MSB - 91

To: Distribution

From: J.H. Saltzer and D.K. Gifford

Date: February 14, 1973

Subject: Draft of "Metering Manual"

The enclosed document is a first pass at pulling together, in one place, a description of all the meters of the system. It is apparent, upon looking it over, that much more in the way of explanations is needed, but it seems useful to distribute even this primitive draft, in order to obtain comments and suggestions.

Only tools known to be installed in the system are described here. Please report any that have been missed. Comments should be directed to Dave Gifford, Project MAC, room 518.

MULTICS METERING MANUAL

February 14, 1973

DRAFT

David K. Gifford
Project MAC, MIT

FOREWARD

The MULTICS METERING MANUAL is designed to be a central source of metering information. All installed system performance tools are listed in the manual.

Command	Short Description
disk_queue	report on I/O request queues
file_system_meters	print out select system wide file system meters
meter_fim	measures the performance of the fault intercept module
meter_signal	measures the performance of the signalling mechanism
page_multilevel_meters	report on the activity of the page multilevel algorithm
pre_page_meters	print out system wide statistics about the pre-paging mechanism
print_configuration_deck	print the configuration of Multics
system_performance_graph	gather together in a system of graphs metering information concerning system performance and operation.
total_time_meters	print out the CPU time percentage and average CPU time spent doing various tasks
traffic_control_queue	report on the state of the traffic control queue
traffic_control_meters	print out the values of various traffic control meters

Name: disk_queue, dq

This command prints out the waiting I/O request queues for a given disk storage unit. For each waiting request, the type of request (read or write), the physical device to which the request will be directed, the priority of the request, and the core address from or to which I/O will be done are printed.

Usage:

disk_queue devname

1) devname is either -d170 for information about the DSU170 disk or -d270 for information about the DSU270 disk.

Sample Output:

```
disk_queue (-d270 -d170 -d181)

Connects = 5164, 1217.

P  RW  D  CORE

1  R   4   13100
0  W   2    4060

Connects = 5410.

Queue empty.

disk_queue: Code 1 not found in error_table_.
```

Name: file_system_meters, fsm

This command is used to meter certain file system variables and functions.

Usage:

fsm option1 ... optionn

1) option — if no options are specified, fsm will print out the options available. Option may be chosen from the following list of options.

-all,-a — Print all system variables which fsm recogmizes.

-device,-dv — Print meters about device usage.

-ast — Print meters relating to Active Segment Table (ast) usage.

-page,-pg — Print meters relating to paging traffic.

-reset,-rs — This option resets for the invoking process only those meters specified.

-long,-lg — Specification of this option causes more meters to be printed than if -ast and -page had been specified. The meters only printed when -lg is specified are indicated by **.

Notes:

The following are brief descriptions of each of the variables printed out by fsm. All variables are metered on a system wide basis, no per-process meters are displayed.

The following meters reflect the activity of the AST. The two columns contain the total number of occurrences of the specified item and the average time between occurrences.

Item	Meaning
Deactivations	The total number of segment deactivations.
Seg Faults	The total number of segment faults.
Bound Faults	The total number of bounds faults.
**Setfaults (all)	The total number of set faults performed during segment deactivation and during the handling of bound faults.
**Setfaults (acc)	The total number of set faults performed because the access was changed on a segment.
**Updates	The total number of times branch information was updated from an AST entry.
**Steps	The total number of steps taken around the AST lists searching for an AST entry.
**Skips (eh)	The total number of times an entry was skipped because the entry hold switch was on.
**Skips (inf)	The total number of times an entry was skipped because there were active segments inferior to the directory entry.
**Skips (level)	The number of times an entry was skipped because it had too many pages in core.
**Skips (init)	The number of times an entry was skipped to give it a grace lap after all its pages were removed from core.
**Skips (lock)	The number of times an entry was skipped because its parent could not be locked.

**Skips (pc) The number of times an entry was skipped because page control could not clean all pages out of core.

The following items represent a table indexed by page table size and show the activity and use of the four AST lists.

AST sizes This shows the current page table sizes being used by the system.

Number This shows the number of entries of the specified size.

Need This shows how many entries of the specified size were needed.

Steps This shows the number of steps taken around the specified list.

Ave Steps This shows the average number of steps around the specified list to locate an entry.

Grace This gives the lap time for the specified list.

(The above are typed by specifying -ast)

Needc The count and average time between page faults and pre-pagings which required a block of core before being satisfied.

