digital

Software Services

AMAR

Automatic Measurement, Analysis, and Reporting
A Computer Performance Monitoring System
for the
DECsystem-10 and DECSYSTEM-20

AMAR provides performance measurement and system management capabilities to a wide range of users including data center managers, software specialists, and application developers.

AMAR consists of two parts called System AMAR and Workload AMAR. System AMAR monitors the activity of the system as a whole, collecting data from the monitor tables and control blocks and, on a DECsystem-10 KL10, the performance meter. Workload AMAR monitors the corresponding activity of terminal sessions and batch jobs. On a DECsystem-10, it collects data from the job tables. On a DECSYSTEM-20, it collects data at breakpoints dynamically inserted into the operating system.

Samples are taken once a minute for System AMAR metrics and once every 5 minutes for Workload AMAR metrics. AMAR data collection runs continuously with very low overhead. AMAR will automatically maintain a set of history files on system and workload activity according to retention periods specified by the user. It will automatically produce daily, weekly, and monthly reports according to a predefined schedule. The user may also request special reports.

AMAR REPORTS WILL HELP YOU:

- Develop a feel for what is "normal" for your system.
- Recognize periods of abnormally high utilization and/or poor performance and understand the causes.
- Identify programs which are heavy resource consumers and should be optimized or restricted to non-prime periods.
- Recognize typical peaks and valleys in utilization so that you can work toward a more level load.
- Recognize overloaded subsystems so that you can balance the load, restrict the load, or obtain additional resources.
- Recognize trends in utilization to assist in planning.
- Reduce the time to diagnose performance problems because relevant data will have already been collected.

MAJOR FEATURES OF AMAR:

- Continuous data collection with very low overhead
- Extensive set of system-wide and job-specific metrics
- Historical database
  - Data summarized at the hourly, daily, weekly, monthly, and "typical day" levels.
  - Flexible retention periods for data.
  - Automatic deletion of old data.
- Powerful reporting capabilities.
  - Menu of automatic daily, weekly, and monthly reports.
  - Simple on-line inquiry.
  - Highlighting of system performance items exceeding redefinable limits.
  - Flexible grouping and sorting of work-load data.

AMAR SYSTEM REPORTS:

System AMAR contains five categories of reports which may be produced automatically or on demand. In addition, two special reporting programs enable the user to query the database, outputting the data at the terminal or into a file for later printing or use in the user's own programs.

- System Utilization Report
  - Characterizes utilization for a day, week, or month.
  - Includes a graph of system utilization and overhead.
  - Highlights problem resources and problem periods.
  - Gives supporting details: hourly on the daily report, daily on the weekly, weekly on the monthly.
- System Utilization Typical Day Report
  - Characterizes "average" day for a week or a month.
- Trend Analysis Report
  - Shows trends over the past 13 weeks or 12 months.
  - Graph shows utilization and overhead for reported period.
- Disk Report
  - Characterizes disk usage for a day, week, or month.
  - Details per pack include time mounted, time in use, average data transfer rate, and free space.
- Tape Report
  - Characterizes tape usage for a day, week, or month.
  - Details per drive include time assigned, time in use, and average data transfer rate.
- Special inquiry reports
  - Averages or frequency distributions for any measured system parameters.

AMAR WORKLOAD REPORTS:

Workload AMAR reports are generated by a program which allows flexibility in defining report contents. The user may specify how a report is to be summarized and sorted. Several standard reports are provided. The following are examples of some of the types of reports you can obtain:

- Hourly report by user and program name, with suppression of detail lines representing less than 1% of processor usage.
- Weekly or monthly report showing program usage by shift, sorted in descending order of processor use.
- Weekly or monthly report showing resource usage by user by shift.

For additional information about AMAR, contact your local Software Services organization.

WHAT AMAR MEASURES:

Listed below are some of the more than 60 AMAR-10 metrics and 90 AMAR-20 metrics.

AMAR-10 SYSTEM METRICS:

- Percentages of processor time - idle, lost, overhead, at each priority interrupt level
- Monitor call rate
- Scheduler response time
- Channel busy time
- Runnable jobs in memory
- Active swapping ratio
- Blocks swapped per second
- Context switch rate
- Queue lengths: Run, Interactive I/O Noninteractive I/O, Channel Wait
- For each logical or physical disk - blocks read/written per second, mounted time, in use time, position wait queue length
- For each tape unit - I/O rate, assigned, time, in use time
- Terminal characters in/out

AMAR-20 SYSTEM METRICS:

- Percentages of processor time - idle, file wait, swap wait, user, memory management, scheduler, and background time
- Number of processes - active, in memory, in balance set
- Rates - context switching, process wake-ups, working set loads and removals, balance set adjustments, memory management
- Pages per second - faulted, swapped in/out, saved from replaceable queue
- For each logical or physical disk - pages read/written per second, seeks per second, mounted time, in use time, position wait queue length
- For each tape unit - I/O rate, assigned time, in use time
- Terminal characters in/out

AMAR-10 WORKLOAD METRICS:

- Project/programmer number
- Scheduler class
- HPQ number
- Date/time of login
- Batch or timesharing
- Program name
- Percentage of processor
- Job size
- Disk blocks read/written
- Monitor call rate
- Job state
- Node/line number
- TTY number

AMAR-20 WORKLOAD METRICS:

- Login directory (15 chrs.)
- Account string (15 chrs.)
- Date/time of login
- Batch or timesharing
- Job program name
- Percentage of processor
- Process working set size
- Swap page faults
- File page faults
- Average CPU utilization between page faults
- Interactive responses per minute
- Seconds per interactive response
- Stretch ratio for interactive responses

digital

Software
Services

AMAR-10
REFERENCE MANUAL
VERSION 4.1

AMAR-10
REFERENCE MANUAL
VERSION 4.1

Digital Equipment Corporation

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

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

AMAR-10 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-10 Reference Manual provides an overview of how AMAR works on the DECsystem-10. 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-10 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, PPN, program name, core size, CPU utilization, I/O activity, scheduling class, 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, PPN, charge number) 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-10. 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, CPUn Idle Time (CPIn) 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.

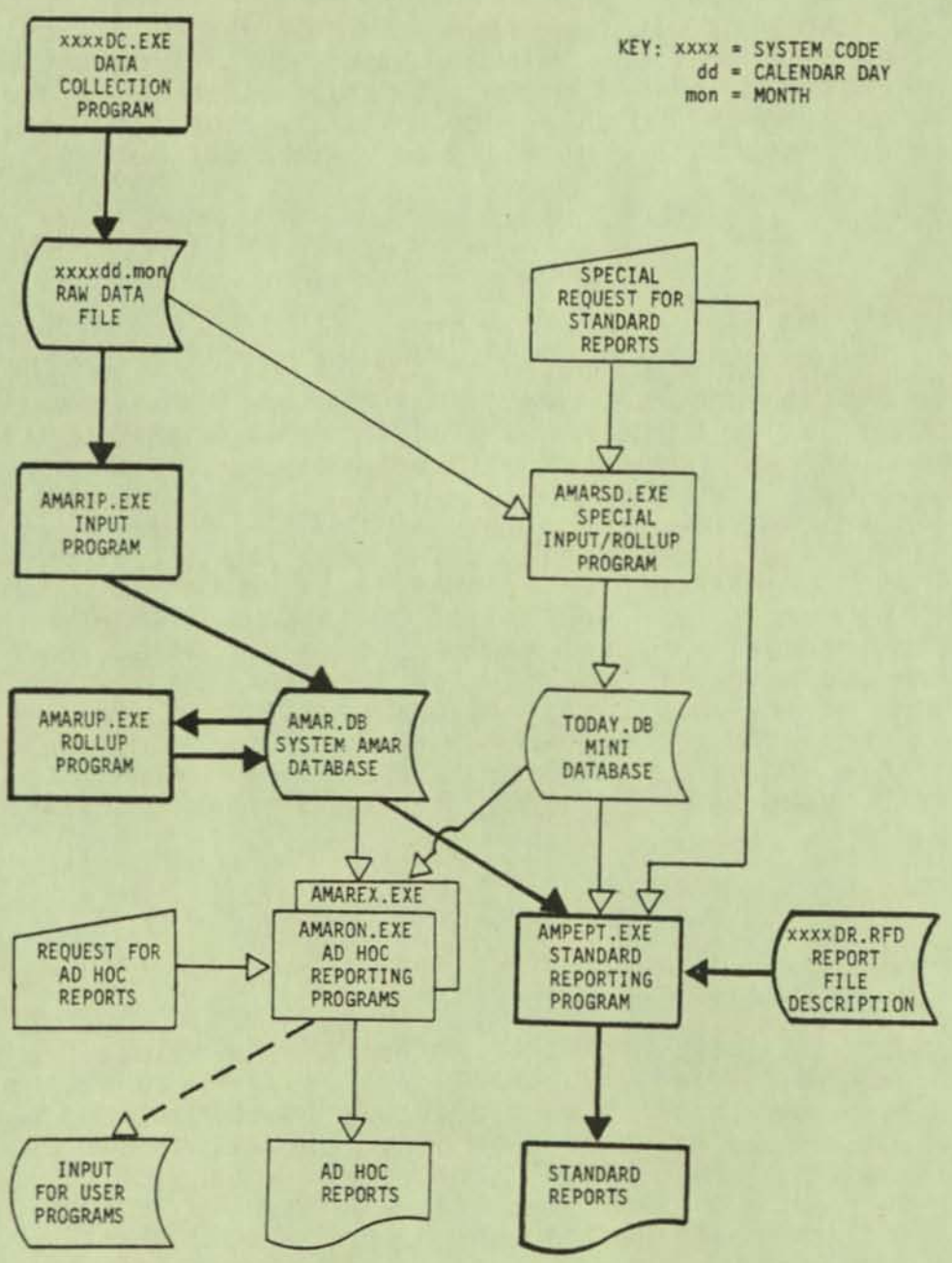
1.2.1 Data Collection

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 an OPSER 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 summary file is then forwarded to AMREPT for output in 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 an OPSER subjob startable by the OPR.ATO 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 jobs. 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.

A prime/non-prime time summary of key utilization items such as idle time, overhead, swapping rate, disk I/O rate, 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 jobs.

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 (Page 1) - 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) represents processor 0. It shows CPU utilization (*) and overhead (#) for each hour of the day. The difference between the CPU utilization (B) and overhead (C) lines represents CPU time consumed by user jobs, lost time, and some embedded priority interrupt (PI) time.

Below the graph are prime and non-prime averages for ten "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 3 and 4 of the sample report.

The last section on page 1 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 2) - 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 3).

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 high overhead) 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 3-4) - 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 jobs, refer to the Workload AMAR "Hourly Report Showing Major CPU Users". 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 ACT SWAP RATIO.

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: 15-FEB-82 (MONDAY)

- AMAR -
DAILY SYSTEM UTILIZATION SUMMARY REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

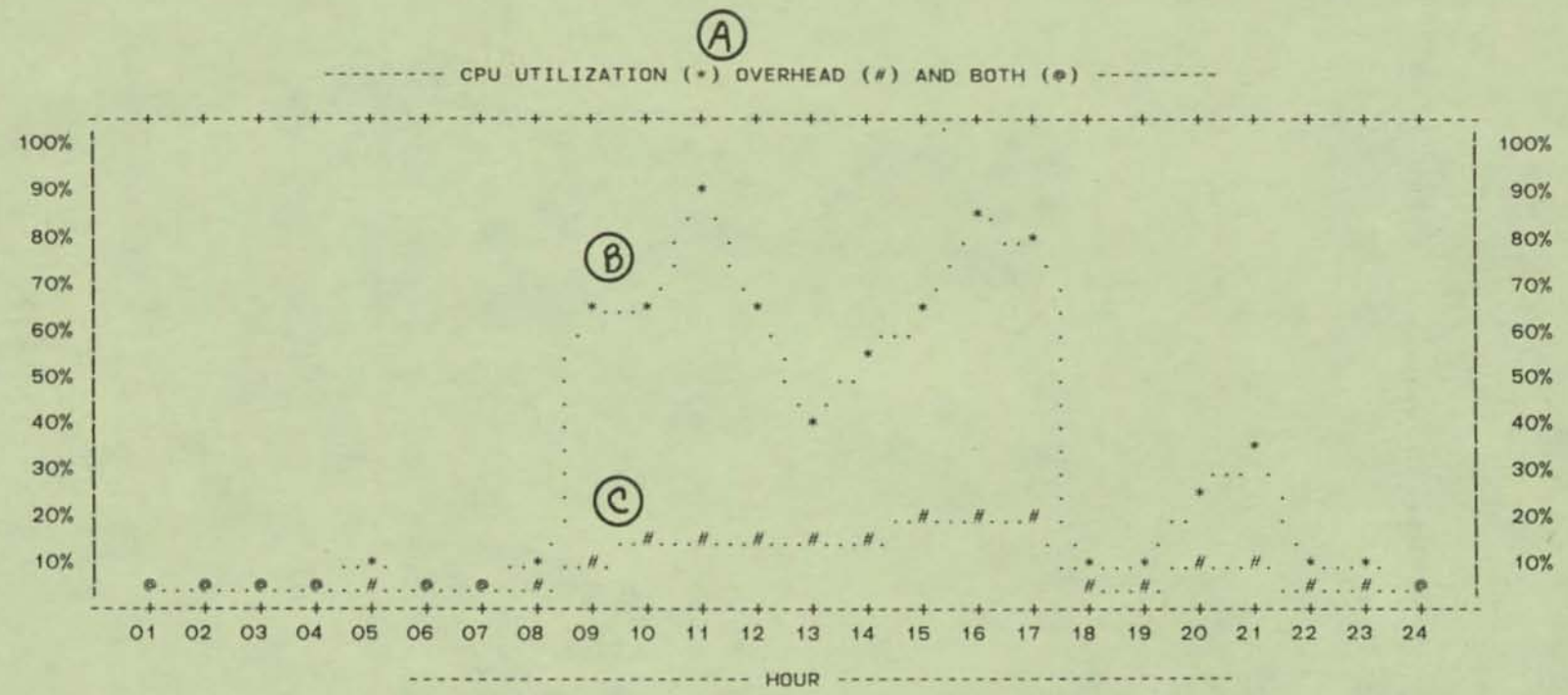


Figure 1-2

D

SUMMARY OF KEY UTILIZATION ITEMS

-------AVERAGE-----	# JOBS LOGGED IN	# LINES IN USE	% IDLE TIME	% LOST TIME	% OVHD TIME	ACT SWAP RATIO	SWAPPING BLKS/SEC	USER DSK BLKS/SEC	# MTAS ASSIGNED	PGS USER MEMORY	NO. OF HOURS KEY ITEMS OVER LIMITS
----PRIME TIME-----	58	35	34	1	14	.2	35	60	0	1737	1 (E)
---NON-PRIME TIME--	23	4	91	0	5	.0	0	14	0	1736	

PERCENT FREE SPACE LEFT ON DISK PACKS

-----PACK NAME-----	CASEO	COREO	DEVPO	DSKEO	DSKWO	HRS2O	PAWSO	PENSO	PER6O	PLT1O	STARO	TRNGO	USRSO
---% FREE TODAY----	42	46	16	30	78	44	49	13	23	42	41	74	19 (F)
-% FREE YESTERDAY--	49	46	3	29	80		49	13			41		26 (G)
----DIFFERENCE-----	- 7	+ 0	+13	+ 1	- 2		+ 0	+ 0			+ 0		- 7 (H)

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

DATE: 15-FEB-82 (MONDAY)

- AMAR -
DAILY SYSTEM UTILIZATION PROBLEM REPORT

PUT ANY TITLE HERE
SYSTEM: PATH 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)
USER DSK BLKS/SEC	P	
USER UUOS/SEC	PPPP	
% SWAP SPC LEFT	P	
# JOBS TTY IO Q	PPPP	
%PK DEVPO FREE SPC	PPPPPP	
PK DEVPO BLKS/SEC	P	

COMMENTS		(M)
WARNING	DISK I/O RATE HIGH: CHK PACK I/O	
SERIOUS	USER PGM PROBLEM: CHK WORKLD DATA	
WARNING	SWAP SPACE LOW: ALLOCATE MORE	
SERIOUS	INTERACTIVE USE HIGH: CHK RESPONSE	
CRITICAL	DELETE UNNECESSARY FILES	
WARNING	I/O RATE HIGH: CHK FOR CONTENTION	

(N)
SYSTEM AVAILABILITY SUMMARY

	(P) SYSTEM UPTIME	(Q) % TIME MEASURED	(R) RELOADS	PERIODS OF DOWNTIME (S)
PRIME TIME :	100.0%	100.0%		17:28 - 17:30
NON-PRIME TIME:	99.8%	96.5%	1	

Figure 1-2 (continued)

DATE: 15-FEB-82 (MONDAY)

- AMAR -
DAILY SYSTEM UTILIZATION DETAIL REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

PERIOD	# JOBS LOGGED IN	# LINES IN USE	% IDLE TIME	% LOST TIME	% OVHD TIME	ACT SWAP RATIO	SWAPPING BLKS/SEC	USER DSK BLKS/SEC	# MTAS ASSIGNED	PGS USER MEMORY	# ITEMS OVER LIMITS: THIS PAGE	# ITEMS OVER LIMITS: ALL PAGES
00:00 - 01:00	21	4	96	0	3	.0	0	3	0	1737		
01:00 - 02:00	22	4	96	0	3	.0	0	1	0	1737		
02:00 - 03:00	22	4	96	0	3	.0	0	0	0	1737		
03:00 - 04:00	22	4	96	0	3	.0	0	0	0	1737		
04:00 - 05:00	22	4	94	0	4	.0	0	3	0	1737		
05:00 - 06:00	22	4	96	0	3	.0	0	0	0	1737		
06:00 - 07:00	22	4	96	0	3	.0	0	0	0	1737		
07:00 - 08:00	24	5	90	0	4	.0	2	6	0	1737		
08:00 - 09:00	37	19	39	4	10	.1	48	51	0	1737		
09:00 - 10:00	55	32	38	1	11	.2	17	45	0	1737		
10:00 - 11:00	62	37	10	0	14	.2	18	159 *	0	1737	1	4
11:00 - 12:00	55	32	35	0	11	.1	6	21	0	1737		1
12:00 - 13:00	58	34	61	1	12	.1	7	45	1	1737		1
13:00 - 14:00	61	38	48	1	15	.2	19	35	0	1737		3
14:00 - 15:00	62	39	38	1	18	.2	22	52	0	1737		3
15:00 - 16:00	71	46	19	3	19	.2	136	43	1	1737		4
16:00 - 17:00	59	35	22	1	16	.2	43	87	1	1737		1
17:00 - 18:00	24	6	92	0	5	.0	3	27	0	1735		
18:00 - 19:00	24	5	92	0	5	.0	1	4	0	1735		
19:00 - 20:00	24	4	78	0	7	.0	0	61	0	1735		
20:00 - 21:00	24	4	68	0	10	.0	0	91	1	1735		
21:00 - 22:00	24	6	90	0	5	.0	1	12	1	1735		
22:00 - 23:00	23	5	93	0	4	.0	0	3	0	1735		
23:00 - 24:00	22	4	95	0	4	.1	0	0	0	1735		
----PRIME TIME-----												
AVERAGE VALUE:	58	35	34	1	14	.2	35	60	0	1737		
SHORT TERM LIMIT:	NONE	NONE	<5%	>5%	>20%	>.7	>200	>150	NONE	<768		
% TIME OVER LIMIT:			31.1%					7.5%				
# HOURS OVER LIMIT:								1			1	17
--NON-PRIME TIME---												
AVERAGE VALUE:	23	4	91	0	5	.0	0	14	0	1736		
SHORT TERM LIMIT:	NONE	NONE	<5%	>5%	>20%	>.7	>200	>150	NONE	<768		
% TIME OVER LIMIT:												
# HOURS OVER LIMIT:												

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

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

Figure 1-2 (continued)

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

PERIOD	USER UUOS/SEC	# JOBS BLK IO Q	%PK DEVPO FREE SPC	PK DEVPO BLKS/SEC	% SWAP SPC LEFT	# JOBS TTY IO Q	# ITEMS OVER LIMITS THIS PAGE	# ITEMS OVER LIMITS ALL PAGES
00:00 - 01:00	7	.0	24	0	89	0		
01:00 - 02:00	9	.0	24	0	89	0		
02:00 - 03:00	8	.0	24	0	89	0		
03:00 - 04:00	8	.0	24	0	89	0		
04:00 - 05:00	15	.1	24	0	89	0		
05:00 - 06:00	8	.0	24	0	89	0		
06:00 - 07:00	10	.0	24	0	89	0		
07:00 - 08:00	49	.2	24	0	85	1		
08:00 - 09:00	144	.9	24	5	67	8		
09:00 - 10:00	141	.4	24	2	49	17		
10:00 - 11:00	201	1.0	10 *	101 *	39	20 *	3	4
11:00 - 12:00	173	.4	3 *	0	43	13	1	1
12:00 - 13:00	205	.3	3 *	11	38	16	1	1
13:00 - 14:00	597 *	.9	3 *	0	32	20 *	3	3
14:00 - 15:00	573 *	1.0	3 *	0	35	20 *	3	3
15:00 - 16:00	470 *	.9	3 *	0	19 *	27 *	4	4
16:00 - 17:00	358 *	.7	12	20	33	17	1	1
17:00 - 18:00	58	.1	13	0	83	2		
18:00 - 19:00	50	.1	13	0	83	1		
19:00 - 20:00	182	.3	13	0	84	0		
20:00 - 21:00	248	.6	13	0	85	0		
21:00 - 22:00	42	.4	13	0	85	1		
22:00 - 23:00	24	.1	13	0	87	1		
23:00 - 24:00	17	.2	13	0	87	0		
----PRIME TIME-----								
AVERAGE VALUE:	318	.7	9	16	39	17 (W)		
SHORT TERM LIMIT:	>300	>2.0	<10%	>100	<25	>20 (X)		
% TIME OVER LIMIT:	32.2%	18.8%	68.9%	2.4%	13.8%	39.1% (Y)		
# HOURS OVER LIMIT:	4		6	1	1	4 (Z)	16	17
--NON-PRIME TIME---								
AVERAGE VALUE:	49	.1	19	0	87	0		
SHORT TERM LIMIT:	>300	>2.0	<10%	>100	<25	>20		
% TIME OVER LIMIT:								
# HOURS OVER LIMIT:								

* = 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.

A prime/non-prime time summary for the week and each day in the week of utilization and performance items such as idle time, overhead, 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 (Page 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 "Program Name Report", "Batch Vs. Timesharing Report", or the "PPN Report".

The graph at the top of page 1 (A) represents processor 0. It is a bar chart which shows CPU utilization (*) and overhead (#) for each 2 hour period in the day starting at midnight. The difference between the top of the CPU utilization (B) and overhead (C) bars represents CPU time consumed by user jobs, lost time, and some embedded priority interrupt (PI) time.

Below the graph are prime and non-prime time averages for eleven 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 (E) 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 high overhead) 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.

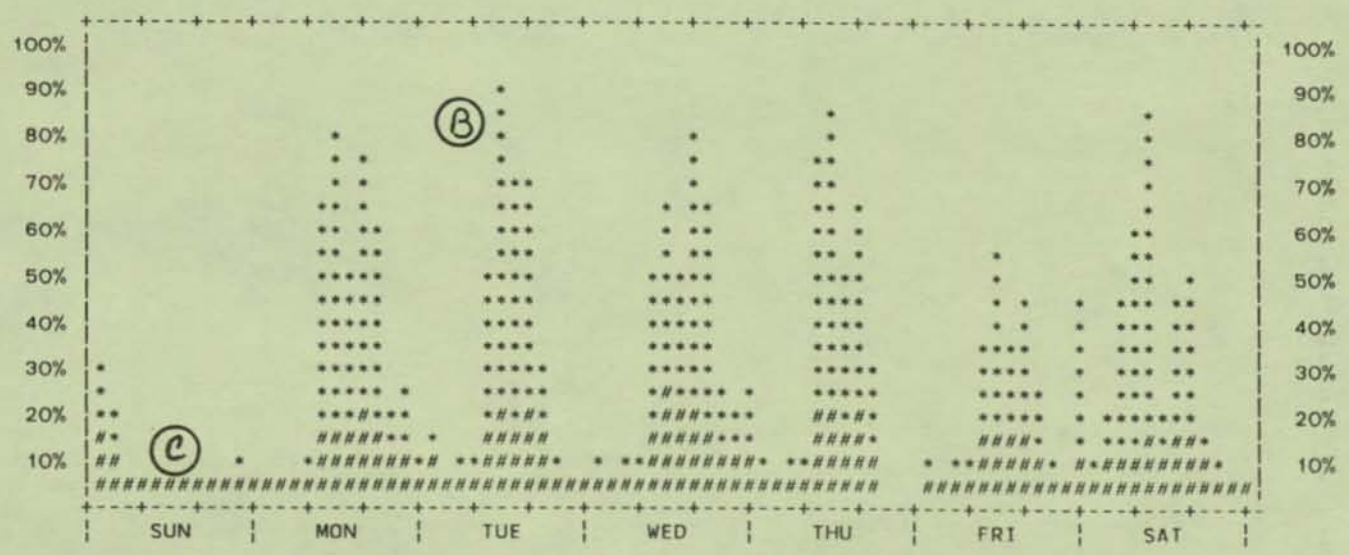
In addition to reporting which items had problems over the past week, this section also reports which days experienced the most time in a problem state. Again the item is given along with the percent of time that the item's values were over the long term limit during each day (R) in the week. The prime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

FROM: 14-FEB-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
WEEKLY UTILIZATION SUMMARY REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

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



---------- EACH COLUMN REPRESENTS 2 HOURS ----------

---------- SUMMARY OF KEY UTILIZATION ITEMS ---------- (D)

-------AVERAGE-----	% CPU UTIL	# JOBS LOGGED IN	# LINES IN USE	% IDLE TIME	% LOST TIME	% OVHD TIME	ACT SWAP RATIO	SWAPPING BLKS/SEC	USER DSK BLKS/SEC	# MTAS ASSIGNED
----PRIME TIME-----	59%	59	35	41%	1%	15%	.2	33	79	0
---NON-PRIME TIME--	13%	24	5	87%	0%	5%	.0	1	30	0

-------AVERAGE-----	PGS USER MEMORY	% SYSTEM UPTIME	% AMAR CLK TIME	# SYSTEM RELOADS
----PRIME TIME-----	1735	99.9%	98.8%	1
---NON-PRIME TIME--	1735	96.4%	95.0%	4
		(E)	(F)	(G)

Figure 1-3

FROM: 14-FEB-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
WEEKLY UTILIZATION DETAIL REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

--- PRIME TIME ---

KEY UTILIZATION ITEMS		AVERAGE OF CURRENT -WEEK-	-SUN-	-MON-	-TUE-	-WED-	-THU-	-FRI-	-SAT-
		DAILY AVERAGE							
% CPU	UTIL	59%		66%	64%	62%	63%	39%	
# JOBS	LOGGED IN	59		58	56	60	64	59	
# LINES	IN USE	35		35	36	36	37	33	
% IDLE	TIME	41%		34%	36%	38%	37%	61%	
% LOST	TIME	1%		1%	1%	1%	1%	2%*	
% OVHD	TIME	15%		14%	15%	17%	16%	12%	
ACT SWAP	RATIO	.2		.2	.2	.2	.2	.1	
SWAPPING	BLKS/SEC	33		35	23	33	45	28	
USER DSK	BLKS/SEC	79		60	82	80	114 *	61	
# MTAS	ASSIGNED	0		0	0	1 .	0	1	
PGS USER	MEMORY	1735		1737	1735	1735	1735	1732	
% SYSTEM	UPTIME	99.9%		100.0%	100.0%	99.6%	100.0%	100.0%	
% AMAR	CLK TIME	98.8%		100.0%	100.0%	94.0%	100.0%	100.0%	
# SYSTEM	RELOADS	1		0	0	1	0	0	

--- PRIME TIME ---

OTHER UTILIZATION ITEMS		AVERAGE OF CURRENT -WEEK-	-SUN-	-MON-	-TUE-	-WED-	-THU-	-FRI-	-SAT-
		DAILY AVERAGE							
USER	UUOS/SEC	249 *		318 *	219 *	296 *	238 *	179	
# JOBS	BLK IO Q	1.0		.7	1.5 *	1.0	1.1	.6	
%PK DEVPO	FREE SPC	9%*		9%*	8%*	4%*	10%*	11%	
%PK HRS20	FREE SPC	13%		---	18%	11%	11%	6%*	
# JOBS	IN RUN Q	2		2	3	3	3	2	
% SWAP	SPC LEFT	33		39	51	33	18 *	25 *	
# JOBS	TTY IO Q	17 *		17 *	16 *	16 *	18 *	17 *	

--- NON-PRIME TIME ---

KEY UTILIZATION ITEMS		AVERAGE OF CURRENT -WEEK-	-SUN-	-MON-	-TUE-	-WED-	-THU-	-FRI-	-SAT-
		DAILY AVERAGE							
% CPU	UTIL	13%	7%	9%	8%	15%	8%	11%	28%
# JOBS	LOGGED IN	24	21	23	24	25	26	26	27
# LINES	IN USE	5	4	4	5	5	6	5	5
% IDLE	TIME	87%	93%	91%	92%	85%	92%	89%	72%
% LOST	TIME	0%	0%	0%	0%	0%	0%	0%	0%
% OVHD	TIME	5%	5%	5%	4%	5%	4%	5%	7%
ACT SWAP	RATIO	.0	.0	.0	.0	.0	.0	.0	.0
SWAPPING	BLKS/SEC	1	0	0	1	2	2	0	1
USER DSK	BLKS/SEC	30	6	14	7	52	5	27	74
# MTAS	ASSIGNED	0	0	0	0	0	0	0	0
PGS USER	MEMORY	1735	1737	1736	1735	1735	1735	1732	1732

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

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

Figure 1-3 (continued)

FROM: 14-FEB-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
WEEKLY UTILIZATION DETAIL REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

--- NON-PRIME TIME ---

KEY UTILIZATION ITEMS	AVERAGE OF CURRENT -WEEK-	DAILY AVERAGE -SUN-	-MON-	-TUE-	-WED-	-THU-	-FRI-	-SAT-
% SYSTEM UPTIME	96.4%	100.0%	99.8%	99.8%	99.8%	71.4%	100.0%	100.0%
% AMAR CLK TIME	95.0%	100.0%	96.5%	97.6%	99.6%	64.9%	100.0%	100.0%
# SYSTEM RELOADS	4	0	1	1	1	1	0	0

--- NON-PRIME TIME ---

OTHER UTILIZATION ITEMS	AVERAGE OF CURRENT -WEEK-	DAILY AVERAGE -SUN-	-MON-	-TUE-	-WED-	-THU-	-FRI-	-SAT-
%PK DEVPO FREE SPC	9%*	3%*	19%	9%*	5%*	2%*	11%	11%
%PK HRS20 FREE SPC	15%		---	27%	11%	10%*	7%*	---

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

Figure 1-3 (continued)

FROM: 14-FEB-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
WEEKLY UTILIZATION PROBLEM REPORT

PUT ANY TITLE HERE
SYSTEM: PATH 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
% IDLE	TIME	41%	<30%	37.9% (L)	87%	<30%	3.2%	CPU PRESSED: CHK WORKLD DATA FIRST
% LOST	TIME	1%	>2%	17.7%	0%	>2%	.7%	MEMORY SHORTAGE OR SWAP DEVICE SLOW
% OVHD	TIME	15%	>20%	14.3%	5%	>20%	.0%	OVERHEAD TOO HIGH: INVESTIGATE
USER	UUOS/SEC	249 (K)	>200	49.3%	48	>200	7.2%	USER PGM PROBLEM: CHK WORKLD DATA
# JOBS	BLK IO Q	1.0	>1.5	25.5%	.3	>1.5	4.4%	BACKUP OF I/O JOBS: CHK CONTENTION
%PK DEVPO	FREE SPC	9%	<10%	46.7%	9%	<10%	41.0%	DELETE UNNECESSARY FILES
%PK HRS20	FREE SPC	13%	<10%	26.0%	15%	<10%	35.9%	DELETE UNNECESSARY FILES
# JOBS	IN RUN Q	2	>5	11.9%	1	>5	.2%	CPU BOTTLENECK OR SCHEDULER SLOW
% SWAP	SPC LEFT	33	<30	46.1%	78	<30	.1%	SWAP SPACE LOW: ALLOCATE MORE
# JOBS	TTY IO Q	17	>16	64.1%	1	>16	.0%	INTERACTIVE USE HIGH: CHK RESPONSE
USER DSK	BLKS/SEC	79	>100	26.0%	30	>100	9.2%	DISK I/O RATE HIGH: CHK PACK I/O
% CPU	UTIL	59%	>70%	37.9%	13%	>70%	3.2%	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 (R)	SAT
% LOST	TIME	>2%						29.6%	
USER	UUOS/SEC	>200		49.9%	44.7%	68.9%	57.4%		
# JOBS	BLK IO Q	>1.5			40.2%				
%PK DEVPO	FREE SPC	<10%		68.9%	56.1%	100.0%	11.7%		
%PK HRS20	FREE SPC	<10%						100.0%	
% SWAP	SPC LEFT	<30					93.5%	78.8%	
# JOBS	TTY IO Q	>16		63.1%	62.6%	58.5%	69.1%	66.7%	
USER DSK	BLKS/SEC	>100					43.0%		

SUMMARY OF PROBLEM PERIODS

--- NON-PRIME TIME ---

ITEM		LONG TERM LIMIT	% TIME OVER LIMIT SUN	MON	TUE	WED	THU	FRI	SAT
%PK DEVPO	FREE SPC	<10%	89.5%		45.3%	78.5%	82.4%		
%PK HRS20	FREE SPC	<10%					17.6%	100.0%	

* = 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.

A prime/non-prime time summary for the month and each week in the month of utilization and performance items such as idle time, overhead, swapping rate, disk I/O rate, 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 "Program Name Report", "Batch Vs. Timesharing Report", or the "PPN Report". 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 (25-JAN-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 the top of the CPU utilization (D) and overhead (E) bars represents CPU time consumed by user jobs, lost time, and some embedded priority interrupt (PI) 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 eleven 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.

