

OmniOffender V2.2



OmniOffender
User's Guide
V2.2
August 26, 2008

OmniOffender is licensed software from Opsol Integrators Inc. and can be purchased as a stand-alone version or integrated with OmniDashboards. Contact Yash Kapadia at +1 408 446 9274 or email Yash@Opsol.com for more information.

Contents

1	Introduction	1
1.1	Only Two Commands!	1
1.2	HELP	2
1.3	Command Overview	3
1.4	Understanding Windows	5
2	Commands.....	7
2.1	Cpu [<cpu-report-option-list>] [(<cpu-list>)]	7
2.2	CPUS.....	11
2.3	CPUS (<cpu-list>)	12
2.4	DELAY <number>	12
2.5	Disc [<disc-report-option-list>] [(<cpu-list>)] [(<disc-list>)]	13
2.6	DISCS	14
2.7	DISCS (<disc-list>)	15
2.8	DISCS <discs-control>	15
2.9	ENV	16
2.10	EXIT.....	16
2.11	FC.....	17
2.12	File [<file-report-option>]	17
2.13	GO	18
2.14	HELP [ALL <command>]	19
2.15	LOG	19
2.16	LOG [TO] <log-file>	19
2.17	LOG STOP.....	19
2.18	MEASURE.....	19
2.19	PRocess [<process-report-option-list>] [(<cpu-list>)]	20
2.20	QUIT	22
2.21	REPort <report-list>	22
2.22	STATS	22
2.23	STatus <cpu> [, <pin>]	23
2.24	Window	24
2.25	Window <number>	24
2.26	Window <number> <window-size>	25
2.27	Window <number> <window-size> <report-list>	25
2.28	?.....	26
3	Installation.....	27
4	About OmniOffender	29
4.1	Version 2.2	29
4.2	Version 2.1	29
5	Index	31

1 Introduction

OmniOffender is an easy to use, fast, interactive performance monitor. OmniOffender displays the CPU utilization of the system by CPU and process. OmniOffender will automatically detect and display potential system bottlenecks.

System resources monitored and displayed are:

- CPU time utilization by interrupt handler, system processes, and user processes.
- CPU resource usage by memory, IPC, interrupts, or TNS usage.
- Process utilization of CPU time, messages per second, pages of memory, page faults, receive queue depth.
- Disc utilization, request rate, I/O rates, receive queue depth by volume.
- File activity.

The displays may be bar graphs, or tabular, for all CPUs of the system, or a subset. OmniOffender may be used either in conversational mode interactively, or in a multi screen block mode with automatic updates at specified intervals.

OmniOffender is a MEASURE application which is safe to use. It does not use privileged code to go behind the systems back to get measurement data. It does not MUTEX, access counters itself, or fire up samplers.

OmniOffender is much faster to use than MEASURE. It will automatically and very quickly configure the minimum necessary measurements.

1.1 *Only Two Commands!*

The only two commands necessary are the <ENTER> or <RETURN> key, and the "?" mark key.

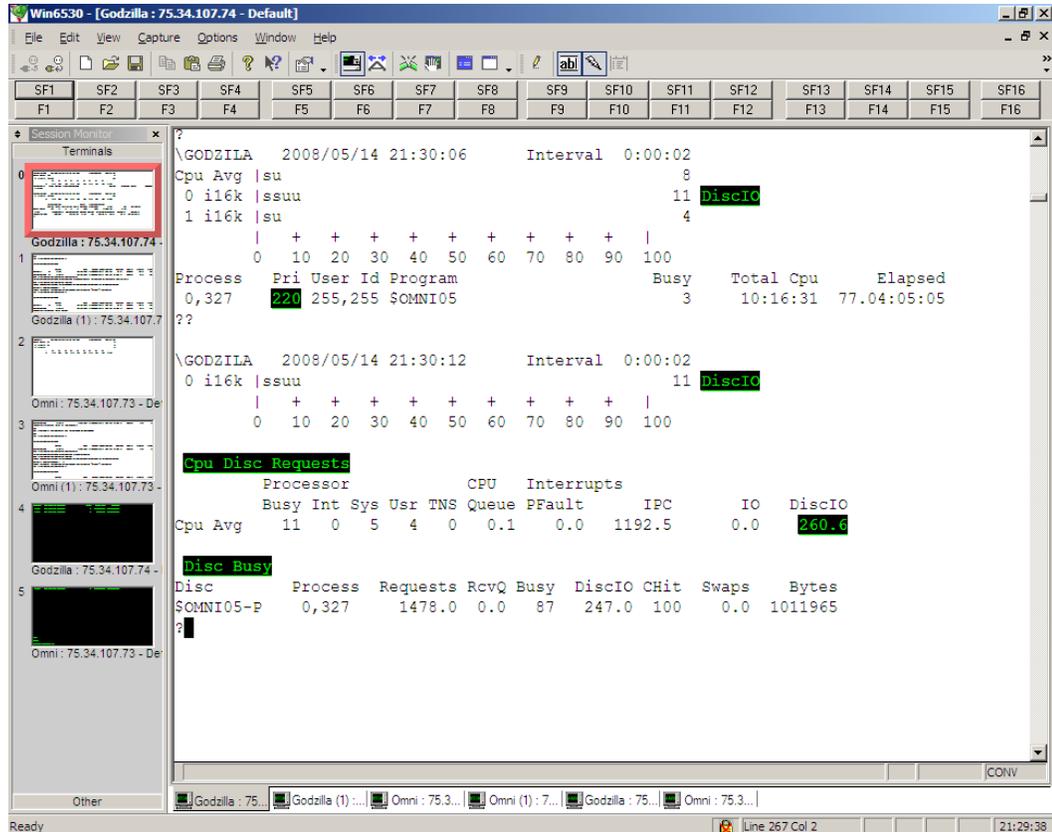
<Enter> or <RETURN> will sample all CPUs and display the default report. The default report will display, CPU busy, which processes are using the most CPU time, and any potential bottlenecks.

"?" will detail any bottleneck warnings. Bottleneck warnings are underlined on the terminal to the right of the CPU bar graph.

OmniOffender V2.2

Monitored resources are: ready queue depth, memory queue depth, page life in seconds, send busy time, I/O rates, etc.

Note the "???" in the display below – the second "?" is the ? command requesting detail for the DiscIO warning.



1.2 HELP

Type 'HELP' for a general introduction.

Type 'HELP ALL' for a complete help listing.

Type 'HELP <command>' command specific help.

1.3 Command Overview

This section will give an overview of commands; detail command descriptions can be found in the next section.

Command entry is case insensitive.

The use of upper/low case in command syntax is to indicate required characters.

Commands are:

Cpu, CPUS, DELAY, Disc, DISCS, ENV, EXIT, FC, File, GO, HELP, LOG, MEASURE, PProcess, QUIT, REPort, STATS, SStatus, Window, and '?'.

Typing <RETURN>, <ENTER>, or an empty line will sample all the CPUs and display the default report.