Ceiling The count and average time between times when too many write requests were queued at once.

Laps The count and average time between steps thru the core map looking for a page to allocate.

**Skip wired The count and average time between the times a page was skipped because it was wired down.

**Skip used The count and average time betwwen the times a page was skipped because it was used in the time since the last lap.

**Skip mod
The count and average time between times a page was skipped because it was modified.

**Skip os
The count and average time between times a page was skipped because it was out of service.

Average steps
the average number of steps required thru the core map to locate an allocatable page of core.

(The above are typed by specifying -page)

Left
The number of free records on the specified device.

Reads
ATB
The count and average time between (in milliseconds) read requests on the indicated device.

Writes
ATB
The count and average time between (in milliseconds) write requests on the indicated device.

ATB I/O
The average time between any I/O request for the specified device.

% Cpcty
The percentage of nominal channel capacity for the specified device. Nominal is defined as:

drum 480 transfers/sec
dsu270 19 transfers/sec
dsu170 9 transfers/sec

Ave Latency
The average latency for the specified device. Latency here includes transfer time.

N Errors
The total number of errors encountered on the specified device.

F Errors
The total number of fatal errors encountered on the specified device.

(The above is typed out by specifying -device)

Sample output:

```
fsm -all

Total metering time     0:27:44

                          #              ATB

Deactivations           1452          1.146 sec.
Seg Faults              4699           .354 sec.
Bound Faults             221          7.530 sec.
Setfaults (all)         6799        244.760 msec.
Setfaults (acc)           75         22.138 sec.
Updates                 2522        659.843 msec.
Steps                   3323        500.790 msec.
Skips (ehs)              244          6.820 sec.
Skips (inf)              498          3.342 sec.
Skips (level)            249          6.683 sec.
Skips (init)               0          0.000 sec.
Skips (ring)               6        277.354 sec.
Skips (lock)              22         75.642 sec.
Skips (pc)                 0          0.000 sec.

AST Sizes                 4          16          64         256
Number                  426         220          44           1
Need                   1649         494         161           0
Steps                  1745         432         154           0
Ave Steps               1.1          .9         1.0         0.0
Grace (sec)           406.3       847.5       475.5      1664.1

                     #            ATB

Needc            92235      18.042 msec.
Ceiling             17       1.631 min.
Laps              1048       1.588 sec.
Steps           352307       4.724 msec.
Skip wired       10940     152.114 msec.
Skip used       213788       7.784 msec.
Skip mod         30129      55.233 msec.
Skip os           5186     320.888 msec.

299 pages, 51 wired.
Average steps         3.820

                        DRUM       DSU270       DSU170

Left                       0         1948         2462
Reads                  78414         5203         1908
  ATB                 21.222      319.840      872.183
Writes                 45649         2113         1231
  ATB                 36.455      787.565     1351.848
ATB I/O               13.414      227.464      530.145
% Cpcty                   15           23            6
Ave Page Wait         15.414       60.724       87.861
Ave Chan Time              -            -       86.054
% Overlap                0.0          0.0         10.2
Ave Queue Size             -            -           .9
```

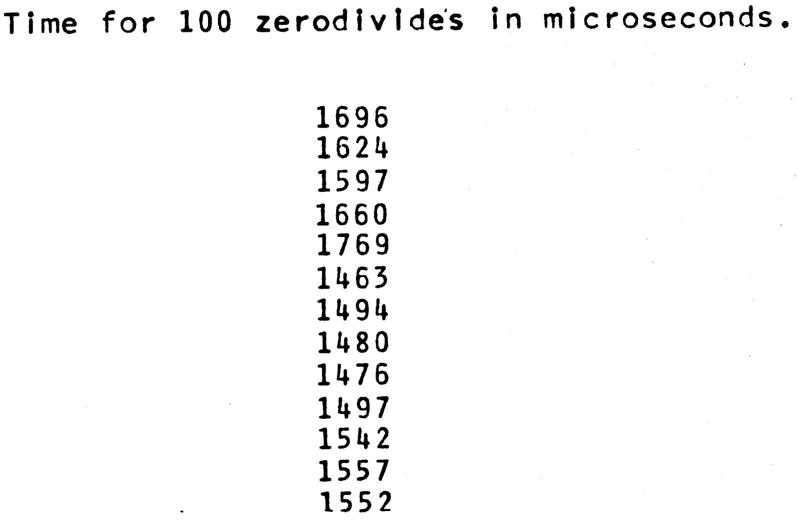
Name: meter_fim

This command measures the amount of time spent in the Fault Intercept Module (FIM) by causing 100 each of the following faults: zerodivide, mme1, and simfault_000001 and displaying the time in microseconds on the console.

