SHARE

SHARE SESSION REPORT

61	M503	VM Capacity Allocation Program	125
SHARE NO.	SESSION NO.	SESSION TITLE	ATTENDANCE

CME (Computer Management & Evaluation)	Sal Asad	RCH
PROJECT	SESSION CHAIRMAN	INST. CODE

RCA/CISS Bldg 204-2 Route 38, Cherry Hill, N.J. 08054 (609) 338-6543

SESSION CHAIRMAN'S COMPANY, ADDRESS, AND PHONE NUMBER

VM Capacity Allocation Program

George S. Peters

IBM

Information Products Division

Boulder, Colorado

Installation Code: IBB

Computer Management and Evaluation Project

Session M503

483

ABSTRACT

In any computer environment poor response time, system thrashing, and inequitable distribution of computer resources arise when the prime-shift workload exceeds system capacity. This paper describes CAP (Capacity Allocation Program)--a facility that apportions system resources in a Virtual Machine (VM) environment. User groups are granted an allocation based on their usage forecast. Those groups that use less than their allocation are rewarded with good service; those groups that use more than their allocation may receive degraded service.

4/R/rcg/2

OVERVIEW

When the prime-shift load of a computing system exceeds the system capacity, three consequences normally surface: a) the system begins to thrash and system throughput and DP investment are substantially reduced, b) response time is poor and user productivity drops dramatically, and c) user groups complain that the resource is not being distributed commensurate with the needs of the business (i.e., groups with critical schedules are not receiving the best service). This paper describes the Capacity Allocation Program (CAP) that represents a solution to these real problems.

CAP consists of a series of programs that permit an installation to represent the existing VM capacity as a 360-degree pie. The pie is divided among the various user groups based on their forecasts. Service is doled out depending on whether a group is using more or less than its allocation. Groups using less than their allocation receive good service; groups using more than their allocation receive degraded service.

Because CAP is active only during prime shift, those users over their allocation, as well as those groups desiring to stay under theirs, are motivated to move work offshift, thus reducing prime-shift demand and increasing offshift usage. Finally, groups identified by management as working on the most important projects can be given a large enough allocation to continue their work efficiently--even during periods when demand exceeds capacity.

System Thrashing

Software systems run efficiently when there is a sufficient supply of each of the required resources. When there is contention for a resource, a bottleneck develops, queues build, and the service time for each transaction increases. For example, when the demand for real storage is greater than the supply, transactions wait longer in the ready queue for storage to become available. When the system detects this shortage, it begins to "steal" pages from other non-active transactions which, when activated, add further to the overhead. When the system incurs this compounding overhead, the system is said to be thrashing, that is, the system loses its ability to do productive work because it begins serving itself rather than its users. To the extent that DP management permits this to happen will determine the extent to which the equipment investment is lost. CAP helps reduce system thrashing.

User Productivity Loss

Previous studies show that user productivity varies greatly with system load. Therefore, it becomes extremely important that DP management either ensure adequate capacity or control the workload so that the system performance remains good at all times. Otherwise, user productivity suffers, product schedules slip, and the associated cost increases.

When user response deteriorates from one to three seconds, a significant portion of a user's time is wasted in idleness. At $30 per hour and 200 users logged on, lost user productivity is substantial. Because CAP helps stabilize the system, user response time is more consistent, thus user productivity is enhanced.

Allocation Problem

VM requirements of user groups depend on a number of variables. Schedules, workload characteristics, and previous-day availability are just a few. When too much work arrives at the same time--regardless of the reason--the system begins to thrash; users get frustrated; and schedules slip. Some project managers feel thwarted and demand an explanation. When this occurs, the existing resources need to be allocated to the most worthy projects--at the expense of the remaining groups. CAP fulfills this need by apportioning existing capacity consistent with the needs of the business.

CAP permits the VM system processor to be viewed as a collection of smaller processors of varying capacity. Initially, a user group is given a small processor (small slice) but as requirements of the project increase, the user group is given a larger processor. A group working on a less important project is given the smaller processor. The resource granted to a particular group can be varied as the needs of the business change.