Commands that control the displays:

REPort, DELAY, GO, Window, CPUS, DISCS

To display CPU activity:

Cpu [BAR | BUSY | INTS | IPC | IPU | MEM | TMF | TNS]
[(*<cpu-list>*)]

To change the CPU selection criteria:

CPUS (*<cpu-list>*)

To display program activity:

PProcess [BUSY | FLTS | MEM | MSGS | RCVO | TIME] [(*<cpu-list>*)]
SStatus { cpu | cpu, pin }

To display disc activity:

Disc [BAR | BUSY] [(*<cpu-list>*)] [(*<disc-list>*)]

To change the disc selection criteria:

OmniOffender V2.2

```
DISCS { BUSY > <number> | RESET | (<disc-list> ) }
```

To display file activity:

```
File [ ACCESSED | BUSY | BYTES | DISC | ESCALATIONS | INFO |  
      LOCKS | MSGS | READS | RECORDS | TIMEOUTS | UPDATES |  
      WRITES | WRITEREADS  
      ]
```

Typing '?' after a warning will present the detailed report with the appropriate statistics.

The process busy time is displayed as a percent of a CPU's available processor time.

CPU time in the bar graph is displayed as:

```
'i' = interrupt handler process time.  
's' = process time of processes at priority 200 and above.  
'u' = user process time  
't' = TNS time.
```

DISC time in the bar graph is displayed as:

```
'w' = write qbusy time  
'r' = read qbusy time
```

More detailed information can be obtained with one of the display commands, or the default report can be changed.

Commands generally display either events per second, percent of a CPU per sample, or Queue counters. Queue counters (state time counters), display the number of items times how long each item was queued, each second.

The display commands can be restricted to search only certain CPUs or discs. Each command accepts a CPU list and/or a disc list at the end of the line, or a default CPU or disc list can be specified with the CPUS or DISCS command.

The syntax is for a <cpu-list> and <disc-list>:

```
<cpu-list> ::= { * | <cpu> [ , <cpu> ... ] | <cpu>/<cpu> }  
<disc-list> ::= { $* | $<vol> [ , $<vol> ... ] }
```

OmniOffender V2.2

For example:

```
Cpu BUSY, MEM (0/3)
```

will display CPU busy and memory statistics for CPUs 0, 1, 2 and 3;

```
Disc BUSY (0, 1) ($*)
```

will display all discs in CPUs 0 and 1 overriding the default CPU and disc lists.

1.4 Understanding Windows

A window is a 6530 conversational page. Eight windows of 24 lines are allowed. Conversational page mode is used when periodic sampling is enabled with the 'GO' command.

For VT100 only Window 1 is available.

Windows are useful to create a multiple reports which can be viewed by paging with the terminals next and previous page keys.

For example, the following specifies 6 windows. W 1, which matches the default, followed by the other CPU display options. "delay 60" specifies a 60 Second update interval when the GO command is used.

```
w 1, 24, cpu bar *, pr busy *  
w 2, 24, cpu busy  
w 3, 24, cpu int  
w 4, 24, cpu ipc  
w 5, 24, cpu mem  
w 6, 24, cpu tns  
delay 60
```

After entering "GO", OmniOffender will, every 60 Seconds (the interval was specified in the earlier "DELAY 60" command), display the specified 6 reports beginning with W 1, which matches the default report. Use the page up/down keys to view the remaining 5 reports. Each time the reports are displayed, the display will be homed to the first report.

OmniOffender V2.2



To display the current report configuration enter "w". Assuming the above, the following is displayed

```
?w
Window[ 1 ]    24 lines
  Cpu BAR      *
  PRocess BUSY *
Window[ 2 ]    24 lines
  Cpu BUSY     *
Window[ 3 ]    24 lines
  Cpu INT      *
Window[ 4 ]    24 lines
  Cpu IPC      *
Window[ 5 ]    24 lines
  Cpu MEM      *
Window[ 6 ]    24 lines
  Cpu TNS      *
?
```

To reset a report enter "w <window-number>". For example, the following will reset window 6 and display the report configuration:

```
?w 6; w
Window[ 1 ]    24 lines
  Cpu BAR      *
  PRocess BUSY *
Window[ 2 ]    24 lines
  Cpu BUSY     *
Window[ 3 ]    24 lines
  Cpu INT      *
Window[ 4 ]    24 lines
  Cpu IPC      *
Window[ 5 ]    24 lines
  Cpu MEM      *
?
```

The following will display a good summary of a system.

```
w 1, 24, cpu bar *, pr busy *
w 2, 24, pr busy
w 3, 24, disc busy
w 4, 24, file busy
delay 60
```

2 Commands

Command entry is case insensitive.

The use of upper/low case in command syntax is to indicate required characters.

For example, "CPU" in the CPU command may be entered as "C" or "CPU" using any mix of upper and lower case.

2.1 *Cpu* [**<cpu-report-option-list>**] [**<cpu-list>**]

Displays the selected CPU activity report/s for the specified CPUs.

```
<cpu-report-option> ::= { BAR | BUSY | INTs | IPC | IPU | MEM | TMF
                        | TNS
                        }
```

```
<cpu-list> ::= { * | <cpu> [ , <cpu> ... ] | <cpu>/<cpu> }
```

- **BAR** (default) a bar graph of CPU time as follows:
 - i – interrupt handler process time.
 - s – process time for priority 200+ processes.
 - u – process time below priority 200.
 - t – TNS time.
- **BUSY** displays summary of key queues and CPU times.
- **INTs** displays CPU interrupt rates.
- **IPC** displays CPU and message system statistics.
- **IPU** displays CPU IPU statistics (not supported on S-Series).
- **MEM** displays CPU and process memory statistics.
- **TNS** displays CPU time and emulation trap rate.
- **TMF** displays CPU TMF statistics.

OmniOffender V2.2



Sample outputs are shown below.

Cpu BAR report.

```

?cpu bar
\OMNI      2008/05/14 22:21:08      Interval  0:01:14
Cpu Avg |iissuuuuu      24
0 S74k |iisssuuuuuuu      38 Comp DiscIO
1 S74k |issuu      11 Comp
      | + + + + + + + + + +
      0 10 20 30 40 50 60 70 80 90 100
??

\OMNI      2008/05/14 22:21:15      Interval  0:01:14
0 S74k |iisssuuuuuuu      38 Comp DiscIO
      | + + + + + + + + + +
      0 10 20 30 40 50 60 70 80 90 100

CPI busy time or Compatability traps
Use Measure and/or XLTRACE to find the processes.
Processor      Emulation
Busy Int Sys Usr TNS      Traps
Cpu Avg      37  3 11 22  0      23054.1

Cpu Disc Requests
Processor      CPU      Interrupts
Busy Int Sys Usr TNS Queue PFault      IPC      IO      DiscIO
Cpu Avg      37  3 11 22  0      0.7  0.1  547.7  391.4  379.1

Disc Busy
Disc      Process  Requests RcvQ Busy  DiscIO CHit Swaps  Bytes
SOMNI01-P  0,291    928.1  0.0  82  257.0  88  0.4  2736079
SOMNI02-P  0,290    61.7  0.0  63  121.0  99  0.0  1360266
SOMNI04-P  1,270    66.1  0.0  11  20.3  95  0.4  402861
Next Page?
    
```

The underlined values for CPU 1 are potential performance issues. Use the '?' command as in the above example for additional information.