Usage:

meter_fim

Sample output:



```
        Time for 100 zerodivides in microseconds.

                        1696
                        1624
                        1597
                        1660
                        1769
                        1463
                        1494
                        1480
                        1476
                        1497
                        1542
                        1557
                        1552
```

Page faults from PD — number of times the page faults were serviced from the paging device.

Pre-page reads — total number of pre-page reads

% faults from PD — percentage of the time a page fault could be serviced from the paging device

Ratio PD to other — ratio of times a page was found on the paging device to times a page was on another device

**(HISTOGRAM) — the histogram gives a distribution of the times a page remains on the paging device but is no longer being used

**Grace time — is the average time of the above histogram

**Lap time estimate — is an estimate of the lap time for the paging device used list

Sample output:

```
Total metering time    0:43:56

PD records                2031
Pages moved to PD         7069
Core blocks needed      176798
New pages                 4972
Page faults from PD     154501
Pre-page reads           42856
% faults from PD          89.9
Ratio PD to other        8.9:1

       RANGE             COUNT          AVE

 0 sec to 32 sec           421      12.5 sec.
32 sec to  2 min           376       1.3 min.
 2 min to  4 min          2768       3.5 min.
 4 min and more            916        .2 hr.

Grace time                 5.2 min.

Lap time estimate        7.546 min.
```

Name: pre_page_meters, ppmt

The ppmt command prints out system wide statistics about the pre-paging mechanism. It can be reset so that selected periods of time may be metered.

Usage:

pre_page_meters -options-

1) options may be chosen from the following list of options.

-all, -a This causes all the meters to be printed out If this is not specified, no meters will be printed.

-reset, -rs If this option is specified, any subsequent calls to ppmt will meter from the time of this call.

If no options are specified, a list of acceptable options is printed out.

Notes:

The following are brief descriptions of each of the variables printed by ppmt.

Item Meaning

Working-set factor The number the calculated working set is multiplied by to obtain the reported working set.

Working-set add Factor added to reported working set

Min-eligible Minimum number of processes that will be loaded

Max-eligible Maximum number of processes that will be loaded

% bad pre-paging Percent of pages the were pre-paged but never referenced

<u>Sample output:</u>

```
pre_page_meters
ppmt -options-
options: -reset (-rs), -all (-a)
ppmt -all

Total metering time    0:45:23

Working-set factor         .25
Working-set addend           0
Min-eligible                 2
Max-eligible                 6
% bad pre-paging         17.41
Drum faults/pre-paging    1.48
% drum priority moves    21.20
% misses                  1.20
Ave post size            37.09
Ave purge size            8.30
% purged                 22.37
Ave pre size             19.40
Ave pre-pagings          11.11
% pre-paged              57.25
Thrashing percentage      9.50
Ave post in core         26.89
Ave working-set size      6.94
Ave used in quantum      33.75
Pre-page time            42.29
Post-purge time          25.43
Calls                     4108
```

Name: print_configuration_deck, pcd

pcd will print out the current configuration deck of Multics. Consult the BOS operators manual for a description of the individual configuration cards.

Usage:

print_configuration_deck

Sample output:

```
pcd

cpu   a   6
cpu   b   7
mem   e   200   on
mem   c   200   on
mem   d   200   on
clok   a   0   25   est   5
gioc   a   2   0   7   11   13
drum   0   4096.   0   4   5   6   2
d270   0   29936.   a   27   12.   51217061320   71421101522
d170   0   57772.   a   37   13.   101112131415   70605040302   1   3   1
part   dump   0   0   0   0   57260.   512.   0   0
page   drum   0   2040.
part   page   0   10000   0   0   0   0   0   0
part   mult   0   0   0   29424.   0   57260.   0   0
part   salv   0   0   29424.   512.   0   0   0   0
sst   32.   426.   220.   44.   1.
prph   a   tap7   34   6   tap9   34   0   imp   40   0
prph   a   prta   40.   300.   prtb   34.   300.
tape   1   2
thrs   0.   2000.
schd   200000   20   20   100   2   6
int   227   230   231   232   37
ttyb   5
tty   a   60   3   1200.
tty   a   70   3   1200.
tty   a   100   32.   133.
tty   a   200   32.   133.
tty   a   300   24.   150.
tty   a   400   14.   110.
tcd   5   72.   150.   154.
intk   77   mult
```

