			
# FORKS	BS WAIT	>2		62.9%		27.8%			
CONTEXT	SWTS/SEC	>40		65.5%	58.4%	69.4%		56.1%	
% IDLE	TIME	<10%			78.5%	87.3% (R)		100.0%	
PG FAULT	TRPS/SEC	>50			54.9%				
% SCHED	TIME	>10%			48.5%	62.3%		66.5%	
TTY OUT	CHRS/SEC	>500		76.5%	61.2%	56.7%	53.9%	93.5%	
# WORK	SET PGS	>1500		100.0%	100.0%	100.0%	100.0%	100.0%	
% CPU	UTIL	>70%		68.5%	79.5%	83.0%		100.0%	

SUMMARY OF PROBLEM PERIODS

--- NON-PRIME TIME ---

ITEM		LONG TERM LIMIT	% TIME OVER LIMIT SUN	MON	TUE	WED	THU	FRI	SAT
% IDLE	TIME	<10%					88.6%	99.2%	
%ARDBAKO	FREE SPC	<10%	100.0%						
# WORK	SET PGS	>1500			30.5%		55.9%	74.0%	
% CPU	UTIL	>70%					86.3%	97.4%	

* = OVER LONG TERM LIMITS > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

Figure 1-3 (continued)

1.3.1.3 Monthly Utilization Report -

The Monthly Utilization Report provides the following features:

A graph which shows the average CPU utilization and overhead for each day in the month split out by prime and non-prime time. Overhead is defined under Summary Report below.

A prime/non-prime time summary for the month and each week in the month of utilization and performance items such as idle time, background time, scheduler time, swapping rates, disk I/O rates, system uptime, etc.

A prime/ non-prime time summary of key items and problem items during the month.

The Monthly Utilization Report is separated into three sections - a Summary Report, a Detail Report, and a Problem Report. Refer to Figure 1-4. The format of the report header is the same as for the Daily System Utilization Report.

Summary Report (Page 1) - This report section gives management a quick overview of system utilization and peak processing periods during the previous month. It is often used in conjunction with the Workload AMAR "Monthly Report by User", or the "Monthly Report by Program". The graph at the top of page 1 (A) represents processor 0. It is a bar chart which shows CPU utilization (*) and overhead (#) for prime and non-prime time periods for each day in the month. The first day typically printed on the prime time graph will be a Monday (B), the second day in the report period (4-OCT-82 in this example). To help you locate other days in the month, a fiscal week ending date is printed followed by the symbol "¦" (C) which points to that date on the graph.

The difference between 100% and the CPU utilization (D) line represents idle time plus file wait time. See % IDLE TIME and % IDLE IO TIME in the Appendix called "System AMAR Item Definitions." The overhead (E) line represents scheduler and background time (see % SCHED TIME and % BACKGND TIME.) Bars approaching 100% indicate days when the CPU was heavily loaded. These days should be examined more closely by looking at the Daily System Utilization Report.

Below the graph are prime and non-prime time averages for twenty-one key items (F). These averages are intended to give a quick picture of the previous month's usage. You have the option of selecting key items from the entire list of items that AMAR collects. The items and their definitions are listed in the Appendix called "System AMAR Item Definitions". In addition to key items, the percent of system uptime (G), the percent of time that AMAR measured the system (H), and the total number of system reloads (I) during the past month is presented.

Detail Report (Pages 2-3) - This report section enables you to

compare the total resource usage for the month with the resource usage during each of the weeks in the month. This report section, in conjunction with the Problem Report, allows you to trace back through the month to determine during which weeks the most resource consumption and/or problems occurred. You can then refer to the Weekly and Daily System Utilization Reports to trace back and find the problem days and hours.

Averages for key utilization items are presented first, followed by averages for other utilization items. Other utilization items are those whose averages have exceeded the long term limit as listed on the Problem Report and as defined in the xxxxDR.RFD Report File Description. Each average exceeding the limit is flagged by an asterisk (*) (J). The monthly average is given first (K) followed by the weekly averages (L). Prime time data for both key and other items is given before non-prime time data.

Problem Report (Page 4) - This section, like that of the Daily System and Weekly Utilization Problem Reports, gives you an overview of problem resources and problem periods. Only those items whose averages (M) have exceeded the long term limit, or whose sample values (N) have exceeded the long term limit more that 10% of the time, are reported upon. The average values for both prime and non-prime time (P), the long term limit (Q), and the percent of time over limit (R) are all given followed by a comment line (S). The comment line is intended to point you to an area for further investigation. As in the Daily System and Weekly Utilization Problem Reports, the comments are intended only as a pointer to a possible problem area or to recommend some further course of action. The comments are not intended to suggest a solution to a problem. Since there may be more than one reason for an item exceeding limits and more than one solution to a given problem, careful analysis and weighing of alternatives is recommended. Comments are defined in the xxxxDR.RFD file. You may modify or delete them.

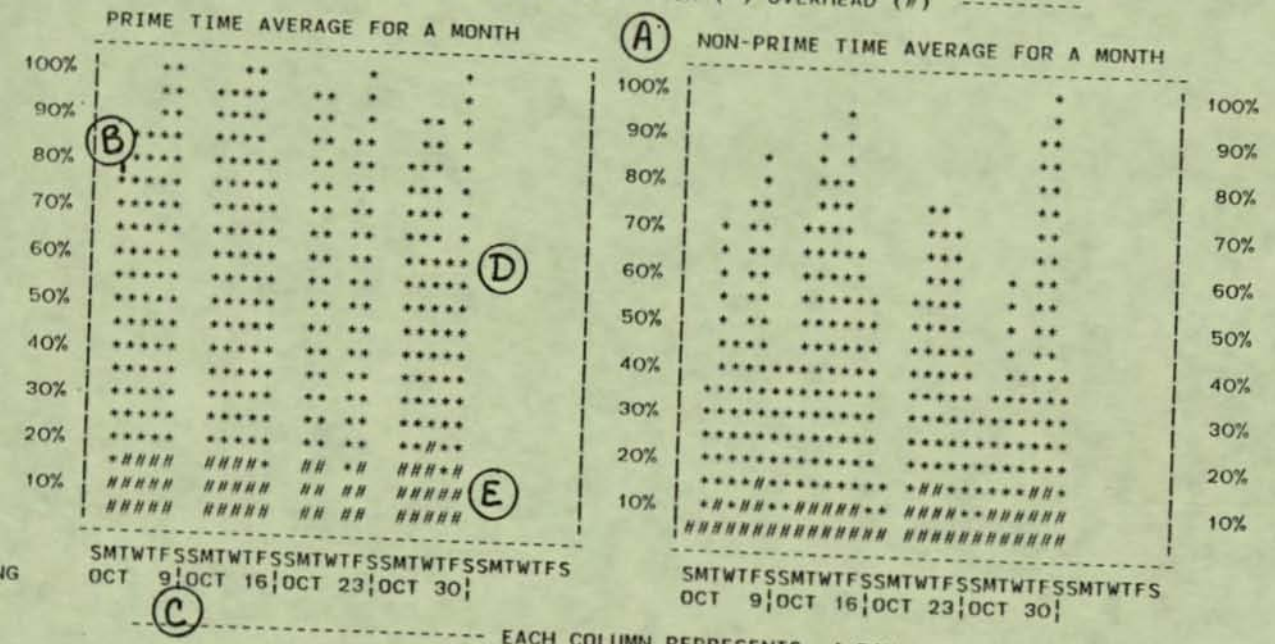
In addition to reporting which items had problems over the past month, this section also reports during which weeks the items spent the most time in a problem state. Again the item is given along with the percent of time that the item's values were over the long term limit during each week (T) in the past month. The prime time summary is given before the non-prime time summary.

FROM: 03-OCT-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
MONTHLY UTILIZATION SUMMARY REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

--------- CPU UTILIZATION (*) OVERHEAD (#) ---------



---------- EACH COLUMN REPRESENTS 1 DAY ----------

---------- SUMMARY OF KEY UTILIZATION ITEMS ----------

F

-------AVERAGE-----	% CPU UTIL	# ACTIVE FORKS	# MEMORY WRK SETS	# FORKS BS WAIT	BAL SET ADJS/SEC	# LINES IN USE	# FORK WAKS/SEC	% IDLE TIME	% USED TIME	% BACK GND TIME
----PRIME TIME-----	88%*	8	152	2 *	.9	48	10	7%*	76%	3%
---NON-PRIME TIME--	51%	1	44	0	.6	3	3	38%	45%	2%

-------AVERAGE-----	% SCHED TIME	% PAGING TIME	PG FAULT TRPS/SEC	CONTEXT SWTS/SEC	SWAP RDS PGS/SEC	SWAP WRS PGS/SEC	DSK RDS PGS/SEC	DSK WRS PGS/SEC	# MTAS ASSIGNED	# FREE MEM PGS
----PRIME TIME-----	10%*	6%	47	44 *	3	2	23	10	1	673
---NON-PRIME TIME--	4%	4%	41	20	0	0	18	8	1	3257

-------AVERAGE-----	PGS USER MEMORY	% SYSTEM UPTIME	% AMAR CLK TIME	# SYSTEM RELOADS
----PRIME TIME-----	4470	90.8%	85.2%	19
---NON-PRIME TIME--	4483	86.5%	83.2%	39

G H I

Figure 1-4

FROM: 03-OCT-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
MONTHLY UTILIZATION DETAIL REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

(K)

--- PRIME TIME ---

KEY UTILIZATION ITEMS		AVERAGE OF CURRENT -MONTH-	WEEKLY AVERAGE -OCT 9-	-OCT 16-	-OCT 23-	-OCT 30-
% CPU	UTIL	88%* (J)	86%*	93%*	91%*	83%*
# ACTIVE	FORKS	8	7	14	8	5
# MEMORY	WRK SETS	152	159	151	145	152
# FORKS	BS WAIT	2 *	2 *	3 *	1	2 *
BAL SET	ADJS/SEC	.9	.9	.9	.9	.8
# LINES	IN USE	48	51	49	38	50
# FORK	WAKS/SEC	10	10	9	9	10
% IDLE	TIME	7%*	9%*	3%*	5%*	11%
% USED	TIME	76%	74%	80%	79%	70%
% BACK	GND TIME	3%	1%	2%	3%	3%
% SCHED	TIME	10%*	9%	10%*	9%	9%
% PAGING	TIME	6%	6%	7%	6%	5%
PG FAULT	TRPS/SEC	47	49	53 *	50 *	38
CONTEXT	SWTS/SEC	44 *	44 *	49 *	42 *	41 *
SWAP RDS	PGS/SEC	3	3	3	2	3
SWAP WRS	PGS/SEC	2	2	3	2	2
DSK RDS	PGS/SEC	23	23	30	23	18
DSK WRS	PGS/SEC	10	11	11	11	8
# MTAS	ASSIGNED	1	1	2	1	1
# FREE	MEM PGS	673	596	690	1010	507
PGS USER	MEMORY	4470	4542	4542	4542	4292
% SYSTEM	UPTIME	90.8%	94.8%	94.5%	73.9%	99.8%
% AMAR	CLK TIME	85.2%	90.9%	86.4%	65.4%	98.2%
# SYSTEM	RELOADS	19	5	4	8	2

(L)

--- PRIME TIME ---

OTHER UTILIZATION ITEMS		AVERAGE OF CURRENT -MONTH-	WEEKLY AVERAGE -OCT 9-	-OCT 16-	-OCT 23-	-OCT 30-
# BALNCE	SET FRKS	8	7	14	8	5
# WORK	SET PGS	4852 *	5059 *	4837 *	4476 *	4922 *
TTY OUT	CHRS/SEC	589 *	589 *	556 *	520 *	663 *
%ARDBAKO	FREE SPC	26%	25%	18%	35%	25%
%PS#O	FREE SPC	17%	12%	14%	13%	27%

* = OVER LONG TERM LIMITS > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE --------

Figure 1-4 (continued)

FROM: 03-OCT-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
MONTHLY UTILIZATION DETAIL REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

--- NON-PRIME TIME ---

KEY UTILIZATION ITEMS		AVERAGE OF CURRENT -MONTH-	WEEKLY AVERAGE -OCT 9-	-OCT 16-	-OCT 23-	-OCT 30-
% CPU	UTIL	51%	42%	64%	50%	50%
# ACTIVE	FORKS	1	1	1	1	1
# MEMORY	WRK SETS	44	43	46	49	40
# FORKS	BS WAIT	0	0	1	0	0
BAL SET	ADJS/SEC	.6	.5	.7	.6	.6
# LINES	IN USE	3	2	2	3	3
# FORK	WAKS/SEC	3	6	2	3	2
% IDLE	TIME	38%	49%	24%	39%	40%
% USED	TIME	45%	37%	58%	43%	43%
% BACK	GND TIME	2%	1%	2%	2%	3%
% SCHED	TIME	4%	4%	4%	4%	4%
% PAGING	TIME	4%	4%	6%	5%	4%
PG FAULT	TRPS/SEC	41	38	53 *	40	32
CONTEXT	SWTS/SEC	20	20	23	20	17
SWAP RDS	PGS/SEC	0	0	0	0	0
SWAP WRS	PGS/SEC	0	0	1	1	0
DSK RDS	PGS/SEC	18	17	22	17	15
DSK WRS	PGS/SEC	8	8	9	9	7
# MTAS	ASSIGNED	1	1	2	1	1
# FREE	MEM PGS	3257	3389	3271	2939	3320
PGS USER	MEMORY	4483	4542	4542	4259	4516
% SYSTEM	UPTIME	86.5%	92.0%	93.8%	63.1%	97.0%
% AMAR	CLK TIME	83.2%	89.7%	86.0%	60.4%	96.9%
# SYSTEM	RELOADS	39	8	16	14	1

--- NON-PRIME TIME ---

OTHER UTILIZATION ITEMS		AVERAGE OF CURRENT -MONTH-	WEEKLY AVERAGE -OCT 9-	-OCT 16-	-OCT 23-	-OCT 30-
# WORK	SET PGS	1449	1362	1497	1566 *	1415
%ARDBAKO	FREE SPC	20%	17%	25%	19%	18%
%DSKW#O	FREE SPC	27%	42%	26%	17%	20%
%PS#O	FREE SPC	16%	10%*	10%*	17%	25%

* = OVER LONG TERM LIMITS > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

Figure 1-4 (continued)

FROM: 03-OCT-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
MONTHLY UTILIZATION PROBLEM REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

SUMMARY OF PROBLEM RESOURCES

ITEM		--- PRIME TIME --- (P) AVG. VALUE	(Q) LONG TERM LIMIT	(R) % TIME OVER LIMIT	--- NON-PRIME TIME --- (P) AVG. VALUE	(Q) LONG TERM LIMIT	(R) % TIME OVER LIMIT	(S) COMMENTS
BAL SET	ADJS/SEC	.9	>1.0	82.7%	.6	>1.0	57.1%	SCHEDULER THRASHING: FIND REASON
% BACK	GND TIME	3%	>5%	25.8%	2%	>5%	.8%	OVERHEAD TOO HIGH: INVESTIGATE
# FORKS	BS WAIT	2	>2	55.0%	0	>2	2.0%	CPU BOTTLENECK OR SCHEDULER SLOW
CONTEXT	SWTS/SEC	44	>40	65.3%	20	>40	11.1%	SCHEDULER TOO FAST: CHK BLOCKING
DSK RDS	PGS/SEC	23	>40	10.7%	18	>40	14.3%	DISK READ RATE HIGH: CHK PACK I/O
% IDLE	TIME	7%	<10%	84.0%	38%	<10%	57.4%	CPU PRESSED: CHK WORKLD DATA FIRST
%ARDBAKO	FREE SPC	26%	<10%	13.7%	20%	<10%	28.1%	DELETE UNNECESSARY FILES
%DSKW#O	FREE SPC	27%	<10%	4.4%	27%	<10%	10.3%	DELETE UNNECESSARY FILES
%PS#O	FREE SPC	17%	<10%	23.1%	16%	<10%	28.2%	DELETE UNNECESSARY FILES
# BALNCE	SET FRKS	8	>20	11.6%	1	>20	.0%	MANY FORKS: CHK WORKLD DATA
PG FAULT	TRPS/SEC	47	>50	36.1%	41	>50	32.1%	PAGING TOO OFTEN: CHK WORKLD DATA
% SCHED	TIME	10%	>10%	52.1%	4%	>10%	8.2%	SCHEDULER PRESSED: FIND CAUSE
TTY OUT	CHRS/SEC	589	>500	59.9% (N)	89	>500	3.6%	TTY CHAR RATE TOO HIGH: FIND CAUSE
# LINES	IN USE	48	>60	30.1%	3	>60	.0%	TTY LINE USE HIGHER THAN PLANNED
# WORK	SET PGS	4852 (M)	>1500	98.6%	1449	>1500	33.7%	PGM SIZES LARGER THAN PLANNED
% CPU	UTIL	88%	>70%	83.4%	51%	>70%	43.6%	CPU PRESSED: CHK WORKLD DATA FIRST

SUMMARY OF PROBLEM PERIODS

--- PRIME TIME ---

ITEM		LONG TERM LIMIT	(T) % TIME OVER LIMIT OCT 9	OCT 16	OCT 23	OCT 30
# FORKS	BS WAIT	>2	62.6%	79.0%		21.4%
CONTEXT	SWTS/SEC	>40	65.6%	77.3%	62.4%	56.5%
% IDLE	TIME	<10%	77.6%	94.2%	92.5%	
PG FAULT	TRPS/SEC	>50		50.7%	37.0%	
% SCHED	TIME	>10%		65.2%		
TTY OUT	CHRS/SEC	>500	62.6%	55.1%	49.6%	68.6%
# WORK	SET PGS	>1500	100.0%	94.4%	100.0%	100.0%
% CPU	UTIL	>70%	79.7%	93.2%	88.6%	74.7%

SUMMARY OF PROBLEM PERIODS

--- NON-PRIME TIME ---

ITEM		LONG TERM LIMIT	% TIME OVER LIMIT OCT 9	OCT 16	OCT 23	OCT 30
%PS#O	FREE SPC	<10%	40.5%	61.2%		
PG FAULT	TRPS/SEC	>50		43.8%		
# WORK	SET PGS	>1500			42.8%	

* = OVER LONG TERM LIMITS > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

Figure 1-4 (continued)

1.3.2 Trend Analysis Reports

1.3.2.1 Weekly Trend Analysis Report -

The Weekly Trend Analysis Report provides the following features:

Graphs of CPU utilization and overhead for up to the past 13 weeks split out by prime and non-prime time.

A table of prime and non-prime time averages for key items for up to the past 13 weeks. The table allows for quick comparison of item values. The default list of key items in this report is usually longer than the default list of key itmes in the Daily Utilization Summary Report. The list may be changed by editing the xxxxDR.RFD file.

A trend analysis section which shows relative usage for each week plus the growth/month during the period and an indication of whether or not a short term linear trend exists.

The format of the report header (refer to Figure 1-5) is as follows:

Lines 1 and 2 - Specify the beginning and ending dates of the report, the name of the report, and the page number. The report period begins at approximately 00:01 AM of the first date and ends at approximately 23:59 PM of the second date.

Line 3 - Specifies the Data Center Name.

Line 4 - Specifies the 4 character system code and the period of time during the day which is considered prime time.

The graphs at the top of page 1 represent processor 0. They show the CPU utilization (*) and overhead (#) averages for the past 13 weeks. Definitions for CPU utilization and overhead are given in the Section describing the Daily System Utilization Report. Prime time averages are on the left (A) and non-prime on the right (B). At most 13 data points will be plotted for each item. Fewer may be plotted if the database retention period for the weekly granularity records is less than 13. The graphs and the tables which follow are always read from left to right with the most recent week appearing on the right (C). The symbols *, #, and @ represent the actual data point. The dots in between the data points are for visual effect only and do not represent any item values.

By observing the shape and slope of the two graphs, one may obtain an impression of relative utilization of the processor during prime and non-prime time. This will be helpful in determining whether some load balancing of the machine is

required. Processing cycles, such as for monthly financial closings, may also begin to show up. It may also be possible to observe some short term linear trend in usage. However, it is difficult to be sure that a trend exists just from observing the data points. For verification purposes, a trend line is computed by the reporting program and the percent of fit of the data points (TREND LINE FIT) is given starting on page 4 of the sample report.

Below the graph is a table of prime time averages for the past 13 weeks for key utilization items. Key utilization items are defined by you in the xxxxDR.RFD file. You have the option of selecting "key" items from the entire list of items that AMAR measures. The items and their definitions are listed in the Appendix called "System AMAR Item Definitions". The set of key items selected for the Trend Analysis Reports may differ from those selected for any of the other types of reports.

The format for the table header is as follows:

Line 1 (D) - Specifies the fiscal quarter, month and week for which the averages were computed. The earliest week in the period is given first and the most recent week (H) is given last.

Line 2 (E) - Specifies the actual calendar date of the last day in the week (week ending day).

Line 3 (F) - Indicates the distance in time from the most recent week. For example, the week ending SEP 4 (G) was 8 weeks prior to the most recent week listed on the report, OCT 30 (H).

On the left most side of the report are listed the item descriptions. On the right most side (I), are listed the long term limits for each item. The long term limits are specified in the xxxxDR.RFD file and are also user settable. The long term limits apply to daily, weekly and monthly data summary levels. Any average which equals or exceeds the long term limit will be flagged with an asterisk (*) (J). This enables you to immediately spot problem items and periods.

Following the table of prime time averages is the table of non-prime time averages (page 2). Note that there is no table of other items as in the System Utilization Reports. A separate table of other items should not be used for the Trend Analysis Reports.

The actual trend analysis information begins on page 4 of the sample report. Prime time data comes first, followed by non-prime time data. Again, the item description is given on the left followed by the averages for the first week (K) in the reporting period (AUG 7 in this example.) Next comes the TABLE OF RELATIVE USAGE PER WEEK (L). There is one column for each week in the reporting period. The order of the columns corresponds to the order of the weeks in the preceeding tables. The table is

scaled so that 8 represents the average value for the item. The TABLE OF RELATIVE USAGE PER WEEK enables you to get a quick picture of the amount of variance among the weekly averages. For example, one can see that there was very little variation in BAL SET SWPS/SEC (M) from week to week while the values for # SHARED PAGES (N) varied widely. The numbers in the TABLE OF RELATIVE USAGE PER WEEK have meaning only in relation to one another. The numbers from one item cannot be compared with the numbers from another item. For example, an 8 for BAL SET SWPS/SEC has no relationship to an 8 for # SHARED PAGES and, in fact, their actual averages will typically be very different.

Following the TABLE OF RELATIVE USAGE PER WEEK is the column of averages for the most recent week (P), OCT 30 in this example. Next comes the growth per month (Q) calculated over the reporting period followed by the percent of trend line fit (R). The growth per month is expressed in the same units as the item's values. The reporting period is typically 13 weeks although it may be shorter if there are fewer than 13 weeks worth of data contained in the database. A negative number in the GROWTH/MONTH column means that the average values are decreasing; a positive number means an increase in growth. The TREND LINE FIT column indicates how much scatter there is around the trend line. For example, a 90% trend line fit means that most of the data points lie very close to the trend line. The TREND LINE FIT will be given only if it is greater than or equal to 70%. There are two comments that may appear in place of the percentages. ERRATIC VALUES means that data was widely scattered around the trend line and no strong linear pattern could be found in the data. The TREND LINE FIT would be less than 70%. UNCHANGING values refers to values that were relatively constant from week to week. The GROWTH/MONTH would be 0 and the TREND LINE FIT would be at least 70%.

There are no predictions given based on weekly data. Predictions are found only on the Monthly Trend Analysis Report.

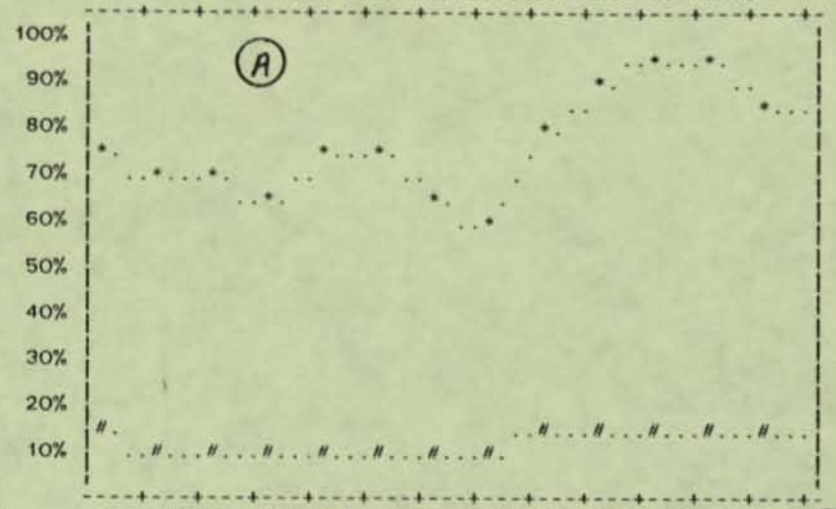
FROM: 25-JUL-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
WEEKLY TREND ANALYSIS REPORT

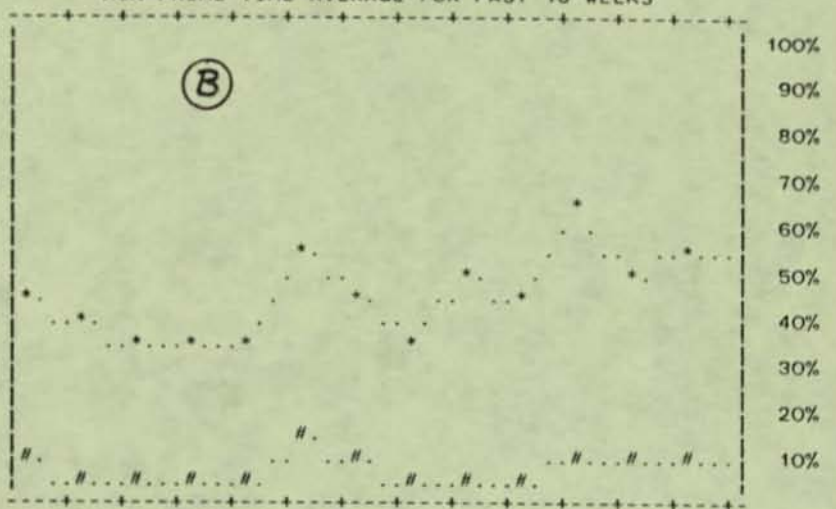
TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

--------- CPU UTILIZATION (*) OVERHEAD (#) AND BOTH (@) ---------

PRIME TIME AVERAGE FOR PAST 13 WEEKS



NON-PRIME TIME AVERAGE FOR PAST 13 WEEKS



WEEK ENDING: AUG 7, AUG 14, AUG 21, AUG 28, SEP 4, SEP 11, SEP 18, SEP 25, OCT 2, OCT 9, OCT 16, OCT 23, OCT 30 (C)

PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL WEEK ENDING		Q1M2W1 AUG 7	Q1M2W2 AUG 14	Q1M2W3 AUG 21	Q1M2W4 AUG 28	Q1M3W1 SEP 4	Q1M3W2 SEP 11	Q1M3W3 SEP 18	Q1M3W4 SEP 25	Q1M3W5 OCT 2	Q2M1W1 OCT 9	Q2M1W2 OCT 16	Q2M1W3 OCT 23	Q2M1W4 OCT 30	LONG TERM LIMIT
------- ITEM -------		W-12	W-11	W-10	W-09	W-08	W-07	W-06	W-05	W-04	W-03	W-02	W-01	W-00	
% CPU	UTIL	75*	67	68	61	71*	70*	61	55	79*	86*	93*	91*	83*	>70%
BAL SET	ADJS/SEC	.8	.7	.7	.6	.7	.8	.6	.5	.8	.9	.9	.9	.8	>1.0
PGS REAS	MEMORY	3013	2662	3307	4062	3012	4548	4548	4548	4548	4548	4548	4548	4298	
% BACK	GND TIME	2	1	1	2	1	1	1	1	1	1	2	3	3	>5%
# FORKS	BS WAIT	1	1	1	1	1	1	1	1	1	2*	3*	1	2*	>2
CONTEXT	SWTS/SEC	42*	42*	40*	34	41*	38	34	29	70*	44*	49*	42*	41*	>40
DSK RDS	PGS/SEC	17	18	16	15	17	23	15	13	17	23	30	23	18	>40
DSK WRS	PGS/SEC	7	6	5	7	6	6	5	7	6	11	11	11	8	>40
SWAP RDS	PGS/SEC	7	5	5	2	3	2	1	1	2	3	3	2	3	>40
SWAP WRS	PGS/SEC	5	4	4	2	3	2	1	1	2	2	3	2	2	>40
% BS WQ	RD WAIT	54	57	55	58	60	68	63	56	69	71	79	71	68	
% BS WQ	WR WAIT	12	13	14	20	13	16	18	24	14	13	7	11	13	
% IDLE	IO TIME	10	12	11	12	11	10	13	12	7	5	4	4	5	
# FREE	MEM PGS	293	168	385	624	232	1084	773	719	715	596	690	1010	507	<25
GEN FREE	PGS/SEC	8	6	6	3	4	3	2	2	3	4	4	3	4	>20
% IDLE	TIME	15	21	21	27	19	20	26	33	14	9*	3*	5*	11	<10%
BIAS	CONTROL	11	11	11	11	11	14	11	11	11	11	11	11	11	
FRK FREE	PGS/SEC	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	>5.0

(G) (H) (D) (E) (F) (J) (I)

* = EXCEEDS LONG TERM LIMIT > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE ----------

Figure 1-5

- AMAR -
WEEKLY TREND ANALYSIS REPORT

FROM: 25-JUL-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL WEEK ENDING		Q1M2W1 AUG 7	Q1M2W2 AUG 14	Q1M2W3 AUG 21	Q1M2W4 AUG 28	Q1M3W1 SEP 4	Q1M3W2 SEP 11	Q1M3W3 SEP 18	Q1M3W4 SEP 25	Q1M3W5 OCT 2	Q2M1W1 OCT 9	Q2M1W2 OCT 16	Q2M1W3 OCT 23	Q2M1W4 OCT 30	LONG TERM
ITEM		W-12	W-11	W-10	W-09	W-08	W-07	W-06	W-05	W-04	W-03	W-02	W-01	W-00	LIMIT
# LOCKED	PAGES	6	6	6	6	6	6	6	6	6	6	6	6	6	>500
# MTAS	ASSIGNED	1	0	1	0	0	1	0	0	0	1	2	1	1	
# BALNCE	SET FRKS	3	3	2	2	3	4	2	1	5	7	14	8	5	>20
MEM MGMT	CYCS/SEC	.1	.0	.1	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	>10.0
WORK SET	LDS/SEC	.3	.3	.2	.1	.2	.1	.1	.1	.1	.2	.2	.1	.2	>1.0
BAL SET	SWPS/SEC	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	>1.0
# ACTIVE	FORKS	3	3	2	2	3	4	2	1	5	7	14	8	5	>30
PG FAULT	TRPS/SEC	38	33	34	32	35	39	29	29	35	49	53*	50*	38	>50
# MEMORY	WRK SETS	99	85	107	135	95	139	154	158	155	159	151	145	152	
# PTYS	IN USE	9	9	8	8	9	9	8	8	8	8	10	11	11	>40
RQ SAVES	PGS/SEC	10.9	7.7	11.8	13.3	11.5	13.0	12.4	11.0	13.8	17.0	13.8	19.1	12.5	
# SHARED	PAGES	1535	1886	1241	486	1536	0	0	0	0	0	0	0	250	
% SCHED	TIME	9	8	8	7	8	7	6	6	12*	9	10*	9	9	>10%
% SWAP	SPC FREE	58	56	60	71	60	70	74	70	68	64	64	73	61	<10%
% BS WQ	SWP WAIT	23	21	22	8	14	6	7	9	9	10	9	8	11	
% IDLE	SWP TIME	3	4	3	1	2	1	1	1	1	1	1	0	1	
% MGMT	MEM TIME	.2	.2	.2	.2	.2	.2	.2	.2	.2	.2	.3	.2	.2	>1.0%
% PAGING	TIME	5	4	4	4	4	5	3	3	4	6	7	6	5	>10%
TTY INTR	CHRS/SEC	0	0	0	0	0	0	0	0	0	0	0	0	0	>10
TTY IN	CHRS/SEC	7	8	8	7	12	7	7	7	8	8	7	8	7	>100
TTY OUT	CHRS/SEC	480	535*	493	421	464	481	473	450	531*	589*	556*	520*	663*	>500
# LINES	IN USE	45	48	46	39	40	40	40	43	47	51	49	38	50	>60
PGS USER	MEMORY	3007	2656	3301	4056	3006	4542	4542	4542	4542	4542	4542	4542	4292	<768
# WORK	SET PGS	3426*	3016*	3730*	4472*	3388*	4424*	4980*	5110*	4997*	5059*	4837*	4476*	4922*	>1500
% USED	TIME	61	55	56	52	60	61	52	47	65	74	80	79	70	>90%
# FORK	WAKS/SEC	9	10	9	9	10	8	9	9	40*	10	9	9	10	>20
% SYSTEM	UPTIME	100.0	100.0	99.2	99.2	94.2	100.0	100.0	100.0	100.0	94.8	94.5	73.9	99.8	
% AMAR	CLK TIME	100.0	80.5	99.0	93.9	92.1	100.0	100.0	100.0	100.0	90.9	86.4	65.4	98.2	
# SYSTEM	RELOADS	0	1	1	3	2	0	0	0	0	5	4	8	2	

NON-PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL WEEK ENDING		Q1M2W1 AUG 7	Q1M2W2 AUG 14	Q1M2W3 AUG 21	Q1M2W4 AUG 28	Q1M3W1 SEP 4	Q1M3W2 SEP 11	Q1M3W3 SEP 18	Q1M3W4 SEP 25	Q1M3W5 OCT 2	Q2M1W1 OCT 9	Q2M1W2 OCT 16	Q2M1W3 OCT 23	Q2M1W4 OCT 30	LONG TERM
ITEM		W-12	W-11	W-10	W-09	W-08	W-07	W-06	W-05	W-04	W-03	W-02	W-01	W-00	LIMIT
% CPU	UTIL	44	39	32	30	35	51	41	30	48	42	64	50	50	>70%
BAL SET	ADJS/SEC	.5	.5	.4	.4	.4	.6	.5	.4	.6	.5	.7	.6	.6	>1.0
PGS REAS	MEMORY	3338	2335	3019	4213	3526	4548	4548	4522	4548	4548	4548	4265	4522	
% BACK	GND TIME	3	1	1	1	1	1	1	1	1	1	2	2	3	>5%
# FORKS	BS WAIT	0	0	0	0	0	0	0	0	0	0	1	0	0	>2
CONTEXT	SWTS/SEC	17	18	14	13	14	90*	43*	13	21	20	23	20	17	>40
DSK RDS	PGS/SEC	18	18	14	14	15	13	15	14	22	17	22	17	15	>40
DSK WRS	PGS/SEC	8	8	6	7	8	6	7	6	11	8	9	9	7	>40
SWAP RDS	PGS/SEC	0	0	0	0	0	0	0	0	0	0	0	0	0	>40
SWAP WRS	PGS/SEC	0	0	0	0	0	0	0	0	0	0	1	1	0	>40

* = EXCEEDS LONG TERM LIMIT > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE ----------

Figure 1-5 (continued)

FROM: 25-JUL-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
WEEKLY TREND ANALYSIS REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

NON-PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL WEEK ENDING		Q1M2W1 AUG 7	Q1M2W2 AUG 14	Q1M2W3 AUG 21	Q1M2W4 AUG 28	Q1M3W1 SEP 4	Q1M3W2 SEP 11	Q1M3W3 SEP 18	Q1M3W4 SEP 25	Q1M3W5 OCT 2	Q2M1W1 OCT 9	Q2M1W2 OCT 16	Q2M1W3 OCT 23	Q2M1W4 OCT 30	LONG TERM
------- ITEM -------		W-12	W-11	W-10	W-09	W-08	W-07	W-06	W-05	W-04	W-03	W-02	W-01	W-00	LIMIT
% BS WQ	RD WAIT	47	41	36	34	34	45	45	34	46	42	66	50	48	
% BS WQ	WR WAIT	35	39	46	48	44	41	39	49	33	40	22	34	36	
% IDLE	IO TIME	13	11	9	9	9	8	11	10	13	9	13	11	10	
# FREE	MEM PGS	2129	1165	1889	3007	2362	3337	3447	3467	3425	3389	3271	2939	3320	<25
GEN FREE	PGS/SEC	1	1	1	1	1	1	1	1	1	1	1	1	1	>20
% IDLE	TIME	43	50	59	60	57	41	48	60	39	49	24	39	40	<10%
BIAS	CONTROL	11	11	11	11	11	12	11	11	11	11	11	11	11	
FRK FREE	PGS/SEC	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	>5.0
# LOCKED	PAGES	6	6	6	6	6	6	6	6	6	6	6	6	6	>500
# MTAS	ASSIGNED	1	1	1	1	1	1	1	0	1	1	2	1	1	
# BALNCE	SET FRKS	1	1	1	1	1	1	1	1	1	1	1	1	1	>20
MEM MGMT	CYCS/SEC	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	>10.0
WORK SET	LDS/SEC	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	>1.0
BAL SET	SWPS/SEC	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	>1.0
# ACTIVE	FORKS	1	1	1	1	1	1	1	1	1	1	1	1	1	>30
PG FAULT	TRPS/SEC	32	30	25	24	28	25	27	24	42	38	53*	40	32	>50
# MEMORY	WRK SETS	45	41	38	47	47	47	39	34	36	43	46	49	40	
# PTYS	IN USE	8	8	8	8	8	8	7	7	8	8	9	10	11	>40
RQ SAVES	PGS/SEC	15.4	9.9	10.0	8.5	10.8	15.1	12.3	9.0	17.2	19.1	37.8	19.4	16.3	
# SHARED	PAGES	1210	2213	1529	335	1022	0	0	26	0	0	0	283	26	
% SCHED	TIME	3	3	2	3	3	13*	6	2	4	4	4	4	4	>10%
% SWAP	SPC FREE	84	84	84	87	86	84	85	82	83	84	85	83	79	<10%
% BS WQ	SWP WAIT	3	4	3	2	2	2	2	2	2	2	2	2	2	
% IDLE	SWP TIME	0	0	0	0	0	0	0	0	0	0	0	0	0	
% MGMT	MEM TIME	.1	.1	.1	.1	.1	.1	.1	.1	.1	.1	.1	.1	.1	>1.0%
% PAGING	TIME	4	4	3	3	3	3	3	3	5	4	6	5	4	>10%
TTY INTR	CHRS/SEC	0	0	0	0	0	0	0	0	0	0	0	0	0	>10
TTY IN	CHRS/SEC	2	3	2	1	1	1	2	3	2	1	1	2	1	>100
TTY OUT	CHRS/SEC	78	68	40	38	33	37	33	36	47	78	61	73	133	>500
# LINES	IN USE	2	3	3	2	2	2	2	2	2	2	2	3	3	>60
PGS USER	MEMORY	3332	2329	3013	4207	3520	4542	4542	4516	4542	4542	4542	4259	4516	<768
# WORK	SET PGS	1460	1379	1327	1415	1377	1450	1276	1239	1291	1362	1497	1566*	1415	>1500
% USED	TIME	37	34	28	27	31	38	34	27	43	37	58	43	43	>90%
# FORK	WAKS/SEC	2	3	2	2	2	78*	30*	2	5	6	2	3	2	>20
% SYSTEM	UPTIME	95.7	77.8	93.3	95.0	95.2	95.7	95.0	98.2	72.0	92.0	93.8	63.1	97.0	
% AMAR	CLK TIME	94.4	68.9	92.0	92.8	93.6	95.4	94.4	97.3	70.8	89.7	86.0	60.4	96.9	
# SYSTEM	RELOADS	4	5	5	7	4	1	2	2	4	8	16	14	1	

* = EXCEEDS LONG TERM LIMIT > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE ----------

Figure 1-5 (continued)

FROM: 25-JUL-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
WEEKLY TREND ANALYSIS REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