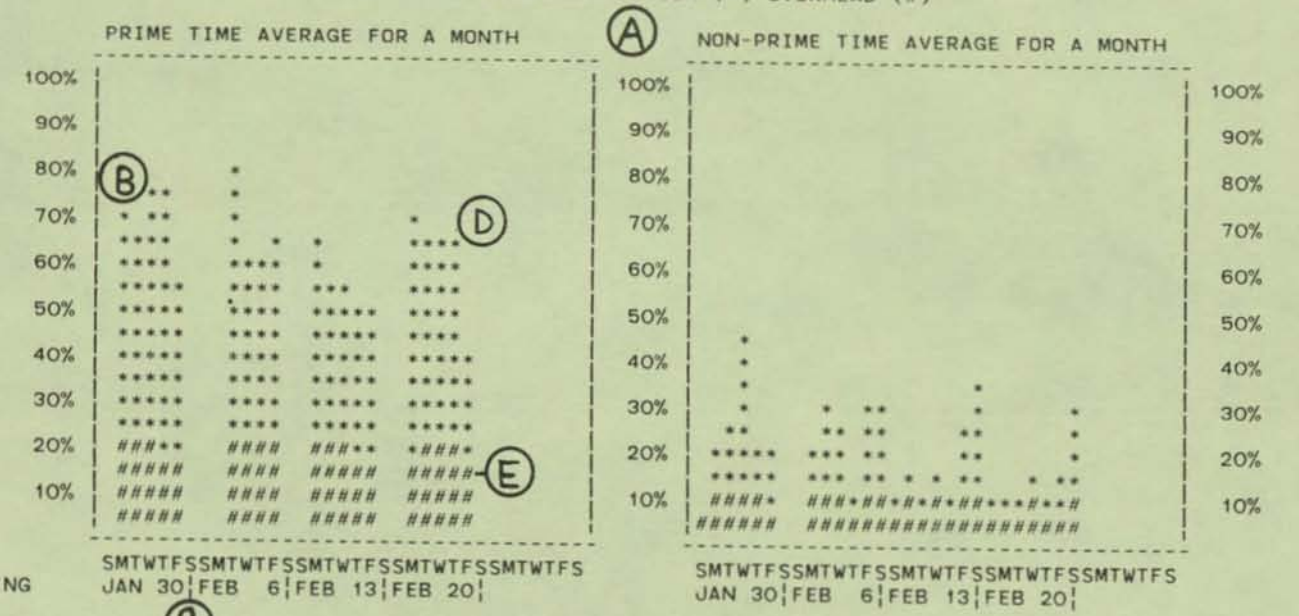
In addition to reporting which items had problems over the past month, this section also reports during which weeks the items spent the most time in a problem state. Again the item is given along with the percent of time that the item's values were over the long term limit during each week (T) in the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

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY UTILIZATION SUMMARY REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

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



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

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

F

-------AVERAGE-----	% CPU UTIL	# JOBS LOGGED IN	# LINES IN USE	% IDLE TIME	% LOST TIME	% OVHD TIME	ACT SWAP RATIO	SWAPPING BLKS/SEC	USER DSK BLKS/SEC	# MTAS ASSIGNED
----PRIME TIME-----	60%	59	36	40%	1%	15%	.2	33	72	0
---NON-PRIME TIME--	18%	24	5	82%	0%	6%	.0	1	32	0

-------AVERAGE-----	PGS USER MEMORY	% SYSTEM UPTIME	% AMAR CLK TIME	# SYSTEM RELOADS
----PRIME TIME-----	1736	94.7%	93.1%	6
---NON-PRIME TIME--	1735	83.9%	82.7%	9

G H I

Figure 1-4

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

--- PRIME TIME ---

KEY UTILIZATION ITEMS		AVERAGE OF CURRENT -MONTH-	WEEKLY AVERAGE -JAN 30-	-FEB 6-	-FEB 13-	-FEB 20-
% CPU	UTIL	60%	65%	64%	54%	59%
# JOBS	LOGGED IN	59	62	62	56	59
# LINES	IN USE	36	37	36	35	35
% IDLE	TIME	40%	35%	36%	46%	41%
% LOST	TIME	1%	1%	1%	1%	1%
% OVHD	TIME	15%	15%	17%	15%	15%
ACT SWAP	RATIO	.2	.2	.2	.2	.2
SWAPPING	BLKS/SEC	33	41	35	25	33
USER DSK	BLKS/SEC	72	69	67	71	79
# MTAS	ASSIGNED	0	0	1	1	0
PGS USER	MEMORY	1736	1737	1733	1736	1735
% SYSTEM	UPTIME	94.7%	99.9%	79.0%	99.9%	99.9%
% AMAR	CLK TIME	93.1%	98.7%	77.1%	98.0%	98.8%
# SYSTEM	RELOADS	6	2	2	1	1

--- PRIME TIME ---

OTHER UTILIZATION ITEMS		AVERAGE OF CURRENT -MONTH-	WEEKLY AVERAGE -JAN 30-	-FEB 6-	-FEB 13-	-FEB 20-
USER	UUOS/SEC	253 *	245 *	287 *	239 *	249 *
# JOBS	BLK IO Q	1.3	1.2	1.5 *	1.3	1.0
%PK DEVPO	FREE SPC	26%	47%	38%	11%	9%*
# JOBS	IN RUN Q	3	3	3	2	2
% SWAP	SPC LEFT	39	38	36	47	33
# JOBS	TTY IO Q	17 *	17 *	16 *	16 *	17 *

--- NON-PRIME TIME ---

KEY UTILIZATION ITEMS		AVERAGE OF CURRENT -MONTH-	WEEKLY AVERAGE -JAN 30-	-FEB 6-	-FEB 13-	-FEB 20-
% CPU	UTIL	18%	19%	24%	16%	13%
# JOBS	LOGGED IN	24	24	25	23	24
# LINES	IN USE	5	5	6	5	5
% IDLE	TIME	82%	81%	76%	84%	87%
% LOST	TIME	0%	0%	0%	0%	0%
% OVHD	TIME	6%	6%	7%	6%	5%
ACT SWAP	RATIO	.0	.1	.0	.0	.0
SWAPPING	BLKS/SEC	1	1	1	1	1
USER DSK	BLKS/SEC	32	25	43	30	30
# MTAS	ASSIGNED	0	0	0	0	0
PGS USER	MEMORY	1735	1737	1733	1736	1735
% SYSTEM	UPTIME	83.9%	72.6%	70.3%	96.3%	96.4%

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

Figure 1-4 (continued)

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY UTILIZATION DETAIL REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

--- NON-PRIME TIME ---

KEY UTILIZATION ITEMS	AVERAGE OF CURRENT -MONTH-	WEEKLY AVERAGE -JAN 30-	-FEB 6-	-FEB 13-	-FEB 20-
% AMAR CLK TIME	82.7%	71.9%	69.6%	94.5%	95.0%
# SYSTEM RELOADS	9	1	1	3	4

--- NON-PRIME TIME ---

OTHER UTILIZATION ITEMS	AVERAGE OF CURRENT -MONTH-	WEEKLY AVERAGE -JAN 30-	-FEB 6-	-FEB 13-	-FEB 20-
%PK DEVPO FREE SPC	25%	51%	36%	13%	9%*

Figure 1-4 (continued)

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

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY UTILIZATION PROBLEM REPORT

PUT ANY TITLE HERE
SYSTEM: PATH 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
% IDLE	TIME	40%	<30%	38.8% (N)	82%	<30%	4.9%	CPU PRESSED: CHK WORKLD DATA FIRST
% LOST	TIME	1%	>2%	17.0%	0%	>2%	.7%	MEMORY SHORTAGE OR SWAP DEVICE SLOW
% OVHD	TIME	15%	>20%	16.7%	6%	>20%	.3%	OVERHEAD TOO HIGH: INVESTIGATE
USER	UUOS/SEC	253 (M)	>200	53.0%	51	>200	6.5%	USER PGM PROBLEM: CHK WORKLD DATA
# JOBS	BLK IO Q	1.3	>1.5	33.4%	.3	>1.5	5.5%	BACKUP OF I/O JOBS: CHK CONTENTION
%PK DEVPO	FREE SPC	26%	<10%	26.3%	25%	<10%	25.3%	DELETE UNNECESSARY FILES
# JOBS	IN RUN Q	3	>5	13.1%	1	>5	.3%	CPU BOTTLENECK OR SCHEDULER SLOW
% SWAP	SPC LEFT	39	<30	30.9%	79	<30	.0%	SWAP SPACE LOW: ALLOCATE MORE
# JOBS	TTY IO Q	17	>16	60.7%	1	>16	.0%	INTERACTIVE USE HIGH: CHK RESPONSE
USER DSK	BLKS/SEC	72	>100	23.7%	32	>100	11.3%	DISK I/O RATE HIGH: CHK PACK I/O
% CPU	UTIL	60%	>70%	38.8%	18%	>70%	4.9%	CPU PRESSED: CHK WORKLD DATA FIRST

SUMMARY OF PROBLEM PERIODS

--- PRIME TIME ---

ITEM		LONG TERM LIMIT	% TIME OVER LIMIT (T) JAN 30	FEB 6	FEB 13	FEB 20
USER	UUOS/SEC	>200	54.1%	60.5%	49.8%	49.3%
# JOBS	BLK IO Q	>1.5		41.8%		
%PK DEVPO	FREE SPC	<10%				46.2%
# JOBS	TTY IO Q	>16	65.5%	56.6%	55.7%	64.1%

SUMMARY OF PROBLEM PERIODS

--- NON-PRIME TIME ---

ITEM		LONG TERM LIMIT	% TIME OVER LIMIT JAN 30	FEB 6	FEB 13	FEB 20
%PK DEVPO	FREE SPC	<10%				41.0%

* = 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. 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 on page 3 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 DEC 12 (G) was 8 weeks prior to the most recent week listed on the report, FEB 6 (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. If you want a table of other items, you must define it in the RFD file.

The actual trend analysis information begins on page 3 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 (NOV 28 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 average jobsize (M) from week to week while the values for user UUO's per second (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 average jobsize has no relationship to an 8 for user UUO's per second 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), FEB 6 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%.

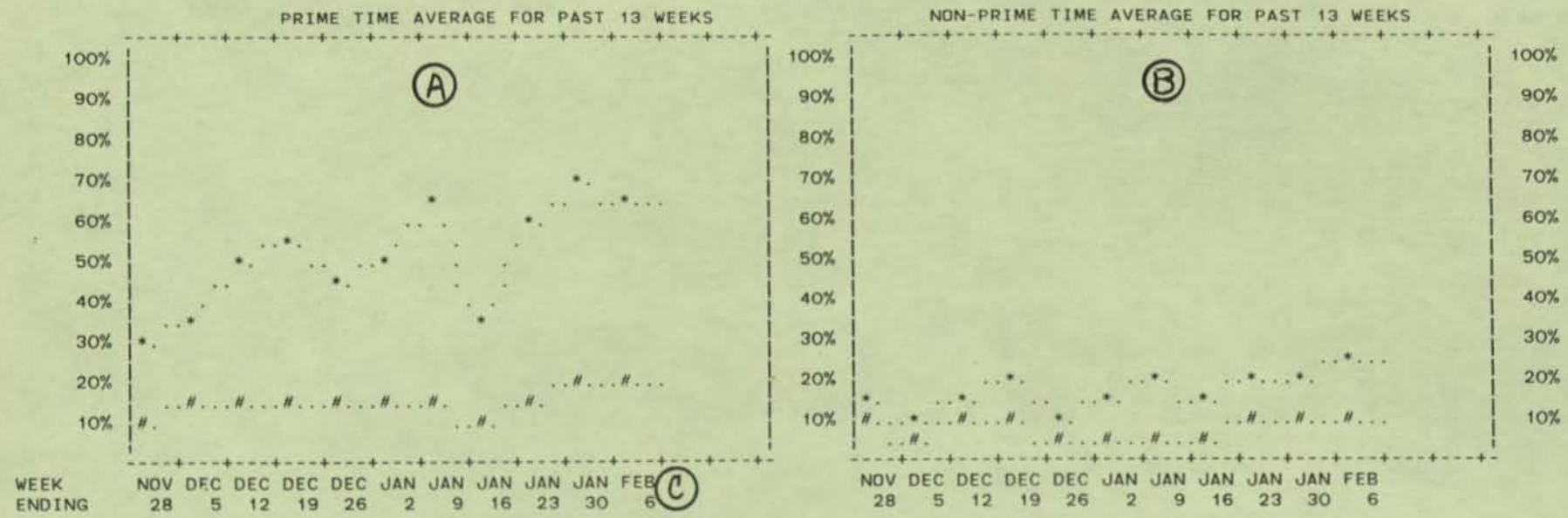
There are no predictions given based on weekly data. Predictions are found only on the Monthly Trend Analysis Report.

FROM: 22-NOV-81 (SUNDAY)
THRU: 06-FEB-82 (SATURDAY)

- AMAR -
WEEKLY TREND ANALYSIS REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

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



PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL WEEK ENDING		Q2M3W1 NOV 28	Q2M3W2 DEC 5	Q2M3W3 DEC 12	Q2M3W4 DEC 19	Q2M3W5 DEC 26	Q3M1W1 JAN 2	Q3M1W2 JAN 9	Q3M1W3 JAN 16	Q3M1W4 JAN 23	Q3M2W1 JAN 30	Q3M2W2 FEB 6	LONG TERM LIMIT
ITEM		W-10	W-09	W-08	W-07	W-06	W-05	W-04	W-03	W-02	W-01	W-00	
% CPU	UTIL	28	32	47	53	45	48	62	31	55	65	64	>70%
% IDLE	TIME	72	68	53	47	55	52	38	69	45	35	36	<30%
% OVHD	TIME	10	11	12	13	12	13	13	10	15	15	17	>20%
% LOST	TIME	2*	1	1	0	0	0	0	1	1	1	1	>2%
USER	UUOS/SEC	143	183	193	207*	165	194	935*	166	240*	245*	287*	>200
CONTEXT	SWTS/SEC					61	67	66	42	80	80	91	
ACT SWAP	RATIO	.1	.1	.1	.1	.1	.1	.1	.1	.2	.2	.2	>.5
ACTV JOB	% USR CR					9	12	10	8	18	20	19	
ALL JOBS	% USR CR					61	66	71	64	94	100	102	
% RN JOB	IN MEM					100	100	100	100	100	100	100	
AVG JOB	SIZE	36	32	33	31	34	36	36	35	39	39	40	
# JOBS	LOGGED IN	35	36	39	40	42	40	44	40	54	62	62	
PGS USER	MEMORY	1228	1230	1339	1743	1741	1736	1739	1740	1737	1737	1733	<768
# JOBS	TTY IO Q	7	6	7	8	5	6	6	6	13	17*	16*	>16
USER DSK	BLKS/SEC	32	39	53	64	51	59	78	44	59	69	67	>100
# JOBS	IN RUN Q	1	1	2	2	2	2	2	1	2	3	3	>5
SWAPPING	BLKS/SEC	21	15	18	7	5	6	10	8	35	41	35	>160
PK DSKEO	SWPS/SEC	21	15	18	7	5	6	10	8	35	41	35	>160

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

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

- AMAR -

FROM: 22-NOV-81 (SUNDAY)
THRU: 06-FEB-82 (SATURDAY)

WEEKLY TREND ANALYSIS REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL WEEK ENDING		Q2M3W1 NOV 28	Q2M3W2 DEC 5	Q2M3W3 DEC 12	Q2M3W4 DEC 19	Q2M3W5 DEC 26	Q3M1W1 JAN 2	Q3M1W2 JAN 9	Q3M1W3 JAN 16	Q3M1W4 JAN 23	Q3M2W1 JAN 30	Q3M2W2 FEB 6	LONG TERM
ITEM		W-10	W-09	W-08	W-07	W-06	W-05	W-04	W-03	W-02	W-01	W-00	LIMIT
AVG SCHD	RSP TIME	6	4	7	5	3	4	5	4	10	28	10	>200
# MTAS	ASSIGNED	0	0	0	0	1	1	0	0	0	0	1	
# LINES	IN USE	30	30	32	28	22	18	21	19	31	37	36	
%TY CHNK	IN USE					36	39	38	38	40	44	40	
SCN INTR	RCVS/SEC					4	9	9	8	11	11	11	
SCN INTR	XMTS/SEC					116	239	212	193	257	282	252	
% SYSTEM	UPTIME	87.7	98.9	96.6	84.6	100.0	99.7	99.8	77.4	92.3	99.9	79.0	
% AMAR	CLK TIME	86.2	90.9	94.1	78.7	96.6	95.1	97.8	61.7	84.8	98.7	77.1	
# SYSTEM	RELOADS	3	6	6	3	0	3	3	6	10	2	2	

NON-PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL WEEK ENDING		Q2M3W1 NOV 28	Q2M3W2 DEC 5	Q2M3W3 DEC 12	Q2M3W4 DEC 19	Q2M3W5 DEC 26	Q3M1W1 JAN 2	Q3M1W2 JAN 9	Q3M1W3 JAN 16	Q3M1W4 JAN 23	Q3M2W1 JAN 30	Q3M2W2 FEB 6	LONG TERM
ITEM		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	12	9	13	18	8	13	20	11	17	19	24	>70%
% IDLE	TIME	88	91	87	82	92	87	80	89	83	81	76	<30%
% OVHD	TIME	5	4	5	5	4	5	5	5	6	6	7	>20%
% LOST	TIME	0	0	0	0	0	0	0	0	0	0	0	>2%
USER	UUOS/SEC	50	44	49	50	31	50	757*	44	56	45	64	>200
CONTEXT	SWTS/SEC					9	16	10	14	23	23	36	
ACT SWAP	RATIO	.0	.0	.0	.0	.0	.0	.0	.0	.0	.1	.0	>.5
ACTV JOB	% USR CR					2	4	3	3	4	5	6	
ALL JOBS	% USR CR					31	30	29	27	28	31	35	
% RN JOB	IN MEM					100	100	100	100	100	100	100	
AVG JOB	SIZE	25	23	21	22	25	26	25	25	25	25	28	
# JOBS	LOGGED IN	23	22	24	22	24	23	22	22	23	24	25	
PGS USER	MEMORY	1298	1229	1444	1743	1741	1738	1739	1740	1738	1737	1733	<768
# JOBS	TTY IO Q	1	1	1	0	0	0	1	1	1	1	1	>16
USER DSK	BLKS/SEC	28	17	27	33	16	31	17	23	37	25	43	>100
# JOBS	IN RUN Q	1	1	1	1	1	1	1	1	1	1	1	>5
SWAPPING	BLKS/SEC	0	0	0	0	0	0	0	0	0	1	1	>160
ÞK DSKEO	SWPS/SEC	0	0	0	0	0	0	0	0	0	1	1	>160
AVG SCHD	RSP TIME	3	1	2	3	2	10	2	2	2	1	3	>200
# MTAS	ASSIGNED	0	0	0	0	0	0	0	0	0	0	0	
# LINES	IN USE	19	18	18	17	6	5	5	4	4	5	6	
%TY CHNK	IN USE					34	35	35	36	36	37	35	
SCN INTR	RCVS/SEC					0	0	1	0	0	0	1	
SCN INTR	XMTS/SEC					6	8	11	12	17	18	23	
% SYSTEM	UPTIME	85.8	96.5	70.4	85.8	97.4	99.9	96.9	82.8	92.3	72.6	70.3	
% AMAR	CLK TIME	83.8	94.7	69.8	85.8	96.2	99.4	95.4	80.8	87.7	71.9	69.6	
# SYSTEM	RELOADS	4	3	2	1	3	1	3	10	11	1	1	

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

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

Figure 1-5 (continued)

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

PRIME TIME TREND OF KEY UTILIZATION ITEMS

ITEM		(K) FIRST WEEKLY AVG	(L) TABLE OF RELATIVE USAGE PER WEEK	(P) LAST WEEKLY AVG	--11 WEEK-- TREND -- GROWTH /MONTH	LINE FIT
% CPU	UTIL	28	[4 5 7 8 7 7 10 5 9 10 10]	64	ERRATIC VALUES	
% IDLE	TIME	72	[11 10 8 7 8 8 6 10 7 5 5]	36	ERRATIC VALUES	
% OVHD	TIME	10	[6 6 7 8 7 8 8 6 9 9 10]	17	ERRATIC VALUES	
% LOST	TIME	2	[14 9 9 4 3 3 4 6 12 10 10]	1	ERRATIC VALUES	
USER	UUOS/SEC (N)	143	[4 5 5 6 4 5 25 4 6 7 8]	287	ERRATIC VALUES	
CONTEXT	SWTS/SEC	61	[7 7 7 4 9 9 10]	91	ERRATIC VALUES	
ACT SWAP	RATIO	.1	[6 7 8 6 5 6 6 5 10 11 10]	.2	ERRATIC VALUES	
ACTV JOB	% USR CR	9	[5 6 5 4 10 11 10]	19	+8.62	70%
ALL JOBS	% USR CR	61	[6 6 7 6 9 9 10]	102	+33.28	82%
% RN JOB	IN MEM	100	[8 8 8 8 8 8 8]	100	-0.35	94%
AVG JOB	SIZE (M)	36	[8 7 7 7 7 8 8 8 9 8 9]	40	ERRATIC VALUES	
# JOBS	LOGGED IN	35	[6 6 7 7 7 7 7 9 10 10]	62	+11.05	78%
PGS USER	MEMORY	1228	[6 6 6 8 8 8 8 8 8 8 8]	1733	ERRATIC VALUES	
# JOBS	TTY IO Q	7	[6 5 6 6 4 5 5 5 11 14 13]	16	ERRATIC VALUES	
USER DSK	BLKS/SEC	32	[4 5 7 9 7 8 10 6 8 9 9]	67	ERRATIC VALUES	
# JOBS	IN RUN Q	1	[5 6 7 8 7 7 8 5 9 10 11]	3	ERRATIC VALUES	
SWAPPING	BLKS/SEC	21	[8 6 7 3 2 2 4 3 14 17 14]	35	ERRATIC VALUES	
PK DSKEQ	SWPS/SEC	21	[8 6 7 3 2 2 4 3 14 17 14]	35	ERRATIC VALUES	
AVG SCHD	RSP TIME	6	[5 4 7 4 3 4 5 4 9 25 9]	10	ERRATIC VALUES	
# MTAS	ASSIGNED	0	[7 5 8 6 12 13 5 3 8 8 14]	1	ERRATIC VALUES	
# LINES	IN USE	30	[8 8 9 8 6 5 6 5 9 10 10]	36	ERRATIC VALUES	
%TY CHNK	IN USE	36	[7 7 7 7 8 8 8]	40	ERRATIC VALUES	
SCN INTR	RCVS/SEC	4	[3 7 7 6 9 9 9]	11	+4.42	70%
SCN INTR	XMTS/SEC	116	[4 8 7 6 9 9 8]	252	ERRATIC VALUES	
% SYSTEM	UPTIME	87.7	[7 8 8 7 8 8 8 6 8 8 7]	79.0	ERRATIC VALUES	
% AMAR	CLK TIME	86.2	[8 8 8 7 9 8 9 5 8 9 7]	77.1	ERRATIC VALUES	
# SYSTEM	RELOADS	3	[6 12 12 6 0 6 6 12 19 4 4]	2	ERRATIC VALUES	

NON-PRIME TIME TREND OF KEY UTILIZATION ITEMS

ITEM		FIRST WEEKLY AVG	TABLE OF RELATIVE USAGE PER WEEK	LAST WEEKLY AVG	--11 WEEK-- TREND -- GROWTH /MONTH	LINE FIT
% CPU	UTIL	12	[7 5 7 9 4 7 10 6 9 10 13]	24	ERRATIC VALUES	
% IDLE	TIME	88	[8 8 8 7 8 8 7 8 7 7 7]	76	ERRATIC VALUES	
% OVHD	TIME	5	[8 7 8 8 6 8 7 7 9 9 11]	7	ERRATIC VALUES	
% LOST	TIME	0	[10 8 8 4 2 2 4 8 10 13 19]	0	ERRATIC VALUES	
USER	UUOS/SEC	50	[3 3 3 3 2 3 48 3 4 3 4]	64	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: 22-NOV-81 (SUNDAY)
THRU: 06-FEB-82 (SATURDAY)

- AMAR -
WEEKLY TREND ANALYSIS REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

NON-PRIME TIME TREND OF KEY UTILIZATION ITEMS

ITEM		FIRST WEEKLY AVG	TABLE OF RELATIVE USAGE PER WEEK	LAST WEEKLY AVG	--11 WEEK-- -- TREND -- GROWTH /MONTH	LINE FIT
CONTEXT	SWTS/SEC	9	[4 7 4 6 10 10 15]	36	+16.77	80%
ACT SWAP	RATIO	.0	[8 6 6 8 6 6 5 5 8 9 8]	.0	ERRATIC VALUES	
ACTV JOB	% USR CR	2	[5 8 6 6 8 10 11]	6	+2.08	73%
ALL JOBS	% USR CR	31	[8 8 7 7 7 8 9]	35	ERRATIC VALUES	
% RN JOB	IN MEM	100	[8 8 8 8 8 8 8]	100	ERRATIC VALUES	
AVG JOB	SIZE	25	[8 7 7 7 8 8 8 8 8 8 9]	28	ERRATIC VALUES	
# JOBS	LOGGED IN	23	[8 7 8 7 8 8 8 7 8 8 8]	25	ERRATIC VALUES	
PGS USER	MEMORY	1298	[6 6 7 8 8 8 8 8 8 8 8]	1733	ERRATIC VALUES	
# JOBS	TTY IO Q	1	[9 7 9 4 2 4 9 11 7 12 13]	1	ERRATIC VALUES	
USER DSK	BLKS/SEC	28	[8 5 8 9 5 9 5 7 11 7 12]	43	ERRATIC VALUES	
# JOBS	IN RUN Q	1	[7 7 7 8 7 8 8 7 8 9 9]	1	+0.04	99%
SWAPPING	BLKS/SEC	0	[9 7 7 4 2 2 4 6 8 16 24]	1	ERRATIC VALUES	
PK DSKEQ	SWPS/SEC	0	[9 7 7 4 2 2 4 6 8 16 24]	1	ERRATIC VALUES	
AVG SCHD	RSP TIME	3	[10 4 5 9 7 30 4 6 5 4 9]	3	ERRATIC VALUES	
# MTAS	ASSIGNED	0	[8 4 14 12 4 10 4 5 9 8 9]	0	ERRATIC VALUES	
# LINES	IN USE	19	[15 14 14 13 4 4 4 3 3 4 5]	6	-7.37	71%
%TY CHNK	IN USE	34	[7 8 8 8 8 8 8]	35	ERRATIC VALUES	
SCN INTR	RCVS/SEC	0	[2 2 13 8 8 7 14]	1	ERRATIC VALUES	
SCN INTR	XMTS/SEC	6	[4 5 6 7 10 10 13]	23	+11.48	97%
% SYSTEM	UPTIME	85.8	[8 9 6 8 9 9 9 7 8 6 6]	70.3	ERRATIC VALUES	
% AMAR	CLK TIME	83.8	[8 9 6 8 9 9 9 7 8 7 6]	69.6	ERRATIC VALUES	
# SYSTEM	RELOADS	4	[8 6 4 2 6 2 6 21 23 2 2]	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 SEP 26 (B) contained five weeks and was the third month in the first quarter (Q1M3W5).

The trend analysis section of the report, beginning on page 3 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. Listed first is a range of values where the item's average is expected to fall within 6 months (C) and then within 12 months (D). An underlying assumption in these predictions is that the current linear trend will continue. If that occurs, there is a 90% chance that the item's actual averages will fall within the predicted ranges. Only the past usage history of the machine is taken into account, not the future plans of the data center or its users. For example, in this sample report, LINES IN USE is expected to grow to between 28-56 (E) by the end of six months. If the data center were to move users to another machine, this number might fall. If several new applications were added, the number might increase more rapidly. Note that some items, such as # MTAS ASSIGNED, 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. Dates in this column (H) are given in the format yymmm where yy is the actual calendar <u>not</u> fiscal year and mmm is the month. The comment, ALREADY BY, means that the long term limit has already been exceeded. NONE LIKELY means that the trend line is 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.

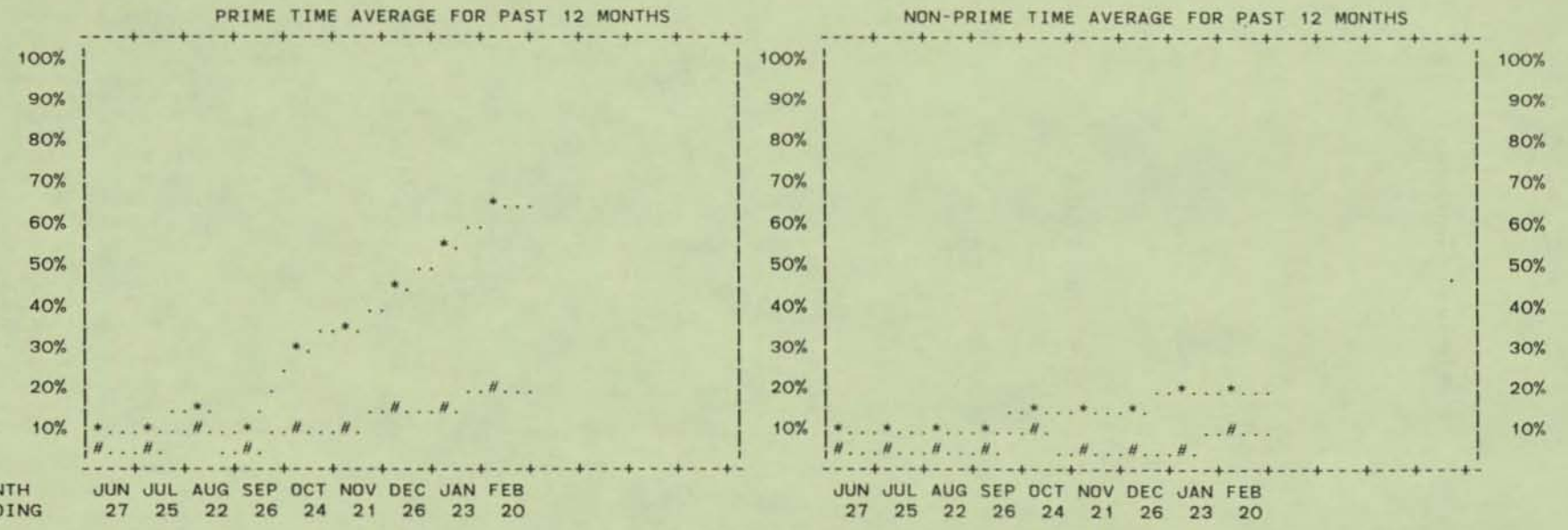
- AMAR -

FROM: 22-FEB-81 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

MONTHLY TREND ANALYSIS REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

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



PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL MONTH ENDING		Q4M3W5 JUN 27	Q1M1W4 JUL 25	Q1M2W4 AUG 22	Q1M3W5 SEP 26	Q2M1W4 OCT 24	Q2M2W4 NOV 21	Q2M3W5 DEC 26	Q3M1W4 JAN 23	Q3M2W4 FEB 20	LONG TERM LIMIT
ITEM		M-08	M-07	M-06	M-05	M-04	M-03	M-02	M-01	M-00	
% CPU	UTIL	7	8	12	7	27	33	42	51	60	>70%
% IDLE	TIME	93	92	88	93	73	67	58	49	40	<30%
% OVHD	TIME	4	4	5	4	9	10	11	13	15	>20%
% LOST	TIME	0	0	0	0	0	0	1	1	1	>2%
USER	UUOS/SEC	21	29	44	28	79	136	182	427*	253*	>200
CONTEXT	SWTS/SEC							61	65	83	
ACT SWAP	RATIO	.0	.0	.0	.0	.0	.1	.1	.1	.2	>.5
ACTV JOB	% USR CR							9	12	18	
ALL JOBS	% USR CR							61	74	100	
% RN JOB	IN MEM							100	100	100	
AVG JOB	SIZE	24	25	33	27	30	33	33	37	40	
# JOBS	LOGGED IN	23	21	28	24	32	34	39	45	59	
PGS USER	MEMORY	1757	1754	1745	1722	1742	1741	1445	1738	1736	<768
# JOBS	TTY IO Q	2	1	2	1	4	6	7	8	17*	>16
USER DSK	BLKS/SEC	14	15	19	5	29	41	49	62	72	>100
# JOBS	IN RUN Q	1	1	1	1	1	2	2	2	3	>5
SWAPPING	BLKS/SEC	0	0	0	0	2	4	13	15	33	>160
PK DSKEO	SWPS/SEC	0	0	0	0	2	4	13	15	33	>160

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

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

Figure 1-6

FROM: 22-FEB-81 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY TREND ANALYSIS REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL MONTH ENDING		Q4M3W5 JUN 27	Q1M1W4 JUL 25	Q1M2W4 AUG 22	Q1M3W5 SEP 26	Q2M1W4 OCT 24	Q2M2W4 NOV 21	Q2M3W5 DEC 26	Q3M1W4 JAN 23	Q3M2W4 FEB 20	LONG TERM
------- ITEM -------		M-08	M-07	M-06	M-05	M-04	M-03	M-02	M-01	M-00	LIMIT
PK PER10	SWPS/SEC		0								>160
AVG SCHD	RSP TIME	40	1	1	0	2	3	5	6	14	>200
# MTAS	ASSIGNED	0	0	0	0	0	1	0	0	0	
# LINES	IN USE	18	16	21	20	24	28	29	23	36	
%TY CHNK	IN USE							36	39	41	
SCN INTR	RCVS/SEC							4	9	11	
SCN INTR	XMTS/SEC							116	227	261	
% SYSTEM	UPTIME	99.8	96.8	84.3	94.7	99.3	95.6	93.5	91.9	94.7	
% AMAR	CLK TIME	91.8	95.8	82.0	94.4	98.4	91.2	88.9	84.3	93.2	
# SYSTEM	RELOADS	2	5	12	1	3	18	18	22	6	

