PEEK Reference Manual

Abstract

This manual describes PEEK, a utility used to monitor the statistical data of processors in HP Integrity NonStop™ BladeSystems, HP Integrity NS-series systems, and HP NonStop S-series systems. It is intended for use by individuals responsible for operating, managing, and servicing NonStop systems.

Product Version

PEEK G09

Supported Release Version Updates (RVUs)

This publication supports J06.03 and all subsequent J-series RVUs, H06.03 and all subsequent H-series RVUs, and G06.12 and all subsequent G-series RVUs, until otherwise indicated by its replacement publications.

Part Number	Published
529657-005	May 2010

Document History

Part Number	Product Version	Published
529657-001	PEEK G09	June 2005
529657-002	PEEK G09	November 2005
529657-003	PEEK G09	May 2008
529657-004	PEEK G09	February 2009
529657-005	PEEK G09	May 2010

Legal Notices

Printed in the US

PEEK Reference Manual

Glossary | Index | Figures | Tables

2. PEEK Syntax and Examples (continued)

Glossary

Index

Figures

Tables

Tables (continued)

What's New in This Manual

Manual Information

Abstract

This manual describes PEEK, a utility used to monitor the statistical data of processors in HP Integrity NonStop™ BladeSystems, HP Integrity NS-series systems, and HP NonStop S-series systems. It is intended for use by individuals responsible for operating, managing, and servicing NonStop systems.

Product Version

PEEK G09

Supported Release Version Updates (RVUs)

This publication supports J06.03 and all subsequent J-series RVUs, H06.03 and all subsequent H-series RVUs, and G06.12 and all subsequent G-series RVUs, until otherwise indicated by its replacement publications.

Part Number	Published
529657-005	May 2010

Document History

Part Number	Product Version	Published
529657-001	PEEK G09	June 2005
529657-002	PEEK G09	November 2005
529657-003	PEEK G09	May 2008
529657-004	PEEK G09	February 2009
529657-005	PEEK G09	May 2010

New and Changed Information

Changes to the 529657-005 manual:

- Updated the PEEK POOL output on page 1-3, 2-6, 2-11, 2-19, 2-20, 2-41, 2-45.
- Updated the elements of the POOL Display table on page 2-42.

Changes to the H06.17/J06.06 Manual

- Updated description of the CME option on pages 1-5 and 2-2.

- Updated Figure 2-4, PEEK ALL Listing (for H-series RVUs and J-series RVUs), on page 2-11.
- Updated information on CME Option on page 2-13.

Changes to the H06.14/J06.03 Manual

- References to Release Version Updates (RVUs) throughout this manual have been updated to include references to J-series RVUs, where appropriate.
- Updated the example for `PEEK CME` command under CME Option on page 2-13.
- Removed information about `EXPEDITED` and `IO` parameters under HELP Option on page 2-18.
- Updated the PEEK output display with `Num IPUs` option for J-series in , Figure 2-2 shows an example of the PEEK default for H-series RVUs and J-series., on page 2-6.

Changes to the H06.04 Manual

- Added the NSAA option under:
  - Syntax to Run PEEK on page 2-1
  - ALL option on page 2-2
  - PEEK ALL listing for H-series RVUs on page 2-10
- Added the description of NS[AA] on page 2-3.
- Updated the PEEK ALL output display with new elements, BLADE A, BLADE B, and BLADE C, in Figure 2-4, PEEK ALL Listing (for H-series RVUs and J-series RVUs) on page 2-12.
- Added the description for the new elements, BLADE A, SUCCESSREINT, LASTSUCCREINTTIME, BLADE B, BLADE C, and CPU *n*, in Table 2-6, NSAA Elements Reported in the NSAA Display on page 2-34.

Changes to the H06.03 Manual

- Added the description for these elements under Using MQCINFO Option in Table 2-5 on page 2-30:
  - TOTAL ALLOCATED SEGMENTS
  - HIGH ALLOCATED SEGMENTS
  - TOTAL ALLOCATED PAGES
  - HIGH ALLOCATED PAGES
  - TOTAL FREE PAGES

- Added the output details under Examples after Report 2: on page 2-32.
- Added the output of the PEEK MQCINFO command on page 2-28.
- Added the output of the PEEK command in Figure 2-1 on page 2-5.
- Added the output of the PEEK ALL command in Figure 2-3 on page 2-8.
- Added the description of the NSAA Option on page 2-33.

About This Manual

This manual describes PEEK, a utility that reports statistical information maintained by the HP NonStop operating system.

Who Should Use This Manual

This manual is intended for those who manage and service NonStop systems. Because PEEK is designed to monitor statistical data about elements found within the operating system, this manual is most useful to those who manage a system or who perform performance analysis and tuning.

Readers of this manual should understand NonStop system operations and the NonStop operating system.

Purpose of This Manual

This manual provides you with information needed to monitor statistical information about system tables and resources, storage pools, physical memory, and other entities.

This manual's primary purpose is to describe the complete syntax of all PEEK options. This manual also provides quick answers to specific questions readers have about PEEK and its options.

The secondary purpose of this manual is to provide illustrations and detailed examples of PEEK displays and to describe important considerations for the use and understanding of PEEK.

How This Manual Is Organized

This manual contains these sections:

Section	Contents
Getting Started With PEEK	General information about PEEK and the type of statistical information you can obtain with this utility.
PEEK Syntax and Examples	The complete syntax of all PEEK options. Considerations for using the options and examples of their use are also included.
Glossary	Definitions of technical terms used in this manual.
Index	Index entries

Related Reading

You might want to refer these manuals for more information:

- The *TACL Reference Manual* for information about TACL in general and about using the RUN command and its options
- The *System Generation Manual for G-Series RVUs* for explanations of some entities that PEEK monitors, such as time-list elements (TLEs)
- The *Measure Reference Manual* and the *Measure User's Guide* for information about the Measure program, which provides more detailed information about a system's activity and status
- The *NonStop S-Series Operations Guide* and *NonStop NS-Series Operations Guide* for operations instructions

Notation Conventions

General Syntax Notation

This list summarizes the notation conventions for syntax presentation in this manual.

UPPERCASE LETTERS. Uppercase letters indicate keywords and reserved words. Enter these items exactly as shown. Items not enclosed in brackets are required. For example:

MAXATTACH

lowercase italic letters. Lowercase italic letters indicate variable items that you supply. Items not enclosed in brackets are required. For example:

file-name

computer type. `Computer type` letters within text indicate C and Open System Services (OSS) keywords and reserved words. Type these items exactly as shown. Items not enclosed in brackets are required. For example:

```
myfile.c
```

italic computer type. *`Italic computer type`* letters within text indicate C and Open System Services (OSS) variable items that you supply. Items not enclosed in brackets are required. For example:

`pathname`

[ ] Brackets. Brackets enclose optional syntax items. For example:

TERM [*system-name*.]$*terminal-name*

INT[ERRUPTS]

A group of items enclosed in brackets is a list from which you can choose one item or none. The items in the list may be arranged either vertically, with aligned brackets on

each side of the list, or horizontally, enclosed in a pair of brackets and separated by vertical lines. For example:

```
LIGHTS [ ON             ]
       [ OFF            ]
       [ SMOOTH [ num ] ]

K [ X | D ] address-1
```

… Ellipsis. An ellipsis immediately following a pair of brackets indicates that you can repeat the enclosed sequence of syntax items any number of times. For example:

```
M address-1 [ , new-value ]...

[ - ] {0|1|2|3|4|5|6|7|8|9}...
```

An ellipsis immediately following a single syntax item indicates that you can repeat that syntax item any number of times. For example:

```
"s-char..."
```

Punctuation. Parentheses, commas, semicolons, and other symbols not previously described must be entered as shown. For example:

```
error := NEXTFILENAME ( file-name ) ;

LISTOPENS SU $process-name.#su-name
```

Quotation marks around a symbol such as a bracket or brace indicate that the symbol is a required character that you must enter as shown. For example:

```
"[" repetition-constant-list "]"
```

Item Spacing. Spaces shown between items are required unless one of the items is a punctuation symbol such as a parenthesis or a comma. For example:

```
CALL STEPMOM ( process-id ) ;
```

If there is no space between two items, spaces are not permitted. In this example, there are no spaces permitted between the period and any other items:

```
$process-name.#su-name
```

Line Spacing. If the syntax of a command is too long to fit on a single line, each continuation line is indented three spaces and is separated from the preceding line by a blank line. This spacing distinguishes items in a continuation line from items in a vertical list of selections. For example:

```
ALTER [ / OUT file-spec / ] CONTROLLER

   [ , attribute-spec ]...
```

Notation for Messages

This list summarizes the notation conventions for the presentation of displayed messages in this manual.

Nonitalic text. Nonitalic letters, numbers, and punctuation indicate text that is displayed or returned exactly as shown. For example:

```
Backup Up.
```

% Percent Sign. A percent sign precedes a number that is not in decimal notation. The % notation precedes an octal number. The %B notation precedes a binary number. The %H notation precedes a hexadecimal number. For example:

```
%005400
```

```
P=%p-register E=%e-register
```

Change Bar Notation

Change bars are used to indicate substantive differences between this manual and its preceding version. Change bars are vertical rules placed in the right margin of changed portions of text, figures, tables, examples, and so on. Change bars highlight new or revised information. For example:

> The message types specified in the REPORT clause are different in the COBOL environment and the Common Run-Time Environment (CRE).
>
> The CRE has many new message types and some new message type codes for old message types. In the CRE, the message type SYSTEM includes all messages except LOGICAL-CLOSE and LOGICAL-OPEN.

1 Getting Started With PEEK

What is PEEK?

PEEK is a utility that reports the statistical information maintained by the operating system. You can use PEEK to monitor processor activity for system storage pools, paging activity, message information, send instructions, and interrupt conditions.

This section describes how to:

- Enter the command to run PEEK
- Monitor another system in the network
- Use PEEK options
- Use RUN options to specify a processor on which you want PEEK to run a report and to direct the output of PEEK reports to a device

Running PEEK

To obtain a PEEK report on a processor on your current system, enter:

```
> PEEK
```

This command results in the default PEEK display, which gives statistical data about three areas of a processor's activity: TIME, POOL, and PAGING statistics.

```
PEEK - T9050H02 - (02AUG10) - (23FEB10) - (AUI)     SYSTEM \TAHOE
(C)1981 TANDEM (C)2004-2008 HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.
                              SYSTEM   \TAHOE
23 MAR 2010,  6:48___ELAPSD    1:25:19___CPU  3(NSE-D/NS14000)
TIME:     PROCESSBUSY TIME         INTERRUPT TIME          IDLE TIME
          0:01:04.650   1.26%     0:00:02.317   0.04%     1:24:12.085  98.69%

          MAXIMUM USED  CURRENT USAGE  # CONFIGURED  # OF FAILURES
TLE             16             15          20000             0
PCB      9:     47      8:     47    255:   7830      0:      0
NRL            459            244          32767             0
PTLE             1              1            681             0
PME              2              2          65501             0

           MAX.SIZE  CUR.SIZE  INIT.CNF  MAX.USED  CUR.USED MAX.FRAG CUR.FRAG
SYSPOOL        5520      5520     13249      5520      5520        0        0
EXTPOOL           0         0    262144         0         0        0        0
MAPPOOL       94656     94656    196585     90688     88864       35       19
FLEXPOOL    3931792   3931792   2097106   3783640   3770152       38       20
SEG TBL       13591     13591     13591      1990      1990    10233       18

                        FLEXPOOL SUBPOOL USAGE
         MAXIMUM         CURRENT       ALLOCATED     DEALLOCATED
            8               8               8              0

POOL64 USAGE         TOTAL SIZE          ALLOCATED   LOCKED/WIRED    SEGMENTS
------------     ------------------     ----------   ------------    --------
                 INIT    CUR    MAX     CUR    MAX     CUR    MAX    CUR  MAX
MAPPOOL64        64MB   64MB   64MB  4397KB 4417KB       -      -      1    1
FLEXPOOL64     2048MB 2048MB 2048MB  39184B 39664B  49152B 49152B      1    1

PAGES:  PHYSCL  SWAPBL   FREE   FREEMIN FREEQTA FREERED  UNDUMPED
(16Kb) 1048576 1048517 1013151  497491     640     320         0

PAGES:      LOCKED       LOCKED(KSEG0)
(16Kb)    28569/917453    2882/12244
```

```
                FAULTS      ALLOCS   DISKREADS DISKWRITES  MUTEXCRAX NONMUTEXCRAX
TOTAL            11983      739987        6741         29        185          104
(per sec)         2.34      144.54        1.31       0.00       0.03         0.02

                REDHIT    REDBUSY    REDTASK
TOTAL                0          0          0
(per sec)         0.00       0.00       0.00

CLEANQ:   FULLS  FRLST:HITS  CLOCK:CALLS    FAILS    CYCLES   ALIASES:  FAILS
0             0      739234          753        0      1.02          0       0
```

Note.

- PEEK examines the internal data structures of the operating system. These internal data structures might differ with each RVU. To avoid compatibility issues, use the product version (PV) of PEEK that matches the RVU of the operating system running in the processor that PEEK examines.
- For systems running J06.06 and earlier J-series RVUs or H06.17 and earlier H-series RVUs, NRL is displayed as a four-line output with the small-index (S) and medium-index (M) entries for NRL and PPL on separate lines. This four-line output is compressed to a single-line NRL output from J06.07/H06.18 series onwards.
- For systems running J06.07 and later J-series RVUs or H06.18 and later H-series RVUs, the PME table information is included in the output.
- For systems running J06.09 and later J-series RVUs or H06.20 and later H-series RVUs, the FLEXPOOL64 information is included in the output.

Monitoring Another System

You can also use PEEK to monitor processor activity on another system in your network. For example:

```
> \KONA.PEEK
```

This command returns the default TIME, POOL, and PAGING statistics for a processor on the node named \KONA.

Using PEEK Options

Use the PEEK options listed in Table 1-1 to obtain specific statistical information about processors and to help you run the PEEK program.

Table 1-1. PEEK Options

Option	Function
ALL	Displays the information produced by all of the PEEK options.
CME	Displays relevant information related to correctable memory errors (CMEs) that have occurred since the processor was last loaded.
delay	Is the amount of time, in seconds, that you want PEEK to pause between samples. The range is 0 through 999. *delay* must be used together with *samples* and must appear after *samples*.
DYNAMIC	Used in combination with *samples* and other options, displays processor statistics at different time intervals.
HELP	Displays a syntax summary of all PEEK options.
INIT	(Super group only) Resets TLE, process control block (PCB), link control block (LCB), and process time-list element (PTLE) maximums to equal the values in the CURRENT columns in the PEEK report.
INTERRUPTS	Displays a count of software interrupts by type.
MESSAGES	Displays the number of unsequenced packets, control packets, and data messages the processor sends.
MQCINFO	Displays information about message quick cell (MQC) resources.
NSAA (H-series RVUs)	Displays VRO and *Inappropriate I/O Buffer Access* counters and reintegration status information. *This information is not applicable to J-series and will be removed from J-series output in a future release of J-series.
PAGING	Displays processor activity for paging statistics (default).
POOL	Displays the state of system tables and resources (default).
samples	Specifies the number of times you want PEEK to report system values. The range is 0 through 999. The number you enter for *samples* must appear before *delay*.
TIME	Displays the amount of time a given processor has spent on processes, interrupts, and idle time (default).

Using RUN Options

You can include any valid option of the TACL command interpreter RUN command when you run PEEK. Three useful RUN options are:

- CPU *nn* , which specifies the processor about which a PEEK report is run.
- OUT *filename*, which specifies a destination for a PEEK report, such as a disk file or printer.
- NOWAIT, which returns your TACL prompt while a lengthy PEEK report is being directed to a disk file, printer, or other device.

Specifying a Processor With the CPU Option

Unless you specify a processor, PEEK normally reports on the processor in your system that is running the TACL command interpreter or on a processor chosen by $CMON. However, you can select a different processor by using the CPU option of the RUN command. For example:

```
> PEEK / CPU 2 /
```

This command returns the TIME, POOL, and PAGING statistics for processor 2 on your current system.

Directing PEEK Reports With the OUT Option

Use the OUT *filename* option of the RUN command to direct the output of a PEEK report to any device, such as a line printer, or to a disk file, including an EDIT file. For example:

```
> PEEK / OUT $DISK.MYFILES.PEEK1 /
```

This command directs the PEEK report to a file named PEEK1 on a subvolume named MYFILES on the volume $DISK on the current system.

Determining When to Use the NOWAIT Option

The NOWAIT option is useful when you combine it with the OUT *filename* option to produce a series of reports. For example:

```
> PEEK / OUT $DISK.MYFILES.PEEK1, NOWAIT / DYNAMIC, 96, 900
```

This command directs the PEEK report to the file PEEK1 and produces reports every 15 minutes over a 24-hour period. While the reports are being generated, you retain access to the TACL command interpreter.

Combining RUN Options

You can combine RUN options on a single command line to perform multiple operations more quickly. This example runs a PEEK report on processor 2, directs the output to the EDIT file PEEK1, and returns immediately to the TACL prompt:

```
> PEEK / CPU 2, OUT $DISK.MYFILES.PEEK1, NOWAIT /
```

For detailed instructions on using the RUN command and its options, see the *TACL Reference Manual*.

Examples

1. If you do not enter an option, information for the TIME, POOL, and PAGING options is displayed by default. For example:

   ```
   > PEEK
   ```

2. To use a PEEK option, enter it after the PEEK command at your TACL prompt. For example:

   ```
   > PEEK TIME
   ```

3. You can enter multiple PEEK options in any order on a single command line, but you must separate them from each other with either spaces or commas. For example:

   ```
   > PEEK TIME, CME, MESSAGES
   ```

4. When you provide values for *samples* and *delay*, PEEK displays information about the data you specify one or more times, delaying between displays for a time interval that you specify. For example:

   ```
   > PEEK 4, 30
   ```

   This command returns four default reports at intervals of 30 seconds. If you do not include values for *samples* and *delay*, PEEK returns one report.