PRIME TIME TREND OF KEY UTILIZATION ITEMS

ITEM		(K) FIRST WEEKLY AVG	TABLE OF RELATIVE USAGE PER WEEK (L)	(P) LAST WEEKLY AVG	-QUARTERLY- -- TREND -- GROWTH /MONTH (Q)	LINE FIT (R)
% CPU	UTIL	75	[8 7 7 6 7 7 6 6 8 9 10 10 9]	83	ERRATIC VALUES	
BAL SET	ADJS/SEC	.8	[8 7 7 7 7 8 6 5 8 9 10 9 8]	.8	ERRATIC VALUES	
PGS REAS	MEMORY	3013	[6 5 6 8 6 9 9 9 9 9 9 9 8]	4298	ERRATIC VALUES	
% BACK	GND TIME	2	[11 6 6 7 6 5 6 5 6 6 10 13 15]	3	ERRATIC VALUES	
# FORKS	BS WAIT	1	[8 7 6 5 7 9 5 3 7 9 17 9 9]	2	ERRATIC VALUES	
CONTEXT	SWTS/SEC	42	[8 8 7 6 7 7 6 5 13 8 9 8 7]	41	ERRATIC VALUES	
DSK RDS	PGS/SEC	17	[7 7 7 6 7 9 6 5 7 9 12 9 7]	18	ERRATIC VALUES	
DSK WRS	PGS/SEC	7	[7 6 6 7 6 6 5 8 6 12 11 12 9]	8	ERRATIC VALUES	
SWAP RDS	PGS/SEC	7	[17 13 13 5 9 5 4 4 5 7 8 5 7]	3	ERRATIC VALUES	
SWAP WRS	PGS/SEC	5	[15 12 12 6 9 5 4 4 6 7 8 6 7]	2	ERRATIC VALUES	
% BS WQ	RD WAIT	54	[7 7 7 7 7 8 8 7 8 9 10 9 8]	68	ERRATIC VALUES	
% BS WQ	WR WAIT	12	[6 7 8 10 7 9 10 12 7 7 3 6 7]	13	ERRATIC VALUES	
% IDLE	IO TIME	10	[8 10 9 11 9 8 11 10 6 4 3 4 5]	5	ERRATIC VALUES	
# FREE	MEM PGS	293	[4 2 5 8 3 14 10 9 9 8 9 13 7]	507	ERRATIC VALUES	
GEN FREE	PGS/SEC	8	[15 12 12 6 8 5 5 5 6 7 8 6 7]	4	ERRATIC VALUES	
% IDLE	TIME	15	[7 9 9 12 8 9 11 14 6 4 1 2 5]	11	ERRATIC VALUES	
BIAS	CONTROL	11	[8 8 8 8 8 9 8 8 8 8 8 8 8]	11	ERRATIC VALUES	
FRK FREE	PGS/SEC	.0	[8 8 8 8 8 8 8 8 8 8 8 8 8]	.0	ERRATIC VALUES	
# LOCKED	PAGES	6	[8 8 8 8 8 8 8 8 8 8 8 8 8]	6	ERRATIC VALUES	
# MTAS	ASSIGNED	1	[10 6 8 5 4 7 3 3 6 8 22 15 7]	1	ERRATIC VALUES	
# BALNCE	SET FRKS	3	[5 5 4 4 5 7 3 3 8 13 25 15 8]	5	ERRATIC VALUES	
MEM MGMT	CYCS/SEC	.1	[9 6 9 6 6 6 6 6 6 6 6 6 6]	.0	ERRATIC VALUES	
WORK SET	LDS/SEC	.3	[12 11 11 6 8 5 5 5 6 7 7 5 7]	.2	ERRATIC VALUES	
BAL SET	SWPS/SEC (M)	.0	[8 8 8 8 8 8 8 8 8 8 8 8 8]	.0	ERRATIC VALUES	
# ACTIVE	FORKS	3	[5 5 4 4 5 7 3 3 8 13 25 15 8]	5	ERRATIC VALUES	
PG FAULT	TRPS/SEC	38	[8 7 7 7 7 8 6 6 7 10 11 10 8]	38	ERRATIC VALUES	
# MEMORY	WRK SETS	99	[6 5 6 8 5 8 9 9 9 9 9 8 9]	152	ERRATIC VALUES	
# PTYS	IN USE	9	[8 7 7 7 8 8 7 7 7 7 8 9 9]	11	ERRATIC VALUES	
RQ SAVES	PGS/SEC (N)	10.9	[7 5 7 8 7 8 7 7 8 10 8 11 8]	12.5	ERRATIC VALUES	
# SHARED	PAGES	1535	[22 27 18 7 22 0 0 0 0 0 0 0 4]	250	ERRATIC VALUES	
% SCHED	TIME	9	[8 7 7 6 7 7 6 5 11 9 10 9 9]	9	ERRATIC VALUES	
% SWAP	SPC FREE	58	[7 7 7 8 7 8 9 8 8 8 8 9 7]	61	ERRATIC VALUES	
% BS WQ	SWP WAIT	23	[15 13 14 5 9 4 4 5 6 6 6 5 7]	11	ERRATIC VALUES	
% IDLE	SWP TIME	3	[17 18 18 5 10 4 5 5 4 4 4 2 4]	1	ERRATIC VALUES	
% MGMT	MEM TIME	.2	[9 7 8 7 7 8 8 8 8 9 10 8 9]	.2	ERRATIC VALUES	
% PAGING	TIME	5	[8 7 7 6 7 8 6 6 7 10 11 10 8]	5	ERRATIC VALUES	
TTY INTR	CHRS/SEC	0	[8 8 10 8 10 5 5 5 8 10 10 10 10]	0	ERRATIC VALUES	
TTY IN	CHRS/SEC	7	[7 8 8 7 12 7 7 7 8 8 7 8 8]	7	ERRATIC VALUES	
TTY OUT	CHRS/SEC	480	[7 8 7 6 7 7 7 7 8 9 8 8 10]	663	ERRATIC VALUES	
# LINES	IN USE	45	[8 8 8 7 7 7 7 7 8 9 9 7 9]	50	ERRATIC VALUES	
PGS USER	MEMORY	3007	[6 5 6 8 6 9 9 9 9 9 9 9 8]	4292	ERRATIC VALUES	

PREDICTIONS ARE ONLY MADE USING MONTHLY DATA.
GIVEN THE CURRENT TREND, THERE IS A 90% CHANCE THAT THE ACTUAL VALUES WILL FALL WITHIN THE PREDICTED RANGES.
UNCHANGING VALUES: REFERS TO RELATIVELY CONSTANT VALUES. ERRATIC VALUES: REFERS TO THE LACK OF A STRONG LINEAR PATTERN IN THE DATA.
* = EXCEEDS LONG TERM LIMIT > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO ---------- CONTINUED NEXT PAGE ----------

Figure 1-5 (continued)

FROM: 25-JUL-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
WEEKLY TREND ANALYSIS REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

PRIME TIME TREND OF KEY UTILIZATION ITEMS

ITEM		FIRST WEEKLY AVG	TABLE OF RELATIVE USAGE PER WEEK	LAST WEEKLY AVG	-QUARTERLY- -- TREND -- GROWTH /MONTH	LINE FIT
# WORK	SET PGS	3426	[6 5 7 8 6 8 9 9 9 9 8 8 9]	4922		ERRATIC VALUES
% USED	TIME	61	[8 7 7 6 7 8 6 6 8 9 10 10 9]	70		ERRATIC VALUES
# FORK	WAKS/SEC	9	[6 6 6 6 6 6 6 6 26 7 6 6 7]	10		ERRATIC VALUES
% SYSTEM	UPTIME	100.0	[8 8 8 8 8 8 8 8 8 8 8 6 8]	99.8		ERRATIC VALUES
% AMAR	CLK TIME	100.0	[8 7 8 8 8 8 8 8 8 8 7 5 8]	98.2		ERRATIC VALUES
# SYSTEM	RELOADS	0	[0 4 4 12 8 0 0 0 0 19 15 31 8]	2		ERRATIC VALUES

NON-PRIME TIME TREND OF KEY UTILIZATION ITEMS

ITEM		FIRST WEEKLY AVG	TABLE OF RELATIVE USAGE PER WEEK	LAST WEEKLY AVG	-QUARTERLY- -- TREND -- GROWTH /MONTH	LINE FIT
% CPU	UTIL	44	[8 7 6 6 6 9 7 6 9 8 12 9 9]	50		ERRATIC VALUES
BAL SET	ADJS/SEC	.5	[8 7 6 6 6 9 8 6 9 7 11 9 9]	.6		ERRATIC VALUES
PGS REAS	MEMORY	3338	[6 4 6 8 7 9 9 9 9 9 9 8 9]	4522		ERRATIC VALUES
% BACK	GND TIME	3	[20 4 4 5 4 4 4 4 4 4 10 13 18]	3		ERRATIC VALUES
# FORKS	BS WAIT	0	[8 7 6 6 7 7 6 5 9 8 14 10 8]	0		ERRATIC VALUES
CONTEXT	SWTS/SEC	17	[5 5 4 4 4 27 13 4 6 6 7 6 5]	17		ERRATIC VALUES
DSK RDS	PGS/SEC	18	[8 8 7 7 7 6 7 7 11 8 10 8 7]	15		ERRATIC VALUES
DSK WRS	PGS/SEC	8	[8 8 6 7 8 6 7 7 11 9 9 9 7]	7		ERRATIC VALUES
SWAP RDS	PGS/SEC	0	[11 12 9 6 8 6 6 6 9 7 8 8 7]	0	-0.04	74%
SWAP WRS	PGS/SEC	0	[8 9 7 7 7 6 7 6 9 7 10 10 8]	0		ERRATIC VALUES
% BS WQ	RD WAIT	47	[8 7 6 6 6 8 8 6 8 7 12 9 9]	48		ERRATIC VALUES
% BS WQ	WR WAIT	35	[7 8 9 9 9 8 8 10 6 8 4 7 7]	36		ERRATIC VALUES
% IDLE	IO TIME	13	[9 8 7 7 7 6 8 7 10 7 9 8 7]	10		ERRATIC VALUES
# FREE	MEM PGS	2129	[6 3 5 8 6 9 9 9 9 9 9 8 9]	3320		ERRATIC VALUES
GEN FREE	PGS/SEC	1	[8 9 7 7 7 7 7 6 9 8 10 10 8]	1		ERRATIC VALUES
% IDLE	TIME	43	[7 8 10 10 9 7 8 10 6 8 4 6 7]	40		ERRATIC VALUES
BIAS	CONTROL	11	[8 8 8 8 8 8 9 8 8 8 8 8 8 8]	11		ERRATIC VALUES
FRK FREE	PGS/SEC	.0	[8 8 8 8 8 8 8 8 8 8 8 8 8 8]	.0		ERRATIC VALUES
# LOCKED	PAGES	6	[8 8 8 8 8 8 8 8 8 8 8 8 8 8]	6		ERRATIC VALUES
# MTAS	ASSIGNED	1	[9 8 6 6 7 8 6 5 8 8 16 7 9]	1		ERRATIC VALUES
# BALNCE	SET FRKS	1	[8 7 5 5 6 9 7 5 9 8 14 10 9]	1		ERRATIC VALUES
MEM MGMT	CYCS/SEC	.0	[8 8 8 8 8 8 8 8 8 8 8 8 8 8]	.0		ERRATIC VALUES
WORK SET	LDS/SEC	.0	[8 8 8 8 8 4 8 8 8 8 8 8 8 8]	.0		ERRATIC VALUES
BAL SET	SWPS/SEC	.0	[8 8 8 8 8 8 8 8 8 8 8 8 8 8]	.0		ERRATIC VALUES
# ACTIVE	FORKS	1	[8 7 5 5 6 9 7 5 9 8 14 10 9]	1		ERRATIC VALUES
PG FAULT	TRPS/SEC	32	[8 7 6 6 7 6 7 6 10 9 13 10 8]	32		ERRATIC VALUES

PREDICTIONS ARE ONLY MADE USING MONTHLY DATA.
GIVEN THE CURRENT TREND, THERE IS A 90% CHANCE THAT THE ACTUAL VALUES WILL FALL WITHIN THE PREDICTED RANGES.
UNCHANGING VALUES: REFERS TO RELATIVELY CONSTANT VALUES. ERRATIC VALUES: REFERS TO THE LACK OF A STRONG LINEAR PATTERN IN THE DATA.
* = EXCEEDS LONG TERM LIMIT > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE ----------

Figure 1-5 (continued)

FROM: 25-JUL-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
WEEKLY TREND ANALYSIS REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

NON-PRIME TIME TREND OF KEY UTILIZATION ITEMS

ITEM		FIRST WEEKLY AVG	TABLE OF RELATIVE USAGE PER WEEK												-QUARTERLY- LAST WEEKLY AVG	-- TREND -- GROWTH /MONTH	LINE FIT
# MEMORY	WRK SETS	45	[8	7	7	9	9	7	6	7	8	8	9	7]	40	ERRATIC VALUES	
# PTYS	IN USE	8	[8	8	7	7	7	7	7	8	8	8	9	10]	11	ERRATIC VALUES	
RQ SAVES	PGS/SEC	15.4	[8	5	5	4	5	8	6	5	9	10	19	10	8]	16.3	ERRATIC VALUES
# SHARED	PAGES	1210	[19	35	24	5	16	0	0	0	0	0	0	4	0]	26	ERRATIC VALUES
% SCHED	TIME	3	[6	6	4	4	5	22	11	4	7	6	8	8	7]	4	ERRATIC VALUES
% SWAP	SPC FREE	84	[8	8	8	8	8	8	8	8	8	8	8	8	7]	79	ERRATIC VALUES
% BS WQ	SWP WAIT	3	[11	13	9	6	6	6	6	6	8	6	7	8	8]	2	ERRATIC VALUES
% IDLE	SWP TIME	0	[11	14	11	5	8	5	6	6	9	6	7	6	5]	0	-0.04 99%
% MGMT	MEM TIME	.1	[8	7	7	8	8	9	9	9	9	9	9	10	9]	.1	ERRATIC VALUES
% PAGING	TIME	4	[8	8	6	6	7	6	7	6	11	9	12	9	7]	4	ERRATIC VALUES
TTY INTR	CHRS/SEC	0	[8	8	8	8	8	8	8	8	8	8	8	8	8]	0	ERRATIC VALUES
TTY IN	CHRS/SEC	2	[7	14	8	7	6	4	11	12	7	6	5	9	5]	1	ERRATIC VALUES
TTY OUT	CHRS/SEC	78	[10	9	5	5	4	5	4	5	6	10	8	10	18]	133	ERRATIC VALUES
# LINES	IN USE	2	[8	10	8	6	6	7	6	7	7	8	8	10	10]	3	ERRATIC VALUES
PGS USER	MEMORY	3332	[6	4	6	8	7	9	9	9	9	9	9	8	9]	4516	ERRATIC VALUES
# WORK	SET PGS	1460	[8	8	7	8	8	8	7	7	7	8	8	9	8]	1415	ERRATIC VALUES
% USED	TIME	37	[8	7	6	6	6	8	7	6	9	8	12	9	9]	43	ERRATIC VALUES
# FORK	WAKS/SEC	2	[1	2	2	2	2	52	20	1	3	4	2	2	1]	2	ERRATIC VALUES
% SYSTEM	UPTIME	95.7	[8	7	8	8	8	8	8	8	6	8	8	5	8]	97.0	ERRATIC VALUES
% AMAR	CLK TIME	94.4	[8	6	8	8	8	8	8	9	6	8	8	5	9]	96.9	ERRATIC VALUES
# SYSTEM	RELOADS	4	[5	7	7	10	5	1	3	3	5	11	22	19	1]	1	ERRATIC VALUES

PREDICTIONS ARE ONLY MADE USING MONTHLY DATA.
GIVEN THE CURRENT TREND, THERE IS A 90% CHANCE THAT THE ACTUAL VALUES WILL FALL WITHIN THE PREDICTED RANGES.
UNCHANGING VALUES: REFERS TO RELATIVELY CONSTANT VALUES. ERRATIC VALUES: REFERS TO THE LACK OF A STRONG LINEAR PATTERN IN THE DATA.
* = EXCEEDS LONG TERM LIMIT > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

Figure 1-5 (continued)

1.3.2.2 Monthly Trend Analysis Report -

The Monthly Trend Analysis Report is almost identical to the Weekly Trend Analysis Report (refer to the previous section). Major differences are:

All averages are monthly averages.

All dates on the report refer to the fiscal month ending date.

At most 12 months of data are reported.

The format of the first header line (A) for the PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS table indicates the number of fiscal weeks in each particular month. For example, the month ending OCT 2 (B) contained five weeks and was the third month in the first quarter (Q1M3W5).

The trend analysis section of the report, beginning on page 4 of this sample (refer to Figure 1-6), will contain predictions if at least 6 months worth of monthly data is present in the database.

Predictions follow the TREND LINE FIT column. Predictions will NOT occur if a tend is not present. If no trends exist, as in this example, ERRATIC VALUES will be printed under the GROWTH/MONTH and TREND LINE FIT columns. If a trend is present, a range of values where the item's average is expected to fall within 6 months (C) and then within 12 months (D) would be listed. An underlying assumption in these predictions is that the current linear trend would continue. If that were to occur, there would be a 90% chance that the item's actual averages would fall within the predicted ranges. Only the past usage history of the machine is taken into account, not the future plans of the computer center or its users. For example, assume % CPU UTIL was expected to grow to between 90-100% by the end of six months. If the computer center were to move users to another machine, this percentage might decrease. If several new applications were added, the percentage might increase more rapidly. Note that some items, such as WORK SET LDS/SEC, show widely varying index numbers (F) while the actual averages are almost always close to or equal to 0. This occurs because the index numbers are computed using higher precision values in the database records. The report values are usually rounded to a lower precision thus masking some of the variation.

The next column, PREDICTED PERIOD WHEN LONG TERM LIMIT REACHED (G), gives the month or range of months when the confidence interval around the trend line exceeds (if it is a high limit) or falls below (if it is a low limit) the long term limit, <u>not</u> when an individual average exceeds or falls below the limit. The confidence interval denotes the edges of a band around the trend line where 90% of the averages fall. Again, since no items in this example show a trend, the column is blank. Had there been a

trend, dates in this column (H) would have been given in the format yymmm where yy is the actual calendar not fiscal year and mmm is the month. The comment, ALREADY BY, would mean that the long term limit had already been exceeded. NONE LIKELY would mean that the trend line was below a high long term limit or above a low long term limit and diverging away from it. Predictions are not made further than 2 years out. If the trend line is expected to exceed the long term limit sometime after 2 years have elapsed, the comment AFTER 24 MO will be printed. The final column on this page specifies the long term limits.

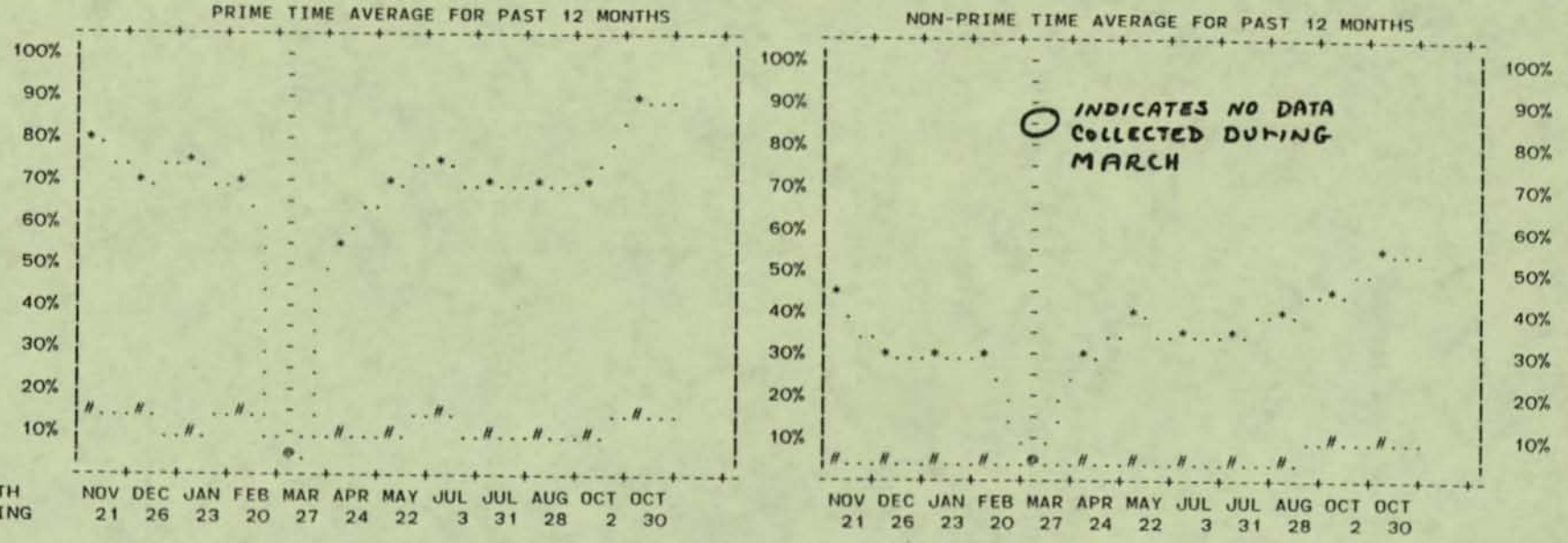
FROM: 25-OCT-81 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
MONTHLY TREND ANALYSIS REPORT

PAGE: 1

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

--------- CPU UTILIZATION (*) OVERHEAD (#) AND BOTH (@) ---------



PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL MONTH ENDING		Q2M2W4 NOV 21	Q2M3W5 DEC 26	Q3M1W4 JAN 23	Q3M2W4 FEB 20	Q3M3W5 MAR 27	Q4M1W4 APR 24	Q4M2W4 MAY 22	Q4M3W6 JUL 3	Q1M1W4 JUL 31	Q1M2W4 AUG 28	Q1M3W5 (B) OCT 2	Q2M1W4 (A) OCT 30	LONG TERM LIMIT
------- ITEM -------		M-11	M-10	M-09	M-08	M-07	M-06	M-05	M-04	M-03	M-02	M-01	M-00	
% CPU	UTIL	78*	70*	74*	70*		53	66	71*	69	68	67	88*	>70%
BAL SET	ADJS/SEC	.8	.7	.8	.7		.5	.6	.7	.7	.7	.7	.9	>1.0
PGS REAS	MEMORY	1991	2647	3015	3015		4548	4175	4120	4449	3279	4261	4476	
% BACK	GND TIME	2	1	1	2		2	2	2	2	2	1	3	
# FORKS	BS WAIT	2*	1	1	1		1	1	1	1	1	1	2*	>5%
CONTEXT	SWTS/SEC	49*	44*	44*	42*		30	37	42*	39	40*	42*	44*	>2
DSK RDS	PGS/SEC	22	19	20	16		13	16	18	17	16	17	23	>40
DSK WRS	PGS/SEC	7	6	6	6		5	5	6	6	6	6	10	>40
SWAP RDS	PGS/SEC	7	6	3	4		1	2	3	2	5	2	3	>40
SWAP WRS	PGS/SEC	6	5	3	4		1	2	3	2	4	2	2	>40
% BS WQ	RD WAIT	62	56	65	60		59	64	65	65	56	63	72	
% BS WQ	WR WAIT	10	16	13	15		22	17	15	16	15	17	11	
% IDLE	IO TIME	11	10	11	9		10	9	10	10	11	10	5	
# FREE	MEM PGS	111	328	270	204		1070	474	401	595	374	712	673	<25
GEN FREE	PGS/SEC	8	8	5	5		2	3	4	3	6	3	4	>20
% IDLE	TIME	11	20	15	21		37	26	19	20	21	22	7*	<10%
BIAS	CONTROL	11	11	11	11		11	11	11	11	11	12	11	
FRK FREE	PGS/SEC	.0	.0	.0	.0		.0	.0	.0	.0	.0	.0	.0	>5.0

* = EXCEEDS LONG TERM LIMIT > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE ----------

Figure 1-6

FROM: 25-OCT-81 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
MONTHLY TREND ANALYSIS REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL MONTH ENDING		Q2M2W4 NOV 21	Q2M3W5 DEC 26	Q3M1W4 JAN 23	Q3M2W4 FEB 20	Q3M3W5 MAR 27	Q4M1W4 APR 24	Q4M2W4 MAY 22	Q4M3W6 JUL 3	Q1M1W4 JUL 31	Q1M2W4 AUG 28	Q1M3W5 OCT 2	Q2M1W4 OCT 30	LONG TERM
------- ITEM -------		M-11	M-10	M-09	M-08	M-07	M-06	M-05	M-04	M-03	M-02	M-01	M-00	LIMIT
# LOCKED	PAGES	6	6	6	6		6	6	6	6	6	6	6	>500
# MTAS	ASSIGNED	0	1	1	0		0	0	1	0	1	0	1	
# BALNCE	SET FRKS	3	3	3	3		2	3	4	3	3	3	8	>20
MEM MGMT	CYCS/SEC	.1	.1	.0	.0		.0	.0	.0	.0	.0	.0	.0	>10.0
WORK SET	LDS/SEC	.3	.3	.2	.2		.1	.2	.2	.1	.2	.1	.1	>1.0
BAL SET	SWPS/SEC	.0	.0	.0	.0		.0	.0	.0	.0	.0	.0	.0	>1.0
# ACTIVE	FORKS	3	3	3	3		2	3	4	3	3	3	8	>30
PG FAULT	TRPS/SEC	40	37	38	33		26	30	36	34	35	33	47	>50
# MEMORY	WRK SETS	49	74	94	92		149	152	146	153	107	141	152	
# PTYS	IN USE	9	9	9	9		9	9	9	9	9	8	10	>40
RQ SAVES	PGS/SEC	8.9	9.4	13.1	9.5		9.6	9.8	11.3	12.0	11.0	12.4	15.3	
# SHARED	PAGES	2560	1904	1536	1536		0	373	428	99	1269	287	72	
% SCHED	TIME	9	9	8	8		6	7	8	8	8	8	10*	>10%
% SWAP	SPC FREE	45	53	54	48		72	64	59	66	61	68	65	<10%
% BS WQ	SWP WAIT	22	19	14	17		7	13	13	10	19	9	10	
% IDLE	SWP TIME	3	3	2	2		1	2	2	1	3	1	1	
% MGMT	MEM TIME	.2	.2	.2	.2		.2	.2	.2	.2	.2	.2	.2	>1.0%
% PAGING	TIME	5	5	4	4		3	3	4	4	4	4	6	>10%
TTY INTR	CHRS/SEC	0	0	0	0		0	0	0	0	0	0	0	>10
TTY IN	CHRS/SEC	59	8	9	10		13	9	8	9	7	8	8	>100
TTY OUT	CHRS/SEC	511*	459	509*	564*		438	548*	576*	535*	480	480	589*	>500
# LINES	IN USE	50	47	47	53		40	51	54	47	44	42	48	>60
PGS USER	MEMORY	1985	2641	3009	3009		4542	4169	4114	4443	3273	4255	4470	<768
# WORK	SET PGS	2028*	2720*	3277*	3291*		4569*	4797*	4752*	4978*	3681*	4599*	4852*	>1500
% USED	TIME	64	57	63	57		45	55	60	59	56	57	76	>90%
# FORK	WAKS/SEC	9	9	9	11		9	10	11	10	9	15	10	>20
% SYSTEM	UPTIME	56.7	99.5	60.8	12.8	.0	28.0	81.9	89.6	83.6	99.6	98.9	90.8	
% AMAR	CLK TIME	55.5	98.4	57.4	12.8	.0	25.8	79.2	87.9	77.7	93.3	98.4	85.2	
# SYSTEM	RELOADS	4	4	0	0	0	2	7	11	4	5	2	19	

NON-PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL MONTH ENDING		Q2M2W4 NOV 21	Q2M3W5 DEC 26	Q3M1W4 JAN 23	Q3M2W4 FEB 20	Q3M3W5 MAR 27	Q4M1W4 APR 24	Q4M2W4 MAY 22	Q4M3W6 JUL 3	Q1M1W4 JUL 31	Q1M2W4 AUG 28	Q1M3W5 OCT 2	Q2M1W4 OCT 30	LONG TERM
------- ITEM -------		M-11	M-10	M-09	M-08	M-07	M-06	M-05	M-04	M-03	M-02	M-01	M-00	LIMIT
% CPU	UTIL	42	30	28	27		29	37	31	34	36	41	51	>70%
BAL SET	ADJS/SEC	.5	.4	.4	.3		.3	.5	.4	.4	.4	.5	.6	>1.0
PGS REAS	MEMORY	1991	2710	3015	3015		4548	4111	4026	4429	3288	4330	4489	
% BACK	GND TIME	1	1	1	1		2	1	2	2	1	1	2	>5%
# FORKS	BS WAIT	0	0	0	0		0	0	0	0	0	0	0	>2
CONTEXT	SWTS/SEC	18	14	13	12		12	15	13	14	15	37	20	>40
DSK RDS	PGS/SEC	16	13	12	13		11	15	13	14	16	15	18	>40
DSK WRS	PGS/SEC	7	6	5	6		5	7	6	6	7	7	8	>40
SWAP RDS	PGS/SEC	0	0	0	0		0	0	0	0	0	0	0	>40
SWAP WRS	PGS/SEC	0	0	0	0		0	0	0	0	0	0	0	>40

* = EXCEEDS LONG TERM LIMIT > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE ----------

Figure 1-6 (continued)

FROM: 25-OCT-81 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
MONTHLY TREND ANALYSIS REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

NON-PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL MONTH ENDING		Q2M2W4 NOV 21	Q2M3W5 DEC 26	Q3M1W4 JAN 23	Q3M2W4 FEB 20	Q3M3W5 MAR 27	Q4M1W4 APR 24	Q4M2W4 MAY 22	Q4M3W6 JUL 3	Q1M1W4 JUL 31	Q1M2W4 AUG 28	Q1M3W5 OCT 2	Q2M1W4 OCT 30	LONG TERM
------- ITEM -------		M-11	M-10	M-09	M-08	M-07	M-06	M-05	M-04	M-03	M-02	M-01	M-00	LIMIT
% BS WQ	RD WAIT	45	36	35	28		37	44	39	37	40	40	52	
% BS WQ	WR WAIT	34	44	48	55		50	41	46	47	42	42	33	
% IDLE	IO TIME	15	13	11	10		8	12	9	9	10	10	11	
# FREE	MEM PGS	942	1646	2015	2025		3259	2899	2917	3146	2109	3199	3257	<25
GEN FREE	PGS/SEC	1	1	0	0		0	1	1	1	1	1	1	>20
% IDLE	TIME	43	58	61	63		64	51	59	57	53	49	38	<10%
BIAS	CONTROL	11	11	11	11		11	11	11	11	11	11	11	
FRK FREE	PGS/SEC	.0	.0	.0	.0		.0	.0	.0	.0	.0	.0	.0	>5.0
# LOCKED	PAGES	6	6	6	6		6	6	6	6	6	6	6	>500
# MTAS	ASSIGNED	1	1	1	1		1	1	1	1	1	1	1	
# BALNCE	SET FRKS	1	1	1	1		1	1	1	1	1	1	1	>20
MEM MGMT	CYCS/SEC	.0	.0	.0	.0		.0	.0	.0	.0	.0	.0	.0	>10.0
WORK SET	LDS/SEC	.0	.0	.0	.0		.0	.0	.0	.0	.0	.0	.0	>1.0
BAL SET	SWPS/SEC	.0	.0	.0	.0		.0	.0	.0	.0	.0	.0	.0	>1.0
# ACTIVE	FORKS	1	1	1	1		1	1	1	1	1	1	1	>30
PG FAULT	TRPS/SEC	28	23	21	21		19	27	22	24	28	28	41	>50
# MEMORY	WRK SETS	34	38	33	33		59	47	41	48	43	41	44	
# PTYS	IN USE	9	9	9	9		9	9	9	8	8	8	9	>40
RQ SAVES	PGS/SEC	13.0	8.5	9.6	6.5		8.7	12.1	8.8	10.3	11.1	12.6	23.2	
# SHARED	PAGES	2560	1841	1536	1536		0	437	522	119	1260	218	59	
% SCHED	TIME	3	2	2	3		2	3	3	3	3	6	4	>10%
% SWAP	SPC FREE	81	83	81	81		86	84	82	83	85	84	83	<10%
% BS WQ	SWP WAIT	3	2	2	1		1	2	2	2	3	2	2	
% IDLE	SWP TIME	0	0	0	0		0	0	0	0	0	0	0	
% MGMT	MEM TIME	.1	.1	.1	.1		.1	.1	.1	.1	.1	.1	.1	>1.0%
% PAGING	TIME	3	3	3	3		2	3	3	3	3	3	4	>10%
TTY INTR	CHRS/SEC	0	0	0	0		0	0	0	0	0	0	0	>10
TTY IN	CHRS/SEC	83	1	1	1		1	1	1	9	2	2	1	>100
TTY OUT	CHRS/SEC	30	26	45	20		53	55	51	60	56	37	89	>500
# LINES	IN USE	4	3	3	3		3	3	3	3	3	2	3	>60
PGS USER	MEMORY	1985	2704	3009	3009		4542	4105	4020	4423	3282	4324	4483	<768
# WORK	SET PGS	1203	1221	1150	1169		1537*	1416	1300	1523*	1397	1328	1449	>1500
% USED	TIME	37	26	25	23		24	33	27	29	32	34	45	>90%
# FORK	WAKS/SEC	2	2	2	2		2	2	2	2	2	25*	3	>20
% SYSTEM	UPTIME	60.4	86.0	78.8	11.7	.0	31.3	77.6	92.6	88.1	90.4	91.2	86.5	
% AMAR	CLK TIME	60.2	85.1	75.5	11.7	.0	29.4	75.1	91.6	86.1	87.0	90.3	83.2	
# SYSTEM	RELOADS	6	16	4	1	0	4	12	15	9	21	13	39	

* = EXCEEDS LONG TERM LIMIT > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE ----------

Figure 1-6 (continued)

FROM: 25-OCT-81 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
MONTHLY TREND ANALYSIS REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

PRIME TIME TREND OF KEY UTILIZATION ITEMS

ITEM		FIRST MONTHLY AVG	TABLE OF RELATIVE USAGE PER MONTH	LAST MONTHLY AVG	ANNUAL -- TREND -- GROWTH /MONTH	LINE FIT	PREDICTED RANGE OF VALUES WHERE AVERAGE WILL FALL IN 06 MONTHS	12 MONTHS	PREDICTED PERIOD WHEN LONG TERM LIMIT REACHED	LONG TERM LIMIT
% CPU	UTIL	78	[9 8 9 8 6 8 8 8 8 8 10]	88	ERRATIC	VALUES	(C)	(D)	(G)	>70
BAL SET	ADJS/SEC	.8	[9 8 9 8 6 7 8 8 8 8 10]	.9	ERRATIC	VALUES				>1.0
PGS REAS	MEMORY	1991	[4 6 7 7 10 9 9 10 7 10 10]	4476	ERRATIC	VALUES				
% BACK	GND TIME	2	[7 7 7 8 8 8 9 11 8 6 12]	3	ERRATIC	VALUES			(H)	>5
# FORKS	BS WAIT	2	[10 10 8 8 4 6 8 7 8 7 12]	2	ERRATIC	VALUES				>2
CONTEXT	SWTS/SEC	49	[10 9 9 8 6 7 8 8 8 8 9]	44	ERRATIC	VALUES				>40
DSK RDS	PGS/SEC	22	[10 9 9 7 6 7 8 8 7 8 10]	23	ERRATIC	VALUES				>40
DSK WRS	PGS/SEC	7	[9 8 8 8 7 7 7 8 8 8 13]	10	ERRATIC	VALUES				>40
SWAP RDS	PGS/SEC	7	[16 15 8 9 3 6 7 6 11 5 6]	3	ERRATIC	VALUES				>40
SWAP WRS	PGS/SEC	6	[16 14 9 10 4 6 7 6 10 5 6]	2	ERRATIC	VALUES				>40
% BS WQ	RD WAIT	62	[8 7 9 8 8 9 9 9 7 8 10]	72	ERRATIC	VALUES				
% BS WQ	WR WAIT	10	[6 9 7 9 12 9 8 9 8 10 6]	11	ERRATIC	VALUES				
% IDLE	IO TIME	11	[10 9 9 8 9 8 9 9 10 9 4]	5	ERRATIC	VALUES				
# FREE	MEM PGS	111	[2 6 5 4 19 8 7 10 7 12 12]	673	ERRATIC	VALUES				<25
GEN FREE	PGS/SEC	8	[15 14 8 9 4 6 7 6 10 5 6]	4	ERRATIC	VALUES				>20
% IDLE	TIME	11	[5 9 6 9 16 11 8 9 9 10 3]	7	ERRATIC	VALUES				<10
BIAS	CONTROL	11	[8 8 8 8 8 8 8 8 8 9 8]	11	ERRATIC	VALUES				
FRK FREE	PGS/SEC	.0	[8 8 8 8 8 8 8 8 8 8 8]	.0	ERRATIC	VALUES				>5.0
# LOCKED	PAGES	6	[8 8 8 8 8 8 8 8 8 8 8]	6	ERRATIC	VALUES				>500
# MTAS	ASSIGNED	0	[5 8 15 6 7 5 8 5 9 6 16]	1	ERRATIC	VALUES				
# BALNCE	SET FRKS	3	[8 8 8 7 4 6 8 7 6 7 20]	8	ERRATIC	VALUES				>20
MEM MGMT	CYCS/SEC	.1	[10 10 7 7 7 7 7 7 8 7 7]	.0	ERRATIC	VALUES				>10.0
WORK SET	LDS/SEC (F)	.3	[14 11 9 11 5 7 8 7 10 6 6]	.1	ERRATIC	VALUES				>1.0
BAL SET	SWPS/SEC	.0	[8 8 8 8 8 8 8 8 8 8 8]	.0	ERRATIC	VALUES				>1.0
# ACTIVE	FORKS	3	[8 8 8 7 4 6 8 7 6 7 20]	8	ERRATIC	VALUES				>30
PG FAULT	TRPS/SEC	40	[9 9 9 8 6 7 8 8 8 8 11]	47	ERRATIC	VALUES				>50
# MEMORY	WRK SETS	49	[3 5 6 6 10 10 10 10 7 10 10]	152	ERRATIC	VALUES				
# PTYS	IN USE	9	[8 8 9 8 8 8 8 9 8 8 9]	10	ERRATIC	VALUES				>40
RQ SAVES	PGS/SEC	8.9	[7 7 10 7 7 7 8 9 8 9 11]	15.3	ERRATIC	VALUES				
# SHARED	PAGES	2560	[25 19 15 15 0 4 4 1 13 3 1]	72	ERRATIC	VALUES				
% SCHED	TIME	9	[9 9 9 9 6 7 9 8 8 8 10]	10	ERRATIC	VALUES				>10
% SWAP	SPC FREE	45	[6 7 7 7 10 9 8 9 8 9 9]	65	ERRATIC	VALUES				<10
% BS WQ	SWP WAIT	22	[13 12 9 11 4 8 8 6 11 5 6]	10	ERRATIC	VALUES				
% IDLE	SWP TIME	3	[14 12 9 11 4 8 8 6 13 5 3]	1	ERRATIC	VALUES				
% MGMT	MEM TIME	.2	[8 9 8 8 7 8 9 8 8 8 9]	.2	ERRATIC	VALUES				>1.0
% PAGING	TIME	5	[10 9 9 8 5 7 8 8 8 7 11]	6	ERRATIC	VALUES				>10
TTY INTR	CHRS/SEC	0	[8 8 8 8 4 6 6 8 6 4 8]	0	ERRATIC	VALUES				>10
TTY IN	CHRS/SEC	59	[42 6 6 7 10 6 6 6 5 6 5]	8	ERRATIC	VALUES				>100
TTY OUT	CHRS/SEC	511	[8 7 8 9 7 9 9 9 8 8 9]	589	ERRATIC	VALUES				>500
# LINES	IN USE	50	[9 8 8 9 7 9 9 8 8 7 8]	48	ERRATIC	VALUES				>60
PGS USER	MEMORY	1985	[4 6 7 7 10 9 9 10 7 10 10]	4470	ERRATIC	VALUES				<768

PREDICTIONS ARE ONLY MADE USING MONTHLY DATA.
GIVEN THE CURRENT TREND, THERE IS A 90% CHANCE THAT THE ACTUAL VALUES WILL FALL WITHIN THE PREDICTED RANGES.
UNCHANGING VALUES: REFERS TO RELATIVELY CONSTANT VALUES. ERRATIC VALUES: REFERS TO THE LACK OF A STRONG LINEAR PATTERN IN THE DATA.
* = EXCEEDS LONG TERM LIMIT > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE ----------

REMAINDER OF REPORT OMITTED FROM THIS EXAMPLE.

Figure 1-6 (continued)

1.3.3 'Typical Day' Report

The 'Typical Day' Report can be generated weekly or monthly. The format of the report is identical for both the weekly and monthly versions, only the summary level of the data varies (weekly averages on the weekly report; monthly averages on the monthly report.) A monthly report is used as a sample here. 'Typical day' Reports are produced in pairs. The first report in the pair (shown in Figure 1-7) represents an average workday by combining data for Monday thru Friday, excluding holidays, of the week/month. The other report (not shown) combines data for Saturday, Sunday, and any holidays in the week/month. The method of reporting data in the 'Typical Day' Report corresponds most closely with the method of retaining and reporting on data in Workload AMAR. That is, all 24 hours in the day are represented and the breakdown is not by prime vs. non-prime time but by weekday vs. weekend and holiday.

The 'Typical Day' Reports are intended primarily to aid in load leveling machine usage. These reports show "typical" slack periods (where users could get more work done) and "typical" busy periods (where resources are at a premium and performance may be poor.) By using the 'Typical Day' Reports in conjunction with the corresponding workload reports, data center management can select the appropriate work to shift to less busy periods and thus make better use of system resources while improving overall thruput of the machine.

You should be aware, however, that the 'Typical Day' Reports intentionally smooth out the random day-to-day variations. They may also mask certain systematic variations such as a heavy processing day once a week (for example,the day before the payroll is due) or a heavy processing week once a month (for example, a monthly financial closing.) To identify patterns such as these, you should check the Weekly or Monthly Utilization Reports or the Weekly Trend Analysis Report.

The contents of the 'Typical Day' Report differs from that of the Daily System Utilization Report in the following areas (refer to the section on the Daily System Utilization Report for a description of the major report features and the format):

The 'Typical Day' Report contains a header (A) surrounded by dashes which identifies the report and denotes whether it represents data for weekdays (B) or weekends and holidays.