OmniOffender V2.2



Cpu BUSY, INTs, and IPC reports.

The screenshot shows a terminal window with three reports: ?cpu busy, ?cpu int, and ?cpu ipc. The ?cpu int report has a red box highlighting the 'Rate' and 'Fault' columns for processor 1 S74k.

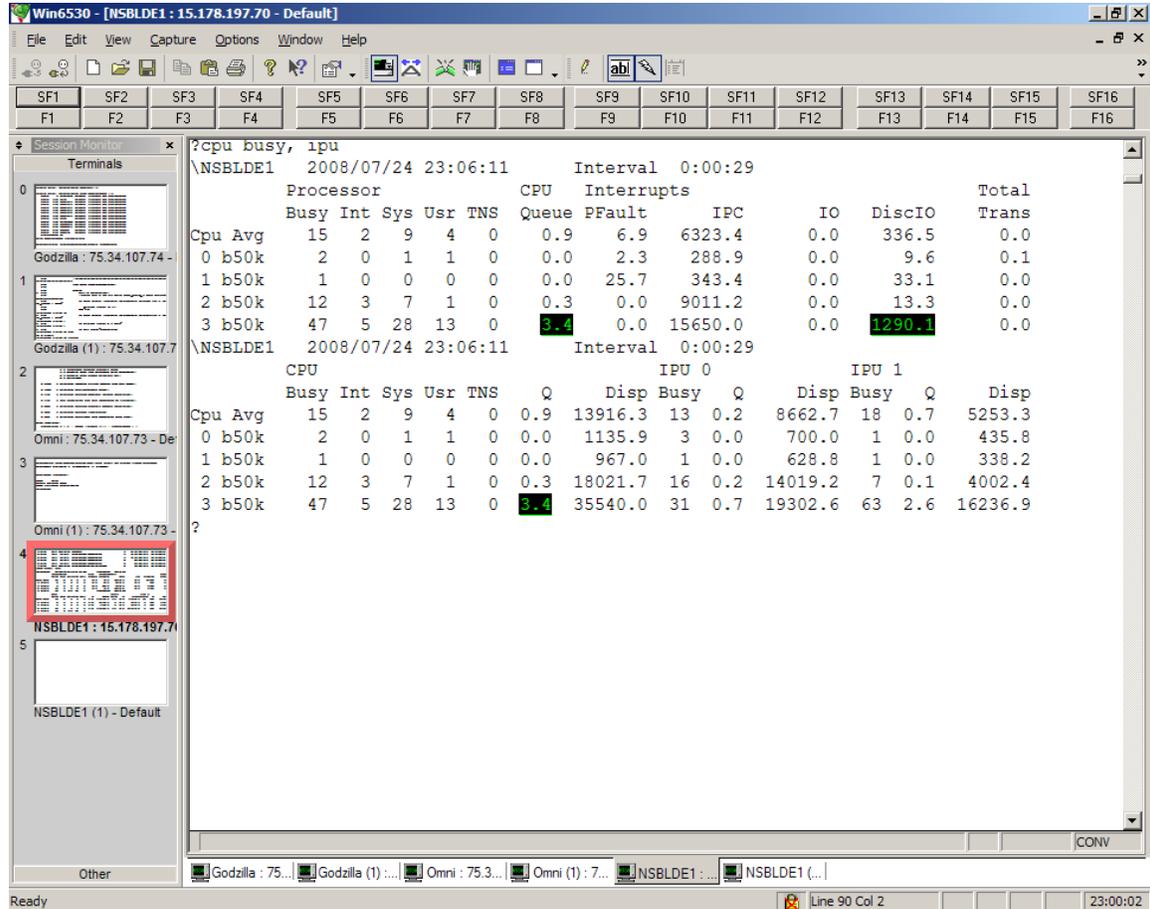
Processor	Busy	Int	Sys	Usr	TNS	Queue	PFault	IPC	IO	DiscIO
Cpu Avg	21	0	6	13	0	0.4	31.0	160.6	46.9	32.0
0 S74k	19	0	4	13	0	0.5	39.7	169.5	44.4	26.4
1 S74k	23	1	8	13	0	0.4	22.3	151.7	49.4	37.7

Interrupt	Rate	Page	Disp	IPC	IO	DiscIO	CHit
Cpu Avg	2	3290.7	38.4	2791.6	396.8	71.1	61.8
0 S74k	1	1826.0	75.9	1322.8	367.1	102.3	98.7
1 S74k	2	4756.9	0.9	4262.0	426.4	39.8	24.9

Interrupt	Send	Percent	Message Sizes						
Busy	Rcv/s	Busy	Send/s	UnSeq	Seq	0-64	65-256	<4097	>4096
Cpu Avg	1	287.9	0.0	253.7	8	91	0	0	0
0 S74k	1	275.9	0.0	252.0	8	91	0	0	0
1 S74k	2	299.9	0.0	255.7	8	91	0	0	0

OmniOffender V2.2

Cpu BUSY and IPU report; the IPU report is only available on J-Series.