5. You can combine PEEK options with RUN options. For example:

   ```
   > PEEK / CPU 2 / TIME, 4, 30
   ```

   This command returns four TIME reports on processor 2 at intervals of 30 seconds.

6. To run a PEEK TIME report on processor 4 of the remote system \KONA, enter:

   ```
   > \KONA.PEEK / CPU 4 / TIME
   ```

7. To run four default PEEK reports on processor 2 of your current system at intervals of 30 seconds, direct the reports to the EDIT file PEEK1, and return immediately to your TACL prompt:

   ```
   > PEEK / CPU 2, OUT $DISK.MYFILES.PEEK1, NOWAIT / 4, 30
   ```

2 PEEK Syntax and Examples

This section contains the following:

- Syntax to run PEEK with brief descriptions of each of the PEEK options
- Default listing formats for PEEK
- Detailed information about each of the PEEK options with illustrations and examples

---

Note. Because of width constraints, some examples in this manual might not be aligned exactly as the PEEK output is.

Syntax to Run PEEK

The syntax of the command to run PEEK is:

```
[ \node. ] PEEK [/ run-options /] [ option ] [ ,option ] ...
```

where *option* can be any of:

```
samples
delay
ALL
CME
D[YNAMIC]
HELP
INIT
INT[ERRUPTS]
MES[SAGES]
MQC[INFO]
NS[AA]
PA[GING]
PO[OL]
TIME
```

\node

is the name of a node on your network (other than your current node) on which you want to run PEEK. If you do not specify *\node*, the PEEK report is run for your current node.

run-options

is one or more options for the TACL RUN command. For examples of using *run-options* with PEEK, see Section 1, Getting Started With PEEK. For a complete list and explanation of *run-options,* see the *TACL Reference Manual*.

Include the TACL CPU option to specify the number of the logical processors for which you want data. Otherwise, PEEK reports on the processor that is running the TACL command interpreter or on a processor chosen by $CMON. Also, include the

OUT option if you want to send PEEK output to a file or location other than your home terminal.

option

is one or more of these PEEK options. Separate the options from each other with either a space or a comma. Each option is described in greater detail later in this section. If you do not enter any options, the PEEK default listing format displays information for the TIME, POOL, and PAGING options.

Except for the *delay* option, you can enter the PEEK options in any order. The *delay* option must always be preceded by the *samples* option.

samples

is the number of times you want PEEK to report the system values. The range of values is 0 through 999. If you omit *samples*, or type 0 or 1, PEEK creates one sample.

You can enter *samples* before or after any of the other options except *delay. Samples* must always appear before *delay*.

delay

is the amount of time, in seconds, that you want PEEK to pause between successive samples. The range is 0 through 999. If you omit *delay* or enter 0 or 1, PEEK responds with a 1-second delay.

If you enter *delay*, you must also enter *samples* before *delay*.

ALL

displays information about:

TIME
POOL
NSAA
PAGING
MESSAGES
MQCINFO
INTERRUPTS
CME

CME

displays relevant information related to CMEs that have occurred since the processor was last loaded.

D[YNAMIC]

must be used in combination with *samples*. The first sample displays information about a processor's activity since the processor was loaded. If you specify DYNAMIC, successive samples display only the activity that occurs during each sample interval.

Specify DYNAMIC instead of INIT when you want to monitor processor activity for a relatively short time period (15 minutes or less). DYNAMIC preserves the measured maximums that are listed in each PEEK report.

HELP

displays a syntax summary of all PEEK options.

INIT (Super group (255,*) only)

resets the values for pool elements that are stored by PEEK. When you specify INIT, PEEK resets all pool maximums to equal the values in the CURRENT columns in the PEEK report. INIT also resets the MAXIMUM USED values for time-list element (TLE) entries and the process control block (PCB) entries to the CURRENT USED values.

Use INIT only when you want to initialize (and thus destroy the past history of) pool-related maximums stored by PEEK. Specify DYNAMIC instead of INIT when you want to monitor processor activity for a relatively short time period (15 minutes or less).

INT[ERRUPTS]

displays a count of software interrupts by type. For more information, see INTERRUPTS Option on page 2-21.

MES[SAGES]

displays the number of unsequenced packets, control packets, and data messages sent by the processor and provides statistical data about the processor's message quick cells (MQCs). For more information, see MESSAGES Option on page 2-25.

MQC[INFO]

displays information on the message quick cells (MQCs). Message quick cells are data structures that the message system uses for interprocess communication. The system automatically builds and allocates MQCs as it needs them. MQCs serve a purpose similar to link control blocks (LCBs) and extended memory link control blocks (XLIs), which were used in earlier RVUs. For more information, see MQCINFO Option on page 2-27.

`NS[AA]`

displays voluntary rendezvous opportunities (VRO) and inappropriate I/O buffer access counters and reintegration status. For more information, see NSAA Option on page 2-33.

`PA[GING]`

displays paging statistics. For more information, see PAGING Option on page 2-35.

`PO[OL]`

displays pool-management statistics. For more information, see POOL Option on page 2-40.

`TIME`

displays the amount of time the processor has spent on processes, interrupts, and idle time. For more information, see TIME Option on page 2-47.

Consideration

On high-activity systems or systems that rarely undergo a system load, the internal structures that collect PEEK statistics can reach their maximum capacity and then overflow. In PEEK output, an overflow condition is indicated by a negative number or a string of asterisks (****).

Examples

These examples show the `PEEK` command with some of its options and with the CPU option of the `TACL RUN` command. For examples that show the output of each PEEK option, see the subsection describing the option later in this section.

1. To run two PEEK samples on the default processor with a 1-second delay (the default value) between samples:

   ```
   > PEEK 2
   ```

2. To run two PEEK samples on processor 1 with a 10-second delay between samples:

   ```
   > PEEK / CPU 1 / 2, 10
   ```

3. To run two PEEK CME samples on processor 1 with a 10-second delay between samples:

   ```
   > PEEK / CPU 1 / CME, 2, 10
   ```

4. To display four `PAGING` reports about processor 1 with a 10-second delay, in order to compare paging statistics for different time periods:

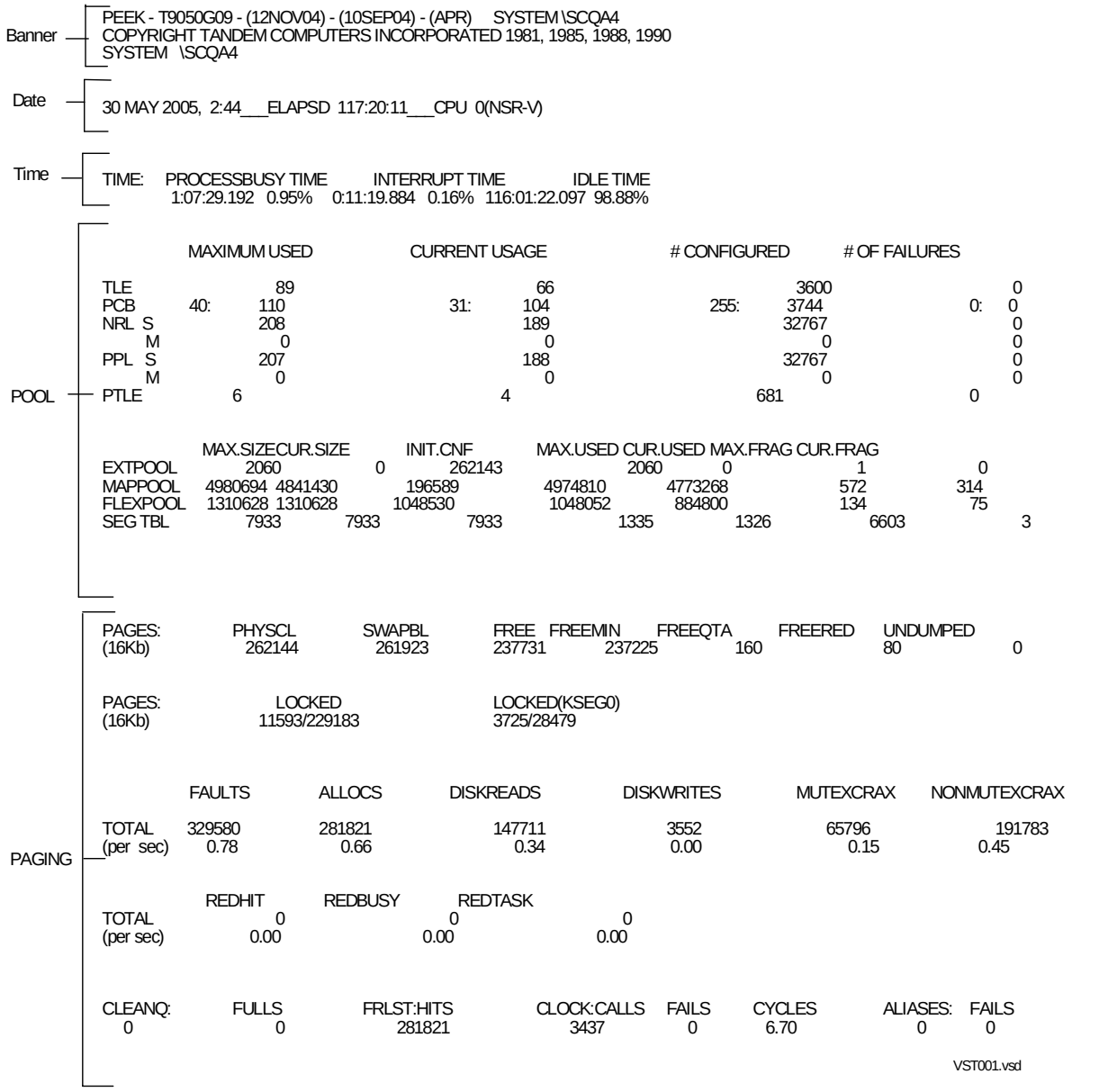
   ```
   > PEEK / CPU 1 / 4, 10, PAGING, DYNAMIC
   ```

PEEK Default Listing Format

When you specify PEEK with no options, a report is produced that is equivalent to the reports displayed by the TIME, POOL, and PAGING options.

Figure 2-1 on page 2-5 shows an example of the default PEEK display for systems running G-series RVUs.

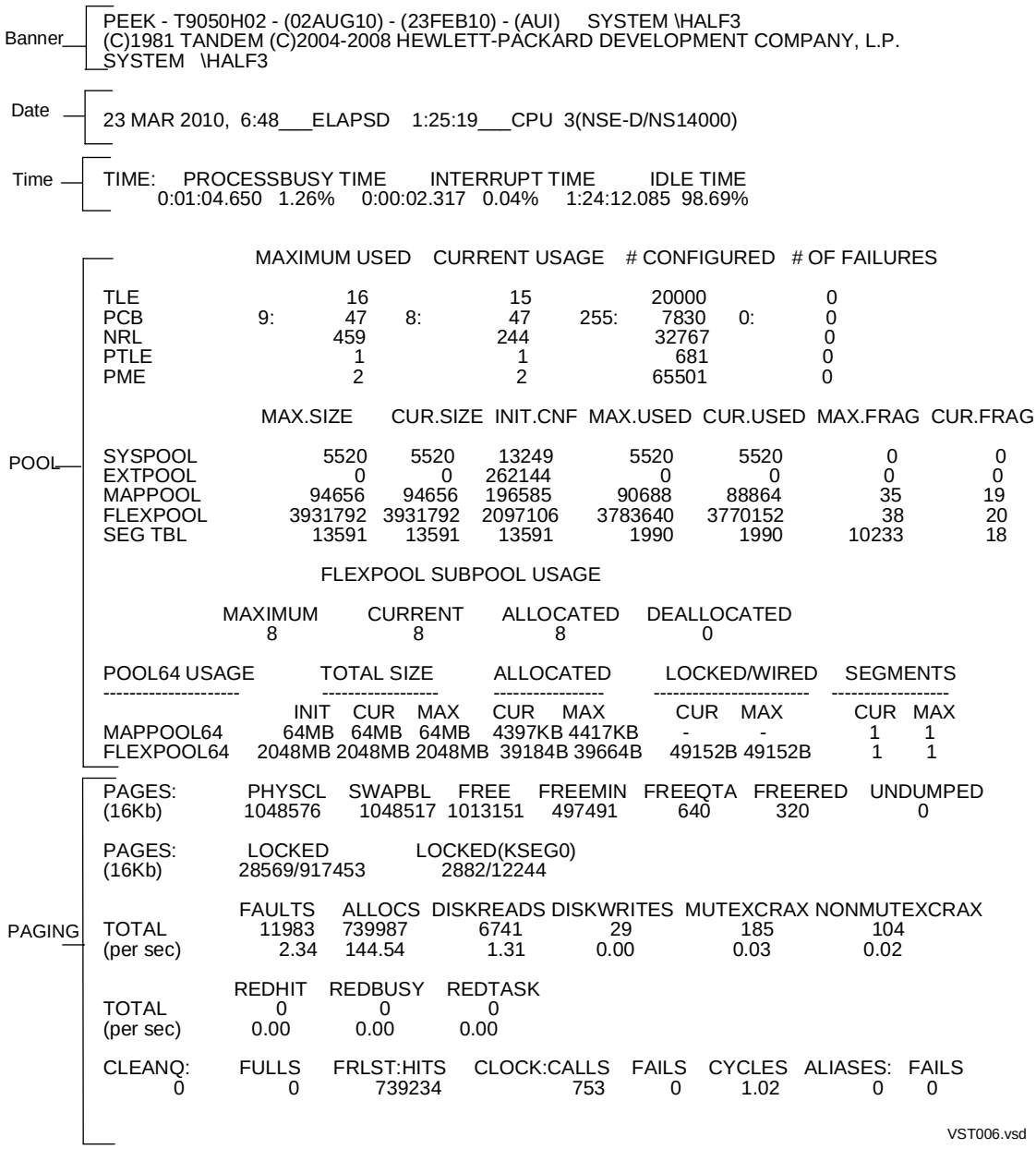
Figure 2-1. PEEK Default Listing (for G-series RVUs)



Note. SYSPOOL data is not included in the default PEEK display but is included in the PEEK ALL and PEEK POOL displays. For information about SYSPOOL statistics, see POOL Option on page 2-40.

Figure 2-2 shows an example of the PEEK default for H-series RVUs and J-series.

Figure 2-2. PEEK Default Listing (for J-series RVUs and H-series RVUs)



Note. The copyright information "**2004-2008**" and the **"Num IPUs**" option are displayed only in a J-series output. All other examples of PEEK output for H-series and J-series display the H-series output, similar to the J-series output, except for the copyright information and "**Num IPUs**" option displayed in this example.

Listing Headers

The PEEK default listing headers are explained in Table 2-1. The elements reported in the TIME, POOL, and PAGING headers are explained in detail later in this section.

Table 2-1. PEEK Default Listing Headers

Header	Explanation
Banner	PEEK product version, the system on which the report is being run, and copyright information.
Date	Date and time this display began.
	ELAPSD — Amount of time since the processor in this report was last loaded (in *hours:minutes:seconds*).
	CPU — Number and type of the processor that the statistical information describes (for example, *CPU 3 (NSR-T)*).
	SAMP, DELAY — Number of samples PEEK is to display and the period of delay between them. If you do not enter a value for the number of samples, PEEK reports one sample and does not display this element.
TIME	Information returned by the TIME option. See TIME Option on page 2-47.
POOL	Information returned by the POOL option. See POOL Option on page 2-40.
PAGING	Information returned by the PAGING option. See PAGING Option on page 2-35.

ALL Option

When you specify PEEK ALL on a system running G-series RVU, a report is produced that includes information about:

TIME
POOL
PAGING
MESSAGES
MQCINFO
INTERRUPTS
CME

Example

This example displays all PEEK statistics for processor 1:

```
> PEEK / CPU 1 / ALL
```

Figure 2-3 on page 2-8 shows the results of this command for an NSR-G processor.

In this figure, the elements in the listing are identified along the left margin. The elements Banner and Date are described earlier in this section. For information on the remaining elements, see TIME Option, POOL Option, PAGING Option, MESSAGES Option, MQCINFO Option, INTERRUPTS Option, and CME Option.