NON-PRIME TIME AVERAGES OF KEY UTILIZATION ITEMS

FISCAL MONTH ENDING		Q4M3W5 JUN 27	Q1M1W4 JUL 25	Q1M2W4 AUG 22	Q1M3W5 SEP 26	Q2M1W4 OCT 24	Q2M2W4 NOV 21	Q2M3W5 DEC 26	Q3M1W4 JAN 23	Q3M2W4 FEB 20	LONG TERM
------- ITEM -------		M-08	M-07	M-06	M-05	M-04	M-03	M-02	M-01	M-00	LIMIT
% CPU	UTIL	6	7	8	9	12	11	12	15	18	>70%
% IDLE	TIME	94	93	92	91	88	89	88	85	82	<30%
% OVHD	TIME	4	4	4	4	6	5	5	5	6	>20%
% LOST	TIME	0	0	0	0	0	0	0	0	0	>2%
USER	UUOS/SEC	20	22	32	43	47	48	45	232*	51	>200
CONTEXT	SWTS/SEC							9	16	24	
ACT SWAP	RATIO	.0	.0	.0	.0	.0	.0	.0	.0	.0	>.5
ACTV JOB	% USR CR							2	3	4	
ALL JOBS	% USR CR							31	29	33	
% RN JOB	IN MEM							100	100	100	
AVG JOB	SIZE	23	24	29	28	24	24	23	25	26	
# JOBS	LOGGED IN	22	19	23	23	24	23	23	22	24	
PGS USER	MEMORY	1757	1753	1746	1737	1742	1742	1475	1739	1735	<768
# JOBS	TTY IO Q	2	1	1	0	1	1	1	1	1	>16
USER DSK	BLKS/SEC	12	14	20	21	23	24	24	27	32	>100
# JOBS	IN RUN Q	1	1	1	1	1	1	1	1	1	>5
SWAPPING	BLKS/SEC	0	0	0	0	0	0	0	0	1	>160
PK DSKEO	SWPS/SEC	0	0	0	0	0	0	0	0	1	>160
PK PER10	SWPS/SEC										>160
AVG SCHD	RSP TIME	1	1	2	2	1	1	2	3	2	>200
# MTAS	ASSIGNED	0	0	0	0	0	0	0	0	0	
# LINES	IN USE	18	15	17	18	19	19	16	5	5	
%TY CHNK	IN USE							34	36	36	
SCN INTR	RCVS/SEC							0	0	1	
SCN INTR	XMTS/SEC							6	12	20	
% SYSTEM	UPTIME	97.6	93.6	93.1	81.2	96.3	73.0	86.6	93.1	83.9	
% AMAR	CLK TIME	94.6	92.8	92.0	80.7	95.8	69.2	85.5	91.0	82.7	
# SYSTEM	RELOADS	6	9	14	10	8	13	13	25	9	

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

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

Figure 1-6 (continued)

FROM: 22-FEB-81 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY TREND ANALYSIS REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

PRIME TIME TREND OF KEY UTILIZATION ITEMS

ITEM		FIRST MONTHLY AVG	TABLE OF RELATIVE USAGE PER MONTH	LAST MONTHLY AVG	09 MONTH TREND GROWTH /MONTH	09 MONTH TREND LINE FIT	PREDICTED RANGE OF VALUES WHERE AVERAGE WILL FALL IN 06 MONTHS (C)	12 MONTHS (D)	PREDICTED PERIOD WHEN LONG TERM LIMIT REACHED (G)	LONG TERM LIMIT
% CPU	UTIL	7	[2 2 3 2 8 10 13 15 18]	60	+7.14	92%	73-100		BY 82JUL	>70
% IDLE	TIME	93	[11 11 10 11 8 8 7 6 5]	40	-7.14	92%	0-26		BY 82JUL	<30
% OVHD	TIME	4	[4 4 5 4 8 9 11 12 15]	15	+1.46	92%	18-28		82MAR-82SEP	>20
% LOST	TIME	0	[0 0 1 1 4 8 19 16 25]	1	+0.15	85%	1-3		BY 82NOV	>2
USER	UUOS/SEC	21	[1 2 3 2 5 9 12 27 16]	253	+41.78	70%	239-863		BY 82JUN	>200
ACT SWAP	RATIO	.0	[3 3 3 3 5 8 13 14 20]	.2	+0.02	99%	.2-.3		83MAY-83JUN	>.5
AVG JOB	SIZE	24	[6 7 9 7 8 9 9 10 11]	40	+1.73	82%	39-58			
# JOBS	LOGGED IN	23	[6 5 7 6 8 8 9 11 15]	59	+4.15	85%	55-95			
PGS USER	MEMORY	1757	[9 9 9 8 9 9 7 8 8]	1736	ERRATIC VALUES					
# JOBS	TTY IO Q	2	[3 2 4 2 6 9 10 13 26]	17	+1.55	74%	10-31			<768
USER DSK	BLKS/SEC	14	[3 4 5 1 7 10 12 15 18]	72	+7.82	84%	72-151		BY 83FEB	>16
# JOBS	IN RUN Q	1	[6 6 6 6 7 9 10 11 14]	3	+0.18	86%	2-4		BY 83JAN	>100
SWAPPING	BLKS/SEC	0	[0 0 0 0 2 4 14 16 36]	33	+3.48	71%	17-68		82SEP-83DEC	>5
PK DSKEO	SWPS/SEC	0	[0 0 0 0 2 4 14 16 36]	33	+3.48	71%	17-68		84JAN-84FEB	>160
AVG SCHD	RSP TIME	40	[62 2 1 1 2 4 8 9 21]	14	ERRATIC VALUES				84JAN-84FEB	>160
# MTAS	ASSIGNED	0	[4 4 4 4 11 16 9 9 12] (F)	0	ERRATIC VALUES				(H)	>200
# LINES	IN USE	18	[6 6 7 7 8 10 10 8 12]	36	+1.87	71%	28-56 (E)			
% SYSTEM	UPTIME	99.8	[9 9 7 8 9 8 8 8 8]	94.7	ERRATIC VALUES					
% AMAR	CLK TIME	91.8	[8 9 7 9 9 8 8 8 9]	93.2	ERRATIC VALUES					
# SYSTEM	RELOADS	2	[2 4 10 1 3 16 16 19 5]	6	ERRATIC VALUES					

NON-PRIME TIME TREND OF KEY UTILIZATION ITEMS

ITEM		FIRST MONTHLY AVG	TABLE OF RELATIVE USAGE PER MONTH	LAST MONTHLY AVG	09 MONTH TREND GROWTH /MONTH	09 MONTH TREND 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	6	[4 5 6 7 9 8 9 12 13]	18	+1.34	92%	19-29			>70
% IDLE	TIME	94	[9 9 9 9 8 8 8 8 8]	82	-1.34	92%	71-81		AFTER 24 MO	<30
% OVHD	TIME	4	[7 7 8 8 10 9 8 9 10]	6	ERRATIC VALUES				AFTER 24 MO	>20
% LOST	TIME	0	[3 3 3 3 6 6 11 6 22]	0	+0.01	99%	0-0			>2
USER	UUOS/SEC	20	[3 3 4 6 6 7 6 31 7]	51	ERRATIC VALUES				AFTER 24 MO	>200
ACT SWAP	RATIO	.0	[8 8 8 8 8 11 11 8 14]	.0	ERRATIC VALUES					>.5
AVG JOB	SIZE	23	[8 8 10 9 8 8 8 8 9]	26	ERRATIC VALUES					
# JOBS	LOGGED IN	22	[8 7 9 9 9 8 8 8 9]	24	ERRATIC VALUES					
PGS USER	MEMORY	1757	[9 9 9 8 9 9 7 8 8]	1735	ERRATIC VALUES					
# JOBS	TTY IO Q	2	[15 9 11 5 7 8 6 7 9]	1	ERRATIC VALUES					<768
USER DSK	BLKS/SEC	12	[5 5 7 8 9 9 9 10 12]	32	+2.14	94%	36-50			>16
# JOBS	IN RUN Q	1	[8 8 8 8 8 8 9 9 9]	1	+0.02	99%	1-1		AFTER 24 MO	>100
									AFTER 24 MO	>5

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 1-6 (continued)

FROM: 22-FEB-81 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY TREND ANALYSIS REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

NON-PRIME TIME TREND OF KEY UTILIZATION ITEMS

ITEM		FIRST MONTHLY AVG	TABLE OF RELATIVE USAGE PER MONTH	LAST MONTHLY AVG	--09 MONTH-- 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
SWAPPING	BLKS/SEC	0	[1 2 3 3 8 8 10 7 31]	1	ERRATIC VALUES					
PK DSKEQ	SWPS/SEC	0	[1 2 3 3 8 8 10 7 31]	1	ERRATIC VALUES					>160
AVG SCHD	RSP TIME	1	[4 6 11 7 6 5 10 15 9]	2	ERRATIC VALUES					>160
# MTAS	ASSIGNED	0	[5 5 8 8 14 9 10 10 9]	0	+0.01	99%	0-0			>200
# LINES	IN USE	18	[10 8 10 11 11 11 9 3 3]	5	ERRATIC VALUES					
% SYSTEM	UPTIME	97.6	[9 9 9 8 9 7 8 9 8]	83.9	ERRATIC VALUES					
% AMAR	CLK TIME	94.6	[9 9 9 8 9 7 8 9 8]	82.7	ERRATIC VALUES					
# SYSTEM	RELOADS	6	[4 6 10 7 6 9 9 18 6]	9	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 1-6 (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 3 of the sample report, the value for #JOBS LOGGED IN (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 3, and #HOURS OVER LIMITS (F) also on page 3 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: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY 'TYPICAL DAY' SUMMARY REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

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

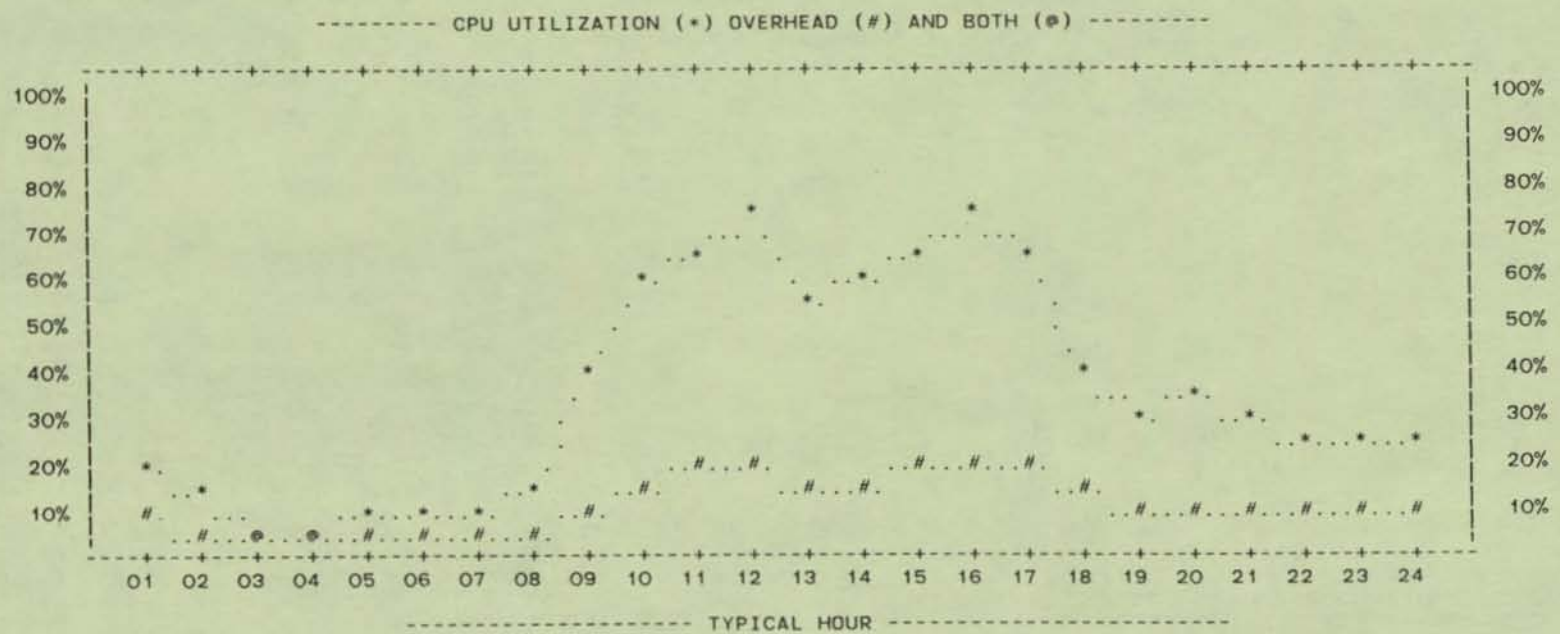


Figure 1-7

SUMMARY OF KEY UTILIZATION ITEMS

-------AVERAGE-----	# JOBS LOGGED IN	# LINES IN USE	% IDLE TIME	% LOST TIME	% OVHD TIME	ACT SWAP RATIO	SWAPPING BLKS/SEC	USER DSK BLKS/SEC	# MTAS ASSIGNED	PGS USER MEMORY	(D) NO. OF HOURS KEY ITEMS OVER LIMITS
----PRIME TIME-----	59	36	40	1	15	.2	33	72	0	1736	4
---NON-PRIME TIME--	24	5	83	0	6	.0	1	27	0	1735	

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY 'TYPICAL DAY' PROBLEM REPORT

PUT ANY TITLE HERE
SYSTEM: PATH 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		% IDLE TIME	PP	SERIOUS	CPU PRESSED: CHK WORKLD DATA FIRST
01:00 - 02:00		% LOST TIME	PP	SERIOUS	MEMORY SHORTAGE OR SWAP DEVICE SLOW
02:00 - 03:00		USER UUOS/SEC	PPPPPPPP	CRITICAL	USER PGM PROBLEM: CHK WORKLD DATA
03:00 - 04:00		# JOBS BLK IO Q	PPP	SERIOUS	BACKUP OF I/O JOBS: CHK CONTENTION
04:00 - 05:00		# JOBS TTY IO Q	PPPPPP	CRITICAL	INTERACTIVE USE HIGH: CHK RESPONSE
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					

SYSTEM AVAILABILITY SUMMARY

	SYSTEM UPTIME	% TIME MEASURED
PRIME TIME :	94.7%	93.2%
NON-PRIME TIME:	89.7%	87.7%

Figure 1-7 (continued)

- AMAR -

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

MONTHLY 'TYPICAL DAY' DETAIL REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

(E)

PERIOD	# JOBS LOGGED IN	# LINES IN USE	% IDLE TIME	% LOST TIME	% OVHD TIME	ACT SWAP RATIO	SWAPPING BLKS/SEC	USER DSK BLKS/SEC	# MTAS ASSIGNED	PGS USER MEMORY	# ITEMS OVER LIMITS: THIS PAGE	# ITEMS OVER LIMITS: ALL PAGES
	KEY UTILIZATION ITEMS											
00:00 - 01:00	23	5	84	0	6	.0	0	18	0	1735		
01:00 - 02:00	23	4	89	0	5	.0	0	11	0	1735		
02:00 - 03:00	22	4	96	0	3	.0	0	1	0	1735		
03:00 - 04:00	22	4	96	0	3	.0	0	0	0	1735		
04:00 - 05:00	22	4	93	0	4	.0	0	4	0	1735		
05:00 - 06:00	22	4	94	0	3	.0	0	0	0	1735		
06:00 - 07:00	22	4	92	0	4	.0	0	1	0	1735		
07:00 - 08:00	24	6	89	0	5	.0	2	11	0	1735		
08:00 - 09:00	40	20	64	1	10	.1	12	45	0	1735		
09:00 - 10:00	58 (C)	36	41	1	15	.2	31	95	0	1735		2
10:00 - 11:00	64	40	35	1	17	.2	36	92	1	1735		2
11:00 - 12:00	64	39	28 *	1	17	.2	38	84	1	1735	1	4
12:00 - 13:00	60	35	46	1	14	.2	11	56	0	1735		1
13:00 - 14:00	62	38	43	1	15	.2	29	57	0	1736		2
14:00 - 15:00	65	40	35	2 *	17	.2	43	69	0	1736	1	3
15:00 - 16:00	67	41	29 *	2 *	18	.2	62	72	1	1736	2	5
16:00 - 17:00	57	31	37	1	16	.2	38	78	0	1736		2
17:00 - 18:00	36	13	60	0	11	.1	3	70	0	1735		
18:00 - 19:00	29	8	71	0	8	.1	3	57	0	1735		
19:00 - 20:00	27	6	66	0	8	.1	1	75	0	1735		
20:00 - 21:00	25	5	70	0	8	.1	1	62	0	1735		
21:00 - 22:00	24	5	77	0	6	.1	2	38	0	1735		
22:00 - 23:00	24	5	75	0	7	.1	1	58	0	1735		
23:00 - 24:00	23	5	75	0	7	.1	2	30	0	1735		
----PRIME TIME-----												
AVERAGE VALUE:	59	36	40	1	15	.2	33	72	0	1736		
LONG TERM LIMIT:	NONE	NONE	<30%	>2%	>20%	>.5	>160	>100	NONE	<768		
% TIME OVER LIMIT:			38.8%	17.0%	16.7%			23.7%				
# HOURS OVER LIMIT: (F)			2	2							4	21
--NON-PRIME TIME---												
AVERAGE VALUE:	24	5	83	0	6	.0	1	27	0	1735		
LONG TERM LIMIT: (G)	NONE	NONE	<30%	>2%	>20%	>.5	>160	>100	NONE	<768		
% TIME OVER LIMIT:												
# HOURS OVER LIMIT:												

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

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

Figure 1-7 (continued)

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY 'TYPICAL DAY' DETAIL REPORT

PAGE: 4

PUT ANY TITLE HERE
SYSTEM: PATH   PRIME TIME: 0800 - 1700

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

PERIOD	USER UUOS/SEC	# JOBS BLK IO Q	%PK DEVPO FREE SPC	# JOBS IN RUN Q	% SWAP SPC LEFT	# JOBS TTY IO Q	# ITEMS OVER LIMITS THIS PAGE	# ITEMS OVER LIMITS ALL PAGES
00:00 - 01:00	43	.2	29	1	82	1		
01:00 - 02:00	27	.1	28	1	82	1		
02:00 - 03:00	7	.0	28	1	83	0		
03:00 - 04:00	7	.0	28	1	84	1		
04:00 - 05:00	14	.1	28	1	83	1		
05:00 - 06:00	7	.0	28	1	83	1		
06:00 - 07:00	10	.0	29	1	83	1		
07:00 - 08:00	44	.2	29	1	81	2		
08:00 - 09:00	154	.8	29	2	63	9		
09:00 - 10:00	229 *	1.3	28	2	43	16 *	2	2
10:00 - 11:00	259 *	1.4	27	3	37	18 *	2	2
11:00 - 12:00	277 *	1.5 *	26	3	36	18 *	3	4
12:00 - 13:00	241 *	.9	25	2	37	15	1	1
13:00 - 14:00	293 *	.9	24	2	35	18 *	2	2
14:00 - 15:00	303 *	1.4	23	3	31	19 *	2	3
15:00 - 16:00	284 *	1.5 *	23	3	31	20 *	3	5
16:00 - 17:00	244 *	1.5 *	25	3	38	15	2	2
17:00 - 18:00	136	1.0	25	2	60	5		
18:00 - 19:00	81	.5	24	1	69	2		
19:00 - 20:00	79	.6	24	1	72	1		
20:00 - 21:00	91	.5	25	1	75	1		
21:00 - 22:00	64	.3	27	1	78	1		
22:00 - 23:00	91	.4	26	1	79	1		
23:00 - 24:00	89	.3	25	1	80	1		
----PRIME TIME-----								
AVERAGE VALUE:	253	1.3	26	3	39	17		
LONG TERM LIMIT:	>200	>1.5	<10%	>5	<30	>16		
% TIME OVER LIMIT:	53.0%	33.4%	26.3%	13.1%	30.9%	60.7%		
# HOURS OVER LIMIT:	8	3				6	17	21
--NON-PRIME TIME---								
AVERAGE VALUE:	50	.3	27	1	79	1		
LONG TERM LIMIT:	>200	>1.5	<10%	>5	<30	>16		
% TIME OVER LIMIT:			21.9%					
# HOURS OVER LIMIT:								

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

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, channel usage, etc., is broken out from individual disk pack information such as mount time, blocks transferred per second, 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 blocks swapped per second system-wide (D).

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

The average busy time for each channel (F) expressed as percentages. (KL10 only.)

The average busy time for each priority interrupt level (G) expressed as percentages. (KL10 only.)

The average channel transfer wait queue length (H) for each disk channel on the system.

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 block 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: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY DISK REPORT

PUT ANY TITLE HERE
SYSTEM: PATH 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		170:27	412:59 (B)
HH:MM AMAR MEASURED THE SYSTEM		167:40	406:59 (C)
SWAPPING	BLKS/SEC	33	1 (D)
% SWAP	SPC LEFT	39	79 (E)
CHAN 0 %	TIM BUSY	1.67%	.22%
CHAN 1 %	TIM BUSY	.22%	.08%
CHAN 2 %	TIM BUSY	2.04%	.46%
CHAN 3 %	TIM BUSY	.44%	.33%
CHAN 4 %	TIM BUSY	.15%	.10% (F)
CHAN 5 %	TIM BUSY	.00%	.00%
CHAN 6 %	TIM BUSY	.00%	.00%
CHAN 7 %	TIM BUSY	.00%	.00%
PI 0 %	TIM BUSY	.15%	.01%
PI 1 %	TIM BUSY	.00%	.00%
PI 2 %	TIM BUSY	.15%	.14%
PI 3 %	TIM BUSY	.82%	.13% (G)
PI 4 %	TIM BUSY	1.84%	.47%
PI 5 %	TIM BUSY	.15%	.01%
PI 6 %	TIM BUSY	.00%	.00%
PI 7 %	TIM BUSY	6.69%	3.74%
CHAN 00	WAIT Q	.0	.0
CHAN 01	WAIT Q	.0	.0
CHAN 02	WAIT Q	.0	.0 (H)
CHAN 04	WAIT Q	.0	.0

PRIME TIME PACK SUMMARY

(I) PACK NAME	TOTAL TIME(HH:MM) MOUNTED (J) - IN USE (K)	% OF TIME MOUNTED (L) - IN USE (M)	% MOUNTED TIME IN USE (N)	(P) PACK FREE SPC	PACK BLKS/SEC	PACK SWPS/SEC	PACK WAIT Q
CASEO	166:47 - 155:26	99% - 93%	93%	31%	21		.1
COREO	166:47 - 79:20	99% - 47%	48%	45%	3		.0
DEVPO	166:47 - 88:44	99% - 53%	53%	26%	12		.0
DSKEO	166:47 - 166:47	99% - 99%	100%	33%	3	33	.0
DSKTO	44:46 - 5:09	27% - 3%	12%	100%	0		.0
DSKWO	166:47 - 135:42	99% - 81%	81%	77%	5		.0
HRS2O	36:54 - 6:26	22% - 4%	17%	13%	5		.0
PAWSO	65:16 - 13:36	39% - 8%	21%	89%	3		.0
PENSO	120:12 - 8:06	72% - 5%	7%	13%	0		.0
PER10	30:57 - 6:49	18% - 4%	22%	59%	6		.0
PER20	14:16 - 5:31	9% - 3%	39%	26%	18		.0
PER60	144:04 - 16:05	86% - 10%	11%	23%	1		.0
PLT10	165:13 - 79:15	99% - 47%	48%	42%	3		.0
STARO	160:03 - 79:00	95% - 47%	49%	42%	3		.0
TRNGO	57:23 - 3:45	34% - 2%	7%	74%	0		.0
TST10	11:59 - 0:30	7% - 0%	4%	68%	4		.0

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

Figure 1-8

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY DISK REPORT

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

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 BLKS/SEC	PACK SWPS/SEC	PACK WAIT Q
TST20	8:36 - 2:18	5% - 1%	27%	38%	29		.0
TST30	5:22 - 0:07	3% - 0%	2%	71%	0		.0
TST40	7:25 - 0:14	4% - 0%	3%	98%	0		.0
USREO	64:29 - 7:37	38% - 5%	12%	99%	0		.0
USRSO	166:47 - 165:59	99% - 99%	100%	23%	15		.1
YEARO	18:05 - 0:51	11% - 1%	5%	21%	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 BLKS/SEC	PACK SWPS/SEC	PACK WAIT Q
BKP20	23:18 - 12:18	6% - 3%	53%	52%	14		.0
BKP21	23:18 - 13:12	6% - 3%	57%	52%	14		.0
BKP22	23:18 - 13:29	6% - 3%	58%	53%	14		.0
CASEO	405:42 - 59:07	100% - 15%	15%	37%	5		.0
COREO	405:42 - 10:47	100% - 3%	3%	44%	0		.0
DEVPO	405:42 - 36:09	100% - 9%	9%	25%	3		.0
DSKEO	405:42 - 405:41	100% - 100%	100%	32%	1	1	.0
DSKTO	106:27 - 4:31	26% - 1%	4%	100%	0		.0
DSKWO	405:42 - 101:02	100% - 25%	25%	77%	4		.0
HRS10	20:03 - 0:31	5% - 0%	3%	87%	3		.0
HRS20	74:38 - 1:25	18% - 0%	2%	21%	1		.0
KLADO	1:52 - 0:17	0% - 0%	15%	39%	0		.0
PAWSO	166:39 - 3:47	41% - 1%	2%	80%	1		.0
PENSO	255:16 - 4:15	63% - 1%	2%	13%	2		.0
PER10	112:40 - 19:25	28% - 5%	17%	65%	5		.0
PER20	84:51 - 13:19	21% - 3%	16%	30%	8		.0
PER30	1:34 - 0:57	0% - 0%	61%	71%	61		.0
PER60	218:57 - 3:37	54% - 1%	2%	23%	1		.0
PER70	1:21 - 0:32	0% - 0%	40%	84%	38		.0
PERSO	5:31 - 1:24	1% - 0%	25%	35%	40		.0
PLT10	323:10 - 7:35	79% - 2%	2%	42%	0		.0
STARO	261:00 - 7:04	64% - 2%	3%	42%	1		.0
TRNGO	82:30 - 0:46	20% - 0%	1%	74%	1		.0
TST10	0:19 - 0:01	0% - 0%	5%	38%	0		.0
TST20	0:19 - 0:00	0% - 0%	0%	15%	0		.0
TST30	17:01 - 0:08	4% - 0%	1%	97%	0		.0
TST40	34:01 - 0:10	8% - 0%	0%	98%	0		.0
USREO	106:54 - 2:40	26% - 1%	2%	99%	0		.0
USRSO	405:41 - 211:43	100% - 52%	52%	23%	9		.0
YEARO	10:31 - 0:00	3% - 0%	0%	21%	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 not 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 BLKS/SEC (K), gives the transfer rate on each drive expressed in equivalent disk blocks (128 words per block) per second. This block size is independent of any actual physical block size on the tape. The data is expressed in equivalent disk blocks to make it easier to relate tape data to workload activity, for example, BACKUP usage where only the disk blocks read/written are given in the workload data.

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 never more than three drives in use at any one time (P). Two drives were always free.

In data 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: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY TAPE REPORT

PUT ANY TITLE HERE
SYSTEM: PATH 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	170:27	412:59 (B)
HH:MM AMAR MEASURED THE SYSTEM	167:40	406:59 (C)
# MTAS ASSIGNED	0	0 (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 BLKS/SEC (K)
MTA430	4:09 - 1:14	2% - 1%	30%	62
MTA431	8:55 - 1:18	5% - 1%	15%	28
MTA432	12:52 - 3:50	8% - 2%	30%	29
MTA433	34:53 - 11:17	21% - 7%	32%	20
MTA434	10:20 - 5:16	6% - 3%	51%	40

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	65%	65%
1	29%	94%
2	5%	99%
3	1% (P)	100%
4	0%	100%
5	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 BLKS/SEC
MTA430	2:28 - 2:16	1% - 1%	92%	80
MTA431	10:53 - 5:35	3% - 1%	51%	41
MTA432	25:30 - 21:21	6% - 5%	84%	87
MTA433	34:13 - 21:09	8% - 5%	62%	55
MTA434	12:01 - 5:57	3% - 1%	50%	65

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

Figure 1-9

FROM: 24-JAN-82 (SUNDAY)
THRU: 20-FEB-82 (SATURDAY)

- AMAR -
MONTHLY TAPE REPORT

PUT ANY TITLE HERE
SYSTEM: PATH 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	83%	83%
1	15%	97%
2	2%	99%
3	1%	100%
4	0%	100%
5	0%	100%

Figure 1-9 (continued)

blank page

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 <u>not</u> 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.

```
.RUN AMARON[,,AMAR]

DATABASE NAME: AMAR.DB
Histogram Function (Y/N): N
Output at (T)erminal or in (F)ile: T
Start Date: 820215
End Date: 820215
Granularity Level: H
Starting Hour: 07
Ending Hour: 17
(P)rime (N)on Prime (B)oth or (W)eekend: B
Item 1: TTYU
Item 2: TIOW
Item 3: CPUC
Item 4: CPIC
Item 5: LUIC?
LUIC         LUIOBKP10   LUIOBKP11   LUIOBKP12   LUIOBKP20   LUIOBKP21
LUIOBKP22    LUIOCASEC   LUIOCCREC   LUIODEVPC   LUIODSKEC   LUIODSKRC
LUIODSKTC    LUIODSKWC   LUIOFOO6C   LUIOFOO7C   LUIOHRS1C   LUIOHRS2C
LUIOKLADC    LUIOPAWSC   LUIOPENSC   LUIOPER1C   LUIOPER2C   LUIOPER3C
LUIOPER6C    LUIOPER7C   LUIOPERSC   LUIOPLT1C   LUIOPLT2C   LUIORP22C
LUIORP23C    LUIOSTARC   LUIOTRNGC   LUIOTST1C   LUIOTST2C   LUIOTST3C
LUIOTST4C    LUIOUSREC   LUIOUSRSC   LUIOYEARC
Item 5: LUIODSKEC
Item 6: LUIODSKRC
Item 7:

                              (A)                               (B)        (C)
                          SYSTEM PATH                 FROM 820215 TO 820215
 DATE  TIME           TTYU        TIOW         CPUC       CPICLUIODSKEC  LUIODSKRC
 (E)   (F)      (D)
820215:0700           4.00         .00         9.52       96.33       .08      -1.00
820215:0800           5.02         .72        48.84       89.77      1.13      -1.00
820215:0900          19.08        8.16       144.30       38.97      7.71      -1.00
820215:1000          31.95       17.22       140.67       38.17      1.22      -1.00
820215:1100          36.63       20.32       201.19       10.05      1.81      -1.00
820215:1200     (G)  32.22       12.66       172.91       35.44      1.52      -1.00
820215:1300          34.27       15.54       204.52       61.15      3.74      -1.00
820215:1400          37.78       19.92       597.12       47.59      1.35      -1.00
820215:1500          38.52       19.55       573.49       37.87      2.95      -1.00
820215:1600          46.29       26.95       470.13       18.84      2.64      -1.00
820215:1700          34.54       16.65       357.76       22.02      2.56      -1.00
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 user UUO's per second. Again refer to Appendix B for a definition of each item and its data type, i.e., percentage, blocks 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.

```
.run amaron[,,amar]

DATABASE NAME: amar
Histogram Function (Y/N): y
Output at (T)erminal or in (F)ile: t
Start Date: 820215
End Date: 820215
Granularity Level: h
Starting Hour: 15
Ending Hour: 15
(P)rime (N)on Prime (B)oth or (W)eekend: p
Item 1: cpu0
Item 2:
Suppress Blank Ranges (Y/N)? y
                          (A)                  (B)
                   SYSTEM: PATH        ITEM: CPU0 (D)           (E)
                  DAY 02/15/82 (C)   IN RANGE 02/15/82 TO 02/15/82
            PRIMETIME        TIME: 1500 (G)
                 (F)
                                     NUMBER OF     FREQUENCY %        % OF CUMULATIVE
         RANGE OF VALUES              SAMPLES       OF TOTAL               TOTAL

   50.01 -  (H)  100.00           (J)  1        (K) 1.6%          (L)      1.6%
  100.01 -       150.00                3            5.0%                   6.6%
  150.01 -       200.00                1            1.6%                   8.3%
  200.01 -       250.00                4            6.6%                  14.9%
  250.01 -       300.00                7           11.6%                  26.6% ——(M)
  300.01 -       350.00               14           23.3%                  49.9%
  350.01 -       400.00                6           10.0%                  59.9%
  400.01 -       450.00                5            8.3%                  68.3%
  450.01 -       500.00                4            6.6%                  74.9%
  500.01 -       550.00                3            5.0%                  79.9%
  600.01 -       650.00                1            1.6%                  81.6%
  650.01 -       700.00                2            3.3%                  84.9%
  750.01 -       800.00                1            1.6%                  86.6%
  800.01 -       850.00                1            1.6%                  88.3%
 1150.01 -      1200.00 ——— (I)        1            1.6%                  89.9%
 1350.01 -      1400.00                1            1.6%                  91.6%
 1550.01 -      1600.00                1            1.6%                  93.3%
 2050.01 -      2100.00                1            1.6%                  94.9%
 2100.01 -      2150.00                1            1.6%                  96.6%
 2750.01 -      2800.00                1            1.6%                  98.3%
 3500.01 -      3550.00                1            1.6%                  99.9%

          AVERAGE VALUE FOR            60 SAMPLES IS          573.49
                                      (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 user UUO's per second were less than or equal to 300 only 26.6% (M) of the time. Conversely, the values for user UUO's per second were greater than 300, 73.4% of the time (100%-26.6%).

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. CPU0, LUI0)	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

```
.run amarex[,,amar]

DATABASE NAME:amar
OUTPUT:extrc1.txt
RECORD TYPE:ps
DATE:820215:1301-820215:1500
[820215:1301-820215:1500 = 8232421301-8232421500]
GRANULARITY LEVEL:s
RESTRICTING ANY FISCAL PERIOD?no
ITEM:cpi0,cpo0,cpu0,nrjr,luiodske0,luiodskr0
PRIMETIME:all
[ EXTRACTING ]

[ SPECIFY NEXT EXTRACTION CRITERIA ]

OUTPUT:^C

.
.ty extrc1.txt
AAPS          PATH82021514008232421CPIO          P00000000600000004759
AAPS          PATH82021514008232421CPOO          P00000000600000001489
AAPS          PATH82021514008232421CPUO          P00000000600000059712
AAPS          PATH82021514008232421LUIODSKEO     P00000000600000000135
AAPS          PATH82021514008232421NRJR          P00000000600000000194
AAPS          PATH82021515008232421CPIO          P00000000600000003787
AAPS          PATH82021515008232421CPOO          P00000000600000001778
AAPS          PATH82021515008232421CPUO          P00000000600000057349
AAPS          PATH82021515008232421LUIODSKEO     P00000000600000000295
AAPS          PATH82021515008232421NRJR          P00000000600000000259
A B  C   D    E   F G H I J K ||||  Q       R    S     T       U     V
                              LMNOP
```

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., CPU0, LUI0)	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.

```
.run amarex[,,amar]

DATABASE NAME:amar
OUTPUT:extrc2.txt
RECORD TYPE:pd
DATE:820215:1401-820215:1500
[820215:1401-820215:1500 = 8232421401-8232421500]
GRANULARITY LEVEL:s
RESTRICTING ANY FISCAL PERIOD?no
ITEM:cpu0,luiodske0
PRIMETIME:all
[ EXTRACTING ]


[ SPECIFY NEXT EXTRACTION CRITERIA ]

OUTPUT:^C

.
.ty extrc2.txt
AAPD          PATH8202151500823242 1CPU0          P00000000010000008516
AAPD          PATH8202151500823242 1CPU0          P00000000030000014276
AAPD          PATH8202151500823242 1CPU0          P00000000010000018792
AAPD          PATH8202151500823242 1CPU0          P00000000040000023163
AAPD          PATH8202151500823242 1CPU0          P00000000070000028168
AAPD          PATH8202151500823242 1CPU0          P00000000140000032897
AAPD          PATH8202151500823242 1CPU0          P00000000060000037070
AAPD          PATH8202151500823242 1CPU0          P00000000050000043445
AAPD          PATH8202151500823242 1CPU0          P00000000040000048792
AAPD          PATH8202151500823242 1CPU0          P00000000030000052562
AAPD          PATH8202151500823242 1CPU0          P00000000010000064512
AAPD          PATH8202151500823242 1CPU0          P00000000020000066909
AAPD          PATH8202151500823242 1CPU0          P00000000010000076099
AAPD          PATH8202151500823242 1CPU0          P00000000010000082005
AAPD          PATH8202151500823242 1CPU0          P00000000010000116729
AAPD          PATH8202151500823242 1CPU0          P00000000010000139049
AAPD          PATH8202151500823242 1CPU0          P00000000010000159924
AAPD          PATH8202151500823242 1CPU0          P00000000010000209549
AAPD          PATH8202151500823242 1CPU0          P00000000010000212768
AAPD          PATH8202151500823242 1CPU0          P00000000010000278830
AAPD          PATH8202151500823242 1CPU0          P00000000010000354622
AAPD          PATH8202151500823242 1LUIODSKE0     P00000000570000000205
AAPD          PATH8202151500823242 1LUIODSKE0     P00000000020000001422
AAPD          PATH8202151500823242 1LUIODSKE0     P00000000010000003189
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.