The screenshot shows a Win6530 terminal window with the following content:

```

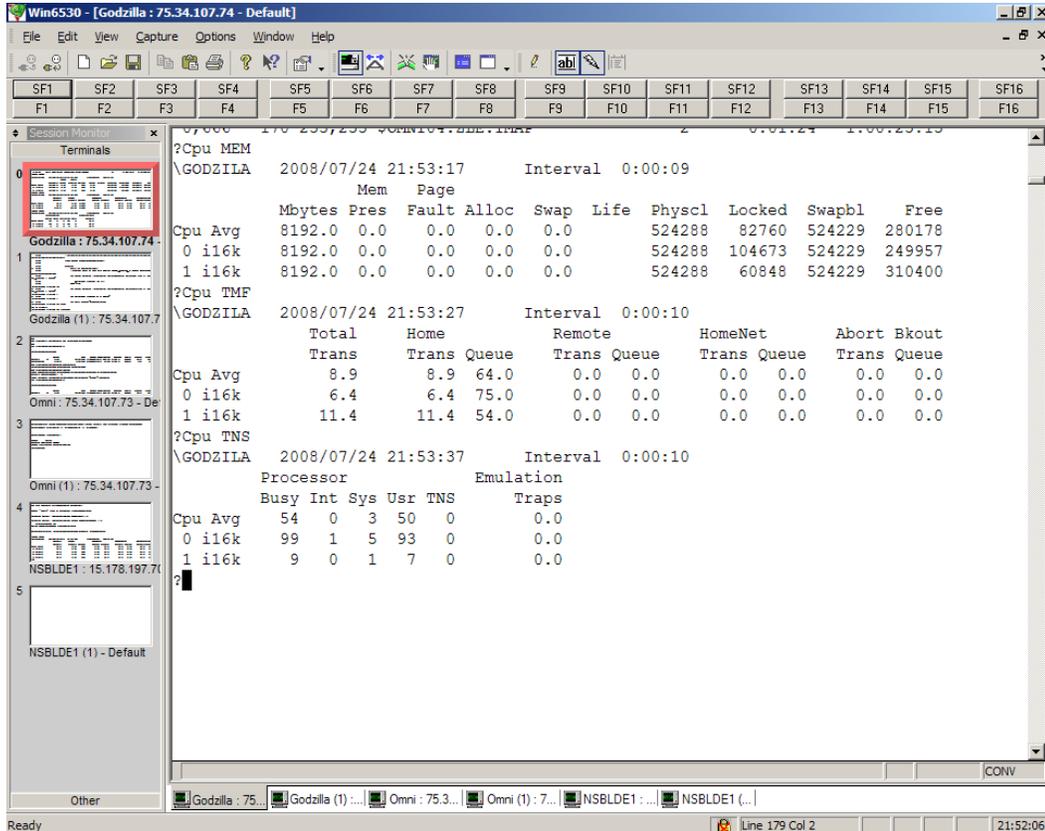
?cpu busy, ipu
\NSBLDE1 2008/07/24 23:06:11 Interval 0:00:29
Processor CPU Interrupts Total
Busy Int Sys Uxr TNS Queue PFault IPC IO DiscIO Trans
Cpu Avg 15 2 9 4 0 0.9 6.9 6323.4 0.0 336.5 0.0
0 b50k 2 0 1 1 0 0.0 2.3 288.9 0.0 9.6 0.1
1 b50k 1 0 0 0 0 0.0 25.7 343.4 0.0 33.1 0.0
2 b50k 12 3 7 1 0 0.3 0.0 9011.2 0.0 13.3 0.0
3 b50k 47 5 28 13 0 3.4 0.0 15650.0 0.0 1290.1 0.0
\NSBLDE1 2008/07/24 23:06:11 Interval 0:00:29
CPU IPU 0 IPU 1
Busy Int Sys Uxr TNS Q Disp Busy Q Disp Busy Q Disp
Cpu Avg 15 2 9 4 0 0.9 13916.3 13 0.2 8662.7 18 0.7 5253.3
0 b50k 2 0 1 1 0 0.0 1135.9 3 0.0 700.0 1 0.0 435.8
1 b50k 1 0 0 0 0 0.0 967.0 1 0.0 628.8 1 0.0 338.2
2 b50k 12 3 7 1 0 0.3 18021.7 16 0.2 14019.2 7 0.1 4002.4
3 b50k 47 5 28 13 0 3.4 35540.0 31 0.7 19302.6 63 2.6 16236.9
?
  
```

The terminal window also shows a 'Terminals' sidebar on the left with several sessions listed, including 'Godzilla' and 'Omni' sessions. A red box highlights a terminal window titled 'NSBLDE1: 15.178.197.7'.

OmniOffender V2.2



Cpu MEM, TMF, and TNS reports.



2.2 CPUS

Displays the list of CPUs to be sampled.

```
?cpus
Cpus : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
?cpus
```

2.3 CPUS (<cpu-list>)

Establishes the list of CPUs to be sampled.

```
<cpu-list> ::= { * | [ - ] <cpu> [ , <cpu> ... ] | [ - ] <cpu>/<cpu> }
```

- * Marks all CPUs to be sampled.
- - Subtracts CPUs from the already established list. This is used when there are configured down CPUs that should be ignored.

```
?cpus
Cpus : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
?cpus (-2)
Cpus : 0, 1, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
?cpus (-10/14)
Cpus : 0, 1, 3, 4, 5, 6, 7, 8, 9, 15
? cpus (*)
Cpus : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15
?
```

2.4 DELAY <number>

Establishes the default sample interval in seconds which is activated by GO.

The sample interval should be ≥ 10 seconds although the lower limit is one second.

A delay of zero will turn off sampling.

2.5 **Disc [<disc-report-option-list>] [(<cpu-list>)] [(<disc-list>)]**

Displays the selected disc activity report/s:

<disc-report-option> ::= { BAR | BUSY }

<disc-list> ::= { \$* | \$<vol> [, \$<vol> ...] }

<cpu-list> ::= { * | <cpu> [, <cpu> ...] | <cpu>/<cpu> }

- **BAR** (default) a bar graph of disc busy time.

w – disc write qbusy time.

r – disc read qbusy time.

- **BUSY** Displays the following disc volume statistics.

Disc Busy – Total time that the disc was reading or writing.

IOs – Physical IOs per second. Includes reads, writes, and control operations

Bytes – Bytes transferred per second to and from the disc.

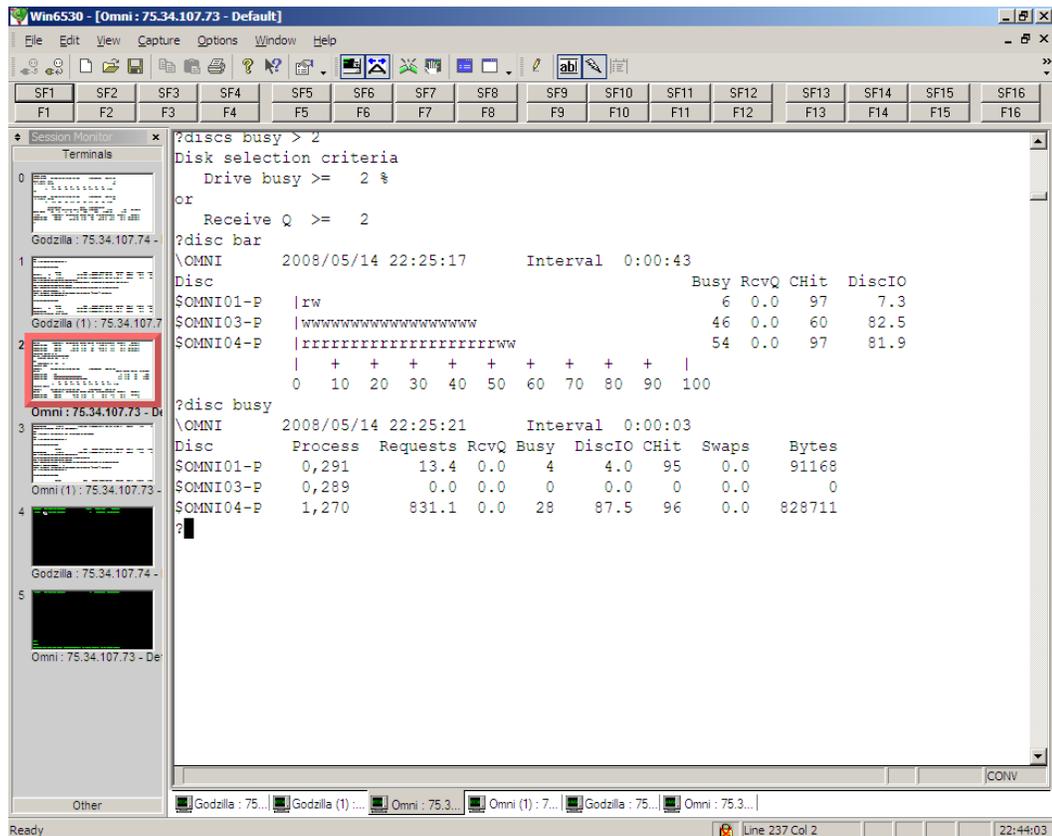
Reqs – Logical disc requests. May not result in an physical IO if it was for control information, or a hit in the disc cache.