On the 'Typical Day' Report, there is no list of disk packs showing free space today, yesterday, and the difference; there is no list of periods of downtime; and the number of reloads is not given.

Hourly values on the 'Typical Day' Report are computed by averaging the values of the corresponding time period from each of the days in the reporting interval. For example, on

page 4 of the sample report, the value for # ACTIVE FORKS (C) for 10:00 AM is computed by averaging the 10:00 AM values for each Monday through Friday (minus holidays) in the month.

NO. OF HOURS KEY ITEMS OVER LIMITS (D) on page 1, # ITEMS OVER LIMITS (E) on page 4, and #HOURS OVER LIMITS (F) also on page 4 refer to the total for the "typical" day not the total during the period measured.

Long term (G) rather than short term thresholds are used.

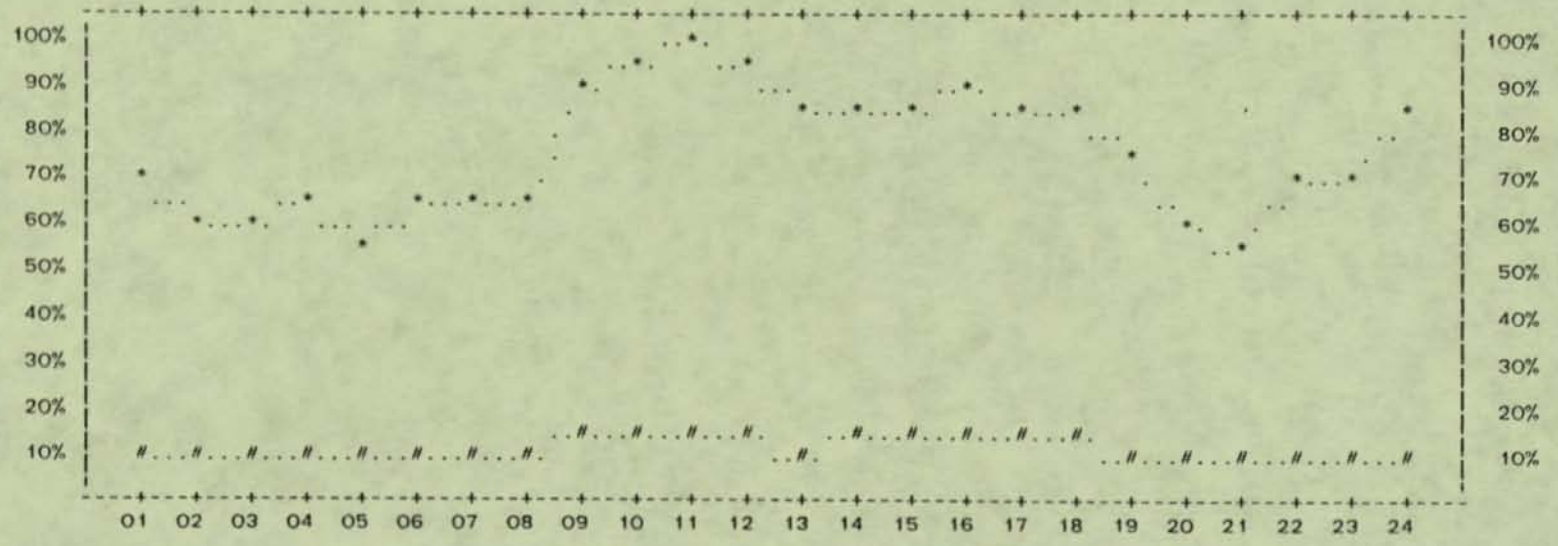
FROM: 03-OCT-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
MONTHLY 'TYPICAL DAY' SUMMARY REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

(A) THIS REPORT REPRESENTS AN AVERAGE DAY
OBTAINED BY COMBINING DATA FOR
(B) MONDAY THRU FRIDAY EXCEPT HOLIDAYS

--------- CPU UTILIZATION (*) OVERHEAD (#) AND BOTH (@) ---------



---------- TYPICAL HOUR ----------

Figure 1-7

---------- SUMMARY OF KEY UTILIZATION ITEMS ----------

-------AVERAGE-----	# ACTIVE FORKS	# MEMORY WRK SETS	# FORKS BS WAIT	BAL SET ADJS/SEC	# LINES IN USE	# FORK WAKS/SEC	% IDLE TIME	% USED TIME	% BACK GND TIME	% SCHED TIME	(D) NO. OF HOURS KEY ITEMS OVER LIMITS
----PRIME TIME-----	8	152	2	.9	48	10	7	76	3	10	19
---NON-PRIME TIME--	1	46	1	.7	4	5	25	57	2	6	5

+ = MORE THAN YESTERDAY - = LESS THAN YESTERDAY

---------- CONTINUED NEXT PAGE ---------

FROM: 03-OCT-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
MONTHLY 'TYPICAL DAY' SUMMARY REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

SUMMARY OF KEY UTILIZATION ITEMS

-------AVERAGE-----	% PAGING TIME	PG FAULT TRPS/SEC	CONTEXT SWTS/SEC	SWAP RDS PGS/SEC	SWAP WRS PGS/SEC	DSK RDS PGS/SEC	DSK WRS PGS/SEC	# MTAS ASSIGNED	# FREE MEM PGS	PGS USER MEMORY	NO. OF HOURS KEY ITEMS OVER LIMITS
----PRIME TIME-----	6	47	44	3	2	23	10	1	673	4470	9
---NON-PRIME TIME--	6	55	25	0	1	23	11	1	3218	4529	11

Figure 1-7 (continued)

FROM: 03-OCT-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
MONTHLY 'TYPICAL DAY' PROBLEM REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

PROBLEM PERIODS

EACH * = 1 KEY ITEM OVER WATCHDOG LIMIT
EACH + = 1 OTHER ITEM OVER LIMIT

(SEE THE FIRST PAGE OF DETAIL REPORT)
(FOR HOURLY AVERAGES OF KEY ITEMS)

PROBLEM RESOURCES

EACH P = 1 PRIME HOUR WHEN THE ITEM WAS OVER THE WATCHDOG LIMIT
EACH N = 1 NON-PRIME HOUR WHEN THE ITEM WAS OVER LIMIT

-----HOUR----	------NO. OF ITEMS------	--------ITEM--------	----NO. OF HOURS--------	---------- COMMENTS ----------
00:00 - 01:00	*	# FORKS BS WAIT	PPPPPPPN	CRITICAL CPU BOTTLENECK OR SCHEDULER SLOW
01:00 - 02:00		BAL SET ADJS/SEC	PP	SERIOUS SCHEDULER THRASHING: FIND REASON
02:00 - 03:00		# FORK WAKS/SEC	N	WARNING TOO MANY WAKEUPS: FIND CAUSE
03:00 - 04:00		% IDLE TIME	PPPPPPNN	CRITICAL CPU PRESSED: CHK WORKLD DATA FIRST
04:00 - 05:00		% SCHED TIME	PPPPN	SERIOUS SCHEDULER PRESSED: FIND CAUSE
05:00 - 06:00	*	PG FAULT TRPS/SEC	PPPNNNNNNNN	SERIOUS PAGING TOO OFTEN: CHK WORKLD DATA
06:00 - 07:00		CONTEXT SWTS/SEC	PPPPPPNN	CRITICAL SCHEDULER TOO FAST: CHK BLOCKING
07:00 - 08:00	+++	DSK RDS PGS/SEC	N	WARNING DISK READ RATE HIGH: CHK PACK I/O
08:00 - 09:00	+++	TTY OUT CHRS/SEC	PPPPPPP	CRITICAL TTY CHAR RATE TOO HIGH: FIND CAUSE
09:00 - 10:00	++++++	# WORK SET PGS	PPPPPPPPPNNNN	CRITICAL PGM SIZES LARGER THAN PLANNED
10:00 - 11:00	+++++++			
11:00 - 12:00	++++++++			
12:00 - 13:00	++++			
13:00 - 14:00	++++			
14:00 - 15:00	++++			
15:00 - 16:00	+++++++			
16:00 - 17:00	+			
17:00 - 18:00	+++++++			
18:00 - 19:00	++			
19:00 - 20:00	++			
20:00 - 21:00				
21:00 - 22:00	*			
22:00 - 23:00	*			
23:00 - 24:00	**			

SYSTEM AVAILABILITY SUMMARY

	SYSTEM UPTIME	% TIME MEASURED
PRIME TIME :	90.8%	85.2%
NON-PRIME TIME:	82.1%	78.3%

Figure 1-7 (continued)

FROM: 03-OCT-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
MONTHLY 'TYPICAL DAY' DETAIL REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

(E)

PERIOD	KEY UTILIZATION ITEMS: # ACTIVE FORKS	# MEMORY WRK SETS	# FORKS BS WAIT	BAL SET ADJS/SEC	# LINES IN USE	# FORK WAKS/SEC	% IDLE TIME	% USED TIME	% BACK GND TIME	% SCHED TIME	# ITEMS OVER LIMITS: THIS PAGE	ALL PAGES
00:00 - 01:00	1	41	0	.7	2	2	27	59	2	5		1
01:00 - 02:00	1	42	0	.7	2	2	31	51	2	4		
02:00 - 03:00	1	43	0	.6	1	2	34	52	2	4		
03:00 - 04:00	1	43	0	.6	2	2	32	56	2	4		
04:00 - 05:00	1	42	0	.6	1	2	38	49	2	4		
05:00 - 06:00	1	45	0	.7	1	2	28	55	2	5		1
06:00 - 07:00	1	48	0	.7	2	2	27	56	3	4		
07:00 - 08:00	1	67	0	.6	8	26 *	30	53	3	7	1	3
08:00 - 09:00	9	135	1	.9	37	8	7 *	76	3	9	1	3
09:00 - 10:00	14	159	2 *	.9	55	11	4 *	79	3	11 *	3	6
10:00 - 11:00	12 (C)	165	2 *	1.0 *	58	11	1 *	83	3	11 *	4	7
11:00 - 12:00	11	159	2 *	1.0 *	56	10	2 *	82	2	11 *	4	8
12:00 - 13:00	7	135	2 *	.8	42	9	10 *	74	2	8	2	4
13:00 - 14:00	6	156	2 *	.8	48	11	12	69	2	9	1	4
14:00 - 15:00	5	165	2 *	.8	52	10	12	69	2	9	1	4
15:00 - 16:00	7	164	2 *	.9	50	9	7 *	76	2	10 *	3	7
16:00 - 17:00	5	135	1	.8	35	8	12	72	2	8		1
17:00 - 18:00	3	69	2 *	.9	12	6	7 *	73	2	10 *	3	7
18:00 - 19:00	2	50	1	.8	7	5	15	64	2	8		2
19:00 - 20:00	1	42	0	.6	5	4	33	50	2	5		2
20:00 - 21:00	1	40	0	.6	3	3	32	47	2	6		
21:00 - 22:00	1	36	1	.8	3	3	17	61	2	7		1
22:00 - 23:00	2	37	1	.8	2	3	21	61	2	5		1
23:00 - 24:00	2	40	1	.9	2	2	8 *	73	3	6	1	2
----PRIME TIME-----												
AVERAGE VALUE:	8	152	2	.9	48	10	7	76	3	10		
LONG TERM LIMIT:	>30	NONE	>2	>1.0	>60	>20	<10%	>90%	>5%	>10%		
% TIME OVER LIMIT:			48.6%	82.7%	30.1%		84.0%		22.4%	52.2%		
# HOURS OVER LIMIT:			7	2			6			4	19	44 (F)
--NON-PRIME TIME---												
AVERAGE VALUE:	1	46	1	.7	4	5	25	57	2	6		
LONG TERM LIMIT:	>30	NONE	>2	>1.0	>60	>20	<10%	>90%	>5%	>10% (G)		
% TIME OVER LIMIT:			5.8%	69.4%		.4%	69.8%			14.2%		
# HOURS OVER LIMIT:			1			1	2			1	5	20

* = OVER LIMITS > = GREATER THAN OR EQUAL TO < = LESS THAN OR EQUAL TO

---------- CONTINUED NEXT PAGE --------

THE REMAINING PAGES IN THIS EXAMPLE ARE OMITTED

Figure 1-7 (continued)

1.3.4 Disk Reports

Disk Reports can be generated daily, weekly, or monthly. The formats are identical, only the length of report period varies. A Monthly Disk Report is used as a sample here. Refer to Figure 1-8. The Disk Report provides the following features:

All disk related information is presented in one place for ease of analysis.

System wide information such as system uptime, swapping rate, etc., is broken out from individual disk pack information such as mount time, free space, read, seeks, writes, etc.

Data is organized by logical pack name for ease of reference.

Mount time and in use time are provided and are expressed both in hours and minutes and as percentages of AMAR measured time.

The Disk Report is separated into three sections - a General Usage Summary and Prime/Non-prime Time Pack Summaries. The format for the report header is the same as that of the Trend Analysis Report. Please refer to that section for a description. The General Usage Summary follows the report header and is located in the upper left hand corner of the first page. The contents of the General Usage Summary are fixed. That is, it always contains the following information:

The number of hours (A) theoretically available in the reporting period (prime time followed by non-prime time) assuming the system was up 24 hours a day.

The number of hours and minutes the system was detected by AMAR as being up (B). This time may not be 100% accurate if the system came up and then went down again before AMAR was restarted. This time will always represent the minimum uptime possible, i.e., the actual uptime may be greater than that shown here.

The number of hours and minutes AMAR measured the system (C). This time should always be accurate.

The total number of disk pages read per second system-wide (D).

The total number of disk pages written per second system-wide (E).

The total number of swap reads per second (F) during the reporting period.

The total number of swap writes per second (G) during the reporting period.

The average percent of swap space left (H) during the reporting period.

The Prime Time Pack Summary follows the General Usage Summary. The first column (I) gives the logical pack name. Packs are listed in alphabetical order, one line per pack.

Following the pack name is a pair of columns, TOTAL TIME (HH:MM) MOUNTED-IN USE, which show the total time, in hours and minutes, that the pack was mounted (J) and "in use" (K). A pack is considered "in use" during a minute if at least one page is read from it or written to it during that minute. The disk report will also include physical unit information if it is collected and specified in the RFD file. The next pair of columns, % OF TIME MOUNTED-IN USE, expresses mounted time (L) and "in use" time (M) as a percentage of the time AMAR measured the system not the system uptime. The values in these two pairs of columns may be anywhere from 1% to 3% low because they are based on an assumed sample count of 60 samples per hour. In actual practice, slightly fewer samples may occasionally be taken , especially if the machine is very heavily used.

The single column, % MOUNTED TIME IN USE (N), gives the percent of mounted time that the pack was actually being used. It is derived by dividing the first "in use" time (K) by the mount time (J). This is a better indication of how heavily a pack is being used than the "in use" percentage (M) which is based on the whole period of measured time regardless of the mount status of the pack. If a pack is mounted a large percentage of the time and in use a relatively small percentage, it is important to know whether that usage occurred in one chunk or was scattered throughout the reporting interval before attempting to free up the spindle. This can be determined by examining the hourly averages for the period in question with the AMARON Online Inquiry Program.

The remaining columns (P) in the Prime Time Pack Summary are items directly measured by AMAR and selected for display on this report via the xxxxDR.RFD file. For a definition of these items please refer to the Appendix called "System AMAR Item Definitions".

The Non-prime Time Pack Summary follows the Prime Time Pack Summary and is identical in format to it.

FROM: 03-OCT-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
MONTHLY DISK REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

GENERAL USAGE SUMMARY

	PRIME TIME	NON-PRIME TIME
HOURS THEORETICALLY AVAILABLE	180:00	492:00 (A)
HH:MM SYSTEM WAS UP	163:22	425:23 (B)
HH:MM AMAR MEASURED THE SYSTEM	153:22	409:29 (C)
DSK RDS PGS/SEC	23	18 (D)
DSK WRS PGS/SEC	10	8 (E)
SWAP RDS PGS/SEC	3	0 (F)
SWAP WRS PGS/SEC	2	0 (G)
% SWAP SPC FREE (H)		

PRIME TIME PACK SUMMARY

(P)

PACK NAME (I)	TOTAL TIME(HH:MM) MOUNTED (J)	TOTAL TIME(HH:MM) IN USE (K)	% OF TIME MOUNTED (L)	% OF TIME IN USE (M)	% MOUNTED TIME IN USE (N)	PACK FREE SPC	PACK RDS/SEC	PACK SEKS/SEC	PACK WRS/SEC
ACCT 0	67:00	1:30	44%	1%	3%	42%	0	0	0
ARD 0	151:51	145:13	99%	95%	96%	18%	9	5	1
ARD2 0	9:30	0:05	6%	0%	1%	94%	0	0	0
ARDBAK0	128:58	21:47	84%	14%	17%	26%	0	0	1
ARDBAK1	128:58	19:00	84%	12%	15%		0	0	1
CUF 0	12:45	1:13	8%	1%	10%	17%	1	0	0
DSKR 0	151:51	98:35	99%	64%	65%	60%	1	0	1
DSKT 0	151:48	141:13	99%	92%	93%	24%	3	1	0
DSKT 1	151:48	143:15	99%	93%	94%		3	1	2
DSKW 0	151:51	123:20	99%	80%	81%	27%	3	1	1
IRA 0	151:48	60:47	99%	40%	40%	29%	1	0	0
KLAD 0	0:20	0:15	0%	0%	75%	52%	1	0	0
MAP 0	151:51	123:07	99%	80%	81%	43%	1	1	1
PS 0	151:06	149:59	99%	98%	99%	17%	12	2	11
PS 1	151:51	150:43	99%	98%	99%		2	2	1
TEST1 0	12:29	8:32	8%	6%	68%	53%	0	0	1
TEST1 1	12:29	1:21	8%	1%	11%		0	0	2
TEST2 0	12:14	1:27	8%	1%	12%	32%	0	0	3
TEST3 0	78:45	18:22	51%	12%	23%	40%	0	0	2
USRT 0	151:48	69:46	99%	45%	46%	35%	0	0	0
USRT 1	151:48	67:51	99%	44%	45%		0	0	0

NON-PRIME TIME PACK SUMMARY

PACK NAME	TOTAL TIME(HH:MM) MOUNTED	TOTAL TIME(HH:MM) IN USE	% OF TIME MOUNTED	% OF TIME IN USE	% MOUNTED TIME IN USE	PACK FREE SPC	PACK RDS/SEC	PACK SEKS/SEC	PACK WRS/SEC
ACCT 0	173:31	0:53	42%	0%	1%	42%	0	0	0
ARD 0	406:20	201:12	99%	49%	50%	20%	5	1	1
ARD2 0	41:51	0:12	10%	0%	0%	94%	0	0	0
ARDBAK0	331:51	9:29	81%	2%	3%	20%	0	0	0
ARDBAK1	331:51	10:27	81%	3%	3%		0	0	0

---------- CONTINUED NEXT PAGE ----------

Figure 1-8

FROM: 03-OCT-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
MONTHLY DISK REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

NON-PRIME TIME PACK SUMMARY

PACK NAME	TOTAL TIME(HH:MM) MOUNTED - IN USE	% OF TIME MOUNTED - IN USE	% MOUNTED TIME IN USE	PACK FREE SPC	PACK RDS/SEC	PACK SEKS/SEC	PACK WRS/SEC
CHECKD0	0:58 - 0:26	0% - 0%	45%	61%	0	0	18
CHECKD1	0:36 - 0:13	0% - 0%	36%		0	0	0
CUF 0	46:40 - 1:46	11% - 0%	4%	15%	0	0	0
DSKR 0	406:20 - 149:53	99% - 37%	37%	57%	3	1	2
DSKT 0	405:53 - 105:02	99% - 26%	26%	28%	1	0	1
DSKT 1	405:29 - 123:24	99% - 30%	30%		1	0	1
DSKW 0	405:58 - 168:42	99% - 41%	42%	27%	3	1	2
IRA 0	406:20 - 20:24	99% - 5%	5%	31%	1	0	0
MAP 0	406:20 - 41:00	99% - 10%	10%	43%	1	0	0
PS 0	406:20 - 245:26	99% - 60%	60%	16%	1	1	1
PS 1	406:20 - 240:08	99% - 59%	59%		1	0	0
TEST1 0	18:16 - 6:36	4% - 2%	36%	45%	0	0	0
TEST1 1	18:16 - 2:09	4% - 1%	12%		1	0	0
TEST2 0	18:52 - 0:52	5% - 0%	5%	23%	0	0	0
TEST3 0	198:06 - 8:04	48% - 2%	4%	37%	0	0	0
USRT 0	406:20 - 33:09	99% - 8%	8%	36%	0	0	0
USRT 1	406:20 - 33:29	99% - 8%	8%		0	0	0

Figure 1-8 (continued)

1.3.5 Tape Reports

Tape Reports can be generated daily, weekly, or monthly. The formats are identical, only the length of the report period varies. A Monthly Tape Report is used as a sample here. Refer to Figure 1-9. The Tape Report provides the following features:

All tape related information is presented in one place for ease of analysis.

Data is organized sequentially by tape drive number for ease of reference.

Mount time and in use time for individual drives are provided and expressed both in hours and minutes and as percentages of AMAR measured time.

The Tape Report is similar to the Disk Report. The Tape Report is separated into three sections - a General Usage Summary and Prime/Non-prime Time Tape Summaries. The format of the report header is identical to that of the Trend Analysis Report. Please refer to that section for its description. The General Usage Summary follows the report header and is located in the upper left hand corner of the first page. The General Usage Summary always contains the following information:

The number of hours (A) theoretically available in the report period (prime time followed by non-prime time), assuming the system was up 24 hours a day.

The number of hours and minutes AMAR detected the system as being up (B). This time may not be 100% accurate if the system came up and then went down again before AMAR was restarted. This time will always represent the minimum uptime possible, i.e., the actual uptime may be greater than that shown here.

The number of hours and minutes AMAR measured the system (C). This time should always be accurate.

The average number of tape drives assigned (D) during the reporting period.

After the General Usage Summary comes the Prime Time Tape Summary. The first column (E) gives the drive number. Drives are listed sequentially, one line per drive. Following the drive number is a pair of columns, TOTAL TIME (HH:MM) ASSIGNED-IN USE, which shows, in hours and minutes, the total time that tapes were assigned to the drive (F) and actually were "in use" (G) on the drive. A tape drive is considered "in use" during a minute if at least one frame of data is read from it or written to it during that minute. The next pair of columns, % OF TIME ASSIGNED-IN USE, expresses assigned time (H) and "in use" time (I) as a percentage of the time AMAR measured the system <u>not</u> the system

uptime. The values in these two pairs of columns may be anywhere from 1% to 3% low because, as in the Disk Report, they are based on an assumed sample count of 60 samples per hour. In actual practice, slightly fewer samples may occasionally be taken, especially if the system is very heavily used.

The column, % ASSIGNED TIME IN USE (J), gives the percent of assigned time that a tape was actually being used. It is derived by dividing the first "in use" time (G) by the assigned time (F). This percentage can quickly indicate how much of the time drives may have been left assigned unnecessarily.

The final column, TAPE PGS/SEC (K), gives the transfer rate on each drive expressed in equivalent disk pages (512 words per page) per second. This page size is independent of any actual physical block size on the tape.

The set of three columns which are underneath those just described provide information on simultaneous tape usage. The data can be used to determine whether or not more drives are needed or if any excess drives can be eliminated. The first column (L) gives the possible number of drives which could be assigned at one time. This number will vary from 0 to the maximum mumber on the system. The next column (M) gives the percentage when exactly 0, 1, 2, etc. drives were assigned. The last column (N) gives the cumulative percentage of assigned drives, i.e., the percent of time when at most 0, 1, 2, etc. drives were assigned. In this sample, it can be seen that during prime time there were no drives in use more than half (55%) of the time (P).

In computer centers where tape drives are shared between systems via dual porting, you should check both sets of Tape Reports in order to correctly analyze drive usage.

FROM: 03-OCT-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
MONTHLY TAPE REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

GENERAL USAGE SUMMARY

	PRIME TIME	NON-PRIME TIME
HOURS THEORETICALLY AVAILABLE	180:00	492:00 (A)
HH:MM SYSTEM WAS UP	163:22	425:23 (B)
HH:MM AMAR MEASURED THE SYSTEM	153:22	409:29 (C)
# MTAS ASSIGNED	1	1 (D)

PRIME TIME TAPE SUMMARY

TAPE DRIVE	TOTAL TIME(HH:MM) ASSIGNED (F)	IN USE (G)	% OF TIME ASSIGNED (H)	IN USE (I)	% ASSIGNED TIME IN USE (J)	TAPE PGS/SEC (K)
MTA0	58:13	7:46	38%	5%	13%	1
MTA1	33:22	5:33	22%	4%	17%	2
MTA2	31:26	18:36	20%	12%	59%	4
MTA3	20:49	14:19	14%	9%	69%	7
MTA4	5:04	3:24	3%	2%	67%	10
MTA5	0:09	0:03	0%	0%	33%	0

NUMBER OF DRIVES ASSIGNED SIMULTANEOUSLY (L)	% OF TIME WHEN EXACTLY THIS NO. WERE ASSIGNED (M)	% OF TIME WHEN AT MOST THIS NO. WERE ASSIGNED (N)
0	55% (P)	55%
1	15%	70%
2	15%	84%
3	9%	93%
4	6%	99%
5	1%	100%
6	0%	100%

NON-PRIME TIME TAPE SUMMARY

TAPE DRIVE	TOTAL TIME(HH:MM) ASSIGNED	IN USE	% OF TIME ASSIGNED	IN USE	% ASSIGNED TIME IN USE	TAPE PGS/SEC
MTA0	196:24	33:34	48%	8%	17%	3
MTA1	92:49	44:16	23%	11%	48%	7
MTA2	73:35	52:38	18%	13%	72%	9
MTA3	45:21	32:59	11%	8%	73%	13
MTA4	9:31	6:11	2%	2%	65%	13
MTA5	2:01	0:55	0%	0%	45%	0

---------- CONTINUED NEXT PAGE ----------

Figure 1-9

FROM: 03-OCT-82 (SUNDAY)
THRU: 30-OCT-82 (SATURDAY)

- AMAR -
MONTHLY TAPE REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

NUMBER OF DRIVES ASSIGNED SIMULTANEOUSLY	% OF TIME WHEN EXACTLY THIS NO. WERE ASSIGNED	% OF TIME WHEN AT MOST THIS NO. WERE ASSIGNED
0	47%	47%
1	22%	68%
2	19%	87%
3	9%	96%
4	4%	99%
5	1%	100%
6	0%	100%

Figure 1-9 (continued)

1.3.6 Online Inquiry Reports

The AMARON online inquiry program produces two types of reports - Tables of Average Values and Histograms - which can either be displayed at the terminal or stored in a file for later printing or for processing by user programs.

1.3.6.1 Table Of Average Values -

Refer to the report sample in Figure 1-10. For a detailed explanation of the report dialogue shown in the sample, refer to the Appendix called "Online Inquiry Program (AMARON) Dialogue".

The format of the Table of Average Values is as follows:

Line 1 - The first line in the report header contains the 4 character system code (A) and the starting (B) and ending (C) dates of the report period specified by you.

Line 2 - The second line of the report header identifies the date/time column and then gives the 4 character item (D) code or 7 character subitem code positioned over its associated data column. Refer to the Appendix called "System AMAR Item Definitions" for a list of valid items and their codes. Items and subitems are listed in the order specified by you. From 1 to 10 items may be specified on a single report.

Lines 3 - on - Contain:

The ending date (E) of the fiscal period reported in the format yymmdd (yy = normal calendar not fiscal year; mm = month; dd = day);

For hourly level data, the hour (F) in the format hhss (hh = hours; ss = minutes); and

The average value (G) for each item/subitem specified. Note that a value of -1 indicates that no data was gathered during that interval. For example, in the case of disk packs, the pack may not have been mounted.

Lines 1 and 2 (the report header) may be automatically deleted from the report if it is stored in a file rather than printed at the terminal. If the headers are deleted, you must be careful to retain a description of the report contents since there will be no internal identification.

```
@RU AM:AMARON
DATABASE NAME: AMAR
Histogram Function (Y/N): N
Output at (T)erminal or in (F)ile: T
Start Date: 821026
End Date: 821026
Granularity Level: H
Starting Hour: 07
Ending Hour: 17
(P)rime (N)on Prime (B)oth or (W)eekend: B
Item 1: CPIO
Item 2: BGND
Item 3: USED
Item 4: LURD?
LURD           LURDACCT  O LURDARD    O LURDARD    1 LURDARD2   O LURDARDBAKO
LURDARDBAK1 LURDCHECKDO LURDCHECKD1 LURDCUF     O LURDDSKR   O LURDDSKT   O
LURDDSKT  1 LURDDSKW  O LURDDSKW   1 LURDFOD    O LURDFOD    1 LURDIRA    O
LURDKLAD  O LURDMAP   O LURDMAP    1 LURDNEWRD  O LURDNEWRD  1 LURDOLD1   O
LURDOLD1  1 LURDPS    O LURDPS     1 LURDSPEC   O LURDSUP    O LURDSUPP   O
LURDTEST1 O LURDTEST1 1 LURDTEST2  O LURDTEST3  O LURDUSRT   O LURDUSRT   1
Item 4: LURDPS     O
Item 5: LUWRPS     O
Item 6:

                                      (A)
                                                               (B)        (C)
DATE  TIME          (D)CPIO    SYSTEM TTSS                    FROM 821026 TO 821026
 (E)   (F)                         BGND       USEDLURDPS      OLUWRPS    O
821026:0700          92.64(G)      4.10          2.91          .01        .09
821026:0800          58.02         4.16         34.83          .20        .37
821026:0900          10.84         4.47         75.89         1.64       1.47
821026:1000            .78         4.99         82.89         6.65      16.06
821026:1100           1.84         4.91         82.32         2.67       1.95
821026:1200           1.28         4.87         82.65        45.49      55.38
821026:1300          20.66         4.51         68.02         1.23       1.00
821026:1400          38.82         4.51         47.74         1.02       1.12
821026:1500          31.19         4.52         54.06         2.15       1.80
821026:1600           1.30         4.78         85.38         1.28       1.54
821026:1700          10.44         4.58         78.65          .67       1.07
More Requests (Y/N): N
EXIT
```

Figure 1-10

1.3.6.2 Histograms -

Refer to the report sample in Figure 1-11. For a detailed explanation of the report dialogue shown in the sample, refer to the Appendix called "Online Inquiry Program (AMARON) Dialogue".

The format of the histogram report is as follows:

Line 1 - The first line in the report header contains the 4 character system code (A) and the 4 character item or 7 character subitem code (B). Refer to the Appendix called "System AMAR Item Definitions" for a list of valid items and their codes. From 1 to 10 items/subitems may be selected in a single report. The histograms for each item/subitem are given in the same order as that specified in the dialogue. Histograms are listed chronologically according to fiscal period specified. That is, if 2 days worth of data for 4 items is requested, all of day 1's data will be given first followed by day 2's data.

Line 2 - Contains the end date (C) of the fiscal period represented by the data and the starting (D) and ending (E) dates of the report period specified by you. All dates are displayed in the format mm/dd/yy (mm = month; dd = day; yy = normal calendar not fiscal year).

Line 3 - Denotes whether the data represents prime or non-prime time (F) and, for hourly data, the hour (G) represented.

Lines 4 - 5 - The last lines of the report header contain the titles for each column.

The first column (H) gives the possible range of values for the item or subitem. The width of the value ranges (normally called classes) are defined in the system AMAR database at installation time. The class width for each item and subitem can be obtained by running the AMRGEN program using the LIST CLASS command. Zero (0) values are always treated as a separate class. This is different from ranges where no sample values fall. Such ranges are called blank ranges (I). Printing of blank ranges may be suppressed by you.

The range of values in the sample report represents % USED TIME. Again refer to Appendix B for a definition of each item and its data type, i.e., percentage, pages per second, etc.

The second column (J) lists the number of samples whose values fell within the range.

The third column (K), FREQUENCY % OF TOTAL, gives the percent of samples whose values fell within the range. Since a sample is typically a minute in length, this can be thought of as the percent of time that the values fell within the range.

```
@RU AM:AMARON
DATABASE NAME: AMAR
Histogram Function (Y/N): Y
Output at (T)erminal or in (F)ile: T
Start Date: 821026
End Date: 821026
Granularity Level: H
Starting Hour: 15
Ending Hour: 15
(P)rime (N)on Prime (B)oth or (W)eekend: P
Item 1: USED
Item 2:
Suppress Blank Ranges (Y/N)? Y
```

(A) SYSTEM: TISS (B) ITEM: USED (D) (E)
DAY 10/26/82 (C) IN RANGE 10/26/82 TO 10/26/82
(F) PRIMETIME TIME: 1500 (G)

RANGE OF VALUES (H) (I)	NUMBER OF SAMPLES (J)	FREQUENCY % OF TOTAL (K)	% OF CUMULATIVE TOTAL (L)
20.01 - 25.00	1	1.6%	1.6%
25.01 - 30.00	5	8.3%	9.9%
30.01 - 35.00	5	8.3%	18.3%
35.01 - 40.00	6	10.0%	28.3%
40.01 - 45.00	3	5.0%	33.3%
45.01 - 50.00	5	8.3%	41.6%
50.01 - 55.00	6	10.0%	51.6%
55.01 - 60.00	7	11.6%	63.3%
60.01 - 65.00	6	10.0%	73.3%
65.01 - 70.00	1	1.6%	74.9%
70.01 - 75.00	5	8.3%	83.3% (M)
75.01 - 80.00	3	5.0%	88.3%
80.01 - 85.00	5	8.3%	96.6%
85.01 - 90.00	2	3.3%	99.9%

AVERAGE VALUE FOR 60 SAMPLES IS 54.06
(N) (P)

```
More Requests (Y/N): N
EXIT
```

Figure 1-11

The final column (L), % OF CUMULATIVE TOTAL, gives the percent of all samples whose values fell into the previous ranges plus the percent of those samples whose values fell into the current range. For example, on the sample report, values for % USED TIME were less than or equal to 75%, 83.3% (M) of the time. Conversely, the values for % USED TIME were greater than 75%, 16.7% of the time (100%-83.3%).

At the bottom of the report, the total number of samples (N) and the average value (P) for the report period are given. The average value is the same value that would be displayed if you had elected to use a Table of Average Values instead of a Histogram for this item and time period.

blank page

1.3.7 Data Extraction Records

The Data Extraction Program, AMAREX, is used to extract four types of records from either the System AMAR database (AMAR.DB) or the output file produced by AMARSD (TODAY.DB). The record types are:

Performance Summary
Performance Detail
Granularity
System Calendar

The AMAREX program extracts data for display at the terminal or for input to your own programs. AMAREX eliminates the need for you to have to deal with System AMAR's internal database format.

1.3.7.1 Performance Summary (PS) Record -

The PS records contain mean (or average) values for specific time periods such as hours, days, weeks, etc. The number of samples taken during the period and their mean value are given. This particular type of record will be produced when AMAREX is run with the parameters shown in the example on the next page. Record type is the only critical parameter that must be typed as shown. The others can be varied. Refer to the Appendix called "Data Extraction Program (AMAREX) Dialogue" for further information.

	Field Description	Width
A.	Record format	2
B.	Record type (e.g., PS)	2
C.	Record length (blank)	4
D.	Blank	5
E.	Four-character system code	4
F.	Normal calendar year	2
G.	Normal calendar month	2
H.	Normal calendar day	2
I.	Hour (military time)	2
J.	Minutes	2
K.	Fiscal year	2
L.	Fiscal quarter	1
M.	Fiscal month*	1
N.	Fiscal week*	1
O.	Fiscal day*	1
P.	Granularity (summary level)	1
Q.	System AMAR item (e.g. BGND, CPIO)	4
R.	System AMAR subitem (i.e., item specified for a particular peripheral device)	9
S.	Prime indicator (P,N or blank)	1
T.	Count of samples	10
U.	Mean value (integer part)	8
V.	Mean value (decimal part)	2

```
@RU AM:AMAREX
DATABASE NAME:AMAR
OUTPUT:EXTRC1.TXT
RECORD TYPE:PS
DATE:821026:1501-821026:1600
[821026:1501-821026:1600 = 8321431501-8321431600]
GRANULARITY LEVEL:S
RESTRICTING ANY FISCAL PERIOD?NO
ITEM:CPIO,BGND,CTXS,LUWRPS     O,LUWRUSRT  O
PRIMETIME:ALL
[ EXTRACTING ]


[ SPECIFY NEXT EXTRACTION CRITERIA ]

OUTPUT:C
@
@TYP EXTRC1.TXT
AAPS          TTSS8210261600832143 1BGND          P000000006000000000478
AAPS          TTSS8210261600832143 1CPIO          P000000006000000000130
AAPS          TTSS8210261600832143 1CTXS          P000000006000000003984
AAPS          TTSS8210261600832143 1LUWRPS     O  P000000006000000000154
AAPS          TTSS8210261600832143 1LUWRUSRT   O  P000000006000000000026
```

A B C,D E F G H I J K L M N O P Q R S T U V

Figure 1-12

*If a higher granularity level is selected for reporting, these fields will contain the number representing the last day, week, or month (as appropriate) in the fiscal period specified.

1.3.7.2 Performance Detail (PD) Record -

The PD records contain frequency distribution data which can be used to produce histograms showing the percent of samples whose values fell into specific ranges. The number of samples and the mean (average) values for the range are given. The boundaries of the range must be derived from the class widths defined in the System AMAR database. The PD record will be produced when AMAREX is run with the parameters shown in the example on the next page. Record type is the only critical parameter that must be typed as shown. The others can be varied. Refer to the Appendix called "Data Extraction Program (AMAREX) Dialogue" for further information.

	Field Description	Width
A.	Record format	2
B.	Record type (e.g., PD)	2
C.	Record length (blank)	4
D.	Blank	5
E.	Four-character system code	4
F.	Normal Calendar year	2
G.	Normal Calendar month	2
H.	Normal Calendar day	2
I.	Hour (military time)	2
J.	Minutes	2
K.	Fiscal year	2
L.	Fiscal quarter	1
M.	Fiscal month*	1
N.	Fiscal week*	1
O.	Fiscal day*	1
P.	Granularity (summary level)	1
Q.	System AMAR item (e.g., CPIO)	4
R.	System AMAR subitem (i.e., item specified for a particular peripheral device)	9
S.	Prime indicator (P,N or blank)	1
T.	Count of samples	10
U.	Mean value (integer part) for the frequency class	8
V.	Mean value (decimal part)	2

*If a higher granularity level is selected for reporting, these fields will contain the number representing the last day, week, or month (as appropriate) in the fiscal period specified.

```
@RU AM:AMAREX
DATABASE NAME:AMAR
OUTPUT:EXTRC2.TXT
RECORD TYPE:PD
DATE:821026:1401-821026:1500
[821026:1401-821026:1500 = 8321431401-8321431500]
GRANULARITY LEVEL:S
RESTRICTING ANY FISCAL PERIOD?NO
ITEM:CPIO.LUWRPS     0
PRIMETIME:ALL
[ EXTRACTING ]


[ SPECIFY NEXT EXTRACTION CRITERIA ]

OUTPUT:C
@
@TYP EXTRC2.TXT
AAPD          TTSS8210261500832143 1CPIO          P000000000700000002 77
AAPD          TTSS8210261500832143 1CPIO          P000000000200000008 53
AAPD          TTSS8210261500832143 1CPIO          P000000000600000012 30
AAPD          TTSS8210261500832143 1CPIO          P000000000300000016 92
AAPD          TTSS8210261500832143 1CPIO          P000000000700000023 43
AAPD          TTSS8210261500832143 1CPIO          P000000000400000027 36
AAPD          TTSS8210261500832143 1CPIO          P000000000600000032 48
AAPD          TTSS8210261500832143 1CPIO          P000000000400000037 39
AAPD          TTSS8210261500832143 1CPIO          P000000000400000042 53
AAPD          TTSS8210261500832143 1CPIO          P000000000300000047 22
AAPD          TTSS8210261500832143 1CPIO          P000000000700000050 98
AAPD          TTSS8210261500832143 1CPIO          P000000000300000056 77
AAPD          TTSS8210261500832143 1CPIO          P000000000300000062 11
AAPD          TTSS8210261500832143 1CPIO          P000000000100000067 39
AAPD          TTSS8210261500832143 1LUWRPS 0      P000000000600000001 80
```

A | B | C,D | E | F | G | H | I | J | K | L M N O P | Q | R | S | T | U | V

Figure 1-13

1.3.7.3 Granularity (GR) Record -

The GR records contain a timestamp indicating the fiscal period for which data has been collected. This particular type of record will be produced when AMAREX is run with the parameters shown in the example on the next page. Record type is the only critical parameter that must be typed as shown. The others can be varied. Refer to the Appendix called "Data Extraction Program (AMAREX) Dialogue" for further information.