```
.run amarex[,,amar]

DATABASE NAME:amar
OUTPUT:extrc3.txt
RECORD TYPE:gr
DATE:820215-820215
[820215:0001-820215:2400 = 8232420001-8232422400]
GRANULARITY LEVEL:s
RESTRICTING ANY FISCAL PERIOD?no
[ EXTRACTING ]


[ SPECIFY NEXT EXTRACTION CRITERIA ]

OUTPUT:^C

.
.ty extrc3.txt
AAGR          PATH8202150100823242 1
AAGR          PATH8202150200823242 1
AAGR          PATH8202150300823242 1
AAGR          PATH8202150400823242 1
AAGR          PATH8202150500823242 1
AAGR          PATH8202150600823242 1
AAGR          PATH8202150700823242 1
AAGR          PATH8202150800823242 1
AAGR          PATH8202150900823242 1
AAGR          PATH8202151000823242 1
AAGR          PATH8202151100823242 1
AAGR          PATH8202151200823242 1
AAGR          PATH8202151300823242 1
AAGR          PATH8202151400823242 1
AAGR          PATH8202151500823242 1
AAGR          PATH8202151600823242 1
AAGR          PATH8202151700823242 1
AAGR          PATH8202151800823242 1
AAGR          PATH8202151900823242 1
AAGR          PATH8202152000823242 1
AAGR          PATH8202152100823242 1
AAGR          PATH8202152200823242 1
AAGR          PATH8202152300823242 1
AAGR          PATH8202152400823242 1
```

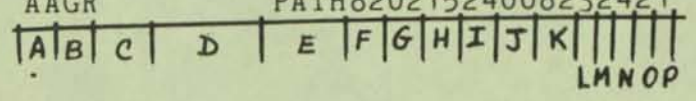


Figure 1-14

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.

```
.run amarex[,,amar]

DATABASE NAME:amar
OUTPUT:extrc4.txt
RECORD TYPE:sc
DATE:820215-820217
[820215:0001-820217:2400 = 8232420001-8232442400]
[ EXTRACTING ]


[ SPECIFY NEXT EXTRACTION CRITERIA ]

OUTPUT:^C

.
.ty extrc4.txt
AASC                 8202112318
0000000000                                                003889034284000000324627
AASC                 8202151730
0000000000                 D      E                       003889359005000000107284
A→AASC  K B    C     820216 2320          F
0000000000                                                003889466405 000000005689
     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 as an OPSER subjob. It should also be run under [1,2] in order to collect those variables obtained from the performance meter. See the Appendix called "System AMAR Item Definitions" for a list of such variables. There will be a corresponding subjob to collect data for Workload AMAR.

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

```
:SLOGIN 1,2
:DEFINE AMAR=
AMAR-RUN structure:xxxxDC[ppn]
```

These commands may be entered directly to OPSER 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.

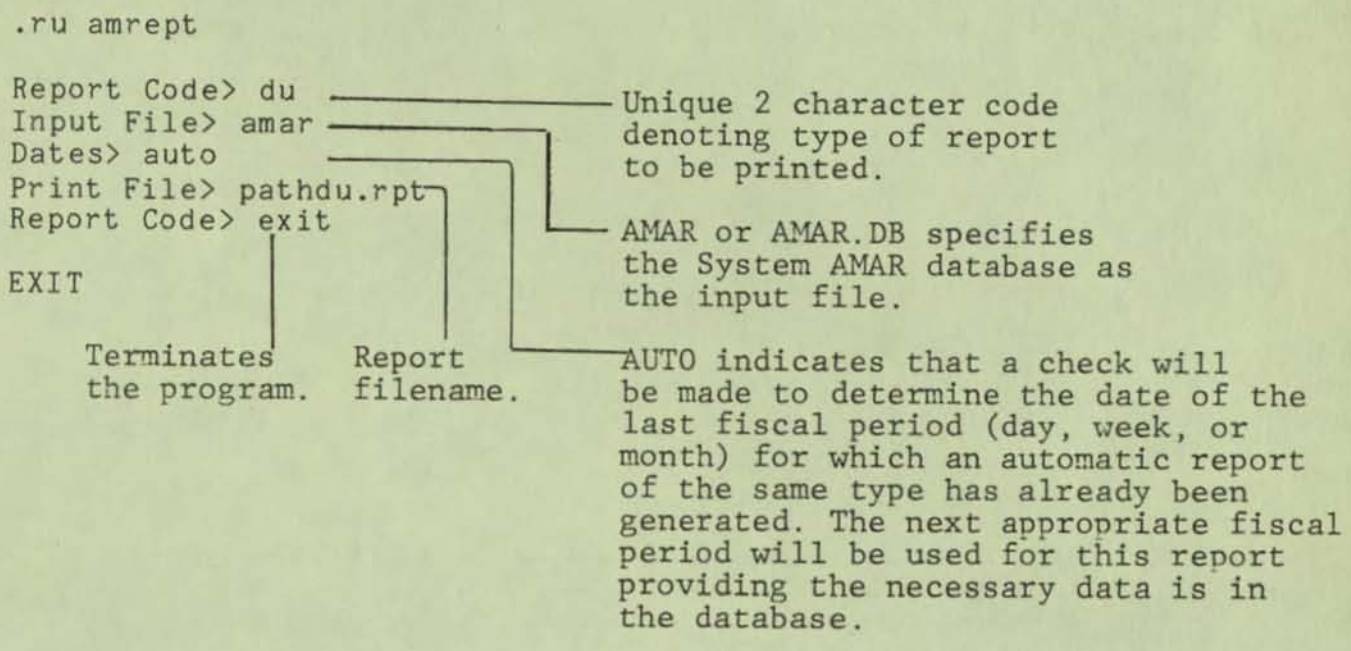


Figure 1-16
* * * * * * * * * *

1.4.3 Generating Special Reports - What Program Do I Use?

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

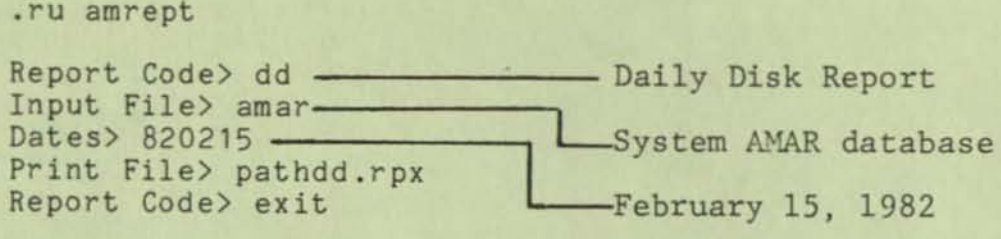
Use AMREPT alone to produce standard reports from the database. See Figure 1-17 and the Appendix called "Report Program (AMREPT) Dialogue".

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 which has not been entered into the database. 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".

* * * * * * * * *



Dialogue to Produce a Standard Report
Figure 1-17

* * * * * * * * *

FROM: 15-FEB-82 (MONDAY)
THRU: 15-FEB-82 (MONDAY)

- AMAR -
DAILY DISK REPORT

PUT ANY TITLE HERE
SYSTEM: PATH 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	14:58
HH:MM AMAR MEASURED THE SYSTEM	9:00	14:29
SWAPPING BLKS/SEC	35	0
% SWAP SPC LEFT	39	87
CHAN 0 % TIM BUSY	1.19%	.03%
CHAN 1 % TIM BUSY	.38%	.00%
CHAN 2 % TIM BUSY	1.58%	.30%
CHAN 3 % TIM BUSY	1.14%	.23%
CHAN 4 % TIM BUSY	.00%	.00%
CHAN 5 % TIM BUSY	.00%	.00%
CHAN 6 % TIM BUSY	.00%	.00%
CHAN 7 % TIM BUSY	.00%	.00%
PI 0 % TIM BUSY	.14%	.01%
PI 1 % TIM BUSY	.00%	.00%
PI 2 % TIM BUSY	.14%	.14%
PI 3 % TIM BUSY	.94%	.08%
PI 4 % TIM BUSY	1.25%	.21%
PI 5 % TIM BUSY	.14%	.01%
PI 6 % TIM BUSY	.00%	.00%
PI 7 % TIM BUSY	7.07%	3.65%
CHAN 00 WAIT Q	.0	.0
CHAN 01 WAIT Q	.0	.0
CHAN 02 WAIT Q	.0	.0
CHAN 04 WAIT Q	.0	.0

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 BLKS/SEC	PACK SWPS/SEC	PACK WAIT Q
CASE0	8:57 - 8:26	99% - 94%	94%	40%	15		.0
CORE0	8:57 - 2:56	99% - 33%	33%	46%	1		.0
DEVP0	8:57 - 4:10	99% - 46%	47%	9%	16		.0
DSKE0	8:57 - 8:57	99% - 99%	100%	29%	3	35	.0
DSKW0	8:57 - 7:46	99% - 86%	87%	78%	6		.0
HRS20	1:37 - 1:01	18% - 11%	63%	60%	31		.0
PENS0	8:57 - 0:45	99% - 8%	8%	13%	0		.0
PER60	3:39 - 1:04	41% - 12%	29%	23%	1		.0
PLT10	7:49 - 2:28	87% - 27%	32%	42%	1		.0
STAR0	8:57 - 2:06	99% - 23%	23%	41%	0		.0
TRNG0	8:13 - 0:38	91% - 7%	8%	74%	1		.0
USRS0	8:57 - 8:54	99% - 99%	99%	19%	13		.1

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

Figure 1-17 (continued)

* * * * * * * * *

```
.run amarsd

System ID>path
YYMMDD Date of file>820217
Prime periods>0800-1700

[ARMDSF Data segment full - storing in first free page]
[ARMDSF Data segment full - storing in first free page]

[AMIHDS Hourly data stored for 820217]
[AMIDUD Database up to date]
[Use TODAY.DB as input file to the AMREPT report program]

EXIT

..run amrept

Report Code> du
Input File> today.db
Dates> 820217
Print File> pathxx.rpx
Report Code> exit

EXIT
```

Raw file from February 17, 1982. The raw filename is PATH17.FEB.

Message indicating that preallocated storage has been used up. Secondary database storage is now being automatically used.

Mini-database created by AMARSD above and used as the input file here.

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

DATE: 17-FEB-82 (WEDNESDAY)

- AMAR -
DAILY SYSTEM UTILIZATION SUMMARY REPORT

PAGE: 1

PUT ANY TITLE HERE
SYSTEM: PATH PRIME TIME: 0800 - 1700

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

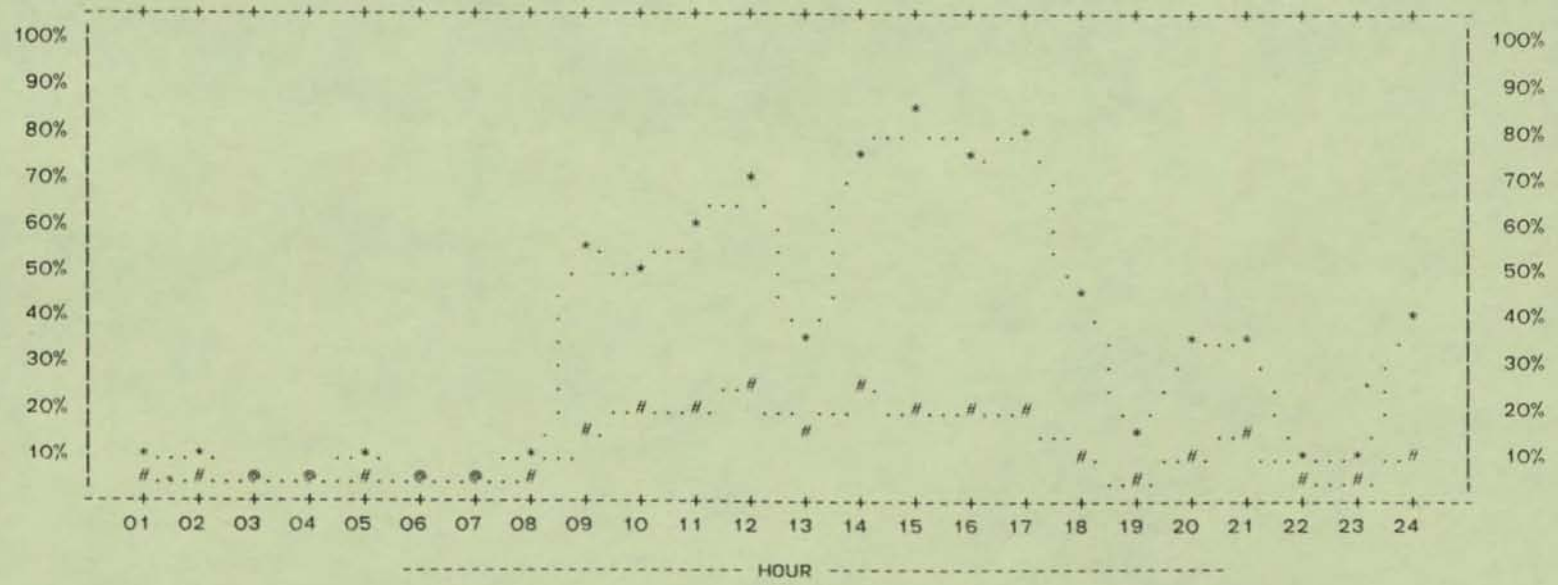


Figure 1-18 (continued)

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

-------AVERAGE-----	# JOBS LOGGED IN	# LINES IN USE	% IDLE TIME	% LOST TIME	% OVHD TIME	ACT SWAP RATIO	SWAPPING BLKS/SEC	USER DSK BLKS/SEC	# MTAS ASSIGNED	PGS USER MEMORY	NO. OF HOURS KEY ITEMS OVER LIMITS
----PRIME TIME-----	60	36	38	1	17	.2	33	80	1	1735	2
---NON-PRIME TIME--	25	5	85	0	5	.0	2	52	0	1735	2

---------- PERCENT FREE SPACE LEFT ON DISK PACKS ----------

-----PACK NAME-----	BKP20	BKP21	BKP22	CASEO	COREO	DEVPO	DSKEO	DSKTO	DSKWO	HRS20	PAWSO	PENSO	PER10	PER20	PER60	PLT10	STARO
---% FREE TODAY----	55	55	55	38	46	4	31	100	75	11	66	13	46	15	23	41	42
-% FREE YESTERDAY--																	
----DIFFERENCE-----																	

\+ = MORE THAN YESTERDAY - = LESS THAN YESTERDAY

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

blank page

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 3 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
.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. The example RFD in this Guide would cause the trend analysis sections to be supressed. Note that for databases created before Release 4.1 of AMAR, the trend analysis sections will be produced even if 03 is not specified.

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     PATH      _ PUT ANY TITLE HERE
.SS     *#@  10,10_WARNING   25,50_SERIOUS   50,80_CRITICAL  10,10_WARNING   25,50_SERIOUS   50,80_CRITICAL
.C      ******** ITEM DESCRIPTION SECTION ********
.ID     ACCR          _ ACTV JOB _ % USR CR _ NNNNNNR.%
.ID     ACSR          _ ACT SWAP _   RATIO  _ NNNNNN.R      >000000.7,>000000.5   CPU/MEMORY SHORTAGE: CHK AMOUNT
.ID     AVJS          _ AVG JOB  _   SIZE   _ NNNNNNR.                            JOB SIZES TOO BIG: CHK WORKLD DATA
.ID     AVRT          _ AVG SCHD _ RSP TIME _ NNNNNNR.      >0000200.,>0000200.   SCHEDULER SLOW: CHK HPQ USE, WORKLD
.ID     CH00          _ CHAN 0 % _ TIM BUSY _ NNNNN.NR%
.ID     CH01          _ SLV CHAN _ 0 % BUSY _ NNNNN.NR%
.ID     CH10          _ CHAN 1 % _ TIM BUSY _ NNNNN.NR%
.ID     CH11          _ SLV CHAN _ 1 % BUSY _ NNNNN.NR%
.ID     CH20          _ CHAN 2 % _ TIM BUSY _ NNNNN.NR%
.ID     CH21          _ SLV CHAN _ 2 % BUSY _ NNNNN.NR%
.ID     CH30          _ CHAN 3 % _ TIM BUSY _ NNNNN.NR%
.ID     CH31          _ SLV CHAN _ 3 % BUSY _ NNNNN.NR%
.ID     CH40          _ CHAN 4 % _ TIM BUSY _ NNNNN.NR%
.ID     CH41          _ SLV CHAN _ 4 % BUSY _ NNNNN.NR%
.ID     CH50          _ CHAN 5 % _ TIM BUSY _ NNNNN.NR%
.ID     CH51          _ SLV CHAN _ 5 % BUSY _ NNNNN.NR%
.ID     CH60          _ CHAN 6 % _ TIM BUSY _ NNNNN.NR%
.ID     CH61          _ SLV CHAN _ 6 % BUSY _ NNNNN.NR%
.ID     CH70          _ CHAN 7 % _ TIM BUSY _ NNNNN.NR%
.ID     CH71          _ SLV CHAN _ 7 % BUSY _ NNNNN.NR%
.ID     CPA0          _ % AMAR TI_ CPU 0 UP _ NNNNNNR.%  <0000090.,<0000095.   TOO MUCH CPU0 DOWNTIME: CHK REASON
.ID     CPA1          _ % AMAR TI_ CPU 1 UP _ NNNNNNR.%  <0000090.,<0000095.   TOO MUCH CPU1 DOWNTIME: CHK REASON
.ID     CPC0          _ CONTEXT  _ SWTS/SEC _ NNNNNNR.
.ID     CPC1          _ SLV CTXT _ SWTS/SEC _ NNNNNNR.
.ID     CPI0          _  % IDLE  _   TIME   _ NNNNNNR.%  <0000005.,<0000030.   CPU PRESSED: CHK WORKLD DATA FIRST
.ID     CPI1          _ SLV IDLE _   TIME   _ NNNNNNR.%  <0000005.,<0000030.   SLAVE CPU PRESSED: CHK WORKLD DATA
.ID     CPL0          _  % LOST  _   TIME   _ NNNNNNR.%  >0000005.,>0000002.   MEMORY SHORTAGE OR SWAP DEVICE SLOW
.ID     CPL1          _ SLV LOST _   TIME   _ NNNNNNR.%  >0000005.,>0000002.   MEMORY SHORTAGE OR SWAP DEVICE SLOW
.ID     CPO0          _  % OVHD  _   TIME   _ NNNNNNR.%  >0000020.,>0000020.   OVERHEAD TOO HIGH: INVESTIGATE
.ID     CPO1          _ SLV OVHD _   TIME   _ NNNNNNR.%  >0000020.,>0000020.   OVERHEAD TOO HIGH: INVESTIGATE
.ID     CPS0          _  CACHE   _ SWEEP/SEC_ NNNNNNR.   >0000100.,>0000080.   CACHE SWEEPS HIGH: INVESTIGATE
.ID     CPS1          _ SLV CACH _ SWEEP/SEC_ NNNNNNR.   >0000100.,>0000080.   CACHE SWEEPS HIGH: INVESTIGATE
.ID     CPU0          _   USER   _ UUOS/SEC _ NNNNNNR.   >0000300.,>0000200.   USER PGM PROBLEM: CHK WORKLD DATA
.ID     CPU1          _ SLV USER _ UUOS/SEC _ NNNNNNR.   >0000300.,>0000200.   USER PGM PROBLEM: CHK WORKLD DATA
.ID     CTWQ  ??      _ CHAN ??  _  WAIT Q  _ NNNNNN.R   >000001.0,>000000.8   CHANNEL CONTENTION: CHK I/O BALANCE
.ID     IOWP          _  # JOBS  _ BLK IO Q _ NNNNNN.R   >000002.0,>000001.5   BACKUP OF I/O JOBS: CHK CONTENTION
.ID     JLOG          _  # JOBS  _LOGGED IN _ NNNNNNR.                         INCREASE JOBMAX OR RESTRICT LOGINS
.ID     LUFS  ?????   _%PK ?????_ FREE SPC _ NNNNNNR.%  <0000010.,<0000010.   DELETE UNNECESSARY FILES
.ID     LUIO  ?????   _ PK ?????_ BLKS/SEC _ NNNNNNR.   >0000100.,>0000080.   I/O RATE HIGH: CHK FOR CONTENTION
.ID     LUSW  ?????   _ PK ?????_ SWPS/SEC _ NNNNNNR.   >0000200.,>0000160.   SWAPPING HIGH: CHK MEMORY AMOUNT
.ID     LUWQ  ?????   _ PK ?????_  WAIT Q  _ NNNNNN.R   >000001.0,>000000.8   FILE/PACK CONTENTION: INVESTIGATE
.ID     MTAU          _  # MTAS  _ ASSIGNED _ NNNNNNR.                         MANY TAPES ASSIGNED: CHK ACTUAL USE
.ID     MTIO  ??????  _ MT ?????? _ BLKS/SEC _ NNNNNNR.   >0000150.,>0000100.   TAPE I/O RATE HIGH: INVESTIGATE USE
.ID     NRJR          _  # JOBS  _ IN RUN Q _ NNNNNNR.   >0000006.,>0000005.   CPU BOTTLENECK OR SCHEDULER SLOW
.ID     PFCU          _ % MON FR _ CORE USED_ NNNNNNR.%  >0000075.,>0000060.   FREE CORE LOW: FIND CAUSE/UP AMOUNT
.ID     PHCR          _ ALL JOBS _ % USR CR _ NNNNNNR.%
.ID     PI00          _  PI 0 %  _ TIM BUSY _ NNNNN.NR%
.ID     PI01          _ SLV PI 0 _  % BUSY  _ NNNNN.NR%
.ID     PI10          _  PI 1 %  _ TIM BUSY _ NNNNN.NR%
.ID     PI11          _ SLV PI 1 _  % BUSY  _ NNNNN.NR%
.ID     PI20          _  PI 2 %  _ TIM BUSY _ NNNNN.NR%
.ID     PI21          _ SLV PI 2 _  % BUSY  _ NNNNN.NR%
.ID     PI30          _  PI 3 %  _ TIM BUSY _ NNNNN.NR%
.ID     PI31          _ SLV PI 3 _  % BUSY  _ NNNNN.NR%
```

Figure 1-19

```
.ID     PI4O            _  PI 4 %  _ TIM BUSY _ NNNNN.NR%
.ID     PI41            _ SLV PI 4 _  % BUSY  _ NNNNN.NR%
.ID     PI5O            _  PI 5 %  _ TIM BUSY _ NNNNN.NR%
.ID     PI51            _ SLV PI 5 _  % BUSY  _ NNNNN.NR%
.ID     PI6O            _  PI 6 %  _ TIM BUSY _ NNNNN.NR%
.ID     PI61            _ SLV PI 6 _  % BUSY  _ NNNNN.NR%
.ID     PI7O            _  PI 7 %  _ TIM BUSY _ NNNNN.NR%
.ID     PI71            _ SLV PI 7 _  % BUSY  _ NNNNN.NR%
.ID     PRJC            _ % RN JOB _  IN MEM  _ NNNNNNR.%
.ID     PTCU            _ %TY CHNK _  IN USE  _ NNNNNNR.%
.ID     PUFS    ?????   _%UN ????? _ FREE SPC _ NNNNNNR.% <0000010.,<0000010.   DELETE UNNECESSARY FILES
.ID     PUIO    ?????   _ UN ????? _ BLKS/SEC _ NNNNNNR.  >0000100.,>0000080.   I/O RATE HIGH: CHK FOR CONTENTION
.ID     PUSW    ?????   _ UN ????? _ SWPS/SEC _ NNNNNNR.% >0000200.,>0000160.   SWAPPING HIGH: CHK MEMORY AMOUNT
.ID     PUWQ    ?????   _ UN ????? _  WAIT Q  _ NNNNNN.R  >000001.0,>000000.8   FILE/PACK CONTENTION: INVESTIGATE
.ID     SRCV            _ SCN INTR _ RCVS/SEC _ NNNNNNR.
.ID     SWIO            _ SWAPPING _ BLKS/SEC _ NNNNNNR.  >0000200.,>0000160.   SWAPPING HIGH: CHK MEMORY AMOUNT
.ID     SWPS            _ % SWAP   _ SPC LEFT _ NNNNNNR.  <0000025.,<0000030.   SWAP SPACE LOW: ALLOCATE MORE
.ID     SXMT            _ SCN INTR _ XMTS/SEC _ NNNNNNR.
.ID     TIOW            _  # JOBS  _ TTY IO Q _ NNNNNNR.  >0000020.,>0000016.   INTERACTIVE USE HIGH: CHK RESPONSE
.ID     TTYU            _ # LINES  _  IN USE  _ NNNNNNR.                        TTY LINE USE HIGHER THAN PLANNED
.ID     UDIO            _ USER DSK _ BLKS/SEC _ NNNNNNR.  >0000150.,>0000100.   DISK I/O RATE HIGH: CHK PACK I/O
.ID     UMEM            _ PGS USER _  MEMORY  _ NNNNNNR.  <0000768.,<0000768.   MEMORY DOWN: MAY CAUSE SWAPPING
.ID     VMPF            _ VIR MEM  _ FAULT/SEC_ NNNNNN.R  >000005.0,>000005.0   VIRTUAL USERS PAGING TOO OFTEN
.ID     XAMT            _ % AMAR   _ CLK TIME _ NNNNNN.R%                       AMAR NOT RUNNING ENOUGH: CHK REASON
.ID     XRLD            _ # SYSTEM _ RELOADS  _ NNNNNNR.                        TOO MANY RELOADS: CHK REASONS
.ID     XUPT            _ % SYSTEM _  UPTIME  _ NNNNNN.R%                       DOWNTIME HIGH: CHK RELOAD REASONS
.ID     _CPU            _  %  CPU  _   UTIL  _ NNNNNNR.%  >0000095.,>0000070.   CPU PRESSED: CHK WORKLD DATA FIRST

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

.RD     _KEY UTILIZATION ITEMS  _       PAGING BEFORE   PAGING AFTER    DU WC MC

.RI     JLOG            FORCED
.RI     TTYU            FORCED
.RI     CPIO            FORCED
.RI     CPLO            FORCED
.RI     CPOO            FORCED
.RI     ACSR            FORCED
.RI     SWIO            FORCED
.RI     UDIO            FORCED
.RI     MTAU            FORCED
.RI     UMEM            FORCED

.RD     _KEY UTILIZATION ITEMS          _       PAGING BEFORE   PAGING AFTER    WU MU

.RI     _CPU            FORCED
.RI     JLOG            FORCED
.RI     TTYU            FORCED
.RI     CPIO            FORCED
.RI     CPLO            FORCED
.RI     CPOO            FORCED
.RI     ACSR            FORCED
.RI     SWIO            FORCED
.RI     UDIO            FORCED
.RI     MTAU            FORCED
.RI     UMEM            FORCED
.RI     XUPT            FORCED
.RI     XAMT            FORCED
.RI     XRLD            FORCED
```

Figure 1-19 (continued)

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

.RI     ACCR            TESTED
.RI     AVJS            TESTED
.RI     AVRT            TESTED
.RI     CH00            TESTED
.RI     CH10            TESTED
.RI     CH20            TESTED
.RI     CH30            TESTED
.RI     CH40            TESTED
.RI     CH50            TESTED
.RI     CH60            TESTED
.RI     CH70            TESTED
.RI     CH01            TESTED
.RI     CH11            TESTED
.RI     CH21            TESTED
.RI     CH31            TESTED
.RI     CH41            TESTED
.RI     CH51            TESTED
.RI     CH61            TESTED
.RI     CH71            TESTED
.RI     CPC0            TESTED
.RI     CPC1            TESTED
.RI     CPS0            TESTED
.RI     CPU0            TESTED
.RI     CPI1            TESTED
.RI     CPL1            TESTED
.RI     CP01            TESTED
.RI     CPS1            TESTED
.RI     CPU1            TESTED
.RI     CTWQ    ??      TESTED
.RI     IOWP            TESTED
.RI     LUFS    ?????   TESTED
.RI     LUIO    ?????   TESTED
.RI     LUSW    ?????   TESTED
.RI     LUWQ    ?????   TESTED
.RI     MTIO    ??????  TESTED
.RI     NRJR            TESTED
.RI     PFCU            TESTED
.RI     PHCR            TESTED
.RI     PI00            TESTED
.RI     PI10            TESTED
.RI     PI20            TESTED
.RI     PI30            TESTED
.RI     PI40            TESTED
.RI     PI50            TESTED
.RI     PI60            TESTED
.RI     PI70            TESTED
.RI     PI01            TESTED
.RI     PI11            TESTED
.RI     PI21            TESTED
.RI     PI31            TESTED
.RI     PI41            TESTED
.RI     PI51            TESTED
.RI     PI61            TESTED
.RI     PI71            TESTED
.RI     PRJC            TESTED
.RI     PTCU            TESTED
.RI     PUFS    ?????   TESTED
.RI     PUIO    ?????   TESTED
.RI     PUSW    ?????   TESTED
```

Figure 1-19 (continued)

```
.RI     PUWQ    ?????   TESTED
.RI     SRCV            TESTED
.RI     SWPS            TESTED
.RI     SXMT            TESTED
.RI     TIOW            TESTED
.RI     VMPF            TESTED

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

.RI     _CPU            FORCED
.RI     CPIO            FORCED
.RI     CPOO            FORCED
.RI     CPLO            FORCED
.RI     CPUO            FORCED
.RI     CPCO            FORCED
.C      CPI1            FORCED
.C      CPO1            FORCED
.C      CPL1            FORCED
.C      CPU1            FORCED
.C      CPS1            FORCED
.C      CPC1            FORCED
.RI     ACSR            FORCED
.RI     ACCR            FORCED
.RI     PHCR            FORCED
.RI     PRJC            FORCED
.RI     AVJS            FORCED
.RI     JLOG            FORCED
.RI     UMEM            FORCED
.RI     TIOW            FORCED
.RI     UDIO            FORCED
.RI     NRJR            FORCED
.RI     SWIO            FORCED
.RI     LUSW    ?????   FORCED
.RI     PUSW    ?????   FORCED
.RI     AVRT            FORCED
.RI     MTAU            FORCED
.RI     TTYU            FORCED
.RI     PTCU            FORCED
.RI     SRCV            FORCED
.RI     SXMT            FORCED
.RI     XUPT            FORCED
.RI     XAMT            FORCED
.RI     XRLD            FORCED

.RD     _GENERAL SUMMARY                _       PAGING BEFORE                   DD WD MD

.RI     SWIO            FORCED
.RI     SWPS            FORCED
.RI     CHOO            FORCED
.RI     CH1O            FORCED
.RI     CH2O            FORCED
.RI     CH3O            FORCED
.RI     CH4O            FORCED
.RI     CH5O            FORCED
.RI     CH6O            FORCED
.RI     CH7O            FORCED
.C      CHO1            FORCED
.C      CH11            FORCED
.C      CH21            FORCED
.C      CH31            FORCED
```