Chits – Cache Hits. Percentage of cache hits for both reads and writes on all four DP2 disc caches.

RcvQ – the receive queue average depth of the head disc process. Queue length, not queue time is displayed.

OmniOffender V2.2

Disc BAR and BUSY reports. Note the command “discs busy > 2”, this overrides the default threshold of 20% and was done so as to create output.



```

Win6530 - [Omni: 75.34.107.73 - Default]
File Edit View Capture Options Window Help
SF1 SF2 SF3 SF4 SF5 SF6 SF7 SF8 SF9 SF10 SF11 SF12 SF13 SF14 SF15 SF16
F1 F2 F3 F4 F5 F6 F7 F8 F9 F10 F11 F12 F13 F14 F15 F16

+ Session Monitor x
Terminals
0
Godzilla : 75.34.107.74-
1
Godzilla (1) : 75.34.107.7
2
Omni : 75.34.107.73 - De
3
Omni (1) : 75.34.107.73-
4
Godzilla : 75.34.107.74-
5
Omni : 75.34.107.73 - De

?discs busy > 2
Disk selection criteria
Drive busy >= 2 %
or
Receive Q >= 2
?disc bar
\OMNI 2008/05/14 22:25:17 Interval 0:00:43
Disc Busy RcvQ CHit DiscIO
SOMNI01-P |rw 6 0.0 97 7.3
SOMNI03-P |wwwwwwwwwwwwwwwwwwww 46 0.0 60 82.5
SOMNI04-P |rrrrrrrrrrrrrrrrrrrrrrww 54 0.0 97 81.9
| + + + + + + + + + + |
0 10 20 30 40 50 60 70 80 90 100
?disc busy
\OMNI 2008/05/14 22:25:21 Interval 0:00:03
Disc Process Requests RcvQ Busy DiscIO CHit Swaps Bytes
SOMNI01-P 0,291 13.4 0.0 4 4.0 95 0.0 91168
SOMNI03-P 0,289 0.0 0.0 0 0.0 0 0.0 0
SOMNI04-P 1,270 831.1 0.0 28 87.5 96 0.0 828711
?
  
```

2.6 DISCS

Displays the list of disc volumes to measure.

```

?discs
Discs : $*
?
  
```

2.7 DISCS (<disc-list>)

Establishes the list of disc volumes to measure.

```
<disc-list> ::= { $* | <vol> [ , $<vol> ... ] }
```

```
?DISCS ($OMNI04, $AUDIT)
Discs : $OMNI04, $AUDIT
?
```

2.8 DISCS <discs-control>

Disks are displayed if they exceed the configurable amount of disk busy.

Each selected disk remains in the disk display, along with its mirror and the list is not sorted on busy, rather the disk's position is the order in which it met the selection criteria and was included in the display.

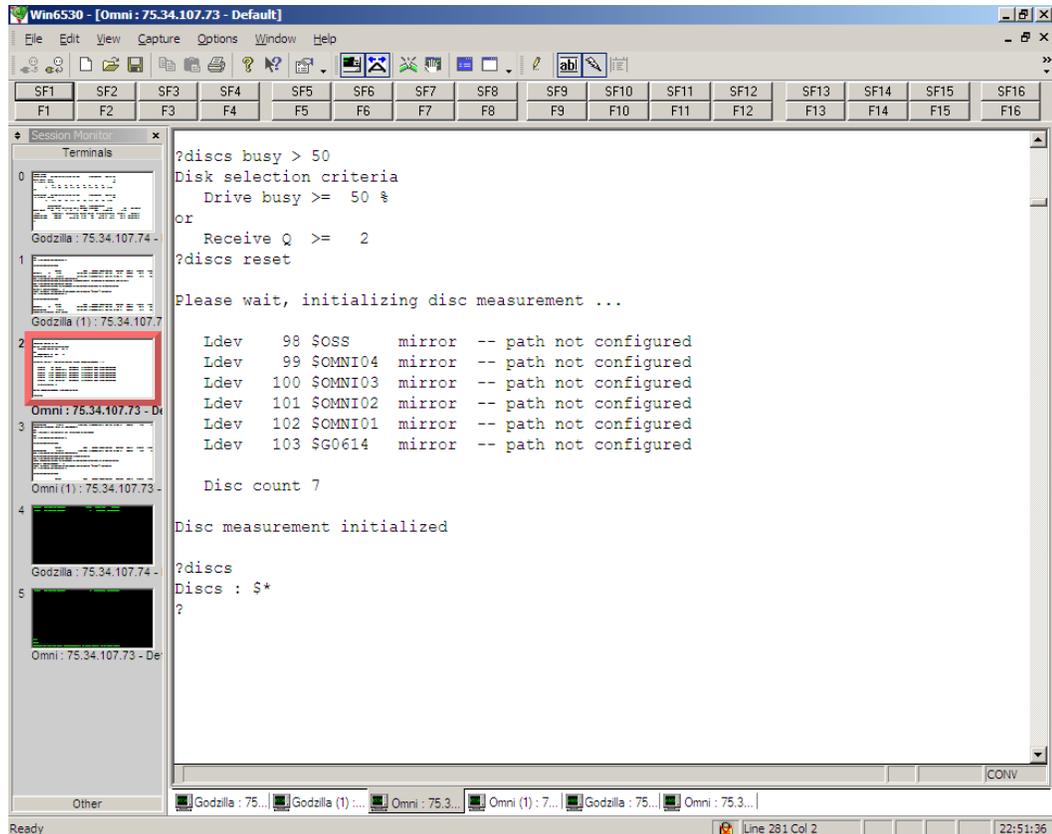
The display order can be reset using the reset control which empties the disk display list.

```
<discs-control> ::= { BUSY > <number> | RESET }
```

BUSY > <percent> Only disks busier than <percent> will be displayed. The default is 20%.

RESET Resets the disk display, disks will be added again when they meet the selection criteria controlled by the BUSY control

OmniOffender V2.2



2.9 ENV

Displays measure status and logging status.

```
?env
OmniOffender T0658J06^24JUL08
System: \NSBLDEL CFB 035120 J06.03.00
Measure File: $SYSTEM.OFFENDER.DOFF3, permanent.
Logging is off
?
```

2.10 EXIT

Terminates OFFENDER.

2.11 FC

Allows correction of the previous command.

2.12 File [*<file-report-option>*]

Displays FILE activity sorted by the specific FILE entity counter; FILE reports are available only for the busiest processes.