	Field Description	Width
A.	Record format	2
B.	Record type (e.g., GR)	2
C.	Record length (blank)	4
D.	Family (type of processor (blank))	5
E.	Four-character system code	4
F.	Normal calendar year	2
G.	Normal calendar month	2
H.	Normal calendar day	2
I.	Hour (military time)	2
J.	Minutes	2
K.	Fiscal year	2
L.	Fiscal quarter	1
M.	Fiscal month*	1
N.	Fiscal week*	1
O.	Fiscal day*	1
P.	Granularity (summary level)	1

*If a higher granularity level is selected for reporting, these fields will contain the number representing the last day, week, or month (as appropriate) in the fiscal period specified.

```
@RU AM:AMAREX
DATABASE NAME:AMAR
OUTPUT:EXTRC3.TXT
RECORD TYPE:GR
DATE:821026-821026
[821026:0001-821026:2400 = 8321430001-8321432400]
GRANULARITY LEVEL:S
RESTRICTING ANY FISCAL PERIOD?NO
[ EXTRACTING ]


[ SPECIFY NEXT EXTRACTION CRITERIA ]

OUTPUT:C
@
@TYP EXTRC3.TXT
AAGR          TTSS8210260100832143 1
AAGR          TTSS8210260200832143 1
AAGR          TTSS8210260300832143 1
AAGR          TTSS8210260400832143 1
AAGR          TTSS8210260500832143 1
AAGR          TTSS8210260600832143 1
AAGR          TTSS8210260700832143 1
AAGR          TTSS8210260800832143 1
AAGR          TTSS8210260900832143 1
AAGR          TTSS8210261000832143 1
AAGR          TTSS8210261100832143 1
AAGR          TTSS8210261200832143 1
AAGR          TTSS8210261300832143 1
AAGR          TTSS8210261400832143 1
AAGR          TTSS8210261500832143 1
AAGR          TTSS8210261600832143 1
AAGR          TTSS8210261700832143 1
AAGR          TTSS8210261800832143 1
AAGR          TTSS8210261900832143 1
AAGR          TTSS8210262000832143 1
AAGR          TTSS8210262100832143 1
AAGR          TTSS8210262200832143 1
AAGR          TTSS8210262300832143 1
AAGR          TTSS8210262400832143 1
```

A B C,D E F G H I J K L M N O P

Figure 1-14

1.3.7.4 System Calendar (SC) Record -

The SC records contain the date and time of a system reload and the system uptime. This particular type of record will be produced when AMAREX is run with the parameters shown in the example on the next page. Record type is the only critical parameter that must be typed as shown. The others can be varied. Refer to the Appendix called "Data Extraction Program (AMAREX) Dialogue" for further information.

	Field Description	Width
A.	Record format	2
B.	Record type	2
C.	Blank	13
D.	Date system reloaded (year, month, day)	6
E.	Time system reloaded (hours, minutes)	4
F.	Blank	46
G.	Zeros	10
H.	Blank	34
I.	Date/time of system reload*	12
J.	System up-time (seconds)	12

*The date/time is expressed as the number of seconds since 00:00:00 (midnight) on November 17,1858. If this number is divided by the number of seconds in a day, the quotient will be the left half of a National Bureau of Standards date/timestamp.

```
@RU AM:AMAREX
DATABASE NAME:AMAR
OUTPUT:EXTRC4.TXT
RECORD TYPE:SC
DATE:821026-821029
[821026:0001-821029:2400 = 8321430001-8321462400]
[ EXTRACTING ]

[ SPECIFY NEXT EXTRACTION CRITERIA ]

OUTPUT:C
@
@TYP EXTRC4.TXT
AASC              8210220545
0000000000                                   0039108303090000000531448
AASC              8210280927
0000000000                                   0039113620260000000001983
AASC              8210281000
0000000000                                   0039113640100000000032797
A  AASC|<-B   C   |D      |E  |     F
   AASC           |821028|2245|
   0000000000|                               |003911409954|000000188077
        G                 H                        I            J
```

Figure 1-15

1.4 HOW TO RUN THE PROGRAMS

1.4.1 Data Collection

The Data Collection Program, xxxxDC (where xxxx is the 4 character system code), should be run under SYSJOB. There will be a corresponding subjob to collect data for Workload AMAR.

The following commands should be inserted into the SYSJOB.RUN file to ensure automatic startup and continuous data collection.

```
JOB N \LOG amar-dir
ENABLE
CONNECT struc:amar-dir
RUN xxxxDC
\
```

These commands may be entered directly to SYSJOB or PTYCON to get xxxxDC started the first time. The third command line may be used to restart xxxxDC if it has stopped because of disk parity errors or the like.

Each day xxxxDC creates an output file named xxxxdd.mmm where xxxx is the 4 character system code; dd is the day; and mmm is the alpha abbreviation for the month. For example, on the "PATH" system, the raw file created on January 9th would be called PATH09.JAN. Since xxxxDC will write to the raw file throughout the day, the pack used for the raw file must be permanently mounted. Otherwise, data will be irretrievably lost.

1.4.2 Generating Automatic Reports

The AMREPT program, the xxxxDR.RFD file and the System AMAR database are used to generate automatic reports. AMREPT is normally run as part of a nightly batch stream, AMAR.CTL, which is self-submitting. xxxxDR.RFD contains all the commands used to define each report's contents. The System AMAR database keeps track of the records on which AMREPT has already reported. AMREPT checks the database to see when data for a fiscal day, week or month is ready for reporting. It then produces the appropriate daily, weekly and monthly reports. Several report requests are included in the AMAR.CTL stream as defaults. You can specify your own set of automatic reports by commenting in or commenting out the appropriate report codes and associated dialogue.

AMREPT is explained in detail in the Appendix called "Report Program (AMREPT) Dialogue". The appendix also describes how to use AMREPT to generate reports on demand. For quick reference, the dialogue to generate automatic reports is shown in Figure 1-16.

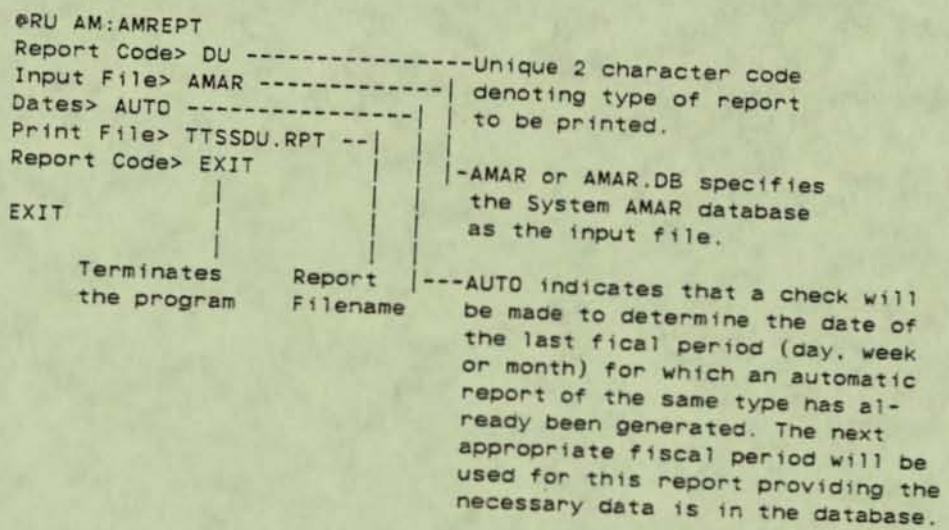


Figure 1-16

* * * * * * * * * *

1.4.3 Generating Special Reports - What Program Do I Use?

There are four programs for generating special reports - AMARSD, AMREPT, AMARON, and AMAREX.

Use AMREPT alone to produce standard reports from the database. See Figure 1-17 and the Appendix called "Report Program (AMREPT) Dialogue".

Use AMARSD and AMREPT together if you want to look at today's data which will not be in the database until after midnight. Also use AMARSD and AMREPT to look at any other daily raw file. See Figure 1-18 and the Appendix called "Raw File Preprocessor Program (AMARSD) Dialogue".

Use AMARON if you want to look at selected items from the database in either histogram or tabular format. Data can be examined directly at the terminal or put into files for later processing by your own programs or statistical packages. See Figures 1-10 and 1-11 in the Section called "Online Inquiry Reports" and the Appendix called "Online Inquiry Program (AMARON) Dialogue".

Use AMAREX as an alternative to AMARON for extracting data into sequential format for later processing by your own programs. See Figures 1-12 through 1-15 in the Section called "Data Extraction Records" and the Appendix called "Data Extraction Program (AMAREX) Dialogue".

* * * * * * * * * *

```
@RU AM:AMREPT
Report Code> DD ----------------Daily Disk Report
Input File> AMAR ------------|
Dates> 821026 -------------| |-System AMAR database
Print File> TTSSDD.RPX     |
Report Code> EXIT          |---October 26, 1982

EXIT
```

Dialogue to Produce a Standard Report
Figure 1-17

* * * * * * * * * *

FROM: 26-OCT-82 (TUESDAY)
THRU: 26-OCT-82 (TUESDAY)

- AMAR -
DAILY DISK REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

GENERAL USAGE SUMMARY

	PRIME TIME	NON-PRIME TIME
HOURS THEORETICALLY AVAILABLE	9:00	15:00
HH:MM SYSTEM WAS UP	9:00	15:00
HH:MM AMAR MEASURED THE SYSTEM	9:00	15:00
DSK RDS PGS/SEC	22	20
DSK WRS PGS/SEC	13	11
SWAP RDS PGS/SEC	3	0
SWAP WRS PGS/SEC	2	1
% SWAP SPC FREE		

PRIME TIME PACK SUMMARY

PACK NAME	TOTAL TIME(HH:MM) MOUNTED - IN USE	% OF TIME MOUNTED - IN USE	% MOUNTED TIME IN USE	PACK FREE SPC	PACK RDS/SEC	PACK SEKS/SEC	PACK WRS/SEC
ARD 0	8:56 - 8:55	99% - 99%	100%	14%	9	3	1
ARDBAK0	8:56 - 3:55	99% - 44%	44%	37%	0	0	2
ARDBAK1	8:56 - 4:09	99% - 46%	46%		1	0	3
DSKR 0	8:56 - 5:35	99% - 62%	63%	55%	1	0	0
DSKT 0	8:56 - 8:53	99% - 99%	99%	19%	3	1	0
DSKT 1	8:56 - 8:47	99% - 98%	98%		4	2	2
DSKW 0	8:56 - 7:26	99% - 83%	83%	19%	2	0	0
IRA 0	8:56 - 1:57	99% - 22%	22%	28%	0	0	1
MAP 0	8:56 - 6:46	99% - 75%	76%	43%	1	1	9
PS 0	8:53 - 8:51	99% - 98%	100%	31%	7	2	1
PS 1	8:56 - 8:56	99% - 99%	100%		2	2	3
TEST3 0	8:56 - 4:34	99% - 51%	51%	39%	0	0	0
USRT 0	8:56 - 4:56	99% - 55%	55%	26%	0	0	0
USRT 1	8:56 - 4:48	99% - 53%	54%		0	0	0

NON-PRIME TIME PACK SUMMARY

PACK NAME	TOTAL TIME(HH:MM) MOUNTED - IN USE	% OF TIME MOUNTED - IN USE	% MOUNTED TIME IN USE	PACK FREE SPC	PACK RDS/SEC	PACK SEKS/SEC	PACK WRS/SEC
ARD 0	14:53 - 8:56	99% - 60%	60%	19%	6	1	1
ARDBAK0	14:53 - 0:17	99% - 2%	2%	23%	0	0	0
ARDBAK1	14:53 - 0:17	99% - 2%	2%		0	0	0
DSKR 0	14:53 - 4:39	99% - 31%	31%	55%	2	0	2
DSKT 0	14:53 - 6:56	99% - 46%	47%	23%	3	1	2
DSKT 1	14:53 - 6:29	99% - 43%	44%		1	0	1
DSKW 0	14:53 - 7:00	99% - 47%	47%	17%	4	0	3
IRA 0	14:53 - 1:05	99% - 7%	7%	29%	1	0	0
MAP 0	14:53 - 1:44	99% - 12%	12%	44%	1	0	1
PS 0	14:53 - 10:19	99% - 69%	69%	31%	0	0	1
PS 1	14:53 - 9:54	99% - 66%	67%		0	0	0
TEST3 0	13:35 - 0:07	91% - 1%	1%	38%	0	0	0

---------- CONTINUED NEXT PAGE ----------

Figure 1-17 (continued)

* * * * * * * * * *

```
@RU AM:AMARSD
System ID>TTSS
YYMMDD Date of file>821027 -----------Raw file from Oct. 27,1982.
Prime periods>0800-1700                The raw filename is TTSS27.OCT.
[AMIHDS Hourly data stored for 821027]
[AMIDUD Database up to date]
[Use TODAY.DB as input file to the AMREPT report program]
          |
EXIT      |-----------
@                     |
@RU AM:AMREPT         |
Report Code> DU       |
Input File> TODAY --|----Mini-database created by AMARSD above and
Dates> 821027             used as the input file here.
Print File> TTSSXX.RPX
Report Code> exit
```

Dialogue to Look at Today's Data
Figure 1-18

DATE: 27-OCT-82 (WEDNESDAY)

- AMAR -
DAILY SYSTEM UTILIZATION SUMMARY REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

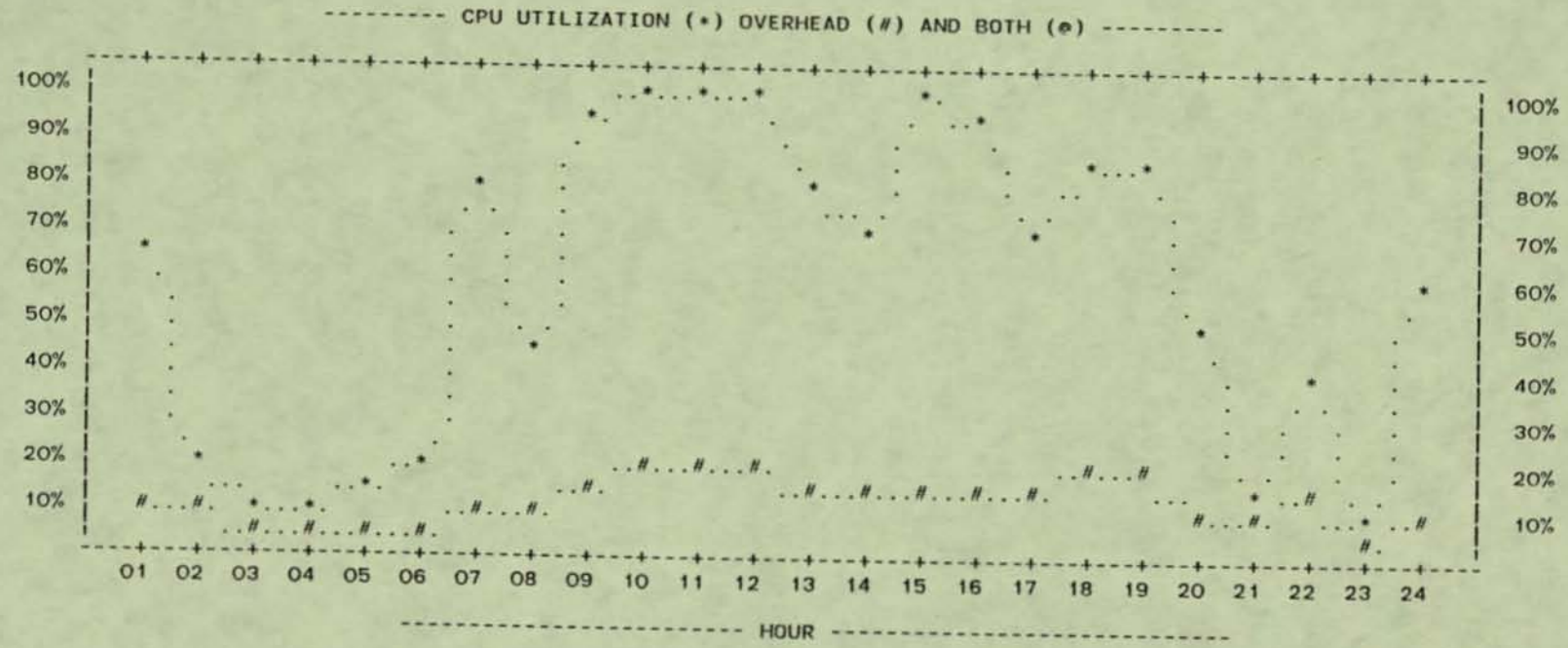


Figure 1-18 (continued)

---------- SUMMARY OF KEY UTILIZATION ITEMS ----------

-------AVERAGE-----	# ACTIVE FORKS	# MEMORY WRK SETS	# FORKS BS WAIT	BAL SET ADJS/SEC	# LINES IN USE	# FORK WAKS/SEC	% IDLE TIME	% USED TIME	% BACK GND TIME	% SCHED TIME	NO. OF HOURS KEY ITEMS OVER LIMITS
----PRIME TIME-----	6	164	3	.9	53	10	5	74	5	10	33
---NON-PRIME TIME--	1	40	0	.4	6	3	52	31	4	4	6

-------AVERAGE-----	% PAGING TIME	PG FAULT TRPS/SEC	CONTEXT SWTS/SEC	SWAP RDS PGS/SEC	SWAP WRS PGS/SEC	DSK RDS PGS/SEC	DSK WRS PGS/SEC	# MTAS ASSIGNED	# FREE MEM PGS	PGS USER MEMORY	NO. OF HOURS KEY ITEMS OVER LIMITS
----PRIME TIME-----	5	36	44	3	2	19	8	1	472	4542	6
---NON-PRIME TIME--	3	31	17	0	0	16	7	1	3289	4542	9

+ = MORE THAN YESTERDAY - = LESS THAN YESTERDAY

---------- CONTINUED NEXT PAGE ---------

DATE: 27-OCT-82 (WEDNESDAY)

- AMAR -
DAILY SYSTEM UTILIZATION SUMMARY REPORT

TOPS20 SAMPLE SYSTEM
SYSTEM: TTSS PRIME TIME: 0800 - 1700

PERCENT FREE SPACE LEFT ON DISK PACKS

-----PACK NAME-----	ARD	ARDBAK	DSKR	DSKT	DSKW	IRA	MAP	PS	TEST3	USRT
---% FREE TODAY----	17	25	45	18	17	31	44	29	42	24
-% FREE YESTERDAY--	18	29	55	22	16	27	44	32	37	26
----DIFFERENCE-----	- 1	- 4	-10	- 4	+ 1	+ 4	+ 0	- 3	+ 5	- 2

Figure 1-18 (continued)

+ = MORE THAN YESTERDAY - = LESS THAN YESTERDAY PACK DISMOUNTED ( ) UNMEASURED (__) WRONG DAY (?) BAD [illegible] (!)

1.4.4 Examining/Changing Database Parameters (AMRGEN)

The AMRGEN program allows you to examine certain parameters within the System AMAR database, notably data retention times, the prime time schedule, and the list of valid items and class widths. It may also be used to change data retention times. AMRGEN should be run before the end of each fiscal year to define the holiday list for the next year. Otherwise, holidays will be treated as normal workdays.

AMRGEN is command driven. It is procedural - certain commands depend on prior commands having been issued.

AMRGEN prompts with an asterisk (*).

Valid commands are:

```
SET RETENTION HOURLY <number of retained periods>
              DAILY
              WEEKLY
              MONTHLY
              COWEEKLY
              COMONTHLY
              LOG
```

Function: To specify retention times for each granularity (summary) level. Note that retention time directly affects the space required for the database. See the Appendix called "Installation and Resource Requirements" for space estimates.

<number of retained periods> = 1-3 digit retention count for a granularity level.

HOURLY = number of days hourly data is to be kept (default=7)

DAILY = number of days daily data is to be kept (default=35)

WEEKLY = number of weeks weekly data is to be kept (default=13)

MONTHLY = number of months monthly data is to be kept (default=12)

COWEEKLY = number of weeks weekly composite data is to be kept (default=5)

COMONTHLY = number of months monthly composite data is to be kept (default=3)

LOG = number of days uptime log records are to be kept (default=91)

LIST NAME

Function: To list the contents of the System Header Record which includes rollup date, retention times, last time automatic reports were generated, and date of last input.

SET WEEKDAY<hhmm-1>-<hhmm-2>,...,<hhmm-7>-<hhmm-8>

Function: To change the prime time schedule for all weekdays. Changes can only be made to dates for which no data has been entered. Up to 4 prime time pairs may be specified.

hh = hours; mm = minutes.

SET HOLIDAY <yymmdd>

Function: To define a given date <yymmdd> as a holiday. In addition to setting holidays for System AMAR, holidays must also be set for Workload AMAR. See the Section called "Examining/Changing the Workload Holidays (WCFIX)".

yy = normal calendar year; mm = month; dd = day.

LIST HOLIDAY

Function: To list the holidays.

DROP HOLIDAY <yymmdd>

Function: To remove a holiday from the holiday list.

yy = normal calendar year; mm = month; dd = day.

LIST PRIMETIMES <yymmdd-yymmdd>

Function: To list the prime times of the days within the date range.

yy = normal calendar year; mm = month; dd = day.

LIST CLASS <Item><Subitem>
<Item>
ALL

Function: To list one or more items and subitems and their class widths.

<Item> - 4 character item code

<Subitem> - 7 character subitem code

ALL - All items and subitems in the database.

ADD ITEM <Item><Subitem>

<Item>
ALL

Function: To enable an item and/or subitem for database storage. Only items already enabled in the data collection program should be named here. Don't make up random item or subitem names!

ALL - The rest of the valid items not yet enabled.

DROP ITEM <Item><Subitem>
<Item>

Function: To delete an item and/or subitem and all its data from the database.

EXIT DATABASE

Function: To gracefully terminate the program.

1.5 HOW TO TAILOR THE REPORT CONTENTS

1.5.1 Editing The Report File Description

The file called xxxxDR.RFD (where xxxx is the 4 character system code) contains information used to control the content, and, to some extent, the format of the System AMAR reports. Refer to Figure 1-19. The RFD file controls which items and subitems get printed on the reports, the titles used to describe these items and subitems, the short and long term thresholds used to test for potential problems, the contents of comment fields, and the paging of subreports. Through editing this file, you can tailor the reports to fit your site's special requirements.

Each field within the RFD file is described below. All fields are separated by tabs. For further information, you can also refer to the RFD.HLP file in your System AMAR area. Following the field descriptions are examples of some common changes to the RFD file. Changes will remain in effect until they are changed again by you -- there is no automatic resetting or reverting feature. Changes will only apply to the way the reports look. You will not be altering any data in the database or data collection programs.

System Description Section: This section contains system and site identification data and test criteria for printing severity codes in the comment field of the Daily System Utilization Report.

A. Command Code. The first 4 characters of each line indicate the purpose of the line. Only these command codes

are valid:

.C	-	Comment
.SD	-	System Description
.SS	-	System Specification
.ID	-	Item Description
.IDC	-	CPU Dependent Item Description
.RD	-	Report Description
.RI	-	Report Items

B. System Name. The 4 character system code which is used to identify this file, the data collection program, raw data files, etc.

C. System ID. The 20 character system description which appears in all standard report headings. This field must be delimited by underscores (_) and must be exactly 20 characters in length (blank fill if necessary).

D. Plot Graphics. Symbols printed for data points on all graphs produced by AMAR.

First symbol (*)	-	Denotes CPU utilization.
Second symbol (#)	-	Denotes overhead.
Third symbol (@)	-	Denotes where CPU utilization and overhead values are the same. CPU utilization includes overhead.

E. Sample Level Limits. These fields are used for testing the percent of samples over the watchdog limits and for specifying an appropriate code to denote the severity of the situation. The severity code is printed in the comments section on the "Problem Report" page of the Daily Utilization Summary Report. Each field is parsed as follows:

<Percentage for prime time testing>,<Percentage for non-prime time testing> Severity Code

F. Hourly Level Limits. These fields are used for testing the percent of hourly averages over the watchdog limits and for specifying an appropriate code for the severity of the situation. See "Sample Level Limits" above.

Item Description Section: This section defines the title to be used for identifying an item on a report, the short and longterm watchdog limits, and the comment to be printed when an item or subitem exceeds its limits.

G. Item Code. This code uniquely identifies the item measured by System AMAR. You should never change it. Refer to the Appendix called "System AMAR Item Definitions" for the meaning of each code.

H. Subitem Code. This code uniquely identifies the subitem measured by System AMAR. Only devices whose codes appear in your database should be listed here. Use the AMRGEN program to find out the valid subitem codes. A string of question marks ("??...") represents "all" devices or "all other" devices (if at least one device of the same type has been explicitly listed). Question marks, if used, should always preceed the explicit device names. The device names should be in numerical or alphabetical order.

I. Item/Subitem Title. This field contains a 20 character short title for the item or subitem. The title must have exactly 20 characters, including blanks. It is split into two groups of 10 characters each - preceeded, separated, and followed by an underscore ("_").

J. Picture Specification. This field contains the print format for the item/subitem values. "R" in this field denotes the position at which rounding will occur. The values of all items/subitems are stored in the database with 2 decimal positions regardless of the format shown here.

K. Data Type. This field contains a single character to be printed following the value of the item/subitem. It tells in what units the values have been expressed. Normally the only character used is "%" indicating percentage.

L. Short Term Threshold. This field defines the value for the watchdog limit when testing for the percent of samples (or whether or not the average is) over (">" greater than) or under ("<" less than) the watchdog limit. It is used for testing time periods of an hour or day. If no value is specified, testing will not occur.

M. Long Term Threshold. This field defines the value used for the watchdog limit when testing the percent of samples (or whether or not the average is) over (">" greater than) or under ("<" less than) the watchdog limit. It is used for testing periods of a week or month. If no value is specified, the short term threshold, if one exists, will be used.

N. Comment. This field defines a comment to be displayed on the Utilization Reports whenever the applicable watchdog limit has been exceeded. The comment may contain up to 35 characters. The comment usually describes the probable cause for the limit having been exceeded or a procedure to follow to investigate or correct a problem.

Report Description Section: This section defines which items get printed on the reports and whether or not they are always printed or printed only when watchdog limits are exceeded.

P. Subsection Title. Each System AMAR report contains one or more subsections where items are grouped and printed.

This field defines the title which will preceed the subsection. Examples of such default titles are "KEY UTILIZATION ITEMS", "OTHER UTILIZATION ITEMS", and "GENERAL SUMMARY". This field must be exactly 30 characters including blanks.

Q. Before Command. This field controls paging prior to printing the subsection defined here. If the field contains the words "PAGING BEFORE", the printer will eject to a new page before printing the subsection. If the field is blank, there will be no paging prior to printing the subsection.

R. After Command. This field controls paging after printing the subsection defined here. If it contains the words "PAGING AFTER", the printer will eject to a new page after printing the subsection. If the field is blank, there will be no paging after printing the subsection unless the next subsection description for the same report specifies "PAGING BEFORE".

S. Report Code. This field lists the 2 character report codes which define the reports for which the subsection should be printed. The field may contain up to 14 report codes separated by a blank. To receive the trend analysis sections (typically pages 3 - on) of the Weekly or Monthly Trend Analysis Reports, you must follow the applicable report code with the number 03 preceeded by a blank.

T. Item Code. This field contains the 4 character item code for each item to be printed in this subsection. Items may be listed in any order here and will be printed in that order. To delete an item from a report subsection, just delete the appropriate ".RI" line from that subsection. Conversly, to cause an item to be printed in a report subsection, add the appropriate ".RI" line with the item code to that subsection.

NOTE: Do not list an item more than once in any subsection. This will cause the report program to loop and exhaust your disk quota. Also, in order to get the disk report, at least one item must be specified and this item must have data in the System AMAR database.

U. Subitem Code. This field contains the subitem code (up to 6 characters) for each subitem to be printed in this subsection. Individual device names may be listed here even if they have not been explicitly defined in the "Item Description Section". Subitems, except for disk subitems, may be listed in any order here and will be printed in that order. Disk subitems will always be printed in alphabetical order regardless of their order in the RFD file. Question marks in this field indicate that "all" or "all other" devices should be printed. Question marks, if used, should follow the list of explicitly named devices. On disk reports, all subitems should be explicitly listed or only

question marks should be used. Ususally disk subitems will be explicitly listed only if you want to prevent some disks from printing.

NOTE: Do not list a subitem more than once per subsection. This will cause the report program to loop and exhaust your disk quota. Also do not mix item and subitem groups. For example, list all LUFS specifications together, then all LUIO specifications, etc. The order of the item groups controls the print order. In the example just given, LUFS would print before LUIO, etc.

Failure to observe the conventions in this RFD file may have unexpected results.

V. Treatment Code. This field determines whether or not values for the item or subitem will always be printed. "FORCED" means always print the values for the item or subitem. "TESTED" means print the values for the item or subitem only if the appropriate watchdog limit has been exceeded. Watchdog limits are considered to be exceeded if in the case of a high limit, the average value or at least 10% of the samples equal or exceed the limit, or, in the case of a low limit, the average value or at least 10% of the samples equal or fall below the limit.

NOTE: TESTED has meaning only when using the DU, WU, and MU report codes. Any item listed for other reports will always be FORCED even if TESTED is specified.

W. The last line of the xxxxDR.RFD file should always be:

.RD<tab>_END REPORTS

This tells AMREPT that no further report descriptions have been specified.

```
.C      ******** SYSTEM DESCRIPTION SECTION ********
.SD     TTSS    _TOPS20 SAMPLE SYSTEM_
.SS     *#@ 10,10_WARNING   25,50_SERIOUS   50,80_CRITICAL  10,10_WARNING   25,50_SERIOUS   50,80_CRITICAL
.C      ******** ITEM DESCRIPTION SECTION ********
.ID     AJBL            _ BAL SET  _ ABUS/SEC _ NNNNNN.R  >000001.0,          SCHEDULER THRASHING: FIND REASON
.ID     AMEM            _ PGS REAS _  MEMORY  _ NNNNNNR.  <0000768.,0000768.  MEMORY DOWN: MAY CAUSE SWAPPING
.ID     BGND            _  % BACK  _ GND TIME _ NNNNNNR.% >0000005.,          OVERHEAD TOO HIGH: INVESTIGATE
.ID     BSWT            _ # FORKS  _ BS WAIT  _ NNNNNNR.  >0000002.,          CPU BOTTLENECK OR SCHEDULER SLOW
.IDC    CPAO            _ %AMAR TI _ CPU O UP_ NNNNNNR.%
.IDC    CPAA            _ %AMAR TI _ CPU AVAL _ NNNNNNR.%
.IDC    CPAI            _ %AMAR TI _ CPU IDLE _ NNNNNNR.%
.IDC    CPAL            _ %AMAR TI _ CPU LOST _ NNNNNNR.%
.IDC    CPAO            _ %AMAR TI _ CPU OVHD _ NNNNNNR.%
.IDC    CPAU            _ %AMAR TI _ CPU UTIL _ NNNNNNR.%
.IDC    CPIO            _ CPU IDLE _   TIME   _ NNNNNNR.%                     CPU PRESSED: CHK WORKLD DATA FIRST
.IDC    CPOO            _ CPU OVHD _   TIME   _ NNNNNNR.%                     OVERHEAD TOO HIGH: INVESTIGATE
.ID     CTXS            _ CONTEXT  _ SWTS/SEC _ NNNNNNR.  >0000040.,          SCHEDULER TOO FAST: CHK BLOCKING
.ID     DKRD            _ DSK RDS  _ PGS/SEC  _ NNNNNNR.  >0000040.,          DISK READ RATE HIGH: CHK PACK I/O
.ID     DKWR            _ DSK WRS  _ PGS/SEC  _ NNNNNNR.  >0000040.,          DISK WRITE RATE HIGH: CHK PACK I/O
.ID     DMRD            _ SWAP RDS _ PGS/SEC  _ NNNNNNR.  >0000040.,          SWAPPING HIGH: CHK MEMORY AMOUNT
.ID     DMWR            _ SWAP WRS _ PGS/SEC  _ NNNNNNR.  >0000040.,          SWAPPING HIGH: CHK MEMORY AMOUNT
.ID     DSKR            _ % BS WQ  _ RD WAIT  _ NNNNNNR.%                     PGM I/O RATE HIGH: CHK WORKLD DATA
.ID     DSKW            _ % BS WQ  _ WR WAIT  _ NNNNNNR.%                     PGM I/O RATE HIGH: CHK WORKLD DATA
.ID     FILW            _  % IDLE  _ IO TIME  _ NNNNNNR.%                     JOB MIX I/O BOUND: CHK WORKLD DATA
.ID     FPGS            _ # FREE   _ MEM PGS  _ NNNNNNR.  <0000025.,          FREE CORE LOW: FIND CAUSE-UP AMOUNT
.ID     GCCW            _ GEN FREE _ PGS/SEC  _ NNNNNNR.  >0000020.,          CHK IF PAGE FAULT BOTTLENECK
.ID     IDLE            _  % IDLE  _   TIME   _ NNNNNNR.% <0000010.,          CPU PRESSED: CHK WORKLD DATA FIRST
.ID     KNOB            _   BIAS   _ CONTROL  _ NNNNNNR.                      BIAS CONTROL
.ID     LCCW            _ FRK FREE _ PGS/SEC  _ NNNNNN.R  >000005.0,          PGM THRASHING: CHK WORKLD DATA
.ID     LKPG            _ # LOCKED _  PAGES   _ NNNNNNR.  >0000500.,          TOO MUCH CORE PREEMPTED: FIND CAUSE
.ID     LUFS ???????    _ %??????? _ FREE SPC _ NNNNNNR.% <0000010.,<0000010. DELETE UNNECESSARY FILES
.ID     LURD ???????    _ ??????? _ RDS/SEC  _ NNNNNNR.  >0000040.,>0000040. I/O RATE HIGH: CHK FOR CONTENTION
.ID     LUSK ???????    _ ??????? _ SEKS/SEC  _ NNNNNNR.  >0000030.,          CHK FILE PLACEMENT/FRAGMENTATION
.ID     LUWQ ???????    _ ??????? _  WAIT Q   _ NNNNNN.R  >000001.0,>000000.8 FILE/PACK CONTENTION: INVESTIGATE
.ID     LUWR ???????    _ ??????? _ WRS/SEC  _ NNNNNNR.  >0000040.,>0000040. I/O RATE HIGH: CHK FOR CONTENTION
.ID     MTAU            _ # MTAS   _ ASSIGNED _ NNNNNNR.                      MANY TAPES ASSIGNED: CHK ACTUAL USE
.ID     MTIO ?????      _ MT ????? _  PGS/SEC  _ NNNNNNR.  >0000035.,>0000035. TAPE I/O RATE HIGH: INVESTIGATE USE
.ID     NBAL            _ # BALNCE _ SET FRKS _ NNNNNNR.  >0000020.,          MANY FORKS: CHK WORKLD DATA
.ID     NCOR            _ MEM MGMT _ CYCS/SEC _ NNNNNN.R  >000010.0,          MAPPED PAGE REUSE LOW: INVESTIGATE
.ID     NLOD            _ WORK SET _  LDS/SEC _ NNNNNN.R  >000001.0,          CHK JOB MIX/SWAPPING I/O RATE
.ID     NREM            _ BAL SET  _ SWPS/SEC _ NNNNNN.R  >000001.0,          CORE BOTTLENECK: RESTRICT FORKS
.ID     NRUN            _ # ACTIVE _  FORKS   _ NNNNNNR.  >0000030.,          MORE FORKS THAN PLANNED
.ID     NTRP            _ PG FAULT _ TRPS/SEC _ NNNNNNR.  >0000050.,          PAGING TOO OFTEN: CHK WORKLD DATA
.ID     NWSM            _ # MEMORY _ WRK SETS _ NNNNNNR.                      ?
.ID     PTYU            _  # PTYS  _  IN USE  _ NNNNNNR.  >0000040.,          PTY LINE USE HIGHER THAN PLANNED
.ID     PUFS ????       _ %UN ???? _ FREE SPC _ NNNNNNR.% <0000010.,<0000010. DELETE UNNECESSARY FILES
.ID     PURD ????       _ UN ????  _ RDS/SEC  _ NNNNNNR.  >0000040.,>0000040. I/O RATE HIGH: CHK FOR CONTENTION
.ID     PUSK ????       _ UN ????  _ SEKS/SEC _ NNNNNNR.  >0000030.,          CHK FILE PLACEMENT/FRAGMENTATION
.ID     PUWQ ????       _ UN ????  _  WAIT Q  _ NNNNNN.R  >000001.0,>000000.8 FILE/PACK CONTENTION: INVESTIGATE
.ID     PUWR ????       _ UN ????  _ WRS/SEC  _ NNNNNNR.  >0000040.,>0000040. I/O RATE HIGH: CHK FOR CONTENTION
.ID     RPQS            _ RQ SAVES _ PGS/SEC  _ NNNNNN.R                      ?
.ID     SHPG            _ # SHARED _  PAGES   _ NNNNNNR.                      ?
.ID     SKED            _ % SCHED  _   TIME   _ NNNNNNR.% >0000010.,          SCHEDULER PRESSED: FIND CAUSE
.ID     SWFS            _  % SWAP  _ SPC FREE _ NNNNNNR.% <0000010.,          SWAPPING SPACE LOW: FIND CAUSE
.ID     SWPR            _ % BS WQ  _ SWP WAIT _ NNNNNNR.%                     ?
.ID     SWPW            _  % IDLE  _ SWP TIME _ NNNNNNR.%                     SWAPPING SLOW: CHK WORKLD DATA
.ID     TCOR            _  % MGMT  _ MEM TIME _ NNNNNN.R% >000001.0,          MAPPED PAGE REUSE LOW: INVESTIGATE
.ID     [redacted]      _ % PAGING _   TIME   _ NNNNNNR.% >[redacted]010.,    PAGING TOO OFTEN: CHK WORKLD DATA
```

Figure 1-19

```
.ID     TTCC
.ID     TTIN            _ TTY INTR  _ CHRS/SEC  _ NNNNNNR.      >0000010..
.ID     TTOU            _  TTY IN   _ CHRS/SEC  _ NNNNNNR.      >0000100..
.ID     TTYU            _ TTY OUT   _ CHRS/SEC  _ NNNNNNR.      >0000500..
.ID     TWQC    ?       _ # LINES   _  IN USE   _ NNNNNNR.      >0000060..
.ID     UMEM            _  CHAN ?   _ WAIT Q    _ NNNNNN.R      >000001.0,>000000.8
.ID     UPGS            _ PGS USER  _ MEMORY    _ NNNNNNR.      <0000768..<0000768.
.ID     USED            _  # WORK   _ SET PGS   _ NNNNNNR.      >0001500..
.ID     WAKE            _  % USED   _  TIME     _ NNNNNNR.%     >0000090..
.ID     XAMT            _  # FORK   _ WAKS/SEC  _ NNNNNNR.      >0000020..
.ID     XRLD            _ % AMAR    _ CLK TIME  _ NNNNNN.R%
.ID     XUPT            _ # SYSTEM  _ RELOADS   _ NNNNNNR.
.ID     _CPU            _ % SYSTEM  _ UPTIME    _ NNNNNN.R%
                        _  %  CPU   _  UTIL     _ NNNNNNR.%     >0000095..>0000070.

.C      ******** REPORT DESCRIPTION SECTION ********

.RD     _       KEY UTILIZATION ITEMS   _       PAGING BEFORE   PAGING AFTER    DU WQ MC
.RI     NRUN            FORCED
.RI     NWSM            FORCED
.RI     BSWT            FORCED
.RI     AJBL            FORCED
.RI     TTYU            FORCED
.RI     WAKE            FORCED
.RI     IDLE            FORCED
.RI     USED            FORCED
.RI     BGND            FORCED
.RI     SKED            FORCED
.RI     TRAP            FORCED
.RI     NTRP            FORCED
.RI     CTXS            FORCED
.RI     DMRD            FORCED
.RI     DMWR            FORCED
.RI     DKRD            FORCED
.RI     DKWR            FORCED
.RI     MTAU            FORCED
.RI     FPGS            FORCED
.RI     UMEM            FORCED

.RD     _KEY UTILIZATION ITEMS          _       PAGING BEFORE   PAGING AFTER    WU MU
.RI     _CPU            FORCED
.RI     NRUN            FORCED
.RI     NWSM            FORCED
.RI     BSWT            FORCED
.RI     AJBL            FORCED
.RI     TTYU            FORCED
.RI     WAKE            FORCED
.RI     IDLE            FORCED
.RI     USED            FORCED
.RI     BGND            FORCED
.RI     SKED            FORCED
.RI     TRAP            FORCED
.RI     NTRP            FORCED
.RI     CTXS            FORCED
.RI     DMRD            FORCED
.RI     DMWR            FORCED
.RI     DKRD            FORCED
.RI     DKWR            FORCED
.RI     MTAU            FORCED
.RI     FPGS            FORCED
```

```
TOO MANY TTY INTERRUPTS: FIND CAUSE
TTY CHAR RATE TOO HIGH: FIND CAUSE
TTY CHAR RATE TOO HIGH: FIND CAUSE
TTY LINE USE HIGHER THAN PLANNED
CHANNEL CONTENTION: CHK I/O BALANCE
USER MEMORY LOW: FIND CAUSE
PGM SIZES LARGER THAN PLANNED
CPU PRESSED: CHK JOB MIX
TOO MANY WAKEUPS: FIND CAUSE
AMAR NOT RUNNING ENOUGH: CHK REASON
TOO MANY RELOADS: CHK REASONS
DOWNTIME HIGH: CHK RELOAD REASONS
CPU PRESSED: CHK WORKLD DATA FIRST
```

Figure 1-19 (continued)

```
.RI     UMEM            FORCED
.RI     XUPT            FORCED
.RI     XAMT            FORCED
.RI     XRLD            FORCED

.RD     _OTHER UTILIZATION ITEMS        _                   PAGING AFTER    DU WU MU WC MC

.RI     FILW            TESTED
.RI     SWPW            TESTED
.RI     SWFS            TESTED
.RI     NBAL            TESTED
.RI     NLOD            TESTED
.RI     NREM            TESTED
.RI     NCOR            TESTED
.RI     TCOR            TESTED
.RI     RPQS            TESTED
.RI     GCCW            TESTED
.RI     LCCW            TESTED
.RI     DSKR            TESTED
.RI     DSKW            TESTED
.RI     SWPR            TESTED
.RI     UPGS            TESTED
.RI     AMEM            TESTED
.RI     LKPG            TESTED
.RI     SHPG            TESTED
.RI     TTCC            TESTED
.RI     TTIN            TESTED
.RI     TTOU            TESTED
.RI     PTYU            TESTED
.RI     KNOB            TESTED
.RI     TWQC    ?       TESTED
.RI     LUFS    ???????? TESTED
.RI     LURD    ???????? TESTED
.RI     LUSK    ???????? TESTED
.RI     LUWQ    ???????? TESTED
.RI     LUWR    ???????? TESTED
.RI     MTIO    ?????   TESTED
.RI     PUFS    ????    TESTED
.RI     PURD    ????    TESTED
.RI     PUSK    ????    TESTED
.RI     PUWQ    ????    TESTED
.RI     PUWR    ????    TESTED

.RD     _KEY UTILIZATION ITEMS          _           PAGING BEFORE   PAGING AFTER    WA 03 MA 03

.RI     _CPU            FORCED
.RI     AJBL            FORCED
.RI     AMEM            FORCED
.RI     BGND            FORCED
.RI     BSWT            FORCED
.RI     CTXS            FORCED
.RI     DKRD            FORCED
.RI     DKWR            FORCED
.RI     DMRD            FORCED
.RI     DMWR            FORCED
.RI     DSKR            FORCED
.RI     DSKW            FORCED
.RI     FILW            FORCED
.RI     FPGS            FORCED
.RI     GCCW            FORCED
.RI     I               FORCED
```

Figure 1-19 (continued)

```
.RI     KNOB            FORCED
.RI     LCCW            FORCED
.RI     LKPG            FORCED
.RI     MTAU            FORCED
.RI     NBAL            FORCED
.RI     NCOR            FORCED
.RI     NLOD            FORCED
.RI     NREM            FORCED
.RI     NRUN            FORCED
.RI     NTRP            FORCED
.RI     NWSM            FORCED
.RI     PTYU            FORCED
.RI     RPOS            FORCED
.RI     SHPG            FORCED
.RI     SKED            FORCED
.RI     SWFS            FORCED
.RI     SWPR            FORCED
.RI     SWPW            FORCED
.RI     TCOR            FORCED
.RI     TRAP            FORCED
.RI     TTCC            FORCED
.RI     TTIN            FORCED
.RI     TTOU            FORCED
.RI     TTYU            FORCED
.RI     UMEM            FORCED
.RI     UPGS            FORCED
.RI     USED            TESTED
.RI     WAKE            TESTED
.RI     XUPT            FORCED
.RI     XAMT            FORCED
.RI     XRLD            FORCED

.RD     _GENERAL SUMMARY

                        -       PAGING BEFORE           DD WD MD
.RI     DKRD            FORCED
.RI     DKWR            FORCED
.RI     DMRD            FORCED
.RI     DMWR            FORCED
.RI     SWFS            FORCED
.RI     TWQC    ?       FORCED

.RD     _PACK NAME

                        -       PAGING AFTER    DD WD MD
.RI     LUFS    ???????? FORCED
.RI     LURD    ???????? FORCED
.RI     LUSK    ???????? FORCED
.RI     LUWQ    ???????? FORCED
.RI     LUWR    ???????? FORCED

.RD     _UNIT NAME

                        -                       DD WD MD
.RI     PUFS    ????    FORCED
.RI     PURD    ????    FORCED
.RI     PUSK    ????    FORCED
.RI     PUWQ    ????    FORCED
.RI     PUWR    ????    FORCED

.RD     _GENERAL SUMMARY

                        -       PAGING BEFORE           DT WT MT
```

Figure 1-19 (continued)

```
.RI     MTAU            FORCED
.RD     _TAPE DRIVE                      _                       DT WT MT
.RI     MTIO     ?????  FORCED
.RD     _END REPORTS                     _
```

Figure 1-19 (continued)

1.5.2 Examples Of Some Common Changes To The RFD File

Changes which are commonly made to the RFD file include adjusting threshold limits (especially for testing for disk free space), revising the comments that get printed when thresholds get exceeded, and forcing certain items or subitems to be always printed.