User Involvement

Good school administrators and teachers want parents to become involved in the school system because they know that involved parents are generally more satisfied with the system. Users of DP equipment are no different.

When they become involved in the problems of resource distribution and understand the needs of the total community, users are much less likely to complain about the service they receive. CAP encourages user involvement by forcing users to meet together, to understand the business needs and to distribute the computing resources equitably.

Prime Shift Demand

As more of the work force goes online, more people depend on the system during prime shift. As DP management acquires more hardware to meet prime shift demands, more of the offshift's wasted. Figure 1 graphically illustrates the offshift capacity that goes unused in many VM installations. This trend renders prime-shift time more expensive and offshift less expensive. To the extent that users can be motivated to use the offshift cycles (preferably by moving the work, not the people, offshift) less offshift is wasted and prime-shift demand is lessened. CAP provides the motivation necessary for people to change their working habits--to reduce their prime shift requirements and to increase their offshift usage. For example, a technical writer who is formatting a 120-page document on prime shift might want to break it into smaller pieces and format only the smaller sections during prime shift and schedule the formatting of the complete document to run during the offshift.

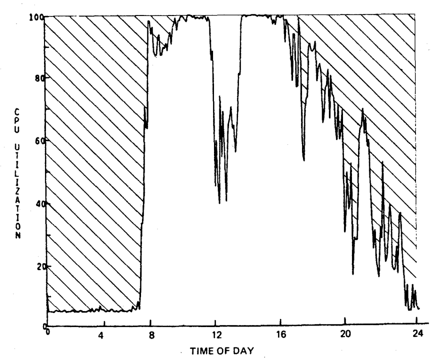

Figure 1. CPU Utilization Versus Time of Day

Forecasting Accuracy

Workload forecasting is an important part of capacity planning. Forecasters who take their job seriously and do a good job of prediction do a better job capacity-planning. Because groups that forecast poorly are visible, CAP can be used to degrade the service of those groups, resulting in a higher motivation to forecast well.

CPU Usage Sensitivity

In many installations users take the system resource for granted. Even when the application requires one-third of the CPU, the user feels as though the required resource should be available. Users sometimes forget that with 200 other users logged on, one-third of the CPU just isn't going to be available.

The real problem, of course, is that most users are not sensitive to the CPU resource. Users often do not know how much resource their application requires. With CAP, group users are reminded how much resource they are using. When a user on a given day uses 1/3 of the CPU and causes others in the group to exceed their allocation, peer pressure motivates the user to look carefully at the application to determine if it can be (a) run on the offshift using the batch facility (b) modified to take fewer resources (c) run less frequently or (d) replaced with a more efficient process.

CAP DESCRIPTION AND PROCEDURE

CAP helps solve these problems by helping to control system workload. Control is effected in four ways: a) users in the groups that have used more than their fair share will have their priority lowered. This reduces the overall workload by giving the resource to the users who are under allocation at the expense of those over their allocation; b) when a group gets close to or exceeds its allocation, the managers or coordinators of the group can meet and discuss ways of moving work (not people) offshift by carefully distinguishing real, prime-shift work from offshift work traditionally run on prime shift; c) when the need arises, groups can help reduce prime-shift load by staggering work hours. Some members arrive early and leave early; others arrive late and stay late; and finally, d) the ultimate but undesirable workload reducer is to force off the system the users in those groups that have used more than their fair share.