Figure 2-3. PEEK ALL Listing (for G-series RVUs) (page 1 of 2)

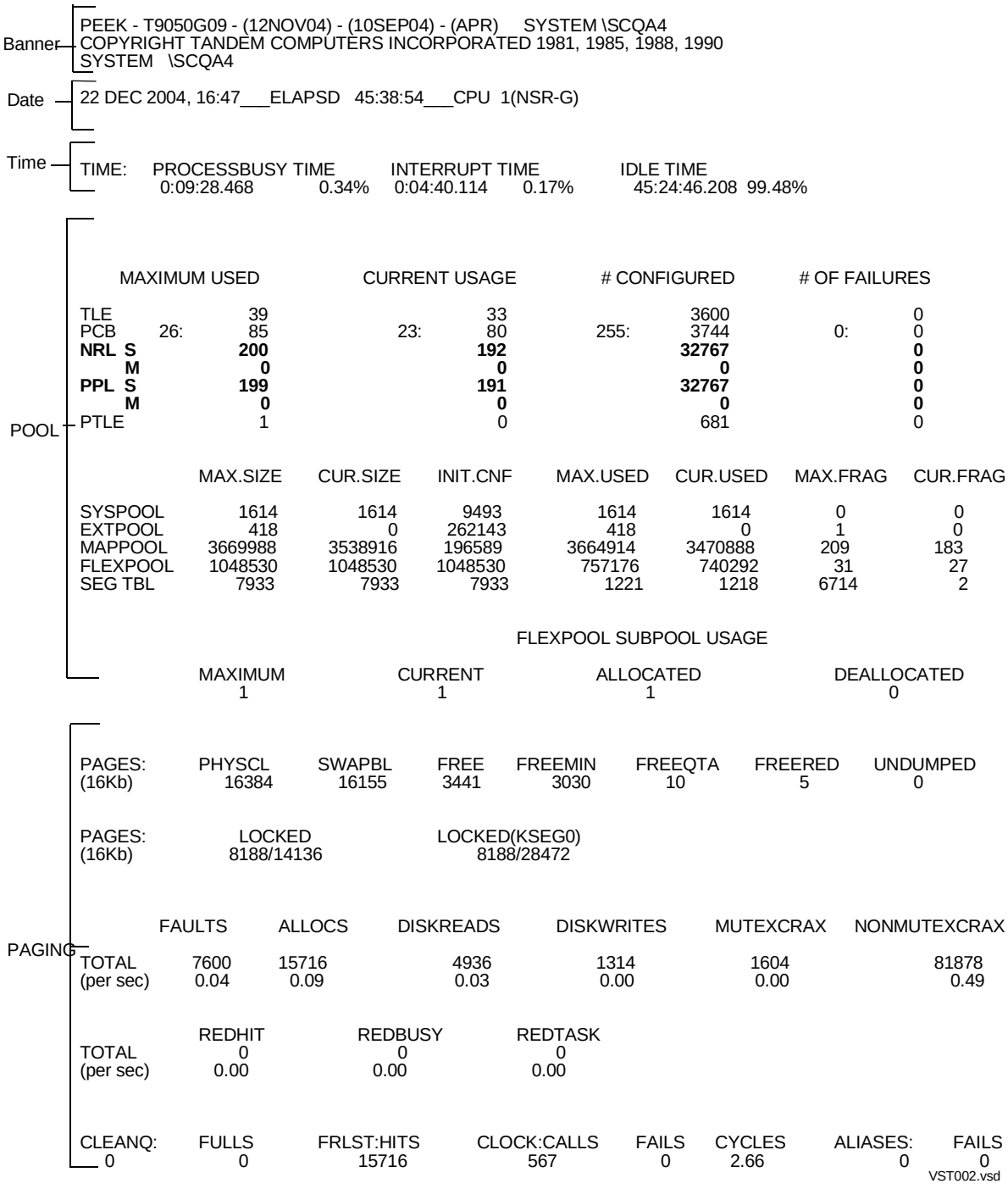
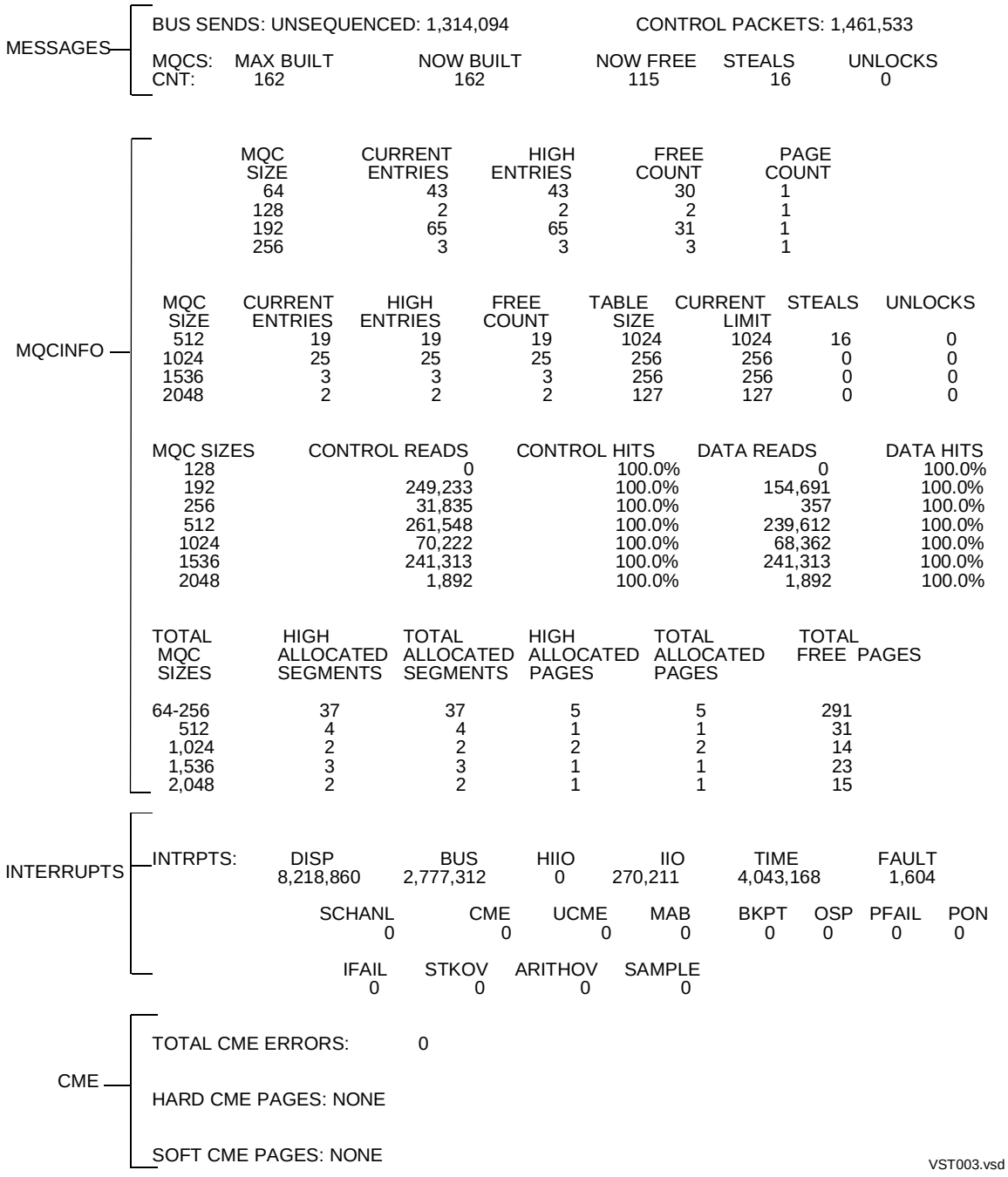


Figure 2-3. PEEK ALL Listing (for G-series RVUs) (page 2 of 2)



MESSAGES

BUS SENDS: UNSEQUENCED: 1,314,094 CONTROL PACKETS: 1,461,533

MQCS:	MAX BUILT	NOW BUILT	NOW FREE	STEALS	UNLOCKS
CNT:	162	162	115	16	0

MQCINFO

MQC SIZE	CURRENT ENTRIES	HIGH ENTRIES	FREE COUNT	PAGE COUNT
64	43	43	30	1
128	2	2	2	1
192	65	65	31	1
256	3	3	3	1

MQC SIZE	CURRENT ENTRIES	HIGH ENTRIES	FREE COUNT	TABLE SIZE	CURRENT LIMIT	STEALS	UNLOCKS
512	19	19	19	1024	1024	16	0
1024	25	25	25	256	256	0	0
1536	3	3	3	256	256	0	0
2048	2	2	2	127	127	0	0

MQC SIZES	CONTROL READS	CONTROL HITS	DATA READS	DATA HITS
128	0	100.0%	0	100.0%
192	249,233	100.0%	154,691	100.0%
256	31,835	100.0%	357	100.0%
512	261,548	100.0%	239,612	100.0%
1024	70,222	100.0%	68,362	100.0%
1536	241,313	100.0%	241,313	100.0%
2048	1,892	100.0%	1,892	100.0%

TOTAL MQC SIZES	HIGH ALLOCATED SEGMENTS	TOTAL ALLOCATED SEGMENTS	HIGH ALLOCATED PAGES	TOTAL ALLOCATED PAGES	TOTAL FREE PAGES
64-256	37	37	5	5	291
512	4	4	1	1	31
1,024	2	2	2	2	14
1,536	3	3	1	1	23
2,048	2	2	1	1	15

INTERRUPTS

INTRPTS:

DISP	BUS	HIIO	IIO	TIME	FAULT
8,218,860	2,777,312	0	270,211	4,043,168	1,604

SCHANL	CME	UCME	MAB	BKPT	OSP	PFAIL	PON
0	0	0	0	0	0	0	0

IFAIL	STKOV	ARITHOV	SAMPLE
0	0	0	0

CME

TOTAL CME ERRORS: 0

HARD CME PAGES: NONE

SOFT CME PAGES: NONE

VST003.vsd

When you specify PEEK ALL on a system running H-series RVU or J-series RVU, a report is produced that includes information about:

- TIME
- POOL
- PAGING
- MESSAGES

MQCINFO
DISPATCHES
INTERRUPTS
VRO
INAPPACC
NSAA
CME

Example

This example displays all PEEK statistics for processor 0:

```
> PEEK / CPU 0 / ALL
```

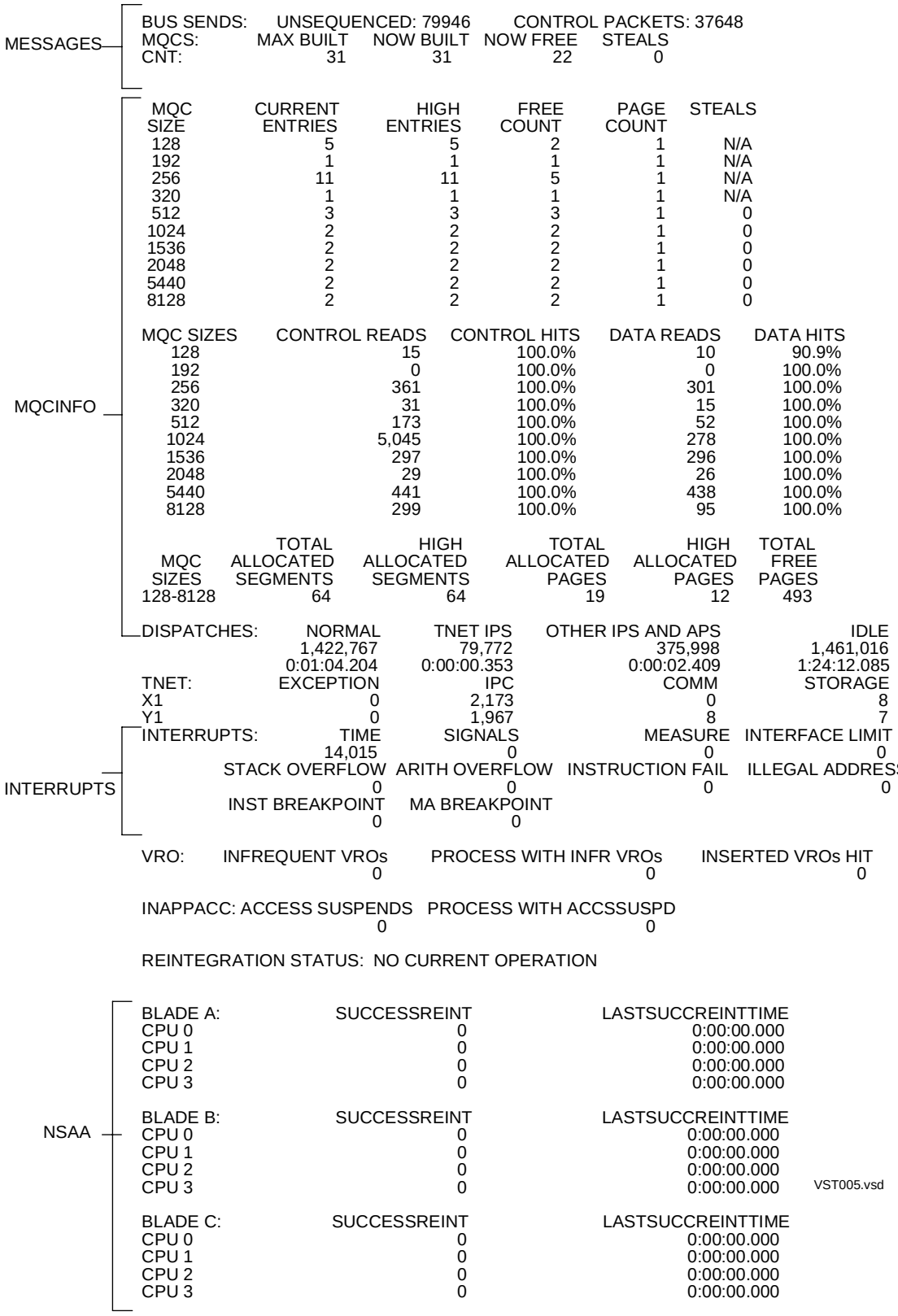
Figure 2-4 shows the results of this command for an NSE-D processor.

When you enter the command `PEEK /CPU 0/ ALL` on a system running H-series or J-series RVU, an output similar to this example is generated:

Figure 2-4. PEEK ALL Listing (for H-series RVUs and J-series RVUs) (page 1 of 3)

```
Banner  PEEK - T9050H02 - (02AUG10) - (23FEB10) - (AUI)     SYSTEM \HALF3
        (C)1981 TANDEM (C)2004-2008 HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.
        SYSTEM   \HALF3

Date    23 MAR 2010,  6:48___ELAPSD    1:25:19___CPU  3(NSE-D/NS14000)

Time    TIME:     PROCESSBUSY TIME       INTERRUPT TIME         IDLE TIME
              0:01:04.650   1.26%    0:00:02.317   0.04%    1:24:12.085  98.69%

                        MAXIMUM USED    CURRENT USAGE    # CONFIGURED   # OF FAILURES

        TLE                       16               15           20000              0
        PCB               9:      47       8:      47      255:  7830       0:     0
        NRL                      459              244           32767              0
        PTLE                       1                1             681              0
        PME                        2                2           65501              0

                         MAX.SIZE     CUR.SIZE  INIT.CNF  MAX.USED  CUR.USED  MAX.FRAG  CUR.FRAG

POOL    SYSPOOL              5520         5520     13249      5520      5520         0         0
        EXTPOOL                 0            0    262144         0         0         0         0
        MAPPOOL             94656        94656    196585     90688     88864        35        19
        FLEXPOOL          3931792      3931792   2097106   3783640   3770152        38        20
        SEG TBL             13591        13591     13591      1990      1990     10233        18

                                FLEXPOOL SUBPOOL USAGE

                        MAXIMUM       CURRENT      ALLOCATED     DEALLOCATED
                              8             8              8               0

        POOL64 USAGE          TOTAL SIZE          ALLOCATED        LOCKED/WIRED     SEGMENTS
        --------------------  ------------------  ---------------  ---------------  ----------
                               INIT   CUR   MAX    CUR     MAX      CUR     MAX      CUR  MAX
        MAPPOOL64              64MB  64MB  64MB   4397KB  4417KB   -       -        1    1
        FLEXPOOL64           2048MB 2048MB 2048MB 39184B  39664B   49152B  49152B   1    1

        PAGES:          PHYSCL    SWAPBL    FREE    FREEMIN  FREEQTA  FREERED   UNDUMPED
        (16Kb)         1048576   1048517 1013151    497491      640      320          0

        PAGES:          LOCKED          LOCKED(KSEG0)
        (16Kb)       28569/917453        2882/12244

                       FAULTS    ALLOCS  DISKREADS DISKWRITES  MUTEXCRAX NONMUTEXCRAX
PAGING  TOTAL          11983    739987       6741         29        185         104
        (per sec)       2.34    144.54       1.31       0.00       0.03        0.02

                      REDHIT   REDBUSY   REDTASK
        TOTAL              0         0         0
        (per sec)       0.00      0.00      0.00

        CLEANQ:      FULLS    FRLST:HITS    CLOCK:CALLS   FAILS   CYCLES   ALIASES:   FAILS
              0          0        739234            753       0     1.02          0       0

                                                                                VST004.vsd
```

Figure 2-4. PEEK ALL Listing (for H-series RVUs and J-series RVUs) (page 2 of 3)



MESSAGES

```
BUS SENDS:     UNSEQUENCED: 79946      CONTROL PACKETS: 37648
MQCS:      MAX BUILT   NOW BUILT  NOW FREE   STEALS
CNT:              31          31         22       0
```

MQCINFO

```
 MQC        CURRENT       HIGH       FREE       PAGE   STEALS
 SIZE       ENTRIES    ENTRIES      COUNT      COUNT
  128             5          5          2          1      N/A
  192             1          1          1          1      N/A
  256            11         11          5          1      N/A
  320             1          1          1          1      N/A
  512             3          3          3          1        0
 1024             2          2          2          1        0
 1536             2          2          2          1        0
 2048             2          2          2          1        0
 5440             2          2          2          1        0
 8128             2          2          2          1        0

MQC SIZES     CONTROL READS   CONTROL HITS   DATA READS   DATA HITS
      128                15         100.0%           10       90.9%
      192                 0         100.0%            0      100.0%
      256               361         100.0%          301      100.0%
      320                31         100.0%           15      100.0%
      512               173         100.0%           52      100.0%
     1024             5,045         100.0%          278      100.0%
     1536               297         100.0%          296      100.0%
     2048                29         100.0%           26      100.0%
     5440               441         100.0%          438      100.0%
     8128               299         100.0%           95      100.0%

              TOTAL       HIGH       TOTAL       HIGH      TOTAL
  MQC     ALLOCATED  ALLOCATED   ALLOCATED  ALLOCATED       FREE
 SIZES     SEGMENTS   SEGMENTS       PAGES      PAGES      PAGES
128-8128        64         64          19         12        493

DISPATCHES:      NORMAL     TNET IPS   OTHER IPS AND APS       IDLE
                1,422,767    79,772        375,998        1,461,016
                0:01:04.204  0:00:00.353   0:00:02.409    1:24:12.085
TNET:        EXCEPTION           IPC          COMM       STORAGE
X1                   0         2,173             0             8
Y1                   0         1,967             8             7
```

INTERRUPTS

```
INTERRUPTS:        TIME       SIGNALS       MEASURE  INTERFACE LIMIT
                 14,015             0             0                0
          STACK OVERFLOW  ARITH OVERFLOW  INSTRUCTION FAIL  ILLEGAL ADDRESS
                       0             0                 0                0
          INST BREAKPOINT  MA BREAKPOINT
                       0             0
```

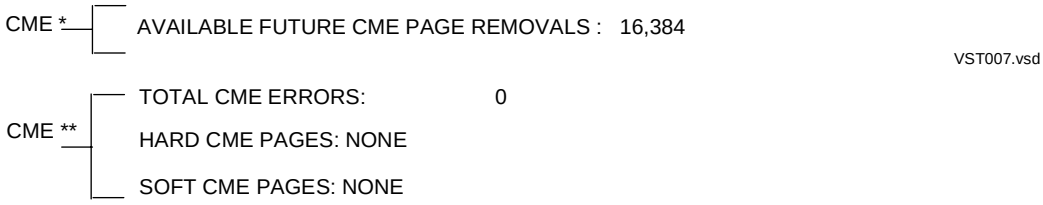
```
VRO:    INFREQUENT VROs      PROCESS WITH INFR VROs      INSERTED VROs HIT
                      0                           0                      0

INAPPACC: ACCESS SUSPENDS   PROCESS WITH ACCSSUSPD
                        0                        0

REINTEGRATION STATUS:  NO CURRENT OPERATION
```

NSAA

```
BLADE A:          SUCCESSREINT          LASTSUCCREINTTIME
CPU 0                        0              0:00:00.000
CPU 1                        0              0:00:00.000
CPU 2                        0              0:00:00.000
CPU 3                        0              0:00:00.000

BLADE B:          SUCCESSREINT          LASTSUCCREINTTIME
CPU 0                        0              0:00:00.000
CPU 1                        0              0:00:00.000
CPU 2                        0              0:00:00.000
CPU 3                        0              0:00:00.000

BLADE C:          SUCCESSREINT          LASTSUCCREINTTIME
CPU 0                        0              0:00:00.000
CPU 1                        0              0:00:00.000
CPU 2                        0              0:00:00.000
CPU 3                        0              0:00:00.000
```

VST005.vsd

Figure 2-4. PEEK ALL Listing (for H-series RVUs and J-series RVUs) (page 3 of 3)



* H06.16 and later H-series RVUs and J06.05 and later J-series RVUs
** Releases before H06.16 and J06.05 RVUs

Note.