Example 1 - Changing the Free Space Watchdog Limit:

By default, any pack that has less than ten percent free space will appear on the Utilization Reports as being under the acceptable watchdog limit. Typically, page two of these reports will contain the pack name, the number of hours when the free space was less than 10% (using P's and N's) and the message "DELETE UNNECESSARY FILES". For most packs this 10% free space limit is acceptable. However, if a particular pack, perhaps DSKX, consistently has 5% free space, it is not necessary or useful to see DSKX show up every day with every hour flagged with asterisks. Asterisks generally mean that this is something important to look at or a potential problem to solve. You can make a couple of quick edits to the RFD file to change the free space threshold to perhaps 3%. To do this the "ITEM DESCRIPTION SECTION" is changed to specifically add DSKX 0 with the new limits. The pack name must be EXACTLY 7 characters (DSKX 0 here). Include blanks if necessary between the pack name and its sequence number in the logical structure. The sequence number appears always as the 7th character. The item code for logical unit free space is LUFS. In the RFD file, insert another line immediately after the .ID LUFS line. This new line should have the same format as the LUFS line with the "wild card" question marks in the subitem code field changed to DSKX 0. The easiest way to do this is to copy the LUFS line under itself. You will end up with two LUFS lines. Then change the question marks on the second line to the pack name. Next, change the limits. The short term/long term limit fields respectively are "<0000010.,<0000010." on the LUFS line. On the new LUFS DSKX 0 line change the fields to "<0000003.,<0000003." This will make the short term/long term limits both 3%. Values for this pack will be flagged now only if it has less than 3% free space. Any number of packs can be added to the RFD in this way. There are only two cautions: if a list of packs is added, they must be in alphabetical order; the question marks on the LUFS line serve as a "wild card" that will allow the line to apply to all packs not specified by name. Do not accidentally delete the "wild carded" line while editing.

Example 2 - Changing the Comment Field:

Another change that can be made is to revise the comment line that appears when an item exceeds its limits. This can be useful, for example, when DSKX is owned by a particular user group that wants to know when their pack has less than

3% free space. After DSKX 0 has been specified in the LUFS list, the "DELETE UNNECESSARY FILES" text on the LUFS DSKX 0 line can be change to "NOTIFY USER GROUP". With that edit, whenever the hourly average for DSKX is 3% or less, the notify message will appear. For the other packs, the old delete message will still be printed.

Example 3 - Forcing Specific Items/Subitems to Print:

The REPORT DESCRIPTION SECTION of the RFD controls which items are designated as "KEY" in the reports (i.e., always appear) and which are designated as "OTHER" (i.e., appear only if they are flagged as over or under their limits). A simple change, as an example, would be to remove MTAU (number of MTA's assigned) from the key item list of the Daily System Utilization Report and replace it with something more interesting such as SWPW, % IDLE SWP TIME. In Figure 1-19, the second section for key utilization items immediately under the ".C ********REPORT DESCRIPTION SECTION" controls the key items for the Weekly/Monthly Utilization Reports. Simply change the code MTAU to SWPW. % IDLE SWP TIME will always appear now as a key item.

1.6 PROCEDURE FOR RUNNING AMAR.CTL

1.6.1 Overview Of AMAR.CTL

This stream runs daily. It takes the data which has been collected by xxxxDC, inputs it into the system AMAR database, creates summary records, deletes expired records from the database, performs housekeeping on the AMAR area, and creates the automatic reports. Reference the Appendix called "System AMAR Batch Stream - AMAR.CTL" for a sample of the stream and step descriptions.

1.6.2 Resource Requirements

To run System AMAR on a continuous basis, you will need a SYSJOB subjob and a permanently mounted disk area with approximately 950 pages for program and raw file storage. The size of your system AMAR database will probably vary from 1500-2500 pages depending on the type and amount of data retained. You will need to reserve space for the original database plus a backup copy, preferably on permanent storage. See also the Appendix called "Installation and Resource Requirements".

1.6.3 Submission

The stream normally resubmits itself after running each night. If both the System AMAR database and its backup are corrupt (parity errors, etc.), the stream stops and the operator must restart it after restoring the database from a good copy. The stream should always be restarted from the beginning. There should always be an AMAR.CTL in the batch queue, set to run /AFTER:TODAY+1:00.

1.6.4 Restart Procedure

If a system crash occurs while the stream is running, the stream should automatically restart at the proper checkpoint. If the crash occurs while AMARIP or AMARUP is running, the database will become corrupted. The stream will test for a corrupted database and automatically restore from the disk backup, if necessary. The only time the stream should need manual restarting is if the batch queue were destroyed or if both the primary and backup System AMAR databases are corrupted.

Blank Page

CHAPTER 2

WORKLOAD AMAR

2.1 MAJOR FEATURES

Features of Workload AMAR include:

1. Low overhead continuous data collection.

2. An historical workload database featuring:

   a. Separate files for different time periods to minimize I/O.

   b. Detail data retained in compressed form.

   c. Automatic deletion of old data.

3. Flexible reporting programs featuring:

   a. Grouping of resource usage by one or more items such as user name, account name, program name, batch vs. timesharing category, and individual session.

   b. Sorting of detail lines by above items or more likely by resource usage to highlight heavy users.

   c. Optional suppression of insignificant detail lines.

   d. Selection of time period to be reported and whether to describe it in a single report or a series of interval reports.

4. Single daily batch stream which will:

   a. Provide useful reports automatically (daily, weekly, monthly).

   b. Prevent buildup of data files on disk.

2.2 OVERVIEW OF WORKLOAD AMAR

The three functions of Workload AMAR (also referred to as the workload system) are data collection, database management and reporting. The three functions are performed by four separate programs which are described briefly below. Refer to Figure 2-1 for an overview of program and data flow.

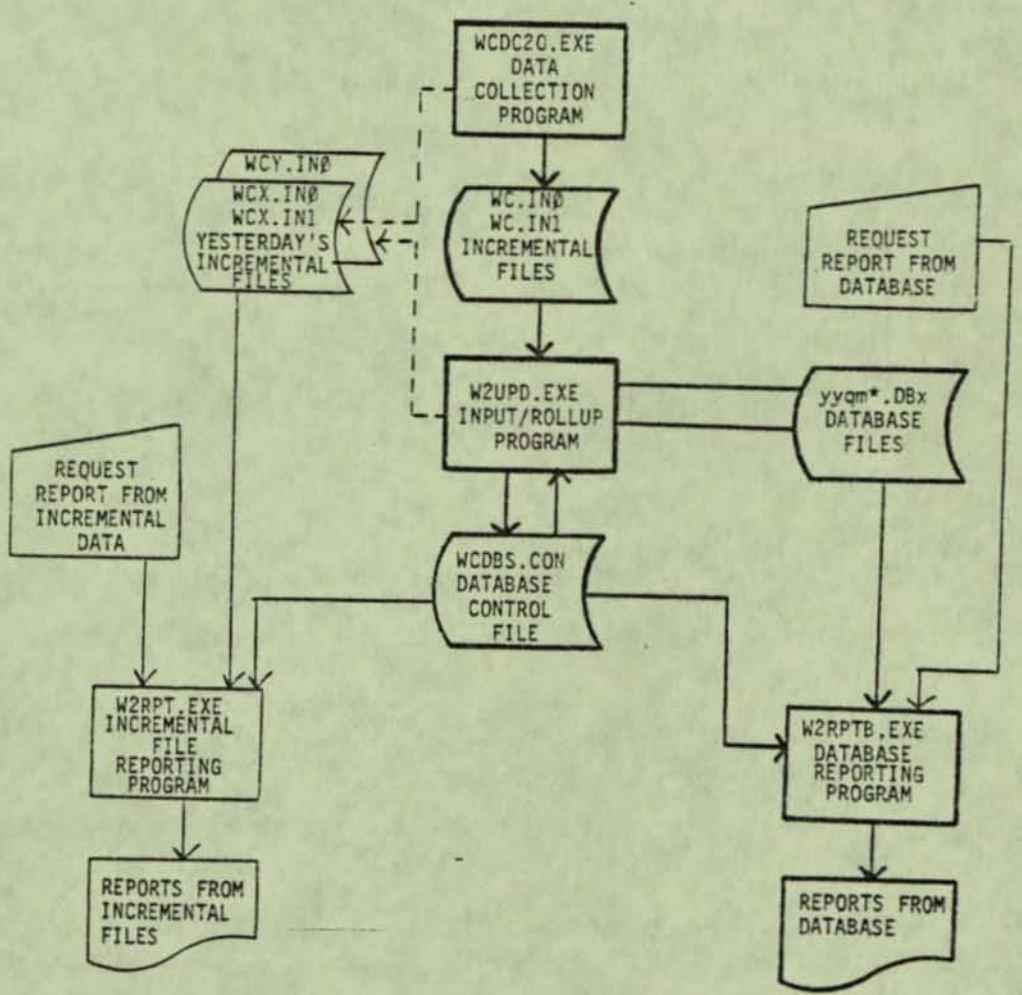
2.2.1 Data Collection

WCDC20 collects resource utilization, identification, and response time data about each job on the system at "checkpoint intervals" which occur typically every 5 minutes. It creates two output files: one which contains incremental usage by job and one which summarizes incremental usage by checkpoint interval. WCDC20 collects data about individual jobs and forks not normally collected by the operating system. To accomplish this, WCDC20 dynamically patches into the operating system a small amount of executable code and a moderate amount of data tables. Currently this code and data must fit in the SNOOP pages. Systems which support large numbers of jobs and forks ususally do not have enough default SNOOP pages to accomodate the workload data collection program. Hence it is usually necessary to rebuild the operating system with more SNOOP pages. It is also highly recommended that if the job program name table (JOBPNM) is non-resident, that the operating system be rebuilt to make it resident. This is necessary to allow WCDC20 to collect the job program name which is more useful than the subsystem name. It will allow more meaningful reports to be produced. Instructions for rebuilding the monitor are included in the AMAR-20 Installation Guide. WCDC20 runs 24 hours a day, preferably as a SYSJOB subjob.

2.2.2 Database Management

W2UPD performs database management including input, rollup, and deletion. Data for each day is included in the database as a separate file with a section for each hour of the day. Daily data for each weekday is rolled up into a weekly weekday file which has a section for each "typical hour". Daily data for each Saturday, Sunday or holiday in a week is rolled into a similar weekly weekend file. Weekly files are rolled into monthly files. When there are more than the desired number of daily, weekly, or monthly files, the oldest files are automatically deleted.

Key: yy = Fiscal Year
q = Quarter of Fiscal Year
m = Month of Fiscal Year
w = Week of Fiscal Month
d = Day of Fiscal Week
* = Blank, w, or Wd
x = 0 if Weekday, 1 if Weekend



WORKLOAD
AMAR
OVERVIEW

Figure 2-1

2.2.3 Reporting

The main reporting program, W2RPTB, uses the workload database to report on resource utilization during user-specified reporting intervals, which are normally an integral number of hours or "typical hours". Resource usage may be summarized by user-specified identification data. It is also sorted by those keys and/or by amount of resources. Weekly and monthly reports are generated automatically by the single daily batch stream. W2RPT is a variation of W2RPTB which is used to report from the incremental files rather than the database.

2.3 ANNOTATED SAMPLE REPORTS

This section contains four samples of the workload reports produced by the standard daily stream W2RPTB.CTL. The first report is heavily annotated to help you understand the format which is common to all daily workload reports. The minor difference in format for other workload reports (weekly and monthly) is in the first line of the subreport header, which is annotated in the second sample report. Workload AMAR reports are often used in conjunction with the System AMAR Utilization and Trend Analysis Reports. These latter reports show activity on the system as a whole.

2.3.1 Hourly Report By Program And User

Figure 2-2 shows the beginning of WCDY0.RP1, the standard daily report with hourly subreports.

This report is the first place to look for workload reasons for problems reported in the System AMAR Daily Utilization Report.

Each detail line shows resource usage by a particular job running a particular program. Major CPU users are at the top of the list.

The first page in Figure 2-2 shows the report header box and the first subreport. The subreport starts at approximately midnight and runs to 1:00 AM (00:04:18 to 01:04:21).

The second page of Figure 2-2 shows portions of two subreports from a busy period. The subreports start at approximately 16:00 and run to 18:00. Intervening and trailing subreports have been omitted from this example.

The following notes refer to the circled numbers on the sample report:

REPORT HEADER BOX:

1. This box surrounded by asterisks appears at the top of the first page of each workload report. Look for this box when searching for a particular report in a series of workload reports printed without separator pages.

2. "AMAR WORKLOAD REPORT" always appears in the report header box.

3. Site description (up to 90 characters) comes from the file WCDBS.CON. This description is set up at installation time.

4. System code (4 characters) is used to identify workload data as belonging to a particular system. It comes from WCDBS.CON and is included in all workload database files. The system code should be the same as the code used in the System AMAR database.

5. Report description (up to 90 characters) is entered during W2RPTB dialogue (in the batch stream or on-line).

6. Input filename indicates the fiscal period covered.

7. Parentheses enclose the explanation of the input filename.

SUBREPORT HEADINGS:

8. The first line of this subreport heading is typical for daily reports. The first line of a weekly or monthly subreport heading is different. (See the next sample report for an example.)

9. Start of report interval (time, day of the week, date).

10. End of report interval.

11. Length of report interval (HH:MM:SS).

12. Percent of interval measured for workload characterization. Only measured time is used to compute resource usage rates.

13. List of items whose values are held constant to determine what goes into each line of detail data. In this case, USR1, USR2, and USR3 (User Name) identify a user and PNAM (Job Program Name) identifies a particular

program that the user was running. Refer to the Appendix called "Valid Grouping and/or Sort Items" to get a list of the item names which can be used for grouping.

14. Sort key. In this case, the detail lines have been sorted in descending order by percentage of processor use (CPU%), to draw attention to major CPU users at the top of the list. Refer to the Appendix called "Valid Grouping and/or Sort Items" to get a list of the item names which can be used for sorting.

15. Cutoff criteria. In this case, each detail line representing less than 1.0% of the CPU was suppressed.

INTERVAL TOTALS LINE (See page 2 of the example.):

16. The INTERVAL TOTALS line is the summary of resource usage during the reporting interval.

17. The total at the top of this column indicates average number of jobs in use.

18. The average number of forks in use.

19. The average number of forks in memory.

20. The average number of forks actively competing for the CPU or disk.

21. The average working set size in pages in the INTERVAL TOTALS line is a weighted average of working set sizes during the period.

22. Except for seconds per response (SEC/RSP) and stretch ratio (SR), other numbers in the INTERVAL TOTALS line are ordinary totals. SEC/RSP and SR are averages for the reporting interval.

DETAIL LINES:

Refer to the Appendix called "Workload AMAR Item Definitions" for a description of all items which can appear in the detail lines of the workload reports. In addition to the items shown in the detail lines of Figure 2-2, there are addition items which can be obtained from the W2RPTB program. These extra items would appear on an extra line of detail. To get the additional items, respond "Y" to the EXTRA DETAILS? (Y OR N): query when running W2RPTB.

23. Job number (as would be reported by SYSTAT, SYSDPY, WATCH, etc.).

24. Fork number. This number is not usually reported since one fork at a time is rarely looked at.

25. Fraction of the reporting interval that the specified program appeared to be in use. Since this example is broken out by program within session, this value will be 1.0 if the session spanned the entire interval and the same program ran throughout the interval. A value of 0.4 would indicate that the program was running during 40% of the interval.

26. The average number of forks running under the specific job program name (not fork program name) during the interval. If AVG JOBS is .4 and there were 2 forks always running, AVG FRKS would equal .8. 27. Day on which the job logged in, relative to the end of the report interval. For example, 0 means the job logged in during the day in which the interval ended, -1 means the previous day.

28. Time of day when the job logged in.

SUBTOTAL LINES:

29. When some but not all of the detail lines have been suppressed because of cutoff criteria, this line is included to summarize the detail lines printed.

30. This line is included to summarize any suppressed detail lines.

AMAR WORKLOAD REPORT (2)

SITE: TOPS-20 SAMPLE SYSTEM (3)

SYSTEM: TTSS (4)

REPORT DESCRIPTION: HOURLY REPORT BY PROGRAM AND USER (5)

INPUT FILE: 832143.DBO (6) (FISCAL YEAR: 83 QUARTER: 2 MONTH: 1 WEEK: 4 DAY: 3 TUESDAY) (7)

(8) FROM: 0: 4:18 ON (9) TUESDAY 26-OCT-82 TO: 1: 4:21 ON (10) TUESDAY 26-OCT-82 INTERVAL: 1: 0: 3 (11) MEASURED: 100% (12)

GROUPED BY: USR1 USR2 USR3 PNAM (13)

SORTED BY: (14) CPU% CUTOFF: 1.00% OF CPU (15)

JOB #	FRK #	AVG JOBS	AVG FRKS	IN MEM	DEMD	USER NAME	ACCOUNT NAME	PRGRM NAME	PAGES (WS)	CPU%	SWAP PF/S	FILE PF/S	IFA	RSP /MIN	SEC /RSP	SR	B	TTY	LOGIN DAY	AT TIME
		14.3	70.5	33.5	1.20	**********INTERVAL TOTALS**********			55.0	80.85	0.2	7.2	119	84.6	0.0	1				
46		0.4	0.8	0.8	0.34	AT.PRODUCTION	670.A0000002	1274BS	86.8	25.71	0.0	2.1	133	0.6	0.2	1	B	220	-1	21:15
46		0.2	0.5	0.5	0.21	AT.PRODUCTION	670.A0000002	ISAM	32.0	17.84	0.0	1.3	148	1.4	0.1	1	B	220	-1	21:15
46		0.0	0.2	0.2	0.14	AT.PRODUCTION	670.A0000002	RUN	42.4	8.07	0.0	0.3	256	0.5	0.1	2	B	220	-1	21:15
46		0.0	0.1	0.1	0.11	AT.PRODUCTION	670.A0000002	EXEC	44.9	7.57	0.0	1.4	55	0.3	0.0	1	B	220	-1	21:15
23		0.1	0.2	0.2	0.08	ETAMAR.DBS	669.A0000000	W2UPD	54.8	6.86	0.0	0.4	199	0.0	0.0	0	B	221	0	0:00
46		0.1	0.2	0.2	0.07	AT.PRODUCTION	670.A0000002	EXPUNG	47.4	6.18	0.0	0.2	239	0.9	0.3	1	B	220	-1	21:15
23		0.0	0.1	0.1	0.09	ETAMAR.DBS	669.A0000000	RUN	37.8	3.44	0.0	0.6	66	1.7	0.1	2	B		0	
23		0.1	0.1	0.1	0.07	ETAMAR.DBS	669.A0000000	AMARUP	41.6	1.54	0.0	0.4	43	1.1	0.2	2	B	217	0	1:00
46		0.1	0.1	0.1	0.02	AT.PRODUCTION	670.A0000002	1039BS	76.0	1.22	0.0	0.1	80	1.1	0.3	1	B	220	-1	21:15
		1.0	2.2	2.2	1.14	********SUBTOTALS THRU CUTOFF********			55.5	78.43	0.2	6.9	121	7.6	0.2	1				
		13.3	68.3	31.3	0.06	********SUBTOTALS AFTER CUTOFF********			46.3	2.42	0.0	0.3	79	76.9	0.0	1				

FROM: 1: 4:21 ON TUESDAY 26-OCT-82 TO: 2: 4:23 ON TUESDAY 26-OCT-82 INTERVAL: 1: 0: 1 MEASURED: 100%

GROUPED BY: USR1 USR2 USR3 PNAM

SORTED BY: CPU% CUTOFF: 1.00% OF CPU

JOB #	FRK #	AVG JOBS	AVG FRKS	IN MEM	DEMD *	USER NAME	ACCOUNT NAME	PRGRM NAME	PAGES (WS)	CPU%	SWAP PF/S	FILE PF/S	IFA *	RSP /MIN	SEC /RSP	SR	B	TTY	LOGIN DAY	AT TIME

Figure 2-2

	46.0	169.0	125.1	2.04	**********INTERVAL TOTALS**********			39.5	76.71	0.6	5.4	143	534.2	0.1	2					
50	0.7	1.3	1.3	0.66	BARBARA															
	1.6	5.4	5.4	0.32	AT.INQUIRY	670.A0000001	O768BS	39.1	27.68	0.0	0.5	549	0.0	0.0	0	B	217	0	14:31	
50	0.1	0.2	0.2	0.08	BARBARA		REPORT	29.0	11.46	0.0	1.2	96	22.4	0.1	2	T		0		
50	0.1	0.2	0.2	0.09	BARBARA		EXPUNG	36.4	4.92	0.0	0.1	463	0.3	0.1	1	B	217	0	14:31	
35	0.5	1.0	1.0	0.11	IRWIN		EXEC	41.7	4.49	0.0	0.4	126	1.6	0.3	2	B	217	0	14:31	
	6.2	23.9	23.8	0.06	AT.INQUIRY	639.830809.2000	IRSML1	22.1	3.86	0.0	0.0	4224	152.6	0.0	1	T	47	0	10:42	
	1.3	2.5	2.5	0.06	FS.USER	670.A0000001	AT	33.4	2.41	0.0	0.2	113	27.4	0.1	2	T		0		
	4.0	10.1	9.9	0.04	AT.INQUIRY	672.A0000000	COMP3	71.1	2.27	0.0	0.0	2183	2.0	0.5	1	T		0		
	2.0	8.8	8.5	0.03	AT.INQUIRY	670.A0000001	ATINQ	34.9	1.45	0.1	0.1	96	22.4	0.1	2	T		0		
52	0.1	0.1	0.1	0.05	AT.USER	670.A0000001	ADJUST	35.0	1.39	0.0	0.2	96	25.6	0.1	2	T		0		
0	1.0	20.0	12.0	0.05	OPERATOR	670.A0000005	CSHUPD	59.9	1.33	0.0	0.1	134	0.5	0.2	2	T	4	0	16:31	
32	0.3	0.8	0.8	0.06	IP.BROOKS	OPERATOR	SYSJOB	44.3	1.29	0.0	0.1	150	64.5	0.0	2	T	DET	-4	5:45	
(29)	17.8	74.3	65.6	1.61	********SUBTOTALS THRU CUTOFF*********	647.820343	EXEC	84.0	1.02	0.0	0.3	36	2.5	0.2	3	T	165	0	15:22	
(30)	28.2	94.7	59.5	0.43	********SUBTOTALS AFTER CUTOFF*********			39.0	63.58	0.3	3.2	197	321.9	0.1	2					
								41.1	13.12	0.3	2.1	65	212.3	0.1	2					

FROM: 17: 0:49 ON TUESDAY 26-OCT-82 TO: 18: 0:51 ON TUESDAY 26-OCT-82 INTERVAL: 1: 0: 1 MEASURED: 100%

GROUPED BY: USR1 USR2 USR3 PNAM

SORTED BY: CPU% CUTOFF: 1.00% OF CPU

JOB #	FRK #	AVG JOBS (17)	AVG FRKS (18)	IN MEM (19)	DEMD (20)	USER NAME	ACCOUNT NAME	PRGRM NAME	PAGES (WS) (21)	CPU%	SWAP PF/S	FILE PF/S	IFA	RSP /MIN	SEC /RSP (22)	SR	B	TTY	LOGIN AT DAY	TIME
(16)		26.1	102.6	52.5	1.51	**********INTERVAL TOTALS**********			31.7	62.35	0.7	10.5	66	1536.6	0.0	1				
	(25) 1.5	2.8	2.8	0.78	OPERATOR															
	1.9	6.8	6.8	0.21	AT.INQUIRY	670.A0000001	DUMPER	22.1	37.47	0.1	5.9	74	1167.2	0.0	1	B		0		
	0.2	0.4	0.4	0.16	OPERATOR		REPORT	24.8	8.15	0.0	1.8	50	67.0	0.1	1	T		0		
	1.7	6.5	6.5	0.07	AT.INQUIRY	670.A0000001	LOGOUT	70.3	4.59	0.1	1.3	40	50.2	0.1	2	B		0		
(23) 12	(24) 0.1	0.2	0.2	0.08	OPERATOR		AT	27.8	3.04	0.0	0.2	158	22.2	0.2	2	T		0		
0	1.0	20.0	12.0	0.04	OPERATOR	OPERATOR	EXPUNG	53.3	2.20	0.0	0.5	51	4.0	0.1	3	B	217	0	17:04	
	6.3	36.6	28.6	1.35	********SUBTOTALS THRU CUTOFF*********		SYSJOB	42.7	1.09	0.1	0.1	86	64.5	0.0	2	T	DET	-4	5:45	
								31.2	56.53	0.4	9.7	66	1375.1	0.0	1					

(27) (28)

Figure 2-2 (continued)

2.3.2 Program Name Report

Figure 2-3 is a sample of the default report WCWK0.RP2 and shows weekly resource usage by job program name (PNAM) on the basis of typical 8-hour shifts. This report highlights programs which may be candidates for optimization or rescheduling. The detail lines are sorted by percent of CPU used with the heaviest consumers at the top of the list. For example, during prime time, operator jobs running under PTYCON (J) used 17.64% of the CPU. The second heaviest user of the CPU was 1038BS. It used 5.32% of the processor over the five day period.

A CPU cutoff (K) of 1.5% is used to suppress printing of any lines containing programs which used less than 1.5% of the CPU.

Note the first line of the subreport header (A). It is different from the corresponding line on a daily report. It first tells the start time (B) and end time (C) of the "typical period" described. Then it tells what type of day (D) is included. This should be read "WEEKDAYS MINUS HOLIDAYS". The other possibility is "WEEKENDS (+ HOLIDAYS)". Next it tells the first day (E), last day (F), and number of days (G) included. A glance at a fiscal calendar will tell you if all the days have been included. Finally, it tells how much of the time was measured: first as a percentage (H) of the typical period, then as a total number of hours (I). This number of hours can be used to compute resource totals from the reported rates.

Note: Question marks in the USER NAME and ACCOUNT NAME indicate that the values of these items varied; i.e., more than one user ran the program under more than one ACCOUNT NAME.

As another example, the ISAM program was the third heaviest user during prime shift, even though it was being run only 10% (L) of the time.

AMAR WORKLOAD REPORT

SITE: TOPS-20 SAMPLE SYSTEM

REPORT DESCRIPTION: WEEKLY REPORT BY (TYPICAL 8-HOUR SHIFTS) SYSTEM: TTSS

INPUT FILE: 83214 .DBO (FISCAL YEAR: 83 QUARTER: 2 MONTH: 1 WEEK: 4 WEEKDAYS)

(A) (B) (C) (D) 0: 0 - 7:59 WEEKDAYS (-HOLIDAYS) (E) FROM: MON 25-OCT-82 (F) TO: FRI 29-OCT-82 (G) (5 DAYS) MEASURED: 100% = 40.01 HOURS

GROUPED BY: PNAM

SORTED BY: CPU% (K) CUTOFF: 1.50% OF CPU (H) (I)

JOB #	FRK #	AVG JOBS	AVG FRKS	IN MEM	DEMD	USER NAME	ACCOUNT NAME	PRGRM NAME	PAGES (WS)	CPU%	SWAP PF/S	FILE PF/S	IFA	RSP /MIN	SEC /RSP	SR B TTY LOGIN AT DAY TIME
		15.7	74.8	43.6	0.98	**********INTERVAL TOTALS**********			36.4	48.14	0.1	3.6	143	204.2	0.0	1
		1.0	2.0	1.2	0.33	OPERATOR		(J) PTYCON	12.8	17.64	0.0	0.0	6805	4.3	0.0	1 T 206
		0.1	0.3	0.3	0.12	AT.PRODUCTION	670.A0000002	1038BS	66.5	5.32	0.0	0.5	114	0.3	1.0	2 B
		0.1	0.1	0.1	0.05	?????????????	???.?OO????????	ISAM (L)	29.3	2.53	0.0	0.2	124	0.4	0.5	2 B
		0.0	0.1	0.1	0.03	AT.PRODUCTION	670.A0000002	9736DE	48.3	2.41	0.0	0.2	109	0.0	0.4	1 B
		0.1	0.1	0.1	0.05	AT.PRODUCTION	670.A0000002	1274BS	85.7	2.13	0.0	0.2	121	0.1	1.1	2 B
		2.7	6.7	4.5	0.04	?????????????	??????????????	EXEC	43.4	2.08	0.0	0.3	75	12.4	0.0	1
		0.0	0.1	0.1	0.03	AT.PRODUCTION	670.A0000002	0768BS	38.6	1.72	0.0	0.1	143	0.0	0.0	0 B 217
		0.0	0.1	0.1	0.03	?????????????	6??.????????	COPY	49.5	1.68	0.0	0.2	78	6.3	0.0	1 B
		0.4	1.2	1.2	0.03	AT.INQUIRY	670.A0000001	PBARPT	33.8	1.57	0.0	0.2	69	9.3	0.1	1 T
		0.1	0.2	0.2	0.04	?????????????	??????????????	DUMPER	22.8	1.50	0.0	0.2	72	50.1	0.0	1 B
		4.5	10.7	7.7	0.76	********SUBTOTALS THRU CUTOFF*********			34.5	38.59	0.1	2.3	181	83.2	0.0	1
		11.2	64.1	35.9	0.22	********SUBTOTALS AFTER CUTOFF********			43.0	9.55	0.1	1.3	79	120.9	0.0	1

7:59 - 15:59 WEEKDAYS (-HOLIDAYS) FROM: MON 25-OCT-82 TO: FRI 29-OCT-82 (5 DAYS) MEASURED: 99% = 39.86 HOURS

GROUPED BY: PNAM

SORTED BY: CPU% CUTOFF: 1.50% OF CPU

Figure 2-3

JOB #	FRK #	AVG JOBS	AVG FRKS	IN MEM	DEMD	USER NAME	ACCOUNT NAME	PRGRM NAME	PAGES (WS)	CPU%	SWAP PF/S	FILE PF/S	IFA	RSP /MIN	SEC /RSP	SR	B	TTY	LOGIN AT DAY TIME
		65.5	236.8	155.1	5.13	**********INTERVAL TOTALS**********			38.0	69.74	2.1	11.6	57	661.4	0.1	4			
		4.1	13.5	11.1	1.38	AT.???????	???????????	REPORT	24.5	11.91	0.1	4.0	32	46.7	0.4	7	T		
		4.9	14.5	12.2	0.77	AT.INQUIRY	670.A0000001	PBARPT	27.0	8.72	0.1	2.0	44	58.7	0.2	4	T		
		1.5	3.1	1.7	0.26	?P????0??	??????????	PTYCON	13.8	7.01	0.0	0.0	2769	31.5	0.0	3	T		
		7.8	29.8	25.1	0.15	AT.INQUIRY	???????????	AT	29.3	4.71	0.1	0.5	78	49.0	0.2	2	T		
		7.2	21.5	15.9	0.23	AT.INQUIRY	???????????	CASHAP	55.1	4.65	0.2	0.9	50	92.4	0.1	3	T		
		0.3	0.5	0.5	0.25	MAGNUM		FSTCPY	99.6	2.78	0.0	0.0	394	0.1	2.7	16	B		
		4.7	11.9	9.8	0.13	AT.INQUIRY	???????????	ATINQ	31.9	2.70	0.2	0.4	56	27.2	0.2	3	T		
		5.7	13.0	7.1	0.33	????????????????	????????????????	EXEC	33.2	1.75	0.1	0.3	56	20.4	0.2	3			
		0.6	1.7	1.5	0.19	AT.INQUIRY	670.A0000001	PBAMN	58.2	1.73	0.1	0.1	110	6.5	0.2	4	T		
		0.2	0.3	0.3	0.07	IRWIN	639.830809.2000	IRSML1	25.4	1.51	0.0	0.0	1783	33.7	0.1	3	T		
		37.1	110.1	85.1	3.77	********SUBTOTALS THRU CUTOFF********			34.0	47.47	1.0	8.3	57	366.1	0.2	4			
		28.5	126.7	69.9	1.36	********SUBTOTALS AFTER CUTOFF********			49.1	22.27	1.1	3.3	58	295.3	0.1	3			

15:59 - 23:59 WEEKDAYS (-HOLIDAYS) FROM: MON 25-OCT-82 TO: FRI 29-OCT-82 (5 DAYS) MEASURED: 89% = 35.93 HOURS

GROUPED BY: PNAM

SORTED BY: CPU% CUTOFF: 1.50% OF CPU

JOB #	FRK #	AVG JOBS	AVG FRKS	IN MEM	DEMD	USER NAME	ACCOUNT NAME	PRGRM NAME	PAGES (WS)	CPU%	SWAP PF/S	FILE PF/S	IFA	RSP /MIN	SEC /RSP	SR	B	TTY	LOGIN AT DAY TIME
		22.1	94.8	54.2	1.43	**********INTERVAL TOTALS**********			37.8	58.80	0.4	7.4	86	555.2	0.1	2			
		0.5	1.0	1.0	0.29	????????????????	????????????????	DUMPER	26.1	10.48	0.0	1.7	71	297.2	0.0	2			
		1.1	2.1	1.2	0.19	?P????0??	??????????	PTYCON	13.0	8.65	0.0	0.0	16284	24.3	0.0	1	T		
		0.1	0.3	0.3	0.13	????????????????	????????????	0768BS	41.4	8.22	0.0	0.4	196	1.1	0.0	1	B		
		4.2	9.9	6.3	0.09	????????????????	????????????????	EXEC	51.9	3.57	0.1	0.5	69	29.1	0.1	2			
		0.9	3.1	3.1	0.10	AT.???????	???.A0000007	REPORT	27.1	3.18	0.0	0.6	51	14.8	0.1	2	T		
		0.1	0.1	0.1	0.05	AT.PRODUCTION	670.A0000002	0944BS	52.3	2.99	0.0	0.4	73	0.1	1.1	1	B	217	
		0.1	0.1	0.1	0.05	??.??????????	6??.????????	COPY	59.1	2.48	0.0	0.1	208	0.5	0.4	2			
		6.9	16.6	12.0	0.89	********SUBTOTALS THRU CUTOFF********			31.5	39.57	0.1	3.9	110	367.0	0.0	2			

Figure 2-3 (continued)

15.2 78.2 42.1 0.54 **********SUBTOTALS AFTER CUTOFF********** 48.3 19.23 0.2 3.5 60 188.2 0.1 2

Figure 2-3 (continued)

2.3.3 Batch Vs. Timesharing Report

Figure 2-4 is a sample of the default report WCDY0.RP3 and shows daily resource usage by all batch jobs (A) combined vs. all timesharing jobs combined on an 8-hour (B) shift basis.

Note that the average batch job (C) used 12% of the CPU during the prime time period (8:00-16:00). The average timesharing job (D) used .8%. Average batch job usage during third shift (0:00-8:00) was much higher, 31.6%. To find the average percent used by a job, divide the CPU% field by the AVG JOBS field. While batch jobs can be heavy CPU users, the amount of CPU time that batch and timesharing jobs are allowed to consume over a given period can be regulated, to some extent, by the system scheduler. By using the scheduler, it is possible to favor timesharing jobs over batch work during the day and vice versa at night. By reviewing batch vs. timesharing usage over a .longer period, such as a month, it is possible to get an approximate idea of how much additional load to expect when adding users to your system.

From running a special report grouped by USR1, USR2, and USR3 (User Name), we could see that although an average of 14.8 timesharing jobs were running during the evening shift, almost two thirds were OPERATOR jobs. For an example of such a report see (Figure 2-5).

```
                                        AMAR WORKLOAD REPORT

   SITE:  TOPS-20 SAMPLE SYSTEM                                                    SYSTEM:  TTSS

   REPORT DESCRIPTION:  SHIFT REPORT BY BATCH VS. TIMESHARING

   INPUT FILE:  832143.DBO    (FISCAL YEAR: 83   QUARTER: 2   MONTH: 1   WEEK: 4   DAY: 3   TUESDAY)
```

FROM: 0: 4:18 ON TUESDAY 26-OCT-82 TO: 8: 4:25 ON TUESDAY 26-OCT-82 INTERVAL: 8: 0: 7 (B) MEASURED: 100%

GROUPED BY: BATCH (A)

SORTED BY: CPU%

JOB #	FRK #	AVG JOBS	AVG FRKS	IN MEM	DEMD	USER NAME	ACCOUNT NAME	PRGRM NAME	PAGES (WS)	CPU%	SWAP PF/S	FILE PF/S	IFA	RSP /MIN	SEC /RSP	SR	B	TTY	LOGIN AT DAY TIME
		15.9	74.0	37.9	0.65	**********INTERVAL TOTALS**********			48.0	41.44	0.2	5.1	87	241.3	0.0	1			
		1.1	2.2	2.2	0.50	?T???????????	6??.AOOOOO?????	??????	51.9	34.81	0.1	3.9	96	102.9	0.0	1	B		(A)
		14.8	71.8	35.6	0.14	????????????	????????????	??????	33.9	6.63	0.1	1.2	59	138.4	0.0	1	T		

FROM: 8: 4:25 ON TUESDAY 26-OCT-82 TO: 16: 0:48 ON TUESDAY 26-OCT-82 INTERVAL: 7:56:22 MEASURED: 100%

GROUPED BY: BATCH

SORTED BY: CPU%

JOB #	FRK #	AVG JOBS	AVG FRKS	IN MEM	DEMD	USER NAME	ACCOUNT NAME	PRGRM NAME	PAGES (WS)	CPU%	SWAP PF/S	FILE PF/S	IFA	RSP /MIN	SEC /RSP	SR	B	TTY	LOGIN AT DAY TIME
		69.8	240.8	163.6	6.18	**********INTERVAL TOTALS**********			42.8	70.52	2.3	10.8	62	660.8	0.2	4			
		68.6	238.4	155.1	4.82	????????????	????????????	??????	33.1	53.2(D)	2.0	9.5	54	626.1	0.2	4	T		
		1.2	2.4	2.4	1.17	??????????	????????????	??????	82.3	14.39 (C)	0.1	0.9	159	1.0	2.4	15	B		0

FROM: 16: 0:48 ON TUESDAY 26-OCT-82 TO: 0: 0:56 ON WEDNESDAY 27-OCT-82 INTERVAL: 8: 0: 8 MEASURED: 100%

GROUPED BY: BATCH

SORTED BY: CPU%

JOB #	FRK #	AVG JOBS	AVG FRKS	IN MEM	DEMD	USER NAME	ACCOUNT NAME	PRGRM NAME	PAGES (WS)	CPU%	SWAP PF/S	FILE PF/S	IFA	RSP /MIN	SEC /RSP	SR	B	TTY	LOGIN AT DAY TIME

Figure 2-4

2.3.4 User Name Report

Figure 2-5 is a sample of the default report WCMN0.RP4 and shows monthly resource usage by User Name on an 8-hour shift basis. It can be used to determine which users consume the most resources on the system.