Figure 1-19 (continued)

```
.C      CH41            FORCED
.C      CH51            FORCED
.C      CH61            FORCED
.C      CH71            FORCED
.RI     PI00            FORCED
.RI     PI10            FORCED
.RI     PI20            FORCED
.RI     PI30            FORCED
.RI     PI40            FORCED
.RI     PI50            FORCED
.RI     PI60            FORCED
.RI     PI70            FORCED
.C      PI01            FORCED
.C      PI11            FORCED
.C      PI21            FORCED
.C      PI31            FORCED
.C      PI41            FORCED
.C      PI51            FORCED
.C      PI61            FORCED
.C      PI71            FORCED
.RI     CTWQ    ??      FORCED

.RD     _PACK NAME                      _               PAGING AFTER    DD WD MD

.RI     LUFS    ?????   FORCED
.RI     LUIO    ?????   FORCED
.RI     LUSW    ?????   FORCED
.RI     LUWQ    ?????   FORCED

.RD     _UNIT NAME                      _                               DD WD MD

.RI     PUFS    ?????   FORCED
.RI     PUIO    ?????   FORCED
.RI     PUSW    ?????   FORCED
.RI     PUWQ    ?????   FORCED

.RD     _GENERAL SUMMARY                _       PAGING BEFORE           DT WT MT

.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 with the new limits. 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 DSKX0. 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 DSKX0 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 DSKX0 has been specified in the LUFS list, the "DELETE UNNECESSARY FILES" text on the LUFS DSKX0 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 and replace it with something more interesting such as CPCO, context switches per second. 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 CPCO. The number of context switches 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 an OPSER subjob and a permanently mounted disk area with approximately 3.2K blocks for program and raw file storage. The size of your system AMAR database will probably vary from 3K - 6K blocks 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".

The minimum supporting software required is COBOL 12A.

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 submit queue, set to run /AFTER:1:0:0.

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 submit 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 PPN, project, charge number, program name, batch vs. timesharing, scheduler class, 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 four functions of Workload AMAR (also referred to as the workload system) are data collection, preprocessing, database management, and reporting. The four 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

WHOWC collects resource utilization, identification, and state data about each job on the system at user-specified "checkpoint intervals", typically every 5 minutes. WHOWC runs 24 hours a day, preferably as an OPSER subjob.

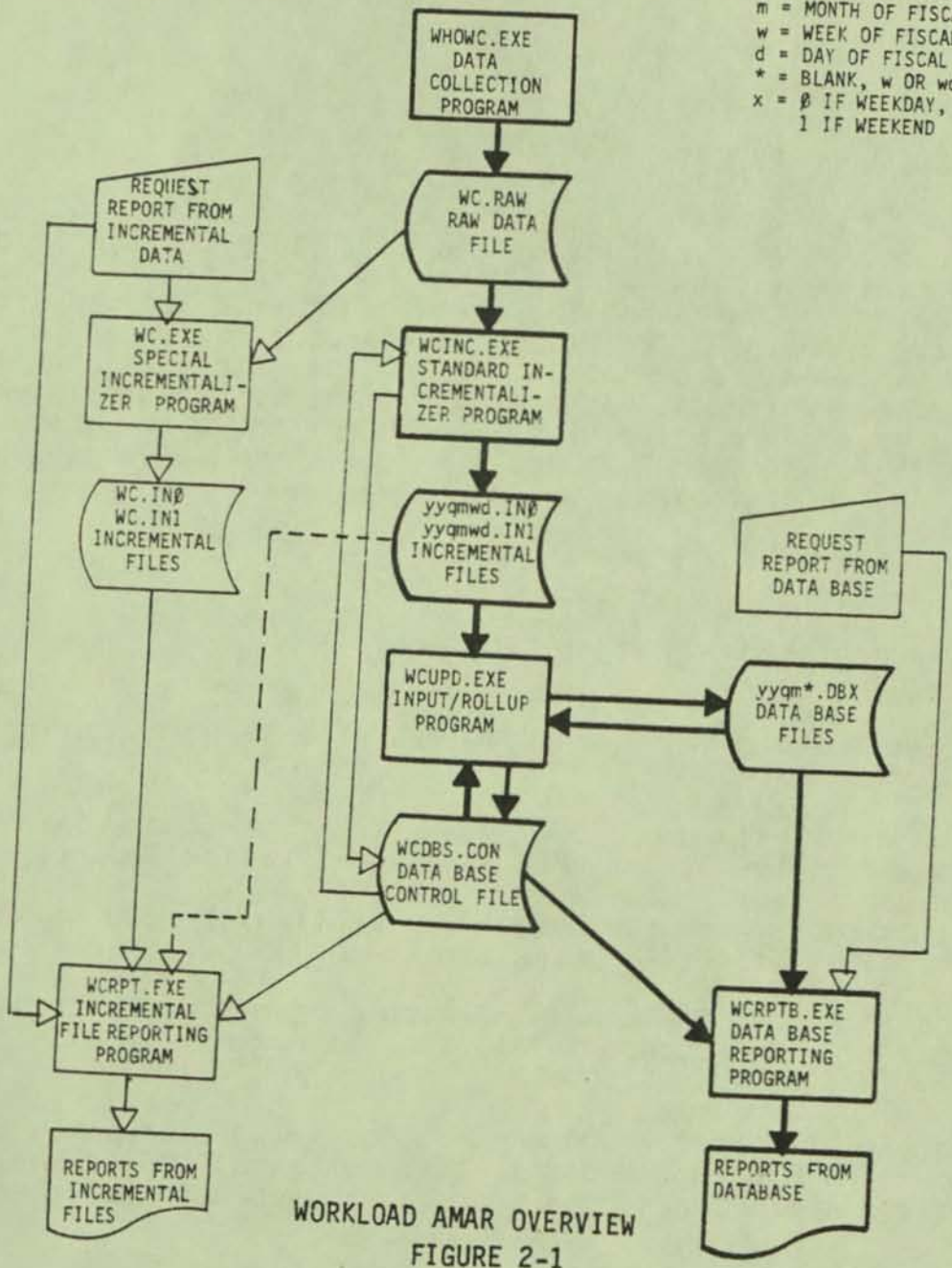
2.2.2 Preprocessing

WCINC converts the data from "checkpoint records" collected by WHOWC into "incremental usage records" required as input to WCUPD. WCINC actually produces two output files: one which contains incremental usage by job and one which summarizes incremental usage by checkpoint interval. WCINC is normally run once a day. The program WC is a special version of WCINC which will let you preprocess current data without disrupting the normal cycle.

2.2.3 Database Management

WCUPD 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 QUARTER
w = WEEK OF FISCAL MONTH
d = DAY OF FISCAL WEEK
* = BLANK, w OR wd
x = Ø IF WEEKDAY,
1 IF WEEKEND



WORKLOAD AMAR OVERVIEW
FIGURE 2-1

2.2.4 Reporting

The main reporting program, WCRPTB, 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 and state 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. WCRPT is a variation of WCRPTB which is used to report from preprocessed data rather than the database.

2.3 ANNOTATED SAMPLE REPORTS

This section contains four samples of the workload reports produced by the standard daily stream WCRPTB.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 Showing Major CPU Users

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 three subreports. The subreports start at approximately midnight and run to 3:00 AM (00:00:02 to 03:00:02).

The second page of Figure 2-2 shows portions of two subreports from a busy period. The subreports start at approximately 15:00 and run to 17:00. 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 WCRPTB 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, JOB number and LOGIN date/time identify a unique session and PRGRM identifies a portion of that session when a particular program was running.

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.

15. Cutoff criteria. In this case, each detail line representing less than 0.90% 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 during this reporting interval.

18. The average job size in pages in the INTERVAL TOTALS line is a weighted average of job sizes during the period. (Other numbers in the INTERVAL TOTALS line are ordinary totals).

DETAIL LINES:

19. Job number (as would be reported by SYSTAT).

20. 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.

21. 1 = job logged in during interval.

22. 0 = job was in progress.

23. 1 = job logged out during interval.

24. 0 = job remained logged in.

25. Project/programmer number (PPN).

26. Charge number (on systems where it is used).

27. Program name. This is the program that was running at the end of each of the samples rolled into this detail line. It is probably the program that used most of the resources, but it may not be if only one or two samples are represented. In this example, one sample would represent about 0.1 in the AVG JOBS column.

28. The average job size is computed by dividing kilocore seconds by CPU seconds (and multiplying by 2 pages / K). Hence, it is the average size while using the processor. If the job size is as expected for the program, it tends to confirm that it was that program which used the resources. Note that if a detail line represents exactly 0% of the CPU, the average job size will be reported as 0.0 pages.

29. CPU% is CPU seconds charged to this detail line divided

by seconds of measured time during the interval (times 100 to make it a percentage). This measure is based on the way the system charges processor usage to a job, with whatever shortcomings that may have. For example, a KL10 using EBOX/MBOX accounting might only charge users for about 70% of the time they use the processor. Also, resources used between the last time WHOWC checkpoints a job and when the job logs out will not be captured. Thus, all resource totals will usually tend to be somewhat low when compared with the corresponding data extracted from accounting (USAGE) files. On the other hand, resource totals may occasionally be high, because WHOWC checkpoints more frequently than DAEMON. Note also that on a dual processor, users may use the equivalent of more than 100% of a processor.

30. UUO's are Monitor calls. It is common for a job to use one UUO for each disk block read or written, but some jobs use many other UUO's.

31. READS/SECOND is short for disk blocks read per second which is what is actually reported.

32. WRITES/SECOND is similar to READS/SECOND.

33. The scheduler class is normally reported in this column. If the job was running in a high priority queue, the high priority queue number is reported, flagged with a minus sign to distinguish it from an ordinary scheduler class.

34. LOGIN category. Login category may be used to group users and then limit the number of simultaneous users in the group who can log in. Note: This value is displayed in octal.

35. T = timesharing.

36. B = batch.

37. Job state. (SL=sleep, R2=run, TI=teletype input wait, etc. See the SYSTAT description in the Operating System Commands manual.) Note that when the job was observed in more than one state during the interval, question marks show that the state varied.

38. Terminal number where the job was running.

39. Node to which the job was attached.

40. Line number on that node. Note: This value is displayed in octal.

41. 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.

42. Time of day when the job logged in.

SUBTOTAL LINES:

43. 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.

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

```
                                    AMAR WORKLOAD REPORT

   SITE:  <Put Any Title Here>                                        SYSTEM:  PATH

   REPORT DESCRIPTION:  HOURLY REPORT BY PROGRAM AND JOB

   INPUT FILE:  823242.DBC  (FISCAL YEAR: 82   QUARTER: 3   MONTH: 2   WEEK: 4   DAY: 2   MONDAY)

FROM:   0: 0: 2  ON  MONDAY     15-FEB-82     TO:   1: 0: 2  ON  MONDAY     15-FEB-82     INTERVAL:   0:59:59     MEASURED: 100%

GROUPED BY:  JOB    PRGRM LOGIN

SORTED BY:  CPU%      CUTOFF:  0.90% OF CPU

JOB   AVG  LOG LOG ------PPN-----  CHARGE PRGRM  PAGES    CPU%   UUOS   READS  WRITES SCD CT B STATE  TTY    NODE--LINE LOGIN AT
 #    JOBS  IN OUT     PROJ  PROG  NUMBER  NAME  (AVG)           /SEC    /SEC    /SEC CLS                              DAY TIME

     21.0   1   0 ******INTERVAL TOTALS******    24.4     0.4    4.4     0.2     0.2


FROM:   1: 0: 2  ON  MONDAY     15-FEB-82     TO:   2: 0: 1  ON  MONDAY     15-FEB-82     INTERVAL:   0:59:59     MEASURED: 100%

GROUPED BY:  JOB    PRGRM LOGIN

SORTED BY:  CPU%      CUTOFF:  0.90% OF CPU

JOB   AVG  LOG LOG ------PPN-----  CHARGE PRGRM  PAGES    CPU%   UUOS   READS  WRITES SCD CT B STATE  TTY    NODE--LINE LOGIN AT
 #    JOBS  IN OUT     PROJ  PROG  NUMBER  NAME  (AVG)           /SEC    /SEC    /SEC CLS                              DAY TIME

     22.0   0   0 ******INTERVAL TOTALS******    89.1     1.2   10.5     1.6     1.2


FROM:   2: 0: 1  ON  MONDAY     15-FEB-82     TO:   3: 0: 2  ON  MONDAY     15-FEB-82     INTERVAL:   1:.0: 0     MEASURED: 100%

GROUPED BY:  JOB    PRGRM LOGIN

SORTED BY:  CPU%      CUTOFF:  0.90% OF CPU

JOB   AVG  LOG LOG ------PPN-----  CHARGE PRGRM  PAGES    CPU%   UUOS   READS  WRITES SCD CT B STATE  TTY    NODE--LINE LOGIN AT
 #    JOBS  IN OUT     PROJ  PROG  NUMBER  NAME  (AVG)           /SEC    /SEC    /SEC CLS                              DAY TIME

     22.0   0   0 ******INTERVAL TOTALS******    22.7     0.4    7.7     0.1     0.1
```

Figure 2-2

AVG JOBS	LOG IN	LOG OUT		PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC
54.5	30	25	***SUBTOTALS AFTER CUTOFF**	46.6	12.6	137.6	24.2	3.6

FROM: 15: 0: 2 ON MONDAY 15-FEB-82 TO: 16: 0: 3 ON MONDAY 15-FEB-82 INTERVAL: 1: 0: 0 MEASURED: 100%

GROUPED BY: JOB PRGRM LOGIN

SORTED BY: CPU% CUTOFF: 0.90% OF CPU

JOB #	AVG JOBS	LOG IN	LOG OUT	PPN PROJ	PPN PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE	LINE	LOGIN AT DAY	TIME
	70.[illegible]	25	24	******INTERVAL TOTALS******				54.[illegible]	36.[illegible]	418.5	29.[illegible]	18.[illegible]									
59	0.3	0	1	[15323	112157]	555267	DIRECT	34.4	6.3	13.8	0.0	0.0	4	0	T	?? ??	TTY37	NODED	37	0	14:40
59	0.5	0	0	[15323	112157]	555267	BACKUP	22.0	5.0	27.3	0.0	0.0	4	0	T	??	TTY37	NODED	37	0	14:40
67	0.6	1	0	[15375	112420]	555571	1022FO	144.4	2.5	1.8	0.6	0.1	4	0	T	??	TTY54	NODED	54	0	15:23
43	1.0	0	0	[1	2]	133085	DLLSPL	29.9	2.4	161.4	0.1	0.0	0	54	T	?? ??	TTY103	NODED	103	0	8:47
47	0.8	0	0	[12400	112167]	750204	1022	49.9	1.4	8.7	2.9	0.0	4	0	T	??	TTY114	NODEM	6	0	12:27
32	0.6	0	0	[20320	112161]	555040	DIRECT	28.2	1.3	11.4	2.6	0.4	4	0	T	??	TTY77	NODED	77	0	14:30
7	1.0	0	0	[1	2]		MIC	25.6	1.1	46.0	0.1	0.0	0	0	T	??	DET313		0	-4	23:18
27	0.3	0	1	[15152	45010]	216300	BACKUP	21.9	1.1	5.9	0.0	0.0	5	0	B	??	PTY23		0	0	15:27
52	0.1	0	0	[15152	45010]	216300	BACKUP	23.8	1.1	13.8	0.0	10.8	5	0	B	R2	PTY24		0	0	15:54
44	1.0	0	0	[12400	132153]	750240	1022	92.8	1.0	6.6	2.4	0.3	4	0	T	??	TTY75	NODED	75	0	11:11
61	0.1	0	0	[15152	45010]	216300	CBL74	78.9	0.9	1.2	0.6	0.6	4	0	T	R2	TTY16	NODED	16	0	13:16
	[illegible]6.2	1	2	***SUBTOTALS THRU CUTOFF***				45.9	24.2	298.0	9.4	12.1									
	[illegible]4.0	24	22	***SUBTOTALS AFTER CUTOFF**				72.0	12.7	120.6	20.2	6.7									

FROM: 16: 0: 3 ON MONDAY 15-FEB-82 TO: 17: 0: 2 ON MONDAY 15-FEB-82 INTERVAL: 0:59:59 MEASURED: 100%

GROUPED BY: JOB PRGRM LOGIN

SORTED BY: CPU% CUTOFF: 0.90% OF CPU

JOB #	AVG JOBS	LOG IN	LOG OUT	PPN PROJ	PPN PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE	LINE	LOGIN AT DAY	TIME
	58.5	25	50	******INTERVAL TOTALS******				46.0	41.1	251.1	36.9	52.9									
28	0.5	0	0	[15152	45010]	216300	ISAM	14.0	13.7	20.5	10.1	10.2	4	0	T	R2	TTY55	NODED	55	0	15:03
6	0.3	0	0	[15152	112157]	216300	0020CC	119.6	3.5	2.2	1.1	0.3	4	0	T	?? ??	TTY17	NODED	17	0	8:15
52	0.4	0	1	[15152	45010]	216300	BACKUP	24.0	2.8	36.8	0.0	29.4	5	0	B	??	PTY24		0	0	15:54
49	0.2	0	0	[20320	112136]	555040	COBOL	57.3	2.5	1.0	0.1	0.1	4	0	T	R2	TTY76	NODED	76	0	9:14
67	0.6	0	1	[15375	112420]	555571	1022FO	141.8	1.8	1.3	0.4	0.1	4	0	T	??	TTY54	NODED	54	0	15:23
49	0.8	0	0	[20320	112136]	555040	SED	68.0	1.6	9.8	0.9	0.3	4	0	T	??	TTY76	NODED	76	0	9:14
32	0.2	1	1	[275	43226]	133105	DIP	32.0	1.4	14.0	0.1	0.0	4	1	T	??	TTY121	NODEM	23	0	16:21
59	0.4	0	0	[15152	112441]	224200	BACKUP	20.6	1.0	7.4	0.1	1.9	4	0	T	?? ??	TTY35	NODED	35	0	16:08
44	1.0	0	0	[12400	132153]	750240	1022	92.6	1.0	7.2	2.6	0.2	4	0	T	??	TTY75	NODED	75	0	11:11
	4.4	1	3	***SUBTOTALS THRU CUTOFF***				45.7	29.4	100.1	15.4	42.5									
	54.1	24	47	***SUBTOTALS AFTER CUTOFF**				46.8	11.7	151.0	21.4	10.4									

Figure 2-2 (continued)

Blank Page

2.3.2 Program Name Report

Figure 2-3 is a sample of the default report WCWK0.RP2 and shows weekly resource usage by program 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, the program 1022 (K) was the heaviest user of the processor. This program had, on average, 9.4 (L) simultaneous users and accounted for approximately 376 hours of connect time over the five days of prime shift. Connect time equals the average number of jobs (9.4 (L)) multiplied by the interval measured (40.0(N)). A CPU cutoff (J) of 0.1% is used to suppress printing of any lines containing programs which used less than 0.1% 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 PPN and charge number fields indicate that the values of these items varied; i.e., more than one user ran the program under more than one charge number.

As another example, the ISAM program was the second heaviest user during prime shift, even though it was being run only 20% (M) of the time.

The third heaviest CPU user was COBOL. Note that here it used 2.8% of the CPU although it ran only 20% of the time. That means that while it was running it used about 14% of the CPU (2.8% divided by 20%).

```
                                    AMAR WORKLOAD REPORT

  SITE:  <Put Any Title Here>                                              SYSTEM:  PATH

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

  INPUT FILE:  82324 .DBO    (FISCAL YEAR: 82   QUARTER: 3   MONTH: 2   WEEK: 4   WEEKDAYS)
```

(A) (B) (C) (D) (E) (F) (G) (H) (I)

```
 0: 0 -  8: 0  WEEKDAYS (-HOLIDAYS)     FROM: MON 15-FEB-82  TO: FRI 19-FEB-82  ( 5 DAYS)     MEASURED:  99% =  39.77 HOURS

GROUPED BY:    PRGRM

SORTED BY:     CPU%     CUTOFF:  0.10% OF CPU

JOB  AVG  LOG LOG ------PPN-----  CHARGE PRGRM   PAGES   CPU%   UUOS   READS  WRITES SCD CT B STATE   TTY     NODE--LINE LOGIN AT
 #   JOBS   IN OUT    PROJ  PROG  NUMBER  NAME   (AVG)          /SEC    /SEC    /SEC CLS                                  DAY TIME

     22.2   95  38 ******INTERVAL TOTALS******    48.9    1.2   11.3     2.2     0.5

      0.0    1   1 [     1      2] 133085 BACKUP   23.8    0.2    1.8     1.1     0.0   0 54 T ??          NODED
      0.1   14   0 [ 124001?2???] 750??? 1022      64.0    0.2    0.7     0.2     0.0   4  0 T ??          NODE?
      0.0    5   5 [  2104 51335] 133070 5955LS    92.8    0.1    0.3     0.1     0.1   5  1 B ??

      0.1   20   6 ***SUBTOTALS THRU CUTOFF***     55.2    0.5    2.8     1.5     0.1

     22.1   75  32 ***SUBTOTALS AFTER CUTOFF**     44.9    0.7    8.5     0.7     0.4


 8: 0 - 16: 0  WEEKDAYS (-HOLIDAYS)     FROM: MON 15-FEB-82  TO: FRI 19-FEB-82  ( 5 DAYS)     MEASURED: 100% =  40.00 HOURS

GROUPED BY:    PRGRM

SORTED BY:     CPU%     CUTOFF:  0.10% OF CPU

JOB  AVG  LOG LOG ------PPN-----  CHARGE PRGRM   PAGES   CPU%   UUOS   READS  WRITES SCD CT B STATE   TTY     NODE--LINE LOGIN AT
 #   JOBS   IN OUT    PROJ  PROG  NUMBER  NAME   (AVG)          /SEC    /SEC    /SEC CLS                                  DAY TIME

     58.5 1255 1079 ******INTERVAL TOTALS******    57.3   30.7  204.6    62.3    23.8

      9.4  296  309 [ ???????????] ?????? 1022      59.3    8.0   37.1    23.6     2.5         ?? ??          ?????
      0.2    3    1 [ ???????????] ?????? ISAM      15.4    3.4    5.1     3.5     3.4      0  ?? ??          ?????
      0.2    5    7 [ ???????????] ?????? COBOL     58.9    2.8    1.3     0.3     0.2   4  0 T ?? ??       ?????
      4.1   95   82 [???????????] ?????? SOS       51.7    1.6   17.2     1.9     1.0   4     T ?? ??       ?????
      1.2   19   15 [ ???????????] ?????? DIP       33.8    1.4   27.7     6.0     5.8               ?? ??  ?????
      0.7   22   22 [ 1????112???] ?5???? 1022FO   143.5    1.3    1.2     0.6     0.0   4  0 T ?? ??       ?????
      1.5   77   59 [???????????] ?????? DIRECT    44.3    1.0    6.7     2.4     0.6               ?? ??  ?????
      2.6  105  107 [???????????] ?????? PIP       47.1    0.9    4.7     1.7     0.5               ?? ??  ?????
      0.2    2   13 [ ???????????] ?????? BACKUP    23.8    0.9    8.8     3.8     1.6               ?? ??  ?????
      0.6   24   27 [???????????] ?????? WHO       59.5    0.6    1.8     0.1     0.0               ?? ??  ?????
      0.0    1    1 [ 15152112157] 216300 0010DE    64.0    0.6    0.0     0.0     0.0   5  0 B R2
      1.1   11    4 [ ???????????] ?????? MIC       31.4    0.5   15.5     0.3     0.1                  T ?? ??       ?????
```

(K) (L) (M) (N) (J)

2.3.2 Program Name Report

Figure 2-3 is a sample of the default report WCWK0.RP2 and shows weekly resource usage by program 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, the program 1022 (K) was the heaviest user of the processor. This program had, on average, 9.4 (L) simultaneous users and accounted for approximately 376 hours of connect time over the five days of prime shift. Connect time equals the average number of jobs (9.4 (L)) multiplied by the interval measured (40.0(N)). A CPU cutoff (J) of 0.1% is used to suppress printing of any lines containing programs which used less than 0.1% 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 PPN and charge number fields indicate that the values of these items varied; i.e., more than one user ran the program under more than one charge number.

As another example, the ISAM program was the second heaviest user during prime shift, even though it was being run only 20% (M) of the time.

The third heaviest CPU user was COBOL. Note that here it used 2.8% of the CPU although it ran only 20% of the time. That means that while it was running it used about 14% of the CPU (2.8% divided by 20%).

```
                                        AMAR WORKLOAD REPORT

    SITE:  <Put Any Title Here>                                                    SYSTEM:  PATH

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

    INPUT FILE:  82324 .DBO    (FISCAL YEAR: 82   QUARTER: 3   MONTH: 2   WEEK: 4   WEEKDAYS)
```

(A) (B) (C) (D) (E) (F) (G) (H) (I)

```
 0: 0 -  8: 0  WEEKDAYS (-HOLIDAYS)     FROM: MON 15-FEB-82  TO: FRI 19-FEB-82  ( 5 DAYS)     MEASURED:  99% =  39.77 HOURS

GROUPED BY:    PRGRM

SORTED BY:     CPU%     CUTOFF:  0.10% OF CPU

JOB  AVG  LOG LOG ------PPN-----  CHARGE PRGRM  PAGES   CPU%   UUOS  READS WRITES SCD CT B STATE  TTY    NODE--LINE LOGIN AT
 #   JOBS   IN OUT     PROJ  PROG  NUMBER  NAME  (AVG)          /SEC   /SEC   /SEC CLS                                DAY TIME

     22.2  95  38 ******INTERVAL TOTALS******   48.9    1.2   11.3    2.2    0.5

      0.0   1   1 [     1     2] 133085 BACKUP  23.8    0.2    1.8    1.1    0.0   0 54 T ??          NODED
      0.1  14   0 [ 124001?2???] 7502?? 1022    64.0    0.2    0.7    0.2    0.0   4  0 T ??          NODE?
      0.0   5   5 [  2104 51335] 133070 5955LS  92.8    0.1    0.3    0.1    0.1   5  1 B ??

      0.1  20   6 ***SUBTOTALS THRU CUTOFF***   55.2    0.5    2.8    1.5    0.1

     22.1  75  32 ***SUBTOTALS AFTER CUTOFF**   44.9    0.7    8.5    0.7    0.4
```

(N)

```
 8: 0 - 16: 0  WEEKDAYS (-HOLIDAYS)     FROM: MON 15-FEB-82  TO: FRI 19-FEB-82  ( 5 DAYS)     MEASURED: 100% =  40.00 HOURS

GROUPED BY:    PRGRM
```

(J)

```
SORTED BY:     CPU%     CUTOFF:  0.10% OF CPU

JOB  AVG  LOG LOG ------PPN-----  CHARGE PRGRM  PAGES   CPU%   UUOS  READS WRITES SCD CT B STATE  TTY    NODE--LINE LOGIN AT
 #   JOBS   IN OUT     PROJ  PROG  NUMBER  NAME  (AVG)          /SEC   /SEC   /SEC CLS                                DAY TIME

     58.5 1255 1079 ******INTERVAL TOTALS******   57.3   30.7  204.6   62.3   23.8
```

(L) (K)

```
      9.4 296 309 [ ???????????] ?????? 1022    59.3    8.0   37.1   23.6    2.5          ?? ??          ?????
```

(M)

```
      0.2   3   1 [ ???????????] ?????? ISAM    15.4    3.4    5.1    3.5    3.4       0  ?? ??          ?????
      0.2   5   7 [ ???????????] ?????? COBOL   58.9    2.8    1.3    0.3    0.2   4   0 T ?? ??          ?????
      4.1  95  82 [???????????] ?????? SOS     51.7    1.6   17.2    1.9    1.0   4     T ?? ??          ?????
      1.2  19  15 [ ???????????] ?????? DIP     33.8    1.4   27.7    6.0    5.8             ?? ??          ?????
      0.7  22  22 [ 1????112???] ?5???? 1022FO 143.5    1.3    1.2    0.6    0.0   4   0 T ?? ??          ?????
      1.5  77  59 [???????????] ?????? DIRECT  44.3    1.0    6.7    2.4    0.6             ?? ??          ?????
      2.6 105 107 [???????????] ?????? PIP     47.1    0.9    4.7    1.7    0.5             ?? ??          ?????
      0.2   2  13 [ ???????????] ?????? BACKUP  23.8    0.9    8.8    3.8    1.6             ?? ??          ?????
      0.6  24  27 [???????????] ?????? WHO     59.5    0.6    1.8    0.1    0.0             ?? ??          ?????
      0.0   1   1 [ 15152112157] 216300 0010DE  64.0    0.6    0.0    0.0    0.0   5   0 B R2
      1.1  11   4 [ ???????????] ?????? MIC     31.4    0.5   15.5    0.3    0.1             T ?? ??          ?????
```

Figure 2-3

```
 1.1  14  15 [ ???????????] ?????? SED      46.8   0.5    5.0    0.3    0.1          T ?? ??          ?????
 0.0   1   0 [   275 42203] 462300 0004HR  257.7   0.4    1.1    1.6    0.1   3  2 B R2
 1.0   1   1 [     1     2]        FILDAE   14.0   0.3    1.8    0.2    0.0  -1  0 T ?? ??
 0.1   3   2 [ 15152??????] 216?00 COBABY  74.3   0.3    2.7    3.7    0.2      0   ?? ??          ?????
 0.0   0   1 [   275 42203] 216300 0100PT  79.6   0.3    1.9    2.1    1.3   3  2 B ??
 2.7   5   6 [     1     2] 13302? LPTSPL  24.9   0.2    1.7    0.2    0.0   0 54 T ?? ??
 1.6  11  18 [ ???????????] ?????? S       48.5   0.2    2.2    0.2    0.1          T ^C ??          ?????
 1.2  12   8 [ 15323112???] 555267 1005PT 140.6   0.2    1.1    1.0    0.4   4  0 T ?? ??          ?????
 0.6  27  27 [ ???????????] ?????? TECO    39.4   0.2    1.0    0.1    0.0          T ?? ??          ?????
 0.1   3   1 [ 20320112161] 555040 ROWMON  62.5   0.2    1.8    0.2    0.1   4  0 T ?? ??          NODED
 0.2   8  11 [ ???????????] ?????? BASIC   59.6   0.2    0.5    0.1    0.0   4    T ?? ??          ?????
 1.0   1   1 [     1     2] 133025 BATCON  27.4   0.2    4.7    0.1    0.1   0 54 T ?? ??
 0.0   1   0 [ 15152112157] 216300 LOOK    36.0   0.2    0.1    0.1    0.0   4  0 T ??           NODED
 1.0   0   1 [     1     2] 133010 QUASAR  26.8   0.2    1.7    0.0    0.1   0 54 T ?? ??
 2.2  20  36 [???????????] ?????? QUEUE    63.0   0.2    1.2    0.3    0.1          ?? ??          ?????
 0.0   0   0 [ 15152 4????] 216?00 CBL74   78.1   0.2    0.3    0.1    0.1   4  0 T ?? ??          NODED
 0.1   1   0 [ 20320112143] 463395 PUSH    78.6   0.2   12.4    0.0    0.0   4  0 T ?? ??          NODED
 0.0   0   0 [ 15152112157] 216300 0020CC 115.2   0.2    0.1    0.1    0.0   4  0 T ?? ??          NODED
 0.0   0   1 [ 15152112403] 555267 0080TN  90.8   0.1    0.9    0.2    1.0   4  0 T ??           NODED
 1.0   1   1 [     1     2]        DAEMON   22.0   0.1    1.0    0.1    0.1      0 T ?? ??
 2.9   6   5 [ ???????????] ?????? OPSER    12.0   0.1    4.7    0.2    0.1          T ?? ??          ?????
 0.1   2   2 [  7700112151] 133100 FRCST    30.0   0.1    0.1    0.0    0.0   4  0 T ??           NODEN
 0.2  34  14 [ ???????????] ?????? UMOUNT   31.7   0.1    1.1    0.3    0.2          ?? ??          ?????

38.9 811 798 ***SUBTOTALS THRU CUTOFF***    55.7  28.1  176.4   55.4   20.3

19.5 444 281 ***SUBTOTALS AFTER CUTOFF**    74.0   2.6   28.3    6.9    3.5
```

Figure 2-3 (continued)

16: 0 - 24: 0 WEEKDAYS (-HOLIDAYS) FROM: MON 15-FEB-82 TO: FRI 19-FEB-82 (5 DAYS) MEASURED: 89% = 35.65 HOURS

GROUPED BY: PRGRM

SORTED BY: CPU% CUTOFF: 0.10% OF CPU

```
JOB  AVG  LOG LOG ------PPN----- CHARGE PRGRM  PAGES   CPU%   UUOS  READS WRITES SCD CT B STATE  TTY    NODE--LINE LOGIN AT
 #   JOBS  IN OUT    PROJ  PROG  NUMBER  NAME  (AVG)          /SEC   /SEC   /SEC CLS                              DAY TIME

32.4 261 450 ******INTERVAL TOTALS******    43.0  11.1   94.1   36.6   14.7

 0.2   0   2 [ 1???????????] ?????? ISAM     15.3   2.7    3.8    2.1    2.1   4  0 T ??           NODE?
 0.2  12  14 [ ???????????] ?????? BACKUP   27.4   1.9   24.6   16.7    0.9          ?? ??          ?????
 2.0  36  95 [ ???????????] ?????? 1022     56.9   1.0    6.2    3.2    0.2   4  0 T ?? ??          ?????
 0.0   0   0 [     1     2] 680160 DDRPI    94.6   0.7    4.4    3.6    3.6   4  2 T ??           NODED
 0.0   1   0 [     1     2] 133085 FSCOPY   47.7   0.4    1.3    5.2    5.0   5 54 B ??
 0.1   0   0 [  7700112121?] 133085 VTTECO  29.9   0.4    6.9    0.0    0.0   0 54 T ??           NODEM
 0.8  12  22 [ ???????????] ?????? SOS      53.5   0.4    3.4    0.4    0.2          T ?? ??        ?????
 1.0   3   2 [210400 62165] 133000 NTRACK   69.3   0.3    8.6    0.0    0.0   5  0 B ?? ??
 0.7  20  39 [ ???????????] ?????? PIP      82.0   0.3    1.2    0.3    0.1          ?? ??          ?????
 0.3  11  10 [ ???????????] ?????? DIP      36.4   0.2    3.3    0.9    0.6          ?? ??          ?????
 2.9  11   7 [     1     2] 13302? LPTSPL   24.5   0.2    1.0    0.2    0.0   0 54 T ?? ??
 0.1   0   5 [ 1?????112???] 555571 1022F0 146.6   0.2    0.3    0.3    0.0   4  0 T ??           NODE?
 0.3  11  17 [ ???????????] ?????? DIRECT   37.7   0.1    1.1    0.4    0.2          T ?? ??        ?????
 1.0   3   2 [     1     2] 133025 BATCON   22.7   0.1    4.6    0.0    0.0   0 54 T ??
 0.1   2   3 [ 20320112153] 555040 VTED     86.6   0.1    1.6    0.0    0.0   4  0 T ??           NODED
 0.2   3   8 [ ???????????] ?????? SED      52.9   0.1    1.2    0.1    0.0          T ?? ??        NODED
 0.0   0   1 [  7700112151] 133100 FRCST    30.0   0.1    0.1    0.0    0.0   4  0 T T?           NODEN
 0.0   1   1 [ ?????112???] ?????? COBOL    57.4   0.1    0.3    0.0    0.0   4  0 T ??           NODE?
```

Blank Page

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.

The prime shift (8:00-16:00) is about three times busier than the evening shift (16:00 to midnight). In both cases, the bulk of the CPU load comes from timesharing jobs (C).

Note that the average batch job (E) used 2.9% of the CPU during the day. The average timesharing job used .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.

Also, note that although an average of 28.4 timesharing jobs were reported during the evening shift, a little more than two thirds are [1,2] jobs running DAEMON, LPTSPL, etc. The number of [1,2] jobs can be determined from a PPN report such as WCMN0.RP4 (Figure 2-5).