- Even if <*value*> does reach zero, CMEs are always corrected.
- For a system running J06.06 or earlier J-series RVUs or H06.17 or earlier H-series RVUs, NRL is displayed as a four line output with the small-index (S) and medium-index (M) entries for NRL and PPL on separate lines. This is compressed to a single line NRL output from H06.18 series and J06.07 series onwards.
- For a system running J06.07 and later J-series RVUs or H06.18 and later H-series RVUs, the PME table information is included in the output.
- For a system running J06.09 and later J-series RVUs or H06.20 and later H-series RVUs, the FLEXPOOL64 information is also included in the output.

CME Option

This section describes the CME option for the following releases:

- CME Option - H06.16/J06.05 and later RVUs
- CME Option - For releases before H06.16/J06.05 RVU

CME Option - H06.16/J06.05 and later RVUs

The CME option displays relevant information related to correctable memory errors (CMEs) that have occurred since the processor was last loaded.

The occurrence of CMEs is considered normal with the high-density memory technology used in Integrity NonStop servers and in HP Neoview platforms. CMEs are handled automatically. No EMS event is logged in response to a CME.

When appropriate, a CME can result in removal of the enclosing memory page from further use. A limit is placed on the number of such page removals, based on a small percentage of total memory size.

When you enter the command PEEK /CPU 0/ CME , an output similar to this example is generated:

```
PEEK - T9050J02 - (01MAY05) - (31MAY05)     SYSTEM \HALF3
(C)1981 TANDEM (C)2004-2008 HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.

                            SYSTEM   \HALF3
8 JUN 2005 , 12:42___ELAPSD   16:17:03___CPU  0(NSE-P)___Num IPUs=2

AVAILABLE FUTURE CME PAGE REMOVALS:   <value>
```

Note. The "Num IPUs" option shown in the above example is visible in the ouput generated by a system running on J-series RVU only and not on a system running on H-series RVU.

The "NSE-P" value shown in the above example is specific to the particular Processor Type and CPU Model that PEEK is running on. For more information about the processor types used, see the *Guardian Procedure Calls Reference Manual*.

---

where,

<*value*> starts (upon reload of the processor) at the maximum number of allowed page removals for the processor and decreases as pages are removed from use.

A <*value*> of zero is the only condition for which a CME-related service action might be considered appropriate (although not required). It is not recommended to take special steps to monitor for this condition; however, if you observe that <*value* > has reached zero, contact the Global Mission Critical Solution Center (GMCSC) to determine whether a service action is appropriate.

The <*value*> is re-initialized to the maximum number of allowed page removals for the processor only upon reload of the processor. This behavior is independent of whether or not a CME-related service action has been performed.

CME Option - For releases before H06.16/J06.05 RVU

The CME option displays the number of correctable memory errors (CMEs) that have occurred since the processor was last loaded. The CME display reflects only initialized pages and can indicate chip or address-line failures. For definitions of the terms, see the Glossary.

When you enter PEEK CME, an output similar to this example is generated:

```
> PEEK CME
PEEK - T9050G09 - (05AUG02)     SYSTEM \VIOLET
COPYRIGHT TANDEM COMPUTERS INCORPORATED 1981, 1985, 1988, 1990

                            SYSTEM   \VIOLET
22 FEB 2004 , 10:52___ELAPSD  158:42:43___CPU  3(NSR-T)

TOTAL CME ERRORS:             0

HARD CME PAGES: NONE
SOFT CME PAGES: NONE
```

Elements of the CME Display

Table 2-2. CME Elements Reported in the CME Display

Element	Description
TOTAL CME ERRORS	Total number of correctable memory errors (CMEs)
HARD CME PAGES	Physical pages found containing hard CMEs
SOFT CME PAGES	Physical pages found containing soft CMEs

Considerations

- When a CME is detected during an access attempt to a specific memory location, the CME interrupt handler is invoked. The memory manager determines if the CME is a soft CME or a hard CME. In general, a soft CME is one that occurs on the first access to a specific memory location but does not occur on the second and subsequent accesses to the same location. A hard CME is one that continues to occur during consecutive access attempts. However, a soft CME can be reclassified as a hard CME if it occurs too frequently.
- The physical pages are in reverse order, so the newest page number appears first.
- The list of soft CME pages clears approximately every 14 days. The list of hard CME pages never clears.

Example

This example displays CME information for processor 2:

```
> PEEK / CPU 2 / CME
  PEEK - T9050G09 - (05AUG02)     SYSTEM \VIOLET
  COPYRIGHT TANDEM COMPUTERS INCORPORATED 1981, 1985, 1988, 1990

                                SYSTEM   \VIOLET
  22 FEB 2004, 14:44___ELAPSD  132:18:54___CPU  2(NSR-T)

  TOTAL CME ERRORS:             560

  HARD CME PAGES:
        7     6452     6451     6450     6449     6447     6446     6445
     6444     6442     6441     6440     6439     6437     6436     6435

  SOFT CME PAGES: NONE
```

DYNAMIC Option

Use the DYNAMIC option in combination with *samples*, *delay*, and other options to display and compare processor statistics during a number of time intervals that you specify.

Specify DYNAMIC instead of the INIT option when you want to monitor processor activity for a relatively short time period (15 minutes or less). DYNAMIC preserves the measured maximums that are listed in each PEEK report.

Example

To display four PAGING reports about processor 1 with a 10-second delay, in order to compare statistics at different times:

```
> PEEK / CPU 1 / PAGING, 4, 10, DYNAMIC
```

Report 1 displays information for the 161 hours since processor 1 was last loaded. Reports 2, 3, and 4 display only the activity that occurred during the 10-second intervals between reports. Specifically, the numbers appearing for FAULTS, CREATES, READS, WRITES, and CALLS change after Report 1.

```
PEEK - T9050G09 - (05AUG02)     SYSTEM \VIOLET
```

Report 1:

```
COPYRIGHT TANDEM COMPUTERS INCORPORATED 1981, 1985, 1988, 1990
                              SYSTEM   \VIOLET

22 FEB 2004 , 13:33___ELAPSD  161:24:17___CPU  1(NSR-T) __SAMP  1/4,DELAY  10

PAGES: PHYSCL  SWAPBL  FREE   LOCKED        FAULTS  CREATES   READS   WRITES
 (4Kb) 32767   32668  18151   4429/28585   1960066  1748555  1130222    8671
                                  (per sec)   3.37     3.00     1.94    0.01
PREPAGE:READS/USED   WRITES   CLOCK:CALLS  CYCLES(per sec) SCANS/CALL  FAILS
           0/      0      0     2158836     95.21( 0.000 )    1.44       0
```

Report 2:

```
                              SYSTEM   \VIOLET
22 FEB 2004 , 13:33___ELAPSD    0:00:10___CPU  1(NSR-T) __SAMP  2/4,DELAY  10

PAGES: PHYSCL  SWAPBL  FREE    LOCKED       FAULTS  CREATES    READS  WRITES
 (4Kb) 32767   32668  18151   4429/28585         2        0        2       0
                                  (per sec)   0.19     0.00     0.19    0.00
PREPAGE:READS/USED   WRITES   CLOCK:CALLS  CYCLES(per sec) SCANS/CALL  FAILS
           0/       0      0           2      0.00( 0.000 )    0.00       0
```

Report 3:

```
                              SYSTEM   \VIOLET
22 FEB 2004 , 13:33___ELAPSD    0:00:10___CPU  1(NSR-T) __SAMP  3/4,DELAY  10

PAGES: PHYSCL  SWAPBL  FREE    LOCKED       FAULTS  CREATES    READS  WRITES
 (4Kb) 32767   32668  18149   4428/28585         0        0        0       0
                                  (per sec)   0.00     0.00     0.00    0.00
PREPAGE:READS/USED   WRITES   CLOCK:CALLS  CYCLES(per sec) SCANS/CALL FAILS
           0/      0      0           0      0.00( 0.000 )    0.00      0
```

Report 4:

```
                              SYSTEM   \VIOLET
22 FEB 2004 , 13:34___ELAPSD    0:00:10___CPU  1(NSR-T) __SAMP  4/4,DELAY  10

PAGES: PHYSCL  SWAPBL  FREE    LOCKED       FAULTS  CREATES    READS  WRITES
 (4Kb) 32767   32668  18149   4426/28585         1        1        0       0
                                  (per sec)   0.10     0.10     0.00    0.00
PREPAGE:READS/USED   WRITES   CLOCK:CALLS  CYCLES(per sec) SCANS/CALL FAILS
           0/      0      0           3      0.00( 0.000 )    0.00      0
```

The numbers appearing under `CLOCK:CALLS` indicate that the algorithm for the page frame selection of the memory manager was invoked five times during the 30 seconds this report took to run.

HELP Option

The HELP option displays a syntax summary of all PEEK options. The G09 product version of the PEEK HELP appears as:

```
> PEEK HELP
PEEK/CPU n/ [PAging]    - displays paging statistics (default)
            [POols]     - displays system pool statistics (default)
            [INIT]      - init pools to current value/zero (super grp)
            [MESsages]  - displays message sending statistics
            [MQCinfo]   - shows MQC utilization stats
            [INTerrupts]- displays interrupt routine statistics
            [TIME]      - shows cpu utilization statistics ONLY
            [CME]       - displays CME statistics ONLY
            [ALL]       - PAGING, POOLS, IO, MES, EXP, INT, TIME, and CME
            [<samples>] - number of time to display data
            [<delay>]   - interval in seconds between samples
            [Dynamic]   - shows current activity in each display
```

INIT Option (Super Group Only)

Use the INIT option to reset the maximums of certain POOL elements to equal the values in the CURRENT columns in the PEEK POOL report.

These POOL elements are reset:

- Time-list elements (TLEs)
- Process control blocks (PCBs)
- Process time-list elements (PTLEs)
- POSIX mapping entry elements (PMEs)
- EXTPOOL entries
- SYSPOOL entries
- MAPPOOL entries

Consideration

Use INIT only when you want to initialize (and thus destroy the past history of) pool-related maximums. Specify the DYNAMIC option instead of INIT when you want to monitor processor activity for a relatively short time period (15 minutes or less). For example, if you run PEEK daily and want a record of each day's activity only, run PEEK when activity is low and specify INIT. Later, run PEEK and specify DYNAMIC to preserve the daily maximums. This technique can help you monitor processor activity where the level of use fluctuates greatly.

If you use DYNAMIC with INIT, the first report displays the reset values. Subsequent reports display activity since the values were reset.

Example

This example resets POOL maximums for processor 3:

```
> PEEK / CPU 3 / INIT
PEEK - T9050H02 - (02AUG10) - (23FEB10) - (AUI)     SYSTEM \TAHOE
(C)1981 TANDEM (C)2004-2008 HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.
                              SYSTEM   \TAHOE

23 MAR 2010,  6:48___ELAPSD    1:25:19___CPU  3(NSE-D/NS14000)

TIME:     PROCESSBUSY TIME         INTERRUPT TIME          IDLE TIME
          0:01:04.650   1.26%     0:00:02.317   0.04%     1:24:12.085  98.69%

          MAXIMUM USED  CURRENT USAGE  # CONFIGURED  # OF FAILURES
TLE             16             15          20000             0
PCB      9:     47      8:     47    255:   7830      0:      0
NRL            459            244          32767             0
PTLE             1              1            681             0
PME              2              2          65501             0

            MAX.SIZE  CUR.SIZE  INIT.CNF  MAX.USED  CUR.USED MAX.FRAG CUR.FRAG
SYSPOOL         5520      5520     13249      5520      5520        0        0
EXTPOOL            0         0    262144         0         0        0        0
MAPPOOL        94656     94656    196585     90688     88864       35       19
FLEXPOOL     3931792   3931792   2097106   3783640   3770152       38       20
SEG TBL        13591     13591     13591      1990      1990    10233       18

                         FLEXPOOL SUBPOOL USAGE

         MAXIMUM          CURRENT       ALLOCATED     DEALLOCATED
            8                8              8              0

POOL64 USAGE          TOTAL SIZE          ALLOCATED   LOCKED/WIRED    SEGMENTS
------------     ------------------     ----------   ------------    --------
                 INIT    CUR    MAX     CUR    MAX     CUR    MAX    CUR  MAX
MAPPOOL64        64MB   64MB   64MB  4397KB 4417KB       -      -      1    1
FLEXPOOL64     2048MB 2048MB 2048MB  39184B 39664B  49152B 49152B      1    1

PAGES:  PHYSCL  SWAPBL   FREE   FREEMIN FREEQTA FREERED  UNDUMPED
(16Kb) 1048576 1048517 1013151  497491     640     320         0

PAGES:      LOCKED       LOCKED(KSEG0)
(16Kb)    28569/917453    2882/12244
```

In this example, the maximum numbers in these columns match the numbers in their corresponding CURRENT columns because they have been reset through `INIT`:

- MAXIMUM USED and CURRENT USAGE (for TLE, PTLE, PCB, and PME entries)
- MAX.SIZE and CUR.SIZE (for EXTPOOL and MAPPOOL entries)
- MAX.USED and CUR.USED (for EXTPOOL and MAPPOOL entries)
- MAX.FRAG and CUR.FRAG (for EXTPOOL and MAPPOOL entries)

The # OF FAILURES values for the POOL elements TLE, PCB, PTLE, and PME are reset to 0 when you specify `INIT`.

When you enter the command `PEEK /CPU 0/ INIT` on a system running H06.20 and later H-series RVUs or J06.09 or later J-series RVUs, an output similar to this example is generated:

```
PEEK - T9050H02 - (02AUG10) - (23FEB10) - (AUI)     SYSTEM \HALF3

(C)1981 TANDEM (C)2004-2008 HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.
                            SYSTEM   \HALF3


23 MAR 2010,  6:48___ELAPSD    1:25:19___CPU  3(NSE-D/NS14000)


TIME:     PROCESSBUSY TIME         INTERRUPT TIME          IDLE TIME
          0:01:04.650   1.26%      0:00:02.317   0.04%     1:24:12.085  98.69%


          MAXIMUM USED  CURRENT USAGE  # CONFIGURED  # OF FAILURES
TLE             16             15          20000             0
PCB      9:     47      8:     47    255:   7830      0:      0
NRL            459            244          32767             0
PTLE             1              1            681             0
PME              2              2          65501             0


           MAX.SIZE  CUR.SIZE  INIT.CNF  MAX.USED  CUR.USED MAX.FRAG CUR.FRAG
SYSPOOL        5520      5520     13249      5520      5520        0        0
EXTPOOL           0         0    262144         0         0        0        0
MAPPOOL       94656     94656    196585     90688     88864       35       19
FLEXPOOL    3931792   3931792   2097106   3783640   3770152       38       20
SEG TBL       13591     13591     13591      1990      1990    10233       18


                        FLEXPOOL SUBPOOL USAGE
         MAXIMUM          CURRENT       ALLOCATED     DEALLOCATED
            8                8              8              0

POOL64 USAGE          TOTAL SIZE           ALLOCATED   LOCKED/WIRED    SEGMENTS
------------     ------------------     ----------   ------------    --------
                 INIT    CUR    MAX     CUR    MAX     CUR    MAX     CUR  MAX
MAPPOOL64         64MB   64MB   64MB  4397KB 4417KB       -      -      1    1
FLEXPOOL64      2048MB 2048MB 2048MB  39184B 39664B  49152B 49152B      1    1


PAGES:  PHYSCL  SWAPBL   FREE   FREEMIN FREEQTA FREERED  UNDUMPED
(16Kb) 1048576 1048517 1013151  497491     640     320         0


PAGES:      LOCKED       LOCKED(KSEG0)
(16Kb)    28569/917453    2882/12244


              FAULTS     ALLOCS  DISKREADS DISKWRITES  MUTEXCRAX NONMUTEXCRAX
TOTAL          11983     739987       6741         29        185          104
(per sec)       2.34     144.54       1.31       0.00       0.03         0.02


              REDHIT    REDBUSY    REDTASK
TOTAL              0          0          0
(per sec)       0.00       0.00       0.00
```

```
CLEANQ:    FULLS  FRLST:HITS  CLOCK:CALLS    FAILS    CYCLES   ALIASES:  FAILS
     0        0      739234          753        0      1.02          0       0
```

Note. For a system running J06.06 or earlier J-series RVUs and H06.17 or earlier H-series RVUs, NRL is displayed as a four-line output with the small-index (S) and medium-index (M) entries for NRL and PPL on separate lines. This four-line output is compressed to a single line NRL output from J06.07/H06.18 series onwards.