CAP User Group

A user group consists of a set of departments defined to the system. To define a group, a group number is specified, followed by a list of departments that compare the group. Figure 2 illustrates the Group Definition File. Group 103 has an allocation of three CPU minutes per day, and consists of users in departments 61G, 61H, and 619.

```
GRP DES ALO DEPT
100 JAC 003 61C,69A,69B,69J,69K,69L,69N,69Q,69R,69S,69Z,691,69U,69V
102 RJJ 007 690,693,694,699
103 DB  003 61G,61H,619
105 JHH 002 61F,664,67V,67Y,674,61E,61F,61J,61K,61L,61M
106 MEG 000 61B,61R,61W,61Y,614,66L,66N,66P,66R,66V,66W,68F,68Q,68R
108 JWW 006 60D,60N,60S,60U,61G,61K,63A,65D,65E,666,68C,68P,68U,68W
109 JWS 003 61D,65U,66A,66B,66D,66E,66G,68V
110 AH  005 61E,631
111 ABH 001 61P,64A,64C,64K,61J
112 RZW 105 619,635,638,639,659
113 WIB 003 63H,633,634,66H,63F
114 DTF 000 64J,63C,63D,63Z,636,68B,61V,611,61Q,63E
115 FC  000 641,615,618
116 DEB 017 60H,60K,60T,60Z,611,64M,69C,69D,69H,61Z,60E,60J,69E,61M
119 RAH 007 617,689,681
120 WBP 004 60M,60R,60Y,61V,68S,63J,63K,63L,63Q,63R,655,656,657,658
123 DA  004 677,63N,651,653,688,645
124 TMH 065 65Y,61N
126 DRM 066 61A,64Y,647,65F
127 CAQ 000 683,684,685,686,61A,61T,61R
128 RFK 001 64P
129 AN  004 60B,64H,64V,65T,65W,650,66K,67M,67Z,60C,63B,63M,63Y,64R
130 RSC 000 60Q,69P
199 MSC 000 61T,61U,663,646,618,64B,616,60W,60G,60V,63W,63T,610,64G
200 MFG 005 905,781,907,144,175,176,161,147,19B,19K,111,744,756,785
201 MFG 004 710,717,711,71A,71B,71D,71H,751,700,706,741,745,145,790
250 ME1 007 400,401,403,407,413,443,445,446,416,406,454,453,450,448
300 ISD 012 919,911,936,937,938,951,953,954,955,950,956,931
301 ISD 012 958,959,951,957
350 ISO 061 931,934,935,939,941,943,944,945,947,948,940,941
400 SS1 010 816,811,831,831,843,847,841,863,865,913,800,893,891,813
450 FIN 002 965,966,967,968,969,961,976,978,979,984,986,987,999,910
500 PER 007 503,504,505,507,509,513,514,518,511,515,519,914
550 PT1 020 78A,78C,78D,78E,78F,78K,78L,78M,78N,78P,78R,78T,78V,79B
600 BSS 004 110
650 CSD 002 991,100,101,103,104,105,106,107,108,109,101,168
990 OTH 002 946,949,71C,196,045,785,535,516
999 BAD 000 XXX
TOT TOT 000
```

Figure 2. Group Definition File

Capacity Pie

The capacity pie is the number of CPU minutes available in a prime shift day. For example, if one expects 95-percent CPU availability at 90-percent busy for a 9.5-hour, prime shift, then the capacity pie equals .95 X .90 X 9.5 X 60 or 487 CPU minutes. (Because the greatest demand is for prime shift, only prime shift is included.)

Pie Slice Allocation

With each user group present at an allocation meeting, the total prime shift capacity is divided. The allocation algorithm is a function of the group's forecast and the previous month's usage. For example,

$$\text{Slice} = 487 \times (F(i)/F(t)) \times R(i)$$

where F(i) is the group's forecast
F(t) is the total of all forecasts
R(i) is an adjustment factor based on the group's previous month's usage, allocation, and forecast

When all slices have been determined, they are entered into the third column of the Group Definition File (Figure 2), and must add to a value equal to the size of the pie.

Usage Tracking

During third shift when an excess of system capacity is available, CAP calculates usage for the previous day. Accounting records are scanned and CPU usage and terminal-connect time are accumulated by department and by group. The value is recorded in the Group Usage File in the column PREV DAY PRIME USAGE (CPU minutes), shown in Figure 3. At this time the 10-day running average is also calculated (and placed in the third column).