Name: system_performance_graph,spg

This command is used to gather together in a system of graphs metering information concerning system performance and operation. Control arguments are provided to determine whether the output is to be directed to a file or to the controlling terminal. Metering information is periodically incrementally presented in an output line. (The initial line contains the cumulative values since system initialization.) Whenever there is a change in system configuratior or any of several parameters affecting system performance, an additional line noting the change is issued before the sample line. In this way, a system of graphs is developed where various metered quanities are plotted against time. Because the sampling is implemented by means of an event call channel, it is possible to use the console in a restricted way for other purposes while metering is in progress. All output is produced on the i/o stream "spg_output_".

Usage:

```
spg sample_time -option1  ...  -optionN-
```

1) sample_time   is a decimal integer giving the time, in minutes, desired between meter display lines.

2) optionI   is a character string which may be chosen from the following list of options:

```
    -halt,-ht       terminates plotting
    -output_file    directs output to a segment called "spg_output".
    -of
```

Description of the Ouptut Pattern

1) There is an initial line giving the date and time that metering sampling is begun.

2) A line is given describing configuration and scheduling parameter settings.

3) The current state of the meters since system initialization are on the next line where the sample time is replaced by the system initialization line.

4) Each subsequent meter display line gives the incremental status of the meters since the previous line. In addition, whenever the

configuration or scheduling parameter settings change, a notification line is interspersed.

Description of the Meter Display Line

Each line contains, in the left margin, the time that the sample was taken. Each sample is scheduled to be taken at an exact minute so that the amount the time given exceeds the minute represents a sample of the response time. (Strictly, the discrepancy is the response time of a trivial request only if the metering computation is less than the quantum and if the command argument sample_time is greater than one minute so that interactive scheduling occurs.)

The remainder of the meter display line consists of a sequence of superimpositions over a grid 100 units wide. The grid is created by vertical bars every 10 spaces with periods at the intervening midpoints between the bars. Over this grid, various metering quantities are superimposed in the following order. When the superimpostion is printed, only the last character superimposed in each postion is printed.

At the right of the grid is the current decimal value of the records of the high speed drum remaining free for storage allocation.

Time usage percentages

Symbol	Location	Meaning
blank	right of y to right margin	user processing not in ring 0 (note: the position of y is an estimate)
blank	right of v to left of y	user processing in ring 0
v		argument validation processing.
w		wall-crossing fault processing
s		segment fault processing
p		page fault processing

t		traffic controller processing
i	right of x	drum interrupt processing
blank	right of *'s to i's	multiprogramming idle
*		non-multiprogramming idle
blank	left margin to left of *'s	zero idle

Other values

The current average is determined from samples taken at one second intervals weighted backwards in time by increasing powers of 63/64. The effect is to average over roughly the last minute.

Symbol	Relative to	Meaning
q	left margin	current average of the ready list length
e	left margin	current average of the number of eligible processes.
r	left margin	current average of the response time in seconds, for trivial requests.
Q	left margin	average over a sample of quits/minute.
S	left margin	average over a sample schedulings/10 second.
d	right margin	average over a sample of DSU170 read and write traffic in pages/(.5 seconds)
D	right margin	average over a sample DSU270 read and write traffic in pages/(.5 seconds)
P	right margin	average over a sample of all read and write traffic in pages/(.5 second)

-	left margin	number of load units at the time of the sample
+	left margin	number of users at the time of the sample

Sample output:

```
spg .5
up= 1653.6, sys_hours= .4, cpu_hours= .7   11/09/72  1714.7 est Thu
           cpu= 2, pages= 299, min_e= 2, max_e= 6, wsa= 0, wsf= .25, tefirst= 2, telast= 2, timax= 8.
1714.81 |Q   e  *qS****************+iixiiiiitttttppppppppppss.    |    .P   |    .    |    .    |    . Dd|  o
1714.88 Qiiiieixiqiiiiiiiiiitttttttttttsttppp+pppppppPppppppppsy    |    .    |    .    |    .    |    .    D |  o
```

Name: total_time_meters, ttm

This command prints out the CPU time percentage and average CPU time spent doing various tasks.

Usage:

```
total_time_meters -option-

1) option          may be chosen from the following list of options.

    -reset, -rs    When this option is specified, after all ouput
                   is finished, the command will reset internal
                   static variables so that subsequent calls to
                   the command will effectively meter from the time
                   this option was last specified.

    -all, -a       When this option is specified, all statistics will
                   be printed out.
```

Notes

The following are brief descriptions of each of the variables output by ttm.