For a system running J06.07 and later J-series RVUs or H06.18 and later H-series RVUs, the PME table information is included in the output.

For a system running J06.09 and later J-series RVUs or H06.20 and later H-series RVUs, the FLEXPOOL64 information is also included in the output.

INTERRUPTS Option

Use the INTERRUPTS option to display counters for interrupt process (IP) and auxillary process (AP) events. For H-series RVUs and J-series RVUs, interrupt handlers are replaced by interrupt processes. Interrupt handlers have one or more corresponding interrupt process. Some interrupt handlers are combined into a single-process general fault handler (GFH), which handles several kinds of events such as traps, process timer timeouts, and pending ownership events. Dispatcher is the only interrupt handler that does not have a corresponding interrupt process.

When you enter PEEK INTERRUPTS, an output similar to this example is generated:

```
> PEEK INTERRUPTS
PEEK - T9050G09 - (05AUG02)     SYSTEM \VIOLET
COPYRIGHT TANDEM COMPUTERS INCORPORATED 1981, 1985, 1988, 1990

                            SYSTEM   \VIOLET
22 FEB 2004 , 13:48__ELAPSD 161:38:32__CPU  2(NSR-T)
INTRPTS:   DISP          BUS   HIIO          IIO         TIME       FAULT
     41,180,423   46,599,852      0            0    3,982,934   2,284,217
         SCHANL      CME    UCME     MAB    BKPT     OSP   PFAIL    PON
              0        0       0       3   5,589       0       0      0
          IFAIL   STKOV ARITHOV  SAMPLE
            100       1      11       0
```

When you enter the command PEEK /CPU 0/ INT on a system running H-series or J-series RVU, an output similar to this example is generated:

```
PEEK - T9050H02 - (01MAY05) - (31MAY05)     SYSTEM \HALF3
(C)1981 TANDEM (C)2004 HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.

                              SYSTEM   \HALF3
8 JUN 2005 , 12:40___ELAPSD   16:15:26___CPU  0(NSE-P)

DISPATCHES:  NORMAL           TNET IPS  OTHER IPS AND APS              IDLE
          9,453,491          1,214,199          8,193,798          8,419,903
        0:09:57.577        0:00:09.142        0:00:30.196       16:04:49.576
TNET:     EXCEPTION               IPC               COMM            STORAGE
X1                0               265              3,191                 15
Y1                0                 0                  0                  0
INTERRUPTS:    TIME            SIGNALS            MEASURE    INTERFACE LIMIT
             21,952                  0                  0                  0
     STACK OVERFLOW     ARITH OVERFLOW   INSTRUCTION FAIL    ILLEGAL ADDRESS
                  0                  0                  0                  0
    INST BREAKPOINT      MA BREAKPOINT
                  0                  0
```

Elements of the INTERRUPTS Display

Table 2-3. INTERRUPTS Elements Reported in the INTERRUPTS Display (page 1 of 2)

Element	Description[1]
INTRPTS	Header identifying the INTERRUPTS display.
DISP	Number of dispatcher interrupts.
BUS	Number of interprocessor communication (IPC) interrupts.
HIIO	Number of high-priority I/O interrupts. For systems running G-series RVUs, it does not apply and always returns 0.
IIO	Number of I/O traffic interrupts.
TIME	Number of time-list interrupts.
FAULT	Number of page-fault interrupts.
SCHANL	Number of special channel error interrupts. For systems running G-series RVUs, it does not apply and always returns 0.
CME	Number of correctable memory error interrupts.
UCME	Number of uncorrectable memory error interrupts.
MAB	Number of memory access breakpoint interrupts.
BKPT	Number of instruction breakpoint interrupts.
OSP	Number of Remote Maintenance Interface (RMI) interrupts. The RMI performs some of the functions that on earlier systems were performed by the Operations and Service Processor (OSP). For systems running G-series RVUs, this element does not apply and always returns 0.
PFAIL	Number of power-fail interrupts.
PON	Number of power-on interrupts.
IFAIL	Number of instruction failure traps.
STKOV	Number of stack overflow traps.
ARITHOV	Number of arithmetic overflow traps.
SAMPLE	Number of Measure sampler interrupts.
DISPATCHES	Header identifying the DISPATCHES display. (Note: The elements mentioned under the DISPATCHES display are applicable for H-series RVUs and J-series RVUs only.)
NORMAL	Number of dispatches for processes other than the idle process, TNet IPs, and other Interrupt processes.
TNET IPS	Number of dispatches for TNet Services IPs.

[1] For definitions of the different types of interrupts and traps reported by the INTERRUPTS option, see the Glossary.

Table 2-3. INTERRUPTS Elements Reported in the INTERRUPTS Display (page 2 of 2)

Element	Description[1]
OTHER IPS AND APS	Number of dispatches for IPs and APs such as General Fault Handler (GFH), TimeList IP, Send Queued Messages AP, Signal Trap Monitor AP, and XRAY IP that do what was considered an Interrupt Handler before the H-series RVUs. CPU Busy Time for IPS and APs such as General Fault Handler (GFH), TimeList IP, Send Queued Messages AP, Signal Trap Monitor AP, and XRAY IP, which perform the functions that an Interrupt Handler performed before the H-series RVUs.
IDLE	Number of dispatches and CPU Busy Time for idle process.
TNET	Header identifying interrupts for TNet Services IPs.
EXCEPTION	Number of TNet Services Error interrupts such as AVT errors, link errors, TNet stack errors, errors from other processes such as the message system, storage, and so on.
IPC	Number of message interprocessor communication (IPC) interrupts.
COMM	Number of COMM I/O interrupts.
STORAGE	Number of STORAGE I/O interrupts.
X1	Number of interrupts on TNet Services X-fabric.
Y1	Number of interrupts on TNet Services Y-fabric.
INTERRUPTS	Header identifying other interrupts, traps, and events. (Note: The elements mentioned under the INTERRUPTS display are applicable only for H-series RVUs and J-series RVUs.)
TIME	Number of time-list interrupts.
SIGNALS	Number of entries to the signal or trap monitor.
MEASURE	Number of measure *processh* events.
INTERFACE LIMIT	Number of interface limit-exceeded traps.
STACK OVERFLOW	Number of stack overflow traps.
ARITH OVERFLOW	Number of arithmetic overflow traps.
INTERUCTION FAIL	Number of instruction failure traps.
ILLEGAL ADDRESS	Number of illegal-address reference traps.
INST BREAKPOINT	Number of instruction breakpoint interrupts.
MAB BREAKPOINT	Number of memory-access breakpoint interrupts.

[1] For definitions of the different types of interrupts and traps reported by the INTERRUPTS option, see the Glossary.

Example

This example displays interrupt conditions for processor 3:

```
> PEEK / CPU 3 / INTERRUPTS
PEEK - T9050G09 - (05AUG02)     SYSTEM \VIOLET
COPYRIGHT TANDEM COMPUTERS INCORPORATED 1981, 1985, 1988, 1990

                              SYSTEM   \VIOLET
22 FEB 2004 , 13:53___ELAPSD  161:43:49___CPU  3(NSR-T)

INTRPTS:       DISP          BUS   HIIO          IIO        TIME       FAULT
         96,204,093   42,113,197      0    8,250,591    3,419,017         304
             SCHANL     CME    UCME     MAB    BKPT     OSP   PFAIL    PON
                  0       0       0      49  10,964       0       0      0
               IFAIL   STKOV ARITHOV  SAMPLE
                 148       4      14       0
```

MESSAGES Option

The MESSAGES option displays the number of unsequenced packets, control packets, and data messages the processor sends. It also displays statistical data about the processor's message quick cells (MQCs).

When you enter PEEK MESSAGES, an output similar to this example is generated:

```
> PEEK / CPU 1 / MESSAGES
PEEK - T9050G09 - (05AUG02)     SYSTEM \VORTEX
COPYRIGHT TANDEM COMPUTERS INCORPORATED 1981, 1985, 1988, 1990

                              SYSTEM   \VORTEX
23 FEB 2004, 14:45___ELAPSD 3005:28:56___CPU  1(NSR-T)

BUS SENDS: UNSEQUENCED: 729,713,708  CONTROL PACKETS: 674,349,063
MQCS:    MAX BUILT  NOW BUILT   NOW FREE    STEALS    UNLOCKS
CNT:           887        719        197    43,202        116
```

To display additional statistics about MQCs, use the MQCINFO Option.

Elements of the MESSAGES Display

Table 2-4. MESSAGES Elements Reported in the MESSAGES Display

Element	Description
BUS SENDS	Number of these SEND instructions that have been run: • UNSEQUENCED — number of 1-packet messages such as “I'm alive” and message acknowledgments • CONTROL PACKETS — number of sequenced transmissions performed
MQCS	Number of message quick cells in these states: • MAX BUILT — maximum number of MQCs that have been allocated • NOW BUILT — current number of MQCs allocated • NOW FREE — number of MQCs currently free • STEALS — number of replacements performed in all MQCs • UNLOCKS — number of deallocations performed in all MQCs
CNT	Number of MQCs in a specified state (MAX BUILT, NOW BUILT, NOW FREE, STEALS, and UNLOCKS)

Example

This example displays message information for processor 2 since it was last loaded:

```
> PEEK / CPU 2 / MESSAGES
PEEK - T9050G09 - (05AUG02)     SYSTEM \TAHOE
COPYRIGHT TANDEM COMPUTERS INCORPORATED 1981, 1985, 1988, 1990

                               SYSTEM   \TAHOE
23 FEB 2004, 12:27___ELAPSD   24:45:54___CPU  2(NSR-T)

BUS SENDS: UNSEQUENCED: 767,746       CONTROL PACKETS: 176,176
MQCS:   MAX BUILT    NOW BUILT     NOW FREE       STEALS      UNLOCKS
CNT:           60           41           40            0           19
```

When you enter the command PEEK /CPU 0/ MESSAGES on a system running H-series or J-series RVU, an output similar to this example is generated:

```
PEEK - T9050H02 - (01MAY05) - (31MAY05)     SYSTEM \HALF3
(C)1981 TANDEM (C)2004 HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.
                               SYSTEM   \HALF3
8 JUN 2005 , 12:38___ELAPSD   16:13:29___CPU  0(NSE-P)
BUS SENDS: UNSEQUENCED: 96,353        CONTROL PACKETS: 0
MQCS:    MAX BUILT  NOW BUILT   NOW FREE     STEALS    UNLOCKS
CNT:            64         64         19          0          0
```

MQCINFO Option

The MQCINFO option displays information about message quick cell (MQC) resources.

Note. The MCQINFO option replaces the EXPEDITED option, which in earlier RVUs such as Dnn.xx displays information about expedited request transmissions.

Definition of MQCs

Message quick cells (MQCs) are data structures that contain information about messages being sent to or received from another process. The message system builds MQCs as needed for incoming and outgoing messages, system status messages, and timer expirations relating to time-list elements (TLEs).

Using SYSGEN parameters, the number of MQCs cannot be modified.

Using the MQCINFO Option

When you enter the command PEEK MQCINFO, an output similar to this example is generated:

```
> PEEK MQCINFO
PEEK - T9050G09 - (12NOV04) - (10SEP04) - (APR)     SYSTEM \SCQA4
COPYRIGHT TANDEM COMPUTERS INCORPORATED 1981, 1985, 1988, 1990
                              SYSTEM   \SCQA4
22 DEC 2004, 16:39___ELAPSD   45:30:46___CPU  1(NSR-G)
  MQC   CURRENT      HIGH      FREE      PAGE
 SIZE   ENTRIES   ENTRIES     COUNT     COUNT
   64        43        43        29         1
  128         2         2         2         1
  192        65        65        29         1
  256         3         3         3         1
  MQC   CURRENT      HIGH      FREE     TABLE   CURRENT    STEALS   UNLOCKS
 SIZE   ENTRIES   ENTRIES     COUNT      SIZE     LIMIT
  512        19        19        19      1024      1024        16         0
 1024        25        25        25       256       256         0         0
 1536         3         3         3       256       256         0         0
 2048         2         2         2       127       127         0         0
MQC SIZES       CONTROL READS   CONTROL HITS    DATA READS      DATA HITS
      128                   0         100.0%             0         100.0%
      192             248,459         100.0%       154,187         100.0%
      256              31,745         100.0%           357         100.0%
      512             260,949         100.0%       239,015         100.0%
     1024              70,032         100.0%        68,176         100.0%
     1536             240,802         100.0%       240,802         100.0%
     2048               1,876         100.0%         1,876         100.0%
            TOTAL       HIGH      TOTAL       HIGH     TOTAL
   MQC  ALLOCATED  ALLOCATED  ALLOCATED  ALLOCATED      FREE
 SIZES   SEGMENTS   SEGMENTS      PAGES      PAGES     PAGES
64-256         37         37          5          5       291
   512          4          4          1          1        31
 1,024          2          2          2          2        14
 1,536          3          3          1          1        23
 2,048          2          2          1          1        15
```

When you enter the command `PEEK /CPU 0/ MQCINFO` on a system running H-series or J-series RVU, an output similar to this example is generated:

```
PEEK - T9050H02 - (01MAY05) - (31MAY05)     SYSTEM \HALF3
(C)1981 TANDEM (C)2004 HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.
                              SYSTEM   \HALF3
8 JUN 2005 , 12:39___ELAPSD   16:14:37___CPU  0(NSE-P)
  MQC    CURRENT      HIGH      FREE      PAGE
 SIZE    ENTRIES    ENTRIES     COUNT     COUNT
   64         28         28         6         1
  128          1          1         1         1
  192         34         34        11         1
  256          1          1         1         1

  MQC    CURRENT      HIGH      FREE     TABLE    CURRENT    STEALS    UNLOCKS
 SIZE    ENTRIES    ENTRIES     COUNT      SIZE      LIMIT
  512          0          0         0      1024       1024         0          0
 1024          0          0         0       256        256         0          0
 1536          0          0         0       256        256         0          0
 2048          0          0         0       127        127         0          0

MQC SIZES       CONTROL READS   CONTROL HITS    DATA READS      DATA HITS
      128                   0         100.0%             0         100.0%
      192                   0         100.0%             0         100.0%
      256                   0         100.0%             0         100.0%
      512                   0         100.0%             0         100.0%
     1024                   0         100.0%             0         100.0%
     1536                   0         100.0%             0         100.0%
     2048                   0         100.0%             0         100.0%

            TOTAL       HIGH      TOTAL       HIGH     TOTAL
   MQC   ALLOCATED  ALLOCATED  ALLOCATED  ALLOCATED      FREE
 SIZES    SEGMENTS   SEGMENTS      PAGES      PAGES     PAGES
64-256         37         37          5          5       291
   512          4          4          0          0        32
 1,024          2          2          0          0        16
 1,536          3          3          0          0        24
 2,048          2          2          1          1        15
```

Elements of the MQCINFO Display

Table 2-5. MQCINFO Elements Reported in the MQCINFO Display

Element	Description
MQC SIZE	Size of each MQC in bytes.
CURRENT ENTRIES	Number of MQC entries currently allocated and in use.
HIGH ENTRIES	Highest number of MQC entries ever allocated.
FREE COUNT	Number of MQCs currently allocated but not in use.
PAGE COUNT	Number of physical memory pages used for each MQC.
TABLE SIZE	Maximum number of MQC entries of a given size that the system can ever contain.
CURRENT LIMIT	Maximum number of entries currently allowed. If CURRENT ENTRIES exceeds CURRENT LIMIT, the system attempts to reduce CURRENT ENTRIES down to CURRENT LIMIT.
STEALS	Number of replacements performed.
UNLOCKS	Number of entries that are no longer allocated.
MQC SIZES	Size of each MQC that is large enough to store a message in cache (in bytes).
CONTROL READS	Number of times that control information has been read from the MQCs by users of the message system.
CONTROL HITS	Percentage of the reads of control information that were satisfied by information still in the MQC cache. These numbers should remain at or near 100 percent most of the time.
DATA READS	Number of times data has been read from the MQCs.
DATA HITS	Percentage of the data reads that were satisfied by information in the MQC cache. These numbers should remain at or near 100 percent most of the time.
TOTAL ALLOCATED SEGMENTS	Total number of segments (8 pages of 16 Kilobytes each) allocated for the MQCs.
HIGH ALLOCATED SEGMENTS	Highest number of segments ever allocated for the MQCs of a particular size.
TOTAL ALLOCATED PAGES	Total number of pages allocated for the MQCs of a particular size.
HIGH ALLOCATEDPAGES	Highest number of pages ever allocated for the MQCs of a particular size.
TOTAL FREE PAGES	Total number of free pages available for allocating MQCs of a particular size.