NEW-10-DAY-USAGE = (OLD-10-DAY X 9 + PREVIOUS-DAY) / 10

The last column of Figure 3 shows the 10-day average connect time in hours. Using the ratio of usage/connect, one can calculate a CPU-group-intensive value where the larger the value, the more CPU intensive the group is. For example, in group 100 the usage is two CPU minutes, and the connect time is 14.1 hours giving a ratio of 2/14 = .14. This means that group 100 uses .14 CPU minutes for each hour of terminal-connect time. Contrast this value with .84 for group 116 (22.8/27.3), a more CPU-intensive group. Also interesting is to compare the individual group values with the TOT value of 452/1826 = .24, which is the intensive value for <u>all</u> system users combined. This CPU-intensive-group value can be helpful in forecasting because it provides usage profile information about the group.

THE GROUP NMBR	PRVDAY PRIME USAGE	AVERAGE PRIME USAGE	GROUP ALLO-CATION	GROUP BASE PRIOR	PRVDAY PRIME CONNECT	AVERAGE PRIME CONNECT
100	0003.1	0002.0	0003	64	0013.4	0014.1
102	0006.4	0007.3	0007	66	0025.2	0043.0
103	0002.7	0004.6	0003	80	0014.8	0019.9
105	0001.8	0002.1	0002	66	0010.7	0013.7
106	0000.0	0000.0	0000	64	0000.0	0000.0
108	0000.8	0004.3	0006	64	0013.7	0012.6
109	0003.5	0002.6	0003	64	0017.5	0015.6
110	0007.3	0004.2	0005	64	0026.9	0016.8
111	0015.3	0020.7	0021	64	0056.8	0053.7
112	0134.6	0099.8	0105	64	0203.0	0183.9
113	0002.5	0003.1	0003	65	0010.8	0017.6
114	0000.4	0000.0	0000	64	0002.1	0001.1
115	0000.0	0000.0	0000	64	0000.0	0000.0
116	0076.8	0022.8	0017	75	0029.2	0027.3
119	0010.9	0010.1	0007	78	0031.7	0030.9
120	0002.3	0001.8	0004	64	0016.4	0011.9
123	0001.3	0002.3	0004	64	0002.6	0008.5
124	0032.5	0055.3	0065	64	0110.2	0139.4
126	0031.4	0039.8	0056	64	0085.6	0115.9
127	0000.0	0000.0	0000	64	0000.0	0000.1
128	0000.4	0000.1	0001	64	0003.9	0002.1
129	0000.3	0001.8	0004	64	0005.6	0007.3
130	0000.0	0000.0	0000	64	0000.0	0000.7
199	0000.2	0000.0	0000	64	0003.1	0000.6
200	0002.3	0005.3	0005	66	0015.6	0024.7
201	0005.6	0006.8	0004	85	0058.9	0074.3
250	0010.5	0007.5	0007	67	0054.1	0041.5
300	0019.1	0016.7	0012	76	0192.4	0160.9
301	0013.2	0012.6	0012	66	0150.1	0121.7
350	0056.7	0063.7	0061	66	0243.6	0235.4
400	0012.6	0011.8	0010	70	0126.9	0137.5
450	0003.1	0003.2	0002	82	0029.0	0027.2
500	0006.5	0007.4	0007	66	0056.8	0049.6
550	0014.4	0025.0	0020	72	0110.4	0125.9
600	0004.8	0002.1	0004	64	0032.5	0032.6
650	0003.6	0004.0	0002	94	0028.5	0030.0
990	0000.9	0001.9	0002	64	0012.3	0024.4
999	0000.8	0000.0	0000	00	0009.0	0003.6
TOT	0488.6	0452.7	0464	00	1803.3	1826.0

Figure 3. Group Prime Shift Usage and Connect Time File

Current Priority

Current Priority (CP) is the priority that a user is currently running with when logged on and active. The value can range from a high priority of 0 to the lowest priority of 98--just reversed from what we would think. As the user's CP **increases**, he gets **less** service than other users competing for CPU time. An increase of 10 in priority results in decreasing a user's service (relative to other users) by one-half. The effect of a given change in priority may vary widely, depending on the nature of work being done by the user and by other users on the system. Priority change primarily affects non-trivial transactions, (meaning, generally, work done in excess of 0.1 second of CPU time after the most recent terminal input or output). Priority change should not significantly affect trivial transactions (i.e., the first 0.1 second of CPU time after terminal input or output).