```
                                    AMAR WORKLOAD REPORT

    SITE:  <Put Any Title Here>

    REPORT DESCRIPTION:  SHIFT REPORT BY BATCH VS. TIMESHARING                    SYSTEM:  PATH

    INPUT FILE:  823242.DBO    (FISCAL YEAR: 82   QUARTER: 3   MONTH: 2   WEEK: 4   DAY: 2   MONDAY)
```

FROM: 0: 0: 2 ON MONDAY 15-FEB-82 TO: 8: 0: 2 ON MONDAY 15-FEB-82 INTERVAL: 7:59:59 MEASURED: 100%

GROUPED BY: BATCH (A)

SORTED BY: CPU%

JOB #	AVG JOBS	LOG IN	LOG OUT	PPN PROJ	PPN PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE--LINE	LOGIN AT DAY TIME
	22.1	13	5	******	INTERVAL	TOTALS	******	52.6	0.8	11.8	0.6	0.4							
	20.1	7	1	[????????????]		??????	??????	30.5	0.6	7.9	0.3	0.2		T	??	??	?????? ?????		
	2.0	6	4	[?????? ?????]		??????	??????	101.3	0.3	3.9	0.3	0.2		B	??		PTY??		0

FROM: 8: 0: 2 ON MONDAY 15-FEB-82 TO: 16: 0: 3 ON MONDAY 15-FEB-82 INTERVAL: 8: 0: 0 (B) MEASURED: 100%

GROUPED BY: BATCH

SORTED BY: CPU%

JOB #	AVG JOBS	LOG IN	LOG OUT	PPN PROJ	PPN PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE--LINE	LOGIN AT DAY TIME
	56.8	224	183	******	INTERVAL	TOTALS	******	57.0	37.9	265.7	41.3	18.5							
	55.7	211	171	[????????????]		??????	??????	56.8	34.5	260.4	40.7	16.9		T	??	??	?????? ?????		
	1.2	13	12	[????????????]		??????	??????	59.0	3.4	5.3	0.6	1.7		B	??	??	PTY??	0	0

(C) (E)

FROM: 16: 0: 3 ON MONDAY 15-FEB-82 TO: 0: 0: 2 ON TUESDAY 16-FEB-82 INTERVAL: 7:59:59 MEASURED: 100%

GROUPED BY: BATCH

SORTED BY: CPU%

JOB #	AVG JOBS	LOG IN	LOG OUT	PPN PROJ	PPN PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE--LINE	LOGIN AT DAY TIME
	28.4	63	111	******	INTERVAL	TOTALS	******	34.8	13.1	103.8	32.4	9.9							

Figure 2-4

2.3.4 PPN Report

Figure 2-5 is a sample of the default report WCMN0.RP4 and shows monthly resource usage by Project-Programmer Number 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 second subreport shows that there were an average of 21.6 operator [1,2] jobs logged on during prime time. These jobs used a total of 3.3% of the CPU during this period. Question marks in the charge number and program name field indicate that values of these items varied, i.e. there were several programs run by [1,2] jobs under several charge numbers.

It is also possible to get a similar report grouped only by Project Number. In addition, if a user operates under several PPN's it is possible to combine those PPN's into a single PPN group for reporting purposes. This latter feature is described in the Appendix called "Grouping PPN's for Reporting Purposes".

```
                                    AMAR WORKLOAD REPORT

    SITE:  <Put Any Title Here>                                              SYSTEM:  PATH

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

    INPUT FILE:  8232  .DBO    (FISCAL YEAR: 82   QUARTER: 3   MONTH: 2   WEEKDAYS)
```

0: 0 - 8: 0 WEEKDAYS (-HOLIDAYS) FROM: MON 25-JAN-82 TO: FRI 19-FEB-82 (20 DAYS) MEASURED: 94% = 151.77 HOURS

GROUPED BY: PPN

SORTED BY: CPU% CUTOFF: 0.50% OF CPU

JOB #	AVG JOBS	LOG IN	LOG OUT	PROJ	PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE--LINE	LOGIN AT DAY TIME
	22.3	333	194	******	INTERVAL	TOTALS	******	67.3	3.4	16.5	3.5	1.7							
	0.1	10	14	[15152	112403]	??????	??????	68.8	1.3	2.0	0.6	0.7		0		??		?????	
	19.5	82	59	[1	2]	??????	??????	39.4	0.8	8.5	0.9	0.3				?? ??		?????	
	0.0	0	3	[275	42203]	224300	??????	93.3	0.6	2.0	1.0	0.3		2	B	??			
	19.6	92	76	***SUBTOTALS THRU CUTOFF***				65.2	2.7	12.5	2.5	1.2							
	2.7	241	118	***SUBTOTALS AFTER CUTOFF**				75.7	0.7	4.0	0.9	0.5							

8: 0 - 16: 0 WEEKDAYS (-HOLIDAYS) FROM: MON 25-JAN-82 TO: FRI 19-FEB-82 (20 DAYS) MEASURED: 94% = 151.22 HOURS

GROUPED BY: PPN

SORTED BY: CPU% CUTOFF: 0.50% OF CPU

JOB #	AVG JOBS	LOG IN	LOG OUT	PROJ	PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE--LINE	LOGIN AT DAY TIME
	58.7	491	641	******	INTERVAL	TOTALS	******	63.7	31.3	208.0	57.4	18.4							
(A)	0.8	105	86	[15152	112403]	??????	??????	56.3	3.3	8.5	5.0	3.0		0		?? ??		?????	
	21.6	255	213	[1	2]	??????	??????	33.6	3.3	55.6	2.7	1.7				?? ??		?????	
	0.6	48	38	[15152	45010]	??????	??????	67.2	2.3	8.0	3.7	2.6		0		?? ??		?????	
	0.7	63	52	[20320	112161]	??????	??????	80.1	1.5	8.6	1.8	0.9		0	T	?? ??		?????	
	0.4	72	75	[12400	112136]	??????	??????	62.0	1.3	5.8	5.7	0.4		0		?? ??		?????	
	0.7	31	15	[20320	112136]	??????	??????	78.1	1.3	4.5	0.8	0.3		0	T	?? ??		?????	
	1.6	136	153	[275	43226]	??????	??????	40.5	1.3	22.8	0.6	0.1				?? ??		?????	
	0.4	136	128	[15152	46013]	??????	??????	61.5	1.1	2.0	1.7	0.9		0		?? ??		?????	
	0.5	73	67	[15152	112424]	??????	??????	129.1	1.1	5.6	3.0	0.8		0		?? ??		?????	
	0.2	50	44	[15152	112160]	??????	??????	63.4	0.8	2.9	1.4	0.5		0		?? ??		?????	
	2.1	110	101	[12400	132153]	??????	??????	79.9	0.8	3.5	1.4	0.1		0	T	?? ??		?????	
	0.9	109	103	[12400	112154]	??????	??????	51.1	0.7	4.7	1.5	0.2		0		?? ??		?????	

Figure 2-5

2.4 HOW TO RUN THE PROGRAMS

2.4.1 Data Collection

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

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

```
:SLOGIN ppn
:DEFINE WC=
WC-RUN structure:WHOWC[ppn]
```

These commands may be entered directly to OPSER to get WHOWC started the first time. The last line may be used to restart WHOWC if it has stopped because of disk parity errors or the like.

WHOWC creates an output file named WC.RAW. The output file is closed after each checkpoint interval.

2.4.2 Generating Automatic Reports

There are three programs used in the automatic reporting process - WCINC, WCUPD, and WCRPTB. WCINC and WCUPD massage the raw file output by the data collection program and create the database files. WCRPTB is the report generating program which operates on the database files. These programs are normally run as part of a nightly batch stream, WCRPTB.CTL, which is self-submitting. By using special wild carded filenames (described under the INPUT FILE.EXT = command in the Appendix called, "Report Program (WCRPT, WCRPTB, and WCRPTC) Dialogue"), it is possible to generate daily, weekly, and monthly reports through this one stream without operator intervention. The reporting program WCRPTB, keeps track of the database files on which it has already reported. It also 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 WCRPTB.CTL. See the Appendix called "Workload AMAR Batch Stream - WCRPTB.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 WCRPTB 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 four programs for generating special reports - WCRPTB, WCRPTC, WC and WCRPT. For detailed explanations of report program dialogues, see the Appendix called "Report Program (WCRPT, WCRPTB, and WCRPTC) Dialogues".

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

Use WCRPTC to report from the database only if you need more detail groups than WCRPTB can handle, for example, if you wanted to report on every program run by every user over the period of a month.

Use WC and WCRPT 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.

```
.run worptb

REPORT DESCRIPTION = primetime node and line usage (1)

INPUT  FILE.EXT = 823242.db0 (2)

OUTPUT FILE.EXT = workn1.rpt (3)

PPN 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: (7)

ANY SPECIAL MASKS OR SORT ORDERS? (Y OR N): n (8)

ID ITEM    0-0: node (9)

ID ITEM    0-1: line (10)

ID ITEM    0-2:

SORT ITEM  1-0: node (11)

SORT ITEM  1-1: jobs (12)

SORT ITEM  1-2:

SORT ITEM  2-0:

MORE REPORTS? (Y OR N): n

END OF EXECUTION
CPU TIME: 1.55  ELAPSED TIME: 14.85
EXIT

.
```

Figure 2-6 Sample WCRPTB Dialogue

1. Free form report description. This report shows the percent of time each line was used on each node. The most heavily used lines are listed first under each node. To determine the percent of time the line was used, look under the AVG JOBS column. 1.0 = 100%; .9 = 90%; etc. 0 means that each time a sample was taken, no one was using that line. Remember samples are taken every 5 minutes.

2. Daily file for February 15,1982 (FY82, third quarter, second month, fourth week, second 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 PM (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. No special cutoffs will be used.

8. No other special reporting features will be used.

9. and 10. Data is grouped by line within each individual node.

11. The major sort is by node.

12. For each node, details will be sorted by AVG JOBS (jobs). This will have the effect of listing the most heavily used lines for that node first. The "node" and "jobs" mnemonics have a default sort order implied.

```
                                    AMAR WORKLOAD REPORT

    SITE:  <Put Any Title Here>                                          SYSTEM:  PATH

    REPORT DESCRIPTION:  primetime node and line usage (1)

    INPUT FILE:  823242.DBO (2) (FISCAL YEAR: 82   QUARTER: 3   MONTH: 2   WEEK: 4   DAY: 2   MONDAY)
```

FROM: 8: 0: 2 ON MONDAY 15-FEB-82 (4) TO: 16: 0: 3 ON MONDAY 15-FEB-82 (5) INTERVAL: 8: 0: 0 (6) MEASURED: 100%

GROUPED BY: NODE LINE (9) (10)

SORTED BY: NODE JOBS (11) (12)

JOB #	AVG JOBS	LOG IN	LOG OUT	PPN PROJ	PPN PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE--LINE	LOGIN AT DAY	LOGIN AT TIME
	56.8	224	183	******INTERVAL TOTALS******				57.0	37.9	265.7	41.3	18.5								
20	1.0	0	0	[12400112206]		750223	1022	60.4	0.0	0.2	0.0	0.0	4	0	T	TI	TTY73	NODED 73	0	7:41
28	1.0	1	1	[15152 4????]		216?00	??????	52.3	1.4	8.3	2.7	1.7	4	0	T	?? ??	TTY55	NODED 55	0	
18	1.0	1	0	[15323112152]		555267	??????	44.8	0.1	1.4	0.5	0.0	4	0	T	?? ??	TTY25	NODED 25	0	8:04
	1.0	1	1	[12400132153]		750240	1022	90.7	0.4	2.3	0.9	0.1	4	0	T	??	TTY75	NODED 75	0	
31	1.0	1	0	[15152112437]		248100	??????	39.4	0.2	3.3	0.2	0.1	4	0	T	?? ??	TTY23	NODED 23	0	8:09
	0.9	9	9	[12400132???]		??????	??????	64.5	0.3	2.2	0.9	0.0		0	T	?? ??	TTY74	NODED 74	0	
17	0.9	0	0	[1	2]	133085	??????	40.0	0.2	1.6	0.1	0.0	0	54	T	?? ??	TTY102	NODED 102	-3	21:59
6	0.9	1	0	[15152112157]		216300	??????	93.0	1.0	5.7	0.7	0.3	4	0	T	?? ??	TTY17	NODED 17	0	8:15
43	0.9	1	0	[1	2]	133085	DLLSPL	29.9	0.3	20.5	0.0	0.0	0	54	T	?? ??	TTY103	NODED 103	0	8:47
49	0.8	1	0	[20320112136]		555040	??????	58.8	13.2	2.8	0.3	0.1	4	0	T	?? ??	TTY76	NODED 76	0	9:14
	0.8	2	1	[15???112171]		??????	??????	30.1	0.0	0.3	0.0	0.0	4	0	T	?? ??	TTY13	NODED 13	0	
	0.8	2	3	[20320112161]		555040	??????	40.7	1.2	10.8	2.0	0.7	4	0	T	?? ??	TTY77	NODED 77	0	
	0.7	3	1	[15152??????]		??????	??????	62.0	0.3	1.5	0.3	0.2	4	0	T	?? ??	TTY16	NODED 16	0	
	0.6	5	4	[1??????172???]		??????	??????	51.5	0.7	4.8	9.4	7.4	4	0	T	?? ??	TTY22	NODED 22	0	
	0.6	5	5	[12400172177]		750277	??????	41.8	0.1	1.2	0.4	0.0	4	0	T	?? ??	TTY106	NODED 106	0	
	0.6	2	1	[12400112206]		750223	????	48.8	0.0	0.3	0.0	0.0	4	0	T	??	TTY71	NODED 71	0	
60	0.4	1	0	[17600112137]		216410	?????	47.0	0.0	0.1	0.0	0.0	4	0	T	?? ??	TTY70	NODED 70	0	12:26
	0.4	2	1	[1??????112???]		?5????	??????	108.0	0.1	1.0	0.3	0.0	4	0	T	?? ??	TTY107	NODED 107	0	
36	0.4	1	0	[20320112143]		463395	??????	78.5	0.9	64.3	0.1	0.0	4	0	T	?? ??	TTY100	NODED 100	0	13:07
	0.3	1	0	[15152 4????]		216300	??????	22.1	0.1	0.8	0.1	0.1	4	0	T	?? ??	TTY27	NODED 27	0	
	0.3	7	7	[15152??????]		??????	??????	31.8	0.0	0.4	0.0	0.0	4	0	T	?? ??	TTY11	NODED 11	0	
	0.3	5	5	[15323112177]		555267	??????	28.8	1.5	5.9	0.1	0.0	4	0	T	?? ??	TTY37	NODED 37	0	
	0.3	9	10	[15???112???]		???2??	????	30.0	0.1	2.1	0.1	0.0	4	0	T	??	TTY35	NODED 35	0	
	0.2	3	2	[15152112???]		??????	?I????	31.2	0.2	1.5	0.4	0.0	4	0	T	?? ??	TTY5	NODED 5	0	
45	0.2	2	1	[15152??????]		271122	????	76.7	0.2	1.2	0.5	0.0	4	0	T	?? ??	TTY4	NODED 4	0	
	0.2	4	2	[15323112147]		555267	??????	161.6	0.3	1.4	0.5	0.2	4	0	T	?? ??	TTY36	NODED 36	0	
	0.2	4	2	[1537511247?]		555571	??????	141.1	0.4	0.6	0.2	0.0	4	0	T	??	TTY54	NODED 54	0	
	0.2	8	7	[??????????]		??????	??????	26.3	0.1	1.0	0.2	0.0		0	T	?? ??	TTY33	NODED 33	0	
	0.2	3	2	[15152???177]		??????	?????	29.9	0.0	0.1	0.0	0.0		0	T	?? ??	TTY30	NODED 30	0	
	0.1	3	0	[15?????????]		?5????	??????	30.6	0.0	0.1	0.0	0.0	4	0	T	?? ??	TTY45	NODED 45	0	
	0.1	2	2	[12400112???]		750???	?????	32.3	0.0	0.1	0.0	0.0	4	0	T	?? ??	TTY32	NODED 32	0	
72	0.1	1	1	[15152112435]		216114	SOS	30.1	0.0	0.1	0.0	0.0	4	0	T	TI	TTY34	NODED 34	0	15:05
39	0.1	1	1	[15152112136]		216119	OPSTAT	41.7	0.1	0.2	0.0	0.0	4	0	T	SL	TTY6	NODED 6	0	12:09
29	0.0	1	0	[1	2]	680160	PIP	31.4	0.0	0.3	0.1	0.0	4	2	T	??	TTY104	NODED 104	0	8:40

Figure 2-6 (continued)

37	0.0	1	0	[1 2]	133085	PIP	19.1	0.0	0.0	0.0	0.0	0	54	T	^C		TTY113	NODED	113	0	8:27
	0.0	2	1	[2 5]		LOGIN	32.0	0.0	0.0	0.0	0.0	0	0	T	SL		TTY26	NODED	26	0	
	0.0	2	1	[2 5]		LOGIN	32.0	0.0	0.0	0.0	0.0	0	0	T	SL		TTY7	NODED	7	0	
39	0.0	1	0	[2 5]		SYSTAT	19.3	0.0	0.0	0.0	0.0	0	0	T	TD		TTY21	NODED	21	0	11:19
70	0.0	1	1	[15323112135]	555267	??????	29.9	0.0	0.1	0.0	0.0	4	0	T	??	??	TTY140	NODEF	13	0	15:34
52	0.5	1	1	[12400112206]	750223	1022	59.4	0.1	0.5	0.2	0.0	4	0	T	T?		TTY126	NODEH	6	0	9:22
	0.2	2	2	[12400112154]	750192	????	46.0	0.1	1.0	0.1	0.0	4	0	T	??	??	TTY13?	NODEH	5	0	
6	0.0	0	1	[210400 62165]	133000	???	69.6	0.0	0.1	0.1	0.0	4	0	T	??		TTY120	NODEH	54	0	7:58
32	0.0	1	1	[12400112154]	750192	1022	49.2	0.0	0.1	0.0	0.0	4	0	T	T?		TTY130	NODEH	1	0	11:00
45	0.0	1	1	[12400112170]	750205	1022	46.1	0.0	0.2	0.3	0.2	4	0	T	TI		TTY126	NODEH	21	0	9:11
	0.2	2	2	[12400112150]	750188	????	39.8	0.0	0.6	0.1	0.0	4	0	T	?I		TTY1??	NODEI	3	0	
64	0.1	1	1	[12400112205]	750222	????	53.0	0.0	0.4	0.3	0.2	4	0	T	?I		TTY134	NODEI	35	0	9:59
21	1.0	0	0	[1 2]	133085	OPSER	8.0	0.1	2.6	0.0	0.0	0	1	T	??	??	TTY115	NODEM	21	-4	23:21
	1.0	1	0	[1 2]	133085	??????	21.3	0.0	0.1	0.0	0.0	0	54	T	^C	??	TTY117	NODEM	20		
	0.9	3	5	[17???172???]	?5????	??????	83.5	0.2	0.8	0.9	0.0	4	0	T	??	??	TTY1??	NODEM	0	0	
	0.9	2	1	[17???112???]	?5????	??????	140.5	0.3	0.4	0.1	0.0	4	0	T	??	??	TTY12?	NODEM	37	0	
	0.8	11	9	[??????????]	??????	??????	45.5	0.7	3.2	1.8	0.4		0	T	??	??	TTY1??	NODEM	1	0	
	0.8	5	3	[???0???????]	??????	??????	44.7	0.5	1.6	2.1	0.2		0	T	??	??	TTY1??	NODEM	3	0	
	0.8	5	4	[????? ?????]	??????	??????	50.2	0.1	0.8	0.3	0.1			T	??	??	TTY12?	NODEM	25	0	
	0.7	8	6	[???0???????]	??????	??????	49.9	0.2	1.2	0.6	0.3		0	T	??		TTY1??	NODEM	2	0	
	0.7	3	2	[????? ?????]	133105	??????	40.7	0.1	0.5	0.2	0.1	4	1	T	??	??	TTY1??	NODEM	24	0	
	0.6	3	3	[12400172???]	750???	1022?	71.5	0.3	1.4	0.7	0.0	4	0	T	??	??	TTY13?	NODEM	5	0	
	0.6	11	9	[12400172????]	750???	??????	47.9	0.2	1.7	0.7	0.0	4	0	T	??		TTY1??	NODEM	4	0	
	0.5	4	3	[??????????]	??????	?????	41.1	1.5	25.0	0.2	0.0			T	??	??	TTY1??	NODEM	23	0	
	0.5	2	2	[???0???????]	??????	??????	56.6	1.1	6.6	4.1	1.0		0	T	??	??	TTY1??	NODEM	6	0	
	0.5	2	0	[7700112136]	133100	??????	25.9	0.0	0.2	0.0	0.0	4	74	T	??	??	TTY13?	NODEM	31	0	
	0.5	5	3	[???0???????]	??????	??????	65.9	1.5	4.0	2.9	1.4		0	T	??	??	TTY1??	NODEM	10	0	
	0.4	7	6	[???0???????]	??????	??????	46.8	0.3	1.6	1.0	0.6		0	T	??	??	TTY13?	NODEM	7	0	
	0.3	3	3	[12400112154]	750192	1022	52.5	0.1	0.7	0.6	0.4	4	0	T	??		TTY1??	NODEM	30	0	
	0.3	4	4	[??????????]	133105	??????	32.7	0.4	7.7	0.1	0.0	4	1	T	??	??	TTY12?	NODEM	22	0	
	0.3	2	3	[12400172???]	750???	??????	48.1	0.1	0.6	0.2	0.0	4	0	T	??		TTY1??	NODEM	11	0	
	0.2	2	1	[17???112???]	?5????	1022??	133.9	0.1	0.3	0.3	0.2	4	0	T	T?	??	TTY1??	NODEM	14	0	
73	0.0	1	0	[16016112136]	555114	1022C	118.8	0.0	0.3	0.0	0.0	4	0	T	TI		TTY137	NODEM	36	0	15:37
70	0.0	1	1	[12400132150]	750244	??????	56.3	0.0	0.2	0.2	0.0	4	0	T	??	??	TTY137	NODEM	13	0	15:18
64	0.1	1	0	[15323112150]	555267	TIP	26.8	0.0	0.1	0.0	0.0	4	0	T	T?		TTY123	NODEN	23	0	14:54
65	0.1	1	0	[15323112150]	555267	RLPSPL	13.3	0.0	0.2	0.0	0.0	4	0	T	??	??	TTY136	NODEN	24	0	14:55
32	0.1	1	1	[7700112151]	133100	?????	30.0	0.2	0.4	0.0	0.0	4	0	T	??		TTY122	NODEN	2	0	11:46
	0.0	5	4	[15152???????]	2??700	??????	31.3	0.0	0.1	0.0	0.0	4	0	T	??	??	TTY1??	NODEN	4	0	
	25.3	23	29	[??????????????]	??????	??????	47.1	5.9	52.3	1.9	2.2				??	??	??????		0		

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 WCRPT according to the general directions in the Appendix called "Report Program (WCRPT, WCRPTB, and WCRPTC) Dialogue", specifying an output filename of the form yyqmwd.ext where yyqmwd represents yesterday's fiscal date (year, quarter, month, week, day) and ext is not .IN0, .IN1, .RP?, .EXE, or .RAW. Before you run WCRPT, make sure that there are two files for yesterday named yyqmwd.IN0 and yyqmwd.IN1, as these are the input files required by WCRPT. If an hourly or higher level report is needed, the WCRPTB 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:

1. Run WC which will produce files WC.IN0 and WC.IN1.

2. Run WCRPT according to general directions in the Appendix called "Report Program (WCRPT, WCRPTB, and WCRPTC) Dialogue", specifying an output filename of the form WC.ext, where .ext is not .IN0, .IN1, .RP?, .EXE, or .RAW.

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 WCINC 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 WCRPTB.CTL

2.5.1 Overview Of WCRPTB.CTL

This stream runs daily. It takes the workload data which has been collected by WHOWC since the last time WCRPTB.CTL ran, preprocesses it, updates the database, and produces daily reports (and weekly and monthly reports if appropriate). See the Appendix called "Workload AMAR Batch Stream - WCRPTB.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 an OPSER subjob to run the WHOWC program and a permanently mounted disk area with at least 5K blocks available for raw data. The average number of simultaneous users is the key factor determining daily file size. The size of your workload database will probably range between 6K blocks and 30K blocks. 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 WCRPTB.CTL in the submit queue, set to run /AFTER:0:30:0.

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 WCRPTB.CTL unless the submit queue entry is destroyed. Each major step is checkpointed.

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 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), snapped (S), or obtained from the performance meter (PM). 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. Performance meter variables are available only on the KL10. For a discussion of the performance meter and how to use it, see the METER monitor call in the DECsystem-10 Monitor Calls Manual.

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-10 ITEMS

ACT SWAP RATIO
ACSR - Active Swapping Ratio
(S)

Active Swapping ratio is a ratio of the amount of memory currently in use by runnable jobs to the total amount of user memory available.

ACTV JOB % USR CR
ACCR - User Core Occupied by Active Jobs (Percent)
(S)

User core occupied by active jobs is the amount of physical core (less the monitor low and high segments) divided into the amount of core occupied by jobs not in long term wait states such as SL, HB, etc. or in a terminal input wait state (TI and JB.UOA clear). Shared segments are counted only once.

ALL JOBS % USR CR
PHCR - User Core Occupied by All Jobs (Percent)
(S)

User core occupied by all jobs is obtained by dividing the amount of core occupied by all jobs by the total amount of physical core available (less the monitor low and high segments). Shared segments are counted only once.

% AMAR CLK TIME
XAMT - AMAR Measured Time (Percent of Realtime)
(M)

AMAR measured time is the percentage of time that the Data Collection Program was running.

% AMAR TI CPU n UP
CPA0, CPA1 - CPUn Uptime
(M)

CPUn uptime is the percent of time measured by AMAR that the CPU was up. CPUn uptime is recorded separately for each CPU within the system. For a multiprocessor configuration, CPA0 is recorded for the first CPU, CPA1 for the second, etc.

AVG JOB SIZE
AVJS - Average Job Size (Pages)
(S)

Average job size is the sum of the sizes, measured in pages, of all jobs logged into the system divided by the number of jobs. The null job is not included. Shared segments are counted only once.

AVG SCHD RSP TIME
AVRT - Average Scheduler Response Time (Ms)
(M)

Average scheduler response time is the average number of milliseconds that elapsed between the issuance of a command to execute a program or remove a job from a TTY input wait state and selection of the job by the scheduler. This is derived from two accumulators maintained by the monitor, one which counts milliseconds and one which counts the number of responses by the CPU.

CACHE SWEEP/SEC
CPS0, CPS1 - CPUn Cache Sweep Rate (Sweeps/Sec)
(M)

Cache sweep rate is the total number of cache sweeps executed on the CPU per second. It is obtained from an accumulator maintained by the monitor. Cache sweeps are memory updates by cache initiated primarily by direct memory accesses, or context switches.

CPUn cache sweep rate is recorded separately for each CPU within the system. For a multiprocessor configuration, CPS0 is recorded for the first CPU, CPS1 for the second, etc. Cache sweeps are only recorded for KL10's.

CHAN n % TIM BUSY
CHn0, CHn1 - Channel Busy Time (Percent)
(PM)

Channel busy time is the percent of real time this channel was busy. n = 0 to 7.

CHAN nn WAIT Q
CTWQ - Channel Transfer Wait Queue Length
(S)

The transfer wait queue length is the count of commands awaiting channel action.

The length of a disk channel transfer wait queue is recorded separately for each channel.

CONTEXT SWTS/SEC
CPC0, CPC1 - Context switches per second
(M)

Context switches per second are recorded separately for each CPU. In a multiprocessor configuration, CPC0 is recorded for the first CPU, CPC1 for the second, etc.

% CPU UTIL
_CPU - CPU Utilization

CPU utilization is derived by the reporting program by subtracting idle time (the average of CPI0 and CPI1) from 100%. It includes lost and overhead time, user program time, and some embedded priority interrupt (PI) processing time. Since this is a derived item, it cannot be obtained directly from the database.

See CPU% in the Appendix called "Valid Grouping and/or Sort Items" for the variable representing individual job CPU utilization.

% IDLE TIME
CPI0, CPI1 - CPUn Idle Time (Percent)
(M)

Idle time is the amount of CPU time the null job was running minus lost time.

At the sample group interval, CPUn idle time is recorded as a percentage of wall clock time over which the item is measured.

CPUn idle time is recorded separately for each CPU within the system. For a multiprocessor configuration, CPI0 is recorded for the first CPU, CPI1 for the second, etc.

JOBS BLK IO Q
IOWP - Jobs in Noninteractive IO Queues
(S)

The number of jobs in noninteractive I/O queues is the count of jobs waiting for paging (PI state), disk I/O (DI state), or input and output to a device (IO state). The number of jobs in the several states are summed.

JOBS IN RUN QUEUE
NRJR - Number of Jobs in the Run Queue
(S)

The number of jobs in the run queue is the number of jobs available for selection (RN state) to run in the CPU. A job will be counted whether it is core resident or swapped.

JOBS LOGGED IN
JLOG - Jobs Logged in Concurrently
(S)

The number of jobs logged in concurrently is the total number of jobs logged into the system not counting the null job.

JOBS TTY IO Q
TIOW - Jobs in Interactive IO Queue
(S)

The number of jobs in the interactive IO queue is the number of jobs waiting for terminal input or output (TI state).

LINES IN USE
TTYU - Terminal Lines in Use Concurrently
(S)

The number of terminal lines in use is a count of all lines (CTY, remote, local, slave, etc.) in use except PTY's.

% LOST TIME
CPL0, CPL1 - CPUn Lost Time (Percent)
(M)

Lost time is the amount of CPU time a runnable job was waiting to be swapped in while the null job ran.

At the sample group interval, CPUn lost time is recorded as a percentage of wall clock time over which the item is measured.

CPUn lost time is recorded separately for each CPU within the system. For a multiprocessor configuration, CPL0 is recorded for the first CPU, CPL1 for the second, etc.

% MON FR CORE USED
PFCU - Free Core in Use (Percent)
(S)

Free core in use is the percentage of 4-word free core blocks in use by the monitor. This is determined by examining two accumulators maintained by the monitor; one contains the current amount of core in use and the other contains the total amount of core allocated to the monitor for use.

MT name BLKS/SEC
MTIO - Magtape I/O (Blocks/Second While Assigned)
(M)

Magtape I/O rate is the number of equivalent disk blocks read and written per second from a drive and is calculated based on the length of time the drive was assigned not the amount of time AMAR was sampling. Internally the monitor maintains the number of frames read and written from a tape unit on a mount per job basis, from the time the tape unit was mounted to the time the tape unit was unloaded. A frame consists of 7 or 9 bits of information (depending on the number of channels on the drive) with 1 bit written in each channel. Frames are converted by the data collection program into blocks of 128 words each.

Magtape I/O rate is recorded separately for each magnetic tape unit active.

While Magtape IO rate is a metered item in so far as accurate counts are maintained by the monitor, it can be sensitive to sample interval. A long sample interval could result in job overlapping between sampling, while a shorter interval would more likely catch the appropriate values before they were reset for a new job.

MTAS ASSIGNED
MTAU - Magtape Units in Use Concurrently
(S)

Magtape units in use is the total number of magnetic tape units assigned to jobs at sample time.

% OVHD TIME
CP00, CP01 - CPUn Overhead Time (Percent)
(M)

Overhead time is the amount of CPU time spent in clock queue processing, short command processing, swapping and scheduling, context switching and interrupt servicing.

At the sample group interval, CPUn overhead time is recorded as a percentage of wall clock time over which the item is measured.

CPUn overhead time is recorded separately for each CPU within the system. For a multiprocessor configuration CP00 is recorded for the first CPU, CP01 for the second, etc.

PGS USER MEMORY
UMEM - User Memory Available (Pages)
(S)

User memory available is the total amount of physical memory less the amount used by the monitor high and low segments and locked (LK) jobs.

PI n % TIME BUSY
PIn0, PIn1 - Priority Interrupt Level Busy Time (Percent)
(PM)

Priority interrupt level busy time is the percent of real time spent at the level (n = 0 to 7). This measurement uses the the monitor round robin mechanism specific to version 7 of the monitor.

PK name BLKS/SEC
LUIO - Logical Unit I/O Rate (Blocks/Second)
(M)

I/O rate is the number of disk blocks read and written per second by user jobs. Swapping and monitor overhead are omitted from this count.

I/O rate is computed separately for each individual logical disk pack.

See READS and WRITE in the Appendix called "Valid Grouping and/or Sort Items" for the variables representing individual job disk reads and writes.

%PK name FREE SPC
LUFS - Logical Unit Free Space (Percent)
(S)

Logical unit free space is the percentage of unused blocks on a logical pack.

Free space is recorded separately for each logical pack. Both

public and private structures are sampled.

PK name SWPS/SEC
LUSW - Logical Unit Swapping Rate (Blocks/Second)
(M)

Logical unit swapping rate is the number of disk blocks read and written per second to the pack while swapping user jobs.

Swapping rate is computed separately for each logical disk pack which has swapping space.

PK name WAIT Q
LUWQ - Logical Unit Position Wait Queue
(S)

The position wait queue is the count of jobs awaiting disk control.

The length of the position wait queue is recorded separately for each logical pack. Both public and private disk structures are sampled.

% RN JOB IN MEM
PRJC - Runnable Jobs in Memory (Percent)
(S)

Runnable jobs in memory is the percent of jobs in the run queues (R1, R2 and HPQ) which are resident in core.

SCN INTR RCVS/SEC
SCRV - SCNSER Interrupts Received/Second
(M)

The number of interrupts received by SCNSER per second.

SCN INTR XMTS/SEC
SXMT - SCNSER Interrupts Transmitted/Second
(M)

The number of interrupts transmitted by SCNSER per second.

SLV CACH SWEEP/SEC - See CACHE SWEEP/SEC.

SLV CHAN n % BUSY - See CHAN n % TIM BUSY.

SLV CTXT SWTS/SEC - See CONTEXT SWTS/SEC.

SLV IDLE TIME - See % IDLE TIME.

SLV LOST TIME - See % LOST TIME.

SLV OVHD TIME - See % OVHD TIME.

SLV PI n % BUSY - See PI n % TIM BUSY.

SLV USER UUOS/SEC - See USER UUOS/SEC.

% SWAP SPC LEFT
SWPS - Swapping Space Left (Percent)
(S)

Swapping space left is the percentage of 1k blocks of virtual core remaining as swapping space.

SWAPPING BLKS/SEC
SWIO - Swapping IO Rate (Blocks/Sec)
(M)

Swapping IO rate is the number of disk blocks read and written per second while swapping user jobs.

SYSTEM RELOADS
XRLD - Number of System Reloads
(M)

Number of System reloads is the number of times the system was rebooted.

% SYSTEM UPTIME
XUPT - System Uptime (Percent of Realtime)
(M)

System uptime is the percentage of time that the system was known to be operational.

%TY CHNK IN USE
PTCU - TTY Chunks in Use (Percent)
(S)

The percent of monitor TTY chunks in use.

UN name BLKS/SEC
PUIO - Physical Unit IO Rate (Blocks/Second)
(M)

Physical unit I/O rate is the number of disk blocks read and written per second by user jobs. Swapping blocks and monitor overhead are omitted from this count.

I/O rate is computed separately for each physical disk unit.

See READS and WRITE in the Appendix called "Valid Grouping and/or Sort Items" for the variables representing individual job disk reads and writes.

%UN name FREE SPC
PUFS - Physical Unit Free Space (Percent)
(S)

Physical unit free space is the percentage of unused blocks on a disk pack.

Free space is recorded separately for each non-fixed head disk unit. Both public and private structures are sampled.

UN name SWPS/SEC
PUSW - Physical Unit Swapping Rate
(M)

Physical unit swapping rate is the number of disk blocks read and written per second from the pack while swapping user jobs.

Swapping rate is recorded separately for each physical pack which has swapping space.

UN name WAIT Q
PUWQ - Physical Unit Position Wait Queue
(S)

The position wait queue is the count of jobs awaiting disk control.

The length of the position wait queue is recorded separately for each non-fixed head disk unit. Both public and private disk structures are sampled.

USER DSK BLKS/SEC
UDIO - User Disk IO Rate (Blocks/Sec)
(M)

User disk I/O rate is the number of disk blocks read and written per second by user jobs. Swapping blocks and monitor overhead are omitted from this count.

See READS and WRITE in the Appendix called "Valid Grouping and/or Sort Items" for the variables representing individual job disk reads and writes.

USER UUOS/SEC
CPU0, CPU1 - CPUn UUO Execution Rate (UUOs/Sec)
(M)

UUO execution rate is the total number of monitor calls, in both executive and user mode, executed on the CPU per second.

CPUn UUO execution rate is recorded separately for each CPU utilized within the system. For a multiprocessor configuration, CPU0 is recorded for the first CPU, CPU1 for the second, etc.

See UUOS in the Appendix called "Valid Grouping and/or Sort

Items" for the variable representing the individual job UUO count.

VIR MEM FAULT/SEC
VMPF - Virtual Memory Paging Rate (Faults/Sec)
(M)

Virtual memory paging rate is the number of page faults per second.

Virtual memory page fault rate is only recorded for systems with the virtual memory option enabled.

APPENDIX B

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 C

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 D

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 <u>not</u> 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, for example, AVRT (average scheduler response time.) 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.

Blank Page

APPENDIX E

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 not 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, 1982, specify 820204:0200.

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'LUWQDSKE0" 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 DSKE. 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 F

REPORT PROGRAM (WCRPT, WCRPTB, AND WCRPTC) DIALOGUE

This appendix explains, for each reporting program each possible prompt, its valid response, and any defaults.

Error messages which may occur while running WCRPTB, WCRPTC or WCRPT are listed in AMAR-10 Error Messages. The dialogue for WCRPTC is identical to that of WCRPTB. The WCRPTC program just contains larger internal arrays for more detail processing. However, some prompts and/or valid responses differ between WCRPTB and WCRPT. In these cases, WCRPTB will be discussed first and "WCRPTB only", or "WCRPT 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 WCRPTB, WCRPTC and WCRPT dialogues. Prompts are listed in approximately the same order that they appear in the dialogue.

.RUN WCRPTB (or .RUN WCRPTC, .RUN WCRPT)

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 = (WCRPTB 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 WCRPTB. 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 = (WCRPTB only. See WCRPT 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 WCRPTB.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 = (WCRPT only. See WCRPTB above).

Requests the name and extension of the desired report file.

Valid Responses:

filename.ext

Before you run WCRPT, 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 WCRPT. WCRPT will not prompt for an input filename; it will assume the existence of .IN0 and .IN1 files. .IN0 and .IN1 files are normally created by WCINC in the nightly batch stream. They may also be created for today's data by running WC before you run WCRPT. As a result, there may be .IN0 and .IN1 files for both yesterday and today. If you have run WC to preprocess today's data, the filename will be WC. Otherwise, you should do a directory in order to determine the precise 6 character filename of the .IN0 and .IN1 files which you want.

The filename must be identical to the filename of your input files which have extensions .IN0 and .IN1. 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.

PPN GROUPING FILE.EXT =

Requests name and extension of desired PPN grouping file.

Valid Responses:

Carriage return or filename and extension of a correctly formatted PPN grouping file. (See the Appendix called "Grouping PPN's for Reporting Purposes".)

Default: Carriage return means do not group PPN's.

ENTER DESIRED START AS HH MM: (WCRPTB only. See WCRPT 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: (WCRPT only. See WCRPT 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: (WCRPTB only. See WCRPT 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: (WCRPT only. See WCRPTB above).

Requests the time (and relative day) of the end of the last reporting interval.

Valid Responses:

Same format as the start time for WCRPT.

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: (WCRPTB only. See WCRPT 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: (WCRPT only. See WCRPTB above).

Requests the size of the reporting interval.

Valid Responses:

A time and date specification like WCRPT'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 WCRPT, not to WCRPTB.

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.

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 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 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 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 G

VALID GROUPING AND/OR SORT ITEMS

The following items (1) may be used for both grouping and sorting when using the WCRPT, WCRPTB, and WCRPTC 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	000000000377	Job number
PPN	A	777777777777	Project/programmer number
PROJ	A	777777000000	Project half of PPN
PROG	A	000000777777	Programmer half of PPN
CHARG	A	777777777777	Charge number
PRGRM	A	777777777777	Program name
HPQ (2)	D	374000000000	High priority queue number
CLASS(2)	A	003740000000	Scheduler class number
SCD(2)	A	003740000000	Scheduler class number
CLS(2)	A	003740000000	Scheduler class number
CATEG	A	000037400000	LOGIN category
CT	A	000037400000	LOGIN category
BATCH	A	400000000000	Batch or timesharing
STATE	A	777777777777	Job state
TTY	A	777777777777	Terminal name
NODE	A	777777777777	Node name
LINE	A	000000377400	Line number on node
LOGIN	A	777777777777	Date/time of login
MISC(3)	A	777777777777	Miscellaneous identification
NOW	A	777777777777	Date/time stamp of sample

The following items should be used only for sorting:

JOBS	D	777777777777	Average simultaneous jobs
LIO	D	777777777777	LOGIN's and KJOB's
IN	D	777777000000	LOGIN's during interval
OUT	D	000000777777	KJOB's during interval
KCT(4)	D	777777777777	Kilo-core ticks
CPU%(5)	D	777777777777	Percent of processor used
UUOS(6)	D	777777777777	UUO's/second
READS(7)	D	777777777777	Disk blocks read/second

WRITE(7)	D	777777777777	Disk blocks written/second

Notes:

1. Items (and valid synonyms) are listed as they appear from left to right across the report.

2. Since high priority queue number overrides scheduler class number in the report, HPQ should always be used immediately before CLASS if CLASS is going to be used for grouping and sorting.

3. The code MISC refers to a word which contains six items (BATCH, HPQ, SCD, CATEG, LINE, and JOB). This code should only be used with a mask as described in the Appendix called "Special Masks and Sort Orders".

4. Kilo-core ticks is a valid sort item, although not reported directly. It may be derived by multiplying PAGES X .5 X CPU% X interval size expressed in ticks. Note that current program logic does not allow sorting by jobsize (PAGES), a derived item.

See the Appendix called "System AMAR Item Definitions" for references 5-7:

5. See "% CPU UTIL" for the variable representing system wide CPU utilization.

6. See "USER UUOS/SEC" for the variable representing the system wide UUO count.

7. See "USER DSK BLKS/SEC" for the variable representing system wide reads and writes. See "PK name BLKS/SEC" and "UN name BLKS/SEC" for the variables representing system wide reads and writes on individual disk packs and physical units.

APPENDIX H

GROUPING PPN'S FOR REPORTING PURPOSES

When using the WCRPT, WCRPTB, or WCPRTC programs, arbitrary groups of PPN's may be combined for reporting purposes through the use of a PPN-grouping file. Figure H-1 shows a dialogue using a PPN-grouping file and a portion of the report it generated.

Each record in a PPN-grouping file has a "before" PPN and an "after" PPN.

When WCRPT finds a PPN in the data which matches one of your "before" PPN's, it replaces that PPN with the corresponding "after" PPN.

If desired, the last "before" PPN in your list may be 0,777777, a special PPN which "matches" any PPN not found earlier in the list and allows you to create an "other" category.

If you have no "other" category, any PPN's not found in the "before" list will be left unchanged.

Both "before" and "after" PPN's must be octal, and not greater than 377777, 777777.

It is probably a good idea to make "group" PPN's distinctively different from actual PPN's.

The following sample PPN-grouping file will group [123,456], [234,567], and [345,670] as [0,1]; [456,701] and [567,012] as [0,2]; [1,2] as [1,2]; and all other PPN's as [0,3]. Note that "before" and "after" PPN's are entered without brackets, separated by a blank. "Wildcarding" with question marks and asterisks is not supported.