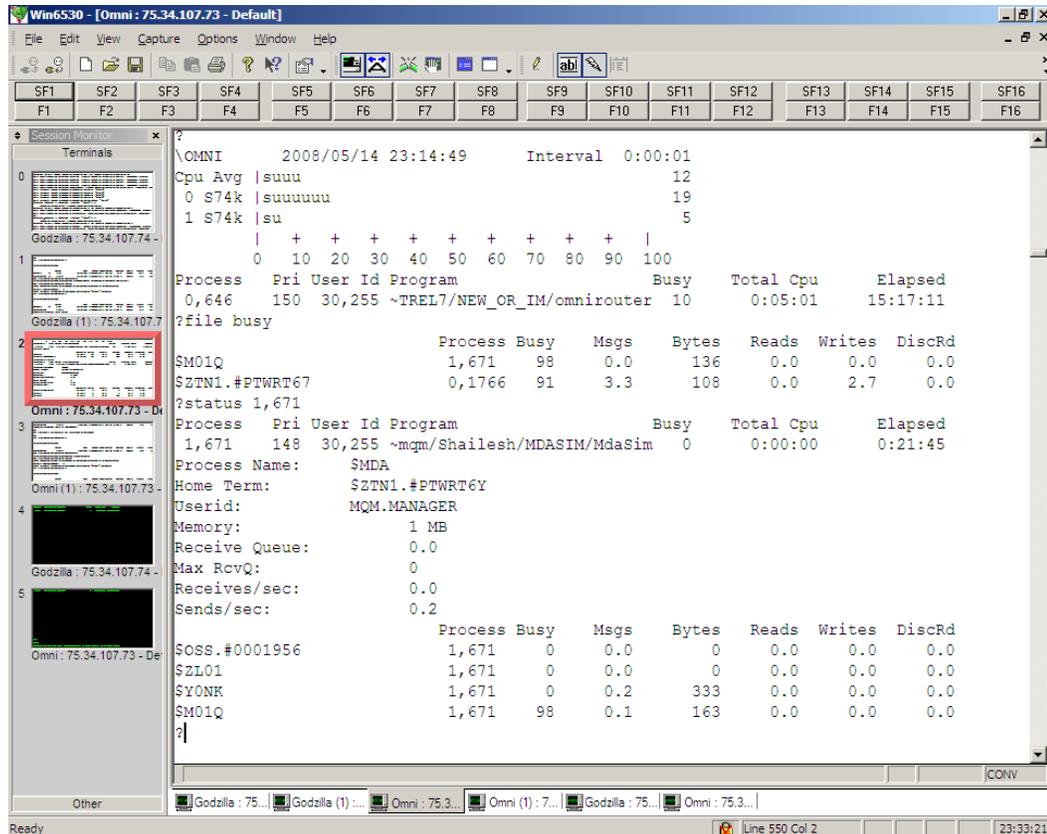
```
<file-report-option> ::= { ACCESSED | BUSY | BYTES | DISC |  
                           ESCALATIONS | INFO | LOCKS | MSGS |  
                           READS | RECORDS | TIMEOUTS | UPDATES  
                           | WRITES | WRITEREADS  
                           }
```

- ACCESSED SQL records accessed per second.
- BUSY (default) Displays files sorted by file busy time.
- BYTES Message bytes send & received per second.
- DISC Physical disc reads per second.
- ESCALATIONS Number of locks escalating to a file level lock.
- INFO FILEINFO calls per second.
- LOCKS Number of lock requests that waited per second.
- MSGS Messages sent per second.
- READS File reads per second.
- RECORDS SQL records used per second.
- TIMEOUTS Number of timeouts or cancels per second.
- UPDATES File updates or replies per second.
- WRITES File writes per second.
- WRITEREADS File deletes or WRITEREADS per second.

OmniOffender V2.2



Example File BUSY report.



Note the command "status 1,671" to obtain information regarding the process \$M01Q.

2.13 GO

Starts cyclic sampling and reporting.

q or <break> will terminate cyclic sampling and reporting.

Space bar or <enter> will report immediately.

Reports will be displayed in conversational page mode.

The first WINDOW will be at the 6530's page 0, the second at the next page and so forth.

For VT100 only one screen is available.

Multiple pages of reports can be specified with the WINDOW command and viewed with the terminal's local <page up> or <page down> keys.

2.14 HELP [ALL | <command>]

Use the "help" command to obtain syntax for all commands a specific command.

```
<command> ::= { Cpu | CPUS | DELAY | Disc | DISCS | ENV | EXIT |  
                FC | Files | GO | HELP | LOG | MEASURE | PProcess  
                | REPort | STATS | SStatus | Window | '?'  
                }
```

2.15 LOG

Reports logging status.

2.16 LOG [TO] <log-file>

Starts logging to <log-file>.

2.17 LOG STOP

Stops logging.

2.18 MEASURE

Displays measure status.

2.19 *PRocess* [*<process-report-option-list>*] [*(<cpu-list>)*]

Displays processes sorted by the requested metric.

```
<process-report-option> ::= { BUSY | FLTS | MEM | MSGS | RCVQ |
                               TIME
                             }
```

```
<cpu-list> ::= { * | { <cpu> [ , <cpu> ... ] | (<cpu>/<cpu> ) }
```

- BUSY (default) CPU busy is ≥ 2 percent
- FLTS process is currently waiting on a page fault or process fault rate is ≥ 1 /second
- MEM total number of pages of memory currently in use
- MSGS message send or receive rate is ≥ 1 /second
- RCVQ receive queue depth
- TIME total process time; this tends to favor I/O processes

OmniOffender V2.2



Sample PProcess BUSY report.

The screenshot shows a terminal window titled "Win6530 - [Omni : 75.34.107.73 - Default]". The terminal displays a "pr busy" report. The report is organized into sections for different terminals, with the first section being the most detailed. The data is as follows:

Terminal	Process	Pri	User Id	Program	Busy	Total Cpu	Elapsed
0	1,565	150	255,255	SOMNI04.ZLE33C.POP	30	3:01:23	16.12:28:53
	1,270	220	255,255	SOMNI04	19	1.09:59:59	68.07:49:33
1	1,270	220	255,255	SOMNI04	11	1.09:59:59	68.07:49:35
	1,565	150	255,255	SOMNI04.ZLE33C.POP	8	3:01:23	16.12:28:55
	0,288	220	255,255	SOMNI04	3	6:40:18	68.07:57:33
	1,284	220	255,255	SOMNI04	2	9:43:35	68.07:49:35
	1,274	220	255,255	SOMNI04	2	11:53:03	68.07:49:35
2	1,280	220	255,255	SOMNI04	2	10:05:26	68.07:49:35

2.20 QUIT

Terminates OFFENDER.

2.21 REPort <report-list>

Is equivalent to Window 1, 24, <report-list>

See window for additional information.