Example

This example displays information about CPU 0, the MQCs from CPU 0, since CPU 0 was last loaded, and for the 10-second period since Report 1:

```
> PEEK / CPU 0 / MQCINFO, DYNAMIC, 2, 10
```

Report 1:

```
PEEK - T9050G09 - (05AUG02)     SYSTEM \TAHOE
COPYRIGHT TANDEM COMPUTERS INCORPORATED 1981, 1985, 1988, 1990

                              SYSTEM   \TAHOE
22 FEB 2004 ,  8:35___ELAPSD  132:26:09___CPU  2(NSR-T) __SAMP  1/2,DELAY  10

  MQC   CURRENT      HIGH      FREE      PAGE
 SIZE   ENTRIES   ENTRIES     COUNT     COUNT
   32        19        19        16         1
   64        48        48        22         1
   96       101       101        48         3
  160         7         7         6         1

  MQC   CURRENT      HIGH      FREE     TABLE   CURRENT    STEALS    UNLOCKS
 SIZE   ENTRIES   ENTRIES     COUNT      SIZE     LIMIT
  512        29        29        26      1024      1024       1293          0
 1024         4         4         4       256       256          0          0
 1536         2         2         2       256       256          0          0
 2048         2         2         2       123       123          0          0

  MQC   CONTROL          CONTROL           DATA READS     DATA
 SIZES  READS             HITS               READS       HITS
   96   294,739           100.0%               4        100.0%
  160   211,728           100.0%          26,148        100.0%
  512   1,848,270         100.0%       1,551,221        100.0%
 1024   443,946           100.0%          27,363        100.0%
 1536   2,086             100.0%           2,063        100.0%
 2048   576               100.0%             576        100.0%

 TOTAL       HIGH      TOTAL       HIGH     TOTAL

   MQC  ALLOCATED  ALLOCATED  ALLOCATED  ALLOCATED      FREE

 SIZES   SEGMENTS   SEGMENTS      PAGES      PAGES     PAGES

64-256         37         37          5          5       291

   512          4          4          1          1        31

 1,024          2          2          2          2        14

 1,536          3          3          1          1        23

 2,048          2          2          1          1        15
```

Report 2:

```
                                SYSTEM   \SCQA4
22 DEC 2004, 17:56___ELAPSD    0:00:10___CPU  1(NSR-G)  __SAMP  2/2,DELAY  10

  MQC   CURRENT      HIGH      FREE      PAGE
 SIZE   ENTRIES   ENTRIES     COUNT     COUNT
   64        43        43        30         1
  128         2         2         2         1
  192        65        65        33         1
  256         3         3         3         1

  MQC   CURRENT      HIGH      FREE     TABLE   CURRENT    STEALS   UNLOCKS
 SIZE   ENTRIES   ENTRIES     COUNT      SIZE     LIMIT
  512        19        19        19      1024      1024         0         0
 1024        25        25        25       256       256         0         0
 1536         3         3         3       256       256         0         0
 2048         2         2         2       127       127         0         0

MQC SIZES       CONTROL READS   CONTROL HITS    DATA READS      DATA HITS
      128                   0         100.0%             0         100.0%
      192                  18         100.0%            12         100.0%
      256                   2         100.0%             0         100.0%
      512                  14         100.0%            14         100.0%
     1024                   3         100.0%             3         100.0%
     1536                 253         100.0%           253         100.0%
     2048                   0         100.0%             0         100.0%

            TOTAL       HIGH      TOTAL       HIGH     TOTAL
   MQC  ALLOCATED  ALLOCATED  ALLOCATED  ALLOCATED      FREE
 SIZES   SEGMENTS   SEGMENTS      PAGES      PAGES     PAGES
64-256         37         37          5          5       291
   512          4          4          1          1        31
 1,024          2          2          2          2        14
 1,536          3          3          1          1        23
 2,048          2          2          1          1        15
```

In this example, Report 1 and Report 2 display the sizes of the MQCs for processor 2 and processor 1 respectively. Both reports also display other nonvariable elements such as MQC SIZE and TABLE SIZE.

Changes in the STEALS variable from one report to the next indicate that your processor is performing normally.

The percentages in the CONTROL HITS and DATA HITS columns are variable and sometimes change in subsequent DYNAMIC reports. However, numbers at or near 100 percent indicate optimal system conditions.

NSAA Option

Use the NSAA option to display voluntary rendezvous opportunities (VRO) and Inappropriate I/O Buffer Access counters and reintegration status.

When you enter the command PEEK NSAA on a system running H-series or J-series RVU, an output similar to this example is generated:

```
> PEEK NSAA
PEEK - T9050H02 - (01MAY05) - (17MAY05)     SYSTEM \HALF4
(C)1981 TANDEM (C)2004 HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.
                              SYSTEM   \HALF4
18 MAY 2005, 15:47___ELAPSD    1:04:13___CPU  0(NSE-P)
VRO:      INFREQUENT VROs   PROCESS WITH INFR VROs        INSERTED VROs HIT
                         0                        0                        0
INAPPACC: ACCESS SUSPENDS   PROCESS WITH ACCSSUSPD
                         0                        0
REINTEGRATION STATUS:  NO CURRENT OPERATION
BLADE A:     SUCCESSREINT     LASTSUCCREINTTIME
CPU 0            1                   0:00:10.918
CPU 1            1                   0:00:11.108

BLADE B:     SUCCESSREINT     LASTSUCCREINTTIME
CPU 0            0                   0:00:00.000
CPU 1            0                   0:00:00.000

BLADE C:     SUCCESSREINT     LASTSUCCREINTTIME
CPU 0            1                   0:00:11.443
CPU 1           2                    0:00:11.267
```

Elements of the NSAA Display

Table 2-6. NSAA Elements Reported in the NSAA Display (page 1 of 2)

Element	Description
VRO	Header identifying the voluntary rendezvous opportunities (VRO) display.
INFREQUENT VROs	Number of infrequent VRO events
PROCESS WITH INFR VROs	Number of processes contributing to the infrequent VRO events
INSERTED VROs	Number of times inserted VROs hit
INAPPACC	Header identifying the Inappropriate I/O Buffer Access events

Table 2-6. NSAA Elements Reported in the NSAA Display (page 2 of 2)

Element	Description
ACCESS SUSPENDS	Number of data access rights faults causing process suspend events
PROCESS WITH ACCSSUSPD	Number of processes contributing to data access rights faults
REINTEGRATION STATUS	Header identifying the reintegration status
NO CURRENT OPERATION	No reintegration in progress
CURRENTLY ONGOING	Reintegration in progress
BLADE A	Header identifying Blade A of a NonStop Blade Complex
SUCCESSREINT	Number of successful reintegrations on a processor element (PE) in a Blade
LASTSUCCREINTTIME	Time taken for last successful reintegration on a processor element (PE) in a Blade
BLADE B	Header identifying Blade B of a NonStop Blade Complex
BLADE C	Header identifying Blade C of a NonStop Blade Complex
CPU *n*	CPU number *n* in the Blade of a NonStop Blade Complex

PAGING Option

Use the PAGING option to monitor processor activity for paging statistics. The default report appears when you run the command PEEK PAGING with no options.

When you enter PEEK PAGING on a system running G-series RVU, an output similar to this example is generated:

```
> PEEK PAGING
PEEK - T9050G09 - (05AUG02)     SYSTEM \TOMMY
COPYRIGHT TANDEM COMPUTERS INCORPORATED 1981, 1985, 1988, 1990

                          SYSTEM   \TOMMY
26 FEB 2004, 15:26___ELAPSD  23:10:29___CPU  0(NSR-T)
PAGES: PHYSCL SWAPBL  FREE FREEMIN FREEQTA FREERED  UNDUMPED
(16Kb) 16384  16181   8884       7      10       5        0

PAGES:       LOCKED       LOCKED(KSEG0)
(16 Kb)   3725/14159       3725/28495

             FAULTS    ALLOCS   DISKREADS DISKWRITES  MUTEXCRAX NONMUTEXCRAX
TOTAL          7378      9156        2117           5        961        65983
(per sec)      0.01      0.02        0.00        0.00       0.00         0.16

             REDHIT   REDBUSY   REDTASK
TOTAL             0         0         0
(per sec)      0.00      0.00      0.00

CLEANQ: FULLS  FRLST:HITS  CLOCK:CALLS   FAILS    CYCLES   ALIASES:   FAILS
     0      0        4347         6053       0      1.17          0        0
```

When you enter PEEK /CPU 0/ PAGING on a system running H-series or J-series RVU, an output similar to this example is generated:

```
>PEEK /CPU 0/ PAGING

PEEK - T9050H02 - (01MAY05) - (31MAY05)     SYSTEM \HALF3

(C)1981 TANDEM (C)2004 HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.


                              SYSTEM   \HALF3

8 JUN 2005 , 12:36___ELAPSD   16:11:42___CPU  0(NSE-P)

PAGES:  PHYSCL  SWAPBL   FREE   FREEMIN FREEQTA FREERED  UNDUMPED

(16Kb)  262144  262085  227869  227840     160      80         0

PAGES:      LOCKED       LOCKED(KSEG0)

(16Kb)    25574/229325    2869/12244

              FAULTS     ALLOCS  DISKREADS DISKWRITES  MUTEXCRAX NONMUTEXCRAX

TOTAL          15446      30496       6691         25       1377         3001

(per sec)       0.26       0.52       0.11       0.00       0.02         0.05

              REDHIT    REDBUSY    REDTASK

TOTAL              0          0          0

(per sec)       0.00       0.00       0.00

CLEANQ:   FULLS  FRLST:HITS  CLOCK:CALLS    FAILS    CYCLES   ALIASES:  FAILS

     0        0       29824          673        0      2.35         1       0
```

Elements of the PAGING Display

Table 2-7. PAGING Elements Reported in the PAGING Display (page 1 of 3)

Element	Description
PAGES	Reports these paging statistics: ● PHYSCL — physical memory size in pages. The number in parentheses below the PAGES heading indicates the size of a page in kilobytes. ● SWAPBL — number of pages not permanently required by the operating system and thus available for swapping. ● FREE — number of swappable pages not currently assigned to any process. ● FREEMIN — minimum number of pages on the free list. ● FREEQUOTA — number of pages reserved for allocation under mutex. ● FREERED — number of free pages reserved for allocation under mutex when the memory management semaphore was not available. ● LOCKED — number of pages currently locked and maximum number of swappable pages that can be locked in the processor by various processes at any given time. The current maximum is 7/8 of the total number of swappable pages in the processor being measured. This maximum prevents all of the memory in the processor from being locked at the same time, thus preventing deadlocks.

Table 2-7. PAGING Elements Reported in the PAGING Display (page 2 of 3)

Element	Description
PAGES (continued)	● LOCKED (KSEG0) — number of pages currently locked in Kseg 0 and maximum number of swappable pages in Kseg 0 that can be locked by various processes at any given time. The current maximum is 7/8 of the total number of swappable pages in Kseg 0. This maximum prevents all of the memory in Kseg 0 from being locked at the same time, thus preventing deadlocks. ● FAULTS — number of page faults that have occurred. ● ALLOCS — number of requests for the allocation of a page. ● DISKREADS— number of times a page was read in from disk. ● DISKWRITES — number of times a page was written to disk. ● MUTEXCRAX — number of times the CRAX (Convert Relative to Absolute Extended) instruction was executed under mutex. ● NONMUTEXCRAX—number of times the CRAX instruction was executed, not under mutex. ● REDHIT — number of times a page had to be allocated under mutex, and the number of free pages was less than the FREERED counter. ● REDBUSY — number of times a page had to be allocated under mutex, and the number of free pages was less than the FREERED counter, but the memory management semaphore was not available. ● REDTASK — number of times a page had to be allocated under mutex, the number of free pages was less than the FREERED counter, and the memory management semaphore was available, but some other memory management requirement could not be met. ● (per sec) — average number per second for each of these elements: FAULTS, ALLOCS, DISKREADS, DISKWRITES, MUTEXCRAX, NONMUTEXCRAX, REDHIT, REDBUSY, REDTASK.
CLEANQ	Reports these statistics for the clean queue (the list of pages to be cleaned): ● CLEANQ—number of pages in the clean queue. ● FULLS—number of times the clean queue was full, preventing a dirty page from being queued.
FRLST	Reports these statistics for the free list: ● HITS—number of times a request for allocation of a page was satisfied from the free list of pages, without having to make a clock call.

Table 2-7. PAGING Elements Reported in the PAGING Display (page 3 of 3)

Element	Description
CLOCK	Reports these statistics about the algorithm for the page frame selection of the memory manager, which identifies and reallocates the oldest allocated but unused page: ● CALLS — number of times the algorithm was invoked to obtain a page frame. ● FAILS — number of times the algorithm failed to find a replaceable page. ● CYCLES — total number of times the entire swappable page set has been searched.
ALIASES	Reports these statistics for addressing of unaliased selectable segments: ● ALIASES—number of times an absolute address was created from a literal address for an unaliased selectable segment. This value should normally be 0. ● FAILS—number of times an attempt to create an absolute address from a literal address for an unaliased selectable segment failed. This value should normally be 0.

Examples

1. This example displays paging statistics for processor 2 since it was last loaded:

```
> PEEK / CPU 2 / PAGING
PEEK - T9050G09 - (05AUG02)     SYSTEM \TOMMY
COPYRIGHT TANDEM COMPUTERS INCORPORATED 1981, 1985, 1988, 1990

                              SYSTEM   \TOMMY
26 FEB 2004, 15:19___ELAPSD   21:31:27___CPU  2(NSR-T)

PAGES: PHYSCL SWAPBL FREE FREEMIN   FREEQTA FREERED   UNDUMPED
(16Kb)  16384  16225 12632      9        10       5          0

PAGES:       LOCKED       LOCKED(KSEG0)
(16 Kb)   3725/14159       3725/28495

         FAULTS  ALLOCS  DISKREADS  DISKWRITES  MUTEXCRAX  NONMUTEXCRAX
TOTAL      1157    4497        928           0        254        3118
(per sec)  0.01    0.05       0.01        0.00       0.00        0.04

          REDHIT  REDBUSY  REDTASK
TOTAL          0        0        0
(per sec)   0.00     0.00     0.00

CLEANQ: FULLS FRLST:HITS CLOCK:CALLS  FAILS  CYCLES    ALIASES:   FAILS
     0      0       2422        3705      0    0.31          0        0
```

2. This example displays paging statistics for processor 2 since it was last loaded and for the 10-second period since Report 1:

```
> PEEK / CPU 2 / PAGING, 2, 10, DYNAMIC
```

Report 1:

```
PEEK - T9050G09 - (05AUG02)     SYSTEM \TOMMY
COPYRIGHT TANDEM COMPUTERS INCORPORATED 1981, 1985, 1988, 1990

                              SYSTEM   \TOMMY
26 FEB 2004, 15:26___ELAPSD  23:10:29___CPU  0(NSR-T)  __SAMP  1/2,DELAY  10

PAGES: PHYSCL SWAPBL  FREE FREEMIN   FREEQTA   FREERED  UNDUMPED
(16Kb)  16384  16181  8356       7        10         5         0

PAGES:       LOCKED       LOCKED(KSEG0)
(16 Kb)   3725/14159       3725/28495

              FAULTS  ALLOCS  DISKREADS  DISKWRITES  MUTEXCRAX  NONMUTEXCRAX
TOTAL           7601    9433       2331           5        824         17622
(per sec)       0.09    0.11       0.02        0.00       0.00          0.21

              REDHIT  REDBUSY  REDTASK
TOTAL              0        0        0
(per sec)       0.00     0.00     0.00

CLEANQ: FULLS   FRLST:HITS  CLOCK:CALLS  FAILS   CYCLES    ALIASES:   FAILS
     0      0         4407         6340      0     0.51           0        0
```

Report 2:

```
                              SYSTEM   \TOMMY
26 FEB 2004, 15:26___ELAPSD    0:00:10___CPU  0(NSR-T)  __SAMP  2/2,DELAY  10

PAGES: PHYSCL SWAPBL  FREE FREEMIN FREEQUOTA FREERED  LOCKED    LOCKED(KSEG0)
(16Kb)  16384  16181  8356       7        10       5  3666/14159   3666/28495

              FAULTS  ALLOCS  DISKREADS  DISKWRITES  MUTEXCRAX  NONMUTEXCRAX
TOTAL              0       0          0           0          0             2
(per sec)       0.00    0.00       0.00         0.0       0.00          0.20

              REDHIT  REDBUSY  REDTASK
TOTAL              0        0        0
(per sec        0.0      0.0     0.00

CLEANQ: FULLS   FRLST:HITS  CLOCK:CALLS  FAILS   CYCLES    ALIASES:   FAILS
     0      0            0            0      0     0.00           0        0
```

POOL Option

The POOL option reports on the state of system tables and resources. When you run the command PEEK POOL with no options, the default report appears.

When you enter the PEEK /CPU 2/ POOL command on the system running G-series RVUs, an output similar to this example is generated:

```
> PEEK POOL
PEEK - T9050G09 - (05AUG02)     SYSTEM \TAHOE
COPYRIGHT TANDEM COMPUTERS INCORPORATED 1981, 1985, 1988, 1990

                              SYSTEM   \TAHOE
26 FEB 2004, 15:59___ELAPSD 736:10:29___CPU 2(NSR-T)

          MAXIMUM USED  CURRENT USAGE  # CONFIGURED  # OF FAILURES
TLE              67             64           600              0
PCB     57:       1     54:       1     255:   244      0:      0
NRL S             40             38         15102              0
    M              0              0             0              0
PPL S             39             37         13796              0
    M              0              0             0              0
PTLE               0              0             0              0

          MAX.SIZE CUR.SIZE  INIT.CNF  MAX.USED CUR.USED  MAX.FRAG CUR.FRAG
SYSPOOL        368      368     21609       368      368         0        0
EXTPOOL        218        0     65536       218        0         0        0
MAPPOOL      76520    68458    786374     76488    68048         7        3
FLEXPOOL    524242   524242    524242     30492    22396         9        8
SEG TBL      16352      981     16352      4096      943      3115        4

                       FLEXPOOL SUBPOOL USAGE

MAXIMUM      CURRENT      ALLOCATED      DEALLOCATED
             1            1          1             0
```

When you enter the command PEEK /CPU 0/ POOL on a system running H06.20 and later H-series RVUs or J06.09 or later J-series RVUs, an output similar to this example is generated:

```
PEEK - T9050H02 - (01AUG10) - (25FEB10) - (ATY)     SYSTEM \HALF6
(C)1981 TANDEM (C)2004-2008 HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.

                              SYSTEM   \HALF6
3 MAR 2010 ,  3:48___ELAPSD    6:34:06___CPU  0(NSE-D/NS14000)

          MAXIMUM USED  CURRENT USAGE  # CONFIGURED  # OF FAILURES
TLE             13             12          20000             0
PCB      6:     42      6:     40    255:   7830      0:      0
NRL            443            227          32767             0
PTLE             1              1            681             0
PME              6              2          65501             0

            MAX.SIZE  CUR.SIZE  INIT.CNF  MAX.USED  CUR.USED MAX.FRAG CUR.FRAG
SYSPOOL         5520      5520     13249      5520      5520        0        0
EXTPOOL            0         0    262144         0         0        0        0
MAPPOOL        70096     70096    196585     69744     67368       31       17
FLEXPOOL     3145498   3145498   2097106   3006424   2995400       26       21
SEG TBL        13591     13591     13591      1984      1984    10233       20

                         FLEXPOOL SUBPOOL USAGE

         MAXIMUM          CURRENT       ALLOCATED     DEALLOCATED
            5                5              5              0

POOL64 USAGE          TOTAL SIZE          ALLOCATED   LOCKED/WIRED    SEGMENTS
------------     ------------------     ----------   ------------    ---------
                 INIT    CUR    MAX     CUR    MAX     CUR    MAX     CUR  MAX
MAPPOOL64        64MB   64MB   64MB  3108KB 3388KB      -      -      1     1
FLEXPOOL64     2048MB 2048MB 2048MB  18752B 22768B  32768B 32768B     1     1
```

Note. For a system running J06.06 or earlier J-series RVUs or H06.17 or earlier H-series RVUs, NRL is displayed as a four line output with the small-index (S) and medium-index (M) entries for NRL and PPL on separate lines. This is compressed to a single line NRL output from H06.18 series and J06.07 series onwards.