```
123,456 0,1
234,567 0,1
345,670 0,1
456,701 0,2
567,012 0,2
1,2 1,2
0,777777 0,3
```

```
.ru worptb

REPORT DESCRIPTION = report splitting [1,2] usage from all other usage (1)

INPUT  FILE.EXT = 823242.db0 (2)

OUTPUT FILE.EXT = workpp.rpt (3)

PPN GROUPING FILE.EXT = ppn.fil (4)

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

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

ANY SPECIAL MASKS OR SORT ORDERS? (Y OR N): n (8)

ID ITEM    0-0: ppn (9)

ID ITEM    0-1:

SORT ITEM  1-0:

MORE REPORTS? (Y OR N): n

END OF EXECUTION
CPU TIME: 1.35  ELAPSED TIME: 11.67
EXIT

.ty ppn.fil
1,2 1,2
0,777777 0,7   (10)
```

Dialogue Using a PPN Grouping File

Figure H-1

1. Free form report description. This report will summarize usage in two detail lines, one showing all [1,2] jobs grouped together and the other showing all other jobs grouped together.

2. Daily file for February 15,1982 (FY82, third quarter, second month, fourth week, second day.)

3. Report filename.

4. A special PPN grouping file will be used. See 10. below for the listing of this file.

5. The report will start at the beginning of the 823242.db0 file and stop at the end of it.

6. The report will contain subreport intervals of 8 hours each.

7. All detail lines (in this case a maximum of 2 per subreport interval) will be shown. No cutoffs will be used.

8. No other special reporting features will be used.

9. The data will be grouped and sorted by PPN.

10. The PPN grouping file. The group of [1,2] jobs will show up as [1,2] on the report. The group of all other jobs will show up as [,7].

```
                                        AMAR WORKLOAD REPORT

    SITE:  <Put Any Title Here>                                                    SYSTEM:  PATII

    REPORT DESCRIPTION:  report splitting [1,2] usage from all other usage (1)

    INPUT FILE:  823242.DBO (2) (FISCAL YEAR: 82   QUARTER: 3   MONTH: 2   WEEK: 4   DAY: 2   MONDAY)
```

(5) (5) (6)

FROM: 0: 0: 2 ON MONDAY 15-FEB-82 TO: 8: 0: 2 ON MONDAY 15-FEB-82 INTERVAL: 7:59:59 MEASURED: 100%

GROUPED BY: PPN

SORTED BY: PPN (9)

JOB #	AVG JOBS	LOG IN	LOG OUT	PPN PROJ	PPN PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE--LINE	LOGIN AT DAY TIME
	22.1	13	5	******	INTERVAL TOTALS	******		52.6	0.8	11.8	0.6	0.4							
	2.1	12	4	[	7]	??????	??????	89.6	0.4	4.7	0.4	0.3		??			?TY???	?????	
	20.0	1	1	[1	2]	??????	??????	23.5	0.5	7.1	0.1	0.2		??	??		??????	?????	

(10)

FROM: 8: 0: 2 ON MONDAY 15-FEB-82 TO: 16: 0: 3 ON MONDAY 15-FEB-82 INTERVAL: 8: 0: 0 MEASURED: 100%

GROUPED BY: PPN

SORTED BY: PPN

JOB #	AVG JOBS	LOG IN	LOG OUT	PPN PROJ	PPN PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE--LINE	LOGIN AT DAY TIME
	56.8	224	183	******	INTERVAL TOTALS	******		57.0	37.9	265.7	41.3	18.5							
	35.4	214	177	[	7]	??????	??????	59.6	34.9	194.8	39.8	18.0		??	??		??????	?????	
	21.5	10	6	[1	2]	??????	??????	25.7	2.9	70.9	1.5	0.6		??	??		??????	?????	

FROM: 16: 0: 3 ON MONDAY 15-FEB-82 TO: 0: 0: 2 ON TUESDAY 16-FEB-82 INTERVAL: 7:59:59 MEASURED: 100%

GROUPED BY: PPN

SORTED BY: PPN

JOB #	AVG JOBS	LOG IN	LOG OUT	PPN PROJ	PPN PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE--LINE	LOGIN AT DAY TIME
	28.4	63	111	******	INTERVAL TOTALS	******		34.8	13.1	103.8	32.4	9.9							

Figure H-1 (continued)

Blank Page

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 three digits of the charge number or the last two characters of the program name. This is possible when using the WCRPT, WCRPTB, or WCRPTC 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 items stored in SIXBIT (such as CHARGE, PRGRM, STATE, TTY, and NODE) 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 the charge number is 777777000000. The mask for the last 2 characters of the program name is 000000007777, which may be abbreviated as 7777, since leading zeroes are implied.

Six of the items (JOB, BATCH, LINE, HPQ, SCD, and CATEG) are all packed together in a single word known as MISC. 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 worptb

REPORT DESCRIPTION = report by last two digits of program name (1)

INPUT  FILE.EXT = 823242.db0 (2)

OUTPUT FILE.EXT = worksm.rpt (3)

PPN 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: 15 (7)
ENTER CPU% CUTOFF: (8)

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

ID ITEM    0-0: prgrm (10)
ID MASK    0-0: 000000007777 (11)
ID ORDER   0-0:

ID ITEM    0-1:

SORT ITEM  1-0: cpu% (12)
SORT MASK  1-0:
SORT ORDER 1-0:

SORT ITEM  1-1:

SORT ITEM  2-0:

MORE REPORTS? (Y OR N): n (13)

END OF EXECUTION
CPU TIME: 1.09  ELAPSED TIME: 13.97
EXIT
```

Dialogue Using a Special Mask
Figure I-1

1. Free form report description. This report will summarize usage by program, grouping the programs by the last two digits of the program name.

2. Daily file for February 15,1982 (FY82, third quarter, second month, fourth week, second 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 PM (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. At most 15 detail lines on programs grouped by their last two digits will be printed.

8. No CPU cutoff will be used. Up to 15 detail lines will be printed regardless of how much CPU the detail line contains. Note that since the report is sorted by CPU percent (12. below), the top 15 users of the CPU will be shown on the report.

9. The special mask feature will be used.

10. The data will be grouped by program (prgrm) name.

11. Program names will also be grouped by their last two digits. A pair of 0's means ignore the corresponding character, a pair of 7's means use the character for grouping.

12. The detail lines will be listed with the heaviest CPU users first.

13. If "y" had been specified, the entire program dialogue would be repeated and you would be able to specify more reports.

```
                                    AMAR WORKLOAD REPORT

    SITE:  <Put Any Title Here>                                          SYSTEM:  PATH

    REPORT DESCRIPTION:  report by last two digits of program name (1)

    INPUT FILE:  823242.DBO (2) (FISCAL YEAR: 82   QUARTER: 3   MONTH: 2   WEEK: 4   DAY: 2   MONDAY)
```

(4) (5) (6)

FROM: 8: 0: 2 ON MONDAY 15-FEB-82 TO: 16: 0: 3 ON MONDAY 15-FEB-82 INTERVAL: 8: 0: 0 MEASURED: 100%

GROUPED BY: (10) PRGRM C00000007777 (11)

SORTED BY: CPU% (12) CUTOFF: 15 LINES (7)

JOB #	AVG JOBS	LOG IN	LOG OUT	PPN PROJ	PPN PROG	CHARGE NUMBER	PRGRM NAME	PAGES (AVG)	CPU%	UUOS /SEC	READS /SEC	WRITES /SEC	SCD CLS	CT	B	STATE	TTY	NODE--LINE	LOGIN DAY	LOGIN TIME
	56.8	224	183	******INTERVAL TOTALS******				57.0	37.9	265.7	41.3	18.5								
	24.6	131	124	[????????????]		??????	????	54.0	13.2	183.2	29.1	13.9				?? ??	??????	?????		
	0.3	1	0	[???????????]		??????	COBOL	58.1	12.6	0.8	0.2	0.2	4	O	T	??	TTY??	NODED	O	
32	0.0	1	1	[15152112157]		216300	0010DE	64.0	2.9	0.1	0.1	0.0	5	O	B	R2	PTY12		O O	8:23
	1.7	16	11	[???????????]		??????	DIRECT	42.5	2.2	11.2	2.7	0.6			T	?? ??	??????	?????		
	0.4	0	2	[???????????]		??????	BACKUP	22.6	1.1	7.5	0.8	1.5				?? ??	?TY??	?????	O	
	1.4	6	3	[15375112477]		555571	1022F0	141.6	0.9	1.4	0.5	0.0	4	O	T	?? ??	??????	?????	O	
	4.7	2	0	[???????????]		??????	???SPL	30.2	0.6	23.4	0.3	0.0			T	?? ??	?TY???	?????		
6	0.1	0	0	[15152112157]		216300	0020CC	113.6	0.6	0.6	0.3	0.1	4	O	T	??	TTY17	NODED	17 O	8:15
1	1.0	0	0	[1	2]		FILDAE	14.0	0.5	2.6	0.2	0.0	-1	O	T	?? ??	DET313		O -4	23:18
	2.1	0	0	[?O?????????]		??????	????ON	43.4	0.5	7.2	0.3	0.3			T	?? ??	??????	?????		
	2.0	8	6	[????????177]		??????	????C	68.1	0.3	2.7	0.3	0.1	4		T	?? ??	???17?	?????		
	0.3	0	0	[15152 47777]		2167?O	?O??HR	71.6	0.3	1.8	0.7	0.5	4	O	T	?? ??	?????	?????	O	
	0.1	0	2	[??O???????]		13310?	????T	29.8	0.2	0.4	0.0	0.0	4		T	??	TTY12?	NODE?	O	
	0.1	1	1	[12400132150]		750244	TN0003	55.6	0.2	1.4	1.6	0.0	4	O	T	??	TTY17?	NODEM	O	
	0.4	6	6	[???????????]		??????	????NT	40.5	0.2	1.7	1.2	0.7				?? ??	?TY???	?????		
	39.2	172	156	***SUBTOTALS THRU CUTOFF***				56.7	36.4	245.8	38.4	17.9								
	17.7	52	27	***SUBTOTALS AFTER CUTOFF**				62.2	1.5	19.8	2.9	0.6								

Figure I-1 (continued)

Blank Page

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. It also checks to see if the data collection program is running. If not the operator is requested to restart the data collection program.

ST040: Checks to se e 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 ST170.

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.

```
;SUBSTITUTIONS SHOULD BE MADE AS FOLLOWS FOR ALL WORDS ENCLOSED IN ANGLE BRACKETS "<>"
;<DB-STRUC> IS THE NAME OF THE PRIMARY (PRODUCTION) DISK PACK
;WHICH CONTAINS THE AMAR DATABASE AND ASSOCIATED FILES.
;<AMAR-PPN> IS THE PPN WHICH CONTAINS THE PRIMARY AMAR DATABASE AND ASSOCIATED
;FILES.
;<DC-STRUC> IS THE NAME OF THE PACK WHICH CONTAINS THE DATA COLLECTION
;PROGRAMS, RAW FILES, AND STREAMS - IT MAY BE THE SAME OR DIFFERENT FROM
;<DB-STRUC>.
;<SYS-ID> IS THE 4 CHARACTER SYSTEM CODE.
;<DC-RTEN> IS THE RETENTION PERIOD IN DAYS FOR DAILY RAW FILES
;ON THE <DC-STRUC>.
;<DB-RTEN> IS THE RETENTION PERIOD IN DAYS FOR DAILY RAW FILES
;ON THE <DB-STRUC>.
;<BKUP-STRUC> IS THE NAME OF THE PACK USED TO BACK UP THE AMAR DATABASE
;AND ASSOCIATED FILES.
;*******************************************************************************
;*******************************************************************************
:AMAR.CTL CAN RESTART ONLY AT RESUBMIT STEP ST180

;/CHARGE:000000

ST000:: !******** MOUNT THE PRODUCTION PACK ********!

.SET WATCH NONE
.SET DSKFUL PAUSE
.MO <DB-STRUC>:/SHOVE
.R SETSRC
*C <DB-STRUC>
.IF (ERROR) .GOTO ST180

ST008:: !******** CHECK FOR INVALID FUTURE RAW FILES ********!

.ERROR
.RU <DB-STRUC>:DIP
*<DC-STRUC>:<SYS-ID>??.*/SINCE:NOW/DIRECT
.IF (NOERROR) .NOERROR
.RU <DB-STRUC>:DIP
*<DB-STRUC>:<SYS-ID>??.*/SINCE:NOW/DIRECT
.IF (ERROR) .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 <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.DB[<AMAR-PPN>]/SINCE:NOW/DIRECT
.IF (ERROR) .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 ********!

.R QUOLST
.NOERROR
.RU <DB-STRUC>:DIP
```

Figure J-1

```
*AMAR??.RPT/OKNONE,AMARUP.IF?/OKNONE/DELETE

STO2O:: !******** COPY RAW FILES TO PRODUCTION PACK ********!

.ERROR %
.OPERATOR $
.RU <DB-STRUC>:DIP
*<DB-STRUC>:/PRESERVE:ALL=<DC-STRUC>:<SYS-ID>??.*/BEFORE:TODAY:O:1O/OKNONE'NOT'<DC-STRUC>:<SYS-ID>DC.EXE
.NOOPERATOR
.IF (ERROR) .GOTO STO4O

STO3O:: !******** DELETE RAW FILES FROM DATA COLLECTION PACK ********!

.ERROR
.RU <DB-STRUC>:DIP
*<DC-STRUC>:<SYS-ID>??.*/BEFORE:-<DC-RTEN>D/OKNONE'NOT'<DC-STRUC>:<SYS-ID>D?.*/DELETE
.IF (NOERROR) .GOTO STO4O

BDO3O:: !!!!!!!!! OPSER DATA COLLECTION PROGRAM MAY HAVE STOPPED !!!!!!!!!
!CHECK IF OPSER <SYS-ID>DC JOB IS RUNNING!

STO4O:: !******** CHECK IF AMAR DATABASE IS CORRUPTED ********!

.R QUOLST
.ERROR %
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.DB[<AMAR-PPN>]/DIRECT
.IF (ERROR) .GOTO STO5O
.RU AMRGEN
*E D
.IF (ERROR) .GOTO STO5O
.DEAS
.GOTO STO8O

STO5O:: !******** SUBSTITUTE BACKUP FOR CORRUPTED DATABASE ********!

.DEAS
.ERROR
.RU <DB-STRUC>:DIP
*<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]/DIRECT
*<DB-STRUC>:AMAR.TAP[<AMAR-PPN>]/OKNONE/DELETE
.IF (ERROR) .GOTO BDO7O
.NOERROR
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.DB[<AMAR-PPN>]/OKNONE/DELETE
.ERROR %
.OPERATOR $
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.DB[<AMAR-PPN>]=<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]
.NOOPERATOR
.IF (ERROR) .GOTO STO6O
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.DB[<AMAR-PPN>]/DIRECT
.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 ********!
```

Figure J-1 (continued)

```
.NOERROR
.DIS <BKUP-STRUC>:/REMOVE
.PLEASE MOUNT <BKUP-STRUC> ON A DIFFERENT DRIVE SINCE READ ERRORS ON CURRENT DRIVE^[
.MO <BKUP-STRUC>:
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.DB[<AMAR-PPN>]/OKNONE/DELETE
.ERROR %
.OPERATOR $
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.DB[<AMAR-PPN>]=<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]
.NOOPERATOR
.IF (ERROR) .GOTO BDO7O

STO7O:: !******** CHECK IF BACKUP DATABASE IS CORRUPTED ********!

.ERROR %
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.DB[<AMAR-PPN>]/DIRECT
.IF (ERROR) .GOTO BDO7O
.RU AMRGEN
*E D
.IF (ERROR) .GOTO BDO7O
.DEAS
.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
.RU <DB-STRUC>:DIP
*<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]/RENAME/OKNONE/PROTECTION:O57
*<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]/DELETE/OKNONE
.ERROR %
.OPERATOR $
.RU <DB-STRUC>:DIP
*<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]/PROTECTION:777=<DB-STRUC>:AMAR.DB[<AMAR-PPN>]
.NOOPERATOR
.IF (ERROR) .GOTO STO9O
.RU <DB-STRUC>:DIP
*<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]/DIRECT
.IF (ERROR) .GOTO BDO8O
.GOTO ST1OO

BDO8O:: !!!!!!!!! INSUFFICIENT DISK SPACE TO CREATE BACKUP AMAR.DB DATABASE !!!!!!!!!
!DELETE UNNECESSARY FILES ON <BKUP-STRUC>!
.GOTO ST18O

STO9O:: !******** IF PROBLEM CREATE BACKUP ON DIFFERENT DRIVE ********!

.NOERROR
.DIS <DB-STRUC>:/REMOVE
.PLEASE MOUNT <DB-STRUC> ON A DIFFERENT DRIVE SINCE READ ERRORS ON CURRENT DRIVE^[
.MO <DB-STRUC>:
.RU <DB-STRUC>:DIP
*<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]/RENAME/OKNONE/PROTECTION:O57
*<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]/DELETE/OKNONE
.IF (ERROR) .GOTO ST18O
```

Figure J-1 (continued)

```
.ERROR %
.OPERATOR $
.RU <DB-STRUC>:DIP
*<BKUP-STRUC>:AMAR.DBK[<AMAR-PPN>]/PROTECTION:777=<DB-STRUC>:AMAR.DB[<AMAR-PPN>]
.NOOPERATOR
.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 ********!

.SET WATCH ALL
.SET WATCH NO FILES ;FOR 7.01
.MO <DB-STRUC>:
.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 ********!

.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 ********!

.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
```

Figure J-1 (continued)

```
;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
*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 ********!

.SET WATCH NONE
```

Figure J-1 (continued)

```
.DEAS
.NOERROR
.PRINT AMAR??.RPT/DIS:RENAME
;.XEROX AMAR??.RPT/DIS:RENAME

.GOTO ST170
ST140:: !******** CHECK IF DAY TO CREATE BACKUP TAPE OF AMAR DATABASE ********!

.ERROR
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.TAP[<AMAR-PPN>]/SINCE:NOW/OKNONE/DELETE
*<DB-STRUC>:AMAR.TAP[<AMAR-PPN>]/RENAME/PROTECTION:057
.IF (ERROR) .GOTO ST150
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.TAP[<AMAR-PPN>]=<DB-STRUC>:AMAR.TAP[<AMAR-PPN>]/BEFORE:-<TAPE-PRD>D
.IF (ERROR) .GOTO ST170

ST150:: !******** CREATE A BACKUP TAPE OF AMAR DATABASE ********!

.ERROR [
.MO MTA:MTAA/WENABL/REE:SCRTCH/VID:"<TAPE-ID>"
.IF (ERROR) .GOTO BD151
.SET DENSITY MTAA 6250
.IF (ERROR) .GOTO BD151
.SET WATCH ALL
.SET WATCH NO FILES
.ERROR %
.OPERATOR $
.R SYS:BACKUP
*TAPE MTAA
*SSNAME AMAR
*SAVE  <DB-STRUC>:[<AMAR-PPN>]AMAR.DB,<SYS-ID>??.*
*PRINT TTY:
*EXIT
.NOOPERATOR
.IF (ERROR) .GOTO BD150
.SET WATCH NONE
.NOERROR
.UNLOAD MTAA
.DIS MTAA
.GOTO ST160

BD150:: !!!!!!!!! ERROR CREATING BACKUP TAPE !!!!!!!!!
!PROVIDE CLEAN ERROR FREE DRIVE AND NEW SCRATCH TAPES!
.SET WATCH NONE
.NOERROR
.UNLOAD MTAA
.DIS 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 ST170

ST160:: !******** RECORD DATE OF BACKUP TAPE CREATION ********!
```

Figure J-1 (continued)

```
.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.TAP[<AMAR-PPN>]/CDATE:TODAY=<DB-STRUC>:<SYS-ID>DR.RFD[<AMAR-PPN>]

ST17O:: !******** DELETE RAW FILES FROM PRODUCTION PACK ********!

.RU <DB-STRUC>:DIP
*<DB-STRUC>:AMAR.TAP[<AMAR-PPN>]/OKNONE/DELETE
*<DB-STRUC>:<SYS-ID>??.*[<AMAR-PPN>]/BEFORE:-<DB-RTEN>D/OKNONE'NOT'<DB-STRUC>:<SYS-ID>??.*[<AMAR-PPN>]/ASINCE:TODAY'NOT'<DB-STRUC>:<
SYS-ID>D?.*[<AMAR-PPN>]/DELETE


ST18O:: !******** RESUBMIT THE AMAR BATCH STREAM ********!

.CHKPNT ST18O
.DEAS
.NOERROR
.SET WATCH NONE
.DEL AMAR.LG
.SUB <DC-STRUC>:AMAR.CTL,<DC-STRUC>:AMAR.LOG/DISPOSE:DELETE/UNIQUE:1/RESTART:1/TIME:1:O:O/AFTER:1:O:O
.GOTO FIN


%TERR::
.GOTO BD18O
%CERR::
.GOTO BD18O
%ERR::
.GOTO BD18O

BD18O:: !!!!!!!!! TIME LIMIT EXHAUSTED OR MONITOR OR PROGRAM ERROR !!!!!!!!!
!!!!!!!!! STREAM IS RESUBMITTED IF THE DATABASE IS OK !!!!!!!!!
!!!!!!!!! CORRECT PROBLEM IF IT WILL RECUR !!!!!!!!!


.ERROR
.RU AMRGEN
*E D
.IF (NOERROR) .BACKTO ST18O

ST19O:: !******** STOP THE AMAR BATCH STREAM ********!

.DEAS
.SET WATCH NONE
.NOERROR
.COP <DB-STRUC>:AMAR.LG=<DC-STRUC>:AMAR.LOG
.DEL <DB-STRUC>:AMAR.TAP[<AMAR-PPN>]

FIN::
%FIN::
```

Figure J-1 (continued)

APPENDIX K

WORKLOAD AMAR BATCH STREAM - WCRPTB.CTL

Reference Figure K-1 for a listing of an unedited WCRPTB.CTL stream.

DOINC: This step runs WCINC which preprocesses raw "checkpoint" data to form incremental data. WCINC "steals" WC.RAW from the data collection program by renaming it yyqmwd.RA0, where yyqmwd is today's fiscal date. (The data collection program will automatically start a new WC.RAW the next time it takes samples). If possible, WCINC "primes the pump" from yyqmwd.RA1, where yyqmwd is yesterday's fiscal date. Then it reads today's .RA0 file and writes a pair of files, yyqmwd.IN0 and yyqmwd.IN1, where yyqmwd is yesterday's fiscal date. The file with extension .IN0 is a summary file with one record per five-minute sampling interval. The file with extension .IN1 is a detail file, with one record per active job per sampling interval. When all of yesterday's data has been processed, WCINC writes a new pump primer file which contains status information and the remainder of today's .RA0 file. WCINC deletes yesterday's .RA1 file and today's .RA0 file.

Note: This discussion has assumed that WCINC is run every day, as it should be. If WCINC is not run for a few days, appropriate files will still be created when it finally is run.