2.22 STATS

Displays current sampling statistics -- this is developer info.

```
?stats
```

```
Sampling stats -- this is developer info.
```

```
Errors are "normal" and expected!
```

```
0 CONFIG_GETINFO_BYNAME_ errors
0 MEASREADACTIVE errors
0 MEASREADACTIVE warnings

1431 PCBs sampled
14 Disk volumes
14 Disk paths sampled
0 Disk paths updated
0 Files in use
0 Files sampled
0 Files created
```

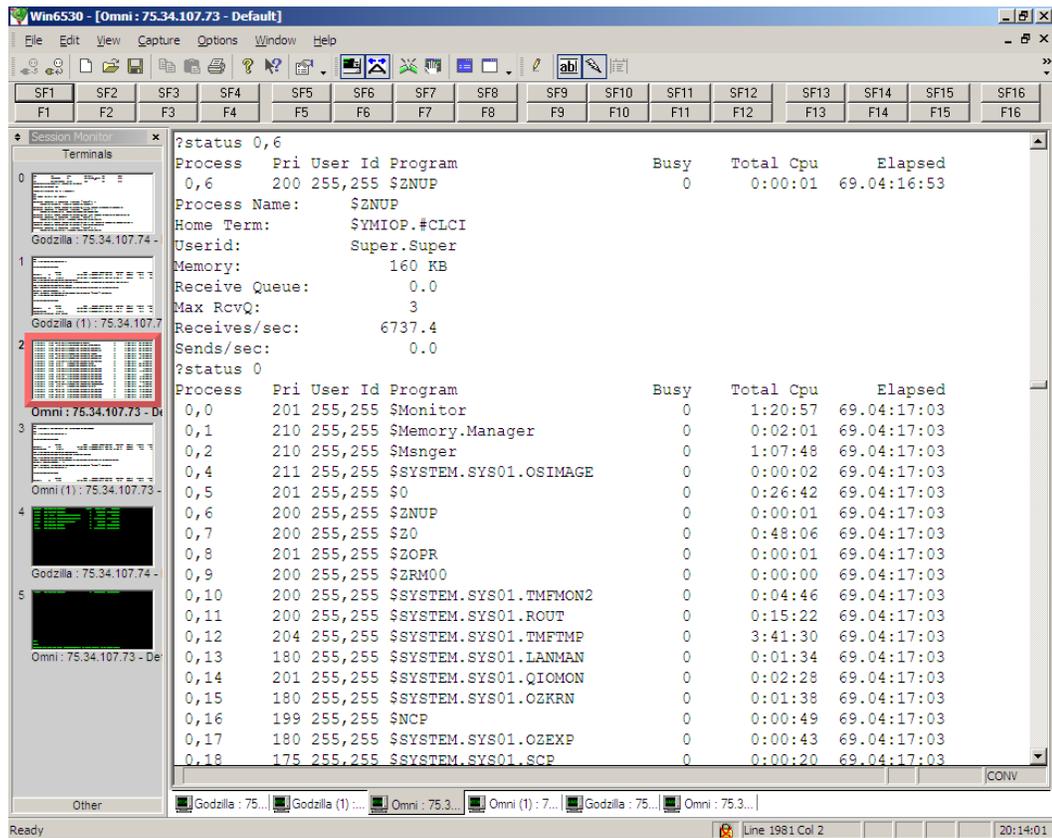
```
?
```

OmniOffender V2.2

2.23 SStatus <cpu> [, <pin>]

Displays detailed statistics for all processes in a CPU or for the specified process.

Sample SStatus reports.



The screenshot shows a terminal window titled "Win6530 - [Omni: 75.34.107.73 - Default]". The terminal displays the output of the command `?status 0,6` and `?status 0`.

Terminal Output for `?status 0,6`:

```

?status 0,6
Process Pri User Id Program Busy Total Cpu Elapsed
0,6 200 255,255 $ZNUP 0 0:00:01 69.04:16:53
Process Name: $ZNUP
Home Term: $YMIOP.#CLCI
Userid: Super.Super
Memory: 160 KB
Receive Queue: 0.0
Max RcvQ: 3
Receives/sec: 6737.4
Sends/sec: 0.0
  
```

Terminal Output for `?status 0`:

```

?status 0
Process Pri User Id Program Busy Total Cpu Elapsed
0,0 201 255,255 $Monitor 0 1:20:57 69.04:17:03
0,1 210 255,255 $Memory.Manager 0 0:02:01 69.04:17:03
0,2 210 255,255 $Msnger 0 1:07:48 69.04:17:03
0,4 211 255,255 $$SYSTEM.SYS01.OSIMAGE 0 0:00:02 69.04:17:03
0,5 201 255,255 $0 0 0:26:42 69.04:17:03
0,6 200 255,255 $ZNUP 0 0:00:01 69.04:17:03
0,7 200 255,255 $Z0 0 0:48:06 69.04:17:03
0,8 201 255,255 $ZOPR 0 0:00:01 69.04:17:03
0,9 200 255,255 $ZRM00 0 0:00:00 69.04:17:03
0,10 200 255,255 $$SYSTEM.SYS01.TMFMON2 0 0:04:46 69.04:17:03
0,11 200 255,255 $$SYSTEM.SYS01.ROUT 0 0:15:22 69.04:17:03
0,12 204 255,255 $$SYSTEM.SYS01.TMFTMP 0 3:41:30 69.04:17:03
0,13 180 255,255 $$SYSTEM.SYS01.LANMAN 0 0:01:34 69.04:17:03
0,14 201 255,255 $$SYSTEM.SYS01.QIOMON 0 0:02:28 69.04:17:03
0,15 180 255,255 $$SYSTEM.SYS01.OZKRN 0 0:01:38 69.04:17:03
0,16 199 255,255 $NCP 0 0:00:49 69.04:17:03
0,17 180 255,255 $$SYSTEM.SYS01.OZEXP 0 0:00:43 69.04:17:03
0,18 175 255,255 $$SYSTEM.SYS01.SCP 0 0:00:20 69.04:17:03
  
```

2.24 Window

Displays the configured reports.

```
?Window
Window[ 1 ]    24 lines
  Cpu Interrupts      *
  PRocess Busy        *
?
```

"Window 1, 24, CPU BAR *, PR BUSY *" is the default report for window 1.

2.25 Window <number>

Clears the reports for the specified window.

```
Window <number>

?w                                     B display current windows
Window[ 1 ]    24 lines
  Cpu Interrupts      *
  PRocess Busy        *
?w 2 24 cpu ipc 24                   B add 2nd window
Window[ 2 ]    24 lines
  Cpu PIN              24
?W                                     B display
Window[ 1 ]    24 lines
  Cpu Interrupts      *
  PRocess Busy        *
Window[ 2 ]    24 lines
  Cpu PIN              24
?W 2                                   B clear 2nd window
?W                                     B display
Window[ 1 ]    24 lines
  Cpu Interrupts      *
  PRocess Busy        *
?
```

2.26 Window <number> <window-size>

Changes the specified window size.

```
Window <number> <window-size>
```

```
?W                                     B display current windows
Window[ 1 ]    24 lines
  Cpu Interrupts      *
  PRocess Busy        *
?w 1 48                                     B change window size
Window[ 1 ]    48 lines
  Cpu Interrupts      *
  PRocess Busy        *
?W                                     B display
Window[ 1 ]    48 lines
  Cpu Interrupts      *
  PRocess Busy        *
?
```

2.27 Window <number> <window-size> <report-list>

For VT100 only Window 1 is available.

```
<number> ::= { 1, 2, ... 8 }
```

```
<window-size> ::= { 1, 2, ... } Note: best if a multiple of 24
```

```
<report> ::= <item>[ , <size> ] [( <cpu-list> ) | ( <disc-list> )]
```

Note: a <report-list> is limited to 10 reports

```
<item> ::= { Cpu <cpu-report-option>
              | Disc <disc-report-option>
              | File <file-report-option>
              | PRocess <process-report-option>
            }
```

```
<size> ::= { * | 1, 2, ... }
```

Note: best if a multiple of 24. * indicates use the rest of the window for the report.