For systems running J06.07 and later J-series RVUs or H06.18 and later H-series RVUs, the PME table information is included in the output.

For systems running J06.09 and later J-series RVUs or H06.20 and later H-series RVUs, the FLEXPOOL64 information is also included in the output.

Elements of the POOL Display

Table 2-8. POOL Elements Reported in the POOL Display (page 1 of 3)

Elements	Description*
TLE, PCB, NRL, PPL, PTLE and PME	Report the state of these system tables: ● Time-list elements (TLEs) ● Process control blocks (PCBs) ● Named resource lists (NRLs) ● Process-pair lists (PPLs) ● Process time-list elements (PTLEs) ● POSIX mapping entry table (PME) **Note:** Starting with the G06.23 RVU, NRL and PPL displays show small-index (S) and medium-index (M) entries on separate lines, as the system now provides full support for the DCT limits extension.
MAXIMUM USED	Maximum number of TLEs, PCBs, NRL and PPL entries, PTLEs, and PMEs that were ever used. For PCBs, the first value (*nn*:) reports low-PIN processes. The second value reports high-PIN processes.
CURRENT USAGE	Number of TLEs, PCBs, NRL and PPL entries, PTLEs and PMEs that are currently in use. For PCBs, the first value (*nn*) reports low-PIN processes. The second value reports high-PIN processes.
# CONFIGURED	Number of TLEs, PCBs, and NRL and PPL entries that were configured, and the number of PTLEs and PMEs that were dynamically configured. The number of TLEs is automatically configured and is constant. Starting with the G06.23 RVU, the Subsystem Control Facility (SCF) can be used to extend the DCT limit from 32,767 to 65,376. The default limit for the DCT is 32,767 entries, but the system reserves 400 entries for internal use. Therefore, the effective limit for DCT is 32,367 entries. For PCBs, the first value (*nn*) reports low-PIN processes. The second value reports high-PIN processes.
# OF FAILURES	Number of allocation failures that occurred for the TLEs, PCBs, NRL and PPL entries, PTLEs, and PMEs. For PCBs, the first value (*nn*) reports low-PIN processes. The second value reports high-PIN processes.
SYSPOOL and EXTPOOL	Report the state of the fixed-size system storage pools.
MAX.SIZE	Largest amount of pool space allocated since system load.
CUR.SIZE	Amount of pool space currently allocated.
INIT.CNF	Configured amount of pool space.

* For definitions of the elements listed here, see the Glossary.

Table 2-8. POOL Elements Reported in the POOL Display (page 2 of 3)

Elements	Description*
MAX.USED	Largest amount of pool space ever used.
CUR.USED	Amount of pool space currently being used.
MAX.FRAG	The highest value of CUR.FRAG since system load.
CUR.FRAG	The number of free space elements that are fragments.
MAPPOOL	Reports the state of MAPPOOL, a system storage pool of variable size.
MAX.SIZE	The largest amount of pool space allocated since system load.
CUR.SIZE	The amount of pool space currently being used.
INIT.CNF	Configured amount of pool space. (MAX.SIZE can exceed INIT. CNF.)
MAX.USED	The largest amount of pool space ever used.
CUR.USED	Amount of pool space currently being used.
MAX.FRAG	Highest value of CUR.FRAG since system load.
CUR.FRAG	Number of free space elements that are fragments.
FLEXPOOL	Reports the state of FLEXPOOL, a system storage pool of variable size.
MAX.SIZE	Total number of bytes that were available in all subpools when the pool contained the largest number of subpools since system load.
CUR.SIZE	Total number of bytes available in all subpools currently allocated.
INIT.CNF	The total virtual space allocated to the pool at system initialization. (MAX.SIZE can exceed INIT. CNF.)
MAX.USED	The largest amount of pool space ever used.
CUR.USED	The amount of pool space currently being used.
MAX.FRAG	Highest value of CUR.FRAG since system load.
CUR.FRAG	Number of free space elements that are fragments.
SEG TBL	Report the state of the segment table for aliased virtual memory segments.
MAX.SIZE	Largest number of segments allocated since system load.
CUR.SIZE	Number of segments currently allocated.
INIT.CNF	Configured number of segments.
MAX.USED	Largest number of segments ever used.
CUR.USED	Number of segments currently being used.
MAX.FRAG	Largest contiguous block of unused segments.
CUR.FRAG	Number of segments currently available.

* For definitions of the elements listed here, see the Glossary.

Table 2-8. POOL Elements Reported in the POOL Display (page 3 of 3)

Elements	Description*
FLEXPOOL SUBPOOL USAGE	Reports the state of the FLEXPOOL subpools.
MAXIMUM	The largest number of subpools to be simultaneously allocated since system load.
CURRENT	The number of subpools currently being used.
ALLOCATED	The total number of subpools allocated since system load.
DEALLOCATED	The total number of subpools deallocated since system load.
MAPPOOL64	Reports the state of MAPPOOL64, a system storage pool of variable size.
TOTAL SIZE	The largest amount of pool space allocated since system load, the amount of pool space currently being used, and the configured amount of pool space.
ALLOCATED	The largest amount of pool space ever used and the amount of pool space currently being used.
LOCKED/WIRED	The amount of pool space that is currently locked.
SEGMENTS	The number of segments allocated using POOL64.
FLEXPOOL64	Reports the state of FLEXPOOL64, a system storage pool of variable size.
TOTAL SIZE	The total number of bytes that were available in all subpools when the pool contained the largest number of subpools since system load, total number of bytes available in all subpools currently allocated, and the total virtual space allocated to the pool at system initialization
ALLOCATED	The largest amount of pool space ever used and the amount of pool space currently being used.
LOCKED/WIRED	The amount of pool space that is currently locked.
SEGMENTS	The number of segments allocated using POOL64.

* For definitions of the elements listed here, see the Glossary.

Considerations

- Because the pool sizes are dynamic, PEEK displays both maximum and current sizes.
- For SYSP00L, EXTP00L, and MAPP00L, all counts are in words.
- For FLEXPOOL, all counts are in bytes.
- The maximum size of the SYSP00L system storage pool for TNS/R (HP Tandem NonStop Series/ RISC) processors and itanium processor can exceed the size configured during system generation. Under the heading MAX.SIZE, the current value reported for this pool can appear quite large.

- The PTLE table has no entries when the processor first comes up after a system load or reload. Entries are added dynamically on a demand basis.
- TLE allocation failures are rare but can occur under certain conditions. Up to 100 TLEs are required for system processes. When TLEs are allocated, system processes are favored over nonsystem processes. If the number of free TLEs drops to 100 or less, nonsystem processes are not allowed to allocate TLEs. TLE allocation failures are indicated in the # OF FAILURES column for TLEs.

  To help identify the source of TLE allocation failures, subtract the MAXIMUM USED total from the # CONFIGURED total:

  - If the difference is between 1 and 100, nonsystem processes might have had TLE allocation failures.
  - If the difference is zero, even system processes might have had TLE allocation failures.
  - If the difference is more than 100, but close to that number, you might want to take preventive action to avoid TLE allocation failures.

  To prevent TLE allocation failures, try to reduce the use of TLEs in the processor. For example, look for processes that call SIGNALTIMEOUT for many different timers in use simultaneously. On systems running G-series and H-series RVUs, the number of TLEs is automatically set to the maximum possible and cannot be changed.

Example

This example displays POOL statistics for processor 2:

```
> PEEK / CPU 2 / POOL
PEEK - T9050H02 - (01AUG10) - (25FEB10) - (ATY)     SYSTEM \TOMMI

(C)1981 TANDEM (C)2004-2008 HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.


                              SYSTEM   \TOMMI

3 MAR 2010 ,  3:48___ELAPSD    6:34:06___CPU  2(NSE-D/NS14000)


          MAXIMUM USED  CURRENT USAGE  # CONFIGURED  # OF FAILURES

TLE             13             12          20000             0
PCB      6:     42      6:     40    255:   7830      0:      0
NRL            443            227          32767             0
PTLE             1              1            681             0
PME              6              2          65501             0


          MAX.SIZE  CUR.SIZE  INIT.CNF  MAX.USED  CUR.USED  MAX.FRAG CUR.FRAG

SYSPOOL       5520      5520     13249      5520      5520         0        0
EXTPOOL          0         0    262144         0         0         0        0
MAPPOOL      70096     70096    196585     69744     67368        31       17
```

```
FLEXPOOL   3145498   3145498   2097106   3006424   2995400       26       21
SEG TBL      13591     13591     13591      1984      1984    10233       20


                          FLEXPOOL SUBPOOL USAGE

         MAXIMUM          CURRENT        ALLOCATED     DEALLOCATED

               5                5                5               0

POOL64 USAGE          TOTAL SIZE           ALLOCATED   LOCKED/WIRED    SEGMENTS
------------     ------------------     ----------   ------------    --------
                 INIT    CUR     MAX     CUR    MAX     CUR    MAX    CUR  MAX
MAPPOOL64         64MB   64MB    64MB  3108KB 3388KB       -      -      1    1
FLEXPOOL64      2048MB 2048MB  2048MB  18752B 22768B  32768B 32768B      1    1
```

Note. For systems running J06.06 or earlier J-series RVUs and H06.17 or earlier H-series RVUs, NRL is displayed as a four-line output with the small-index (S) and medium-index (M) entries for NRL and PPL on separate lines. This four-line output is compressed to a single-line NRL output from J06.07/H06.18 series onwards.

For systems running J06.07 and later J-series RVUs or H06.18 and later H-series RVUs, the PME table information is included in the output.

For systems running J06.09 and later J-series RVUs or H06.20 and later H-series RVUs, the FLEXPOOL64 information is also included in the output.

TIME Option

The TIME option displays the amount of time a given processor has spent on processes, interrupts, and idle time. TIME is one of the default reports displayed when you run PEEK with no options specified.

When you enter the command PEEK TIME, an output similar to this example is generated:

```
> PEEK TIME
PEEK - T9050G09 - (05AUG02)     SYSTEM \VIOLET
COPYRIGHT TANDEM COMPUTERS INCORPORATED 1981, 1985, 1988, 1990

                               SYSTEM   \VIOLET
22 FEB 2004, 15:57___ELAPSD  736:09:17___CPU  4(NSR-T)

TIME:    PROCESSBUSY TIME       INTERRUPT TIME         IDLE TIME
       109:06:38.295  14.82%   25:37:15.202   3.48%  601:25:23.566  81.69%
```

When you enter the command PEEK /CPU 0/ TIME on a system running H-series or J-series RVU, an output similar to this example is generated:

```
PEEK - T9050H02 - (01MAY05) - (31MAY05)     SYSTEM \HALF3
(C)1981 TANDEM (C)2004 HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.

                               SYSTEM   \HALF3
8 JUN 2005 , 12:41___ELAPSD   16:16:32___CPU  0(NSE-P)

TIME:     PROCESSBUSY TIME        INTERRUPT TIME          IDLE TIME
          0:09:58.482   1.02%     0:00:39.380   0.06%    16:05:54.468  98.91%
```

Elements of the TIME Display

Table 2-9. TIME Elements Reported in the TIME Display

Element	Description
PROCESSBUSY TIME	Elapsed time and percentage of total time the processor spent executing processes (in *hours:minutes:seconds*)
INTERRUPT TIME	Elapsed time and percentage of total time the processor has been busy with interrupts (in *hours:minutes:seconds*)
IDLE TIME	Elapsed time and percentage of total time the processor spent idle (in *hours:minutes:seconds*)

Examples

1. This example displays the amount and percentage of time processor 2 has spent on processes and interrupts and the amount and percentage of idle time since the processor was last loaded:

```
> PEEK / CPU 2 / TIME
PEEK - T9050G09 - (05AUG02)     SYSTEM \VIOLET
COPYRIGHT TANDEM COMPUTERS INCORPORATED 1981, 1985, 1988, 1990

                     SYSTEM   \VIOLET
22 FEB 2004, 16:24___ELAPSD  832:35:40___CPU  2(NSR-T)

TIME:     PROCESSBUSY TIME         INTERRUPT TIME          IDLE TIME
         88:08:12.538  10.58%    20:46:18.792   2.49%   723:41:08.875  86.91%
```

   Processor 2 has had this activity during the 832 hours since it was last loaded:

   - It has been busy with processes 10.58 percent of the time.
   - It has spent 2.49 percent of its time on interrupts.
   - It has been idle 86.91 percent of the time.

2. This example displays TIME statistics for processor 2 since it was last loaded and for the ten seconds after Report 1 is generated:

```
> PEEK / CPU 2 / TIME, 2, 10, DYNAMIC
```

Report 1:

```
PEEK - T9050G09 - (05AUG02)     SYSTEM \VIOLET
COPYRIGHT TANDEM COMPUTERS INCORPORATED 1981, 1985, 1988, 1990

                               SYSTEM   \VIOLET
22 FEB 2004, 12:48__ELAPSD 360:51:24__CPU  2(NSR-T)__SAMP 1/2,DELAY 10

TIME:  PROCESSBUSY TIME    INTERRUPT TIME        IDLE TIME
      40:15:43.006 11.15%  8:26:55.183  2.34%  312:08:46.055  86.50%
```

Report 2:

```
                              SYSTEM   \VIOLET
22 FEB 2004, 12:48__ELAPSD 0:00:10__CPU 2(NSR-T) __SAMP 2/2,DELAY 10

TIME:  PROCESSBUSY TIME     INTERRUPT TIME        IDLE TIME
       0:00:00.897  8.74%   0:00:00.269  2.63%   0:00:09.095  88.62%
```

   Note the differences between these elements of Reports 1 and 2:

   - PROCESSBUSY TIME: Since it was last loaded, processor 2 has been busy with processes 11.15 percent of its time. In the 10-second interval shown in Report 2, this processor has been busy with processes 8.74 percent of its time.
   - INTERRUPT TIME: Since it was last loaded, processor 2 has been busy with interrupts 2.34 percent of its time. In the 10-second interval shown in Report 2, this processor has been busy with interrupts 2.63 percent of its time.

- IDLE TIME: Since it was last loaded, processor 2 has been idle 86.50 percent of its time. In the 10-second interval shown in Report 2, this processor has been idle 88.62 percent of its time.

If this example did not specify `DYNAMIC`, both reports would show statistics since the processor was last loaded.

Glossary

This glossary defines technical terms related to PEEK, to the internal design of the operating system, and to the system architecture for the NonStop server.

active process. The process that is currently using the instruction processing unit (IPU) of a processor. Contrast with inactive process.

API. See application program interface (API).

application program interface (API). A set of services (such as programming language functions or procedures) that are called by an application program to communicate with other software components. For example, an application program in the form of a client might use an API to communicate with a server program.

ARITHOV. The PEEK INTERRUPTS element that reports arithmetic overflow traps. See also trap.

backup path. A path not enabled as the preferred path. A backup path can become a primary path when a primary path is disabled. Also called *alternate path.* Contrast with primary path.

BKPT. The PEEK INTERRUPTS element that reports instruction breakpoint interrupts. See also interrupt.

breakpoint. An object code location at which execution will be suspended so that you can interactively examine and modify the process state. With symbolic debuggers, breakpoints are usually at source line or statement boundaries.

In native object code for TNS/R or (H-series RVUs only) TNS/E, breakpoints can be at any MIPS RISC instruction or (H-series RVUs only) Itanium instruction within a statement. In a TNS object file that has not been accelerated, breakpoints can be at any TNS instruction location. In a TNS object file that has been accelerated, breakpoints can be only at certain TNS instruction locations, not at arbitrary instructions. Some source statement boundaries are not available. However, breakpoints can be placed at any instruction in the accelerated code.

BUS. This element reports interprocessor communication (IPC) traffic interrupts. See also interrupt.

byte. Eight bits.

CCL. Mnemonic for the TNS instruction Condition Code Less Than.

CISC. See complex instruction-set computing (CISC).

CISC processor. An instruction processing unit (IPU) that is based on complex instruction-set computing (CISC) architecture. Contrast with RISC processor.

clock. See processor clock or system clock.

clock averaging algorithm. An algorithm used by the operating system to keep the processor clocks in a system synchronized.

CME. (1) See correctable memory error (CME). (2) The PEEK INTERRUPTS element that reports CME interrupts. See also interrupt.

complex instruction-set computing (CISC). A processor architecture based on a large instruction set, characterized by numerous addressing modes, multicycle machine instructions, and many special-purpose instructions. Contrast with reduced instruction-set computing (RISC).

control packets. Sequenced message transmissions.

core services. The portion of the operating system that consists of the low-level functions, including interprocess communications; I/O interface procedures; and memory, time, and process management. Contrast with system services.

correctable memory error (CME). A single-bit error in an addressable memory location, for which the operating system compensates. Contrast with uncorrectable memory error (UCME).

current priority. The priority of a process at this time.

destination control table (DCT). The NonStop operating system data structure that holds information about every device and named process in the system. The DCT consists of the named resource list (NRL) and the process-pair list (PPL). The DCT is replicated in each processor.

dirty pages. Frames (physical pages) of memory that have been changed since they were mapped. If the frame has a swap file, the frame must be written to the swap file before the memory manager can make the frame available to another process.

disk page. A unit of virtual storage. In TNS, TNS/R, and TNS/E systems, a disk page contains 2048 bytes. Contrast with memory page.