The second detail line (A) of the first subreport shows that there were an average of 8.4 OPERATOR jobs logged on during this period. These jobs used a total of 10.52% of the CPU. Question marks in the account name and program name fields indicate that values of these items varied, i.e. there were several programs run by OPERATOR jobs under several accounts.

It is also possible to get a similar report grouped only by Account Name or by both Account Name and User Name.

AMAR WORKLOAD REPORT

SITE: TOPS-20 SAMPLE SYSTEM SYSTEM: TTSS

REPORT DESCRIPTION: MONTHLY REPORT BY USER (TYPICAL 8-HOUR SHIFTS)

INPUT FILE: 8321 .DBO (FISCAL YEAR: 83 QUARTER: 2 MONTH: 1 WEEKDAYS)

0: 0 - 7:59 WEEKDAYS (-HOLIDAYS) FROM: MON 4-OCT-82 TO: FRI 29-OCT-82 (20 DAYS) MEASURED: 78% = 125.97 HOURS

GROUPED BY: USR1 USR2 USR3

SORTED BY: CPU%

JOB #	FRK #	AVG JOBS	AVG FRKS	IN MEM	DEMD	USER NAME	ACCOUNT NAME	PRGRM NAME	PAGES (WS)	CPU%	SWAP PF/S	FILE PF/S	IFA	RSP /MIN	SEC /RSP	SR	B	TTY	LOGIN AT DAY TIME
		14.3	65.2	48.3	1.16	**********INTERVAL TOTALS**********			46.8	53.50	0.2	6.5	88	435.0	0.0	2			
		1.0	2.0	2.0	0.69	AT.PRODUCTION	????A??????????	??????	54.4	36.58	0.1	4.6	84	48.4	0.0	1			
		8.4	45.1	33.3	0.24	OPERATOR	????????????????	??????	26.1	10.51	0.0	0.5	216	348.0	0.0	2			
		1.0	3.2	3.0	0.08	AT.INQUIRY	????????????	??????	36.9	2.22	0.0	0.3	67	15.2	0.1	2	T		
		0.1	0.2	0.2	0.08	PZ.PRODUCTION	??????????	??????	56.1	1.95	0.0	0.7	37	6.2	0.1	1	B		
		0.0	0.1	0.1	0.02	ETAMAR.DBS	669.A0000000	??????	43.5	0.74	0.0	0.1	55	0.7	0.1	2	B		
		0.1	0.1	0.1	0.01	BARBARA		??????	31.3	0.34	0.0	0.1	61	0.1	0.2	2			
		0.0	0.0	0.0	0.01	HANSON		??????	40.3	0.32	0.0	0.0	351	0.0	0.1	3	B		
		0.0	0.0	0.0	0.01	SHARED	???.?00???.?000	??????	27.1	0.20	0.0	0.0	63	0.6	0.1	2	B		
		0.2	0.5	0.4	0.00	PZ.RECEIVING	674.831325	??????	61.3	0.10	0.0	0.0	28	0.8	0.3	3	T		
		2.0	4.0	3.0	0.00	ETAMAR.DCOL	669.A0000000	??????	21.9	0.10	0.0	0.0	91	5.0	0.1	3	T		
		0.0	0.1	0.1	0.00	IP.USER	6??.820375	??????	67.8	0.09	0.0	0.0	72	0.5	0.1	2			
		0.0	0.1	0.1	0.00	AT.USER	670.A0000005	??????	37.4	0.07	0.0	0.0	161	7.1	0.0	1	T		
		0.1	0.2	0.2	0.00	PZ.PURCHASING	674.831325	??????	67.3	0.07	0.0	0.0	32	0.6	0.3	3	T		
		0.0	0.0	0.0	0.00	BFS.MACINNES	676.A0000000	??????	63.1	0.03	0.0	0.0	129	0.1	0.1	1			
		0.0	0.0	0.0	0.00	IP.PROD	665.820305	??????	46.2	0.02	0.0	0.0	149	0.0	0.4	2	B	215	
		0.0	0.0	0.0	0.00	ADMINISTRATION	669.A0000000	????	26.9	0.02	0.0	0.0	78	0.0	0.2	2	B		
		0.0	0.1	0.1	0.00	AN.USER		0001BQ	58.6	0.02	0.0	0.0	40	0.4	0.1	2	T		

Figure 2-5

2.4 HOW TO RUN THE PROGRAMS

2.4.1 Data Collection

The Data Collection program, WCDC20, should be run as a SYSJOB subjob. There will be a corresponding subjob to collect data for System AMAR.

The following commands should be inserted into the SYSJOB.RUN file to ensure automatic startup and continuous data collection:

```
JOB n \LOG amar-dir
ENABLE
SYSDPY E
CONNECT struc:<amar-dir>
RUN WCDC20
\
```

These commands may be entered directly to SYSJOB to get WCDC20 started the first time. SYSDPY must be run before WCDC20 in order to insert some JSYS code into the monitor. The last line may be used to restart WCDC20 if it has stopped because of disk parity errors or the like.

WCDC20 creates two output files named WC.IN0 and WC.IN1. These output files are updated after each checkpoint interval.

2.4.2 Generating Automatic Reports

There are two programs used in the automatic reporting process - W2UPD and W2RPTB. W2UPD massages the incremental files output by the data collection program and creates the database files. W2RPTB is the report generating program which operates on the database files. These programs are normally run as part of a nightly batch stream, W2RPTB.CTL, which is self-submitting. By using special wild carded filenames (described under the INPUT FILE.EXT = command in the Appendix called, "Report Program (W2RPT and W2RPTB) Dialogue"), it is possible to generate daily, weekly, and monthly reports through this one stream without operator intervention. The reporting program W2RPTB

recognizes when a fiscal day, week or month has ended and then produces the appropriate daily, weekly and monthly reports. An exception occurs if the stream is not run every day. Processing will get several days behind. Then only the first fiscal report type for the first fiscal period encountered will be produced. There are six types of fiscal period in Workload AMAR. These periods correspond to the six types of files listed under the "DORPT:" step of W2RPTB.CTL. See the Appendix called "Workload

AMAR Batch Stream - W2RPTB.CTL".

Four default reports are supplied with the package. These reports are described in the previous section and may be produced at your option on a daily, weekly, or monthly schedule or not at all. You can define special reports through the W2RPTB program dialogue. The dialogue responses can be added to the daily batch stream. The special reports will then be produced automatically.

2.4.3 Generating Special Reports

2.4.3.1 What Program Do I Use? -

There are two programs for generating special reports - W2RPTB and W2RPT. For detailed explanations of report program dialogues, see the Appendix called "Report Program (W2RPT and W2RPTB) Dialogues".

Use W2RPTB to report from the workload database. See Figure 2-6 for an example of W2RPTB dialogue and the resultant report.

Use W2RPT to report from yesterday's or today's data for one of two reasons:

1. You want to look at today's data which will not be in the database until after midnight.

2. You want to look at yesterday's data for intervals less than one hour or not starting and/or ending on hour boundaries.

```
@RU AM:W2RPTB

REPORT DESCRIPTION = REPORT SHOWING EXTRA DETAILS (1)

INPUT  FILE.EXT = 832143.DBO (2)

OUTPUT FILE.EXT = WORKDT.RPT (3)
GROUPING FILE.EXT =

ENTER DESIRED START AS     HH MM: 8 (4)
ENTER DESIRED END   AS     HH MM: 16 (5)
ENTER DESIRED INTERVAL SIZE AS  HH MM: (6)

ENTER MAXIMUM DETAIL LINES PER INTERVAL:
ENTER CPU% CUTOFF: 1.0 (7)

EXTRA DETAILS? (Y OR N): Y (8)

ANY SPECIAL MASKS OR SORT ORDERS? (Y OR N): N (9)

ID ITEM    0-0: ACC1 (10)

ID ITEM    0-1: ACT2 (11)

ID ITEM    0-2: ACT3 (12)

ID ITEM    0-3:

SORT ITEM  1-0: CPU% (13)

SORT ITEM  1-1:

SORT ITEM  2-0:

MORE REPORTS? (Y OR N): N
```

Sample W2RPTB Dialogue

Figure 2-6

1. Free form report description. This report shows the extra detail line. Refer to the Appendix called "Workload AMAR Item Definitions" for a description of the items on this line.

2. Daily file for October 26, 1982 (FY83, second quarter, first month, fourth week, third day.)

3. Report filename.

4. Start the report at 8:00 AM. Minutes, seconds, and day default to 0 if only the hour is specified. If carriage return only had been entered, the report would have started at the beginning of the file.

5. The report will stop at 16:00 (4:00 PM). If carriage return only had been entered, the report would have stopped at the end of the file.

6. Since carriage return was entered, the report will cover the entire period between 8:00 AM and 4:00 PM.

7. Only those detail lines with CPU usage of 1% or more will be shown.

8. Extra detail lines will be printed.

9. No other special features will be used.

10., 11. and 12. The data will be grouped by account name.

13. The major sort is by CPU%.

```
                                    AMAR WORKLOAD REPORT

   SITE:  TOPS-20 SAMPLE SYSTEM                                                    SYSTEM:  TTSS

   REPORT DESCRIPTION:  REPORT SHOWING EXTRA DETAILS

   INPUT FILE:  832143.DBO    (FISCAL YEAR: 83   QUARTER: 2   MONTH: 1   WEEK: 4   DAY: 3   TUESDAY)
```

FROM: 8: 4:25 ON TUESDAY 26-OCT-82 TO: 16: 0:48 ON TUESDAY 26-OCT-82 INTERVAL: 7:56:22 MEASURED: 100%

GROUPED BY: ACT1 ACT2 ACT3

SORTED BY: CPU% CUTOFF: 1.00% OF CPU

JOB #	FRK #	AVG JOBS	AVG FRKS	IN MEM	DEMD	USER NAME	ACCOUNT NAME	PRGRM NAME	PAGES (WS)	CPU%	SWAP PF/S	FILE PF/S	IFA	RSP /MIN	SEC /RSP	SR	B	TTY	LOGIN DAY	AT TIME
398167	1340	4644	284207	546032	1223763	322793	542210	183119		0	41840		573	2712	2661		594		3	⊗
	69.8	240.8	163.6	6.18	**********INTERVAL TOTALS**********				42.8	70.52	2.3	10.8	62	660.8	0.2	4				
	561LI	391LO	2969FC	2180FT	0.LD	0.81U	0.00G	4.53B	0.10S	0.62R	0.12W	0.00Q	0.00M							
	37.1	121.9	99.2	3.70	AT.INQUIRY	670.A0000001		??????	31.1	38.07	1.0	7.7	50	344.7	0.2	5				
	449LI	308LO	2322FC	1663FT	0.LD	0.44U	0.00G	2.69B	0.05S	0.47R	0.05W	0.00Q	0.00M						0	
	15.4	59.8	18.7	1.10	???????????			??????	83.7	15.72	0.3	0.6	184	53.6	0.1	4				
	32LI	19LO	146FC	102FT	0.LD	0.17U	0.00G	0.87B	0.01S	0.03R	0.01W	0.00Q	0.00M							
	6.9	19.8	17.2	0.15	PZ.???????????	674.831325		??????	54.8	2.98	0.2	0.7	45	23.2	0.4	3	T		0	
	35LI	28LO	212FC	162FT	0.LD	0.04U	0.00G	0.06B	0.01S	0.03R	0.02W	0.00Q	0.00M						0	
	0.7	1.3	1.3	0.14	IRWIN	639.830809.2000		??????	28.7	2.29	0.0	0.0	487	74.1	0.1	4	T		0	
	2LI	1LO	14FC	11FT	0.LD	0.03U	0.00G	0.11B	0.00S	0.00R	0.00W	0.00Q	0.00M						0	
81	0.3	0.5	0.5	0.24	OHARA	155..0000000000		??????	22.9	1.32	0.0	0.0	238	0.4	0.2	5	T	12	0	8:19
	1LI	1LO	3FC	3FT	0.LD	0.01U	0.00G	0.22B	0.00S	0.00R	0.00W	0.00Q	0.00M						0	
0	1.0	20.0	8.7	0.10	OPERATOR	OPERATOR		SYSJOB	32.4	1.30	0.2	0.1	46	58.1	0.1	4	T	DET	-4	5:45
	0LI	0LO	18FC	18FT	0.LD	0.02U	0.00G	0.06B	0.01S	0.00R	0.01W	0.00Q	0.00M							
	0.9	1.8	1.1	0.20	AT.LEUNG	670.A00000000		??????	15.1	1.13	0.0	0.4	32	2.9	0.6	12	T		0	
	2LI	1LO	14FC	10FT	0.LD	0.01U	0.00G	0.16B	0.00S	0.03R	0.00W	0.00Q	0.00M						0	
	62.2	225.1	146.7	5.63	********SUBTOTALS THRU CUTOFF*********				41.1	62.80	1.8	9.6	63	556.9	0.2	4				
	521LI	358LO	OFC	OFT	0.LD	0.72U	0.00G	4.17B	0.08S	0.57R	0.09W	0.00Q	0.00M							
	7.6	15.7	16.8	0.56	********SUBTOTALS AFTER CUTOFF********				60.2	7.72	0.5	1.2	54	103.9	0.2	4				
	40LI	33LO	2969FC	2180FT	0.LD	0.09U	0.00G	0.36B	0.02S	0.06R	0.02W	0.00Q	0.00M							

⊗ This line only shows when EXTRA DETAILS are requested - it should be ignored.

Figure 2-6 (continued)

2.4.3.2 Special Report On Yesterday's Data -

To obtain a special report on yesterday's data down to the 5 minute level, run W2RPT according to the general directions in the Appendix called "Report Program (W2RPT and W2RPTB) Dialogue", specifying an output filename of the form WCX.ext where WCX represents yesterday's incremental files and ext is not .IN0, .IN1, .RP?, .EXE, or .RAW. Before you run W2RPT, make sure that there are two files for yesterday named WCX.IN0 and WCX.IN1, as these are the input files required by W2RPT. If an hourly or higher level report is needed, the W2RPTB program and the database file for yesterday may be used.

2.4.3.3 Special Report On Today's Data -

To obtain a special report on today's data run W2RPT according to general directions in the Appendix called "Report Program (W2RPT and W2RPTB) Dialogue", specifying an output filename of the form WC.ext, where .ext is not .IN0, .IN1, .RP?, or .EXE.

2.4.4 Examining/Changing The Workload Holidays (WCFIX)

In addition to specifying holidays for Workload AMAR, holidays must also be specified for System AMAR. Although the holidays should be the same throughout AMAR, the processes to set them are separate. The WCFIX program is used to specify holidays for Workload AMAR. For System AMAR, see the Section entitled "Examining/Changing Database Parameters (AMRGEN)". The holiday list in Workload AMAR is currently limited to a maximum of 18 entries. Once the list is full, adding a new entry will cause the earliest date to be deleted from the list. Thus it should never be necessary to explicitly delete holidays unless they have been set by mistake. On the other hand, you should not try to set more than 18 future holidays because the earliest would be deleted from the list before it could do its job of making the corresponding data enter the database as holiday data. WCFIX can also be used to temporarily reset the grace period. The grace period has a threefold purpose:

1. It allows automatic processing to be resumed after as many as 7 days (the standard default) with no special action.

2. It prevents processing of more than 7 days of data at one time which could cause disk problems.

3. It avoids database update when the operator has set the system date more than 7 days into the future. If database

update were allowed in that case, dummy data would be entered into the database which could not be replaced by the real data and old data would be deleted prematurely. In an extreme case, if the operator set the date a year ahead and the longest retention in the database was a year, the whole database could be filled with dummy data and it would be impossible to enter any real data for the next year.

Before using WCFIX to temporarily extend the grace period, you should make sure there is enough disk space to process the extra days, make sure the system date is correctly set, and make sure the workload data collection program is collecting data. If the normal batch stream is going to be used, be sure to request enough time and avoid multiple submissions. The next time W2UPD runs successfully, it will set the grace period back to the default of 7 days.

WCFIX is command driven. Both lower and upper case are valid. WCFIX prompts with an asterisk (*).

Valid Commands:

HELP

Function: To provide a brief sysnopsis of the valid commands.

S H yymmdd

Function: To set a holiday (yy = the normal calendar year; mm = the month; and dd = the day). A holiday must be set before the corresponding data is entered into the database. Otherwise, setting the day as a holiday will have no useful effect.

D H yymmdd

Function: To delete a holiday (yy = the normal calendar year; mm = the month; and dd = the day). Once the corresponding data has been entered into the database as a holiday, deleting the holiday will have no useful effect other than to free up a slot in the holiday list.

L H

Function: To list the holidays. Holidays will be listed in the following format: fiscal date followed by a 3 character day abbreviation (MON-FRI) followed by the normal calendar date.

S G nnn

Function: To temporarily set the grace period to nnn days.

```
EXIT
E D
```

Function: To exit the program and return to monitor level.

2.5 PROCEDURE FOR RUNNING W2RPTB.CTL

2.5.1 Overview Of W2RPTB.CTL

This stream runs daily. It takes the workload data which has been collected by WCDC20 since the last time W2RPTB.CTL ran, updates the database, and produces daily reports (and weekly and monthly reports if appropriate). See the Appendix called "Workload AMAR Batch Stream - W2RPTB.CTL" for a sample of the stream and step descriptions.

2.5.2 Resource Requirements

To run the workload system on a continuous basis, you will need a SYSJOB subjob to run the WCDC20 program and a permanently mounted disk area with at least 3200 pages available for raw data and programs. The average number of simultaneous users is the key factor determining daily file size. The size of your workload database will probably range between 3500 and 20000 pages. Size depends partly on the complexity of your workload and partly on the retention periods you specify. This space may be on a pack separate from the raw data, but should also be permanently mounted, if possible. See the Appendix called "Installation and Resource Requirements" for a more complete discussion of resource requirements.

2.5.3 Submission

The stream resubmits itself in most cases, whether completion is successful or not. This is because it is important not to let raw data build up on disk. There should always be a W2RPTB.CTL in the submit queue, set to run /AFTER:TODAY.

2.5.4 Restart Procedure

If a system crash occurs while the stream is running, the stream automatically restarts at the appropriate checkpoint. The user should never have to resubmit W2RPTB.CTL unless the submit queue entry is destroyed. Each major step is checkpointed.

Blank Page

APPENDIX A

SYSTEM AMAR ITEM DEFINITIONS

This appendix contains a definition for each item which can be monitored by System AMAR. Items are listed in alphabetical order by the title which appears on the System AMAR reports. The title is the first line in each definition below. Preceeding logical and physical names as well as special characters such as # or % are ignored. Following the title line is a line which contains a 4 character code which uniquely identifies the item and its corresponding records in the System AMAR database.

Beneath each item code is a letter, in parentheses, which indicates whether the item is metered (M) or snapped (S). Metered implies that the value of the item is accurate, regardless of the size of the sample interval. Snapped means that the sample taken is a "snapshot" of the item at sample time. Subsequent or prior to the snapshot, the values could have been very different. Thus, the accuracy of the values' correspondence to what is actually happening on the system depends on the number of samples taken during the reporting period.

Item values reported as a "per second" count were obtained by dividing the value observed at sample time by the number of seconds in the sample interval.

DECSYSTEM-20 ITEMS

ACTIVE FORKS
NRUN - Number of Active Forks
(S)

This is the average number of active processes, e.g., those competing for use of the CPU or disk.

% AMAR CLK TIME
XAMT - Percent of real time measured by AMAR
(M)

The percentage of time that the data collection program was running and measuring system performance.

% BACK GND TIME
BGND - Background Tasks Percent of Real Time
(M)

The percent of real time used by the monitor to do background tasks. One background task is moving terminal input characters from a system-wide buffer to the individual terminal input buffers and echoing them.

BAL SET ADJS/SEC
AJBL - Balance Set Adjustments per Second
(M)

The average number of times per second that the system adjusted the balance set.

BALNCE SET FRKS
NBAL - Number of Forks in the Balance Set
(S)

Average number of processes in the balance set.

BAL SET SWPS/SEC
NREM - Balance Set Forks Removed per Second
(M)

The average number of working sets removed from the balance set and swapped out of memory per second.

BIAS CONTROL
KNOB - Bias Control Setting
(S)

The value (1-20) of the bias control setting.

% BS WQ RD WAIT
DSKR - Percent of Balance Set Wait Queue Forks Blocked for Disk
(M) Read

The percent of processes in the balance set wait queue (BSWT) which are waiting for the completion of a disk file read.

% BS WQ SWP WAIT
SWPR - Percent of Balance Set Wait Queue Forks Blocked for
(M) Swapping

Percent of processes in the balance set wait queue (BSWT) which are awaiting the completion of a page swap into memory.

% BS WQ WR WAIT
DSKW - Percent of Balance Set Wait Queue Forks Blocked for Disk
(M) Write

Percent of processes in the balance set wait queue (BSWT) which are awaiting the completion of a disk file write.

CONTEXT SWTS/SEC
CTXS - Context switches per second
(M)

The average number of context switches per second performed by the scheduler.

CPU IDLE TIME
CPIO - Graphed CPU Idle Time
(M)

The percent of real time the system was idle without any demand on the CPU or idle with at least one process blocked for disk file read or write. This metric is reported only in the CPU utilization graph.

CPU OVHD TIME
CPOO - Graphed CPU Overhead Time
(M)

The percent of real time the system was performing background tasks or scheduling processes. This metric is reported only in the CPU utilization graph.

% CPU UTIL
CPU - Graphed Percent of CPU Utilization
(M)

The percent of real time used to perform work directly related to user processes, such as paging and executing processes. This is only reported in the CPU utilization graph.

DSK RDS PGS/SEC
DKRD - Disk File Page Reads per Second
(M)

The average number of disk file pages read per second from the file system.

DSK WRS PGS/SEC
DKWR - Disk File Page Writes per Second
(M)

The average number of disk file pages written per second to the file system.

FORKS BS WAIT
BSWT - Number of Forks in the Balance Set Wait Queue
(S)

The average number of processes in the balance set that are in a wait state.

FORK WAKS/SEC
WAKE - Fork Wakeups per Second
(M)

The average number of process wakeups per second. Some events and processes that awaken processes are: terminal input and output, process termination, TIMER, DISMS, ENQ, IPCF, and IIC.

FREE MEM PGS
FPGS - Number of Free Pages of Memory
(S)

The average number of free pages in physical memory. Within the free pages the monitor maintains a page cache, the use of which significantly affects system performance.

%??????? FREE SPC
LUFS - Disk Pack (Logical Name) Percent of Free Space
(S)

Percent of free space on the logical structure.

FRK FREE PGS/SEC
LCCW - Local Pages of Memory Freed per Second
(M)

The average number of pages freed per second from a specific process by the local memory management service.

GEN FREE PGS/SEC
GCCW - Global Pages of Memory Freed per Second
(M)

The average number of pages freed per second by the global memory management service.

% IDLE IO TIME
FILW - CPU Idle and a Fork Blocked for Disk File Read or Write
(M)

The percent of real time during which the CPU was idle, no process was awaiting memory management service, and at least one active process was awaiting completion of a disk file read or write.

% IDLE SWP TIME
SWPW - CPU Idle and a Fork Blocked for Swapping
(M)

The percent of real time during which the CPU was idle and at least one active process was awaitng the completion of a page swap.

% IDLE TIME
IDLE - CPU Idle Percent of Real Time
(M)

The percent of real time during which the CPU was idle because there were no active processes.

LINES IN USE
TTYU - Number of Terminals Concurrently in Use
(S)

The average number of terminals concurrently assigned and in use.

LOCKED PAGES
LKPG - Number of Locked Memory Pages
(S)

The average number of locked memory pages. These include pages reserved by the monitor as terminal and magnetic tape buffers, etc.

MEM MGMT CYCS/SEC
NCOR - Memory Management Cycles per Second
(M)

The average number of global memory management cycles per second.

MEMORY WRK SETS
NWSM - Number of Working Sets in Memory
(S)

The average number of working sets in memory.

% MGMT MEM TIME
TCOR - Page Request Memory Management Percent of Real Time
(M)

The percent of real time used by the monitor to search for pages in memory that are not in any process's working set. This service is part of the global memory management cycle.

MTAS ASSIGNED
MTAU - Number of Magnetic Tape Drives Concurrently in Use
(S)

The average number of magnetic tape drives concurrently assigned and in use.

MT ????? PGS/SEC
MTIO - Magnetic Tape Drive Pages Transferred per Second
(M)

The average number of pages transferred per second to or from the physical magnetic tape drive.

% PAGING TIME
TRAP - Percent of Real Time to Resolve Page Faults
(M)

The percent of real time used by the monitor to resolve page faults, both from memory and disk. Note that here TRAP is always included in USED even though within the monitor it can be optionally excluded from time charged to users.

PG FAULT TRPS/SEC
NTRP - Page Fault Traps per Second
(M)

The average number of page fault traps per second. A page fault occurs when a process references a page in its virtual address space outside of its working set. A page fault is resolved either by locating that page already in memory, from the replaceable queue or from shareable pages, or by reading that page from disk.

PGS REAS MEMORY
AMEM - Pages of Reassignable Physical Memory
(S)

The average number of pages of physical memory available for allocation as monitor work space and to user processes.

PGS USER MEMORY
UMEM - Number of Memory Pages Available to User Forks
(S)

The average number of physical pages of memory available for allocation to user processes. This excludes the size of the monitor and all locked pages.

PTYS IN USE
PTYU - Number of Psuedo-Terminals Concurrently in Use
(S)

The average number of psuedo-terminals concurrently in use.

??????? RDS/SEC
LURD - Disk Pack (Logical Name) Page Reads per Second
(M)

The average number of page reads per second from the logical pack.

RQ SAVES PGS/SEC
RPQS - Replacement Queue Page Saves per Second
(M)

The average number of pages retrieved from the replacement queue per second to resolve page faults. The replacement queue is maintained in memory and as such can resolve a page fault without a disk read.

% SCHED TIME
SKED - Scheduler Percent of Real Time
(M)

The percent of real time used by the monitor to schedule processes for memory or CPU usage.

??????? SEKS/SEC
LUSK - Disk Pack (Logical Name) Seeks per Second
(M)

The average number of seeks per second on this logical disk pack.

SHARED PAGES
SHPG - Number of Shared Memory Pages
(S)

The average number of shared memory pages. The monitor can resolve a page fault for a shared page without a disk read if it is in memory.

SWAP RDS PGS/SEC
DMRD - Swapper Page Read per Second
(M)

The average number of disk pages read per second from the swapping area.

% SWAP SPC FREE
SWFS - Percent of Swapping Space Free
(S)

Percent of available swapping space.

SWP WRS PGS/SEC
DMWR - Swapper Page Writes Per Second
(M)

The average number of disk pages written per second to the swapping area.

SYSTEM RELOADS
XRLD - Number of System Reloads
(S)

Number of system reloads.

% SYSTEM UPTIME
XUPT - Percent of Time System Was Up
(M)

Percent of real time that the system was running. This may be longer than % AMAR CLK TIME.

TTY IN CHRS/SEC
TTIN - Terminal Input Characters per Second
(M)

The average number of characters input per second from all terminals on the system, including psuedo-terminals.

TTY INTR CHRS/SEC
TTCC - Terminal Interrupt Characters per Second
(M)

The average number of terminal interrupt characters transmitted per second.

TTY OUT CHRS/SEC
TTOU - Terminal Output Characters per Second
(M)

The average number of characters output per second to all terminals on the system, including psuedo-terminals and echoed input.

%UN ???? FREE SPC
PUFS - Disk Drive (Physical Unit) Percent of Free Space
(S)

The percent of free space left on the physical drive.

UN ???? RDS/SEC
PURD - Disk Drive (Physical Unit) Read per Second
(M)

The average number of pages read per second on this physical drive.

UN ???? SEKS/SEC
PUSK - Disk Drive (Physical Unit) Seeks per Second
(M)

The average number of seeks per second on this physical drive.

UN ???? WAIT Q
PUWQ - Disk Drive (Physical Unit) Position Wait Queue length
(S)

The length of the disk position wait queue on this drive.

UN ???? WRS/SEC
PUWR - Disk Drive (Physical Unit) Page Writes per Second
(M)

The average number of pages written per second on this physical drive.

% USED TIME
USED - Used Percent of Real Time
(M)

The percent of real time used by user processes. Note that here TRAP is always included in USED although it can be optionally excluded from the monitor's accounting of user time.

??????? WAIT Q
LUWQ - Disk Pack (Logical Name) Position Wait Queue Length
(S)

The length of the position wait queue on this logical pack.

WORK SET LDS/SEC
NLOD - Balance Set Forks Loaded per Second
(M)

The average number of working sets swapped into memory and loaded into the balance set per second.

WORK SET PGS
UPGS - Number of Memory Pages Allocated to All Working Sets
(S)

The average number of pages of memory which comprise all working sets.

??????? WRS/SEC
LUWR - Disk Pack (Logical Name) Page Writes per Second
(M)

The average number of pages written per second to this logical pack.

APPENDIX B

WORKLOAD AMAR ITEM DEFINITIONS

This appendix contains a definition for each item which appears in the Workload AMAR reports. Items are listed according to the order in which they appear on the detail lines. Each description contains: the item ID as it appears in the header or extra detail line of the report; the code used to select the item for GROUPING and/or SORTING when running the report dialogue; and the definition of the item. Items which are derived and cannot be used for grouping or sorting are flagged as such.

DECSYSTEM-20 ITEMS

JOB #
JOB - Job Number

As listed by SYSTAT, SYSDPY, WATCH, etc.

FRK #
FORK - Fork Number

System wide fork (process) number rather than relative fork within job. This number is not usually reported because one fork at a time is rarely looked at.

AVG JOBS
JELA - Average Number of Simultaneous Jobs

A job is counted in the detail line where its top fork fits. If this number is less than one, it may be interpreted as the fraction of the interval represented by the detail line.

AVG FRKS
FELA - Average Number of Simultaneous Forks

There are at usually at least two forks per job, the EXEC and a user process, but there may be more. Jobs which use a large number of forks unnecessarily are wasting an important resource (table space in memory).

IN MEM
MEMT - Average Number of Forks Simultanously in Memory

For best response, this number should be at least as large as the number of active forks (DEMD).

DEMD
derived - Demand Number of Forks

The number of forks actively competing for the CPU or disk. If this number is less than one, it may be interpreted as the fraction of the interval that the job was competing for CPU or disk.

USER NAME
USR1, USR2, USR3 - User Name

This is the first 15 characters of the logged in directory name. User name is 3 words long. It must be specified by using the three mnemonics above whenever it is used for grouping or sorting.

ACCOUNT NAME
ACT1, ACT2, ACT3 - Account Name

This is the first 15 characters of the account string. Account name is 3 words long. It must be specified by using the three mnemonics above whenever it is used for grouping or sorting.

PRGRM NAME
PNAM - Program Name

This is the "job program name". "Fork program name" is not available.

PAGES (WS)
derived - Average Number of Pages in the Working Set

This is the average fork working set size, not the job working set size as computed by WATCH.

CPU%
CPU% - CPU Percentage

This is the number of CPU seconds charged to this fork or set of forks during the interval divided by the length of the interval in seconds times 100 to make it a percentage.

SWAP PF/S
NSRW - Swap page Faults per Second

This is the rate of swap page faults that actually cause I/O. It does not include preloaded pages or pages saved from the replaceable queue.

FILE PF/S
NFRW - File Page Faults per Second

This is the rate of file page faults that actually cause I/O. It does not include preloaded pages or pages saved from the replaceable queue.

IFA
derived - Inter-fault Average

This is the average number of milleseconds of CPU used between page faults that actually cause I/O.

RSP/MIN
ICNT - Responses per Minute

This is the rate of responses which require no more than two seconds of CPU time.

SEC/RSP
derived - Seconds per Response

This is the average real time required to provide the responses which require no more than two seconds of CPU time.

SR
derived - Stretch Ratio

The total real time required to provide the responses divided by the total CPU time required.

B
BATCH - Batch Indicator

Batch jobs are indicated with a "B". All other jobs are indicated with a "T" for timesharing.

TTY
TTY- Line number

These are octal numbers and if greater than the CTY line number, they represent PTY's.

LOGIN AT DAY TIME
JLIT - Job Login Time

Day and time the job logged in. The day is relative to the end of the report interval. For example, 0 means the job logged in durng the day in which the interval ended; -1 means the previous day.

JLIT is the job login date/time.
JLDY is the job login date (left half of JLIT).
JLTM is the job login time (right half of JLIT).

EXTRA DETAILS

Normally, only one physical line is printed per logical detail line. If "extra details" are requested, a second physical line is printed for each logical detail line. Rather than adding an extra header line, the extra detail values are immediately followed by one or two-character mnemonic tags as follows:

LI
NJLI - Logins

Count not rate of logins during the interval. You must use the left half of the mask for NJLI (777777000000). See the Appendix called "Special Masks and Sort Orders".

LO
NJLI - Logouts

Count not rate of logouts during the interval. You must use the right half of the mask for NJLI (777777). See the Appendix called "Special Masks and Sort Orders".

FC
NFLI - Fork Creates

Count not rate of fork creates during the interval. You must use the left half of the mask for NFLI (777777000000). See the Appendix called "Special Masks and Sort Orders".

FT
NFLI - Fork Terminates

Count not rate of fork terminates during the interval. You must use the right half of the mask for NFLI (777777). See the Appendix called "Special Masks and Sort Orders".

LD
NWSL - Working Set Loads

Count not rate of working set loads during the interval.

The next eight items are fractions of real time which add up to demand time:

U
USED - Used time

The fraction of real time that the fork(s) used the CPU. Because of the way it is measured, used time may be inflated by priority interrupt time.

G
GRDY - Golist Ready Time

The fraction of real time that the fork(s) were active but

could not fit into the balance set.

B
BRDY - Balance Set Ready Time

The fraction of real time that the fork(s) were in the balance set waiting to use the CPU.

S
SWPR - Swap Wait Time

The fraction of real time that the fork(s) were waiting for pages to be swapped in.

R
FILR - Read Wait Time

The fraction of real time that the fork(s) were waiting for pages to be read from disk files.

W
FILW - Write Wait Time

The fraction of real time that the fork(s) were waiting for pages to be written to disk files.

Q
RPQW - Replaceable Queue Wait

The fraction of real time that the fork(s) were waiting for a free page (the replaceable queue was empty).

M
OTHR - Miscellaneous Other Wait States

The fraction of real time that the fork(s) were in other wait states.

Next comes a three-character field where:

The first character is "Y", "N" or blank.
FCREA - Fork Creates

The first character is "Y" if all samples represented included fork creates, "N" if no samples represented included fork creates, and blank otherwise.

The second character is "Y", "N", or blank.
FTERM - Fork Terminates

The second character is "Y" if all samples represented included fork terminates, "N" if none of the samples represented included fork terminates, and blank otherwise.

The third character is "Y", "N" or blank.
TOP - Top Forks

The third character is "Y" if all samples represented were for top forks, "N" if none of the samples represented were for top forks, and blank otherwise.

The last item which may appear on the extra detail line is:

FLIT - Fork Create Date/Time

Fork create date/time will be lined up under login date/time.

FLIT is fork create date/time.
FLDY is fork create date (left half of FLIT).
FLTM is fork create time (right half of FLIT).

APPENDIX C

RAW FILE PREPROCESSOR PROGRAM (AMARSD) DIALOGUE

To obtain special reports on the current System AMAR raw file (today's data), the AMARSD program must first be run. AMARSD always names its output file TODAY.DB. This file may then be input to the AMREPT program to obtain a Daily System Utilization, Disk, or Tape Report. It may also be examined via the AMARON or AMAREX programs. Any System AMAR raw file, including the current day's file, may be run through AMARSD.

.RUN AMARSD

System ID:

Requests the ID of the system whose raw file is to be examined.

Valid Response:

xxxx - 4 character system code

YYMMDD Date of File:

Requests the date of the raw file to be examined:

Valid Response:

yymmdd

Where yy = the normal calendar <u>not</u> fiscal calendar year; mm = the month; and dd = the day.

Prime Periods for yymmdd:

Requests the time period to be considered prime time. The user may enter any prime time period regardless of the specification in the database.

Valid Response:

Start time-End time

Up to four start time-end time pairs may be entered separated by commas. All time is of the format hhss where hh = the hour and ss = the minutes.

Once processing of AMARSD is completed, the following messages will appear:

[AMIHDS Hourly Data Stored for yymmdd]

[Use TODAY.DB as Input File to the AMREPT Program]

You may then run AMREPT, AMARON, or AMAREX to obtain the appropriate report.

APPENDIX D

REPORT PROGRAM (AMREPT) DIALOGUE

AMREPT may be used to generate automatic reports or reports on demand. In either case, a standard set of preformatted reports is obtained. The contents of the reports may be modified via changes to the xxxxDR.RFD File. Generating automatic reports is described in a previous section by that name.

Control-C (^C) may be used to exit at any time. To generate special reports, AMREPT may be run at the terminal as follows:

```
.RUN AMREPT

Report Code>
```

Requests the 2 character code of the standard report to be generated.

Valid Response:

```
DU  Daily System Utilization Report
WU  Weekly Utilization Report
MU  Monthly Utilization Report

WA  Weekly Trend Analysis Report
MA  Monthly Trend Analysis Report

WC  Weekly 'Typical Day' Report
MC  Monthly 'Typical Day' Report

DD  Daily Disk Report
WD  Weekly Disk Report
MD  Monthly Disk Report

DT  Daily Tape Report
WT  Weekly Tape Report
MT  Monthly Tape Report

EXIT
```

EXIT may be used with this query only to terminate the

program.

Default: On subsequent passes through the dialogue, carriage return <CR> will cause the last valid response to this query to be re-used.

Input File>

Requests the name of the AMAR database or the output file produced by the AMARSD program.

Valid Response:

```
AMAR
AMAR.DB
TODAY
TODAY.DB
```

AMAR or AMAR.DB specifies the system AMAR database as the input file. TODAY or TODAY.DB specifies the mini-database created by the AMARSD program as the input file.

Default: On subsequent passes through the dialogue, carriage return <CR> will cause the last valid response to this query to be re-used.

Dates>

Requests the dates of the report period to be used.

Valid Response:

```
start date-end date
AUTO
AUTO:start date-end date
AUTO=end date
AUTO=?
```

All dates are of the format yymmdd (where yy = the normal calendar (not fiscal) year; mm = the month; and dd = the day). If no data exists for the time period specified, the query is repeated.

The "start date-end date" response should be the only one used when generating special reports. It denotes the range of dates for generating one or more reports. A report will be generated for each fiscal period (implied by the Report code used in the "Report Code>" query) which ends within the date range. The exception is the Weekly/Monthly Trend Analysis Reports which will contain up to 13 weeks or 12 months of data on one report.

If a single daily report is desired, only one date need be entered.

The AUTO response should only be used in the AMAR.CTL jobstream. It will modify the section of the database that controls the automatic reporting capability. AUTO indicates that a check will be made to determine the date of the last fiscal period (day, week, or month) for which an automatic report of the same type has already been generated. The next appropriate fiscal period will be used for this report providing the necessary data is in the database. If AUTO reporting is behind several periods, the AUTO switch will bring you up-to-date. You will get reports for all intervening fiscal periods.

AUTO:start date-end date specifies a range of dates for which one or more reports are to be generated. In addition, the date of the last fiscal period reported is entered into the database as the date of the most recent report of this type. Further use of the AUTO response to generate similar reports will be based on this new date.

AUTO=end date modifies the database, resetting the date of the last automatic report of the type requested to this new date. No reports are produced.

AUTO=? requests a display of the date of the last automatic report of this type and the range of dates to be reported on next.

Default: On subsequent passes through the dialogue, carriage return <CR> will cause the last valid response to be re-used.

Print File>

Requests the filename of the report to be generated. Each report should have a unique name.

Valid Response: Any unique filename of the format filename.ext. File names reserved for AMAR use (see the Appendix called "Summary of Programs and Filenames") should not be specified.

Default: Carriage return <CR> will cause the query to be repeated.