Item — Meaning

Page Faults — percentage of the total system CPU time spent in the page fault handler, average time required to process a page fault in microseconds

Drum interrupts — percentage of the total system CPU time spent in the drum interrupt handler, and the average time spent processing a drum interrupt in microseconds.

Getwork — percentage of the total system CPU time spent in the traffic controller looking for a process to run, and the average time required to select a process in microseconds.

Seg Faults — percentage of the total system CPU time spent processing segment faults, and the average time spent processing a segment fault in miroseconds.

Bound Faults — percentage of the total system CPU time spent processing bound faults, and the average time to process a bound fault in microseconds.

Interrupts — percentage of the total system CPU time spent processing non-drum interrupts, and the average time spent processing a non-drum interrupt in microseconds.

Gate Faults — percentage of the total system CPU time spent processing gate faults, and the average time to process a gate fault in microseconds.

MP idle — percentage of the time the system was idle because all of the loaded processes were unrunnable.

Loading idle — percentage of the time the system was idle due to loading

NMP idle — percentage of the time the system was idle for other reasons (ie it really was idle)

Zero idle — percentage of the time the system was idle because it was zeroing a new page for a process

Other — Fudge factor to make it all add up to 100%, contains user processing time

Sample output:

```
Total metering time    0:16:16

                         %        AVE

Page Faults            7.88     5274.587
Drum interrupts        3.43     2905.237
Getwork                3.37     1215.221
Seg Faults             2.13    14115.111
Bound Faults            .35    46584.570
Interrupts             2.69     5008.592
Gate faults             .01     3670.000
MP Idle                 .28
Loading idle            .03
NMP Idle              27.36
Zero idle             10.43
Other                 42.04
```

Name: traffic_control_queue, tcq

This command prints out the state of the traffic control queue at the time of the call.

Usage:

```
traffic_control_queue
```

Notes

The following items are printed out for each user in the queues.

1) flags The flags are one bit indicators in the Active Process Table (APT) entry for the user. The following flags are printed:

W	Interprocess Communication (IPC) wakeup waiting
S	Stop pending
P	Process being preempted
N	Process is non-interruptable
L	Process is loaded
E	Process is eligible
T	Interaction switch
Q	Interprocess signal (IPS) wakeup pending (QUIT)
R	Process is running
I	Process had been pre-paged

2) tu is the total CPU time the process has used in seconds

3) dtu is the incremental CPU time the process has used (in seconds) since tcq was last called.

4) te is the processes value of te in milliseconds

5) ts is the processes value of ts in milliseconds

6) ti is the processes value of ti in milliseconds

7) tssc is the real time since the process's state changed (in seconds)

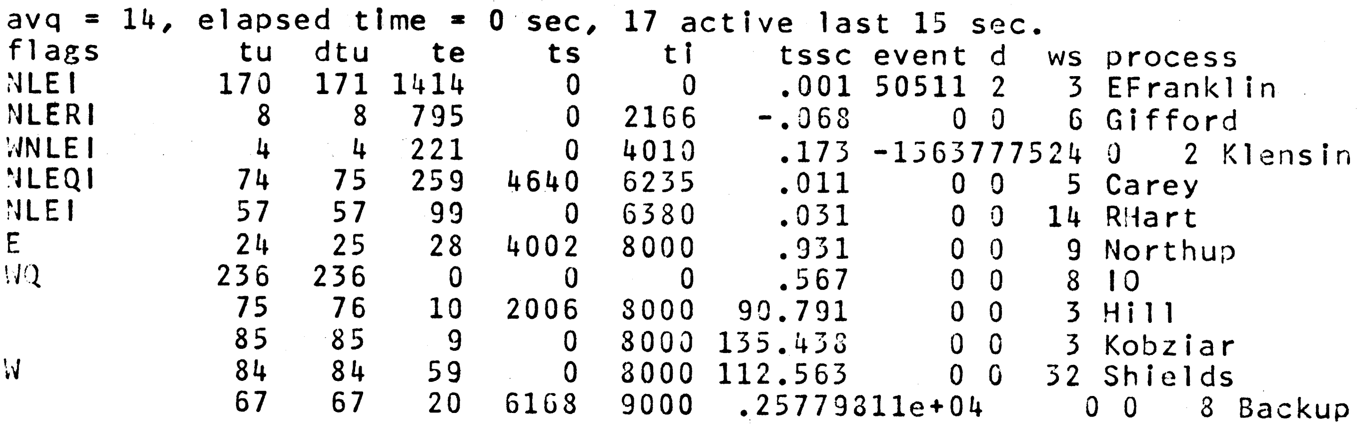
8) event is the event for which the process is waiting. If this value is 0, the process is not waiting.