Normal Priority

The CP can take on the value of the Normal Priority (NP), that value given to the majority of users requiring normal service.

Base Priority

The CP can also take on the value of the Base Priority (BP), a priority calculated by CAP during the offshift. That is, during third shift, the 10-day running average (usage) is compared to the group allocation. If the usage is less than the allocation, the group is considered to be an under-allocation group and is given normal priority. If the group's usage is greater than the allocation, the group is an over-allocation group and a Base Priority is calculated, which depends on the percent that the group exceeds its allocation. In Figure 3, group 103 has an allocation of 3 minutes per day. The group's average usage is 4.6, or 1.6 minutes above their allocation, resulting in a BP of 80 for the next working day. The algorithm used to calculate BP is

$$BP = NP + ((((USAGE * 10) / ALLOC) - 10) * 3)$$

The CP will be set equal to the BP only during prime shift and only when system performance warrants. (This is discussed later under **Prime Shift Execution**.)

Hit List

Again during third shift, and after Base Priorities have been calculated for each group, CAP flags each group that has a Base Priority that exceeds the Normal Priority. From this set, CAP develops a HIT LIST of USERIDS and corresponding Base Priorities. The HIT LIST consists of all those users in groups who are over their allocation. These USERIDS will have their priorities adjusted on prime shift when performance warrants. Figure 4 shows an example of the HIT LIST.

AIELLO	66
AJONES	98
ANELSON	67
BJFOSTER	66
CHAMBERS	93
CLEMENTS	66
DALSETH	66
DAVENPA	66
DRAPER	93
DROBERTS	67
.	.
.	.
WISER	93
WPBROWN	98
WPETERSN	66
YODER	66
YOUNGWV	93
ZIEMKOWS	66
ZIMMER	98

Figure 4. Hit List File

Load Indicator (Expansion Factor)

The Expansion Factor (EF) is a dimensionless number--a measure of system load. In a non-technical sense, it is the time the keyboard is locked, divided by the CPU time it takes to process the transaction. In this paper, it is the output from the INDICATE LOAD command. A high expansion factor means that the system is thrashing and user response time has increased dramatically.

Prime Shift Execution

Throughout prime shift, CAP interrogates the system load via the Expansion Factor, and if necessary, sets the priorities of those on the HIT LIST, sleeps for an interval of 10 to 15 minutes; sets priorities again; sleeps again; etc. Such priorities set by CAP are always greater than or equal to NP.

When CAP awakens from a sleep interval, the current EF is compared with five system values E1, E2, X, Y,and Z, that represent five levels of system degradation--E1 being slightly degraded; Z being very degraded.

When the EF reaches E1, the Base Priorities are invoked for all users on the HIT LIST. This action frees up some resources for the under-allocation users at the expense of the over-allocation users. As the BPs are invoked, the user is notified one time:


```
YOUR PRIORITY HAS BEEN CHANGED BECAUSE YOUR GROUPS USAGE
EXCEEDS ITS FAIR SHARE. YOUR PRIORITY = CP AND WILL VARY
WITH SYSTEM LOAD. PLEASE DO NOT EXECUTE ANY LONG RUNNING
JOBS BECAUSE THEY MAY NOT COMPLETE OR BECAUSE YOU MAY BE
FORCIBLY LOGGED OFF DUE TO A CRITICAL SYSTEM LOAD.
FOR MORE INFO, TYPE CAPNEWS.
```