Blank Page

APPENDIX E

ONLINE INQUIRY PROGRAM (AMARON) DIALOGUE

AMARON is the recommended program for examining either a single item/subitem or groups of items/subitems. AMARON is normally run at a terminal. The output, however, can either be displayed at the terminal or stored in a file for later processing. When output is stored in a file, the user has the option of retaining the report headers or automatically stripping them off. AMARON displays data in either of two formats - Tables of Average Values or Histograms. Refer to the section on Annotated Sample Reports for examples of these formats.

Control-C (^C) may be used at any point to terminate the program. The program will accept responses in either upper or lower case.

```
.RUN AMARON
```

DATABASE NAME:

Requests the filename of the database from which item and subitem values are to be examined.

Valid Response:

```
AMAR
AMAR.DB
TODAY
TODAY.DB
```

AMAR and AMAR.DB refer to the system AMAR database. TODAY and TODAY.DB refer to the output file produced by the AMARSD program.

Default: Carriage return will cause the query to be repeated.

Histogram Function:

Requests whether histogram data or average values should be reported.

Valid Response:

Y - Produce histogram report.
N - Produce tabular report of average values.

Default: Carriage return will cause the query to be repeated.

Output at (T)erminal or in (F)ile:

Requests whether the output should be displayed at the terminal or stored in a file.

Valid Response:

T - terminal
F - file

Default: Carriage return will cause the query to be repeated.

File ID:

This query is displayed only if "F" is specified in response to the Output at (T)erminal or in (F)ile query. It requests the filename of the output file.

Valid Response:

Filename
Filename.ext

Default: Carriage return will cause the query to be repeated.

Start Date:

Requests the start date of the report period.

Valid Response:

yymmdd

Where yy is the normal calendar not fiscal calendar year; mm is the month; and dd is the day.

Default: Carriage return will cause the query to be repeated.

End Date:

Requests the end date of the report period.

Valid Response:

yymmdd

Where yy is the normal calendar not fiscal calendar year; mm is the month; and dd is the day. If data for only one day is required, make the end date the same as the start date.

Default: Carriage return will cause the query to be repeated.

Granularity Level:

Requests the summary (fiscal) level of the data to be reported.

Valid Response:

H - Hour
S - Sample Group Interval (same as hour)
D - Day
W - Week
M - Month
HOURS-WEEK [HRS-WEEK or H-W] - Composite Week Hours
HOURS-MONTH [HRS-MONTH or H-M] - Composite Month Hours

Composite refers to the type of data normally displayed in the Weekly/Monthly 'Typical Day' Reports. If TODAY.DB is being used, only the H or S responses are valid here.

Default: Carriage return will cause the query to be repeated.

Starting Hour:

This query requests the beginning hour of the report period. It will be displayed only if "S", "H", "HOURS-WEEK", etc. has been specified in response to the "Granularity Level:" query.

Valid Response:

nn Where nn = 01 through 24.

Default: Carriage return will cause the query to be repeated.

Ending Hour:

This query requests the ending hour of the report period. It will be displayed only if "S", "H", "HOURS-WEEK", etc. has been specified in response to the "Granularity Level:"

query.

Valid Response:

nn Where nn = 01 through 24.

If data for only one hour is required, make the ending hour the same as the starting hour.

Default: Carriage return will cause the query to be repeated.

(P)rime (N)on Prime (B)oth or (W)eekend:

This query is used to further restrict the time period of the data to be displayed beyond that implied by the "Granularity Level:", "Starting Hour:", and "Ending Hour:" queries.

Valid Response:

P - Display only data marked as prime time.
N - Display only data marked as non-prime time.
B - Display both prime and non-prime data.
W - Display data only for composite weekend and holiday hours.

The "W" response will be accepted only if the composite weekend and holiday summary level has been selected under the "Granularity Level:" query.

"B" is allowed only if "H" or "S" has been specified in response to the "Granularity Level:" query.

The "B" response is not allowed if "Y" has been specified in response to the "Histogram Function:" query.

Default: Carriage return will cause the query to be repeated.

Item n:

This query is repeated up to 10 times (n = 1 to 10). Up to 10 items or subitems may be requested for display in one report.

Valid Response:

aaaa
aaaasssssss
?
aaaa?

Where aaaa is a 4 character item code and sssssss is a 7 character subitem code. See the Appendix called "System

AMAR Item Definitions" for a list of item codes. The AMRGEN program can also be used to obtain the list of items and subitems for your own database. See the Section called "Examining/Changing Database Parameters (AMRGEN)".

? causes the entire list of item and subitem names to be displayed in alphabetical order.

aaaa? causes the subitem names to be listed for the item denoted by aaaa.

If ? or aaaa? is specified, the query is repeated.

Default: Carriage return immediately terminates the list of items and subitems even if none has been specified.

Suppress Blank Ranges (Y/N)?

This query requests whether or not you wish to print ranges of values with a sample count of 0. It is displayed only if "Y" has been specified in response to the "Histogram Function:" query.

Valid Response:

Y - Yes, suppress ranges with a sample count of 0.
N - No, print ranges with a sample count of 0.

Warning: It is recommended that the "Y" response normally be used, especially for items which could occasionally have one or two very large values. Otherwise, a large amount of unnecessary data (all 0's) may be printed.

Default: Carriage return will cause the query to be repeated.

Headings on Report (Y/N)?

This query requests whether or not report headings should be produced. It is displayed only if "F" has been specified in response to the "Output at (T)erminal or in (F)ile:" query.

Valid Response:

Y - Yes, produce headings.
N - No, suppress production of headings.

If report headings are not produced, the user must develop another method of identifying to which items and subitems the reported values belong. No internal identification will be kept in the report.

Default: Carriage return will cause the query to be repeated.

More Requests (Y/N):

Requests whether or not the user wishes to extract more data.

Valid Response:

Y - Yes, repeat dialogue.
N - No, terminate program.

Default: Carriage return will cause the query to be repeated.

APPENDIX F

DATA EXTRACTION PROGRAM (AMAREX) DIALOGUE

AMAREX is normally run at the terminal. It can be used to access either the system AMAR database, AMAR.DB, or the output of the AMARSD program, TODAY.DB. See the Section called "Data Extraction Records" for a sample of the output and a description of the record formats.

In all queries except "DATABASE NAME:" and "OUTPUT:", carriage return may be entered as a response on subsequent passes through the dialogue. Carriage return means to re-use the last valid response given to that query.

Invalid responses to a query will cause the query to be repeated.

```
.RUN AMAREX
```

DATABASE NAME:

Requests the name of the database from which records are to be extracted.

Valid Response:

```
AMAR
AMAR.DB
TODAY
TODAY.DB
```

AMAR or AMAR.DB specifies the system AMAR database as the input file. TODAY or TODAY.DB specifies the mini-database created by the AMARSD program as the input file.

Default: None.

OUTPUT:

Requests the name of the output file which will contain the extracted records.

Valid Response:

filename
filename.ext

Care should be taken not to use any of the reserved names listed in the Appendix called "Summary of Programs and Filenames".

Default: None.

RECORD TYPE:

Requests the 2 character code which designates the type of records to be extracted.

Valid Response:

PD - Performance Detail Records
PS - Performance Summary Records
GR - Granularity Records
SC - System Uptime Log Records

Two or more record codes may also be strung together by commas.

DATE:

Requests the timeframe(s) for which you wish to extract records.

Valid Response:

yymmdd
yymmdd:hhss
yymmdd-yymmdd
yymmdd:hhss-yymmdd:hhss

Where:

yy is the actual calendar <u>not</u> fiscal year
mm is the month
dd is the day
hh is the hour
ss are the minutes

Two or more of the above date and time specifications may be strung together by commas.

All dates and times should be entered in chronological order.

Only data for a fiscal period which terminates within the specified timeframes will be extracted, regardless of the beginning date and time of the fiscal period. (See the

"GRANULARITY LEVEL:" query).

Hourly data is the lowest level of data which may be extracted. To select a specific hour, specify that hour as the start time and/or end time of the interval. For example, to extract data for the hour ending at 2 AM on February 4, 1983, specify 830204:0200.

All time periods, except for hours, are assumed to end at midnight. You only need to specify the end date of the fiscal period for which data is to be extracted.

Whenever time is not specified, the start time is assumed to be 0001 of the first day and the end time is assumed to be 2400 of the last day.

GRANULARITY LEVEL:

This query requests the level of fiscal period to be extracted. It is displayed only if "PD", "PS" or "GR" has been specified in response to the "RECORD TYPE:" query.

Valid Response:

SGI [or S] - Sample Group Interval (hours)
DAY [or D]
WEEK [WK or W]
MONTH [MO or M]
HOURS-WEEK [HRS-WEEK or H-W] - Composite Week Hours
HOURS-MONTH [HRS-MONTH or H-M] - Composite Month Hours

Two or more of the above responses may also be strung together by commas.

Composite refers to those records which are displayed in the Weekly/Monthly 'Typical Day' Reports.

RESTRICTING ANY FISCAL PERIOD?

This query asks whether or not you wish to include or exclude any fiscal periods from the timeframe specified in the "DATE:" query. It is displayed only if "PD", "PS" or "GR" has been specified in response to the "RECORD TYPE:" query.

Valid Response:

Y - Triggers further queries used to specify the fiscal periods to be selected.

N - No restrictions on fiscal periods.

The fiscal calendar is defined within the system AMAR database as follows:

Each hour of the day is defined as a Sample Group Interval (SGI) numbered 1 through 24.

Each day of the week is assigned a number from 1 to 7, Sunday through Saturday.

Each week in a fiscal month is assigned a number from 1 to 4 for the first 2 months in a fiscal quarter and from 1 to 5 for the third month in the quarter. In a fiscal leap year, the last month will have from 1 to 6 fiscal weeks.

Each month in a fiscal quarter is assigned a number from 1 to 3.

Each fiscal quarter is assigned a number from 1 to 4.

Refer to the Appendix called "Fiscal Calendar" for an example of how to relate the fiscal calendar to a normal calendar and to AMAR filenames.

SGI PERIOD:

This query requests the range of hours you want included or excluded from the timeframes specified in the "DATE:" query. It is displayed only if "Y" has been specified in response to the "RESTRICTING ANY FISCAL PERIOD?" query.

Valid Response:

hhss-hhss	Where hh = hours; ss = minutes.
hhss-hhss,...,hhss-hhss	
VOID	Void any previously specified SGI restrictions.

Default: Carriage return <CR> on the first pass through the dialogue causes the next query to be displayed.

DAY PERIOD:

This query requests the range of fiscal days you want included or excluded from the timeframes specified in the "DATE:" query. It is displayed only if "Y" has been specified in response to the "RESTRICTING ANY FISCAL PERIOD?" query.

Valid Response:

n	Where n = 1,2,... or 7.
1,...,7	
1-7	
VOID	Void any previously specified day restrictions.

Default: Carriage return <CR> on the first pass through the

dialogue causes the next query to be displayed.

WK PERIOD:

This query requests the range of fiscal weeks you want included or excluded from the timeframes specified in the "DATE:" query. It is displayed only if "Y" has been specified in response to the "RESTRICTING ANY FISCAL PERIOD?" query.

Valid Response:

n 1,...,5 1-5	Where n = 1,2,... or 5.
VOID	Void any previously specified week restrictions.

Default: Carriage return <CR> on the first pass through the dialogue causes the next query to be displayed.

MO PERIOD:

This query requests the range of fiscal months you want included or excluded from the timeframes specified in the "DATE:" query. It is displayed only if "Y" has been specified in response to the "RESTRICTING ANY FISCAL PERIOD?" query.

Valid Response:

n 1,2,3 1-3	Where n = 1, 2 , or 3.
VOID	Void any previously specified month restrictions.

Default: Carriage return <CR> on the first pass through the dialogue causes the next query to be displayed.

QTR PERIOD:

This query requests the range of fiscal quarters you want included or excluded from the timeframes specified in the "DATE:" query. It is displayed only if "Y" has been specified in response to the "RESTRICTING ANY FISCAL PERIOD?" query.

Valid Response:

n 1,...,4 1-4	Where n = 1, 2, ... or 4.
VOID	Void any previously specified quarter restriction.

Default: Carriage return <CR> on the first pass through the dialogue causes the next query to be displayed.

YR PERIOD:

This query requests the date(s) of the fiscal year(s) you want included or excluded from the timeframes specified in the "DATE:" query. Note that fiscal year granularity records are not normally kept in the database. Thus, you may not get any output when using this query. This query is displayed only if "Y" has been specified in response to the "RESTRICTING ANY FISCAL PERIOD?" query.

Valid Response:

nn	Where nn = 78, 79, ... , 99.
78,...,99	
78-99	
VOID	Void any previously specified year restriction.

Default: Carriage return <CR> on the first pass through the dialogue causes the next query to be displayed.

ITEM:

This query requests the 4 character item code and the 7 character subitem code (see the Appendix called "System AMAR Item Definitions") of any items or subitems you wish to extract. Use the AMRGEN program to get a list of all the items and subitems contained in your database. This query is displayed only if "PD" or "PS" has been specified in response to the "RECORD TYPE:" query.

Valid Response:

aaaa
aaaasssssss
aaaa,....,aaaasssssss
ALL

Where aaaa is a 4 character item code and sssssss is a 7 character subitem code. If the item has subitems and you specify only the 4 characters of the item code, all subitems will be extracted. Items and subitems will be extracted in alphabetical order.

PRIMETIME:

This query requests the code which identifies the type of data (prime, non-prime, weekend, or composite ('Typical Day')) that you want extracted. It is displayed only if "PD" or "PS" has been specified in response to the "RECORD TYPE:" query.

Valid Response:

P - Prime Time Data
N - Non-prime Time Data
N-P - Both Non-prime and Prime Data
W - Weekend and Holiday Hours for Composite Data
ALL - Prime, Non-prime, and Weekend and Holiday Data

The "N-P" and "W" responses will be accepted only if "HOURS-WEEK" or "HOURS-MONTH" were specified in response to the "GRANULARITY LEVEL:" query.

The "N-P" response will cause data to be included for weekdays and excluded for weekends and holidays.

The "W" response will cause data to be excluded for weekdays and included for weekends and holidays.

The recommended way of extracting weekday and weekend and holiday composite data is to run through the AMAREX dialogue twice, once with a response of "N-P" to extract weekday composite data and once with a response of "W" to extract weekend and holiday composite data.

Following the "PRIMETIME:" query, the message [EXTRACTING] will be displayed. This message indicates that the extraction process has begun.

Once processing is complete, and the output file closed, the message [SPECIFY NEXT EXTRACTION CRITERIA] will be displayed. The dialogue will be repeated starting with the "OUTPUT:" query. Further extraction requests may be entered.

Except for input and output file specifications, all selection criteria will remain in effect. To retain the selection criteria for a specific query, hit carriage return in response to that query. To override any previously supplied selection criteria, enter new explicit values. To cancel selection criteria for the "SGI PERIOD:" through "MO PERIOD:" queries, respond "VOID".

To enter exclusions, preceed the value by the phrase 'NOT'. For example, the response "ALL,'NOT'LUFS,'NOT'LUWQDSKR 0" to the "ITEM:" query causes data to be extracted for all items and subitems except LUFS (logical unit free space) and LUWQ (logical unit wait queue) for DSKR. As a guideline, specify the most inclusive response first, then exclusions. If ALL is used as a response, it must be the first in the string of responses to the query. Exclusion only works if a specific reply has aready been made to the query. For example, "'NOT'1" is a valid response to the "DAY PERIOD:" query only if "1,...,7" or "1-7" had already been specified in response to that query.

Blank Page

APPENDIX G

REPORT PROGRAM (W2RPT AND W2RPTB) DIALOGUE

This appendix explains, for each reporting program each possible prompt, its valid response, and any defaults.

Error messages which may occur while running W2RPTB or W2RPT are listed in AMAR-20 Error Messages. Some prompts and/or valid responses differ between W2RPTB and W2RPT. In these cases, W2RPTB will be discussed first and "W2RPTB only", or "W2RPT only" will be indicated in parentheses after the text of the prompt. The rest of this section is the description of each type of prompt in the W2RPTB and W2RPT dialogues. Prompts are listed in approximately the same order that they appear in the dialogue.

.RUN W2RPTB (or .RUN W2RPT)

REPORT DESCRIPTON =

Requests a free-form English description to be printed in the report header box.

Valid Response:

Arbitrary string of up to 90 characters, including blanks and punctuation.

Default: Carriage return says leave description blank.

INPUT FILE.EXT = (W2RPTB only).

Requests the name and extension of a workload database file which contains data for the period to be reported.

Valid Responses:

yyqmwd.DB0	for a weekday daily file
yyqmwd.DB1	for a weekend daily file
yyqmw.DB0	for a weekday weekly file

yyqmw.DB1 for a weekend weekly file

yyqm.DB0 for a weekday monthly file

yyqm.DB1 for a weekend monthly file

Where yy = fiscal year, q = fiscal quarter, m = fiscal month within the quarter, w = fiscal week within the month, d = day of the week (Sunday = 1).

Note: For purposes of automatic reporting, the filename (not the extension) may be implied by an appropriate number of question marks. Six question marks means a daily file, five questions marks means a weekly file, and four question marks means a monthly file. However, Sunday is normally the only day when weekly and monthly files are ready for automatic reporting. Therefore, ?????? is the only "wild name" useful for interactive dialogue with W2RPTB. Six question marks normally means yesterday, but could refer to a prior day if multiple days were input.

Note: Dialogue accepts filename and extension, not a complete file specification.

Default: None.

OUTPUT FILE.EXT = (W2RPTB only. See W2RPT below).

Requests the name and extension of the desired report file.

Valid Responses:

filename.ext

The filename (exclusive of extension) is arbitrary, but certain conventions are followed in the daily stream. (See the Section called "Procedure for Running W2RPTB.CTL".)

Any question mark in the name is replaced by the corresponding character of the input name.

Certain extensions such as .DB0, .DB1, .IN0, .IN1, .RA0, ,RA1, and .CON are not accepted by the program.

Extensions such as .RP1, .RP2, .RP3, and .RP4 are not recommended except in the daily stream.

Note: The dialogue accepts only filename and extension, not a complete file specification.

Default: None.

OUTPUT FILE.EXT = (W2RPT only. See W2RPTB above).

Requests the name and extension of the desired report file.

Valid Responses:

filename.ext

Before you run W2RPT, you should make sure you have a pair of daily files with the extensions of .IN0 and .IN1. These are expected as the input files for W2RPT. W2RPT will not prompt for an input filename; it will assume the existence of .IN0 and .IN1 files.

The output filename must be identical to the filename of your input files which have extensions .IN0 and .IN1. A filename of WC indicates the current workload files; WCX indicates yesterday's files. The extension should be other than .IN0, .IN1, .RAW, or .EXE.

Note: The dialogue accepts only filename and extension, not a complete file specification.

Default: None.

GROUPING FILE.EXT =

Grouping files are not implemented in AMAR-20.

Valid Responses:

Carriage return only.

Default: Carriage return means go to next query.

ENTER DESIRED START AS HH MM: (W2RPTB only. See W2RPT below).

Requests the time of the beginning of the first subreport interval.

Valid Responses:

Carriage return or 1 or 2 integers of the form:

hh mm

Where hh represents the hour, mm represents the minute.

If only one number is specified, it will be taken as the hour.

Since hourly data is the finest granularity in the database, only hours 0 through 23 are valid start times. The minutes, if specified, must be 0.

Default: Carriage return (or 0) says start at midnight.

ENTER DESIRED START AS HH MM SS DD: (W2RPT only. See W2RPTB above).

Requests the time (and relative day) of the beginning of the first reporting interval.

Valid Responses:

Carriage return or 1 to 4 integers of the form:

hh mm ss dd

Where hh represents the hour, mm represents the minutes, ss represents the seconds, and dd represents the number of the day at which you wish to begin the report.

Days are numbered 0, 1, 2, etc. starting with the first day of the file as day 0.

If only one number is specified, it will be taken as the hour; two numbers separated by a blank will be taken as the hour and minutes; and so forth.

For example, if there are three days worth of data in the file for September 8th, 9th, and 10th: 9 30 0 1 means start at 9:30 on the second day of the file (September 9th).

Any date and time before the end time of the input file may be specified. For purposes of synchronization, it is possible to specify a start time before the start time of the input file.

Note:

If you want to start at the beginning of the first day in the input file, respond 0 0 1 which implies 1 second after midnight and is distinct from 0, 0 0, 0 0 0, and 0 0 0 0 which are all equivalent to carriage return and imply start at start time of the file.

Default: Carriage return means start at the start time of the input file.

ENTER DESIRED END AS HH MM: (W2RPTB only. See W2RPT below).

Requests the end time of the last reporting interval.

Valid Responses:

Carriage return or 1 or 2 integers of the form:

hh mm

Where hh represents the hour, mm represents the minutes.

If only one number is specified, it will be taken as the hour.

Since hourly data is the finest granularity in the database, only hours 1 through 24 are valid end times. Another constraint is that end time must be greater than start time. The minutes, if specified, must be 0.

Default: Carriage return (or 0) says end at midnight.

ENTER DESIRED END AS HH MM SS DD: (W2RPT only. See W2RPTB above).

Requests the time (and relative day) of the end of the last reporting interval.

Valid Responses:

Same format as the start time for W2RPT.

Any time after the start time of the file and after the specified start time.

Default: Carriage return means stop the last report interval at the end time of the input file.

ENTER DESIRED INTERVAL SIZE AS HH MM: (W2RPTB only. See W2RPT below).

Requests the size of the subreport interval.

Valid Responses:

Carriage return or 1 or 2 integers of the form:

hh mm

Where hh represents hours, and mm represents minutes.

If only one number is specified, it will be taken as hours.

Since hourly data is the finest granularity in the database, only an integral number of hours between 1 and 24 is acceptable. Minutes, if specified, must be 0.

Note: If the requested interval size does not divide evenly into the time between start time and end time, the last subreport will be short.

Default: Carriage return (or 0) requests that the entire interval from start time to end time should be reported in a single subreport.

ENTER DESIRED INTERVAL SIZE AS HH MM SS DD: (W2RPT only. See W2RPTB above).

Requests the size of the reporting interval.

Valid Responses:

A time and date specification like W2RPT's start time. dd is the number of whole days in the interval.

The interval specified may be any positive interval not significantly less than the average checkpoint interval.

NOTE: When there is a gap in the data which is longer than the report interval (because of a system crash, for example), reports covering single checkpoint intervals will be generated while "catching up". This applies only to W2RPT, not to W2RPTB.

Default: Carriage return means report everything between the specified start time and the specified end time as a single report interval.

ENTER MAXIMUM DETAIL LINES PER INTERVAL:

Requests the maximum number of detail lines you wish printed for any report interval.

Valid Responses:

Carriage return or any integer between 1 and 512 (inclusive).

Default: Carriage return (or 0) means print all detail lines.

Note: You should sort your data by at least one significant resource if you use this type of cutoff; otherwise, you may suppress significant detail lines.

ENTER CPU% CUTOFF:

Requests the minimum percentage of CPU time which qualifies a detail line to be printed.

Valid Responses:

Carriage return or a number between .01 and 100.00.

Default: Carriage return (or 0) means print all detail lines.

Note: This form of cutoff exists only for CPU%. It is independent of sort order. It may even be used simultaneously with a "MAXIMUM DETAIL LINES" cutoff, although the results should be interpreted carefully.

EXTRA DETAILS? (Y OR N):

Allows you to specify whether or not you want a second line

of additional workload items printed on your report. Refer to the Appendix called "Workload AMAR Item Definitions" for a description of the items that would appear on this extra detail line.

Valid Responses:

Carriage return or "Y" or "N".

Default: Carriage return means do not print extra detail lines.

ANY SPECIAL MASKS OR SORT ORDERS? (Y or N):

Allows you to indicate whether or not you wish to use the special masking and/or sorting features described in the Appendix called "Special Masks and Sort Orders".

Valid Responses:

Carriage return or "Y" or "N".

Default: Carriage return (or anything that does not begin with "Y") means no special masks or sort orders.

ID ITEM 0-0:

Requests the first item to be held constant while building detail lines. These items form the "group" for which resource usage will be summarized. A very long list of grouping or sorting items will be truncated after 100 characters in the headings. The truncated portion could include "CUTOFF" criteria.

Valid Responses: Any of the grouping item codes listed in the Appendix called "Valid Grouping and/or Sort Items" (excluding items to be used only for sorting).

Default: None. ID ITEM 0-0: must be specified.

ID ITEM 0-1: (through ID ITEM 0-8:)

Requests the second (through ninth) item to be held constant while building detail lines.

Valid Responses: Carriage return or as above for ID ITEM 0-0:.

Default: Carriage return means no more items are to be held constant.

ID ITEM 0-9:

This query is used to terminate building of the list of grouping items.

Valid Responses: Carriage return only.

Default: No more items are to be held constant.

SORT ITEM 1-0:

Requests the major sort item for the first (usually only) subreport for a given interval.

Valid Response: Carriage return or any of the sort item codes described in the Appendix called "Valid Grouping and/or Sort Items".

Default: Carriage return means produce only one subreport for a given interval, with items sorted as they were grouped.

SORT ITEM 1-1: (through SORT ITEM 1-8:)

Requests a sub-sort item for the first (usually only) subreport for a given interval.

Valid Response: Carriage return or any of the sort item codes described in the Appendix called "Valid Grouping and/or Sort Items".

Default: Carriage return means no additional sub-sort items are to be specified.

SORT ITEM 1-9:

This query is used to terminate building the list of sort items for the first (usually only) subreport for a given interval.

Valid Response: Carriage return only.

Default: No more sort items are to be specified.

SORT ITEM 2-0: (through SORT ITEM 8-0:)

Requests the major sort item for the second (through eighth) subreport for a given interval).

Valid Response: Carriage return or any of the sort item codes described in the Appendix called "Valid Grouping and/or Sort Items".

Default: Carriage return means that no additional subreports for a given interval are to be produced.

SORT ITEM 2-1: (through SORT ITEM 2-8:)

Analagous to SORT ITEM 1-1: (through SORT ITEM 1-8:).

SORT ITEM 2-9:

Analagous to SORT ITEM 1-9:.

SORT ITEM 3-x: (through SORT ITEM 8-x:) (where x = 1 to 8)

Analagous to SORT ITEM 1-1: (through SORT ITEM 1-8:).

SORT ITEM 3-9: (through SORT ITEM 8-9:)

Analagous to SORT ITEM 1-9:.

SORT ITEM 9-0:

This query is used to terminate the building of subreports for each interval.

Valid Responses: Carriage return only.

Default: Carriage return means no additional subreports for a given interval are to be produced.

SORT ITEM 9-1: (through SORT ITEM 9-9:)

This query should not be reached; if it is encountered, you have specified too many lists of sort items and you will have to terminate the run and start over.

Valid Response: Control-C.

Default: If you enter a carriage return you will get a fatal error message and the run will terminate.

ID MASK 0-y: (where y = 0 to 8)

Allows you to specify a special mask for grouping as described in the Appendix called "Special Masks and Sort Orders".

Valid responses: Carriage return or a string of 1 to 12 octal digits.

Default: Carriage return (or an all zero octal mask) will cause the default mask to be used.

ID ORDER 0-y: (where y = 0 to 8)

Allows you to specify a special sort order as described in the Appendix called "Special Masks and Sort Orders".

Valid Response: Carriage return (for the default sort order). Any response starting with "A" (for ascending order). Any response starting with "D" (for descending order).

Default: Carriage return means the default order should be used.

SORT MASK x-y: (where x = 1 to 8, y = 0 to 8)

Allows you to specify a special mask as described in the Appendix called "Special Masks and Sort Orders".

Valid responses: Carriage return or a string of 1 to 12 octal digits.

Default: Carriage return (or an all zero octal mask) will cause the default mask to be used.

SORT ORDER x-y: (where x = 1 to 8, y = 0 to 8)

Allows you to specify a special sort order as described in the Appendix called "Special Masks and Sort Orders".

Valid Responses: Carriage return (for the default sort order). Any response starting with "A" (for ascending order). Any response starting with "D" (for descending order).

Default: Carriage return means the default order should be used.

MORE REPORTS? (Y OR N):

This query permits you to specify additional reports in the same run.

Valid Responses: Carriage return or "Y" or "N".

Default: Carriage return causes the program to terminate.

APPENDIX H

VALID GROUPING AND/OR SORT ITEMS

The following items (1) may be used for both grouping and sorting when using the W2RPT and W2RPTB programs. Any of these items used for sorting should also be used for grouping. Item codes must be spelled exactly as shown here; no other abbreviations are allowed.

MNE-MONIC	DEFLT. ORDER	DEFLT. MASK	DESCRIPTION
JOB	A	000037700000	Job number
FORK	A	000000077700	Fork Number (system wide)
USR1(2)	A	777777777777	User Name (first 5 chars.)
USR2(2)	A	777777777777	User Name (second 5 chars.)
USR3(2)	A	777777777777	User Name (third 5 chars.)
ACT1(2)	A	777777777777	Account Name (first 5 chars.)
ACT2(2)	A	777777777777	Account Name (second 5 chars.)
ACT3(2)	A	777777777777	Account Name (third 5 chars.)
PNAM	A	777777777777	Job program name (SIXBIT)
BATCH	A	000040000000	Batch Indicator
TTY	A	077700000000	Line Number
JLIT	A	777777777777	Job login date/time
JLDY	A	777777000000	Job login date
JLTM	A	000000777777	Job login time
FCREA	A	400000000000	Fork created
FTERM	A	200000000000	Fork terminated
TOP	A	100000000000	Top fork
FLIT	A	777777777777	Fork creation date/time
FLDY	A	777777000000	Fork creation date
FLTM	A	000000777777	Fork creation time

The following items should be used only for sorting:

JELA	D	777777777777	Job elapsed (average jobs)
FELA	D	777777777777	Fork elapsed (average forks)
MEMT	D	777777777777	Fork memory time
WSXT(3)	D	777777777777	Working set demand time integral

CPU%	D	777777777777	CPU percentage
NSRW	D	777777777777	Swap page faults per second
NFRW	D	777777777777	File page faults per second
ICNT	D	777777777777	Interactive responses per minute
IRSP(4)	D	777777777777	Interactive response time
ICPU(5)	D	777777777777	Interactive CPU time
NJLI(6)	D	777777777777	Logins/Logouts
NFLI(7)	D	777777777777	Fork creates/terminates
NWSL	D	777777777777	Working set loads
USED	D	777777777777	Used time
GRDY	D	777777777777	Golist ready time
BRDY	D	777777777777	Balance set ready time
SWPR	D	777777777777	Swap wait time
FILR	D	777777777777	Read wait time
FILW	D	777777777777	Write wait time
RPQW	D	777777777777	Replaceable queue wait time
OTHR	D	777777777777	Other misc. wait time

Notes:

1. Items (except for WSXT, UPXT, IRSP, and ICPU) are listed as they appear from left to right across the report.

2. These items are stored in ASCII.

WSXT, UPXT, IRSP, and ICPU, defined below, can be used for sorting although they cannot be displayed themselves on any Workload AMAR reports.

3. Working set demand time integral (WSXT) is not an item which can be printed on the reports. It can, however, be used for sorting. Working set size (PAGES (WS) on the reports) is derived from WSXT by dividing WSXT by the product of demand fraction (DEMD on the reports) and the measured interval time.

4. Interactive response time (IRSP) is the product of response rate (RSP/MIN on the reports), seconds per response (SEC/RSP), and the measured interval time. It is the total real time required to provide the responses. It is the numerator of the stretch ratio (SR). See the Appendix called "Workload AMAR Item Definitions" for a description of the components of IRSP.

5. Interactive CPU time is the total CPU time required to provide the responses. It is the denominator of the stretch ratio (SR).

6. The left half of NJLI represents logins; the right half represents logouts.

7. The left half of NFLI represents fork creates; the right half represents fork terminates.

APPENDIX I

SPECIAL MASKS AND SORT ORDERS

In certain cases, it may be desirable to group items by a part of one of the standard items, for example, the first four digits of the account name or the last two characters of the program name. This is possible when using the W2RPT or W2RPTB programs if you answer "Y" to the prompt

ANY SPECIAL MASKS OR SORT ORDERS? (Y OR N):

and answer the special prompts appropriately.

Selection of a part of a standard item is done by means of a mask, which is a computer word with "1" bits corresponding to the bits of the item you want to use and "0" bits corresponding to the bits you want to ignore. We represent a mask as 12 octal digits, each representing a group of 3 bits.

For PNAM (job program name) which is stored in SIXBIT, each character you want to use is represented by a pair of 7's in the mask and each character you want to ignore is represented by a pair of 0's. Hence, the mask for the first three characters of PNAM is 777777000000. The mask for the last 2 characters of the PNAM is 000000007777, which may be abbreviated as 7777, since leading zeroes are implied.

For items stored in ASCII (USR1, USR2, USR3, ACT1, ACT2, and ACT3) the mask for the first character is 774000000000; for the second character 003760000000; for the first two characters 777760000000, etc. Each word is composed of 7-bit bytes.

Seven of the items (JOB, FORK, BATCH, TTY, FCREA, FTERM, and TOP) are all packed together in a single word. You can see how they fit together if you examine their default masks in the Appendix called "Valid Grouping and/or Sort Items". If you want to use a part of one of these, the mask you use should have a subset of the "1" bits implied in the default mask. In other words, the mask you enter is used for the whole word, not just the item you named.

Figure I-1 shows a dialogue using a special mask and the

beginning of the report it generated.

The Appendix called "Valid Grouping and/or Sort Items" shows that each item has a default sort order associated with it (A = ascending, D = descending). If you wish to specify the opposite order for some item, answer "Y" to: "ANY SPECIAL MASKS OR SORT ORDERS? (Y OR N):".

```
:RU AM:W2RPTB

REPORT DESCRIPTION = REPORT MASKING FIRST 3 CHARACTERS (1)

INPUT  FILE.EXT = 832143.DBO (2)

OUTPUT FILE.EXT = ACC.RPT (3)
GROUPING FILE.EXT =

ENTER DESIRED START AS     HH MM: 8 (4)
ENTER DESIRED END   AS     HH MM: 16 (5)
ENTER DESIRED INTERVAL SIZE AS  HH MM: (6)

ENTER MAXIMUM DETAIL LINES PER INTERVAL: (7)
ENTER CPU% CUTOFF:

EXTRA DETAILS? (Y OR N): (8)

ANY SPECIAL MASKS OR SORT ORDERS? (Y OR N): Y (9)

ID ITEM    0-0: ACT1 (10)
ID MASK    0-0: 777777700000 (11)
ID ORDER   0-0:

ID ITEM    0-1:

SORT ITEM  1-0: (12)

MORE REPORTS? (Y OR N): N
```

Dialogue Using a Special Mask
Figure I-1

1. Free form report description. This report will summarize usage by account name, grouping the name by its first 3 characters.

2. Daily file for October 26, 1982 (FY83, second quarter, first month, fourth week, third day.)

3. Report filename.

4. Start the report at 8:00 AM. Minutes defaults to 0 if only the hour is specified. If carriage return only had been entered, the report would have started at the beginning of the file (midnight).

5. The report will stop at 16:00 (4:00 PM). If carriage return only had been entered, the report would have stopped at the end of the file (midnight).

6. Since carriage return was entered, the report will cover the entire period between 8:00 AM and 4:00 PM.

7. No cutoffs will be used.

8. No extra detail lines will be printed.

9. The special mask feature will be used.

10. The data will be grouped by the first word in account name (ACT1) since we are interested only in the first 3 characters of the name.

11. To group by the first 3 characters we must mask out all other unwanted characters. For PNAM, a pair of 0's means ignore the corresponding character, a pair of 7's means use the character for grouping. However, for ASCII items such as account name, you must compute your own mask as follows. Write out the ASCII word (5 7-bit bytes) using a 1 for characters you want and a 0 for those to be ignored. Then convert to octal and use those numbers for your mask. For example, the 3 character mask for ACT1 is computed as follows:

```
111,111,1 11,111,11 1,111,111 000,000,0 00,000,00
 7   7   7    7   7    7   7   0   0   0    0   0
```

12. The detail lines will be sorted by account name since no other sort order is specified.

AMAR WORKLOAD REPORT

SITE: TOPS-20 SAMPLE SYSTEM SYSTEM: TTSS

REPORT DESCRIPTION: REPORT MASKING FIRST 3 CHARACTERS OF ACCT

INPUT FILE: 832143.DBO (FISCAL YEAR: 83 QUARTER: 2 MONTH: 1 WEEK: 4 DAY: 3 TUESDAY)

FROM: 8: 4:25 ON TUESDAY 26-OCT-82 TO: 16: 0:48 ON TUESDAY 26-OCT-82 INTERVAL: 7:56:22 MEASURED: 100%

GROUPED BY: ACT1 777777700000

SORTED BY: ACT1 777777700000

JOB #	FRK #	AVG JOBS	AVG FRKS	IN MEM	DEMD	USER NAME	ACCOUNT NAME	PRGRM NAME	PAGES (WS)	CPU%	SWAP PF/S	FILE PF/S	IFA	RSP /MIN	SEC /RSP	SR	B	TTY	LOGIN AT DAY TIME
		69.8	240.8	163.6	6.18	**********INTERVAL TOTALS**********			42.8	70.52	2.3	10.8	62	660.8	0.2	4			
		0.5	1.4	0.9	0.03	AT.INQUIRY	COIN	??????	61.5	0.52	0.0	0.1	52	11.7	0.2	4	T		0
0		1.0	20.0	8.7	0.10	OPERATOR	OPERATOR	SYSJOB	32.4	1.30	0.2	0.1	46	58.1	0.1	4	T	DET	-4 5:45
		15.4	59.8	18.7	1.10	??????????		??????	83.7	15.72	0.3	0.6	184	53.6	0.1	4			
		1.8	3.6	2.6	0.30	????????	155.70000000???	??????	34.9	1.95	0.0	0.2	101	12.5	0.1	4	T		
		0.2	0.3	0.3	0.05	GLASSCOCK	242.	??????	101.8	0.39	0.0	0.1	56	1.6	0.4	4	T		0
		0.0	0.0	0.0	0.01	STANFORD	566.A0000000	??????	153.8	0.09	0.0	0.0	29	0.4	0.2	3			0
		0.4	0.8	0.7	0.07	GH.???	621.G0000000	??????	73.1	0.39	0.0	0.1	59	1.5	0.4	6	T		0
		0.7	1.3	1.3	0.14	IRWIN	639.830809.2000	??????	28.7	2.29	0.0	0.0	487	74.1	0.1	4	T		0
		1.3	2.5	2.3	0.08	IP.??????	647.82037?	??????	48.5	0.98	0.0	0.1	66	8.3	0.2	4	T		0
		0.3	0.7	0.7	0.01	IP????????	665.82037?	??????	37.4	0.19	0.0	0.1	24	1.0	0.2	3	T		0
		2.0	4.0	1.6	0.01	ETAMAR.DCOL	669.A0000000	??????	21.3	0.17	0.1	0.0	15	2.3	0.4	6	T		-4
		38.6	124.8	101.0	3.91	AT.??????????	670.A00000??	??????	30.3	39.42	1.1	8.1	49	353.5	0.2	5			
		0.2	0.3	0.3	0.01	FS.USER	672.A0000000	??????	72.7	0.46	0.0	0.0	131	1.2	0.2	2	T		0
		6.9	19.8	17.2	0.15	PZ.??????????	674.831325	??????	54.8	2.98	0.2	0.7	45	23.2	0.4	3	T		0
63		0.6	1.4	1.3	0.03	BFS.MACINNES	676.A0000000	??????	52.3	0.75	0.0	0.1	104	24.2	0.0	3	T	46	0 11:30

Figure I-1 (continued)

APPENDIX J

SYSTEM AMAR BATCH STREAM - AMAR.CTL

Reference Figure J-1 for a listing of an unedited AMAR.CTL stream.

ST000: Attempts to have the pack mounted which contains the AMAR programs and database. The request is kept pending until the operator responds. If the pack cannot be mounted, control is transferred to ST180.

ST008: Checks to see if the operator had, at some point, set an incorrect date and time on the machine and if AMAR data files were created with a time stamp in the future. If the current machine date is correct, you should delete such files.

ST009: Checks to see if the operator had set an incorrect date and time on the machine and the AMAR database had been updated with the incorrect date/time. If the current machine date is correct, you should delete the database and restore from a good disk or tape backup. Control is transferred to ST180.

ST010: Deletes obsolete files from the primary production pack.

ST020: On some systems, the AMAR daily raw files are kept on a separate pack from the database (production) pack. This step copies the AMAR raw files to the production pack. If there is no separate pack, this step is bypassed and control is passed to ST040.

ST030: Deletes obsolete AMAR daily raw files from the data collection pack.

ST040: Checks to see if the AMAR database is nonexistent or corrupted. If it is, control is passed to ST050. If not, control is passed to ST080. Corruption can occur if the system crashes while either the AMARIP or AMARUP program is running.

ST050: Checks to see if a backup AMAR database exists on disk. If not, control is transferred to BD070. Otherwise, it deletes the file which controls tape backup frequency. It then deletes the corrupted database and copies the backup to the production pack. If the backup attempt fails, control is transferred to ST060. Otherwise control is transferred to ST070.

ST060: Attempts to have the backup pack mounted on a different drive and the AMAR database restored to the production pack. If ST060 fails, control is passed to BD070.

ST070: Checks to see if the restored database is corrupted. If it is, control is passed to BD070. Otherwise, control is passed to ST100.