DISP. The PEEK INTERRUPTS element that reports dispatch interrupts. See also interrupt.

dispatcher. An interrupt handler that sends interprocessor messages, manages semaphores, calculates process execution time, and changes the active process.

dispatching. The task of making a process active.

element. A data structure consisting of a header immediately followed by data.

extended data segment. An area of virtual memory used to contain data. An extended data segment is allocated with contiguous addresses and is treated programmatically as a single object. The two types of extended data segments are selectable segments

and flat segments. Extended data segments are allocated by the ALLOCATESEGMENT or SEGMENT_ALLOCATE_ Guardian procedure.

extensible data segment. An extended data segment for which swap file extents are not allocated until needed.

EXTPOOL. (1) A fixed-size system storage pool. (2) The PEEK POOL element that reports the size of EXTPOOL.

fault address. Part of an absolute extended address that specifies the logical page to be swapped into physical memory by the memory manager.

file system. A set of operating system procedures and data structures that provides for communication between a process and a file, which can be a disk file, a device other than disk, or another process.

FLEXPOOL. (1) A system storage pool of variable size. (2) The PEEK POOL element that reports the size of FLEXPOOL.

FLEXPOOL64. (1) A system storage pool of variable size. (2) The PEEK POOL element that reports the size of FLEXPOOL64.

frame. The smallest unit of memory that the memory manager handles (allocates or deallocates) at one time. The size of a frame varies by system. On most NonStop servers, the frame size is 16,384 bytes (16 Kilobytes.) Also called *physical page*.

free pages. The swappable pages in a system that are not assigned to any process.

free space. The available space in a memory pool.

general input/output interrupt handler. An interrupt handler provided by the operating system that is invoked when a physical input/output operation finishes. The general input/output interrupt handler can optionally invoke a special input/output interrupt handler.

Guardian. An environment available for interactive or programmatic use with the NonStop operating system. Processes that run in the Guardian environment use the Guardian system procedure calls as their application program interface. Interactive users of the Guardian environment use the HP Tandem Advanced Command Language (TACL) or other command interpreter.

hard CME. A correctable memory error (CME) that occurred during consecutive access attempts to a specific memory location. See also correctable memory error (CME) and soft CME.

high PIN. A PIN in the range 256 or higher. See also process identification number (PIN) and low PIN.

HP Integrity NonStop NS-series servers. The HP Integrity NonStop servers having product numbers beginning with the letters NS. These servers implement the ServerNet architecture and run the NonStop operating system.

HP NonStop Blade Complex (NSBC). The set of one, two, or three NonStop Blade Elements and their associated LSUs. For the first release of Integrity NonStop servers, a NonStop Blade Complex contains one to four logical processors. Also called a Blade Complex.

HP NonStop operating system. The operating system for NonStop servers.

idle process. A special process that executes when no other process is able to execute.

IFAIL. The PEEK INTERRUPTS element that reports instruction failure traps. See also trap.

I/O. See input/output (I/O).

IIO. Mnemonic for the TNS instruction Interrogate I/O. This element reports all I/O traffic interrupts. See also interrupt.

IIO CCLs. The PEEK INTERRUPTS element that reports IIO CCL interrupts. See also CCL, IIO, and interrupt.

inactive process. A process that is not currently using the instruction processing unit (IPU) of a processor. Contrast with active process.

input/output (I/O). Data entered into a computer or transmitted out of a computer. (2) The process of entering data into or transmitting data out of a computer.

input/output process (IOP). A running program (part of the NonStop operating system) that manages the I/O functions for one or more ServerNet addressable controllers (SACs) of the same type.

interprocess communication. The exchange of messages between processes in a system or network.

interrupt. The mechanism by which a processor is notified of an asynchronous event that requires immediate processing.

interrupt environment. The software environment that exists when a processor is executing instructions in response to an interrupt.

interrupt handler. A procedure that is invoked by special interrupt firmware when certain events occur in a processor.

interrupt threshold. The maximum number of interrupts that can occur before the operating system begins to contain the errors.

IOP. See input/output process (IOP).

IOS. I/O subsystem.

Itanium Processor. The processor used in the Integrity NonStop server. Contrast with CISC processor and RISC processor.

LDEV . See logical device.

locked pages. The pages in a processor that are not available for swapping because they are currently assigned or are reserved for use by the operating system.

logical device. (1) A process that can be accessed as if it were an I/O device; for example, the operator process is logical device LDEVOPR. (2) An addressable device, independent of its physical environment. Portions of the same logical device can be located in different physical devices, or several logical devices or parts of logical devices can be located in one physical device. (3) The logical device number (LDEV) or the logical I/O address for (1) or (2). See also logical I/O address.

logical I/O address. A 32-bit value that input/output processes (IOPs) use to refer to a unit in the input/output configuration of a processor.

logical memory. The portion of virtual memory that can be accessed by a process in nonprivileged mode.

logical page. See memory page.

low PIN. A PIN in the range 0 through 254. See also process identification number (PIN) and high PIN.

MAB. The PEEK INTERRUPTS element that reports memory access breakpoint interrupts. See also interrupt.

MAPPOOL. (1) A system storage pool. (2) The PEEK POOL element that reports MAPPOOL statistics.

MAPPOOL64. (1) A system storage pool. (2) The PEEK POOL element that reports MAPPOOL64 statistics.

memory manager. A system process that manages physical memory in a processor.

memory page. A unit of virtual storage. In TNS systems, a memory page contains 2048 bytes. In TNS/R systems, the page size is determined by the memory manager and can vary, depending on the processor type. In TNS/R and TNS/E systems, a memory page contains 16,384 bytes. Contrast with disk page.

memory pool. A shared memory area in which allocation is dynamic and temporary.

message quick cell (MQC). A data structure that the message system quickly obtains and uses to perform interprocess communication. The system automatically builds and allocates MQCs as it needs them.

message system. A set of operating system procedures and data structures that handles the mechanics of exchanging messages between processes, which might be running in the same processor or different processors.

millicode. The system's lowest-level machine-dependent code, often coded in assembler language. TNS/R millicode and TNS/E millicode are functionally similar to the microcode on TNS systems. The system has several types of millicode, including machine interrupt handlers, operating system primitives, routines implicitly called from native-compiled code, emulators for TNS floating-point arithmetic, and emulators for privileged-only or long-running TNS machine operations.

module. (1) A physical grouping of procedures and data structures. (2) For NonStop S-series servers, a set of components sharing a common interconnection, such as a backplane. A module is a subset of a group and is usually contained in an enclosure. There is one module in a group. For NonStop NS-series servers, a module is a set of function components and is nominally a single fault zone.

MQC. See message quick cell (MQC).

MQC finder table. A table containing message quick cell (MQC) information. See also message quick cell (MQC).

mutex. A synchronization object that provides mutual exclusion among threads. A mutex is often used to ensure that shared variables are always seen by other threads in a consistent state.

named process. A process to which a process name was assigned when the process was created. Contrast with unnamed process.

named resource list (NRL). An operating system data structure that contains information about the characteristics of named processes and logical devices in the system. See also NRL table.

NonStop S-series servers. The set of NonStop servers having product numbers beginning with the letter S (such as S70000). These servers run the NonStop operating system and implement the ServerNet architecture.

NRL. See named resource list (NRL).

NRL table. A table containing named resource list (NRL) entries. See also named resource list (NRL).

object file. A file generated by a compiler, Binder program, or native link editor (`nld`) utility that contains machine instructions and other information needed to construct the executable code spaces and initial data for a process. The file might be a complete program that is ready for immediate execution, or it might be incomplete and require linking with other object files before execution.

page fault. A reference to a logical page that is not currently in physical memory. Such a reference causes an interrupt, and the interrupt handler begins a sequence of operations that loads the page into memory.

paging. A method of managing virtual memory.

path. The route between a processor and a subsystem. If a subsystem is configured for fault tolerance, it has a primary path (from the primary processor) and a backup path (from the backup processor).

PCB. See process control block (PCB).

PCB table. An operating system data structure that contains information about the resources and environment of processes in a processor. See also process control block (PCB).

PFAIL. The PEEK INTERRUPTS element that reports power-fail interrupts. See also interrupt.

physical location. A set of values that describes the location of a component within an enclosure. The physical location is composed of the group number, module number, and slot number.

physical memory. The semiconductor dynamic random-access memory (DRAM) that is part of every processor. Physical memory is the hardware resource that lies underneath virtual memory. Code and data in physical memory is immediately accessible without the delay of reading from disk.

physical page. See frame.

PIN. See process identification number (PIN).

PME table. A table containing POSIX mapping entries (PME).

PON. The PEEK INTERRUPTS element that reports power-on interrupts. See also interrupt.

pool. See memory pool.

PPL. See process-pair list (PPL).

PPL table. A table containing process pair list (PPL) entries. See also process-pair list (PPL).

prepaging. On TNS processors, a technique that involves transferring extra pages to reduce the number of reduce the number of page faults. Prepaging does not occur on TNS/R and TNS/E processors.

primary path. A path enabled as the preferred path. When a primary path is disabled, an alternate (backup) path becomes the primary path. Contrast with backup path.

process. A program that has been submitted to the operating system for execution, or a program that is currently running in the computer. See also system process and user process.

process control block (PCB). An element of the operating system that monitors and controls the resources and environment of the processes in a processor. See also PCB table.

process environment. The state and contents of the code and data spaces, stacks, and register values that exist when the processor is executing instructions that are part of a user process or system process.

process identification number (PIN). An unsigned integer that identifies a process in a processor. Internally, a PIN is used as an index into the process control block (PCB) table.

process message queue. A linked list of messages and notifications that have been sent to a process. A process has several process message queues, one for each type of request or notification. The headers for most of these linked lists reside in the process control block (PCB) of the process.

process name. A name that can be assigned to a process when the process is created. A process name uniquely identifies a process or process pair in a system.

process-pair list (PPL). An operating system data structure that contains information about the relationships between named processes and logical devices in the system. See also PPL table.

process time. The amount of time that a process has spent in the active substate. (GPG)

process time list. A linked list of process time-list elements (PTLEs) used to manage process time. See also process time-list element (PTLE).

process time-list element (PTLE). An operating-system data structure that can be queued on the process time list to manage process time. See also process time list.

process timer. A clock that measures process execution time.

processor clock. A hardware timer on each processor that keeps processor time (the number of microseconds since system load).

PTLE. See process time-list element (PTLE).

ready processes. Processes that are prepared to become active.

reduced instruction-set computing (RISC). A processor architecture based on a relatively small and simple instruction set, a large number of general-purpose registers, and an optimized instruction pipeline that supports high-performance instruction execution. All

TNS/R processors use the RISC architecture. Contrast with complex instruction-set computing (CISC).

relative extended address. An address that can be used when the processor is in privileged or nonprivileged mode to access the user code, user library, and user data spaces of the process. A relative extended address can also be used in privileged mode to access the system code, system library, and system data spaces of the process. A relative extended address cannot access extended memory.

requester. A process that initiates interprocess communication by sending a message to another process. Contrast with server.

resident cache segment. A type of absolute segment with which no swap file is associated. To use a frame occupied by a logical page of a resident cache segment, the system must first ask permission of the segment owner if the page has been changed while in memory.

resident segment. A type of absolute segment with which no swap file is associated. A logical page in a resident segment must be locked before it can be accessed and must remain locked while it is used.

RISC. See reduced instruction-set computing (RISC).

RISC processor. An instruction processing unit (IPU) that is based on reduced instruction-set computing (RISC) architecture. All TNS/R processors, such as the NSR-G and NSR-W processors, use RISC processors. Contrast with CISC processor.

SAMPLE. The PEEK INTERRUPTS element that reports Measure sampling interrupts. See also trap.

SCHANL. The PEEK INTERRUPTS element that reports special channel error interrupts. See also interrupt.

server. (1) An implementation of a system used as a stand-alone system or as a node in a network. (2) A combination of hardware and software designed to provide services in response to requests received from clients across a network. For example, the NonStop range of servers provides transaction processing, database access, and other services. (3) A process or program that provides services to a client or a requester. Servers are designed to receive request messages from clients or requesters; perform the desired operations, such as database inquiries or updates, security verifications, numerical calculations, or data routing to other computers systems; and return reply messages to the clients or requesters. A server process is a running instance of a server program.

ServerNet system area network (ServerNet SAN). A low-cost, high-speed network, contained in a server, that connects processors to each other and to ServerNet addressable controllers (SACs).

soft CME. A correctable memory error (CME) that occurred on the initial access to a specific memory location, but did not occur during the second access to the same memory location. A soft CME that occurs in the same location with excessive frequency can be reclassified as a hard CME. See also correctable memory error (CME) and hard CME.

special input/output interrupt handler. An interrupt handler, provided and enabled by an input/output process (IOP) developer, that is invoked by the general input/output interrupt handler and that performs processing specific to the IOP.

special interrupt microcode. Microcode that is executed when an interrupt is detected.

STKOV. The PEEK INTERRUPTS option that reports stack overflow traps. See also trap.

swap files. The disk copy of a file that is currently a part of virtual memory. Pages of the file are swapped back and forth between physical memory and disk as they are needed.

swappable pages. The pages in a processor that are not permanently required by the operating system and are thus available for swapping to processes needing them.

swapping. The process of copying information between physical memory and disk storage.

SYSPOOL. (1) A system storage pool. (2) The PEEK POOL element that reports on SYSPOOL.

system. A node. All the hardware, firmware, and software components that are directly connected to form an entity that is managed by one operating system image and operated as one computer.

system clock. A clock consisting of the interval timer and a field in the system globals area that together represent the current local civil time for a system.

system process. A process whose primary purpose is to manage system resources rather than to solve a user's problem. Failure of a system process often causes the processor to fail. Most system processes are automatically created when the processor is system loaded. Contrast with user process.

system services. The tasks performed on behalf of the user or user programs by the operating system, including formatting, process control, I/O support, performance measurement, process-pair support, standard security, and transaction management. Contrast with core services.

system time. The time represented by any synchronized processor clock in the system.

system-load environment. The software environment that exists before the operating system is fully loaded and operational. Also called *cold-load environment*.

TIME. The PEEK INTERRUPTS element that reports time interrupts. See also interrupt.

time list. A linked list of time-list elements (TLEs) that are waiting for the passing of time.

time-list element (TLE). An operating system data structure that can be queued on the time list to manage real time. Time-list elements are also called *time-list control blocks*.

time-list interrupt handler. An interrupt handler that manages the time list.

timekeeping. A function performed by the operating system that involves initializing and maintaining the correct time in a processor.

timing. A function performed by the operating system that involves controlling when certain events occur within a processor.

TLE. See time-list element (TLE).

TNS. HP computers that support the NonStop operating system and that are based on complex instruction-set computing (CISC) technology. The term TNS can refer to the instruction set, the architecture, or the processors. The NonStop Cyclone system is an example of a TNS system. Contrast with TNS/R.

TNS/E. (H-series RVUs only) Refers to fault-tolerant HP computers that support the NonStop operating system and are based on the Itanium processor. TNS/E systems run the Itanium instruction set and can run TNS object files by interpretation or after acceleration. TNS/E systems include all Integrity NonStop systems that use NSE-x processors. Contrast with TNS and TNS/R.

TNS/R. HP computers that support the NonStop operating system and that are based on reduced instruction-set computing (RISC) technology. TNS/R processors implement the TNS/R instruction set and maintain architectural compatibility with TNS processors. The term TNS/R can refer to the instruction set, the architecture, or the processors. Most NonStop K-series servers and all NonStop S-series servers use TNS/R processors. Contrast with TNS.

trap. A system state similar to that caused by an interrupt but synchronous to the system rather than asynchronous as in the case of an interrupt. The PEEK INTERRUPTS option reports these types of traps: instruction failure traps (IFAIL), arithmetic overflow traps (ARITHOV), stack overflow traps (STKOV), and Measure sampling interrupts (SAMPLE).

UCME. (1) See uncorrectable memory error (UCME). (2) The PEEK INTERRUPTS element that reports UCME interrupts. See also interrupt.

uncorrectable memory error (UCME). A multiple-bit error in an addressable memory location. The operating system cannot compensate for such errors. Contrast with correctable memory error (CME).

unlocking memory. The task of allowing previously locked logical pages to be stolen by the memory manager.

unnamed process. A process to which a process name was not assigned when the process was created. Contrast with named process.

unsequenced packets. One-packet messages, such as “I’m alive” and message acknowledgments.

user process. A process whose primary purpose is to solve a user's problem. Contrast with system process.

virtual memory. A range of addresses that processes use to reference a memory storage space that can be considerably larger than physical memory. The system maps such references onto physical memory, transferring the contents of the addressed locations as necessary between physical memory and some mass-storage medium.

virtual page. See memory page.

waiting process. A process that cannot execute until an event occurs, a resource becomes available, or an interval of time passes

Index

H

I

L

M

N

O

P

R

S

T

U

X

Y

Special Characters

Content Feedback

First Name: _____ Last Name: _____
Phone: _____ e-mail address: _____
Company: _____

(All contact information fields are required.)

If you're reporting an error or omission, is your issue:

- ☐ **Minor**: I can continue to work, but eventual resolution is requested.
- ☐ **Major**: I can continue to work, but prompt resolution is requested.
- ☐ **Critical**: I cannot continue to work without immediate response.

Comments (give sufficient detail to help us locate the text):

_____
_____
_____
_____
_____

Thank you for taking the time to provide us with your comments.

You can submit this form online, e-mail it as an attachment to pubs.comments@hp.com, fax it to 408-285-5520, or mail it to:

Hewlett-Packard Company
NonStop Enterprise Division
19333 Vallco Parkway, MS 4421
Cupertino, CA 95014-2599
Attn.: Product Manager, Software Publications