As the EF varies between E1 and E2, the current priority of users on the list will take on values between the Base Priority and the maximum (98) as illustrated in Figure 5. The objective is to have sufficient people on the HIT LIST so that the EF will float between E1 and E2.

```
          98 |--->                   ********************
             |                     * .    .        .   .
             |                   *   .    .        .   .
P            |                 *     .    .        .   .
R            |               *       .    .        .   .
I            |             *         .    .        .   .
O            |           *           .    .        .   .
R       'BP' |--->     *             .    .        .   .
I            |         *             .    .        .   .
T            |         *             .    .        .   .
Y       'NP' |**********             .    .        .   .
             |         .             .    .        .   .
             |         .             .    .        .   .
             |         .             .    .        .   .
             |         .             .    .        .   .
             |         .             .    .        .   .
             |         .             .    .        .   .
             |         .             .    .        .   .
           0 |_________|_____________|____|________|___|
             0         E1            E2   X        Y   Z

                    System Expansion Factor
```

Figure 5. VM Current Priority Versus System Load

When the EF reaches X, the system is considered minimally degraded and more action must be taken. Users on the HIT LIST are sent a plea message--an attempt to bring the system back in line by having all over-allocation users defer their work by voluntarily logging off.

When the EF reaches Y, the system is considered degraded and more action must be taken. At this point all users on the system are sent a plea--an attempt to prevent forced logoff of those on the HIT LIST. This technique is effective if not exercised too often. Frequency can be controlled by adjusting the value of Y.

When the EF reaches Z, the system is thrashing; drastic action must be taken. Those on the HIT LIST are warned they have ten minutes to log off or they will be forced off the system. This action will free up enough resources to bring the system back in line. The intent is to favor those who have managed to their allocation at the expense of those who have not.

Exception List

If some user's work is more important than others in the group, the USERID can be placed in an EXCEPT list. When this is done, the user's usage still accumulates toward the total group usage but the user is exempted from the HIT LIST, and his other priority will be set to the Normal Priority.

Offshift

Figure 6 shows usage and connect-time data for offshift hours. The primary use of this data is to track the change in offshift usage and offshift connect-time when user groups change their usage habits as a result of being constrained by limited capacity. Those groups that move work to the offshift can be favored with a somewhat larger allocation during prime shift. In this way people are motivated to carefully analyze their prime-shift work, ensuring that it needs to be run during prime shift.

THE GROUP NMBR	PRVDAY OFFSHT USAGE	AVERAGE OFFSHFT USAGE	GROUP ALLO- CATION	GROUP BASE PRIOR	PRVDAY OFFSHFT CONNECT	AVERAGE OFFSHFT CONNECT
100	0000.8	0000.2	0003	00	0001.9	0001.1
102	0000.0	0000.5	0007	00	0000.7	0003.2
103	0000.0	0000.0	0003	00	0000.0	0000.0
105	0000.0	0000.0	0002	00	0000.2	0000.2
108	0000.0	0001.5	0006	00	0000.0	0000.8
109	0000.2	0000.0	0003	00	0000.7	0000.2
110	0000.5	0000.8	0005	00	0000.7	0000.6
111	0001.8	0007.4	0021	00	0005.9	0006.0
112	0312.0	0155.4	0105	00	0036.5	0069.8
113	0001.0	0000.1	0003	00	0001.9	0000.9
116	0001.1	0018.6	0017	00	0004.7	0002.5
119	0001.7	0000.1	0007	00	0005.6	0001.1
120	0000.0	0000.0	0004	00	0000.0	0002.2
123	0000.0	0000.0	0004	00	0000.0	0000.0
124	0019.6	0062.1	0065	00	0026.5	0047.1
126	0016.6	0016.0	0056	00	0016.5	0015.4
129	0000.0	0000.1	0004	00	0000.3	0000.9
130	0000.0	0000.0	0000	00	0000.0	0000.0
199	0000.0	0000.0	0000	00	0000.0	0000.0
200	0000.3	0001.2	0005	00	0001.2	0003.4
201	0000.4	0000.5	0004	00	0002.6	0008.9
250	0000.4	0000.4	0007	00	0003.1	0002.7
300	0001.6	0001.3	0012	00	0010.0	0010.9
301	0001.4	0001.3	0012	00	0012.1	0021.4
350	0021.0	0059.3	0061	00	0080.9	0166.6
400	0001.0	0001.8	0010	00	0044.1	0086.4
450	0000.2	0000.0	0002	00	0001.3	0001.0
500	0002.6	0007.1	0007	00	0023.8	0018.7
550	0002.3	0006.1	0020	00	0023.0	0038.8
600	0001.9	0002.1	0004	00	0013.3	0011.5
650	0000.2	0000.2	0002	00	0000.6	0002.4
990	0000.0	0003.3	0002	00	0000.2	0038.9
999	0000.0	0000.0	0000	00	0000.1	0000.0
TOT	0388.6	0347.4	0464	00	0318.4	0563.6