BD070: Asks the operator to restore the AMAR database from tape. This step is reached only if no good copies of the database exist on disk. Transfers control to ST190 which stops the stream. The operator should restore AMAR.DB from tape before restarting.

ST080: If the original database is good, this step creates a backup of it and protects it to prevent it from being accidentally deleted. If there is no space available for the backup, control is passed to ST180. If parity errors have occurred, control goes to ST090. Otherwise control is passed to ST100.

ST090: In the event of parity errors during database backup, the operator is requested to mount the pack on a different drive and try the backup again. If that fails, it is assumed the database is bad. The operator is requested to restore a good copy from tape before resubmitting the stream. Control is passed to ST190 which stops the stream. Otherwise control is passed to ST100.

ST100: Inputs raw data files into the system AMAR database.

ST110: Updates the system AMAR database, creating the proper summary records and deleting obsolete data.

ST120: Generates the automatic AMAR reports.

ST130: Prints the AMAR reports. If the site has decided not to use tape backup in this stream, control passes directly to ST170. Otherwise control passes to ST140.

ST140: Checks the tape backup frequency control file to see if tape backup should occur. If not, control is passed to ST180.

ST150: Creates a tape backup of the system AMAR database. If the mount fails control is passed to ST180. If there is an error during backup, control is passed to BD150 which

requests a clean tape and drive. The operator is requested to cancel the mount after three tries.

ST160: Creates a new tape frequency control file.

ST170: Deletes obsolete raw files from the production pack.

ST180: Resubmits the stream for processing after 1:00 AM the next night.

ST190: Stops the stream. Creates a record of any "bad" logs - AMAR.LG - for later examination.

```
:AMAR.CTL CAN RESTART ONLY AT RESUBMIT STEP ST180

ST000:: !******** MOUNT THE PRODUCTION PACK ********!

@MOUNT STRUCTURE <DB-STRUC>:
@IF (ERROR) @GOTO ST180
@CONNECT <DB-STRUC>:<<AMAR-DIR>>
@IF (ERROR) @GOTO ST180

ST008:: !******** CHECK FOR INVALID FUTURE RAW FILES ********!

@ERROR %
@RU EOF
*-1,0
*<DC-STRUC>:<<AMAR-DIR>><SYS-ID>0%.*,<DC-STRUC>:<<AMAR-DIR>><SYS-ID>1%.*,<DC-STRUC>:<<AMAR-DIR>><SYS-ID>2%.*,<DC-STRUC>:<<AMAR-DIR>>
<SYS-ID>3%.*
@IF (NOERROR) @GOTO ST009

BD008:: !!!!!!!!! FOUND FUTURE RAW FILE(S) !!!!!!!!
!DELETE ABOVE RAW FILE(S) THAT WERE CREATED FOR FUTURE DATE(S)!

ST009:: !******** CHECK FOR INVALID FUTURE DATABASE ********!

@ERROR %
@RU EOF
*-1,0
*<DB-STRUC>:<<AMAR-DIR>>AMAR.DB
@IF (NOERROR) @GOTO ST010

BD009:: !!!!!!!!! FOUND FUTURE DATABASE !!!!!!!!!
!IF TODAY'S DATE IS INCORRECTLY SET TO A PAST DATE DO NOTHING!
!IF TODAY'S DATE IS CORRECT AND LAST STREAM WAS RUN ON FUTURE DATE!
!   THEN RESTORE AMAR DATABASE AND RAW FILES FROM DISK OR TAPE BACKUP!
@GOTO ST180

ST010:: !******** DELETE OBSOLETE FILES FROM PRODUCTION PACK ********!

@INFO DISK
@NOERROR
@DEL AMAR%%.RPT,AMARUP.IF%

ST020:: !******** COPY RAW FILES TO PRODUCTION PACK ********!

@ERROR %
@OPERATOR $
@COPY <DC-STRUC>:<<AMAR-DIR>><SYS-ID>%%.* <DB-STRUC>:<<AMAR-DIR>>*.*
@NOOPERATOR
@IF (ERROR) @GOTO ST040

ST030:: !******** DELETE RAW FILES FROM DATA COLLECTION PACK ********!

@NOERROR
@RU EOF
*<DC-RTEN>,0
*<DC-STRUC>:<<AMAR-DIR>><SYS-ID>0%.*,<DC-STRUC>:<<AMAR-DIR>><SYS-ID>1%.*,<DC-STRUC>:<<AMAR-DIR>><SYS-ID>2%.*,<DC-STRUC>:<<AMAR-DIR>>
<SYS-ID>3%.*

ST040:: !******** CHECK IF AMAR DATABASE IS CORRUPTED ********!

@INFO DISK
@ERROR %
```

Figure J-1

```
@DIR <DB-STRUC>:<<AMAR-DIR>>AMAR.DB
@IF (ERROR) @GOTO STO5O
@RU AMRGEN
*E D
@IF (ERROR) @GOTO STO5O
@GOTO STO8O

STO5O:: !******** SUBSTITUTE BACKUP FOR CORRUPTED DATABASE ********!

@ERROR %
@DIR <BKUP-STRUC>:<<AMAR-DIR>>AMAR.DBK
@IF (ERROR) @GOTO BDO7O
@NOERROR
@DEL AMAR.TAP
@DEL <DB-STRUC>:<<AMAR-DIR>>AMAR.DB
@EXPUNGE
@ERROR %
@OPERATOR $
@COPY <BKUP-STRUC>:<<AMAR-DIR>>AMAR.DBK <DB-STRUC>:<<AMAR-DIR>>AMAR.DB
@NOOPERATOR
@IF (ERROR) @GOTO STO6O
@DIR <DB-STRUC>:<<AMAR-DIR>>AMAR.DB
@IF (ERROR) @GOTO BDO5O
@GOTO STO7O

BDO5O:: !!!!!!!!! INSUFFICIENT DISK SPACE TO RESTORE AMAR.DB DATABASE !!!!!!!!!
!DELETE UNNECESSARY FILES ON <DB-STRUC>!
@GOTO ST18O

STO6O:: !******** IF PROBLEM RESTORE BACKUP FROM DIFFERENT DRIVE ********!

@NOERROR
@DISMOUNT STRUCTURE <BKUP-STRUC>:/REMOVE
@PLEASE MOUNT <BKUP-STRUC> ON A DIFFERENT DRIVE SINCE READ ERRORS ON CURRENT DRIVE^[
@MOUNT STRUCTURE <BKUP-STRUC>:
@DEL <DB-STRUC>:<<AMAR-DIR>>AMAR.DB
@EXPUNGE
@ERROR %
@OPERATOR $
@COPY <BKUP-STRUC>:<<AMAR-DIR>>AMAR.DBK <DB-STRUC>:<<AMAR-DIR>>AMAR.DB
@NOOPERATOR
@IF (ERROR) @GOTO BDO7O

STO7O:: !******** CHECK IF BACKUP DATABASE IS CORRUPTED ********!

@ERROR %
@DIR <DB-STRUC>:<<AMAR-DIR>>AMAR.DB
@IF (ERROR) @GOTO BDO7O
@RU AMRGEN
*E D
@IF (ERROR) @GOTO BDO7O
@GOTO ST1OO

BDO7O:: !!!!!!!!! ERROR RESTORING BACKUP AMAR.DB DATABASE !!!!!!!!!
!RESTORE AMAR.DB DATABASE FROM BACKUP TAPE THEN SUBMIT STREAM!
@GOTO ST19O

STO8O:: !******** CREATE A DISK BACKUP OF AMAR DATABASE ********!

@NOERROR
@REN <BKUP-STRUC>:<<AMAR-DIR>>AMAR.DBK <BKUP-STRUC>:<<AMAR-DIR>>AMAR.DBK;P777700
```

Figure J-1 (continued)

```
@DEL <BKUP-STRUC>:<<AMAR-DIR>>AMAR.DBK
@EXPUNGE
@ERROR %
@OPERATOR $
@COPY <DB-STRUC>:<<AMAR-DIR>>AMAR.DB <BKUP-STRUC>:<<AMAR-DIR>>AMAR.DBK;P777700
@NOOPERATOR
@IF (ERROR) @GOTO ST090
@ERROR %
@DIR <BKUP-STRUC>:<<AMAR-DIR>>AMAR.DBK
@IF (ERROR) @GOTO BD080
@GOTO ST100

BD080:: !!!!!!!!! INSUFFICIENT DISK SPACE TO CREATE BACKUP AMAR.DB DATABASE !!!!!!!!!
!DELETE UNNECESSARY FILES ON <BKUP-STRUC>!
@GOTO ST180

ST090:: !******** IF PROBLEM CREATE BACKUP ON DIFFERENT DRIVE ********!

@NOERROR
@CONNECT
@DISMOUNT STRUCTURE <DB-STRUC>:/REMOVE
@PLEASE MOUNT <DB-STRUC> ON A DIFFERENT DRIVE SINCE READ ERRORS ON CURRENT DRIVE^[
@MOUNT STRUCTURE <DB-STRUC>:
@NOERROR
@CONNECT <DB-STRUC>:<<AMAR-DIR>>
@REN <BKUP-STRUC>:<<AMAR-DIR>>AMAR.DBK <BKUP-STRUC>:<<AMAR-DIR>>AMAR.DBK;P777700
@DEL <BKUP-STRUC>:<<AMAR-DIR>>AMAR.DBK
@EXPUNGE
@ERROR %
@OPERATOR $
@COPY <DB-STRUC>:<<AMAR-DIR>>AMAR.DB <BKUP-STRUC>:<<AMAR-DIR>>AMAR.DBK;P777700
@NOOPERATOR
@IF (ERROR) @GOTO ST090
@ERROR %
@DIR <BKUP-STRUC>:<<AMAR-DIR>>AMAR.DBK
@IF (ERROR) @GOTO BD090
@GOTO ST100

BD090:: !!!!!!!!! ERROR CREATING BACKUP AMAR.DB DATABASE !!!!!!!!!
!RESTORE AMAR.DB DATABASE FROM BACKUP TAPE IF I/O ERROR THEN SUBMIT STREAM!
@GOTO ST190

ST100:: !******** STORE RAW FILE DATA IN THE AMAR DATABASE ********!

@NOERROR
@MOUNT STRUCTURE <DB-STRUC>:
@INFO PROGRAM-STATUS
@ERROR
@OPERATOR $
@RU AMARIP
*ANYDAY
@NOOPERATOR
@IF (ERROR) @GOTO BD100
@GOTO ST110

BD100:: !!!!!!!!! ERROR DURING AMARIP PROGRAM !!!!!!!!!
!CORRECT PROBLEM THEN SUBMIT STREAM!
@GOTO ST190

ST110:: !******** ROLLUP DATA IN THE AMAR DATABASE ********!
```

Figure J-1 (continued)

```
@INFO PROGRAM-STATUS
@OPERATOR $
@RU AMARUP
@NOOPERATOR
@IF (ERROR) @GOTO BD110
@GOTO ST120

BD110:: !!!!!!!!! ERROR DURING AMARUP PROGRAM !!!!!!!!!
!CORRECT PROBLEM THEN SUBMIT STREAM!
@GOTO ST190

ST120:: !******** GENERATE AUTOMATIC AMAR REPORTS ********!

@INFO PROGRAM-STATUS
@ERROR
@OPERATOR $
@RU AMREPT
;DAILY UTILIZATION REPORT
*DU
*AMAR
*AUTO
*AMARDR.RPT
;WEEKLY UTILIZATION REPORT
*WU
*AMAR
*AUTO
*AMARWU.RPT
;MONTHLY UTILIZATION REPORT
*MU
*AMAR
*AUTO
*AMARMU.RPT
;WEEKLY TREND REPORT
*WA
*AMAR
*AUTO
*AMARWA.RPT
;MONTHLY TREND REPORT
*MA
*AMAR
*AUTO
*AMARMA.RPT
;WEEKLY COMPOSITE UTILIZATION REPORT
;*WC
;*AMAR
;*AUTO
;*AMARWC.RPT
;MONTHLY COMPOSITE UTILIZATION REPORT
;*MC
;*AMAR
;*AUTO
;*AMARMC.RPT
;DAILY DISK REPORT
;*DD
;*AMAR
;*AUTO
;*AMARDD.RPT
;WEEKLY DISK REPORT
*WD
*AMAR
*AUTO
```

Figure J-1 (continued)

```
*AMARWD.RPT
;MONTHLY DISK REPORT
*MD
*AMAR
*AUTO
*AMARMD.RPT
;DAILY TAPE REPORT
;*DT
;*AMAR
;*AUTO
;*AMARDT.RPT
;WEEKLY TAPE REPORT
*WT
*AMAR
*AUTO
*AMARWT.RPT
;MONTHLY TAPE REPORT
*MT
*AMAR
*AUTO
*AMARMT.RPT
*EXIT
@NOOPERATOR
@IF (ERROR) @GOTO BD120
@GOTO ST130

BD120:: !!!!!!!!! ERROR DURING AMREPT PROGRAM !!!!!!!!!
!ENSURE ADEQUATE DISK SPACE FOR REPORT FILES!

ST130:: !******** PRINT/XEROX AUTOMATIC AMAR REPORTS ********!

@INFO PROGRAM-STATUS
@APPEND AMAR%%.RPT AMAR.RPT
@NOERROR
@PRINT AMAR.RPT/DELETE/NOTE:"<NOTE>"
;@XEROX AMAR.RPT
@GOTO ST170

ST140:: !******** CHECK IF DAY TO CREATE BACKUP TAPE OF AMAR DATABASE ********!

@NOERROR
@RU EOF
*<TAPE-PRD>,0
*AMAR.TAP
@ERROR %
@DIR AMAR.TAP
@IF (ERROR) @GOTO ST150
@GOTO ST180

ST150:: !******** CREATE A BACKUP TAPE OF AMAR DATABASE ********!

@ERROR %
@PLEASE SHOW TAPE NUMBER IN LOG BY RESPONDING^[
@MOUNT TAPE MTAA:/SCRATCH/REMARK:"<TAPE-ID>"
@IF (ERROR) @GOTO BD151
@ERROR %
@OPERATOR $
@R DUMPER
*TAPE MTAA:
*FILES
*SSNAME AMAR
```

Figure J-1 (continued)

```
*SAVE <DB-STRUC>:<<AMAR-DIR>>AMAR.DB,<DB-STRUC>:<<AMAR-DIR>><SYS-ID>*.*
*EXIT
@NOOPERATOR
@IF (ERROR) @GOTO BD150
@NOERROR
@DISMOUNT TAPE MTAA:
@GOTO ST160

BD150:: !!!!!!!!! ERROR CREATING BACKUP TAPE !!!!!!!!!
!PROVIDE CLEAN ERROR FREE DRIVE AND NEW SCRATCH TAPES!
@NOERROR
@DISMOUNT TAPE MTAA:
@PLEASE SCRATCH TAPE(S) ALREADY CREATED AND LABELLED^[
@PLEASE CLEAN TAPE DRIVE AND MOUNT NEW SCRATCH SINCE BACKUP ERROR^[
@PLEASE IF THIS REMOUNT REQUEST REPEATS 3 OR MORE TIMES CANCEL ENSUING MOUNT^[
@PLEASE THEN THE STREAM WILL SKIP THE BACKUP TAPE STEP ASSUMING I/O ERROR^[
@BACKTO ST150

BD151:: !!!!!!!!! ERROR READING BACKUP INPUT FILES !!!!!!!!!
!IF BACKUP INPUT FILE HAS I/O ERROR RESTORE FILE FROM BACKUP DISK COPY!
!SKIPPING CREATION OF BACKUP TAPE!
@GOTO ST180

ST160:: !******** RECORD DATE OF BACKUP TAPE CREATION ********!

@NOERROR
@DEL <DB-STRUC>:<<AMAR-DIR>>AMAR.TAP
@COPY TTY: <DB-STRUC>:<<AMAR-DIR>>AMAR.TAP.1
@^Z

ST170:: !******** DELETE RAW FILES FROM PRODUCTION PACK ********!

@NOERROR
@RU EOF
*<DB-RTEN>,1
*<DB-STRUC>:<SYS-ID>0%.*,<DB-STRUC>:<SYS-ID>1%.*,<DB-STRUC>:<SYS-ID>2%.*,<DB-STRUC>:<SYS-ID>3%.*

ST180:: !******** RESUBMIT THE AMAR BATCH STREAM ********!

@CHKPNT ST180
@NOERROR
@DEL AMAR.LG
@SUB <DC-STRUC>:AMAR.CTL/LOGDISPOSITION:KEEP/BATCH-LOG:SUPERSEDE/LOGNAME:<DB-STRUC>:AMAR.LOG/UNIQUE:YES/RESTARTABLE:YES/TIME:1:0:0/A
FTER:TODAY+1:00
@GOTO FIN


%TERR::
@GOTO BD180
%CERR::
@GOTO BD180
%ERR::
@GOTO BD180

BD180:: !!!!!!!!! TIME LIMIT EXHAUSTED OR MONITOR OR PROGRAM ERROR !!!!!!!!!
!!!!!!!!! STREAM IS RESUBMITTED IF THE DATABASE IS OK !!!!!!!!!
!!!!!!!!! CORRECT PROBLEM IF IT WILL RECUR !!!!!!!!!

@ERROR
```

Figure J-1 (continued)

```
@RU AMRGEN
*E D
@IF (NOERROR) @BACKTO ST180

ST190:: !******** STOP THE AMAR BATCH STREAM ********!

@NOERROR
@COPY <DC-STRUC>:AMAR.LOG <DB-STRUC>:AMAR.LG
@DEL AMAR.TAP

FIN::
%FIN::
```

Figure J-1 (continued)

APPENDIX K

WORKLOAD AMAR BATCH STREAM - W2RPTB.CTL

Reference Figure K-1 for a listing of an unedited W2RPTB.CTL stream.

DOUPD: This step runs W2UPD which performs database management including input, update, and deletion. First W2UPD reads the appropriate pair of incremental files (.IN0 and .IN1) and creates a database daily file. The name of the created file is yesterday's fiscal date. The extension is .DB0 if yesterday was a normal workday. The extension is .DB1 if yesterday was a Saturday, Sunday, or holiday. Next the new database daily file is rolled up into the appropriate weekly database file. At the end of the fiscal week, the weekly files are rolled up into the appropriate monthly files. After a database daily file is created, its name is entered in a list of files ready for automatic reporting. After a weekly or monthly file has been completed (the last day or week has been included), its name is entered in the list of files ready for automatic reporting. When input and rollup are complete through yesterday, W2UPD deletes the oldest incremental and database files if more than the user-specified number exist.

DORPT: This step produces the automatic reports. It is divided into six substeps, corresponding to the six types of file which may be ready for automatic reporting:

STEPNAME	REPORT PERIOD	TYPE OF DAY REPORTED ON	INPUT FILE SPEC.	REPORT FILENAME
DODY0	Day	Weekday	??????.DB0	WCDY0.ext
DOWK0	Week	Weekday	?????.DB0	WCWK0.ext
DODY1	Day	Weekend	??????.DB1	WCDY1.ext
DOWK1	Week	Weekend	?????.DB1	WCWK1.ext
DOMN1	Month	Weekend	????.DB1	WCMN1.ext
DOMN0	Month	Weekday	????.DB0	WCMN0.ext

Substeps are skipped if the input file for the type of day and report period has not yet been created. For example, even though the stream is run daily, monthly

reports will be produced only once at the end of each fiscal month - not every day. Also question marks must be used in the input file specification. This notifies W2RPTB that automatic reporting is taking place.

DOPRT: This step prints the reports. All workload reports created on a given day are normally concatenated into a single file for efficient printing. Any monthly or weekly files are put at the beginning of this large file so that you won't miss them. The individual report files are left on disk until overwritten by other files of the same name and extension. This leaves the most recent copy of each report on disk.

DOSUB: This step is almost always executed to submit tomorrow's run, even if earlier steps failed.

```
DOUPD:: .CHKPNT DOUPD
@ENABLE
@DEFINE DSK: <DB-STRUC>:<<AMAR-DIR>>,<DC-STRUC>:<<AMAR-DIR>>
@EXPUNGE <DC-STRUC>:<<AMAR-DIR>>
@EXPUNGE
@COPY WCDBS.CON WCDBS.COX
@RUN W2UPD
@IF (ERROR) @GOTO DOSUB
DORPT:: .CHKPNT DORPT
@DISABLE
@DEFINE DSK: <DB-STRUC>:<<AMAR-DIR>>
;HERE TO DO ANY REPORTS ON WEEKDAY DAY FILES
DODYO::
@RUN W2RPTB
*HOURLY REPORT BY PROGRAM AND USER
*??????.DBO
*WCDYO.PR1
*
*0
*24
*1
*
*1
*N
*N
*USR1
*USR2
*USR3
*PNAM
*
*CPU%
*
*
*N
;ONLY LIKELY ERROR IS NO READY WEEKDAY DAY FILE...
;...WHICH MEANS NO WEEKDAY WEEK FILE WILL BE READY
@IF (ERROR) @GOTO DODY1
@RUN W2RPTB
*SHIFT REPORT BY PROGRAM
*??????.DBO
*WCDYO.PR2
*
*0
*24
*8
*
*.5
*N
*N
*PNAM
*
*CPU%
*
*
*N
@IF (ERROR)
@RUN W2RPTB
*SHIFT REPORT BY BATCH VS. TIMESHARING
*??????.DBO
*WCDYO.PR3
*
```

Figure K-1

```
*O
*24
*8
*
*
*N
*N
*BATCH
*
*CPU%
*
*
*N
@IF (ERROR)
@RUN W2RPTB
*SHIFT REPORT BY USER
*??????.DBO
*WCDYO.PR4
*
*O
*24
*8
*
*
*N
*N
*USR1
*USR2
*USR3
*
*CPU%
*
*
*N
@IF (ERROR)
;HERE TO DO ANY REPORTS ON WEEKDAY WEEK FILES
DOWKO::
@RUN W2RPTB
*WEEKLY REPORT BY PROGRAM (TYPICAL 8-HOUR SHIFTS)
*?????.DBO
*WCWKO.PR3
*
*O
*24
*8
*
*
*N
*N
*PNAM
*
*CPU%
*
*
*N
;ONLY LIKELY ERROR IS NO READY WEEKDAY WEEK FILE...
;...WHICH MEANS WE CAN SKIP TO CHECK FOR WEEKEND DAY FILES
@IF (ERROR) @GOTO DODY1
@RUN W2RPTB
*WEEKLY REPORT BY USER (TYPICAL 8-HOUR SHIFTS)
*?????.DBO
```

Figure K-1 (continued)

```
*WCWKO.PR4
*
*O
*24
*8
*
*
*N
*N
*USR1
*USR2
*USR3
*
*CPU%
*
*
*N
@IF (ERROR)
;HERE TO DO ANY REPORTS ON WEEKEND DAY FILES
DODY1::
@RUN W2RPTB
*HOURLY REPORT BY PROGRAM AND JOB
*??????.DB1
*WCDY1.PR1
*
*O
*24
*1
*
*1
*N
*N
*USR1
*USR2
*USR3
*PNAM
*
*CPU%
*
*
*N
;ONLY LIKELY ERROR IS NO READY WEEKEND DAY FILE, WHICH MEANS...
;...NO WEEKEND WEEK, WEEKEND MONTH OR WEEKDAY MONTH FILES WILL BE READY
@IF (ERROR) @GOTO DOPRT
@RUN W2RPTB
*SHIFT REPORT BY PROGRAM
*??????.DB1
*WCDY1.PR2
*
*O
*24
*8
*
*.5
*N
*N
*PNAM
*
*CPU%
*
*
```

Figure K-1 (continued)

```
*N
@IF (ERROR)
@RUN W2RPTB
*SHIFT REPORT BY BATCH VS. TIMESHARING
*??????.DB1
*WCDY1.PR3
*
*O
*24
*8
*
*
*N
*N
*BATCH
*
*CPU%
*
*
*N
@IF (ERROR)
@RUN W2RPTB
*SHIFT REPORT BY USER
*??????.DB1
*WCDY1.PR4
*
*O
*24
*8
*
*
*N
*N
*USR1
*USR2
*USR3
*
*CPU%
*
*
*N
@IF (ERROR)
;HERE TO DO ANY REPORTS ON WEEKEND WEEK FILES
DOWK1::
@RUN W2RPTB
*WEEKLY REPORT BY PROGRAM (TYPICAL 8-HOUR SHIFTS)
*?????.DB1
*WCWK1.PR3
*
*O
*24
*8
*
*
*N
*N
*PNAM
*
*CPU%
*
*
```

Figure K-1 (continued)

```
*N
;ONLY LIKELY ERROR IS NO READY WEEKEND WEEK FILE, WHICH MEANS...
;...NO WEEKEND MONTH OR WEEKDAY MONTH FILES WILL BE READY
@IF (ERROR) @GOTO DOPRT
@RUN W2RPTB
*WEEKLY REPORT BY USER (TYPICAL 8-HOUR SHIFTS)
*?????.DB1
*WCWK1.PR4
*
*O
*24
*8
*
*
*N
*N
*USR1
*USR2
*USR3
*
*CPU%
*
*
*N
@IF (ERROR)
;HERE TO DO ANY REPORTS ON WEEKEND MONTH FILES
DOMN1::
@RUN W2RPTB
*MONTHLY REPORT BY PROGRAM (TYPICAL 8-HOUR SHIFTS)
*????.DB1
*WCMN1.PR3
*
*O
*24
*8
*
*
*N
*N
*PNAM
*
*CPU%
*
*
*N
;ONLY LIKELY ERROR IS NO READY WEEKEND MONTH FILE...
;...WHICH MEANS NO WEEKDAY MONTH FILES WILL BE READY
@IF (ERROR) @GOTO DOPRT
@RUN W2RPTB
*MONTHLY REPORT BY USER (TYPICAL 8-HOUR SHIFTS)
*????.DB1
*WCMN1.PR4
*
*O
*24
*8
*
*
*N
*N
*USR1
```

Figure K-1 (continued)

```
*USR2
*USR3
*
*CPU%
*
*
*N
@IF (ERROR)
;HERE TO DO ANY REPORTS ON WEEKDAY MONTH FILES
DOMNO::
@RUN W2RPTB
*MONTHLY REPORT BY PROGRAM (TYPICAL 8-HOUR SHIFTS)
*????.DBO
*WCMNO.PR3
*
*O
*24
*8
*
*
*N
*N
*PNAM
*
*CPU%
*
*
*N
@IF (ERROR)
@RUN W2RPTB
*MONTHLY REPORT BY USER (TYPICAL 8-HOUR SHIFTS)
*????.DBO
*WCMNO.PR4
*
*O
*24
*8
*
*
*N
*N
*USR1
*USR2
*USR3
*
*CPU%
*
*
*N
@IF (ERROR)
DOPRT:: .CHKPNT DOPRT
@DISABLE
@DEFINE DSK: <DB-STRUC>:<<AMAR-DIR>>
@APPEND WCMN%.PR%,WCWK%.PR%,WCDY%.PR% WORKLD.RPT
@IF (NOERROR) @GOTO DOCPY
@APPEND WCWK%.PR%,WCDY%.PR% WORKLD.RPT
@IF (NOERROR) @GOTO DOCPY
@APPEND WCDY%.PR% WORKLD.RPT
@IF (ERROR) @GOTO DOSUB
DOCPY::
@RENAME *.PR1 *.RP1
```

Figure K-1 (continued)

```
@IF (ERROR)
@RENAME *.PR2 *.RP2
@IF (ERROR)
@RENAME *.PR3 *.RP3
@IF (ERROR)
@RENAME *.PR4 *.RP4
@IF (ERROR)
@PRINT WORKLD.RPT/NOTE:"<NOTE>"/DELETE
:@XEROX WORKLD.RPT/NOTE:"<NOTE>"/DELETE
@IF (ERROR)
@GOTO DOSUB
%TERR::
%CERR::
%ERR::
DOSUB:: .CHKPNT DOSUB
%FIN::
@DISABLE
@DEFINE DSK: <DB-STRUC>:<<AMAR-DIR>>
@SUBMIT W2RPTB.CTL/TIME/AFTER:TODAY/LOGDISP:KEEP/BATCH-LOG:SUPERSEDE
```

Figure K-1 (continued)

Blank Page

APPENDIX L

INSTALLATION AND RESOURCE REQUIREMENTS

INSTALLATION:

This Appendix gives you an overview of the major steps in AMAR installation. For the specific installation procedure, refer to the AMAR-20 Installation Guide". Prior to installation, the computer center must resolve the following issues:

A. A four character code must be selected to represent the computer system running AMAR. This code is embedded in the System AMAR database and jobstream. It uniquely identifies the system to which the reports belong.

B. A directory must be set up on the system which will be running AMAR. The directory name can be anything the site wishes. This directory must have WHEEL and ENQ/DEQ privileges.

C. Next, temporary and permanent disk storage must be set up. The amount of disk space necessary is determined by the sizes of the System AMAR and Workload AMAR databases. The size of each database is a function of the amount of retained data at each summary level (i.e., monthly, weekly, daily, hourly) inside the database. See RESOURCE REQUIREMENTS following this section for a table that can be used to estimate the storage requirements.

D. AMAR has two data collection programs--xxxxDC (where xxxx is the four character code for the system) and WCDC20. These programs are run usually as SYSJOB subjobs and collect data continuously. In order to collect complete workload information, it may be necessary to rebuild the monitor. For a discussion of why this may be necessary refer to the Section called "Overview of Workload AMAR" in this manual or to the "AMAR-20 Installation Guide". In addition, xxxxDC and WCDC20 must be tailored/selected for the specific system on which they will run. Once installation is complete, entries for these programs should be manually started under SYSJOB or PTYCON and the following entries should be entered into the SYSJOB.RUN file:

```
job n \LOG amar-dir
ENABLE
CONNECT struc:<amar-dir>
RUN xxxxDC
\
JOB n+1 \LOG amar-dir
ENABLE
SYSDPY E
CONNECT struc:<amar-dir>
RUN WCDC20
\
```

E. For reporting purposes, each site must determine what portion of each day will be considered prime time. The prime time is recorded into the System AMAR database when it is built. The prime time interval must fall on whole hour boundaries. The default is 0800-1700. Up to 4 daily prime time periods may be specified. Both the System and Workload AMAR databases are built at installation time.

F. The AMAR-20 software monitor has very flexible report generating capabilities. Several System and Workload AMAR reports are supplied in the batch streams as defaults. The System AMAR reports can be tailored by editing the RFD file. The Workload AMAR reports can be tailored through the report program dialog. You can also easily select and deselect automatic reports by editing the batch streams.

G. There are two batch streams that run nightly. These are AMAR.CTL and W2RPTB.CTL. These must be tailored for the specific computer system on which they will run. The streams process the data collected during the day and generate reports. Although the streams are self-submitting, the logs should be checked daily. It is important that both streams run every night. If the streams are not run for an extended period of time, disk overflow and lengthy updating procedures will occur! Therefore, to restart the streams if they disappear from the queue, use the following commands:

```
@SUBMIT
struc:<amar-dir>AMAR.CTL/LOGDISP:KEEP/AFTER:TODAY+1
@SUBMIT
struc:<amar-dir>W2RPTB.CTL/LOGDISP:KEEP/AFTER:TODAY
```

Note: If a stream failed because of errors, simply resubmitting the stream may not work. The error should be looked up in AMAR-20 Error Messages in order to determine the appropriate course of action.

H. AMAR allows the site to set the holidays within the databases. Only holidays inserted into the database before they occur will be treated as holidays. Holidays must be respecified each fiscal year.

RESOURCE REQUIREMENTS:

AMAR is very efficient. Approximately .7% CPU time will be used on a KL10 for data collection, database maintenance, and reporting for both System and Workload AMAR combined. This may vary depending on the number of reports you select, the number of items you choose to measure, and the level of activity on your system.

TOTAL DISK STORAGE ESTIMATES IN PAGES

	System AMAR	Workload AMAR
Program Software	1500	300
Data Files	300	3000
TOTAL (without databases)	1800	3300

	Database*			
Size	Retention	Pages	Retention**	Pages
mini	M-12 W-13 D-35 H-7	1500	M-2 W-2 D-7	3500
midi	M-12 W-13 D-35 H-7 COW-1 COM-1	2000	M-3 W-5 D-14	7000
maxi	M-12 W-13 D-35 H-7 COW-5 COM-3	2500	M-12 W-13 D-35	20000

* A backup copy of the System AMAR database should also be kept on disk. Typically it is kept on a different pack from the primary database, although that is not a requirement.

Note that more than one permanent structure may be used for holding the software, databases, files, etc.

** The abbreviations for the retentions are as follows:

M - monthly
W - weekly
D - daily
H - hourly
COW - composite weekly
COM - composite monthly

Example:

For midi System AMAR and Workload databases the total space required is:

System AMAR programs and files	1,800
Workload Programs and files	3,300
System AMAR database	2,000
Workload database	7,000
Backup System AMAR database	2,000
TOTAL	16,100

It should be understood that these totals are only estimates. The actual disk space used is a function of the amount of data collected and retained. Typically, systems that are very busy will have slightly larger data files and databases than systems that aren't -- despite having the same retention periods.

APPENDIX M

SUMMARY OF PROGRAMS AND FILENAMES

The following programs and files may be found in the AMAR area. Most programs and files are used in daily processing. However, a few are used only in special cases such as installation and error recovery. These program names and filenames are considered reserved for AMAR use only. The user should take care to select other filenames when specifying report files, etc.

System AMAR:

AMAR.BWR	- AMAR Beware File
AMAR.DB	- System AMAR database
AMAR.DBK	- Backup copy of System AMAR database
AMAR.1ST	- Initialization file used only during AMAR installation
AMAR.CTL	- Edited System AMAR batch stream
AMAR.TAP	- Control file used by AMAR.CTL for scheduling of tape backup
AMAR20.CTO	- Unedited batch stream
AMAR20.EXE	- Unedited data collection program
AMAR20.RFD	- Unedited Report File Description
AMAREX.EXE	- Data Extraction Program
AMAREX.HLP	- Data Extraction Program help file
AMARIP.EXE	- Input Program
AMARON.EXE	- Online Inquiry Program
AMARSD.EXE	- Raw File Preprocessor Program
AMARUP.EXE	- Rollup Program
AMARcc.RPT*	- Default reports produced by AMAR.CTL
AMONLD.EXE	- Reserved for use in troubleshooting by the performance specialist
AMREPT.EXE	- Reporting Program
AMRGEN.EXE	- Database Generation Program
EOF.EXE	- File Deletion Program
RFD.HLP	- Report File Description help file
TODAY.DB	- Mini-database output from AMARSD.EXE
xxxxDC.EXE	- Data Collection Program
xxxxDR.RFD	- Report File Description
xxxxdd.mmm*	- Raw data files created by xxxxDC.EXE

*See the notes starting at the end of this section.

Workload AMAR:

W2RPTB.CTO	- Unedited batch stream
W2RPTB.CTL	- Edited batch stream
W2RPT.EXE	- Incremental file report program
W2RPTB.EXE	- Database report program
W2UPD.EXE	- Database update program
WC.IN0	- Summary incremental file output by WCDC20.EXE
WC.IN1	- Detail incremental file output by WCDC20.EXE
WCX.IN0	- Yesterday's summary incremental file
WCX.IN1	- Yesterday's detail incremental file
WCY.IN0	- Two day's ago summary incremental file
WCY.IN1	- Two day's ago detail incremental file
WCDC20.EXE	- Selected data collection program
WCDBS.CON	- Control file for Workload database
WCDBS.COX	- Yesterday's WCDBS.CON (used for recovery)
WCDBS.LOK	- Access control file for database
WCDATE.REL	- Date file
WCINIT.REL	- Database initialization file
WCGEN.REL	- Database generation file
WCFIX.EXE	- Database maintenance utility
WCPARM.MAC	- Parameter file
WCnnnP.EXE	- Selectable data collection programs for resident program name table
WCnnnS.EXE	- Selectable data collection programs for non-resident program name table
WCDYn.RPz*	- Default daily reports produced by W2RPTB.CTL
WCMNn.RPz*	- Default monthly reports produced by W2RPTB.CTL
WCWKn.RPz*	- Default weekly reports produced by W2RPTB.CTL
yyqm.DB0*	- Monthly workload database files for weekdays
yyqm.DB1*	- Monthly workload database files for weekends and holidays
yyqmw.DB0*	- Weekly workload database files for weekdays
yyqmw.DB1*	- Weekly workload database files for weekends and holidays
yyqmwd.DB0*	- Daily workload database files for weekdays
yyqmwd.DB1*	- Daily workload database files for weekends and holidays

*

d = day numbered 1 - 7 depending on its position within the fiscal week

m = month numbered 1 - 3 depending on its position within the fiscal quarter

n = 0 for weekdays, 1 for weekends and holidays

q = quarter numbered 1 - 4 depending on its position with the fiscal year

w = week numbered 1 - 5 (and, on rare occasion, 6) depending on its position within the fiscal month

z = 0, 1, etc.

cc = report code

dd = calendar day

yy = fiscal year

mmm = alpha abreviation for month

nnn = number identifying unedited versions of the data collection program

xxxx = system code

Blank Page

APPENDIX N

THE FISCAL CALENDAR

The fiscal calendar is used primarily to control rollup of data into weekly and monthly summary records, deletion of old data, and automatic reporting. It is based on a year divided into quarters of 3 months each. The first and second months within each quarter contain 4 weeks each. The third month normally contains 5 weeks. Each fiscal week contains 7 days starting on a Sunday and ending on a Saturday. There are exactly 52 weeks in the normal fiscal year with no leftover days. Approximately once every 6 years, an extra week is added to the fiscal calendar to compensate for the days which have been "lost" by this process. This extra 6th week occurs in the third month of the fourth quarter. It last occurred in FY82.

Figure N-1 is an example of the FY83 Digital fiscal calendar. The Digital calendar normally starts at the end of June or beginning of July. This particular calendar begins on July 4, 1982.

Fiscal dates are used extensively by AMAR. They are always specified in the following format:

```
yyqmwd
     |- Fiscal day (1-7)
    |-- Fiscal week (1-5)
   |--- Fiscal month (1-3)
  |---- Fiscal quarter (1-4)
 |----- Fiscal year
```

For example, by looking at Figure N-1, it can be determined that:

September 1, 1982 = 831314

September occurs in quarter 1 of FY82; it is the 3rd month;

September 1 occurs in week 1; it is the 4th day.

It is recommended that you keep a similar copy of your fiscal calendar on hand for quick help in such date translations.

1983 FISCAL CALENDAR 1983

FIRST QUARTER

MONTH	WEEK NO.	S 1	M 2	T 3	W 4	T 5	F 6	S 7
JULY 4 WEEKS	1	4	5	6	7	8	9	10
	2	11	12	13	14	15	16	17
	3	18	19	20	21	22	23	24
	4	25	26	27	28	29	30	31
AUG 4 WEEKS	5	1	2	3	4	5	6	7
	6	8	9	10	11	12	13	14
	7	15	16	17	18	19	20	21
	8	22	23	24	25	26	27	28
SEPT 5 WEEKS	9	29	30	31	1	2	3	4
	10	5	6	7	8	9	10	11
	11	12	13	14	15	16	17	18
	12	19	20	21	22	23	24	25
	13	26	27	28	29	30	1	2

SECOND QUARTER

MONTH	WEEK NO.	S	M	T	W	T	F	S
OCT 4 WEEKS	14	3	4	5	6	7	8	9
	15	10	11	12	13	14	15	16
	16	17	18	19	20	21	22	23
	17	24	25	26	27	28	29	30
NOV 4 WEEKS	18	31	1	2	3	4	5	6
	19	7	8	9	10	11	12	13
	20	14	15	16	17	18	19	20
	21	21	22	23	24	25	26	27
DEC 5 WEEKS	22	28	29	30	1	2	3	4
	23	5	6	7	8	9	10	11
	24	12	13	14	15	16	17	18
	25	19	20	21	22	23	24	25
	26	26	27	28	29	30	31	1

THIRD QUARTER

MONTH	WEEK NO.	S	M	T	W	T	F	S
JAN 4 WEEKS	27	2	3	4	5	6	7	8
	28	9	10	11	12	13	14	15
	29	16	17	18	19	20	21	22
	30	23	24	25	26	27	28	29
FEB 4 WEEKS	31	30	31	1	2	3	4	5
	32	6	7	8	9	10	11	12
	33	13	14	15	16	17	18	19
	34	20	21	22	23	24	25	26
MARCH 5 WEEKS	35	27	28	1	2	3	4	5
	36	6	7	8	9	10	11	12
	37	13	14	15	16	17	18	19
	38	20	21	22	23	24	25	26
	39	27	28	29	30	31	1	2

FOURTH QUARTER

MONTH	WEEK NO.	S	M	T	W	T	F	S
APRIL 4 WEEKS	40	3	4	5	6	7	8	9
	41	10	11	12	13	14	15	16
	42	17	18	19	20	21	22	23
	43	24	25	26	27	28	29	30
MAY 4 WEEKS	44	1	2	3	4	5	6	7
	45	8	9	10	11	12	13	14
	46	15	16	17	18	19	20	21
	47	22	23	24	25	26	27	28
JUNE 5 WEEKS	48	29	30	31	1	2	3	4
	49	5	6	7	8	9	10	11
	50	12	13	14	15	16	17	18
	51	19	20	21	22	23	24	25
	52	26	27	28	29	30	1	2

Please note:
All listed holidays are standard throughout the U.S. Assignable holidays and local location holidays are not shown.

digital

Figure N-1