9) d — if the process is waiting for a page, this is the device id of the device containing the page

10) ws — is the modified value of the working set estimate being used for the process.

11) process — is the name of the user who owns the process.

<u>Sample output</u>:



```
avq = 14, elapsed time = 0 sec, 17 active last 15 sec.
flags        tu  dtu   te    ts    ti     tssc event d  ws process
NLEI        170  171 1414     0     0     .001 50511 2   3 EFranklin
NLERI         8    8  795     0  2166    -.068     0 0   6 Gifford
WNLEI         4    4  221     0  4010     .173 -1363777524 0   2 Klensin
NLEQI        74   75  259  4640  6235     .011     0 0   5 Carey
NLEI         57   57   99     0  6380     .031     0 0  14 RHart
E            24   25   28  4002  8000     .931     0 0   9 Northup
WQ          236  236    0     0     0     .567     0 0   8 IO
             75   76   10  2006  8000   90.791     0 0   3 Hill
             85   85    9     0  8000  135.438     0 0   3 Kobziar
W            84   84   59     0  8000  112.563     0 0  32 Shields
             67   67   20  6168  9000   .2577981le+04    0 0   8 Backup
```

Name: traffic_control_meters, tcm

This command prints out the values of various traffic control meters.

Usage:

traffic_control_meters -options-

1) options may be chosen from the following list of options.

-gen print out general traffic control information and parameters

-counters,-ct Print out the number and frequency of certain paths thru the traffic controller.

-idle Print out the time spent in the various idle states.

-interactions
-int Print out a table of interaction frequency and CPU usage

-queue,-qu Print out certain resource usage as a function of depth in the eligible queue

-all,-a This option prints out all of the above

-reset,-rs When this option is specified, after all output is finished, the command will reset internal static variables so that subsequent calls to the command will effectively meter from the time this option was last specified.

Notes

The options may be given in any order.

```
tcm -all
```

Sample output:

```
Total metering time    0:54:36

Ave queue length      5.08
Ave eligible          4.59
Working-set factor     .25
Working-set addend       0
Te first (seconds)       2
Te last  (seconds)       2
Ti max (seconds)         8

   IDLE TYPE                TIME        %

Total idle                0:12:44   11.98
Multi-prog idle           0:00:41     .65
Loading idle              0:00:06     .10
Non-multi-prog idle       0:08:50    8.32
Zero idle                 0:03:05    2.91

 COUNTER        TOTAL        ATB         #/INT

Interactions     1270      2.580 sec
Loadings         5780       .567 sec      4.551
Blocks           4385       .747 sec
Wakeups          4640       .706 sec
Waits          158359     20.692 msec  124.692
Notifies       439214      7.461 msec
Schedulings      5395       .607 sec      4.248
Pre-empts      119302     27.466 msec   93.939

 Time  %Int  %Cum   Ave     %T    %CumT

 0.0   53    53     .296     9       9
  .5   24    76     .739    11      20
 1.0    6    83    1.235     5      25
 1.5    4    86    1.871     4      29
 2.0    1    87    2.307     1      30
 2.5    1    88    2.882     2      32
 3.0    1    89    3.408     1      33
 3.5    0    89    3.876     1      34
 4.0    0    89    4.481     1      35
 4.5    0    90    4.994     0      35
 5.0    0    90    5.488     1      36
 5.5    0    90    6.004     1      37
 6.0    0    90    6.579     1      37
 6.5    0    90    6.816     0      37
 7.0    9    99    7.343    38      76
 7.5    1   100    8.136    18     100

 DEPTH    %PF    TBPF    %GTW    TBS    %CPU

   1      27.4   33.5    21.9    24.8   25.9
   2      26.1   37.0    21.7    26.4   27.2
   3      21.1   34.4    23.0    18.6   20.4
   4      14.5   33.9    18.4    15.9   13.9
   5       7.9   38.2    10.7    16.5    8.5
   6       3.0   48.9     4.2    20.5    4.1
   7       0.0    0.0     0.0     0.0    0.0
```