Figure 6. Group OffShift Usage and Connect Time

CAP and Adequate Capacity

CAP is activated even during periods of adequate capacity because a) unique events may cause the capacity and workload lines to meet and problems arise. For example, a heavy load on a day when the paging subsystem is only partially operational, b) CAP has benefits such as user involvement and user sensitivity that are ongoing regardless of the state of the system, and c) CAP will not adjust priorities until the expansion factor reaches E1, so users on the HIT LIST are not affected so long as the supply is greater than the demand.

CONCLUSION

CAP works because it reduces prime-shift workload. It motivates users to carefully analyze their work and make sure that only prime-shift work is performed on prime shift. Figure 7 illustrates a typical group's usage-allocation plotted against time. Basically there are two pressure points, P1 and P2. At P1, users are motivated to review their habits (and habits of their peers), because they fear the consequences of exceeding their allocation. Users will invariably identify jobs that need not be run on prime shift. For example, a user may decide that because he does not have time to analyze the results of a simulation run until the next day, he can submit the job to be executed at 2 a.m. and have the output available at 8 a.m. This frees up capacity. Multiplied by 100 users, the regained resource becomes significant.

```
C |
P |        P2->* *
U |----------*----*---------*-------------- ALLOCATION
  |     P1->*       * * * *   * * * * USAGE
M |  * * *
I |
N |
U |
T |
E |
S |
  |_____|_____|_____|_____|_____|_____|____|

              TIME (Weeks)
```

Figure 7. Group Usage and Allocation Versus Time

Note that the next pressure point is at P2. Here, all the users--large and small--are getting the terminal message (discussed previously) that they have used more than their fair share. Because it is in poor taste to use more than a fair share, discussions in the halls and around coffee machines transpire. This communication fosters sensitivity to the CPU resources. Sensitivity results in ideas and actions to reduce usage below the allocation. The fact that groups work to stay under (or to get under) their allocation changes user habits and reduces workload, which in turn, results in a smoother running system.

SUMMARY

Before CAP, users felt that the CPU should always be available, regardless of the workload and current situation. After CAP, users began seeing the computer resource in the same way as managers see headcount. It is a limited resource and must be planned carefully. Estimating either too little or too much can have severe consequences.

ACKNOWLEDGMENT

The author thanks David Chess and other Yorktown professionals for providing many CAP concepts taken from the VM Prioritizer. Also, the design and implementation efforts of Dave Anderson, Bob Greenwalt, and Don Wagler are appreciated.

491

SHARE

SHARE SESSION REPORT

61	M504	SMF and RMF Differences In XA	350
SHARE NO.	SESSION NO.	SESSION TITLE	ATTENDANCE

Computer Management & Eval.	Wes Rushton	DCL
PROJECT	SESSION CHAIRMAN	INST. CODE

Datacrown Inc., 650 McNicoll Ave., Willowdale, Ont. 416-499-1012

SESSION CHAIRMAN'S COMPANY, ADDRESS, AND PHONE NUMBER

Dr. H.W. Barry Merrill, a recognized leader in the field of SMF and RMF Data Analysis, gave a talk on the differences in SMF and RMF records between MVS/370 systems and MVS/XA. The data presented will be of great interest to those computer performance analysts and capacity planners who make regular use of such data.

A copy of the text and foils used in Dr. Merrill's presentation is attached.