OmniOffender V2.2

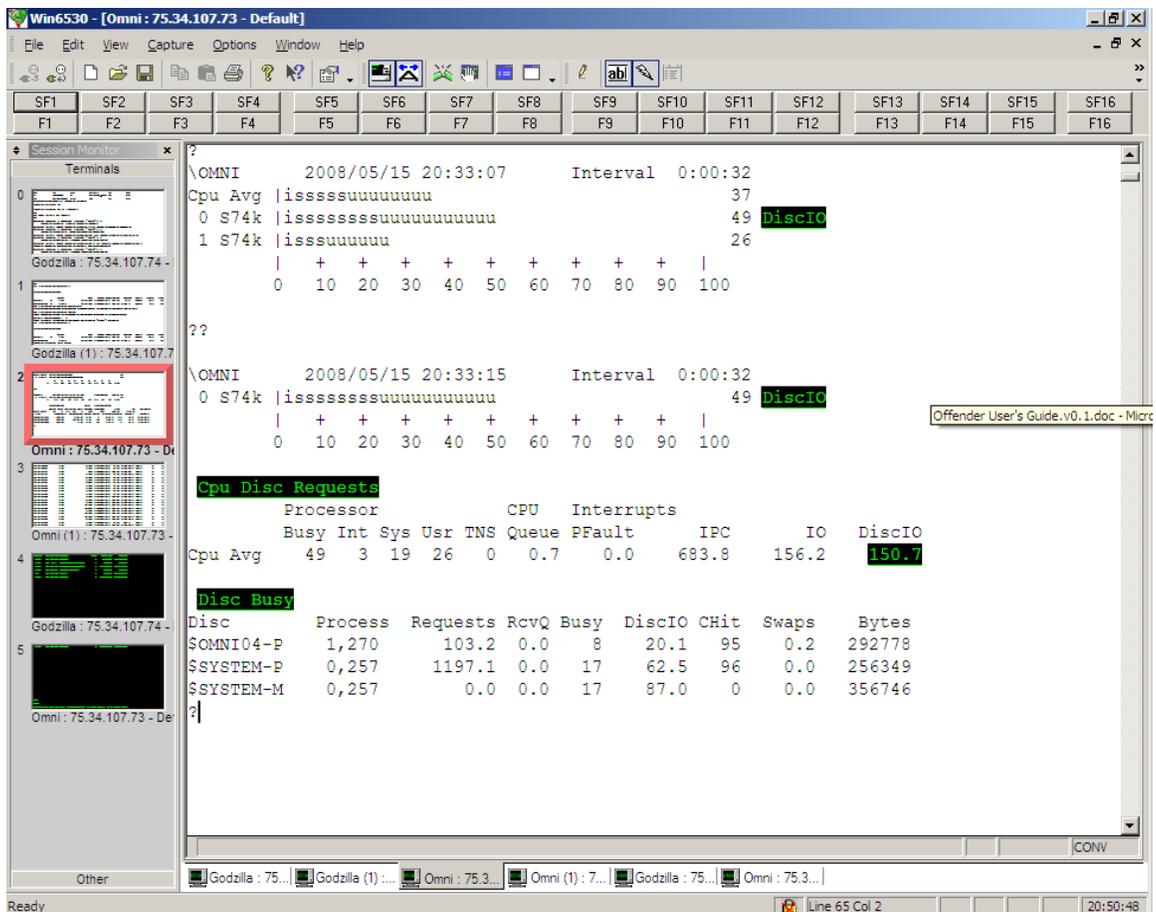
<cpu-list> ::= { * | <cpu> [, <cpu> ...] | <cpu>/<cpu> }

<disc-list> ::= { \$* | \$<vol> [, \$<vol> ...] }

More help information is available for CPU, DISC , FILE, or PROCESS.

2.28 ?

'?' will report any problems noted during previous measurement cycle.



Note the use of '?' to obtain information regarding disc I/Os.

3 Installation

OmniOffender is a MEASURE application which is safe to use. It does not use privileged code to go behind the systems back to get measurement data. It does not MUTEX, access counters itself, or fire up samplers. No calls to debug will be made.

OmniOffender is installed in \$SYSTEM.OFFENDER and needs to be prodir'd to the super group. Any operator id is fine.

Be sure to install the correct binary; at the moment each there is a different binary for the G, H, and J releases of Guardian.

Use file and/or SAFEGUARD security to restrict access as is appropriate for the installation.

OmniOffender requires a license file which must be installed in \$SYSTEM.OFFENDER.LICENSE. The license file must be secured so OmniOffender can read it.

4 About OmniOffender

4.1 *Version 2.2*

Version procedure: T0658G06^26AUG08
Version procedure: T0658H06^26AUG08
Version procedure: T0658J06^26AUG08

- Changed to use standard Opsol License files.

4.2 *Version 2.1*

Version procedure: T0658G06^24JUL08
Version procedure: T0658H06^24JUL08
Version procedure: T0658J06^24JUL08

- Added support J-Series.
- Added CPU IPU option (only supported on J-Series).
- Added CPU TMF option.
- Cpu, Disc, File, and PRocess options will default to BAR, BAR, BUSY, and BUSY respectively.
- Reports will print in the order input; for example cpu busy, tmf will print the busy report first and tmf last. Previously the order was reversed.
- ENV command will report release and offender version.

5 Index

?, 26, 29
<command>, 2, 19
<cpu>, 4, 7, 12, 13, 20, 23, 26
<cpu-list>, 3, 4, 7, 12, 13, 20, 25, 26
<cpu-report-option>, 7, 25
<cpu-report-option-list>, 7
<disc-list>, 3, 4, 13, 15, 25, 26
<disc-report-option>, 13, 25
<disc-report-option-list>, 13
<discs-control>, 15
<file-report-option>, 17, 25
<log-file>, 19
<number>, 15, 24, 25
<percent>, 15
<pin>, 23
<process-report-option>, 20, 25
<process-report-option-list>, 20
<report>, 25
<report-list>, 22, 25
<seconds>, 12
<window-size>, 25
Commands
?, 26, 29
Cpu, 7
CPUS, 11, 12
DELAY, 12
Disc, 13
DISCS, 14, 15
ENV, 16
EXIT, 16
FC, 17
FILE, 17
GO, 18
HELP, 19
LOG, 19
MEASURE, 19
PRocess, 20
QUIT, 22
REPort, 22
STATS, 22
STatus, 23
Window, 24, 25
Cpu, 7
CPUS, 11
DELAY, 12
Disc, 13
DISCS, 14, 15
ENV, 16
EXIT, 16
FC, 17
FILE, 17
GO, 18
HELP, 19
LOG, 19
MEASURE, 19
PRocess, 20
QUIT, 22
REPort, 22
STATS, 22
STatus, 23
Window, 24, 25