DOUPD: This step runs WCUPD which performs database management including input, update, and deletion. First WCUPD reads the appropriate pair of preprocessed 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, WCINC deletes the oldest preprocessed 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
DOMN0	Month	Weekday	????.DB0	WCMN0.ext
DOMN1	Month	Weekend	????.DB1	WCMN1.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 WCRPTB 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 concatenated file is printed /DISP:RENAME to avoid accidental overwriting. 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. It also copies WCRPTB.LOG to WCRPTB.LG, because WCRPTB.LOG will be deleted after printing, and it may occasionally be handy for troubleshooting.

```
.DAYTIME
.SET WATCH ALL
.SET WATCH NO FILES
.IF (ERROR)
.R SETSRC
*C <DB-STRUC>,*
.IF (ERROR) .GOTO DOSUB
.DELETE WC.IN?,WC.RAX,WC.RP?
.IF (ERROR)
.R DIP
*<DB-STRUC>:WC.RAX=WC.RAW
*<DB-STRUC>:WCDBS.COX=WCDBS.CON
.IF (ERROR)
DOINC:: .CHKPNT DOINC
.R SETSRC
*C <DB-STRUC>,*
.IF (ERROR) .GOTO DOSUB
.RU WCINC
.IF (NOERROR) .GOTO DOUPD
.PLEASE STREAM WCRPTB.CTL FAILED IN STEP DOINC^[
.PLEASE NOTIFY AMAR ADMINISTRATION^[
.GOTO DOSUB
DOUPD:: .CHKPNT DOUPD
.R SETSRC
*C <DB-STRUC>,*
.IF (ERROR) .GOTO DOSUB
.RU  WCUPD
.IF (NOERROR) .GOTO DORPT
.PLEASE STREAM WCRPTB.CTL FAILED IN STEP DOUPD^[
.PLEASE NOTIFY AMAR ADMINISTRATOR^[
.GOTO DOSUB
DORPT:: .CHKPNT DORPT
.R SETSRC
*C <DB-STRUC>,*
.IF (ERROR) .GOTO DOSUB
;HERE TO DO ANY REPORTS ON WEEKDAY DAY FILES
DODYO::
.RUN WCRPTB
*HOURLY REPORT BY PROGRAM AND JOB
*??????.DBO
*WCDYO.PR1
*
*O
*24
*1
*
*.9
*N
*JOB
*PRGRM
*LOGIN
*
*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 WCRPTB
*SHIFT REPORT BY PROGRAM
```

Figure K-1

```
*??????.DBO
*WCDYO.PR2
*
*O
*24
*8
*
*.5
*N
*PRGRM
*
*CPU%
*
*
*N
.IF (ERROR)
.RUN WCRPTB
*SHIFT REPORT BY BATCH VS. TIMESHARING
*??????.DBO
*WCDYO.PR3
*
*O
*24
*8
*
*
*N
*BATCH
*
*CPU%
*
*
*N
.IF (ERROR)
.RUN WCRPTB
*SHIFT REPORT BY PPN
*??????.DBO
*WCDYO.PR4
*
*O
*24
*8
*
*
*N
*PPN
*
*CPU%
*
*
*N
.IF (ERROR)
;INSERT EXTRA REPORTS ON WEEKDAY DAY FILES AFTER THIS LINE
;INSERT EXTRA REPORTS ON WEEKDAY DAY FILES BEFORE THIS LINE
;HERE TO DO ANY REPORTS ON WEEKDAY WEEK FILES
DOWKO::
.RUN WCRPTB
*WEEKLY REPORT BY PROGRAM (TYPICAL 8-HOUR SHIFTS)
*?????.DBO
*WCWKO.PR2
*
```

Figure K-1 (continued)

```
*0
*24
*8
*
*.1
*N
*PRGRM
*
*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 WCRPTB
*WEEKLY REPORT BY PPN (TYPICAL 8-HOUR SHIFTS)
*?????.DBO
*WCWKO.PR4
*
*0
*24
*8
*
*
*N
*PPN
*
*CPU%
*
*
*N
.IF (ERROR)
;INSERT EXTRA REPORTS ON WEEKDAY WEEK FILES AFTER THIS LINE
;INSERT EXTRA REPORTS ON WEEKDAY WEEK FILES BEFORE THIS LINE
;HERE TO DO ANY REPORTS ON WEEKEND DAY FILES
DODY1::
.RUN WCRPTB
*HOURLY REPORT BY PROGRAM AND JOB
*??????.DB1
*WCDY1.PR1
*
*0
*24
*1
*
*.9
*N
*JOB
*PRGRM
*LOGIN
*
*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 WCRPTB
*SHIFT REPORT BY PROGRAM
```

Figure K-1 (continued)

```
*??????.DB1
*WCDY1.PR2
*
*O
*24
*8
*
*.5
*N
*PRGRM
*
*CPU%
*
*
*N
.IF (ERROR)
.RUN WCRPTB
*SHIFT REPORT BY BATCH VS. TIMESHARING
*??????.DB1
*WCDY1.PR3
*
*O
*24
*8
*
*
*N
*BATCH
*
*CPU%
*
*
*N
.IF (ERROR)
.RUN WCRPTB
*SHIFT REPORT BY PPN
*??????.DB1
*WCDY1.PR4
*
*O
*24
*8
*
*
*N
*PPN
*
*CPU%
*
*
*N
.IF (ERROR)
;INSERT EXTRA REPORTS ON WEEKEND DAY FILES AFTER THIS LINE
;INSERT EXTRA REPORTS ON WEEKEND DAY FILES BEFORE THIS LINE
;HERE TO DO ANY REPORTS ON WEEKEND WEEK FILES
DOWK1::
.RUN WCRPTB
*WEEKLY REPORT BY PROGRAM (TYPICAL 8-HOUR SHIFTS)
*?????.DB1
*WCWK1.PR2
*
```

Figure K-1 (continued)

```
*0
*24
*8
*
*.1
*N
*PRGRM
*
*CPU%
*
*
*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 WCRPTB
*WEEKLY REPORT BY PPN (TYPICAL 8-HOUR SHIFTS)
*?????.DB1
*WCWK1.PR4
*
*0
*24
*8
*
*
*N
*PPN
*
*CPU%
*
*
*N
.IF (ERROR)
;INSERT EXTRA REPORTS ON WEEKEND WEEK FILES AFTER THIS LINE
;INSERT EXTRA REPORTS ON WEEKEND WEEK FILES BEFORE THIS LINE
;HERE TO DO ANY REPORTS ON WEEKDAY MONTH FILES
DOMNO::
.RUN WCRPTB
*MONTHLY REPORT BY PROGRAM (TYPICAL 8-HOUR SHIFTS)
*????.DBO
*WCMNO.PR2
*
*0
*24
*8
*
*.1
*N
*PRGRM
*
*CPU%
*
*
*N
;ONLY LIKELY ERROR IS NO READY WEEKDAY MONTH FILE...
;...WHICH MEANS NO WEEKEND MONTH FILES WILL BE READY
.IF (ERROR) .GOTO DOPRT
.RUN WCRPTB
*MONTHLY REPORT BY PPN (TYPICAL 8-HOUR SHIFTS)
*????.DBO
*WCMNO.PR4
```

Figure K-1 (continued)

```
*
*O
*24
*8
*
*
*N
*PPN
*
*CPU%
*
*
*N
.IF (ERROR)
;INSERT EXTRA REPORTS ON WEEKDAY MONTH FILES AFTER THIS LINE
;INSERT EXTRA REPORTS ON WEEKDAY MONTH FILES BEFORE THIS LINE
;HERE TO DO ANY REPORTS ON WEEKEND MONTH FILES
DOMN1::
.RUN WCRPTB
*MONTHLY REPORT BY PROGRAM (TYPICAL 8-HOUR SHIFTS)
*????.DB1
*WCMN1.PR2
*
*O
*24
*8
*
*.1
*N
*PRGRM
*
*CPU%
*
*
*N
.IF (ERROR)
.RUN WCRPTB
*MONTHLY REPORT BY PPN (TYPICAL 8-HOUR SHIFTS)
*????.DB1
*WCMN1.PR4
*
*O
*24
*8
*
*
*N
*PPN
*
*CPU%
*
*
*N
.IF (ERROR)
;INSERT EXTRA REPORTS ON WEEKEND MONTH FILES AFTER THIS LINE
;INSERT EXTRA REPORTS ON WEEKEND MONTH FILES BEFORE THIS LINE
DOPRT:: .CHKPNT DOPRT
.R DIP
*<DB-STRUC>:WORKLD.RPT=WCMN?.PR?/OKNONE,WCWK?.PR?/OKNONE,WCDY?.PR?
*<DB-STRUC>:WC???.RP?=WC???.PR?
*<DB-STRUC>:WC???.PR?/DELETE
```

Figure K-1 (continued)

```
.IF (ERROR)
.PRINT WORKLD.RPT/DISP:RENAME
.IF (ERROR)
DOSUB:: .CHKPNT DOSUB
%TERR::
%CERR::
%ERR::
%FIN::
.SUBMIT WCRPTB.CTL/TIME:0:15:0/RESTART:1/UNIQUE:0/AFTER:0:30:0,WCRPTB.LOG/DELETE
.IF (ERROR)
.COPY <DB-STRUC>:WCRPTB.LG=WCRPTB.LOG
```

Figure K-1 (continued)

Blank Page

APPENDIX L

INSTALLATION AND RESOURCE REQUIREMENTS

INSTALLATION:

Prior to installation, the data center must resolve the following issues:

A. A four character code must be selected to represent the 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. An account must be set up on the system which will be running AMAR. The project-programmer numbers can be anything the site wishes. Privileges on this account must be as follows:

SPY/PEK on all of core
SPY/PEK on monitor

With these privileges the word in ACCT.SYS becomes 000003000000. Also a 200 page jobsize limit is needed.

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. Attached is a table that can be used to estimate the amount of storage necessary.

D. AMAR has two data collection programs--xxxxDC (where xxxx is the four character code for the system) and WHOWC. These programs are run as subjobs and collect data continuously. Therefore, once you have been notified that installation is complete, entries for these programs should be inserted in the OPR.ATO file as follows:

```
:SLOG 1,2
:DEF AMAR=
AMAR-RUN structure:xxxxDC[ppn]

:SLOG ppn
:DEF WC=
WC-RUN structure:WHOWC[ppn]
```

The site may decide on which pack this data is to be written. The AMAR subjob will allow hard-coding of the pack name. The WC subjob will write to the first pack in the search list. It should be kept in mind that these jobs run continuously and therefore the pack chosen should be permanently mounted.

E. For reporting purposes, each site must determine what portion of each day will be considered prime time. The prime time of each site is recorded into the database at the time it is created. The prime time interval must fall on whole hour boundaries. The default is 0800-1700. Up to 4 daily prime time periods may be selected.

F. The report generation function of System AMAR is very flexible, five different types of reports are offered at various summary levels. The different types are: Utilization, Trend Analysis, Typical Day, Disk, and Tape. The summary levels are: daily, weekly, and monthly. Each site can easily select and deselect the reports automatically generated by the batch stream, as well as generate any reports interactively in an ad-hoc fashion. The default reports for a new site are: daily utilization; weekly utilization, disk, and tape; monthly utilization, trend analysis, disk, and tape.

G. There are two batch streams that run nightly. These are AMAR.CTL and WCRPTB.CTL. 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:

```
.SUB AMAR.CTL,AMAR.LOG/DISPOSE:DELETE/UNIQUE:1
     /RESTART:1/TIME:1:0:0/AFTER:1:0:0

.SUB WCRPTB.CTL,WCRPTB.LOG/DISPOSE:DELETE/UNIQUE:0
     /RESTART:1/TIME:0:15:0/AFTER:0:31:0
```

Note: If a stream failed because of errors, simply resubmitting the stream may not work. The error should be looked up in AMAR-10 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 .5% 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 BLOCKS

	System AMAR	Workload AMAR
Program Software	2500	500
Daily Raw File	100	2000
Previous Raw Files (7 days retention)	600	2000
TOTAL (without databases)	3200	4500

	Database*			
Size	Retention	Blocks	Retention**	Blocks
mini	M-12 W-13 D-35 H-7	3000	M-2 W-2 D-7	3500
midi	M-12 W-13 D-35 H-7 COW-1 COM-1	4000	M-3 W-5 D-14	6500
maxi	M-12 W-13 D-35 H-7 COW-5 COM-3	6000	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	3,200
Workload Programs and files	4,500
System AMAR database	4,000
Workload database	6,500
Backup System AMAR database	4,000
TOTAL	22,200

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 will typically 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	- System AMAR jobstream
AMAR.TAP	- Control file used by AMAR.CTL for scheduling of tape backup
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
AMREPT.OVL	- Overlay for the Reporting Program - only if using the version of AMREPT.EXE that includes its own object time system
AMRGEN.EXE	- Database Generation 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:

WC.EXE	- Special preprocessor Program for the current raw data file
WC.IN0	- Summary preprocessed file output by WC.EXE
WC.IN1	- Detail preprocessed file output by WC.EXE
WC.RAW	- Raw data file created by WHOWC.EXE
WC.RAX	- Copy of yesterday's WC.RAW file
WCDBS.CON	- Control file for Workload database
WCDBS.COX	- Yesterday's WCDBS.CON (used for recovery)
WCDBS.LOK	- Access control file for database
WCDYn.RPz*	- Default daily reports produced by WCRPTB.CTL
WCFIX.EXE	- Program to set holidays.
WCINC.EXE	- Normal preprocessor Program for raw files
WCMNn.RPz*	- Default monthly reports produced by WCRPTB.CTL
WCRPT.EXE	- Reporting Program for preprocessed files
WCRPTB.EXE	- Reporting Program for database files
WCRPTB.CTL	- Workload jobstream
WCRPTC.EXE	- Reporting Program for large requests from database files
WCUPD.EXE	- Database Management Program
WCWKn.RPz*	- Default weekly reports produced by WCRPTB.CTL
WHOWC.EXE	- Data Collection Program
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
yyqmwd.IN0*	- Summary preprocessor file output by WCINC.EXE
yyqmwd.IN1*	- Detail file output by WCINC.EXE
yyqmwd.RA0*	- Raw data file renamed by WCINC.EXE
yyqmwd.RA1*	- Portion of raw file with today's data (since midnight) left over after WCINC.EXE processes yesterday's data

*

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, 1

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

xxxx = system code

Blank Page

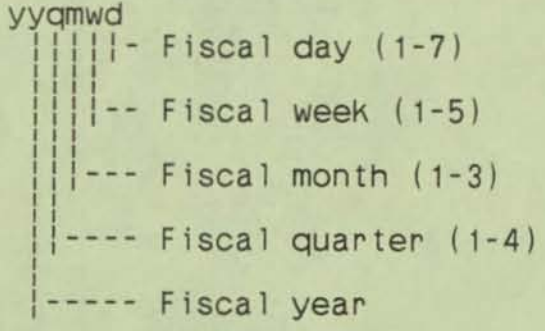
APPENDIX N

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.

Figure N-1 is an example of the FY82 Digital fiscal calendar. The Digital calendar normally starts at the end of June or beginning of July. This particular calendar begins on June 28, 1981. It contains the extra week mentioned above. This extra week occurs at the end of the fiscal year (June 1982).

Fiscal dates are used extensively by AMAR. They are always specified in the following format:



For example, by looking at Figure N-1, it can be determined that:

September 1, 1981 = 821323

September occurs in quarter 1 of FY82; it is the 3rd month;

September 1 occurs in week 2; it is the 3rd day.

It is recommended that you keep a similar copy of your fiscal calendar on hand for quick help in such date translations.

1982 FISCAL CALENDAR 1982

PLEASE NOTE: FY82 IS A 53 WEEK FISCAL YEAR. THE EXTRA WEEK WILL BE TAKEN IN JUNE, MAKING Q4 A 14 WEEK QUARTER.

FIRST QUARTER

MONTH	WEEK NO.	S 1	M 2	T 3	W 4	T 5	F 6	S 7
JULY 4 WEEKS	1	28	29	30	1	2	3	4
	2	5	6	7	8	9	10	11
	3	12	13	14	15	16	17	18
	4	19	20	21	22	23	24	25
AUG 4 WEEKS	5	26	27	28	29	30	31	1
	6	2	3	4	5	6	7	8
	7	9	10	11	12	13	14	15
	8	16	17	18	19	20	21	22
SEPT 5 WEEKS	9	23	24	25	26	27	28	29
	10	30	31	1	2	3	4	5
	11	6	7	8	9	10	11	12
	12	13	14	15	16	17	18	19
	13	20	21	22	23	24	25	26

SECOND QUARTER

MONTH	WEEK NO.	S	M	T	W	T	F	S
OCT 4 WEEKS	14	27	28	29	30	1	2	3
	15	4	5	6	7	8	9	10
	16	11	12	13	14	15	16	17
	17	18	19	20	21	22	23	24
NOV 4 WEEKS	18	25	26	27	28	29	30	31
	19	1	2	3	4	5	6	7
	20	8	9	10	11	12	13	14
	21	15	16	17	18	19	20	21
DEC 5 WEEKS	22	22	23	24	25	26	27	28
	23	29	30	1	2	3	4	5
	24	6	7	8	9	10	11	12
	25	13	14	15	16	17	18	19
	26	20	21	22	23	24	25	26

THIRD QUARTER

MONTH	WEEK NO.	S	M	T	W	T	F	S
JAN 4 WEEKS	27	27	28	29	30	31	1	2
	28	3	4	5	6	7	8	9
	29	10	11	12	13	14	15	16
	30	17	18	19	20	21	22	23
FEB 4 WEEKS	31	24	25	26	27	28	29	30
	32	31	1	2	3	4	5	6
	33	7	8	9	10	11	12	13
	34	14	15	16	17	18	19	20
MARCH 5 WEEKS	35	21	22	23	24	25	26	27
	36	28	1	2	3	4	5	6
	37	7	8	9	10	11	12	13
	38	14	15	16	17	18	19	20
	39	21	22	23	24	25	26	27

FOURTH QUARTER

MONTH	WEEK NO.	S	M	T	W	T	F	S
APRIL 4 WEEKS	40	28	29	30	31	1	2	3
	41	4	5	6	7	8	9	10
	42	11	12	13	14	15	16	17
	43	18	19	20	21	22	23	24
MAY 4 WEEKS	44	25	26	27	28	29	30	1
	45	2	3	4	5	6	7	8
	46	9	10	11	12	13	14	15
	47	16	17	18	19	20	21	22
JUNE 6 WEEKS	48	23	24	25	26	27	28	29
	49	30	31	1	2	3	4	5
	50	6	7	8	9	10	11	12
	51	13	14	15	16	17	18	19
	52	20	21	22	23	24	25	26
	53	27	28	29	30	1	2	3

Please Note:
All listed holidays are standard throughout the U.S. Assignable holidays and local location holidays are not shown.

digital

Figure N-1

digital

Software
Services

AMAR-10
INSTALLATION
GUIDE
VERSION 4.1

AMAR-10
INSTALLATION
GUIDE
VERSION 4.1

Digital Equipment Corporation

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

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

AMAR-10 INSTALLATION GUIDE

RELEASE 4.1

TABLE OF CONTENTS

AMAR-10 INSTALLATION GUIDE

Before you begin installation, it is suggested that you familiarize yourself with AMAR by reviewing as much of the AMAR-10 Reference Manual as possible. Minimally, you should read the "INTRODUCTION", the Sections called "OVERVIEW OF SYSTEM AMAR" and "OVERVIEW OF WORKLOAD AMAR", and the Appendix called "THE FISCAL CALENDAR".

1.0 INSTALLATION

You should decide on the following items prior to beginning installation. The items in this list will be referred to periodically throughout the installation process, therefore it is critical that they follow specifications given below and remain constant throughout the procedure. Special instructions may be given in parentheses and should always be adhered to.

1. sys-id

   A 4 character mnemonic used to identify your AMAR database. (A 'sys-id' must be exactly 4 alphanumeric characters. A completely numeric 'sys-id' is not recommended.)

2. dc-struc

   The permanently mounted disk structure onto which the data collection programs will write their raw data files and on which the data collection programs and streams will reside. Normally this structure is the same as the one on which the databases reside. (The 'dc-struc' includes only the pack name and should not be followed by ":".) To determine final disk quotas, see Section 1.9, Set Disk Quotas.

3. db-struc

   The permanently mounted disk structure on which the databases, the print files, and data processing programs will reside. (The 'db-struc' includes only the pack name and should not be followed by ":".) To determine final disk quotas, see Section 1.9, Set Disk Quotas.

4. amar-ppn

   The Project Programmer Number (PPN) which will contain the programs, files and streams. This account must have the PEEK OR SPY ON ALL OF CORE privilege and a JOBCOR limit of 200 pages. (An 'amar-ppn' includes both the project and programmer numbers separated by a ",". Do

not include "[ ]".)

5. bkup-struc

   The disk structure on which the backup database will reside. Normally this structure is the same as the one on which the databases reside. (The 'bkup-struc' includes only the pack name and should not be followed by ":".) To determine final disk quotas, see Section 1.9, Set Disk Quotas.

6. dc-rten

   The retention time in whole days of the raw data files on dc-struc, usually 7 days.

7. db-rten

   The retention time in whole days of the copied raw data files on db-struc, usually 7 days.

1.1 Read Tape

The installation tape is in BACKUP format at 1600BPI in non-interchange mode.

For the installation procedure, dc-struc and db-struc should each have a working quota of at least 4000 blocks. After the installation, the logged out and working disk quotas will need to be increased to accomodate the raw files and databases, according to the estimates in Section 1.9, Set Disk Quotas.

In order to start installation, copy the tape to disk as follows:

```
.MOUNT MTA MTAA:/REELID:?/WRITE:NO

.R BACKUP
/TAPE MTAA
/DENSITY 1600
/SSNAME AMAR
/RESTORE db-struc:[amar-ppn]=*.*[*,*,*]
/EXIT

.DISMOUNT MTAA:
```

The contents of the tape are:

1.	AMAR.1ST	AMAR.DB database seed file
2.	AMAR10.CTO	System AMAR batch stream
3.	AMAR10.RFD	System AMAR report parameter file for non-SMP systems
4.	SMP.RFD	System AMAR report parameter file for SMP systems
5.	AMAREX.EXE	System AMAR extraction program
6.	AMAREX.HLP	Its help file
7.	AMARIP.EXE	System AMAR input program
8.	AMARON.EXE	System AMAR online inquiry program
9.	AMARSD.EXE	System AMAR special input program
10.	AMARUP.EXE	System AMAR update program
11.	AMONLD.EXE	System AMAR onload program
12.	AMREPT.EXE	System AMAR report program
13.	AMREPT.OVL	Overlay file for AMREPT.EXE
14.	AMRGEN.EXE	System AMAR maintenance program
15.	701.EXE	System AMAR data collection program
16.	RFD.HLP	AMARDR.RFD help file
17.	WC.EXE	Workload AMAR special incrementalizer program
18.	WCRPTB.CTO	Workload AMAR batch stream
19.	WCRPT.EXE	Workload AMAR daily report program
20.	WCRPTB.EXE	Workload AMAR report program
21.	WCRPTC.EXE	Workload AMAR report program
22.	WCINC.EXE	Workload AMAR standard incrementalizer program
23.	WCUPD.EXE	Workload AMAR update program
24.	WCDATE.REL	Workload AMAR date file

25.	WCINIT.REL	Workload AMAR database initialization file
26.	WCGEN.REL	Workload AMAR database generation file
27.	WCFIX.EXE	Workload AMAR maintenance program
28.	WCPARM.MAC	Workload AMAR parameter file
29.	WHOWC.EXE	Workload AMAR data collection program.
30.	DIP.EXE	File copying program

1.2 Data Collection Programs

There are two data collection programs associated with AMAR. They are the System AMAR Data Collection Program and the Workload AMAR Data Collection Program. The System AMAR Data Collection Program must be specifically tailored for each system.

1.2.1 Tailoring Of The System AMAR Data Collection Program -

To tailor this program, you will have to:

- Specify system and disk information which will be used to encode information about placement of the daily raw data files in the data collection program.
- Enable measurements according to your system type and the types of items you want to measure.
- Save the data collection file so that the filename includes your system mnemonic and it resides on a permanently mounted disk structure.

The System AMAR data collection program is tailored specifically for a system by using the reentrant capability of the program as follows:

```
.GET 701
.REENTER
SYS>sys-id
STR>dc-struc
PRJ> Project number of amar-ppn (Do not follow with ",")
PGM> Programmer number of amar-ppn (Do not precede with ",")
BPA? (Y to measure Channel Busy and Priority Interrupts on  KL-10
```

Models or on non KL-10 models to enable this feature if the system is ever upgraded to a KL-10)
(N to not measure Channel Busy or Priority Interrupts)
LU? (Y to measure packs on a logical name basis) *See NOTE
(N to measure only on a physical name basis) *See NOTE
PU? (Y to measure packs on a physical name basis)
(N to measure only on a logical name basis)

.SAVE dc-struc:sys-idDC

NOTE

Logical name refers to identifying the data by pack name (e.g., pack WORK), while physical name refers to identifying the data by drive name (e.g., drive RPA0). The difference between measurements for logical pack name and physical unit name is that a measurement labelled with a logical name is the total for that pack across all drives it may have been mounted on, while a measurement labelled with a physical name is the total for all packs which had been mounted on that particular drive. If the response to the 'LU' query is affirmative, the 'PU' query will also appear, giving you the option of collecting data for both logical name and physical name. If you respond negatively to the 'LU' query, measurement based upon physical unit will be assumed without further query because at least one of the two options is required.

ERROR RECOVERY: If an error is made while generating the System AMAR data collection program, do NOT perform the save command and begin again with the 'GET 701' command. If you have already saved ????DC.EXE, delete it and then start again at the 'GET 701' command.

1.2.2 Copying The Workload AMAR Data Collection Program -

If dc-struc and db-struc are the same skip this step, otherwise:

```
.COPY dc-struc:WHOWC.EXE=db-struc:WHOWC.EXE
```

1.3 Create AMAR Subjobs

After the data collection programs have been tailored and/or copied, the next step is to get these programs running as subjobs.

1.3.1 Edit OPR.ATO -

Temporariliy LOGIN to an account from which you can edit the SYS:OPR.ATO file to add both the System AMAR and Workload AMAR subjob entries.

Note that if Channel Busy and Priority Interrupt Levels are to be measured (i.e., the response to the 'BPA?' query was affirmative), the System AMAR data collection program (sys-idDC) must be run as a [1,2] subjob in order to use the PERF. UUO.

Append the following lines to the list of subjob entries in SYS:OPR.ATO:

```
:SLOG amar-ppn (or) 1/2
:DEFINE AMAR=
AMAR-RUN dc-struc:sys-idDC[amar-ppn]
:SLOG amar-ppn
:DEFINE WC=
WC-RUN dc-struc:WHOWC
```

1.3.2 Start Temporary Subjobs -

The next time the system comes up the AMAR subjobs will start automatically as OPSER subjobs. However, they must be started up now to begin collecting data now.

Go to the operator terminal where OPSER should be running and at command level, start the subjobs interactively by typing them in as described above.

1.4 Building The Databases

There are two data bases associated with the AMAR system. One is the System AMAR database, and the other is the Workload AMAR database. Below are the instructions for generating each.

1.4.1 Build The System AMAR Database -

Using the dialog below, you will record the following information in the database, AMAR.DB.

1. System mnemonic to identify data
2. Item mnemonics to be stored
3. Weekday prime periods (e.g., type 0800-1700 for 8am to 5pm. Note that intervals must fall on whole hour boundaries.)
4. Holidays for forthcoming year (e.g., type 820906 for 6Sep82)
5. Retention times for composite data

Note that several prompts following the format '[USE <SET NAME> COMMAND TO INITIALIZE AMAR.DB]' will be issued from the AMRGEN program. The intent of these messages is to provide information and they do not affect the results of the session.

```
.RUN AMRGEN
*SET NAME PDP10 sys-id
*ADD ITEM ALL (This command will only enable measurement for
CPU-0.  If multiprocessor site give an 'ADD ITEM CPU-1' command
for second processor, 'ADD ITEM CPU-2' for third processor, etc.)
*SET WEEKDAY hhmm-hhmm (or) hhmm-hhmm,hhmm-hhmm (This sets the
prime-time period)
*SET HOLIDAY yymmdd (Use calendar dates.  Only 1 holiday may be
given in each 'SET HOLIDAY' command, so repeat as necessary)
*LIST HOLIDAY (This command lists the holidays that have been set
in a YYM,MDD format)
*SET RETENTION COWEEKLY n (For mini n = 0, for midi n  =  1,  for
maxi skip this command) *See NOTE which follows on the next page
*SET RETENTION COMONTHLY n (Same as above)
*EXIT DATABASE
```

NOTE

You must now select the approximate size of the database you want to generate. There is a choice of 3 sizes: mini (app. 3000 blocks), midi (app. 4000 blocks), or maxi (app. 6000 blocks). It will take a year to grow to this size. The actual size is dependent upon the retention times specified for each granularity of data, the number of jobs running on your system and the number of items you choose to measure. It has been found that varying the retention time for weekly and monthly composite data is the easiest way of controlling database size. (Composite data represents the same hour of the day averaged over a period of a week or month. In respect to the AMAR reports its useful lifetime is a single week or month. For this reason, we suggest adjusting composite data retention as a simple means of controlling data base size.) In this procedure, database size is first established by setting retentions for composite data through the 'SET RETENTION COWEEKLY' and 'SET RETENTION COMONTHLY' commands.

ERROR RECOVERY: If the error was made while intializing the database with the 'SET NAME' command: exit the database, delete the AMAR.DB file and start the session over. For information on how to modify other database parameters which may have been set incorrectly, see the Section in the AMAR-10 Reference Manual called, "Examining/Changing Database Parameters (AMRGEN)".

1.4.2 Build The Workload AMAR Database -

Building the workload database is a 3 step process. The first is to edit the parameter file. The second is to create the controlling file for the database. The third is to specify the upcoming holidays.

1.4.2.1 Edit WCPARM.MAC -

The workload database is a set of disk files. Again the size is dependent upon the retention times specified for each granularity of data plus the number of jobs running on the system. You have a choice of 3 sizes: mini (app. 3500 blocks), midi (app. 6500 blocks), or maxi (app. 20000 blocks). It will take a year to grow to this size.

You must record the following information into the workload database parameter file, WCPARM.MAC as described below:

1. System mnemonic to identify the data
2. Site description for the report banner
3. Database size
4. Next holiday ONLY if next weekday is that holiday

Edit WCPARM.MAC using your favorite editor. PLEASE NOTE, the single quotation marks are used only to identify text strings and should not be considered part of the string.

Substitute sys-id (4 character mnemonic) for the '!!!!' in the string 'NAME (!!!!)'.

Substitute a site description (90 character maximum) for the '####' in the string 'BANNER (####)'.

Substitute 'MINI' or 'MIDI' for the word 'MAXI' in the string 'DB.SIZE (MAXI)' if desired.

If the next weekday is not a holiday, please proceed to Section 1.4.2.2, otherwise include the date of the forthcoming holiday in fiscal date format following '831322' and preceded by a comma within the angle brackets in the string 'HDAYS <831112,831322>)'. The date must be converted to a fiscal date which is based on the Digital fiscal calendar. While AMAR will generate its own internal fiscal calendar each year, you should generate your own corresponding hard copy fiscal calendar to make translation easy. The procedure for doing this is shown in the AMAR-10 Reference Manual, in the Appendix called "THE FISCAL CALENDAR". Use the example calendar and the description of the calendar format given in the first paragraph of that Appendix.

1.4.2.2 Create WCDBS.CON Using The Dialog Below. -

```
.COMP/COMP WCPARM.MAC
.EX/REL WCDATE,WCPARM,WCINIT (*see NOTE)
.EX/REL WCDATE,WCPARM,WCGEN
```

NOTE

Following the execution of this command you will receive the following warning: '% WCOHLE-HOLIDAY LIST ENDS TOO EARLY'. This is merely a warning that the holiday list has not yet been updated and does not impact the installation process. The holiday list update will take place in Section 1.4.2.3 of the installation procedure.

1.4.2.3 Specify Holidays -

Enter the holidays for the forthcoming year in WCDBS.CON. This time use calendar (not fiscal) dates; for instance, type 820906 for 6Sep82.

```
.RUN WCFIX
*S H yymmdd (Use calendar dates. Only 1 holiday may be given in
each 'S H' command, so repeat as necessary)
*L H (This command lists the holidays which have been set)
*E D (This command exits the database)
```

ERROR RECOVERY: If an error was made while setting holidays ('S H' command), the holiday may be dropped by issuing a 'D H yymmdd' command. As with the 'S H' command, calendar dates must be used with the 'D H' command.

1.5 Edit The System AMAR Report Description File

The generation and content of the System AMAR reports is controlled by a file called sys-idDR.RFD. You must tailor a copy of either AMAR10.RFD or SMP.RFD for your site. Sites with single processor systems should use AMAR10.RFD and those running multiprocessors should use SMP.RFD.

.COPY db-struc:sys-idDR.RFD=db-struc:AMAR10.RFD (non SMP sites)

or

.COPY db-struc:sys-idDR.RFD=db-struc:SMP.RFD (SMP sites)

Edit sys-idDR.RFD with your favorite editor. If you use a line editor don't forget to strip the line numbers from the final version, otherwise it will cause the report program to fail. PLEASE NOTE, the single quotation marks are used only to identify text strings and should not be considered part of the string.

Substitute sys-id (4 character system mnemonic) for '????' in the line '.SD ???? _20 LETTERS SYSTEM ID_'.

Substitute a 20 character site description for the phrase '20 LETTERS SYSTEM ID' in the same line. Make sure you use exactly 20 characters. (Include blank spaces if necessary.)

Determine a low threshold for the number of physical pages of memory available to users on your system. Substitute this 7 digit number for both occurrences of '0000768' in the line beginning '.ID UMEM'. This number should be the total number of pages of physical memory minus the size of the monitor, any locked jobs, and a fudge factor representing about 75% of the smallest amount of memory which could be configured offline. The idea is to have reports flag periods when any memory is offline. The number 0000768 is likely to be about right for a 1024 page (half megaword) system.

1.6 Tailor The System AMAR Batch Stream

Copy a version of the master, AMAR10.CTO, to the data collection pack as AMAR.CTL as follows:

.COPY dc-struc:AMAR.CTL=db-struc:AMAR10.CTO

Edit the stream, AMAR.CTL, making global substitutions as follows. Once again, note that single quotation marks have been used to indentify text strings and should not be considered part of the string.

For all occurrences of '<DB-STRUC>' substitute the name of the primary (production) pack which contains the databases and associated files. Replace with db-struc.

For all occurrences of '<AMAR-PPN>' substitute the account containing the databases and associated files. Replace with amar-ppn.

For all occurrences of '<DC-STRUC>' substitute the name of the

structure which contains the data collection programs, raw files, and streams - it may be the same as the db-struc. Replace with dc-struc.

If the same structure is used for data collection and database storage (dc-struc is the same as db-struc) insert the line '.GOTO ST040' as the line preceding that starting with the label 'ST020::'.

For all occurrences of '<SYS-ID>' substitute the 4 character system mnemonic. Replace with sys-id.

For all occurrences of '<DC-RTEN>' substitute the retention period in days of daily raw files. Replace with dc-rten.

For all occurrences of '<BKUP-STRUC>' substitute the name of the structure used to backup the database and associated files - it may be the same as db-struc or dc-struc. Replace with bkup-struc.

Substitue for all occurrences of '<DB-RTEN>' the retention period in days of the number of copied daily raw files you want to keep on disk. Replace with db-rten.

To eliminate a report from the stream, you must comment out the 4 lines associated with it by inserting a semicolon at the beginning of each line. To find the 4 lines you must know the 2 character code for that report. The codes are DU, WU, MU, WA, MA, WC, MC, DD, WD, MD, DT, WT, and MT. See the AMAR-10 Reference Manual for the report descriptions in order to decide if you want to eliminate any.

Then search for the line beginning with '*cd' where cd is the report code. Comment out that line and the next 3 lines. They will then read:

```
;*cd
;*AMAR
;*AUTO
;*AMARcd.RPT
```

As the default, it is assumed that a backup tape of the AMAR database will not be created in AMAR.CTL. It is strongly urged that you do a backup of all AMAR files as part of your normal daily/weekly backup procedures. If a backup tape option is desired, please continue. If you do not wish to generate backup tapes in AMAR.CTL, proceed to Section 1.7, Tailor the Workload AMAR Batch Stream.

To enable the tape backup step delete the line '.GOTO ST170' immediately before the line starting with the label 'ST140::'. The stream sequence will then process ST140, ST150, and ST160 - the tape backup steps.

Replace '<TAPE-ID>' with the text of a tape label for the /VID

switch in the tape mount command. (Include only the information to be contained within the switch. Do not include the quotations marks which are used as delimiters.)

Replace '<TAPE-PRD>' with the number of days between tape backups of the System AMAR database and raw files. This retention should not be greater than that for the raw data files.

1.7 Tailor The Workload AMAR Batch Stream

Copy a version of the master, WCRPTB.CTO, to the data collection pack as WCRPTB.CTL as follows:

```
.COPY dc-struc:WCRPTB.CTL=db-struc:WCRPTB.CTO
```

Edit the stream, WCRPTB.CTL, making global substitutions as follows. Once again, please note that single quotation marks are used to identify text strings and should not be considered part of the string.

For all occurrences of '<DB-STRUC>' substitute the name of the structure containing the databases and associated files. Replace with db-struc.

1.8 Submit The Batch Streams

```
.SUBMIT dc-struc:AMAR.CTL/AFTER:TODAY+1:00
.SUBMIT dc-struc:WCRPTB.CTL/AFTER:TODAY
```

NOTE: Site may set additional switches if desired.

1.9 Set Disk Quotas

If you have not done so already, set the disk quotas for the AMAR account according to at least the amounts specified the chart below.

Functionally there are 3 possible structures on which you will need quota: dc-struc, db-struc, and bkup-struc. You may have elected to use only 1 or 2 of these structures. Account for the quota needed for all three on whichever structures you use.

The following estimates are considered minimal requirements. They do not take into consideration contingencies such as the stream(s) not being run for several days. (This situation allows raw files to accumulate, temporarily requiring additional disk

space.) So, if possible, be generous in your assignment of disk quotas.

Structure	Blocks for System AMAR	Blocks for Workload AMAR
dc-struc	700 (files)	4000 (files)
db-struc	2500 (pgms)	500 (pgms)
	3000 (mini DB)	3500 (mini DB)
	4000 (midi DB)	6500 (midi DB)
	6000 (maxi DB)	20000 (maxi DB)
bkup-struc	3000 (mini DB)	
	4000 (midi DB)	
	6000 (maxi DB)	